

ESCUELA POLITÉCNICA NACIONAL

FACULTAD DE INGENIERÍA DE SISTEMAS

**MODELO DE ESTRATEGIAS DE REPRODUCIBILIDAD PARA
EXPERIMENTOS EN INGENIERÍA DE SOFTWARE**

PROYECTO PREVIO A LA OBTENCIÓN DEL TÍTULO DE DOCTOR EN INFORMÁTICA

CARLOS EDUARDO ANCHUNDIA VALENCIA

carlos.anchundia@epn.edu.ec

DIRECTOR: DR. EFRAÍN RODRIGO FONSECA CARRERA

erfonseca@espe.edu.ec

CODIRECTORA: DRA. SANDRA SANCHEZ-GORDON

sandra.sanchez@epn.edu.ec

Quito, febrero 2023



ESCUELA
POLITÉCNICA
NACIONAL

TESIS

Para la obtención del título de

DOCTOR EN FILOSOFÍA IN INFORMÁTICA

Resolution RPC-SO-43-No.501-2014 del Consejo de Educación Superior

Presentado por

**CARLOS EDUARDO
ANCHUNDIA VALENCIA**

Tesis dirigida por **Efraín Rodrigo Fonseca Carrera, Profesor of the Universidad de las Fuerzas Armadas - ESPE** y co-dirigida por **Sandra Sanchez-Gordon, Profesora de la Escuela Politécnica Nacional**

MODELO DE ESTRATEGIAS DE REPRODUCIBILIDAD PARA EXPERIMENTOS EN INGENIERÍA DE SOFTWARE

Examen oral presentado ante la siguiente comisión:

Pamela Catherine Flores Naranjo, Ph.D.
EPN, Presidente

Carlos Efraín Iñiguez Jarrín, Ph.D.
EPN, Tribunal interno

Silvia Teresita Acuña Castillo, Ph.D.
UAM, Tribunal externo

Oscar Dieste Tubio, Ph.D.
UPM, Tribunal externo

Nombre, Ph.D.
EPN, Tribunal interno

DECLARACIÓN

Declaro bajo juramento que soy el autor de este trabajo, el cual no ha sido presentado previamente para la obtención de ningún título académico o título profesional. También declaro que he consultado las referencias bibliográficas incluidas en este documento.

Mediante esta declaración, transfiero mis derechos de propiedad intelectual correspondientes a la presente tesis, a favor de la Escuela Politécnica Nacional, según lo establece la Ley de Propiedad Intelectual del Ecuador, su Reglamento y las normas institucionales vigentes.

Declaro que este trabajo se basa en los siguientes artículos de mi autoría (como autor principal o coautor) relacionados con el título de esta tesis:

- Anchundia, C.E. *The replication of experiments in software engineering, a dilemma associated with knowledge generation*. ClbSE 2017 - XX Ibero-American Conference on Software Engineering, 2017, pp. 34–41
- Anchundia, C.E., Fonseca C, E.R. *Resources for reproducibility of experiments in empirical software engineering: Topics derived from a secondary study*. IEEE Access, 2020, 8, pp. 8992–9004, 8951162
- Anchundia V., C.E., Sanchez-Gordon, S., Cuadros, A., Efraín Fonseca, C. *The paradox of reproducibility improvement in empirical research in software engineering: Thoughts derived from a case study*. RISTI - Revista Iberica de Sistemas e Tecnologias de Informacaotheris, 2020, 2020(E31), pp. 545–557

También declaro que he reconocido la colaboración de terceros, y el aporte realizado por otro material publicado o inédito.



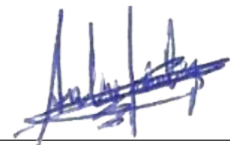
Carlos Eduardo Anchundia Valencia

CERTIFICACIÓN

Certificamos que CARLOS EDUARDO ANCHUNDIA VALENCIA ha realizado su investigación bajo nuestra supervisión. Hasta donde conocemos, las contribuciones de este trabajo son novedosas.



Dr. Efraín Rodrigo Fonseca Carrera
Director del proyecto



Dra. Sandra Sanchez-Gordon
Codirectora del proyecto

AGRADECIMIENTOS

“A mis padres, por el apoyo emocional e incondicional que hizo posible culminar una etapa más de crecimiento académico y personal”.

“A mi director de tesis y amigo Ph.D. Efraín Fonseca, por el invaluable aporte brindado desde el inicio hasta la culminación de este trabajo investigativo, sin el cual no hubiera sido posible lograr”.

“A la Ph.D. Sandra Sanchez-Gordon, codirectora de tesis, por la guía y apoyo en los procesos académicos necesarios para el desarrollo del proyecto”.

Al amigo Dr. Marco Benalcázar por el soporte recibido para este proceso y por su colaboración en todo momento”.

“A la Escuela Politécnica Nacional, entidad mentora que me facilitó los recursos humanos, económicos, y académicos para desarrollar el programa doctoral”.

“A la Universidad de las Fuerzas Armadas, que a través de mi director, me facilitó sus instalaciones y alumnos para llevar a cabo la validación del proyecto”.

“A los expertos y estudiantes que participaron en la validación”.

“A mis amigos y compañeros de labores: Ph.D. Pamela Flores y MSc. Patricio Zambrano, por estar siempre prestos a escuchar y debatir los cuestionamientos durante el desarrollo del presente trabajo, y haber aportado significativa y positivamente en el desarrollo de la investigación”.

“A mis amigos y colaboradores incondicionales: MSc. Andrés Fuseau, MSc. Igor Espín y MSc. Juan Pablo Balda, por el apoyo moral y emocional que fue un pilar fundamental en este emprendimiento”.

DEDICATORIA

“ A la Escuela Politécnica Nacional, Universidad que me acogió en el Programa Doctoral y me brindó la oportunidad de culminar con éxito el Proyecto”.

“A mis estudiantes, quienes son la fuente de motivación de mi superación académica, con la esperanza de ver en ellos reflejada la aplicación de esta investigación”.

Índice general

Resumen	XV
Abstract	XVI
Prólogo	XVII
1. Introducción	1
1.1. Planteamiento del problema sobre la reproducibilidad	1
1.1.1. Antecedentes	1
1.1.2. Contextualización del problema	3
1.1.3. Estado del arte de la problemática	7
1.2. Sistematización del problema	12
1.2.1. Contexto social	12
1.2.2. Contexto del fenómeno	16
1.3. Preguntas y objetivo del diseño de investigación	17
1.4. Metodología de investigación	19
2. Planteamiento de la propuesta: Modelo de reproducibilidad	23
2.1. Componente “Contexto”	24
2.2. Componente “Problema-Solución”	25
2.2.1. Necesidades	26
2.2.2. Descripción del problema	34
2.2.3. Conocimiento de dominio de la solución	37
2.2.4. Alternativas	38
2.2.5. Descripción de la solución	39
2.2.6. Artefacto	39

2.3. Componente “Control”	39
2.3.1. Modelo	40
2.3.2. Criterios de calidad	41
2.4. Preguntas de conocimiento para la propuesta	43
2.4.1. ¿Qué tan genérico o específico debe ser el modelo?	44
2.4.2. ¿Qué tan formalizado debe ser el modelo?	44
2.4.3. ¿El modelo debe ser monolítico o modular?	45
2.4.4. ¿Cómo se puede integrar las necesidades en el modelo?	45
2.4.5. ¿Es posible validar el modelo?	45
2.4.6. ¿Cuánto esfuerzo exige la aplicación del modelo?	46
2.4.7. ¿Cómo se incorpora la reproducibilidad en el modelo?	46
2.5. Definición de la propuesta de solución: Modelo de estrategias e intenciones .	46
2.5.1. Definición de la estructura del modelo	47
2.5.2. Estructura de nivel 1: Reproducibilidad	48
2.5.3. Estructura de nivel 2: Auto descripción	49
2.5.4. Estructura de nivel 2: Transferibilidad	50
2.5.5. Estructura de nivel 2: Estandarización	51
2.6. Supuestos relacionados al planteamiento	53
3. Evaluación de la propuesta: Experimento	58
3.1. Definición del experimento	59
3.1.1. Planteamiento del problema del experimento	59
3.1.2. Impacto en el contexto de experimentación	60
3.1.3. Objetivos del experimento	61
3.2. Planificación y diseño del experimento	62
3.2.1. Parámetros del experimento	62
3.2.2. Participantes	62
3.2.3. Preguntas e hipótesis del experimento	64
3.2.4. Métricas de evaluación	64
3.2.5. Muestras	66
3.2.6. Tipo de diseño de experimento	67
3.2.7. Recolección de datos	67

3.2.8. Instrumentos	68
3.3. Recolección de datos	68
3.3.1. Autorización para recolección de datos	70
3.3.2. Motivación de los participantes	70
3.3.3. Artefactos recolectados	71
3.3.4. Reporte de novedades	74
3.4. Análisis de datos	74
3.4.1. Sensibilidad de estudiantes al modelo	75
3.4.2. Expectativa de reproducibilidad de los expertos	76
3.4.3. Pruebas de hipótesis	79
3.5. Empaquetado de resultados	85
3.5.1. Esquemas de organización	85
3.5.2. Navegación	86
3.5.3. Acceso	86
3.6. Interpretación de resultados	87
3.6.1. El impacto del modelo en los usuarios	87
3.6.2. La perspectiva de los expertos	88
3.6.3. La efectividad del modelo	89
3.7. Amenazas a la validez	90
3.8. Discusión	93
3.8.1. El énfasis a la reproducibilidad de experimentos	93
3.8.2. El destino de las propuestas	93
3.8.3. La crisis de la reproducibilidad	94
4. Conclusiones y trabajos futuros	95
4.1. Lecciones aprendidas	95
4.2. Conclusiones	96
4.3. Aportaciones	99
4.4. Líneas de investigación futura	100
Bibliografía	101
Anexos	107

Anexo 1: Material de inducción	108
Anexo 2: Material de refuerzo	111
Anexo 3: Tarea sobre experimento	121
Anexo 6: Ejecución de experimento	146
Anexo 7: Apreciación de estudiantes	235
Anexo 8: Apreciación de expertos	241
Anexo 9: Resultados de revisión de documentos de Anexo 6	245
Anexo 10: Lista de estudios primarios	249
Anexo 11: Lista de estudios al 2022	253

Índice de figuras

1.1. Espectro de la reproducibilidad adaptado de [1]	2
1.2. Modelo de problema-solución [2]	5
1.3. Evolución del modelo de problema-solución [2]	6
1.4. Resumen de revisión de literatura	9
1.5. Red semántica sobre intercambio de conocimiento	11
1.6. Red semántica sobre comunicación	11
1.7. Red semántica sobre motivación	12
1.8. Red semántica sobre herramientas	13
1.9. Categorías de herramientas por especialización y formalidad	15
1.10. Categorización de herramientas	15
1.11. Causas, problema, efectos	17
1.12. Relación entre el contexto de problema y contexto de creación de solución según DS. [3]	21
2.1. Visión global sobre la propuesta	24
2.2. Red semántica sobre necesidades	30
2.3. Explicación de diagrama de intenciones con UML basado en [4]	41
2.4. Propuesta de factores de reproducibilidad	43
2.5. Relación entre los elementos PSM descritos	47
2.6. Nivel 1 de modelo	54
2.7. Nivel 2 de modelo para auto descripción	55
2.8. Nivel 2 de modelo para transferibilidad	56
2.9. Nivel 2 de modelo para estandarización	57
3.1. Diagrama de secuencia de proceso de experimentación aplicado [4]	59
3.2. Parámetros posibles	63

3.3. Pregunta del experimento	64
3.4. Hipótesis del experimento	64
3.5. Métricas	66
3.6. Parámetros posibles	85
3.7. Red grafo de bóveda de investigación	86
3.8. Diferencia entre utilidades percibidas	87
3.9. Relevancia de los factores de calidad	88
3.10. Relevancia de los factores de calidad	89
3.11. Relevancia de dos características de calidad	91

Índice de cuadros

1.1. Conceptos del proceso de investigación adaptado de [5]	8
1.2. Elementos de control	21
2.1. Citas sobre cooperación en red semántica "Necesidades"	26
2.2. Citas sobre contenido en red semántica "Necesidades"	28
2.3. Citas sobre validación en red semántica "Necesidades"	31
2.4. Citas sobre trazabilidad en red semántica "Necesidades"	32
2.5. Citas sobre motivación en red semántica "Necesidades"	33
2.6. Fragmentos identificados en red semántica "Problemas"	35
2.7. Propuesta de alternativas de solución vs. problemas	39
2.8. Niveles del modelo	48
3.1. Etapa de intervención	60
3.2. Estructuración del objetivo	61
3.3. Características de sujetos	65
3.4. Lista de sujetos	65
3.5. Características de revisores	65
3.6. Perfiles de revisores	69
3.7. Muestra	70
3.8. Características del diseño del experimento	70
3.9. Nomenclatura de artefactos	70
3.10. Lista de actividades de recolección y previas	71
3.11. Instrumentos generados para el experimento	72
3.12. Artefactos recolectados	72
3.13. Reporte de novedades	75
3.14. Protocolo de validación	75

Lista de códigos

3.1. Plantilla de detalle de actividad	69
3.2. Plantilla de detalle de novedad	77
3.3. Tratamiento del archivo CVS sobre percepción de estudiantes	77
3.4. Seguridad percibida por estudiantes con/sin herramienta	78
3.5. Utilidad de las herramientas percibida por estudiantes	78
3.6. Tratamiento del archivo CVS sobre opinión de expertos	80
3.7. Preparación de dataframes para análisis de datos	81
3.8. Preparación de datos para análisis	83
3.9. Preparación de datos para pruebas de hipótesis	84
3.10. Pruebas de hipótesis con Wilcoxon-test	84
3.11. Pruebas adicionales de características con diferencia significativa	90

Resumen

El presente trabajo explora la reproducibilidad de experimentos, el método más utilizado, en la ingeniería de software empírica. La reproducibilidad es un aspecto importante y controversial para la generación del conocimiento científico. El análisis de la reproducibilidad ayuda al campo de la investigación y a la generación de conocimiento confiable para la ingeniería de software basada en la evidencia. Diversos estudios han reportado una baja replicación de investigaciones, asociando este problema a la conocida “crisis de reproducibilidad”. Con esta consideración, se ha investigado este contexto a fin de mejorar la calidad de la experimentación entregando una propuesta de solución a las necesidades de la reproducibilidad.

Se empleó la metodología *Design Science* para la construcción de un artefacto. Con la ayuda revisiones sistemáticas, entrevistas y aplicaciones propias, se investigó la reproducibilidad en el contexto de la ingeniería de software empírica sistematizando el problema y necesidades. Se contempló el modelo de madurez *Problem-Solving Model* para determinar elementos presentes y ausentes en el proceso de experimentación. y así formular un modelo multinivel de intenciones y estrategias para describir el proceso de soporte basado de factores de calidad: la auto descripción, transferibilidad y la estandarización. A través de un experimento, que consideró la participación de estudiantes con nociones sobre la investigación en ingeniería de software y expertos investigadores, se pudo validar la efectividad de la propuesta.

Se realizaron varios análisis determinando que: (i) los estudiantes no perciben diferencias en la complejidad de usar la propuesta; (ii) los expertos tienden a valorar la relevancia de los factores de reproducibilidad de forma distinta cuando son aplicados; (iii) la efectividad de la propuesta no genera un cambio significativo; (iv) se encontró una diferencia en los factores de transferibilidad y estandarización, con una diferencia significativa en el uso de repositorios y descripción de empaquetado.

Palabras clave: Reproducibilidad, calidad, factores, estandarización, transferibilidad, auto descripción, ingeniería de software empírica.

Abstract

In this paper, we explored the reproducibility of experiments in empirical software engineering. Reproducibility has been an important and controvert aspect for scientific knowledge generation. Reproducibility analysis helps the research field and contributes to the generation of reliable knowledge for evidence-based software engineering. Several studies have reported low replication of research, associating this problem with the well-known “reproducibility crisis”. With this consideration, we investigated this context in order to improve the quality of the experimentation by delivering a solution proposal to the needs of reproducibility.

We use the Design Science methodology for the construction of an artifact. With the help of systematic reviews, interviews, and our own applications, we investigate reproducibility in the context of empirical software engineering systematizing the problem and needs. We contemplate the maturity model Problem-Solving Model to determine the presence and absent of elements within the experimentation process. We formulate a multilevel model of intentions and strategies to describe a supporting process based on quality factors: self-description, transferability, and standardization. Through an experiment, which considered the participation of students with notions about research in software engineering and expert researchers, we validated the effectiveness of the proposal.

We carried out several analyzes and determined that: (i) students do not perceive differences in the complexity of using the proposal; (ii) experts tend to assess the relevance of reproducibility factors differently when they are applied; (iii) the effectiveness of the proposal does not generate a significant change; (iv) we found a difference in portability and standardization factors, with a significant difference in the use of repositories and description of packaging.

Keywords: *Reproducibility, quality, factors, standardization, transferability, self-description, empirical software engineering.*

Prólogo

La replicación es un mecanismo fundamental para la generación de conocimiento científico. La ingeniería de software se vale del empirismo para mejorar el conocimiento de la disciplina, entre la que destaca el uso de la experimentación. A través de la experimentación es posible observar las relaciones entre distintas variables dependientes e independientes bajo un contexto en particular. Como resultado se puede poner a prueba tanto las tecnologías, metodologías, prácticas, teorías, y demás conocimiento en torno a la ingeniería de software. El auge de la experimentación aún no es alcanzado; sin embargo, en la actualidad existen grupos de investigación, comunidades y revistas especializadas enfocados a la experimentación y el empirismo.

Sin embargo, la tarea de conformación de conocimiento científico abarca también la revalidación y confirmación de resultados generalizables o especificables. Esta tarea sigue siendo un gran problema en el emprendimiento científico de la ingeniería de software empírica. La importancia de la replicación es reconocida por los investigadores pero pese a este hecho, la replicación representa un reto debido a la manera en que se gestiona de la información, la variedad terminológica existente, la informalidad que representa ciertas tareas dentro del proceso experimental, entre otras.

Esta tesis doctoral se enfoca en comprender los elementos circundantes en las actividades de replicación de experimentos en la ingeniería de software para elaborar una propuesta que incremente la madurez de esta actividad. Entre las principales contribuciones que el proceso doctoral puede aportar a la línea del empirismo en la ingeniería de software tenemos:

1. Sistematización del problema de reproducibilidad con enfoque a la gestión de conocimiento, motivaciones y comunicación
2. Categorización de herramientas por formalidad y especialización
3. Identificación de nivel de madurez de las actividades de experimentación y elementos presentes
4. Formulación del marco conceptual de factores de reproducibilidad
5. Elaboración de modelo estrategias e intenciones de reproducibilidad

Capítulo 1

Introducción

1.1. Planteamiento del problema sobre la reproducibilidad

1.1.1. Antecedentes

La experimentación es un método empírico que permite la evaluación de procesos, teorías y conocimiento de forma sistemática, disciplinada, cuantificable y controlada [6, 7]. En la ingeniería de software (IS), el experimento es una práctica ampliamente utilizada en el campo de la ingeniería de software empírica (ISE) [8]. A través de su ejecución, los investigadores pueden establecer las relaciones causa-efecto de variables dependientes e independientes en un contexto que puede ser manipulado o tratado. Como resultado, se espera que estos hallazgos, una vez validados, puedan ser transferidos a la industria, cerrando el ciclo entre las disciplinas de la ciencia y la ingeniería [9]. En la actualidad, la experimentación ha cobrado una creciente importancia dando lugar a la conformación de diversos grupos de investigación alrededor del mundo y al incremento de publicaciones relacionadas con el empirismo, desde su primera aplicación en los años 1960s [8, 10]. A pesar de esto, en la IS, en contraste con otras disciplinas, parece aún no haber alcanzado un nivel de madurez en lo que respecta a la conformación de conocimiento científico [11, 12].

En la actualidad, los indicadores sobre las prácticas de reproducibilidad parecen ser favorables debido a un incremento en estudios experimentales, sin embargo, existe un problema oculto: la experimentación por sí sola no permite una comprensión general de un fenómeno [5, 13, 7]. Para lograr esta comprensión se requiere de actividades de replicación, que permiten la reproducción de resultados, de análisis o de interpretaciones, sirviendo como complemento a la ejecución experimental [14]. Un estudio experimental puede tomar años, no solo por lo complejo de la investigación, sino también, por los subsecuentes estudios para abordar las especificaciones o generalizaciones de un problema. Este proceso, ejecutado muchas veces, permite generar conocimiento formal o conocimiento científico. Cuando este proceso se desarrolla de forma exitosa, se obtienen beneficios tanto

para la mejora de los procesos de producción científica, así como en fiabilidad o credibilidad de los hallazgos [15] a través de la exploración y comparación de variaciones y condiciones especiales entre diversos estudios relacionados. Esto a su vez se traduce en un facilitador de la aceptación de un estudio, siendo posible alcanzar un mayor impacto, en términos de citas y reconocimiento y en la reutilización de elementos de investigación [7]. En conclusión, la reproducibilidad es necesaria para la formalización del conocimiento a la vez que trae consigo beneficios en los procesos de investigación.

Sin embargo, propiciar la reproducibilidad de un estudio es difícil y costoso. Como sugiere Gundersen [14], *"la reproducibilidad es la habilidad de que un investigador independiente llegue a las mismas conclusiones de un experimento siguiendo la documentación compartida por los investigadores originales"*; el autor recalca la participación de investigadores independientes siguiendo la documentación disponible. Este énfasis destaca la importancia de la documentación y comunicación escrita para facilitar la reproducción o replicación de una investigación; siendo aquí donde radica la dificultad y costes de la reproducibilidad, ya que los procesos comunicativos deben considerar la diversidad terminológica, metodológica, tecnológica y operativa, así como el conocimiento tácito de los diferentes contextos de investigación [13]. Esta diversidad latente puede afectar la transparencia de los resultados, la trazabilidad de la información y la completitud de los protocolos de investigación ejecutados [15, 7]; incluso, puede derivarse de los trasfondos culturales o de malas prácticas de investigación [15, 16]. Como resultado, la reproducibilidad puede ser percibida como una escala gradiente (Espectro de reproducibilidad) donde la cantidad y calidad de datos compartidos pueden categorizar a un estudio desde "No reproducible" a "Estándar de oro"¹ como se muestra en la Figura 1.1 [7, 6].

En definitiva, cuando no se ejecutan los procesos de replicación, como parte del paradigma científico, no se puede contar con un apropiado conocimiento científico [17]. Por ejemplo, Fanelli [18] resume que en una encuesta publicada en la revista Nature, el 90% de los encuestados cree que existe algún tipo de crisis de reproducibilidad. Esta crisis está relacionada con la incapacidad de reproducir un estudio, ya sea por fraude o presión por publicación. No obstante, en la comunidad de ISE se reporta que el número de investigaciones relacionadas a la reproducibilidad se ha incrementado a lo largo del tiempo;

¹ Estándar de oro significa que se ha alcanzado 100% de reproducibilidad de un estudio

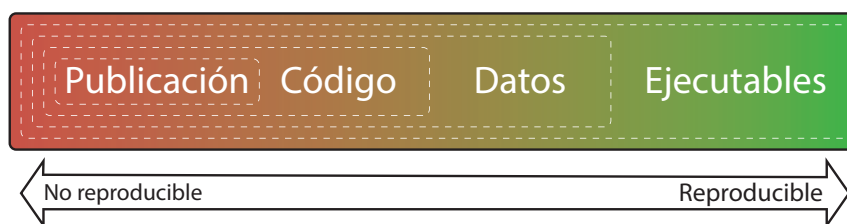


Figura 1.1: Espectro de la reproducibilidad adaptado de [1]

pero a un número reducido [10], debido a que aún no es una práctica establecida [12].

Entre las publicaciones sobre reproducibilidad se encuentran aportes para su ejecución que comprenden desde guías de reporte, hasta la creación de plataformas para la gestión de estudios empíricos como los experimentos. Pero, debido al número reducido de replicaciones reportadas [19] y de los esfuerzos realizados a la fecha [20], se puede deducir que falta un gran camino para propiciar la reproducibilidad en el campo de la ISE.

1.1.2. Contextualización del problema

Para abordar el problema de la reproducibilidad es necesario comprender la crisis y el espectro de reproducibilidad. La crisis de reproducibilidad hace referencia al problema metodológico del proceso de construcción de conocimiento científico; es decir, su verificación y validación. Por otro lado, el espectro de reproducibilidad trata de mostrar, a manera de gradiente, el nivel de acceso a la información de un estudio como habilitante de una investigación reproducible. Aunque estos conceptos son ampliamente abordados en el ámbito científico, quedan por explorar las posibles causas de la dificultad y costos inherentes a la reproducibilidad.

El espectro de la reproducibilidad denota que, a más información, más oportunidades de reproducibilidad; sin embargo, es posible enfocar la reproducibilidad a ciertos grados de acción de un estudio. Por ejemplo, un estudio replicado puede utilizar distintos datos, pero obtener las mismas conclusiones y aun así lograr un nivel alto de reproducibilidad. La reproducibilidad tiene tres grados de acción: los resultados reproducibles, el análisis reproducible, y la interpretación reproducible [14]. Los resultados, o salidas, son los elementos mayormente utilizados para los análisis comparativos; ya que, sin estos, ni el análisis ni la interpretación pueden ser conseguidos. En relación con el análisis, aunque los resultados discrepen, a través de la ejecución de los procesos analíticos, podrían obtenerse interpretaciones semejantes. Finalmente, la reproducibilidad de la interpretación permite que los investigadores externos puedan llegar a las mismas conclusiones. Al considerar estos grados de acción, un enfoque exclusivo a la reproducibilidad de resultados sería lo menos efectivo, pudiendo comprometer los esfuerzos de replicar estudios.

Bausell [21], en su libro *"The problem of science"*, resalta que pese a la atención que la reproducibilidad ha tenido, un estudio replicado es más difícil y menos atractivo de publicar que un estudio "original". El autor examina diversas fuentes de facilitadores que aportan a la no reproducibilidad, destacando prácticas cuestionables de investigación políticas institucionales como: (i) las decisiones sobre los elementos del diseño de investigación; (ii) las capacidades de tutoría dentro de sus grupos y entornos de investigación; (iv) los elementos culturales y disciplinares; (v) presiones grupales sociales y, (vi) lineamientos en las comunidades e instituciones de investigación.

Por otro lado, en [16] se menciona que los problemas de la reproducibilidad comúnmente son: (i) la informalidad sobre las discusiones; (ii) la brevedad de la descripción de resultados; (iii) código y resultados vagamente disponibles y, (iv) la falta de evaluación de la transparencia de los datos y código. La fuente de los problemas de reproducibilidad proviene principalmente por la dificultad de acceso a la información original, y a otros factores que contribuyen, como: manejo del proceso inadecuado de mantención de registros; reportes no transparentes; obsolescencia de artefactos digitales; intentos infructuosos al momento de reproducir investigaciones de otros; barreras en la cultura de investigación.

Como parte del proceso de investigación, se abordó una revisión básica de literatura [22] en la que se exploró el dilema asociado a la generación del conocimiento. En este estudio se exploraron los aspectos que impulsan escenarios de investigación reproducible en la ISE. Se encontró que la disciplina padece de un grado de inmadurez que no permite consolidar prácticas y herramientas que propicien los atributos de reproducibilidad, y aún menos, seguir rigurosamente los procedimientos. De esta manera se podría argumentar que existen aspectos comunicacionales y administrativos, así como motivaciones operativas y personales que decrementan las oportunidades de reproducir un estudio. Se pueden ejemplificar estos factores como: la poca homogeneidad de los lenguajes y términos [23]; empleo de prácticas informales o *ad-hoc* [24]; falta de estandarización de estructuras de información y organización [25]; falta de protocolos de investigación con un alto enfoque a la ejecución de replicaciones internas [12]; dificultad para la transferencia tecnológica o de conocimiento [26].

Siguiendo el hilo del abordaje de la reproducibilidad como una práctica ampliamente difundida, es posible dilucidar el nivel de madurez de ISE a través de una perspectiva denominada *Problem-Solving Model* (PSM). El PSM establece un dominio de conocimiento o contexto en el cual intervienen dos componentes principales: un componente de problema-solución, y un componente de control [27, 2]. Estos dos componentes, dentro de un contexto dado, reflejan cuán maduro es un dominio al observar lo definido de sus componentes. Dentro del componente de problema-solución se determinan: necesidades, descripción de problemas, dominios de la solución, conocimiento, alternativas, descripción de la solución y artefactos. Dentro del componente de control se determinan: las representaciones de un problema (modelos), criterios de calidad, condicionantes, y mecanismos de evaluación (ver Figura 1.2). Mientras más detallados, conocidos, o descritos sean estos componentes y sus detalles, más madura puede considerarse una disciplina. Por lo tanto, al revisar la literatura sobre las posibles propuestas que apuntan a la mejora de la reproducibilidad, se puede considerar que no existe un nivel de madurez adecuado que permita catalogar las propuestas de solución bajo el marco de un dominio o contexto de investigación.

La madurez del proceso experimental, entonces, puede ubicarse en uno de cuatro

niveles evolutivos de madurez (ver Figura 1.3). El primer nivel de madurez es aquel donde una necesidad o problema directamente produce una única solución o artefacto. Esto quiere decir que las soluciones no son transferibles ni recurrentes; es decir, existe un mapeo directo entre el problema y una solución. En el segundo nivel de madurez, una necesidad puede traducirse a un problema descrito, por lo cual también es posible describir una solución que puede ser implementada en un artefacto; es decir, la descripción de un problema explícito permite el diseño de una solución. En el tercer nivel de madurez, la descripción de un problema está relacionada a un dominio de conocimiento de soluciones, por lo cual es posible contar con diferentes alternativas que puedan describir una solución; por lo que, la aplicación de una solución corresponde a un conocimiento de un dominio y la generación de alternativas. Finalmente, en el cuarto nivel es posible analizar las alternativas a través de algún modelo matemático de evaluación que permita seleccionar la posible mejor alternativa; es decir, se aplican mecanismos de control y evaluación para la selección de alternativas apropiadas.

En el caso de los procesos empíricos, como la experimentación, se podría argumentar que el nivel de la disciplina de la ISE carece de madurez procedimental, ya que no existe un proceso para catalogar las diversas alternativas de solución o propuestas, y aún menos establecer métricas heurísticas que permitan la selección de la mejor alternativa. Esto se hace evidente cuando se explora las distintas soluciones que se proponen en la comunidad científica. De hecho, existe un vasto desarrollo de métodos, procedimientos y evaluaciones que no logran formalizarse y usarse de manera extensiva. En el estudio que se aborda en en [20] se realiza una revisión sobre una propuesta enfocada a la mejora de la reproducibilidad

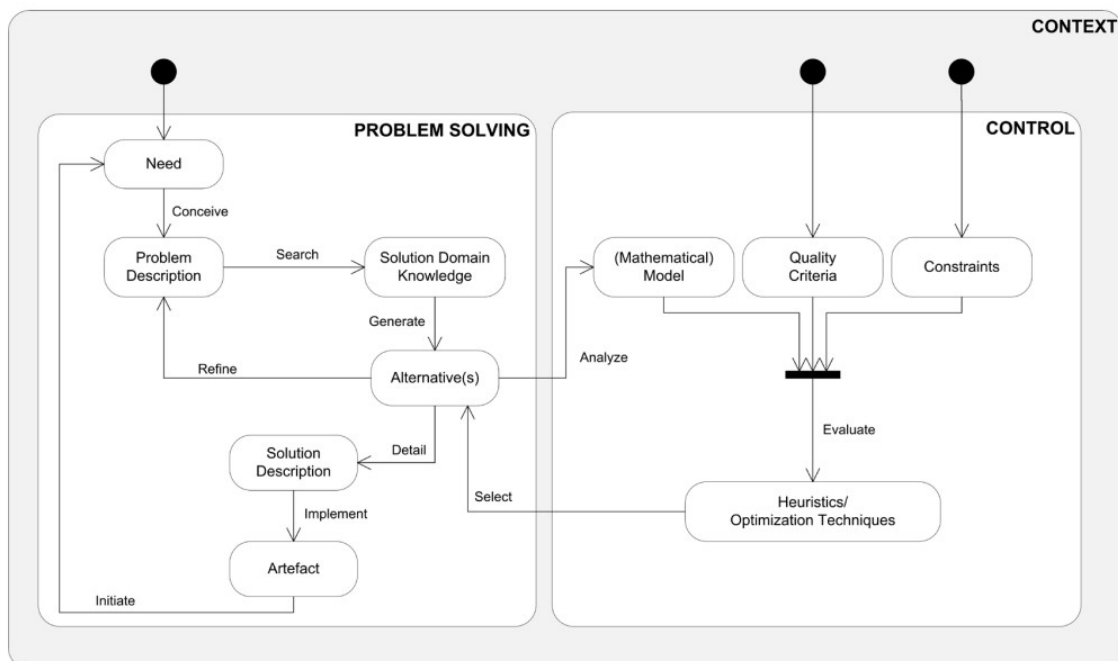


Figura 1.2: Modelo de problema-solución [2]

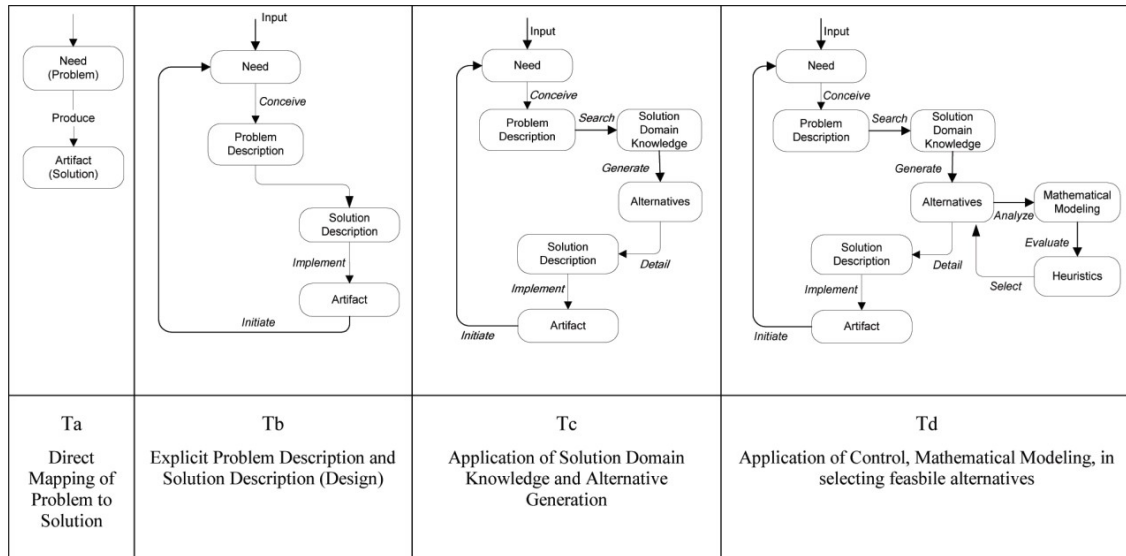


Figura 1.3: Evolución del modelo de problema-solución [2]

de una estrategia de investigación. En este estudio, el enfoque apunta a identificar puntos sensibles en los que la reproducibilidad podría ser mejorada. En primera instancia, la propuesta parece mejorar la comunicación de la información y de procesos de toma de decisiones; pero a la vez, parece que agrega más confusión terminológica y aumenta el costo de los procesos para la gestión de la investigación. Finalmente, la propuesta parece no contribuir directa o explícitamente a la formalización o estandarización de la comunicación o a la trazabilidad de las actividades.

A través de lo expuesto, se puede argumentar que el problema de la reproducibilidad es aún un tema que sigue generando necesidades. Existen importantes estudios para entender las causas y efectos dañinos de diversas prácticas contraproducentes a la replicación. En este sentido se han propuesto soluciones que posiblemente causan un resultado opuesto, esto es generando un efecto “bola de nieve”² sobre las tareas de investigación. Así analizado, el contexto del problema de la reproducibilidad aborda aspectos controlables y no controlables por los investigadores. Posiblemente, dado el nivel de madurez al que cataloga a la ISE, es necesario abordar y formalizar primeramente los aspectos controlables como: las estrategias para transparentar y difundir la información; las estrategias para la gestión del conocimiento y su transferencia; y, las estrategias de cambio cultural y motivacional de la comunidad de investigación.

²EL efecto bola de nieve se refiere a la metáfora de círculo vicioso y que, en este caso, agrega más y más complejidad a cada proceso.

1.1.3. Estado del arte de la problemática

En la literatura es posible encontrar y explorar una gran cantidad de aportes a la mejora de la reproducibilidad, tanto de experimentos como de otros métodos de investigación respecto a la disciplina de la ingeniería de software. Por ejemplo, muchos artículos ahondan en propuestas de distintos mecanismos para mejorar los procesos de replicación como guías de reportes. Otros artículos realizan evaluaciones de estas propuestas en distintos contextos y realizan recomendaciones. Finalmente, también es posible encontrar investigaciones que aplican propuestas tecnológicas y marcos de trabajo como FIRE o ARREST. Al igual que otros procesos de investigación empírica, los procesos de replicación muestran un incremento de producción científica cada año [12]. Sin embargo, en la disciplina del software las prácticas de replicación aún no están totalmente establecidas. De hecho, Zhang et al. [10] reportan que, aunque los investigadores prestan atención a la posibilidad de replicación, la ejecución de estos procesos sigue siendo escasa.

La replicación en la ISE sigue el mismo proceso evolutivo de una replicación general, persiguiendo la necesidad de contestar las siguientes interrogantes [16]: ¿Cómo se puede generar enfoques de programación que generen conocimiento confiable?, ¿Cómo se puede comunicar hipótesis y resultados para que otros lo confirmen, refuten o reúsen?, ¿Como se pueden entender, identificar y exponer sesgos para mejorar la precisión en la generación e interpretación de resultados?, ¿Puede ser posible identificar y eliminar el fraude y las mal interpretaciones intencionales?. Para esto, en las diferentes disciplinas se fueron incorporando ciertas prácticas para tratar de responder estas preguntas. En áreas como la psicología se comenzó a incluir una perspectiva estadística en los años 60s. En los años 70s, en la biomedicina se incorporaron técnicas enfocadas en las amenazas a la validez. En los años 90s se lanzó el movimiento abocado a la investigación reproducible, apoyándose en el creciente uso de flujos de trabajo computacionales. En los 2000's y 2010's los esfuerzos se enfocaron en la disponibilidad de los datos y los códigos como elementos necesarios para habilitar la reproducibilidad, así como la creación de grupos de investigación y revistas científicas especializadas.

En este sentido, la transparencia y la claridad se han convertido en aspectos importantes de estudio en esta década. Las principales actividades o herramientas usadas para compartir información en los métodos de replicación en ISE recogidas por [10] son: paquetes de replicación disponibles en línea; disponibilidad de listas de cuestionarios, entrevistas y artículos; sitios web con datos; recursos de replicación de estudios previos; disponibilidad de proveer recursos de replicación cuando se soliciten. En contraposición a la disponibilidad de los datos, cuyo enfoque se centra en la implementación, Perry [28] enfatiza que los datos, por si solos, no son suficientes para habilitar la reproducibilidad. Es necesario también entender cómo los estos se relacionan con las preguntas de investigación y su interpretación. Becker [5] realizó un análisis de reproducibilidad sobre los datos reportados

Cuadro 1.1: Conceptos del proceso de investigación adaptado de [5]

Concepto	Tipo de información	Nivel de descripción
Fuente de datos	Meta datos de las fuentes	Mayormente parcial
Metodología de captura	Procesos de recolección	Mayormente ausente
Datos sin procesar	Datos de entrada	Ausente
Metodología de extracción	Procesos de tratamiento	Ausente/Parcial
Estudio de parámetros	Criterios de selección	Ausente/Parcial
Datos procesados	Datos salida	Mayormente parcial
Metodología de análisis	Procedimientos de análisis	Completo
Conjunto de datos resultantes	Resultados procesables	Completo/Parcial

a través de paquetes de replicación en el marco de ocho conceptos del proceso de investigación (ver Cuadro 1.1). En su estudio, se denota que el tipo de información suministrada se enfoca mayoritariamente a reportar el proceso de análisis, mientras que las fases que contemplan la preparación de datos previo al análisis son pobremente descritas. Esta práctica puede ser contraproducente para la formalización del conocimiento ya que no reportar los datos iniciales y sus procesos de tratamiento no permitiría la evaluación de la contribución (análisis e interpretación reproducible), sino solamente la comparación de los resultados [29] afectando la credibilidad (validez, transparencia, ética, reflexibilidad y colaboración) del conocimiento mismo [17].

A más de considerar el uso de herramientas de soporte, también es necesario determinar qué prácticas se encuentran inmersas en los trabajos publicados. En esta exploración se realizó en un estudio [30], donde se consideró 40 estudios primarios (Anexo 10) a partir del análisis de 2600 estudios candidatos siguiendo los lineamientos para revisiones sistemáticas de literatura propuestos por Kitchenham [31] cuyo proceso se explica en la Figura 1.4. En este estudio se profundizó sobre los mecanismos de comunicación, la gestión del conocimiento, y las formas explícitas de motivación para replicación reportadas por los investigadores. Se definió como criterios de inclusión aquellos estudios que especificaban y comentaban sobre herramientas, prácticas, procedimientos para propiciar la reproducibilidad o describían errores sobre ellas. También se excluyeron estudios que se enfocaban a mencionar herramientas o procesos, pero sin comentar sus beneficios o limitantes, así como aquellos que se apoyaban en tecnologías o infraestructura consideradas como obsoletas. Para la cadena de búsqueda se recopilaron palabras claves como: comunicación, reporte, transferencia, conocimiento, motivaciones, experiencia, replicación, generalización, descripción, enfoque, entre otras.

Como parte del proceso de calidad de la revisión de literatura como la confiabilidad e integridad, se incorporaron mecanismos de control. Por ejemplo, para el proceso de

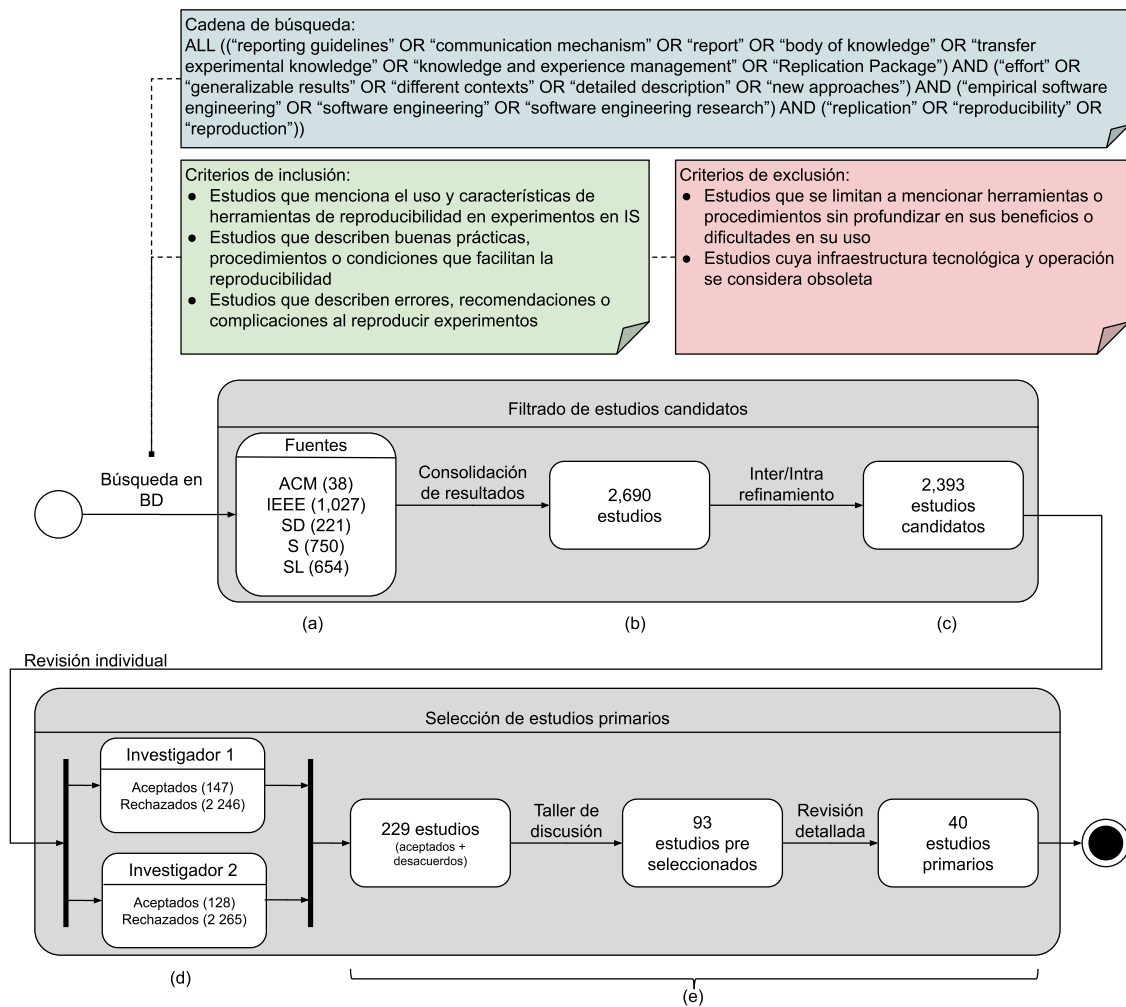


Figura 1.4: Resumen de revisión de literatura

selección se utilizó la estrategia de revisión por pares y talleres de discusión para tratar las discrepancias de selección³, como filtro para continuar con la revisión exhaustiva de los artículos primarios o seleccionados. De igual forma, para las fases de análisis las actividades fueron apoyadas en el uso de la herramienta de análisis cualitativo ATLAS.ti⁴. Esta herramienta informática permitió realizar una exploración de los documentos y añadir elementos denominados códigos⁵. Los códigos permiten definir categorías y asociaciones para crear redes semánticas y así contextualizar diferentes puntos de vista. Como resultado de este proceso, a partir de las asociaciones entre los códigos recabados, se construyeron varias redes semánticas sobre: (i) el intercambio de conocimiento; (ii) las formas de comunicación; (iii) motivación de replications; y (iv) herramientas, métodos o propuestas

³En el trabajo de revisión por pares y talleres de discusión participaron el doctorante y el director de la tesis doctoral

⁴ATLAS.ti es un programa informático para realizar investigación cualitativa o análisis de datos cualitativos y está disponible en www.atlasti.com

⁵Los códigos son una parte fundamental del análisis cualitativo ya que permiten realizar en análisis semánticos y estadísticos sobre ellos [32, 33]

usadas en procesos de experimentación. A continuación, se detallan los hallazgos de estas redes semánticas.

- **El intercambio de conocimiento** (Figura 1.5). Se produce en procesos conocidos como internalización y externalización. Se caracteriza por el uso de herramientas básicas como correos, portales web y talleres, hasta el uso de modelos y diseños metodológicos enfocados en los experimentos. Entre las prácticas y experiencias se menciona la importancia de detallar los procedimientos, eventos e inconvenientes con el propósito de entender la toma de decisiones durante la investigación. Se describe también que los métodos más comunes para el intercambio de conocimiento de este tipo de información son medios directos como llamadas o a través de correos electrónicos.
- **La comunicación** (Figura 1.6). También se relaciona con la internalización y externalización. Es parte del intercambio de conocimiento y se puede realizar a través de uso extendido de los paquetes de replicación o paquetes de laboratorio. Otros medios para la correcta comunicación son las guías para reportar información y también para describir amenazas a la validez. Al igual que con el intercambio de conocimiento, los medios usados para comunicar datos son los correos, medios de comunicación verbal y a través de repositorios, como los de control de versión.
- **La motivación** (Figura 1.7). El abordaje de este punto es escaso en los estudios considerados. En los artículos no se mencionan beneficios o incentivos explícitos de algún tipo para reproducir el estudio. Sin embargo, existen factores que inciden en la factibilidad de hacerlo. Por ejemplo, el esfuerzo de utilizar tecnologías distintas, el acceso a ciertos tipos de recursos o aplicar procesos no estandarizados, agregan costo al proceso de reproducibilidad.
- **Las herramientas** (Figura 1.8). Este aspecto es ampliamente estudiado, en contraposición a la motivación. Las herramientas encontradas pueden presentarse en distintas formas, como, por ejemplo: prácticas y modelos de transferencia de conocimiento, aplicaciones informáticas de apoyo al proceso de replicación, tecnologías adaptadas a la experimentación, guías metodológicas de investigación, y marcos de trabajo, plataformas o ecosistemas especializados a experimentos, entre otras.

Para verificar la evolución de la problemática de este trabajo, se realizó una breve revisión de la búsqueda original al periodo 2020-2022 en las bases de datos de mayor resultado, IEEE Xplore y Scopus (Figura 1.4). La cadena de búsqueda arrojó seis estudios (ver Anexo 11) de los cuáles, dos de ellos son relevantes para el análisis y se exploran temáticas sobre familia de experimentos y la crisis de reproducibilidad en ingeniería de software basado en evidencia. Los resultados son muy pequeños dado el intervalo de tiempo añadido tal como se menciona en [34]. Estos aspectos se relacionan de forma

directa a lo explorado en la problemática. En estos artículos se destacan puntos ya incluidos en la problemática como la necesidad de mecanismos automatizados y reproducibles [35], así como de la gestión del conocimiento en relación a las familias de experimentos para poder realizar la trazabilidad de hallazgos [36].

En conclusión, existe un sinnúmero de recursos como apoyo al aspecto de la reproducibilidad que se generan y evalúan en los ámbitos científicos. La reproducibilidad sigue siendo un tema de estudio debido a su importancia en la consolidación del conocimiento. Sin embargo, la investigación reproducible está lejos de ser una cualidad estándar en la disciplina de la ISE. En el siguiente apartado se presenta un posible esquema de la problemática actual.

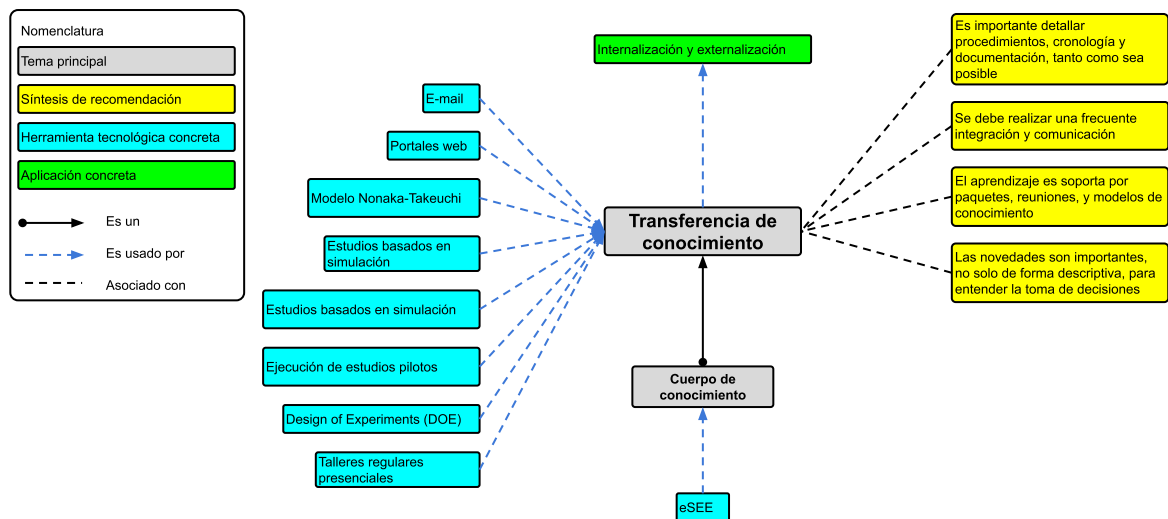


Figura 1.5: Red semántica sobre intercambio de conocimiento

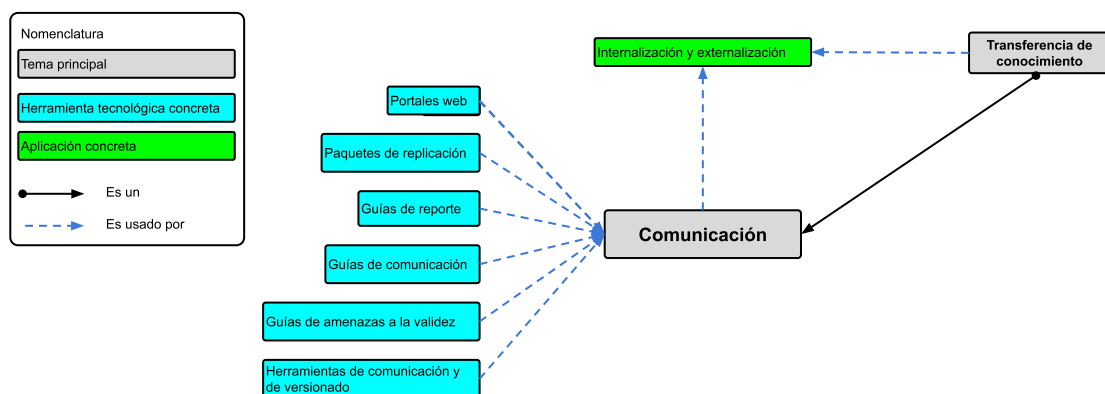


Figura 1.6: Red semántica sobre comunicación

1.2. Sistematización del problema

La sistematización del problema es parte del proceso metodológico que se ha empleado y que será explicado más detalladamente en la Sección 1.4. La sistematización se utilizará como marco conceptual guiando el diseño científico de esta investigación donde él se diseñará e investigará un artefacto dentro del contexto de problema [3] a través de dos actividades principales del proceso metodológico: diseño y validación. Por un lado, la sistematización aporta al diseño estableciendo los contextos sociales donde el problema se presenta. Mientras que, la sistematización permite enfocar la validación hacia el contexto del conocimiento y contribuciones de un artefacto. A continuación, se detallarán los contextos sociales y del fenómeno de la reproducibilidad.

1.2.1. Contexto social

El contexto social de la investigación empírica de la IS puede incluir la academia y la industria. En este sentido, en la academia se han creado diversos grupos de investigación en universidades alrededor del mundo, y las prácticas de empirismo han sido abordadas por importantes empresas de tecnología. También en este escenario aparecen las revistas científicas en donde se publican los avances de investigación. Dado estos escenarios es posible abordar las necesidades de los interesados de acuerdo a las perspectivas de interés [31] del producto científico.

- Investigador. Es quien analiza un estudio con el propósito de descubrir o aprender procedimientos o información nueva o relevante sobre un tópico que le concierne.
- Profesional. Es un lector cuyo enfoque se centra en los resultados y contexto de aplicación de un estudio.
- Analista. Se encarga de extraer información de tipo cuantitativa de los estudios para integrarlos en sus propios estudios.
- Replicador. Su principal objetivo es conocer la mayor información posible sobre los procesos, decisiones, y datos con el propósito de replicarlo.
- Revisor. Es un lector que, en representación de una revista o conferencia, asegura la pertinencia y el valor de una publicación.

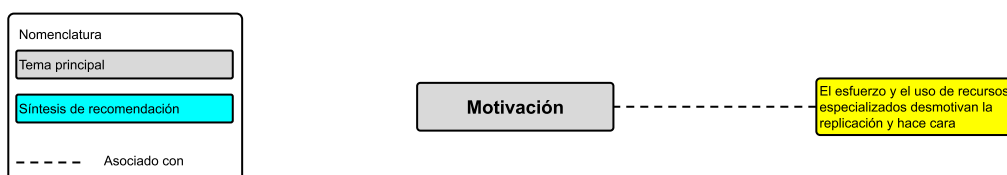


Figura 1.7: Red semántica sobre motivación

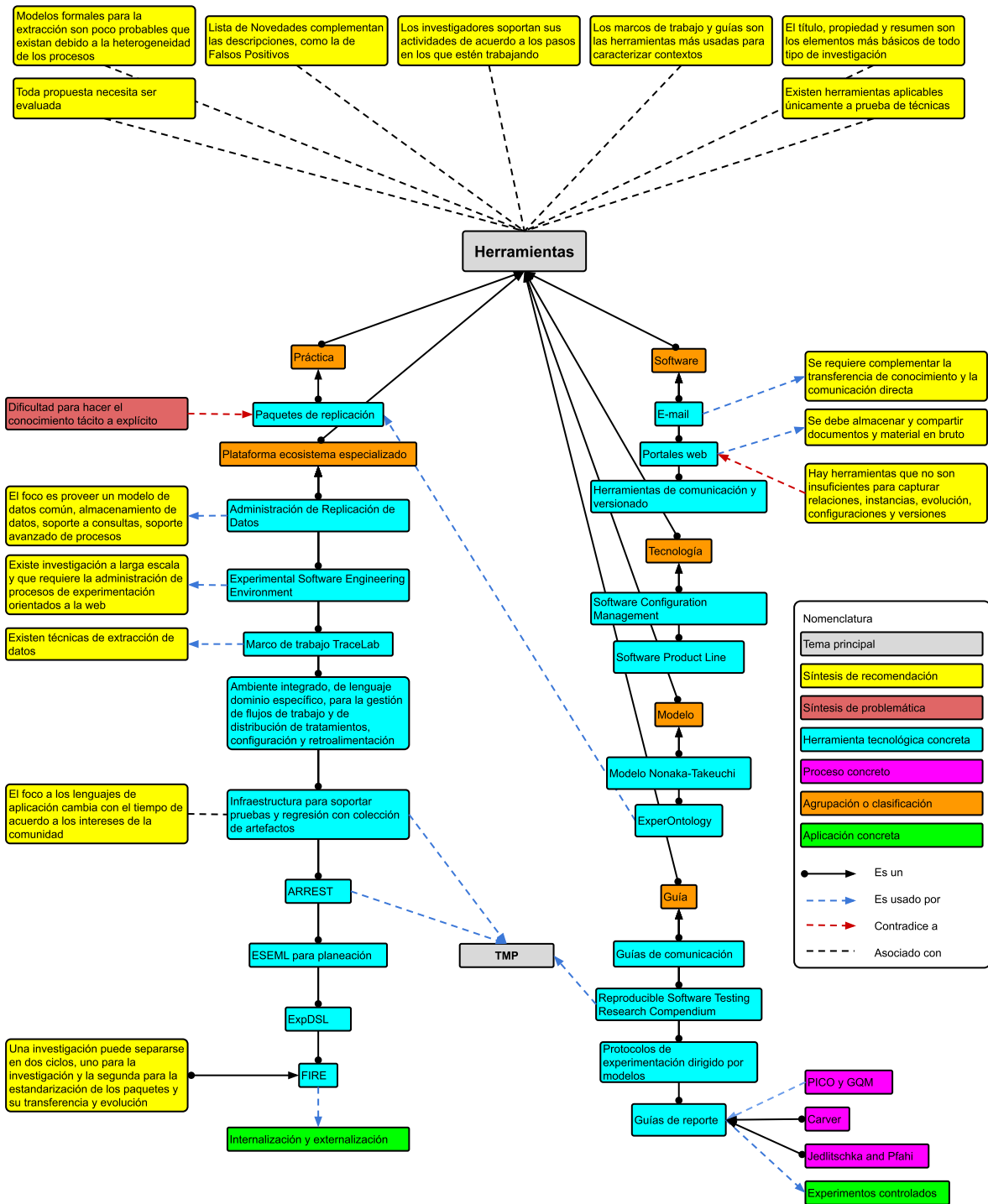


Figura 1.8: Red semántica sobre herramientas

- Autor. Es quien diseña y ejecuta un estudio y está interesado en transparentar su investigación a fin de que sea examinada, replicada o aplicada.

El autor de un estudio debe diseñar, planificar, ejecutar y transmitir la investigación y sus resultados. Por lo tanto, también debe tomar decisiones sobre cómo gestionar el avance de las investigaciones al definir qué herramientas de soporte utilizar. La selección de las herramientas de soporte o de prácticas repercuten de forma positiva o negativa en la reproducibilidad y, por ende, influye en la calidad del estudio. Posiblemente la necesidad de valor para el autor es utilizar herramientas y prácticas no invasivas, fáciles de aprender o implementar y que permitan transparentar la mayor cantidad de información posible ya sea para su revisión o para su reproducción. Por otro lado, los demás interesados podrían encontrar más valor al obtener mayor cantidad de información posible sobre los procesos y resultados de un estudio.

Dado estos deseos de valor, puede existir un posible conflicto lógico entre contar con una herramienta o práctica de bajo esfuerzo pero que permita transparentar todo el conocimiento. Wieringa [3] recomienda explorar los diferentes conflictos entre los deseos de los interesados ya que pueden ser causa de conflicto de un proyecto. En este caso, es necesario comprender qué características poseen las herramientas o prácticas más utilizadas. Para esto se tomó como referencia los conceptos de “dominio de problema” (Figura 1.2) y “formalización procedimental” como parámetros para entender el uso de las herramientas. En la Figura 1.9 se muestra la clasificación propuesta en la que es posible ubicar algunas de las herramientas exploradas (ver Figura 1.10) con las características de cada cuadrante. Tanto el dominio como la formalización surgieron al tratar de clasificar el uso de estas herramientas, en donde se determinó que existen dominios más maduros que otros, siendo estos en donde se debe usar herramientas mucho más formales, y por lo tanto más robustas para lograr documentar y analizar información.

Aunque en el contexto social intervienen estos actores, es necesario reconocer que ellos pueden poseer diferentes niveles de experiencia. Por definición, un experto es capaz de pensar de forma efectiva en soluciones y por lo tanto, afectar el desarrollo de las actividades de investigación. Según [37] la experiencia no se trata de una habilidad general como la memoria o la inteligencia, ni la capacidad de aplicar estrategias; sino que se trata de la capacidad de organizar, representar e interpretar la información a partir de los conocimientos. De este modo, un experto es capaz de: (i) notar características y patrones significativos; (ii) organizar el conocimiento reflejando una profunda comprensión; (iii) acceder a contextos de información en lugar de hechos o proposiciones aisladas; y (iv) recuperar aspectos importantes de conocimiento con poco esfuerzo o de forma automática; entre otras. En el ámbito de la investigación, la transferencia de conocimientos depende en gran medida de los criterios de los investigadores para sintetizar la información relevante. Las implicaciones de los cuatro puntos anteriores entonces pueden jugar un

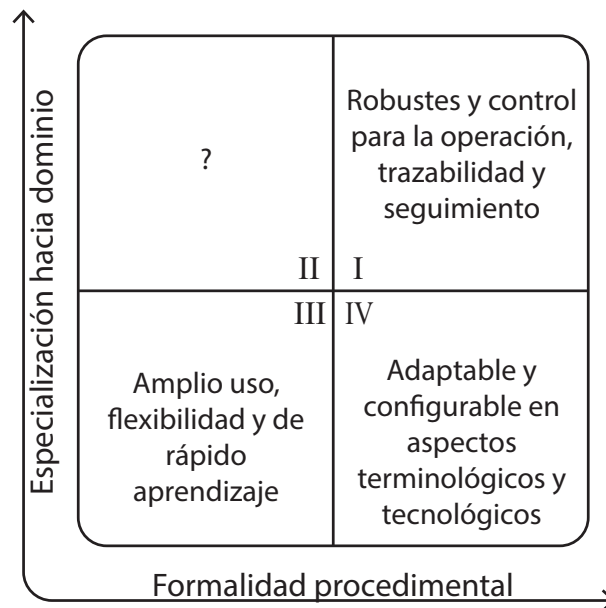


Figura 1.9: Categorías de herramientas por especialización y formalidad

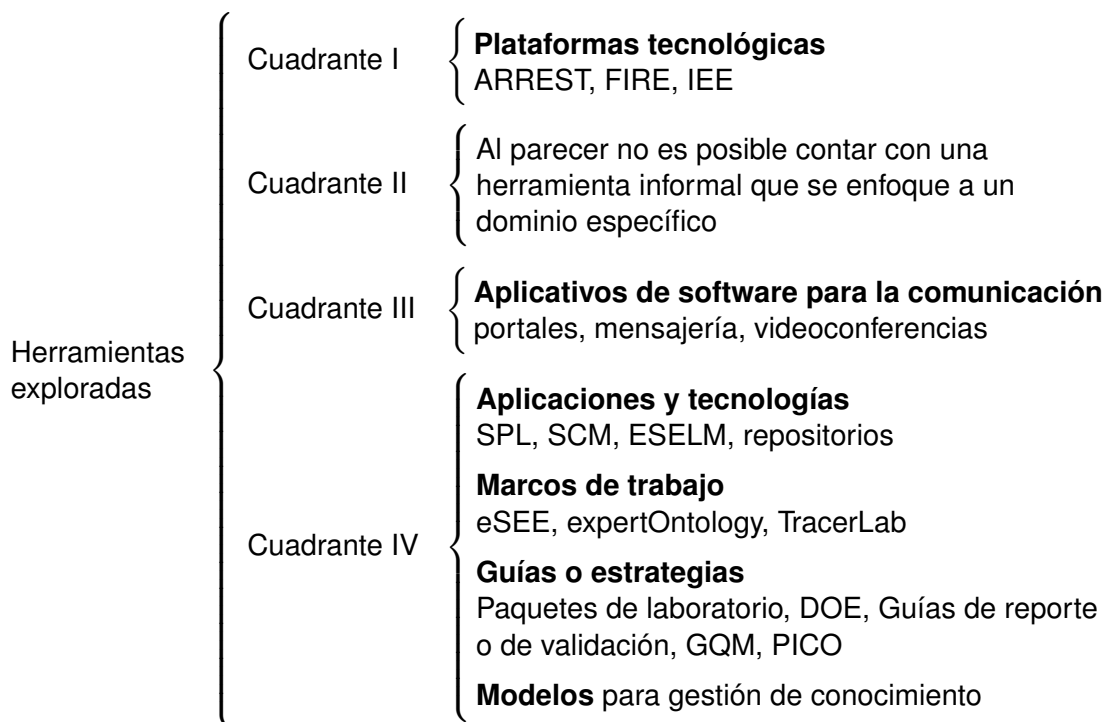


Figura 1.10: Categorización de herramientas

rol importante para transparentar, organizar y transmitir la información de un estudio; ya que la forma de procesar la información entre un novato y un experto es sustancialmente distinta. Posiblemente este aspecto sea la fuente principal del conocimiento tácito inherente al desarrollo de las investigaciones. Tal como afirma [38] la diferencia entre un experto y un novato puede divergir en la calidad del reconocimiento de problemas, su representación, y por ende, determinar las mejores soluciones.

1.2.2. Contexto del fenómeno

Como se vio en la sección anterior, la reproducibilidad en una investigación puede ser un atributo de calidad. Este atributo, aplicado a un estudio, le proporciona características que le permiten mejorar tanto su revisión y reproducción gracias a la mejora en la comunicación, transparencia de sus procesos, datos e interpretaciones. En contraste, en lo que respecta a la ingeniería de software, muchos modelos (McCall, Boehm, FURPS, Dromey, e ISO 25000, CMMi entre otras) [39, 40] establecen estructuras robustas de calidad de producto y procesos en la que intervienen atributos o factores que se relacionan a una serie de criterios o aspectos; entonces es necesario poder identificar estos componentes respecto a la reproducibilidad de una investigación. En los estudios sobre la reproducibilidad de un estudio empírico se han tratado desde su conceptualización y caracterización hasta el planteamiento de herramientas que habilitan la comunicación y la transferencia.

Las herramientas exploradas anteriormente deben contribuir a la reproducibilidad de un estudio de alguna forma. A su vez, en la literatura es posible encontrar muchas propuestas y análisis en torno a la aplicabilidad y resultados respecto a la reproducibilidad de las herramientas. En primera instancia este fenómeno puede contribuir a la madurez de la disciplina poniendo a disposición de los dominios de problemas un catálogo de herramientas. Por otro lado, contrariamente a los posibles beneficios, eso podría convertirse en una bola de nieve⁶. Al contar con tantas alternativas a considerar, los procesos heurísticos para la selección de la mejor podrían llegar a ser ineficientes o confusos. Para soportar este supuesto se ha realizado un estudio para analizar la complejidad arraigada a una propuesta de mejora de reproducibilidad de un método de investigación [20].

Por ejemplo, en una de estas propuestas de mejora a la reproducibilidad[41], la autora estudia el impacto de los procesos y los mecanismos para gestión de conocimiento. Al revisar las contribuciones de la propuesta se determinó el aporte hacia dos grandes cambios. El primer cambio adicionó nuevos procesos tanto en la fase de planeamiento, ejecución y documentación de la investigación. Mientras que el segundo cambio se enfocó a integrar el uso de herramientas para gestionar el conocimiento. Como resultado, el proceso propuesto define más actividades al proceso tradicional e insta al uso integrado de plantillas

⁶El efecto bola de nieve se refiere a una metáfora en la que un estado inicial se desarrolla sobre sí mismo creciendo hasta llegar a ser más serio, peligroso o desastroso a manera de círculo vicioso.

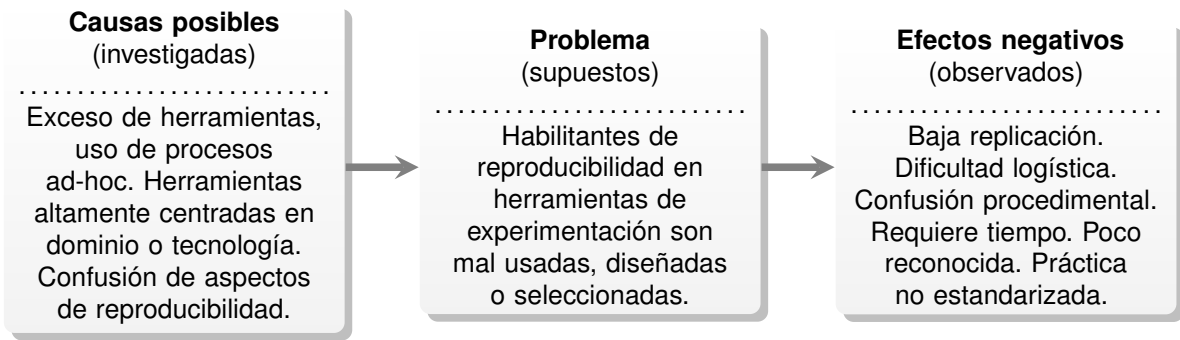


Figura 1.11: Causas, problema, efectos

y herramientas de minería de datos para el tratamiento del conocimiento a futuro. De esta manera, al evaluar la propuesta, se contrastó el aporte respecto con la gestión de la comunicación, conocimiento y motivaciones. Como resultado se determinó que el aporte contribuye principalmente a la gestión de información relevante, y al entendimiento de la toma de decisiones. Por otra parte, se deja de lado la estandarización de materiales, terminología, esquemas de trazabilidad. Finalmente, se discute sobre la agregación de confusión procedimental y costos de adaptación o adopción.

Se puede resumir que las propuestas incorporan procedimientos para habilitar la reproducibilidad. Al hacer esto crean más dificultad en el proceso de selección de herramientas. Al existir numerosas alternativas y no contar con un mecanismo de evaluación o condiciones puede escogerse una herramienta no apropiada. Es necesario conocer qué aportes proveen las herramientas. También las propuestas se enfocan en monolitos que son difíciles de trasladar a otros dominios. Los experimentadores expertos ya tienen sus procesos optimizados pero los novatos pueden encontrar esta actividad exhaustiva. Se puede resumir este fenómeno como un problema en el uso, diseño o selección de herramientas que habilitan los aspectos de reproducibilidad tal como se propone en la Figura 1.11.

1.3. Preguntas y objetivo del diseño de investigación

El anterior marco de problema evidencia posibles efectos negativos en el contexto de la reproducibilidad. Durante la exploración se observa una gran variedad de herramientas posiblemente no orientadas a los habilitantes de reproducibilidad. Ante las interpretaciones y supuestos extraídos de las investigaciones preliminares, surgen los siguientes cuestionamientos:

(P1): ¿Cuáles son las características o habilitantes de la reproducibilidad que tiene una investigación experimental?

Como se ha anunciado previamente, la reproducibilidad puede ser considerada como un atributo de calidad de una investigación. A mayor reproducibilidad, mayor son las oportunidades de ejecutar los procesos de replicación. De este modo, las herramientas exploradas se enfocan en la transparencia de los datos; sin embargo, parece ser que considerar solamente este atributo no es suficiente. La importancia de esta pregunta radica en comprender qué otros factores pueden influenciar en este aspecto de calidad. Conocer estos factores podrían servir tanto en los procesos de selección de herramientas, así como en el diseño de nuevas propuestas.

(P2): ¿De qué forma se pueden esquematizar las necesidades de investigación reproducible de los investigadores?

Durante la exploración inicial se encontró que los investigadores encuentran que las tareas de reproducción son difíciles. Esta dificultad se acrecienta en los procesos de externalización de conocimiento; es decir, cuando el conocimiento sobre una investigación es comunicado y recreado por un grupo de investigación externo al original. Por el contrario, la internalización obedece a criterios y experiencias desarrolladas por y para las necesidades del grupo de investigación; como, por ejemplo, inducción de conocimientos. Aunque las necesidades han sido exploradas, éstas por lo general se enfocan a aspectos procedimentales que se abordan en guías, por ejemplo. Adicionalmente, es necesario explorar otras necesidades, tomando en cuenta que no solo existen expertos sino también novatos en los procesos de replicación.

(P3): ¿Puede desarrollarse un mecanismo que soporte el proceso de selección de herramientas?

Los artefactos de soporte a la reproducibilidad son herramientas que vienen en diferentes presentaciones. Las Figuras 1.9 y 1.10 brinda una idea de categorización de las herramientas que suelen utilizarse. Adicionalmente, tras analizar una propuesta y observado el proceso de análisis de esa mejora, se debe considerar que la formulación de un mecanismo de soporte (artefacto) no se debe enfocar al uso de herramientas, sino que se debe enfocar al proceso de control (madurez del contexto del problema de reproducibilidad), etapa en la que aplican criterios de selección de alternativas de solución (herramientas).

De esta manera se plantea el siguiente objetivo de investigación descrito con la plantilla "problema-objetivo".

Mejorar la calidad de la experimentación en IS **mediante** un modelo de estrategias **que satisfaga** las necesidades de la reproducibilidad de los investigadores **para** que las actividades de replicación sean más fáciles de implementar interna y externamente.

1.4. Metodología de investigación

Dado el supuesto planteado de que las herramientas que habilitan la reproducibilidad en un estudio no son adecuadamente utilizadas, diseñadas o seleccionadas; la solución se enfocaría en proponer un artefacto que asista a un investigador en los procesos de selección y utilización de alternativas de solución (herramientas) considerando características o factores de reproducibilidad. A lo largo de los puntos anteriores se han encontrado propuestas de implementaciones de tipo monolíticas, que parecen no aportar de forma íntegra a la solución del problema, por ejemplo, agregar más procesos a los ya existentes. Por el contrario, resultaría conveniente poder caracterizar las herramientas como parte de procesos de selección de estas. Es importante considerar que debido a que los contextos de investigación no son iguales, una herramienta que asista a los investigadores debería proporcionar el nivel de reproducibilidad acorde a las estrategias que el investigador necesite, aportando a la flexibilización en las actividades de investigación particulares.

Para comprender el alcance de la solución se ha tomado en cuenta los componentes del modelo problema-solución (ver Figura 1.2). En ese contexto se encuentran dos componentes: de “solución del problema”, y de “control”. Dentro del primer componente, el proceso inicia comprendiendo la necesidad que permite describir el problema. A continuación, se busca una solución dentro del dominio de conocimiento, cuyo análisis da lugar a una serie de alternativas de solución. Se ha considerado que el proceso de análisis de alternativas acarrea un problema ya que no se aplica un proceso sistemático que permita la selección apropiada de herramientas que considere a la reproducibilidad. Dentro del segundo componente de control se contempla diversos elementos para realizar la selección de la alternativa, como: modelos, criterios de calidad y restricciones; que permiten evaluar y seleccionar las alternativas. Por lo tanto, la solución requiere comprender y definir los elementos del contexto de control para una selección de herramientas apropiadas. Los elementos pueden consistir en las descripciones planteadas en el Cuadro 1.2, que son (i) alternativas, (ii) modelo, (iii) criterios de calidad, (iv) restricciones, y (v) optimización.

Desde el punto de vista aplicativo, la propuesta puede ser significativa para seleccionar y aplicar adecuadamente un conjunto de prácticas asociadas a necesidades de reproducibilidad. Esto podría ayudar a analizar el costo beneficio de utilizar un proceso

desconocido. Aunque la experiencia tiene un gran peso al aplicar estrategias y prácticas durante la ejecución de un proceso, sería posible evaluar los aspectos de calidad de las prácticas. La solución, entonces, debe poder esquematizar las prácticas de reproducibilidad del proceso de experimentación. La principal limitación radica en la síntesis de todas las posibles herramientas que se utilizan y poder separar los contextos procedimentales de sus contextos prácticos o de sus dominios de aplicación.

La metodología de investigación propuesta plantea el análisis, diseño y construcción de una solución bajo una perspectiva científica tanto para su análisis como para su evaluación. Consecuentemente se ha seleccionado el método denominado *Design Science* (DS) para guiar el diseño e investigación de un artefacto en el contexto de la problemática. Para esto, se necesita explorar y entender los posibles componentes del objeto de estudio en un contexto determinado a través del diseño e investigación de artefactos (Figura 1.12). Como en todo estudio, se debe partir por la indagación del problema, que es el corazón de la investigación, lo cual permite considerar los objetivos idóneos de los interesados. A partir de esto es posible formular preguntas de conocimiento sobre el diseño de un artefacto dentro contexto de investigación. La revisión de la relación entre las preguntas de conocimiento y la formulación de la solución permite el enfoque adecuado de las posibles contribuciones de la propuesta.

Adicionalmente para responder las preguntas de investigación planteadas en la sección anterior, es necesario que el proceso metodológico se apoye en varias estrategias y actividades de investigación que se detallan en los capítulos siguientes de la siguiente manera:

En el capítulo 2 se planteará la propuesta de solución. Para esto se seguirán tres actividades clave. (I) Identificar los habilitantes de la reproducibilidad bajo una perspectiva de calidad, tratando de abordar factores que hacen que una investigación sea reproducible. Esto permitirían evaluar una herramienta de soporte, valorando su aplicación y diseño. Aquí, un enfoque cualitativo sería apropiado para caracterizar y describir los aspectos de calidad, así como explorar e interpretar el uso de las herramientas, y poder formar una teoría respecto a los habilitantes de la reproducibilidad [42]. Como resultado, se podrá formular y contestar preguntas del conocimiento sobre las herramientas o artefactos dentro del contexto de la reproducibilidad. (II) El paso siguiente consistirá en esquematizar las estrategias del proceso de selección de herramientas de soporte a la reproducibilidad. Una vez comprendidos los habilitantes de la reproducibilidad, se investigará y validará si dichos habilitantes cubren las necesidades de los interesados en la investigación. De igual manera es importante determinar cuáles procesos de toma de decisiones se ejecutan. Como resultado de esta actividad se podrá analizar si existen procesos maduros para selección de herramientas o determinar si la experiencia prima en esta actividad y cómo se ejecuta esta práctica. Para este análisis se propone el uso de técnicas como la entrevista

Cuadro 1.2: Elementos de control

Elemento	Descripción
Alternativa	Un conjunto de herramientas que soportan la reproducibilidad categorizadas y organizadas de acuerdo con el aporte del proceso experimental
Modelo	Un artefacto esquematizado de acuerdo con los objetivos de reproducibilidad que permita identificar herramientas aplicables al proceso experimental
Criterio de calidad	Los criterios de reproducibilidad enfocados en habilitar la transparencia de datos, análisis e interpretaciones
Restricción	Las características del contexto que restrinjan el uso o aplicación de una herramienta en un contexto de problema particular
Optimización	Los procesos de evaluación o criterios de selección de las alternativas más adecuadas al contexto del problema o necesidad

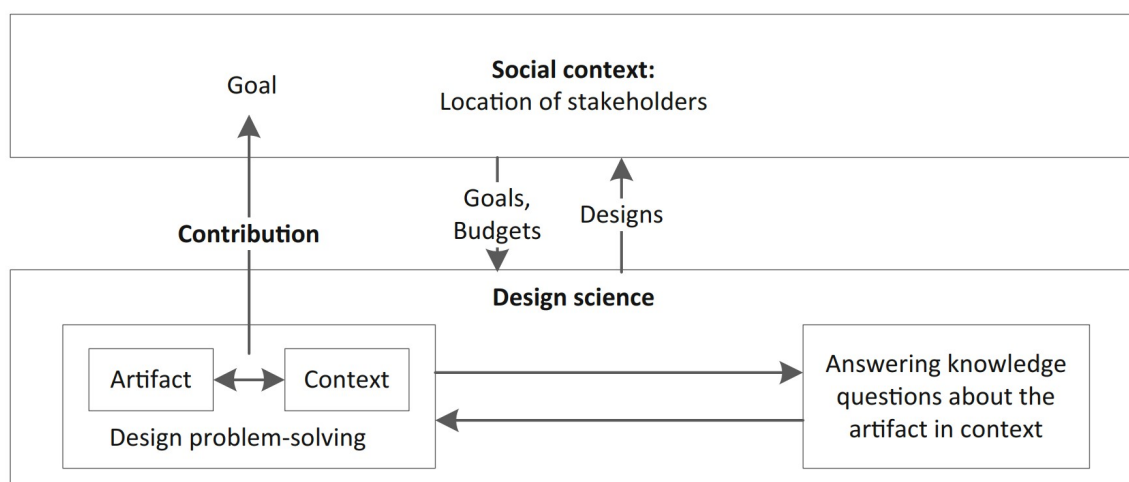


Figura 1.12: Relación entre el contexto de problema y contexto de creación de solución según DS. [3]

a expertos. La recolección de datos y la observación serán importantes para entender las interacciones de los participantes. La interpretación en esta etapa permitirá guiar el contexto del artefacto a desarrollar. (III) Finalmente se plasmará el artefacto considerando los elementos propuestos en el Cuadro 1.2 como parte de su diseño. Por lo tanto, el diseño del artefacto deberá reflejar tanto los factores de calidad, las alternativas, sus restricciones, así como las necesidades del contexto. Considerando la evolución de PSM (Figura 1.3) un modelo caracteriza a un proceso de mayor madurez. Con este preámbulo se puede considerar la creación de un modelo como un posible artefacto para sintetizar y representar los elementos y las interacciones relacionados al contexto del problema.

En el capítulo 3 se ejecutará la validación de la aplicabilidad conforme a la metodología DS. Una vez plasmado el artefacto es necesario evaluar las suposiciones de la solución propuesta sobre los objetivos de los interesados en relación con las contribuciones del artefacto en el contexto de la reproducibilidad. En este escenario, un proceso de experimentación sería idóneo para confirmar la teoría planteada y contrastar el artefacto propuesto. Los resultados de esta evaluación permitirán aplicar actividades de mejora continua o, en su defecto, replantear el modelo y los supuestos en una investigación futura.

Finalmente, en el capítulo 4 se plasmará las conclusiones del trabajo, mostrando la interpretación de resultados, las lecciones aprendidas y las consideraciones a la validez que se manejaron durante el diseño y validación de la propuesta.

Capítulo 2

Planteamiento de la propuesta: Modelo de reproducibilidad

Por lo explorado en el capítulo anterior, la reproducibilidad se considera un atributo de calidad dentro del contexto de la investigación. Este atributo de calidad puede ser observado o medido a través de sus productos finales, como, por ejemplo, en los reportes de resultados, y conjuntos de datos resultantes, haciendo alusión a la calidad del producto. Sin embargo, es posible propiciar el aseguramiento de calidad a través del seguimiento y control de las prácticas utilizadas para el proceso de investigación o experimentación. Como tal, existen varios principios y paradigmas de mejora (como QIP, modelos de calidad como CMMi). Estos coinciden con el proceso metodológico propuesto (DS) ya que se enfocan en contextualizar los entornos de trabajo, establecer objetivos, definir indicadores y esquemas para definir un nivel de “madurez” y así poder explorar y aplicar mejoras. Esta perspectiva de análisis se alinea al enfoque del modelo de madurez que se pretende alcanzar para el abordaje de un artefacto o herramienta apropiada a las necesidades de reproducibilidad.

En este capítulo, como parte de la propuesta de solución, se definirá detalladamente los elementos del contexto de control para la experimentación en conjunto con el aspecto de calidad de la reproducibilidad. Estos elementos abarcan: un modelo, criterios de calidad y restricciones para la evaluación de alternativas de solución. En primera instancia se debe descubrir, analizar y sintetizar los requerimientos que los artefactos (herramientas y prácticas) deben satisfacer para dar soporte al proceso experimental reproducible. En segunda instancia se debe caracterizar los aspectos o factores de calidad que influyen el nivel o grado de reproducibilidad, pero tomando en consideración una perspectiva de transferibilidad y transparencia.

En la sección 1.2.1 se introdujo los elementos dentro del contexto social que incluyen perspectivas de necesidades (investigador, profesional, analista, replicador, revisor, autor), y herramientas que se consideran como alternativas disponibles para su uso dentro de

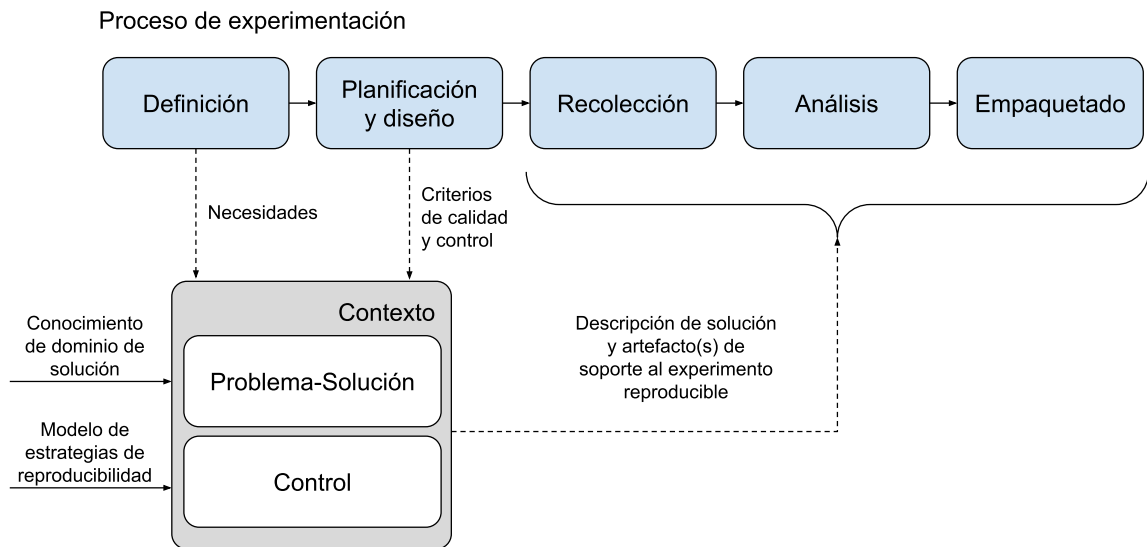


Figura 2.1: Visión global sobre la propuesta

las actividades de experimentación (plataformas, aplicativos, tecnologías, marcos, guías, modelos, entre otras).

Consecuentemente con el PSM, el entorno experimental denominado contexto se conforma de dos grandes componentes: el PSM y el control. El PSM es el proceso principal donde se selecciona el artefacto más apropiado para el contexto del problema. Aquí se contemplan: la necesidad, la descripción del problema, el conocimiento del dominio de la solución, las alternativas de solución, así como la descripción de la solución y el artefacto. El segundo componente es el control, donde se contemplan: el modelo, los factores de calidad, las restricciones y el proceso de optimización de las técnicas y la heurística. A continuación, se ha explorado cada componente y se ha finalizado con el planteamiento de preguntas de conocimiento y su respuesta a fin de que la propuesta pueda servir como soporte a los procesos de experimentación. La Figura 2.1 presenta cómo la propuesta aporta a los procesos de control para la selección de alternativas acordes a una instancia de solución. Se pretende que considerar adecuadamente una estrategia aseguraría o maximizaría la calidad de reproducibilidad de una investigación.

2.1. Componente “Contexto”

A través de la herramienta de análisis cualitativo se realizó la exploración documental y la subsecuente extracción de términos relacionados a la “reproducibilidad” y sus “atributos” estableciéndose las relaciones entre estos conceptos. Entre las principales relaciones encontradas se destacan la diseminación de los datos, la comunicación y colaboración, la capacidad de validarlos a través de la gestión y estructuración de los datos. A continuación,

se detallan estos conceptos.

Diseminación de datos. El contexto en que gira esta investigación se centra en los experimentos en IS con relación a la investigación reproducible. Como se ha indicado anteriormente, la reproducibilidad se relaciona con la transparencia y la transferibilidad; por lo tanto, debe ser consistente con el principio de diseminación [29] ya que, sin ella no es posible la consolidación de conocimiento. En este apartado, Rozier formula dos preguntas: la primera sobre la capacidad que tienen otros para recrear de forma razonable un trabajo; y la segunda sobre la disponibilidad de información suficiente para hacerlo. De esta manera, el contexto de un experimento reproducible incluye la capacidad de diseminar los resultados a partir de detalles (teorías, datos, algoritmos, librerías, código) documentados.

Comunicación y colaboración. Bajo diversos dominios de investigación se menciona la colaboración entre investigadores, principalmente en lo referente a la capacidad de compartir y proporcionar la información [43]. Oliveira hace una clara distinción sobre las diferencias entre información generada en la academia y la generada en la industria, donde en esta última, la disponibilidad y la colaboración es mucho más limitada. Sin embargo, pueden especificarse protocolos para que ciertos aspectos fundamentales puedan estar disponibles. El propósito de la reproducibilidad radica en la actividad de validar y contrastar resultados [5]. Este proceso debe involucrar no solo datos (datos originales, sin procesar, o en bruto) sino los esquemas mentales que dirigen la interpretación de procesos analíticos y conclusiones (ver Cuadro 1.1).

Codificación y estructuración. Los estudios cuya información y documentación poseen esquemas de codificación y estructuración incrementan las propiedades de reproducibilidad. Los investigadores no solo deben contar con habilidades y capacidades de análisis sino también deben tener y desarrollar habilidades de comunicación en la que se incluye la documentación. La organización adecuada de la información permite que esta sea accesible a distintas audiencias y comprender de mejor manera los procesos de toma de decisiones para cualquier tipo de interesado en el contexto de la investigación [5, 44].

2.2. Componente “Problema-Solución”

Para explorar los elementos de esta sección se partió del análisis cualitativo y la creación de redes semánticas para determinar las relaciones entre los códigos (conceptos) descubiertos. Se formularon diversas redes semánticas en relación con las necesidades del contexto de la experimentación reproducible y los problemas que esta enfrenta. A continuación, se detallan los elementos de Modelo PSM de la Figura 1.2.

2.2.1. Necesidades

De acuerdo con el Modelo PSM, el contexto sirve de punto de inicio para la determinación de la solución de un problema. El primer paso del proceso hace referencia a las necesidades. En el caso de esta investigación, cuyo enfoque definido es la experimentación reproducible, se torna muy difícil establecer los puntos focales de las necesidades. por lo tanto, se ha recabado posibles necesidades básicas extraídas de la red semántica mostrada en la Figura 2.2 y que nos permitió considerar y detallar los siguientes ámbitos de necesidad.

- Cooperación
- Contenido
- Validación
- Trazabilidad
- Motivación

Necesidad de cooperación

A partir de los contenidos del Cuadro 2.1 se puede definir esta necesidad de la siguiente manera: La reproducibilidad necesita que una investigación facilite o propicie la cooperación. Los investigadores o grupos de investigación deben colaborar entre sí en tareas de revisión, así como la validación y corroboración de resultados y demás componentes o artefactos generados durante una investigación. La cooperación permite una verdadera y fiable manera de construir conocimiento científico, por lo que es necesario que los ambientes de investigación (experimental) sean capaces de proporcionar disponibilidad a la mayor cantidad de información sobre una investigación, así otros investigadores pueden acceder a lo que consideren interesante. Finalmente, esta información debe ser comprendida o navegable a través de estructuras o mecanismos que permitan explorar toda la documentación, y no solo la reflejada a través de reportes (artículos científicos).

Cuadro 2.1: Citas sobre cooperación en red semántica "Necesidades"

Cita	Interpretación de la cita
------	---------------------------

Cooperación para construir conocimiento	
--	--

Continuación de la tabla 2.1

Cita	Interpretación de la cita
[45, 46]	En el estudio se hace hincapié en las contribuciones sobre el desarrollo y uso guías y paquetes para soportar la replicación. Pero a pesar de lo que se puede lograr con este tipo de contribuciones en relación con la replicación, en el estudio se afirma que “ninguna” de estas establece o direcciona de forma explícita los métodos para compartir conocimiento y permitir la cooperación entre grupos de investigación. Finalmente se enfatiza que, sin métodos adecuados de transferencia, no puede existir la cooperación es necesaria para facilitar la construcción de conocimiento, evolucionar los artefactos experimentales y llegar a conclusiones a lo largo de todos los estudios.
[47]	En el estudio se explica cómo el proceso de replicación interna y externa habilita la construcción del cuerpo del conocimiento sobre familias de técnicas o de técnicas relacionadas y principios para el desarrollo de software.
[46, 48]	En el estudio se muestra que, en literatura sobre investigación en ingeniería de software, la mayoría de los estudios contienen poca o ninguna validación empírica. Los autores catalogan esta situación como “perturbadora” dada la dificultad existente al construir una base de conocimiento utilizable a partir de estudios individuales y aislados.
Compartición y disponibilidad	
[49, 43]	El estudio destaca dos aspectos importantes para la reproducibilidad necesaria para la colaboración: la capacidad de compartir información, y el grado de disponibilidad de esta.
Internalización y externalización	
[47, 50]	En el estudio se habla de las muchas herramientas complementarias en una investigación, como los portales web y su capacidad para contener materias y documentación. También se hace notar que almacenar estos artefactos no es suficiente para capturar las relaciones entre todos estos objetos (artefactos, cuestionarios, sujetos, datos en bruto, etc.), o para soportar su instanciación (usando plantillas y meta modelos), ni su evolución, configuración o control de versiones.

Continuación de la tabla 2.1

Cita	Interpretación de la cita
[31]	En el estudio también se destaca que el uso de la correspondencia electrónica (<i>e-mail</i>) es suficiente para responder preguntas de investigadores externos, pero que no facilita la identificación de los múltiples cuestionamientos que se generan por la natural y compleja interrelación de todo un diseño experimental.
Estructuras colaborativas entre autores y replicadores	
[47, 24]	Los autores argumentan que para una replicación efectiva se “requiere algo más que simples paquetes de laboratorio” sino también de procesos para instanciar, evolucionar y usar los estudios originales. De igual forma se menciona que esta necesidad involucra estructuras de comunicación efectivas entre investigadores para cubrir todo el conocimiento no explícito y para lograr entender el proceso de experimentación.

Necesidad de contenido

A partir de los contenidos del Cuadro 2.2 se puede definir esta necesidad de la siguiente manera: La investigación genera contenido que debe ser documentado para propiciar la reproducibilidad. Este contenido incluye procedimientos y actividades detalladas, no solo respecto a características sino también mediante la especificación explícita de tiempos, novedades, incidentes y decisiones de su ejecución. Este contenido debe ser empaquetado en conjunto con los datos en bruto y datos procesados, siguiendo estructuras de organización que permita la comprensión de la investigación, así como los mecanismos para su instanciación y evolución. Finalmente, es importante considerar la inclusión de reportes dinámicos que permitan explorar eficientemente el contenido desde diferentes perspectivas de los interesados.

Cuadro 2.2: Citas sobre contenido en red semántica "Necesidades"

Cita	Interpretación de la cita
Inclusión de procesos	

Continuación de la tabla 2.2

Cita	Interpretación de la cita
[47, 5]	Los autores describen la importancia de detallar los procesos realizados para una correcta replicación; tanto como sea posible para poder determinar tiempos, documentos, actividades de entrenamiento, entre otras.
[47, 51, 52]	En el estudio se establece que una comunicación efectiva sobre un experimento incluye de forma explícita algo más que una descripción de un paquete de laboratorio. Se debe incluir una lista de problemas y novedades que puedan ser inválidas fuera del alcance de experimento]; así como la especificación clara y detallada de todos los esquemas racionales empleados.
[47, 29]	Los autores creen que la documentación de un experimento debe tener dos versiones, una “semilla” y otra de “oro” que especifique información adicional sobre la información tácita.

Estructuras colaborativas entre autores y replicadores

[53, 47, 50] Los autores argumentan que las estructuras de colaboración son parte de los procesos comunicativos. La documentación “empaquetada” carece de información sobre los procesos y eso hace evidente la necesidad de la comunicación entre investigadores. Sin embargo, esta información debería estar estructurada a manera de “conocimiento”.

Elementos de una investigación

[53, 54] El estudio destaca la falta de información relevante en los reportes (artículos). Los autores sostienen que la problemática puede ser cubierta a través de la conformación de modelos dinámicos de reporte que estructuren la información (objetivos, preguntas, descripciones, comportamientos, simulaciones, modelos, diseños, salidas) de manera que los reportes puedan ser construidos a partir de los contextos estructurales de investigación.

Necesidad de validación

A partir de los contenidos del Cuadro 2.3 es posible definir esta necesidad de la siguiente manera: Una investigación debe ser validada en tres aspectos: datos, análisis e interpretaciones. Los datos y materiales deben ser empaquetados para proporcionar

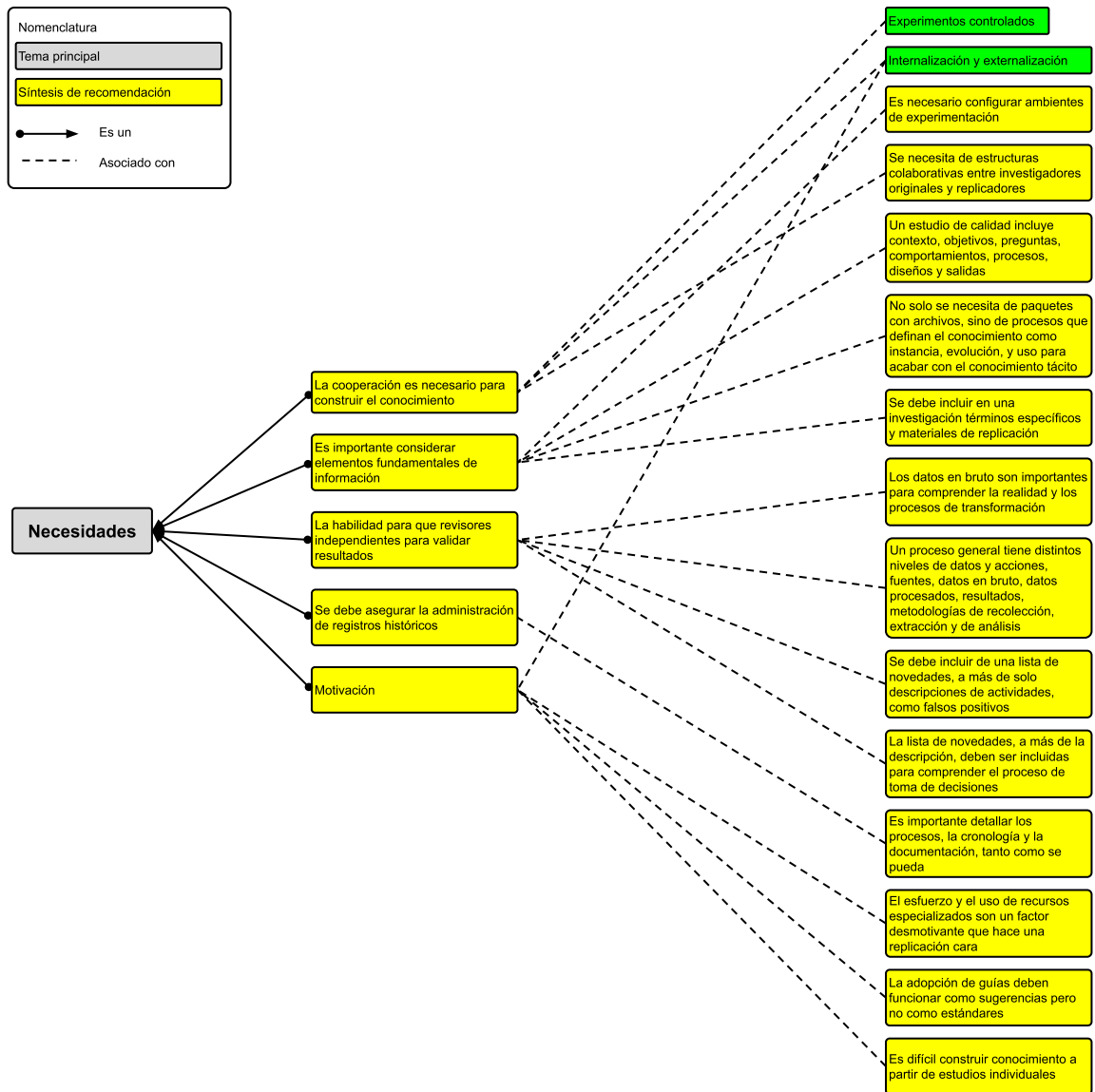


Figura 2.2: Red semántica sobre necesidades

la capacidad de variar parámetros de contextos. Los datos en bruto, los procesos de recolección, la documentación de novedades, la documentación de la racionalización de decisiones, la documentación de falsos positivos son elementos fundamentales para el tratamiento efectivo de conocimiento tácito y que, por lo tanto, debe ser incluida en el empaquetado de la investigación.

Cuadro 2.3: Citas sobre validación en red semántica "Necesidades"

Cita	Interpretación de la cita
Presentación de resultados	
[47, 53, 55]	Se recomienda que en un paquete de laboratorio se presenten resultados y datos en términos de materiales, oportunidades de variación, contextos de aplicación y tratamientos. De esta manera el paquete experimental se podría considerar como una infraestructura para replicaciones futuras.
Presentación de metodologías	
[5, 55]	Se establece la necesidad de una plantilla procedimental para la investigación empírica. De esa manera establecen ocho conceptos para determinar la reproducibilidad de un estudio, pero que los conceptos pueden ser utilizados más allá de la evaluación sino también como un mecanismo de abstracción metodológica del proceso de investigación. Por lo tanto, a través de este proceso se puede mejorar la reproducibilidad.
Datos en bruto	
[5, 51, 50]	Se explica la importancia de hacer explícito los procesos de tratamiento de los datos en bruto, desde su recolección hasta la especificación de su conexión con la realidad. De igual forma explican que es "virtualmente imposible" recomendar una forma universal para documentar esta información para cada situación.
Falsos positivos	
[45]	Se comenta la práctica de reportar solamente la información correcta; pero la creación de listas de "falsos positivos frecuentes" permitirían una inspección adecuada de defectos no aceptados por los experimentadores.
Toma de decisiones	

Continuación de la tabla 2.3

Cita	Interpretación de la cita
[43, 56]	Hablando de la experiencia de los autores, encontraron que la comunicación efectiva de un estudio involucra el entendimiento de las decisiones racionales que se llevaron a cabo en una ejecución, ya sea de forma planificada o de forma reactiva a sucesos dentro del experimento.

Necesidad de trazabilidad

A partir de los contenidos del Cuadro 2.4 se puede definir esta necesidad de la siguiente manera: La investigación debe contemplar elementos que permitan disponer conexiones o vínculos entre toda la información, herramientas, código y demás documentación. Estas conexiones agregan la trazabilidad necesaria para comprender los acontecimientos y relación de datos dentro de una investigación.

Cuadro 2.4: Citas sobre trazabilidad en red semántica "Necesidades"

Cita	Interpretación de la cita
Registros históricos	
[56, 55]	Los autores explican que un producto de una replicación son distintas versiones de paquetes de laboratorio, y como tal, pueden actuar como un registro histórico sobre el evolución de un estudio.
Temporalidad	
[47]	Aquí se destaca la importancia de los detalles procedimentales en donde se resalta la temporalidad de los eventos que permitiría la reconstrucción de hechos suscitados en la experimentación.

Necesidad de motivación

A partir de los contenidos del Cuadro 2.4 se puede definir esta necesidad de la siguiente manera: En la motivación se debe considerar aquellos elementos que agregan restricciones a los intentos de reproducibilidad, como el costo y el esfuerzo requerido por otros investigadores. La aplicación de prácticas, tecnologías y recursos de uso extendido son factores importantes para reducir las barreras y motivar la reproducibilidad.

Cuadro 2.5: Citas sobre motivación en red semántica "Necesidades"

Cita	Interpretación de la cita
Especificar motivaciones	
[57]	Se habla de la falta de motivaciones para “correr una replicación” en IS atribuyendo esto al costo, en términos de recursos y esfuerzo de los experimentadores.
Aplicación de guías y no de estándares	
[54]	Se menciona que, a pesar de las guías provistas por un estudio original, se requiere de una retroalimentación y consultas, no solo por cuestionamientos sobre el conocimiento tácito sino también por la falta de estándares empleados de las herramientas, que actúan más como sugerencias.
Consolidar conocimientos	
[46, 43]	Se aclara que el cuerpo del conocimiento debe ser “usable” y que para esto se necesita de la validación empírica, por lo que es necesario que se apliquen estrategias como la replicación para eso suceda.
[46]	Se urge a la consolidación del cuerpo del conocimiento a través de conclusiones no solo de un estudio, pero si a través de muchos de ellos. La cooperación se relaciona con este objetivo.
Internalización y externalización	
[47, 31]	La internalización de conocimiento en un mismo grupo de investigación suele no exigir procesos de control o seguimiento estrictos y recae principalmente en el uso de mecanismos no formales.
[47]	Existen muchos recursos que se generan durante un proceso de investigación. Su transferencia hacia grupos de investigación externos está altamente relacionado con la calidad ya que exige un proceso de transformación de conocimiento, desde uno tácito a uno explícito, como se detalla en el modelo SECI ¹ .

¹Es el modelo de transferencia de conocimiento que involucra fases cíclicas de “socialización”, “externalización”, “combinación” e “internalización” del conocimiento

2.2.2. Descripción del problema

De acuerdo con el Modelo PSM, una vez planteadas las necesidades, su consecución acarrea diversos problemas que deben ser sorteados mediante estrategias. En el caso de esta investigación, cuyo contexto definido es la experimentación reproducible, se pueden describir los siguientes problemas en el contexto de la reproducibilidad, cuya motivación se expresará en los puntos subsecuentes.

- Habilidades prácticas
- Tipos de conocimiento
- Diseminación y validación
- Administración de datos
- Seguimiento de métodos
- Infraestructura

Problemas de habilidades prácticas

A partir de los contenidos del Cuadro 2.6 se puede definir esta problemática de la siguiente manera: La experimentación y la reproducibilidad se apalancan en diversas habilidades prácticas que abarcan: investigación, gestión, control de calidad, entre otras, como la utilización de herramientas de soporte. Al no contar con las competencias en estos aspectos, la investigación se ve sometida a una ejecución pobremente documentada y gestionada.

Problemas sobre tipos de conocimiento

A partir de los contenidos del Cuadro 2.6 se puede definir esta problemática de la siguiente manera: El conocimiento, para muchos investigadores, recae en la producción de resultados; sin embargo, esto está muy alejado de la realidad. El conocimiento incluye el procesamiento y decisiones sobre la planificación, diseño y ejecución de la investigación; es decir, la información para el correcto uso y evolución de la investigación. El principal problema es que el enfoque se da a la fase de análisis únicamente.

Problemas de diseminación y validación

A partir de los contenidos del Cuadro 2.6 se puede definir esta problemática de la siguiente manera: La diseminación de la investigación habla sobre los mecanismos y objetos de distribución. Principalmente los mecanismos de preferencia son a través de los artículos científicos, sin embargo, estos medios no son suficientes para transferir todo el contenido necesarios para los intereses de todos los tipos de lectores de una investigación.

Consecuentemente, esto dificulta la construcción de un cuerpo de conocimiento que pueda ser validado.

Problemas de administración de datos

A partir de los contenidos del Cuadro 2.6 se puede definir esta problemática de la siguiente manera: Cada instancia de experimentación está sujeta a cambios que van desde la gestión hasta el tratamiento de datos. El problema radica en la utilización de herramientas de naturaleza estática dejadas a selección de los individuos; sin la agregación de la flexibilidad y navegabilidad para soportar los cambios y el crecimiento del conocimiento.

Problemas de seguimiento de métodos

A partir de los contenidos del Cuadro 2.6 se puede definir esta problemática de la siguiente manera: La gestión y tratamiento de los datos de una investigación no siempre se soportan en métodos estrictos. Los investigadores no siguen los métodos elegidos de forma estricta sino más bien los métodos guían a los investigadores en el seguimiento de actividades. Aunque esto reduce el esfuerzo en la ejecución de la investigación, agrega esfuerzo al transferir la información relevante y comprensible a otros investigadores.

Problemas de infraestructura

A partir de los contenidos del Cuadro 2.6 se puede definir esta problemática de la siguiente manera: Las estructuras de soporte a la investigación ayudan al desarrollo y empaquetado de la misma; sin embargo, también pueden volverse una barrera cuando se trata de implementar o instanciar una investigación. La infraestructura puede contemplar aquellas de uso general como las de uso específico y pueden enfocarse a uno o a varios dominios. Al momento de formular infraestructuras de investigación se debe considerar el impacto y el esfuerzo que los usuarios de la investigación requerirán para su despliegue.

Cuadro 2.6: Fragmentos identificados en red semántica "Problemas"

Cita	Interpretación de la cita
------	---------------------------

Habilidades prácticas

Continuación de la tabla 2.6

Cita	Interpretación de la cita
[58, 55, 56]	Se estipula que la experimentación es una actividad que requiere habilidades prácticas. Esto hace difícil conocer si un segundo experimento se ha realizado de forma adecuada para la comparación de resultados haciendo necesario más experimentos para reducir las amenazas a la validez.
Conocimiento	
[46, 50]	Se sostiene que mucha de la información que se esperaría esté contenida en un paquete de replicación, aún seguirá siendo tácita.
[46]	Se insiste en la presencia de conocimiento tácito, que de no ser recopilada, estructurada y registrada de forma explícita no permitirá la colaboración “efectiva” ni la creación de estructuras adecuadas de comunicación.
[29]	Se dice que no todo estudio es reproducible. En la experiencia de los autores cuentan una experiencia propia donde, a pesar de contar con todos los datos y mecanismos para presentar la información, se enfrentaron a “problemas éticos” cuando sanitizaron datos confidenciales a lo que otros investigadores no tendrían acceso. Sin embargo, también existen cuestiones intrínsecas a las publicaciones que dificultan la transferencia de un estudio.
Diseminación con poca validación empírica o poder estadístico	
[29, 31]	Muchos investigadores prominentes han expresado su preocupación sobre los métodos y estándares de la diseminación de las investigaciones en IS. El debate sobre las implicaciones de las actividades de diseminación respecto a la calidad de los procesos de revisión, y por ende a la calidad de los artículos, expone defectos en la evolución del emprendimiento científico.
[31]	La dificultad de crear un cuerpo de conocimiento confiable y generalizable se debe a que la mayoría de los estudios no cuentan con un apropiado mecanismo de validación empírica, más allá de las validaciones de los estudios individuales.
Administración de cambios	

Continuación de la tabla 2.6

Cita	Interpretación de la cita
[53, 50]	Los principales problemas de la investigación es el cambio sobre el tiempo, la elección de modelos de datos y tecnologías de almacenamiento, que generalmente se limitan a decisiones individuales de cada investigador afectado la adaptación y comprensión de los datos en la agenda investigativa.
[47]	La replicación es difícil y los paquetes de laboratorio no son la solución por sí mismos. Sin importar del nivel de experiencia de los experimentadores, todos se enfrentan a las variaciones de supuestos sobre cualquier contexto experimental que un “paquete estático” no es capaz de consolidar.
Seguimiento de métodos	
[13, 59]	La falta de planificación de un experimento hace difícil conseguir resultados relevantes en cualquier investigación. Aunque un “plan es crítico” para una adecuada investigación, no existe unanimidad sobre cómo documentar un plan, más allá de actividades sugeridas por la literatura empírica, y peor aún, otros expertos no siguen adecuadamente dichas guías.
[47, 60]	Debido a que la replicación es difícil debido a la descripción que necesitan de todos los aspectos relevantes, es necesario seguir procesos adecuados y recomendados para considerar la mayor cantidad de información en los contextos de investigación.
Infraestructuras difíciles de reproducir	
[29]	Es entendible que muchos estudios no sean reproducibles debido a que, en sus reimplementaciones, pueden inducirse errores o falta de disponibilidad de recursos especializados.

2.2.3. Conocimiento de dominio de la solución

El siguiente elemento dentro del ámbito PSM es el “conocimiento de dominio de la solución”. Una vez que se han comprendido las necesidades y descrito los problemas del contexto, se realizó una búsqueda de soluciones prescritas a estos requerimientos. Esta búsqueda se la realizó en una base de conocimiento de dominio de la aplicación. El emparejamiento entre problemas y soluciones permitieron generar una serie de alternativas

a considerarse para generar una instancia de solución.

Como conclusión, a través de la exploración del estado del arte se logró revisar cómo se desarrollan las actividades y prácticas, así como las herramientas que se utilizan en la experimentación en ingeniería de software. Se aprendió que existen dominios especializados de experimentación en la que se ha normalizado ciertos procesos y uso de tecnologías; sin embargo, no existe una base de conocimiento sobre la selección de las actividades y herramientas de forma general a manera de solución. Las prácticas más ampliamente extendidas entre los investigadores es el uso de “guías de reproducibilidad” o literatura base para experimentación, pero que no llegan a considerarse como “estándar” de la investigación. En las Figuras 1.8, 1.9 y 1.10 se muestran algunas de las herramientas identificadas en la literatura, que fueron categorizadas por su generalización y su formalidad para entender el impacto de las herramientas en las instancias experimentales. No obstante, estas soluciones no se tienden a orientar de acuerdo con problemas o necesidades de los contextos de reproducibilidad. Las herramientas tienden a soportar las actividades de comunicación, administración de datos e información, y a la gestión de las actividades de investigación. La formalización sobre el uso de herramientas y la aplicación de procesos es elevada en contextos de experimentación especializados, que, de forma general, utilizan plataformas altamente rígidas y poco transferibles a otros dominios de experimentación. Finalmente, dentro de los grupos de investigación se requieren de procesos de transferencia de conocimiento tanto para las inducciones hacia nuevos miembros, así como hacia grupos de investigación externos.

2.2.4. Alternativas

En este apartado, a través del “conocimiento del dominio de la solución” se generan alternativas de solución. Estas alternativas son sometidas a un proceso de control que permiten seleccionar las mejores conforme las consideraciones establecidas en el ámbito de “control”. De esta manera, con la o las alternativas seleccionadas se puede detallar y describir una solución.

A través de la investigación no se pudo determinar que existe una base de conocimiento de soluciones; sin embargo, existen alternativas que suelen usarse en distintos escenarios. Se podría ofrecer una propuesta inicial (Cuadro 2.7) sobre cómo estas alternativas de solución se podrían emparejar a los distintos problemas. Según la Figura 1.3, el nivel en el que se encuentra la EIS es de una transición entre T_b y T_c .

		Tipos de herramientas					
		Prácticas	Plataformas y ecosistemas	Comunicación	Tecnologías de gestión	Modelos de conocimiento	Guías de reporte
Problemas	Habilidades prácticas		✓	✓			
	Tipos de conocimiento			✓		✓	
	Administración de datos	✓	✓	✓	✓	✓	✓
	Seguimiento de métodos	✓	✓			✓	✓
	Infraestructura		✓	✓	✓		

Cuadro 2.7: Propuesta de alternativas de solución vs. problemas

2.2.5. Descripción de la solución

Para detallar la “descripción de la solución” se debe tomar en cuenta la intersección de dos contextos: el contexto de la reproducibilidad y el contexto de la investigación. Para esta actividad se pueden integrar diversas herramientas que den solución a las necesidades y permitan gestionar o contrarrestar las problemáticas determinadas. Esto podrá dar pie a la instanciación de un artefacto que permita ejecutar una investigación reproducible.

2.2.6. Artefacto

En base a lo anteriormente expuesto es posible proponer un artefacto para el contexto de investigación y que incorpore aspectos de calidad como la reproducibilidad. El artefacto entonces incluye es toda la arquitectura de investigación reproducible; es decir, las estructuras lógicas y físicas de un entorno experimental.

2.3. Componente “Control”

El ámbito de “control” es un elemento determinante para aumentar la madurez de una disciplina, como lo es la experimentación en IS. En este ámbito se conjugan criterios de calidad y restricciones que permiten analizar, a través de un algún tipo de modelo, la evaluación de las distintas alternativas de solución. Como se ha mencionado anteriormente,

no se ha podido determinar componentes, elementos o procedimientos de selección. La investigación ha podido determinar que las alternativas (herramientas) son elegidas a través de opiniones y costumbres de los involucrados en una investigación. Sobre esto [58] sostiene que debe existir un marco de trabajo alternativo que considere diversos tratamientos que les permitan a los investigadores escoger los más adecuados de acuerdo con los problemas del estudio empírico. El autor sostiene que la solución debe enfocarse en prácticas de alto nivel enmarcadas en cuatro dimensiones: El realismo, la robustez, los hallazgos y los recursos. Ante esto se proponen los siguientes subcomponentes para el modelo y criterio de calidad del contexto de la investigación reproducible.

2.3.1. Modelo

El “modelo PSM” considera un “modelo” para analizar los componentes de alternativas dentro del ámbito de “control”. El modelo entonces desempeña un proceso analítico de acuerdo con ciertas consideraciones establecidas a través de las necesidades y problemas. Estas consideraciones deben estar incorporadas en el proceso de análisis; sin embargo, también pueden ser valoradas de acuerdo con el proceso de investigación experimental.

Para esta propuesta de solución se propone el uso de modelado de meta proceso. El modelado de meta proceso se particulariza por definir niveles de abstracción de lo que un proceso representa de forma genérica [61] hasta poder desarrollar una instancia ejecutable de este (artefacto). Como parte del desarrollo de un meta proceso que aporte a la obtención de la reproducibilidad de una investigación experimental se ha analizado una representación multinivel que se apoya en el método conocido como CREWS-L'Ecritoire [62].

Este método establece CREWS-L'Ecritoire un “mapa” compuesto de objetivos denominados “intenciones” que se interconectan a través del planteamiento de “estrategias”. Este mapa de intenciones y estrategias podría capturar las necesidades que deben ser satisfechas para la obtención de una investigación reproducible; mientras que las diferentes estrategias podrían hacer referencias a las distintas herramientas, sean desde cuestiones prácticas hasta plataformas tecnológicas, que pueden utilizarse dependiendo de la naturaleza y de las restricciones del contexto. El modelo, como lenguaje gráfico y conceptual, se puede observar en la Figura 2.3. Como resultado de la creación de múltiples estrategias e intenciones se obtendría un grafo multinivel que contaría con todas las características de navegabilidad de procesamiento para el análisis de alternativas de solución y la conformación de un “artefacto”.

Consecuentemente, el modelo a proponer se enfoca en la cobertura del proceso a partir de la especificación de estrategias. La alineación hacia las estrategias permite la evaluación o consideración de distintas alternativas para la realización de una tarea, permitiendo generar un plan de acción consecuente con el o los objetivos perseguidos [63]. El modelo

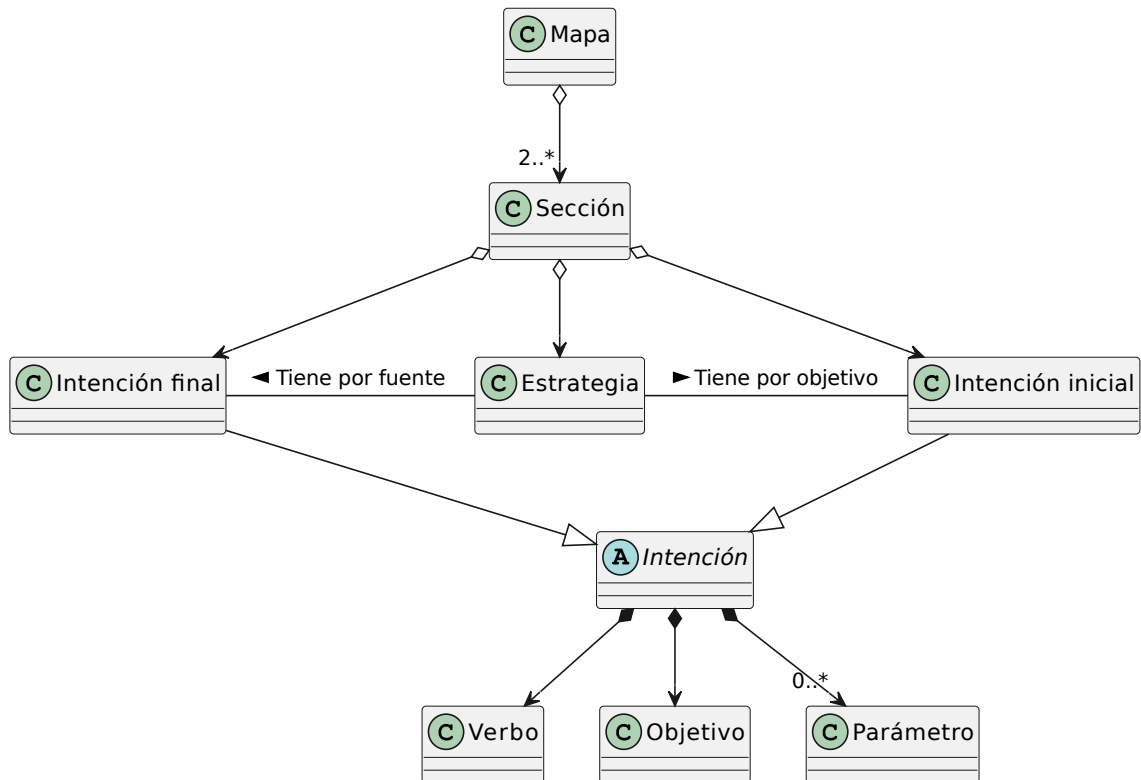


Figura 2.3: Explicación de diagrama de intenciones con UML basado en [4]

debe considerar varias características de calidad como las siguientes [64]:

- **Expresividad**, el grado en que una determinada técnica de modelado es capaz de denotar los modelos de cualquier número y tipo de dominios de aplicación.
- **Arbitrariedad**, el grado de libertad que se tiene al modelar un mismo dominio
- **Idoneidad**, el grado en que una determinada técnica de modelado se adapta a un tipo específico de dominio de aplicación.
- **Comprensibilidad**, la facilidad con la que los participantes entienden la forma de trabajar y la forma de modelar.
- **Coherencia**, el grado en que los sub modelos individuales de una forma de modelado constituyen un todo.
- **Compleitud**, el grado en que todos los conceptos necesarios del dominio de la aplicación están representados en la forma de modelado.
- **Eficiencia**, el grado en que el proceso de modelado utiliza recursos como el tiempo y las personas.
- **Eficacia**, el grado en que el proceso de modelado logra su objetivo.

2.3.2. Criterios de calidad

Como parte del ámbito de control se debe considerar criterios de calidad. En este alcance de investigación se hace énfasis en la reproducibilidad, razón por la que se ha

plateado a lo largo de esta investigación que la reproducibilidad es un atributo de calidad inherente a la investigación; es decir, que debe ser parte de cualquier proceso experimental de forma incondicional. Sin embargo, se ha encontrado muy difícil establecer de qué se compone la reproducibilidad. Se ha mencionado a la reproducibilidad como parte de otros atributos de calidad como la “correctitud” y la “constructibilidad” [29] o como la inclusión de prácticas como la “comunicación” o la “colaboración” [44]. Sin embargo, es factible que ésta se logre a través de otros factores de calidad.

Taxonómicamente hablando, la calidad puede descomponerse en tres elementos: Capacidad, Factor y Métrica [65] . Por lo tanto, se hace necesario establecer la noción de que la capacidad de reproducibilidad se da por factores de calidad que pueden ser medidos de alguna forma. Para extraer posibles factores de calidad de la reproducibilidad se ha realizado una búsqueda de palabras adjetivas en los textos analizados para determinar cuáles son los posibles factores que propician la reproducibilidad.

Consecuentemente, tras revisar las particularidades del contexto (diseminación, la comunicación y la estructuración) en la sección 2.1 es posible extraer los factores de calidad a partir de la síntesis de las características exploradas en las necesidades y problemas del contexto como la cooperación, contenido, validación, trazabilidad y motivación (Sección 2.2.1) dando lugar a las siguientes propuestas conceptuales (ver Figura 2.4):

Auto descripción

La auto descripción puede referirse a la capacidad de un estudio de poder explicarse así mismo. Un estudio debería contemplar un esquema que sea capaz de permitir el entendimiento de su contenido de forma **completa** y documentada a fin de minimizar la dependencia de la **interacción** con los autores; contemplando características como: comunicación y organización [5, 43], procedimientos para la cooperación [45, 46, 47, 48], inclusión detallada de procesos [5, 29, 47, 51, 52], elementos de un reporte de investigación [53, 54], presentación estructurada de resultados [47, 53, 55], presentación estructurada de metodologías [5, 47, 55, 59, 60], y la especificación detallada de motivaciones [57].

Transferibilidad

La transferibilidad se refiere a la capacidad de un estudio de proveer la información utilizada y generada durante su ejecución. Un estudio debe proveer acceso adecuado a la información con el propósito de permitir la **búsqueda** de información **útil**; contemplando características como: la diseminación de información [29], los esquemas de compartición y disponibilidad [29, 31, 43, 46, 49, 50], mecanismos de internalización y externalización [31, 47, 50], el acceso a datos en bruto [5, 29, 50, 51], inclusión de falsos positivos [45],

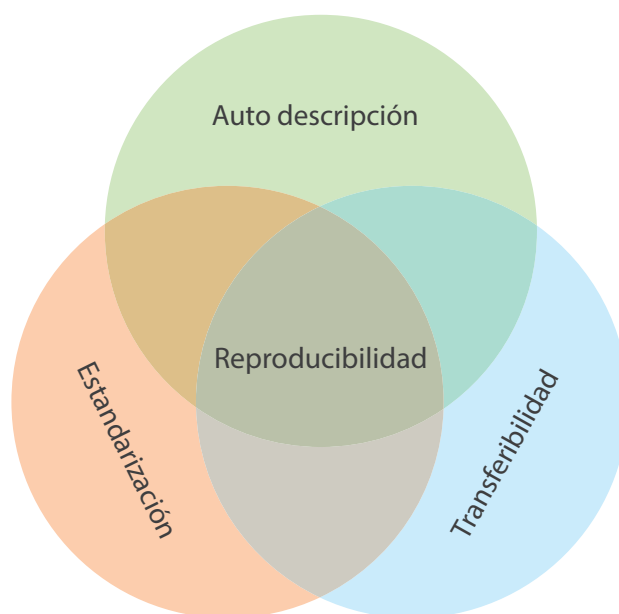


Figura 2.4: Propuesta de factores de reproducibilidad

acceso a procedimientos de toma de decisiones [43, 46, 47, 56], el acceso a registros históricos [56, 55], y especificación de la temporalidad de las actividades [47, 50, 53].

Estandarización

La estandarización se refiere a la capacidad de un estudio de contar con elementos que propicien su **normalización** y **automatización** para que pueda ser compatible e interoperable a las necesidades de otros investigadores; contemplando características como: la codificación y estructuración [5, 44], especificación de tecnologías e infraestructuras colaborativas entre autores y replicadores [24, 47, 55, 56, 58], aplicación de estándares [29, 54], y mecanismos para consolidar y reutilizar conocimientos [43, 46].

2.4. Preguntas de conocimiento para la propuesta

A este punto ya se han establecido elementos necesarios para poder mejorar la madurez del proceso de una investigación experimental que incorpore capacidades de reproducibilidad. Sin embargo, como se recomienda en el proceso metodológico usado en esta investigación, es necesario poder responder preguntas de conocimiento sobre el artefacto (modelo) a diseñar para el contexto de la reproducibilidad (ver Figura 1.12). Consecuentemente se han planteado las siguientes preguntas y respuestas sobre el artefacto.

2.4.1. ¿Qué tan genérico o específico debe ser el modelo?

El grado de generalización o especialización de las herramientas suele estar relacionado con el dominio de la investigación. Un dominio maduro puede establecer una serie de procedimientos, variables, taxonomías, definiciones, etc. En este sentido, las herramientas para dominios maduros son difíciles de migrar a otros dominios. Los dominios menos maduros, por otro lado, utilizan herramientas más generalizadas, lo que da a los investigadores mucha libertad para adaptar sus necesidades, pero conduce al uso de enfoques menos formales y por lo tanto no son seguidos de forma estricta.

Podría ser extremadamente difícil desarrollar una herramienta única que se ajuste a todas las necesidades en cualquier dominio y contexto. El modelo entonces debe proporcionar un alto nivel de especificación para el ciclo de vida de la investigación en lugar de procedimientos de dominio específico. Toda experimentación persigue una meta en el proceso de investigación, sin importar el dominio. Esto significa que hay requisitos establecidos por el nivel más alto de abstracciones de la investigación del modelo científico.

En este nivel alto de especificación existen procesos transversales para cualquier investigación experimental que son necesarios para propiciar la reproducibilidad. Por lo tanto, las diversas alternativas de herramientas de soporte pueden ser aplicadas de acuerdo con las necesidades de cada dominio, ya que unas serán más apropiadas que otras, pero la necesidad seguirá siendo la misma.

2.4.2. ¿Qué tan formalizado debe ser el modelo?

El grado de formalidad está íntimamente relacionada con la pregunta anterior. La formalidad se refiere a características como la sistematización, la organización y la trazabilidad. Los dominios maduros son ampliamente explorados en las investigaciones científicas, consecuentemente incorporan una mayor formalización en las características mencionadas anteriormente. En estos dominios, el conocimiento también es más explícito, por lo que un investigador sabe cómo documentar o registrar información importante y relevante.

Por otro lado, existen mecanismos informales que facilitan la ejecución de tareas inmersas en una investigación, pero a costa de control. Es importante reconocer que una investigación puede involucrar diferentes niveles de formalidad dependiendo si el enfoque está dirigido a humanos o a tecnologías. Adicionalmente se debe considerar que una alta formalidad podría ir en detrimento de que otros investigadores estén abiertos a involucrarse en una investigación.

El modelo debe establecer un nivel de formalidad en cuanto a qué debe ser implementado mas no en el cómo debe ser implementado algo. Esto da a lugar a una

herramienta lo suficientemente flexible para adaptarse a cualquier contexto.

2.4.3. ¿El modelo debe ser monolítico o modular?

La exploración de la literatura muestra muchas plataformas centradas en el dominio y en la tecnología. Estas plataformas no son fáciles de trasladarse hacia dominios de conocimiento distintos a donde fueron concebidos. Muchos estudios reportaron las dificultades que surgen cuando estas plataformas se aplican en otros dominios o contextos, evidenciando que las plataformas requieren muchos cambios o adaptaciones costosas, adicionando un mayor coste y esfuerzo a la tarea de comprensión y reutilización de una investigación.

Para obtener un modelo flexible, las herramientas de soporte deben ser modulares y soportarse en diferentes componentes, paquetes o tecnologías. Un modelo conceptual de alto nivel puede ser adaptado y su composición multinivel puede enfocarse a diversos dominios sin perjudicar su utilidad.

2.4.4. ¿Cómo se puede integrar las necesidades en el modelo?

En concordancia a la respuesta anterior, el modelo debe facilitar la caracterización de las necesidades en relación con cada actividad dentro de un proceso de investigación. Al igual que en el modelo PSM, el modelo podría llegar a incluir, de forma modular, los contextos de solución asociadas a cada dominio de problemas de experimentación, generando tantas alternativas como sea necesario. El contexto del problema podría coincidir con un conjunto de intenciones del dominio específicos de aplicaciones experimentales. Desde luego, las intenciones principales e inmutables deben estar relacionadas con las necesidades y problemas definidos anteriormente sobre el contexto de reproducibilidad de experimentos.

En este escenario el modelo a proponer consiste en un grafo capaz de soportar un esquema multinivel permitiendo describir desde actividades de alta abstracción (meta procesos) hasta detalles operativos relacionados a ejecuciones operativas detalladas correspondientes a dominios específicos. El grafo tiene propiedades que pueden ayudar en el proceso de análisis de acuerdo con pesos, por ejemplo.

2.4.5. ¿Es posible validar el modelo?

La experimentación en IS es aplicable en muchos escenarios de ciclo de vida de desarrollos de software; también es aplicable a enfoques tecnológicos o humanos dando a lugar a distintos niveles de automatización. Esta variedad induce a un problema operativo

y terminológico que es difícil estandarizar. Inclusive, se cuenta con innumerables artículos que evalúan propuestas y presentan, en la mayoría de los casos, cambios o adaptaciones para que se apliquen a otros dominios.

Sin embargo, si se parte desde una perspectiva de factores de calidad, es posible enfocar el modelo, al menos en los niveles altos, a describir habilitantes de la reproducibilidad que son transversales a todos los dominios de experimentación. Las propuestas, que son de índole cualitativa, podrían ser validadas a través de un experimento que evalúe la relación causa-efecto de la aplicación del modelo sobre el resultado percibido de reproducibilidad.

Es posible evaluar la calidad de un modelo [64]; sin embargo, se ha enfocado en la efectividad como valor más significativo.

2.4.6. ¿Cuánto esfuerzo exige la aplicación del modelo?

El modelo debe caracterizar las necesidades de la reproducibilidad y ser capaz de incluir necesidades de distintos dominios de experimentación. No debe tratarse de proponer mecanismos distintos sino mostrar distintas estrategias para poder producir una investigación experimental reproducible. Si se plantea un esquema tipo grafo, se esperaría que el proceso de aprendizaje y adopción sea lo menos intrusivo para que pueda ser destinado a personas que se están iniciando en las actividades de investigación o experimentación.

Cabe recalcar que se persigue que el modelo propuesto incorpore características de comprensibilidad [64].

2.4.7. ¿Cómo se incorpora la reproducibilidad en el modelo?

El modelo PSM, en el ámbito de control, especifica los criterios de calidad. Los criterios de calidad que se están proponiendo pueden ser incluidos en el modelo gracias que los factores de calidad pueden ser análogos a intenciones de reproducibilidad manejada por el meta proceso de intenciones y estrategias.

2.5. Definición de la propuesta de solución: Modelo de estrategias e intenciones

A partir de la información que se ha contemplado sobre necesidades, problemas, objetivos, y demás requerimientos, se ha planteado el siguiente artefacto que consiste en un modelo de estrategias e intenciones. El modelo es un grafo multinivel que busca incorporar

mecanismos para seleccionar las herramientas que mejor se ajusten a los requerimientos de un contexto de experimentación. En primera instancia se cubrirán aquellas estrategias enfocadas a la incorporación de la reproducibilidad de una investigación. A continuación, se presenta la definición del modelo, su estructura y la presentación del artefacto. En la Figura 2.5 se muestra la interrelación entre los puntos más característicos del contexto de experimentación y cómo se espera que los criterios de calidad propuestos aporten a los procesos de selección de herramientas idóneas.

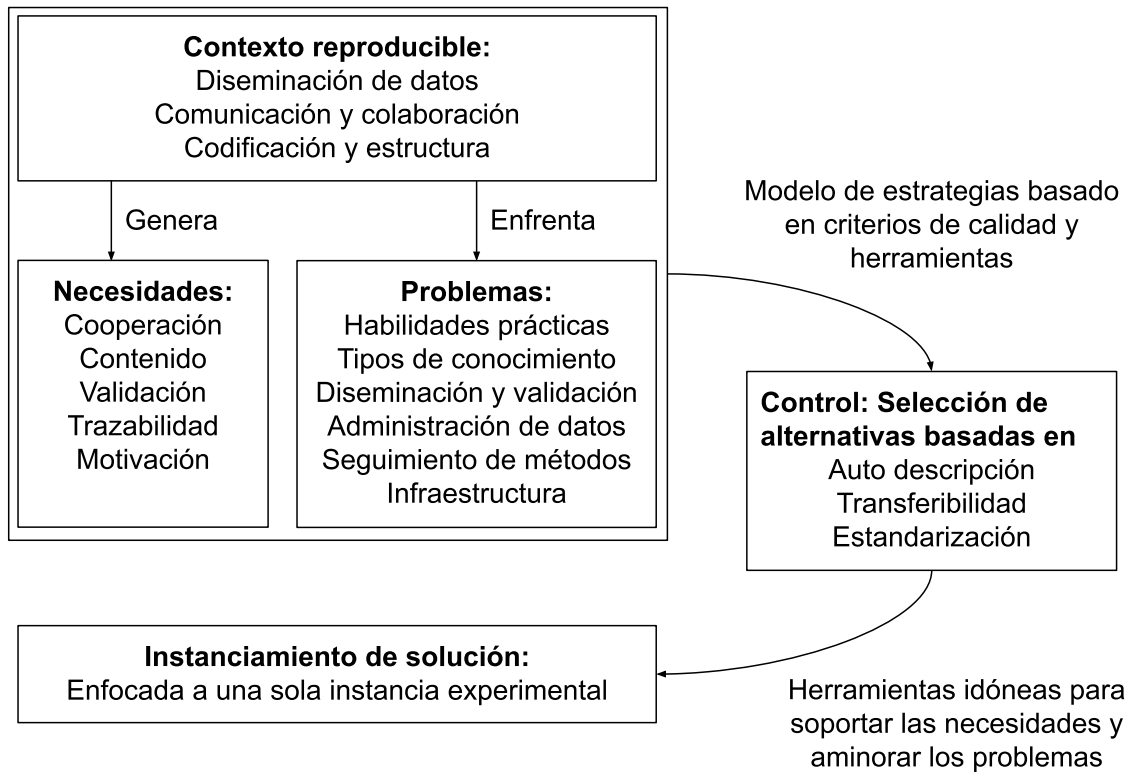


Figura 2.5: Relación entre los elementos PSM descritos

2.5.1. Definición de la estructura del modelo

El modelo de reproducibilidad de experimentos es un grafo multinivel que se compone de distintas estrategias sobre la incorporación de factores de reproducibilidad a los procesos de experimentación. Las estrategias pueden ser ejecutadas de forma no excluyente, es decir, se pueden aplicar tantas estrategias para alcanzar el objetivo de la investigación reproducible.

El modelo de estrategias e intenciones es capaz de expresar distintos tipos de aplicación centrándose en la triplete $\langle Intencion_{inicial}, Intencion_{final}, Estrategia \rangle$. De esta manera, una “intención final” puede ser alcanzada por distintas estrategias o un conjunto de ellas, dotando al proceso de la arbitrariedad que necesita un investigador. Un investigador podrá

Cuadro 2.8: Niveles del modelo

Nivel	Descripción
1	Factores de reproducibilidad
2	Características de implementación
3	Herramientas de soporte
n	Parámetros de configuración

ser capaz de trazar una ruta idónea a sus objetivos de reproducibilidad en su dominio de investigación. Gracias a la característica multinivel del modelo, se persigue la comprensión y coherencia para interpretar la selección de alternativas que debe evaluar el investigador. Para presentar este modelo multinivel se sugieren para esta investigación los niveles mostrados en el Cuadro 2.8.

Como características finales, se ha propuesto un modelo que abarca los conceptos necesarios que se han investigado sobre la reproducibilidad. La eficiencia que se espera lograr con la aplicación del modelo es la de reducir el esfuerzo subyacente en obtener una investigación reproducible. La eficacia del modelo será una medida subjetiva dada las implicaciones prácticas que surgen al momento de intentar reproducir un experimento. Se ha creído imperativo que las estrategias de reproducibilidad deban ser identificadas y aplicadas desde las etapas más tempranas de las actividades de investigación; es decir, desde la concepción de esta.

En esta propuesta, se exploraron los niveles 1 y 2. El nivel 1 hace referencia a la auto descripción, la transferibilidad y la estandarización. El nivel 2 especifica diferentes requerimientos que se han identificado en cada factor de reproducibilidad. En el nivel 3 se presentan distintas herramientas que han podido ser recolectadas a través de las diferentes técnicas de exploración y que corresponden a el nivel 2. En esta investigación no se contempla aún el análisis de posibles niveles inferiores ya que se requiere una investigación particular de acuerdo con las distintas herramientas de soporte. Los niveles 1 y 2 se presentan en los puntos siguientes.

2.5.2. Estructura de nivel 1: Reproducibilidad

El nivel superior del modelo representa los factores de calidad que habilitan la reproducibilidad. En este nivel, sin importar el dominio de aplicación del experimento, es posible optar por una estrategia o un conjunto de ellas para alcanzar el mayor grado de reproducibilidad posible. Para esto se han considerado dos intenciones que generan tres secciones estratégicas dando como resultado la Figura 2.6.

Intenciones de reproducibilidad (I)

- I_1 : Formular un experimento
- I_2 : Obtener una investigación reproducible

Estrategias de reproducibilidad (S)

- S_1 : Elaborando una investigación auto descriptible
- S_2 : Planificando la transferibilidad del conocimiento
- S_3 : Aplicando mecanismos estandarizados

Secciones de mapa de reproducibilidad (MS)

- MS_0 : $\langle I_{Start}, I_1, \text{Inicio} \rangle$
- MS_1 : $\langle I_1, I_2, \text{Elaborando una investigación reproducible} \rangle$
- MS_2 : $\langle I_1, I_2, \text{Planificando la transferibilidad del conocimiento} \rangle$
- MS_3 : $\langle I_1, I_2, \text{Aplicando mecanismos estandarizados} \rangle$
- MS_4 : $\langle I_{Stop}, I_2, \text{Fin} \rangle$

2.5.3. Estructura de nivel 2: Auto descripción

En el nivel dos se presentan las necesidades y problemas que se han identificado para el modelo PSM y que están relacionadas al factor de auto descripción como se ha mencionado en la Sección 2.3.2. Para esto se han considerado cinco intenciones que generan trece secciones estratégicas dando como resultado la Figura 2.7.

Intenciones de auto descripción (I)

- $I_{1.1}$: Formular un experimento
- $I_{1.2}$: Estructurar y organizar la información de la investigación
- $I_{1.3}$: Establecer navegabilidad en la documentación
- $I_{1.4}$: Protocolizar el desarrollo de la investigación
- $I_{1.5}$: Obtener una investigación reproducible

Estrategias de auto descripción (S)

- $S_{1.1}$: Describiendo meta datos de la investigación
- $S_{1.2}$: Enfocándose en un proceso metodológico de investigación
- $S_{1.3}$: Enfocándose a los tipos de artefactos de investigación

- S*_{1.4}: Redactando instrucciones para recorrer la información
- S*_{1.5}: Añadiendo un índice de contenidos
- S*_{1.6}: Entrelazando artefactos y contenidos
- S*_{1.7}: Categorizando artefactos
- S*_{1.8}: Planificando explícitamente un plan de experimentación
- S*_{1.9}: Definiendo protocolos de ejecución
- S*_{1.10}: Definiendo protocolos para registro de novedades
- S*_{1.11}: Incorporando recursos adicionales para la comprensión en lenguaje natural
- S*_{1.12}: Controlando la incorporación de elementos de navegabilidad
- S*_{1.13}: Documentando la investigación de acuerdo con los protocolos

Secciones de mapa de auto descripción (MS)

- MS*_{1.1}: < *I*_{1.1}, *I*_{1.5}, Describiendo meta datos de la investigación >
- MS*_{1.2}: < *I*_{1.1}, *I*_{1.2}, Enfocándose en un proceso metodológico de investigación >
- MS*_{1.3}: < *I*_{1.1}, *I*_{1.2}, Enfocándose a los tipos de artefactos de investigación >
- MS*_{1.4}: < *I*_{1.2}, *I*_{1.3}, Redactando instrucciones para recorrer la información >
- MS*_{1.5}: < *I*_{1.2}, *I*_{1.3}, Añadiendo un índice de contenidos >
- MS*_{1.6}: < *I*_{1.2}, *I*_{1.3}, Entrelazando artefactos y contenidos >
- MS*_{1.7}: < *I*_{1.2}, *I*_{1.3}, Categorizando artefactos >
- MS*_{1.8}: < *I*_{1.2}, *I*_{1.4}, Planificando explícitamente un plan de experimentación >
- MS*_{1.9}: < *I*_{1.2}, *I*_{1.4}, Definiendo protocolos de ejecución >
- MS*_{1.10}: < *I*_{1.2}, *I*_{1.4}, Definiendo protocolos para registro de novedades >
- MS*_{1.11}: < *I*_{1.2}, *I*_{1.4}, Incorporando recursos adicionales para la comprensión en leng. natural >
- MS*_{1.12}: < *I*_{1.3}, *I*_{1.5}, Controlando la incorporación de elementos de navegabilidad >
- MS*_{1.13}: < *I*_{1.4}, *I*_{1.5}, Documentando la investigación de acuerdo con los protocolos >

2.5.4. Estructura de nivel 2: Transferibilidad

En el nivel dos se presentan las necesidades y problemas que se han identificado para el modelo PSM y que están relacionadas al factor de transferencia como se ha descrito en la Sección 2.3.2. Para esto se han considerado seis intenciones que generan doce secciones estratégicas dando como resultado la Figura 2.8.

Intenciones de transferibilidad (I)

- I*_{2.1}: Formular un experimento
- I*_{2.2}: Determinar canales de comunicación y soporte
- I*_{2.3}: Definir un repositorio instanciable
- I*_{2.4}: Gestionar el conocimiento
- I*_{2.5}: Brindar acceso a la investigación
- I*_{2.6}: Obtener una investigación reproducible

Estrategias de transferibilidad (S)

- S*_{2.1}: Estableciendo correos electrónicos institucionales
- S*_{2.2}: Integrando formularios de contacto
- S*_{2.3}: Empleando plataformas de comunicación o redes sociales
- S*_{2.4}: Definiendo esquemas de almacenamiento
- S*_{2.5}: Especificando el tipo y procedimientos de gestión de contenido
- S*_{2.6}: Describiendo el Know-how de las actividades de investigación
- S*_{2.7}: Especificando los artefactos utilizados o generados en la investigación
- S*_{2.8}: Empaquetando la información
- S*_{2.9}: Empleando archivos abiertos, parametrizables y ejecutables
- S*_{2.10}: Difundiendo los canales de comunicación en el repositorio
- S*_{2.11}: Dando a conocer los protocolos de acceso a repositorio
- S*_{2.12}: Distribuyendo la arquitectura de conocimiento configurable
- S*_{2.13}: Documentando y difundiendo la investigación a través de artículos y otros medios

Secciones de mapa de transferibilidad (MS)

- MS*_{2.1}: < *I*_{2.1}, *I*_{2.2}, Estableciendo correos electrónicos institucionales >
- MS*_{2.2}: < *I*_{2.1}, *I*_{2.2}, Integrando formularios de contacto >
- MS*_{2.3}: < *I*_{2.1}, *I*_{2.2}, Empleando plataformas de comunicación o redes sociales >
- MS*_{2.4}: < *I*_{2.1}, *I*_{2.3}, Definiendo esquemas de almacenamiento >
- MS*_{2.5}: < *I*_{2.1}, *I*_{2.3}, Especificando el tipo y procedimientos de gestión de contenido >
- MS*_{2.6}: < *I*_{2.1}, *I*_{2.4}, Describiendo el Know-how de las actividades de investigación >
- MS*_{2.7}: < *I*_{2.1}, *I*_{2.4}, Especificando los artefactos utilizados o generados en la investigación >
- MS*_{2.8}: < *I*_{2.1}, *I*_{2.4}, Empaquetando la información >
- MS*_{2.9}: < *I*_{2.1}, *I*_{2.4}, Empleando archivos abiertos, parametrizables y ejecutables >
- MS*_{2.10}: < *I*_{2.2}, *I*_{2.6}, Difundiendo los canales de comunicación en el repositorio >
- MS*_{2.11}: < *I*_{2.3}, *I*_{2.5}, Dando a conocer los protocolos de acceso a repositorio >
- MS*_{2.12}: < *I*_{2.4}, *I*_{2.5}, Distribuyendo la arquitectura de conocimiento configurable >
- MS*_{2.13}: < *I*_{2.5}, *I*_{2.6}, Documentando y difundiendo investigación con artículos y otros medios >

2.5.5. Estructura de nivel 2: Estandarización

En el nivel dos se presentan las necesidades y problemas que se han identificado para el modelo PSM y que están relacionadas al factor de estandarización como se ha descrito en la Sección 2.3.2. Para esto se han considerado seis intenciones que generan doce secciones estratégicas dando como resultado la Figura 2.9.

Intenciones de estandarización (I)

- I*_{3.1}: Formular un experimento
- I*_{3.2}: Implementar infraestructuras de distribución de propósito general
- I*_{3.3}: Instanciar experimentos en plataformas especializadas
- I*_{3.4}: Redactar reportes de investigación
- I*_{3.5}: Definir arquitectura de investigación
- I*_{3.6}: Obtener una investigación reproducible

Estrategias de estandarización (S)

- S*_{3.1}: Aumentando la capacidad de comunicación con lenguaje natural
- S*_{3.2}: Implementando sistemas de gestión de conocimiento
- S*_{3.3}: Implementando Software Product Lines para experimentos
- S*_{3.4}: Configurando plataforma de experimentación
- S*_{3.5}: Utilizando lenguajes de escritura colaborativa
- S*_{3.6}: Recomendando estrategias de implementación y replicación
- S*_{3.7}: Usando archivos cerrados como anexos a la investigación
- S*_{3.8}: Usando lenguajes de marcado para el contenido de la investigación
- S*_{3.9}: Usando archivos de texto plano para almacenar datos de la investigación
- S*_{3.10}: Configurando una plataforma de reproducibilidad
- S*_{3.11}: Divulgando la investigación mediante esquemas estructurados de reporte
- S*_{3.12}: Presentando instrucciones para instanciación

Secciones de mapa de estandarización (MS)

- MS*_{3.1}: < *I*_{3.1}, *I*_{3.6}, Aumentando la capacidad de comunicación con lenguaje natural >
- MS*_{3.2}: < *I*_{3.1}, *I*_{3.2}, Implementando sistemas de gestión de conocimiento >
- MS*_{3.3}: < *I*_{3.1}, *I*_{3.2}, Implementando Software Product Lines para experimentos >
- MS*_{3.4}: < *I*_{3.1}, *I*_{3.3}, Configurando plataforma de experimentación >
- MS*_{3.5}: < *I*_{3.1}, *I*_{3.4}, Utilizando lenguajes de escritura colaborativa >
- MS*_{3.6}: < *I*_{3.1}, *I*_{3.6}, Recomendando estrategias de implementación y replicación >
- MS*_{3.7}: < *I*_{3.4}, *I*_{3.5}, Usando archivos cerrados como anexos a la investigación >
- MS*_{3.8}: < *I*_{3.4}, *I*_{3.5}, Usando lenguajes de marcado para el contenido de la investigación >
- MS*_{3.9}: < *I*_{3.4}, *I*_{3.5}, Usando archivos de texto plano para registrar datos de >
- MS*_{3.10}: < *I*_{3.3}, *I*_{3.5}, Configurando una plataforma de reproducibilidad >
- MS*_{3.11}: < *I*_{3.4}, *I*_{3.6}, Divulgando la investigación mediante reportes estructurados >
- MS*_{3.12}: < *I*_{3.5}, *I*_{3.6}, Presentando instrucciones para instanciación >

2.6. Supuestos relacionados al planteamiento

Durante el diseño del modelo se han considerado algunos supuestos sobre la problemática de la reproducibilidad como los siguientes:

- Existen muchas propuestas para soportar las actividades de investigación; es por esto que se cree conveniente enfocarnos el proceso de selección de alternativas basadas en un esquema de valoración acorde al modelo PSM.
- Se cree necesario contar con un catálogo de alternativas que se acoplen a las características del nivel 2 a fin de mejorar el proceso de selección. Este catálogo podría estar incluido dentro del modelo.
- El modelo pretende ayudar a investigadores que se están iniciado en las actividades de experimentación a mejorar su proceso de selección de estrategias. A diferencia de ellos, los expertos ya cuentan con un modelo tácito sobre el manejo de la reproducibilidad.

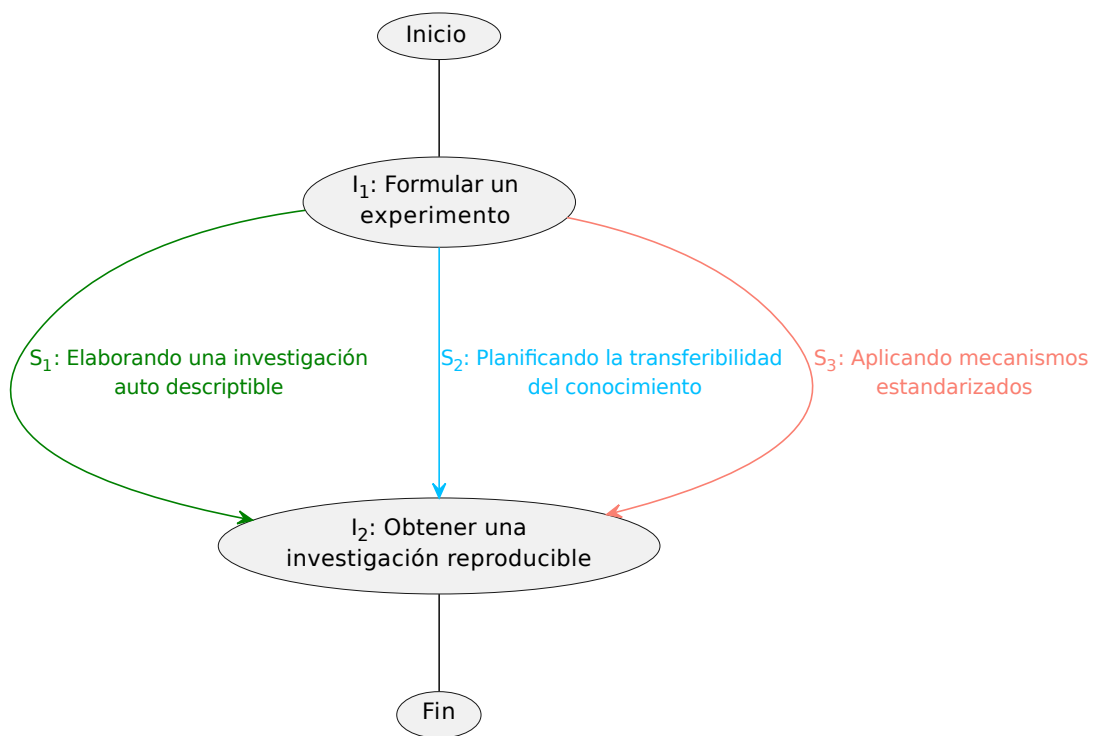


Figura 2.6: Nivel 1 de modelo

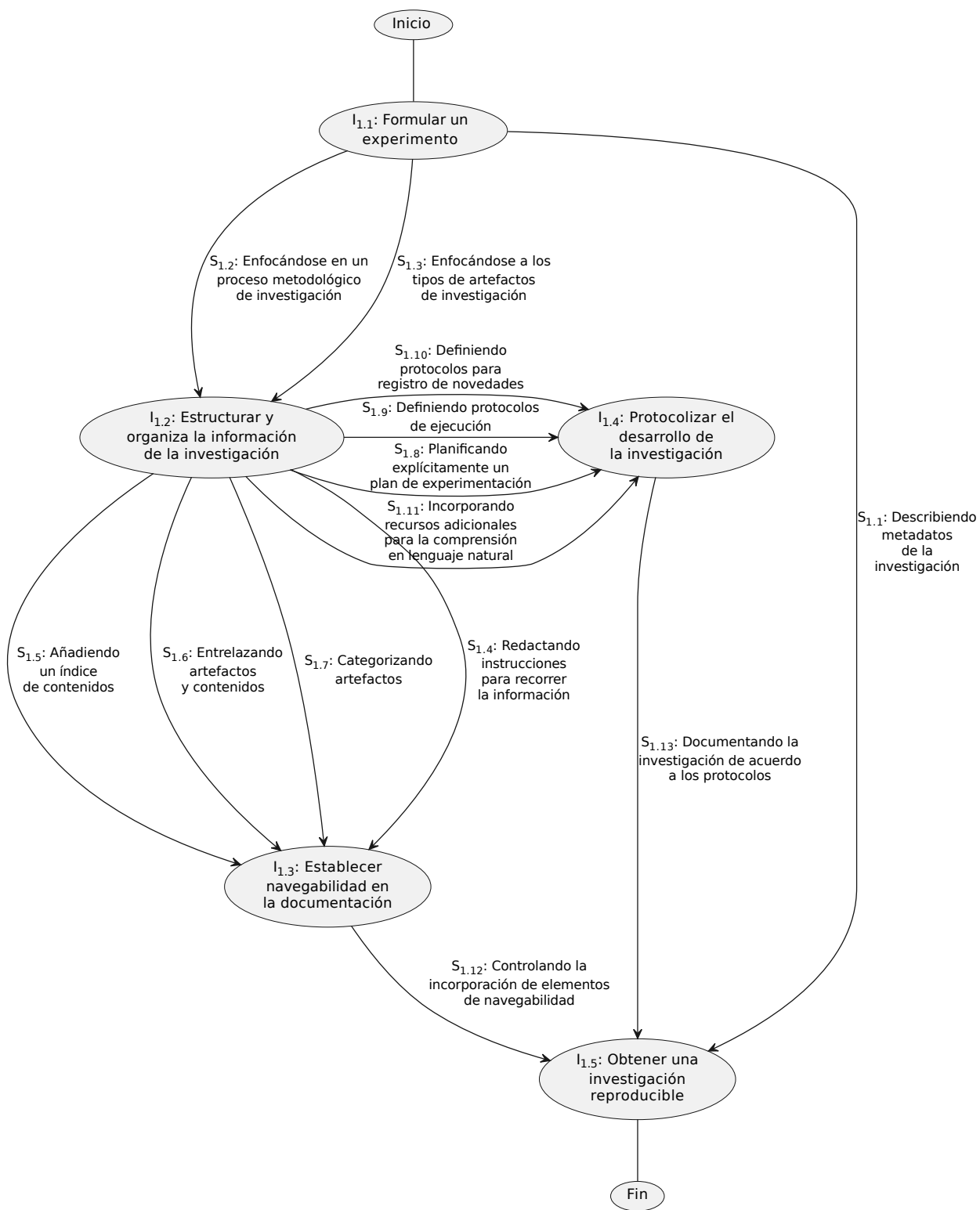


Figura 2.7: Nivel 2 de modelo para auto descripción

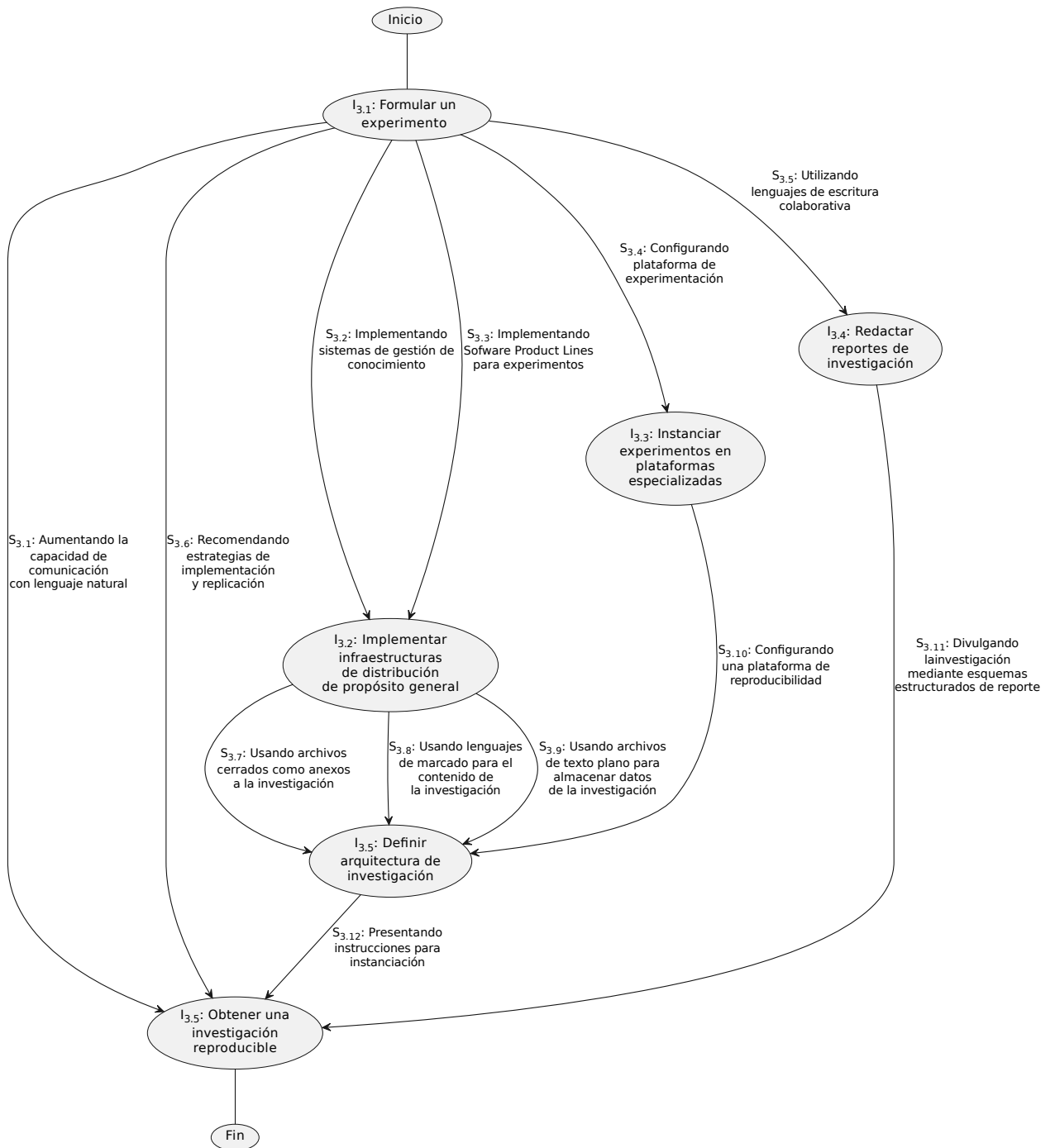


Figura 2.9: Nivel 2 de modelo para estandarización

Capítulo 3

Evaluación de la propuesta: Experimento

El proceso metodológico DS incluye la fase de validación de tratamiento. Principalmente, esta etapa tiene como objetivo verificar la contribución del modelo planteado respecto con los interesados dentro del contexto, o sea a los involucrados dentro del área de la reproducibilidad de experimentos [3]. En esta etapa destacan cuatro preguntas respecto a la validación del modelo de reproducibilidad o artefacto: (i) ¿El modelo de reproducibilidad produce los efectos deseados? (ii) ¿El modelo de reproducibilidad compensa diferentes procesos planteados? (iii) ¿Qué sensibilidad tiene el modelo de reproducibilidad hacia diferentes contextos? (iv) ¿El modelo de reproducibilidad produce efectos que satisfacen los requerimientos? Estas preguntas deben ser contestadas con la ejecución de un proceso de investigación.

Considerando la dificultad que existe para implementar y evaluar los efectos del modelo en distintos escenarios reales, se consideró la utilización del método de experimentación. La ejecución del experimento también brindará la oportunidad de poner en práctica el mismo modelo y de poder capturar diversas experiencias sobre su utilización. Para empezar, se define el proceso que guiará tanto la ejecución como la documentación del experimento. Los lineamientos de la ejecución se basaron en la literatura referencial de Montgomery [66], la especificación orientada a la IS de Wohlin [8] y la taxonomía sintetizada de Goulao & Brito [4]; siendo esta última la que guió la especificación de los elementos del experimento. De esta manera, las secciones que siguen este capítulo contemplan las secciones conforme a la Figura 3.1.

Adicionalmente, y en consecuencia con modelo propuesto, se realizó un análisis sobre distintas plataformas para gestión de conocimiento de este experimento. Se escogió una plataforma de propósito general llamada Obsidian¹. Entre sus características se destacan

¹Obsidian es una aplicación de base de conocimiento que administra notas en es editor Markdown para

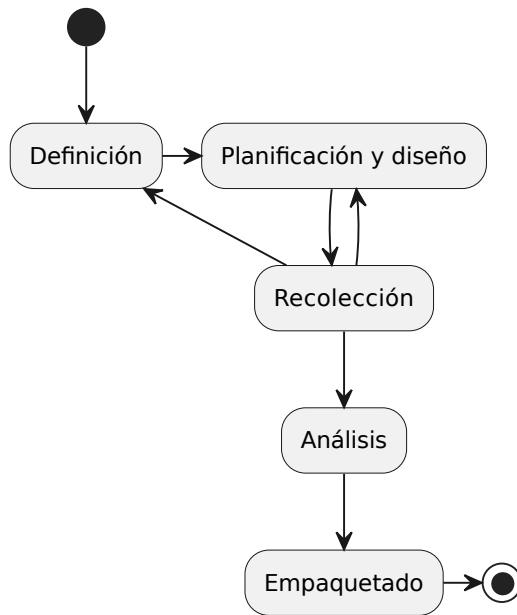


Figura 3.1: Diagrama de secuencia de proceso de experimentación aplicado [4]

el uso de archivos Markdown² y la posibilidad de trabajar de forma Stand-alone, deslindando de la responsabilidad y dependencia a un servidor o servicio. Con estas dos definiciones se planteó y ejecutó el experimento descrito a continuación.

3.1. Definición del experimento

El experimento debe concebirse aclarando el propósito de investigación. En esta etapa se explora el problema que a abordar y también se analiza el impacto de este. Con esta información se plantean los objetivos del experimento. A continuación, se detallan estos puntos.

3.1.1. Planteamiento del problema del experimento

La replicación de un experimento es un proceso que demanda un alto nivel de esfuerzo. Este esfuerzo se relaciona principalmente a retos que un investigador debe enfrentar para analizar, comprender y ejecutar una investigación “original”. Muchos experimentos no pueden ser replicados porque carecen de características que no fueron consideradas desde las etapas tempranas de su concepción, entorpeciendo el entendimiento y ejecución de los replicadores. De forma indirecta, es posible relacionar este fenómeno con una carencia

crear redes de información. Está disponible en obsidian.md

²Markdown es un lenguaje de marcado de licencia DSB que se apoya en el uso de archivos de texto plano y que se utiliza como complemento en muchas plataformas y lenguajes de programación. La información complementaria puede ser accedida en daringfireball.net/projects/markdown/

de calidad que provoca una escasa cantidad de publicaciones reproducidas, fenómeno evidenciado en distintas revisiones de literatura.

Las necesidades y los problemas encontrados en el contexto de la reproducibilidad de experimento fueron examinados a través de revisiones de literatura, una ejecución propia y a través de entrevistas a experimentadores. El cuadro 3.1 resume estos elementos, mismos que se relacionan con prácticas de gestión, comunicación y transferencia de conocimiento. Estos elementos fueron considerados como parte del “modelo problema-solución” o PSM (ver Figura 1.2).

Cuadro 3.1: Etapa de intervención

Ítem	Descripción
Necesidades	<p>Propiciar la cooperación entre investigadores.</p> <p>Gestionar adecuadamente el contenido de toda la investigación.</p> <p>Validar el proceso y datos generados en una investigación.</p> <p>Trazar históricamente las tomas de decisiones de la investigación.</p> <p>Motivar la replicación de la investigación brindando facilidades operativas de la investigación.</p>
Problemas	<p>Las habilidades prácticas requeridas para comunicar y entender información relevante en el ámbito experimental.</p> <p>La existencia de conocimiento tácito sobre la toma de decisiones durante la investigación.</p> <p>Limitaciones en los procesos y prácticas al administrar y diseminar hallazgos.</p> <p>El seguimiento riguroso de estándares, guías o estructuras enfocadas en el reporte de la investigación.</p> <p>La dificultad para recrear arquitecturas tecnológicas, infraestructuras o condiciones de un experimento.</p>

3.1.2. Impacto en el contexto de experimentación

Como se ha mencionado con anterioridad, la baja cantidad de publicaciones replicadas impacta directamente a la conformación del conocimiento científico. El conocimiento solamente puede ser generalizado si las hipótesis son verificadas bajo diferentes contextos de experimentación. Al no contar con esquemas reproducibles en las investigaciones, que incluye a los experimentos, se dificulta que un conocimiento en particular pueda adquirir el carácter de “científico”. En esta investigación doctoral se ha descrito las cualidades que podrían ser consideradas como factores críticos de la reproducibilidad. Es así como las

Cuadro 3.2: Estructuración del objetivo

Elemento	Valor
Calidad (Preocupación)	Efectividad del modelo
Propósito (Por qué)	Planificación de experimento reproducible
Objeto (Qué)	Estrategias de reproducibilidad
Perspectiva (Quién)	Experimentadores no iniciados

malas prácticas en una investigación repercuten en la calidad de la misma, afectando a (i) la auto descripción de la investigación, (ii) su transferencia de conocimiento relevante, y (iii) la estandarización de la operación de la misma. Consecuentemente, el impacto a la calidad producida por malas prácticas se pueden considerar negativos en:

- Los procesos de externalización de una investigación, donde se requiere la transferencia del conocimiento para su reutilización o validación. A diferencia de los procesos de investigación interna, esto no es tan evidente debido a que los grupos de investigación suelen establecer esquemas propios o ya cuentan con un conocimiento heredado.
- Los procesos de investigación en dominios o contextos donde los investigadores cuentan con mucha experiencia, necesidades, procesos y problemas se gestionan de forma informal y tácita.

3.1.3. Objetivos del experimento

Objetivo general

Validar la efectividad del modelo de estrategias de reproducibilidad propuesto para obtener un experimento reproducible a través de la planificación de un experimento por parte de experimentadores no iniciados. (El Cuadro 3.2 muestra objetivo estructurado)

Objetivos específicos

- Esclarecer los elementos relevantes de la gestión y ejecución del experimento que deben considerarse para la auto descripción de un experimento reproducible.
- Maximizar la comprensión de una investigación experimental a través de la estructuración adecuada del conocimiento y de la información.

3.2. Planificación y diseño del experimento

En el punto anterior se definió el contexto y su problemática como la reproducibilidad en los experimentos y sus problemas para la gestión, comunicación y transferencia de conocimiento. A partir de estos elementos, en esta sección se define el diseño del experimento. Esta definición involucra la definición de los parámetros contextuales, los sujetos de experimentación, las hipótesis, las métricas, el tipo de diseño, y las principales actividades del experimento, desde la recolección hasta el empaquetado de todos los productos de la investigación. Esta etapa, en la práctica, es cíclica (ver Figura 3.1), lo que significa que su especificación atravesó diversas interacciones y detalles conforme se ejecutan las fases de recolección de datos. A continuación, se describe la especificación final de esta etapa.

3.2.1. Parámetros del experimento

El experimento debe considerar los parámetros de entrada y de salida, así como el contexto que delimitará, no solo la ejecución del experimento, sino también la interpretación y generalización de los resultados. Como primera instancia, se decidió validar la eficacia del modelo respecto a los factores de reproducibilidad, tal como fue planteado en la sección 2.3.1. Para esto, las entradas del proceso serán los participantes, que se detallarán en la Sección 3.2.2, así como el artefacto propuesto en esta investigación. Finalmente, se espera que el contexto de replicación de experimentos en IS pueda ser genérico; sin embargo, la ejecución no podrá ser llevada a cabo en un entorno real debido al tiempo que llevaría recolectar los datos y realizar la replicación como tal. Pese a esto, se contempla que los factores de reproducibilidad se manifiestan de forma espontánea en un experimento y, por lo tanto, pueden no presentarse en la finalización del experimento original; por el contrario, estos factores deben ser tomados en cuenta desde su concepción. Con esto en mente, es posible evaluar las características de reproducibilidad verificando el diseño de la planificación del experimento. La Figura 3.2 resume los parámetros considerados para este diseño.

3.2.2. Participantes

Medir la eficacia del modelo es una tarea difícil. Presentar una encuesta y calificar los elementos que conforman el modelo propuesto entraría en conflicto con los conocimientos y prácticas de los investigadores. Por esto se ha decidido ejecutar el experimento con la ayuda de dos tipos de participantes. Por una parte, se contará con los sujetos de experimentación que ejecutarán una planificación de un experimento y, por otro lado, se contará con expertos que se encargarán de la evaluación las planificaciones. De esta forma se puede verificar el

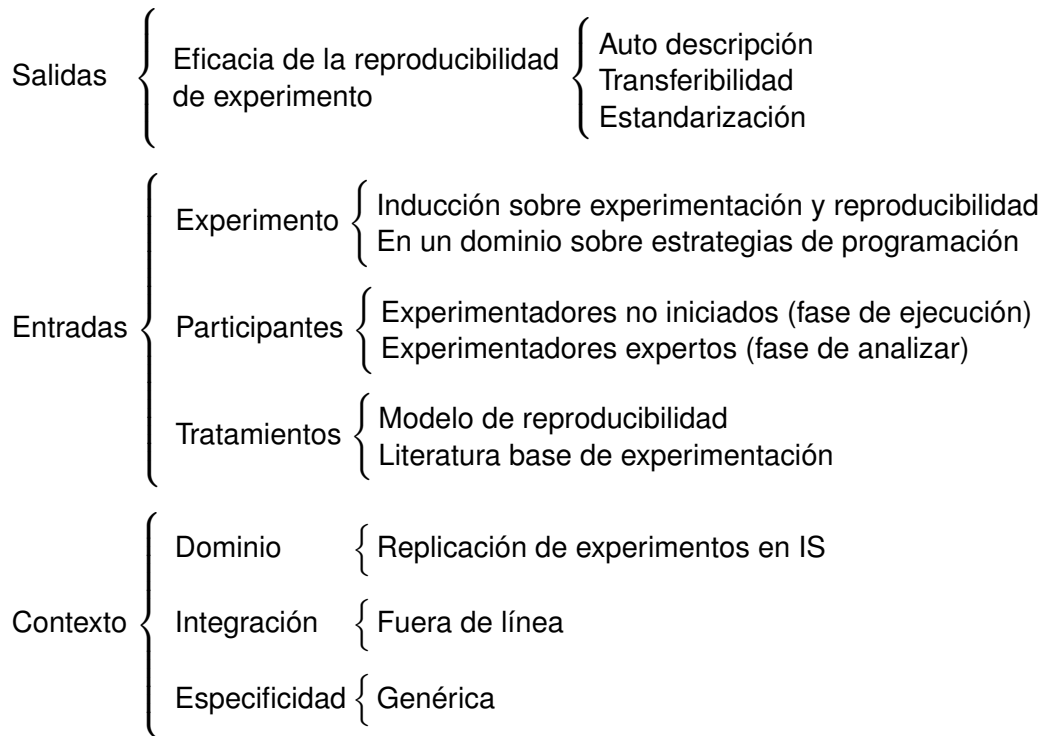


Figura 3.2: Parámetros posibles

grado de reproducibilidad alcanzado. A continuación, se detallan los atributos de los dos participantes.

Sujetos de experimentación

El experimentador no iniciado es un candidato ideal para este proceso [67]. Este tipo de participante carece de un posible sesgo procedimental tanto en la práctica de la experimentación, así como del dominio de una temática de investigación. Adicionalmente, alguien experimentado tiene un grado de conocimiento establecido a partir de su experiencia sobre los procedimientos y capacidad para determinar elementos de importancia sobre un experimento. Las características de los “experimentadores no iniciados” se detallan en Cuadro 3.3. Finalmente se consideran 15 candidatos distribuidos en dos grupos que ejecutarán el experimento utilizando distintas herramientas de soporte (ver la lista en Cuadro 3.4).

Revisores

Por otro lado, un experimentador iniciado es importante para validar la planificación de un experimento. Según Nehm [38], un experto es capaz de notar patrones significativos, organizar contenido y reflejar contextos de aplicabilidad; todo esto con un esfuerzo pequeño. Por esta razón, se podría validar la efectividad de la planificación para considerar la

presencia o ausencia de los elementos de reproducibilidad. Las características de un experimentador experto se detallan a continuación en el Cuadro 3.5. Se ha considerado una lista de candidatos quienes cuentan con una relación cercana a los que conforman esta investigación y en base a la experiencia que tienen en el ámbito de la investigación en IS. La lista de los candidatos que colaboraron denotan las características presentadas en el Cuadro 3.6.

3.2.3. Preguntas e hipótesis del experimento

Para guiar el proceso formal de experimentación se formuló la pregunta e hipótesis respectiva. La hipótesis nula y alternativa son formuladas dado que se trata de un proceso de naturaleza cuantitativa y que deberá ser evaluada a través de una prueba de hipótesis [68]. Para esta evaluación la pregunta cuestiona los factores de calidad planteados (Figura 3.3) mientras que con la hipótesis se trata de predecir la diferencia que presentarían el uso del modelo planteado en comparación con un medio de soporte tradicional, es decir el uso de literatura base (Figura 3.4).

.....
 ¿Los factores de auto descripción, transferibilidad y estandarización
 inciden en la reproducibilidad de una investigación?

Figura 3.3: Pregunta del experimento

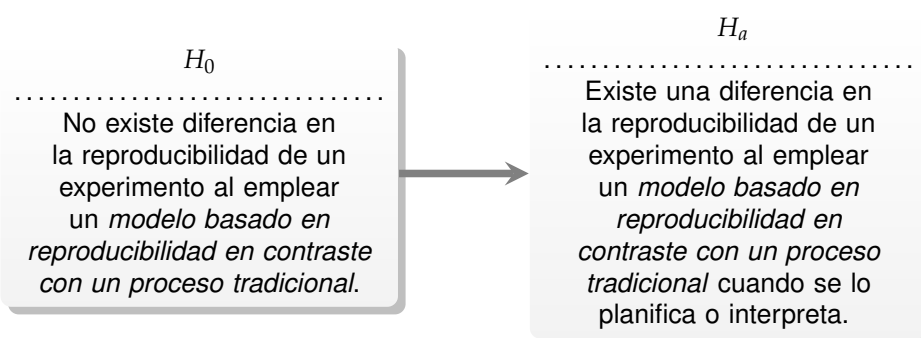


Figura 3.4: Hipótesis del experimento

3.2.4. Métricas de evaluación

Dado que se ha considerado a la reproducibilidad como un aspecto de calidad se deben plantear métricas para cada uno de los factores de calidad (auto descripción, transferibilidad y estandarización). Estas métricas permitirán evaluar las características descritas en la Sección 2.3.1. Por ejemplo, el modelo debe ser capaz de ser idóneo hacia el contexto

Cuadro 3.3: Características de sujetos

Ítem	Detalle
Población	Estudiantes de IS de la ESPE
Generalización	Global
Muestra	Estudiantes de Investigación en IS
Sujetos	Estudiantes de quinto nivel

Cuadro 3.4: Lista de sujetos

Grupo A - Modelo	Grupo B - Libro
Eduardo Antonio Mortensen Franco	Mateo Nicolas Andrade Peñafiel
Shirley Stefania Otuna Rojano	Jeremy Joel Cadena Diaz
Marco Santiago Perez Jacome	Michelle Alexandra Cantuña Chicay
Erick German Riascos Moreno	Andres Alejandro Coronel Rodrigues
Jerico Benjamin Ruiz Mafla	Juan Esteban Gallardo Alava
Maycol Estalin Tituaña Tupiza	Andres Isaias Lopez Almeida
Camila Vanessa Venegas Torres	Dennis Marcelo Malte Villarreal
Christopher Daniel Yopez Chandi	

Cuadro 3.5: Características de revisores

Ítem	Detalle
Población	Investigadores expertos
Generalización	Global
Muestra	Grupos de investigación
Sujetos	Experimentadores

de reproducibilidad de experimentos. También debe ser comprensible por los participantes (sujetos). El modelo debe ser completo a las necesidades de los investigadores. Y por último debe ser eficaz para propiciar la reproducibilidad. Por lo tanto, se han escogido las métricas descritas en la Figura 3.5, mismas que servirán para realizar el análisis cualitativo para evaluar el modelo de la sección 3.4.

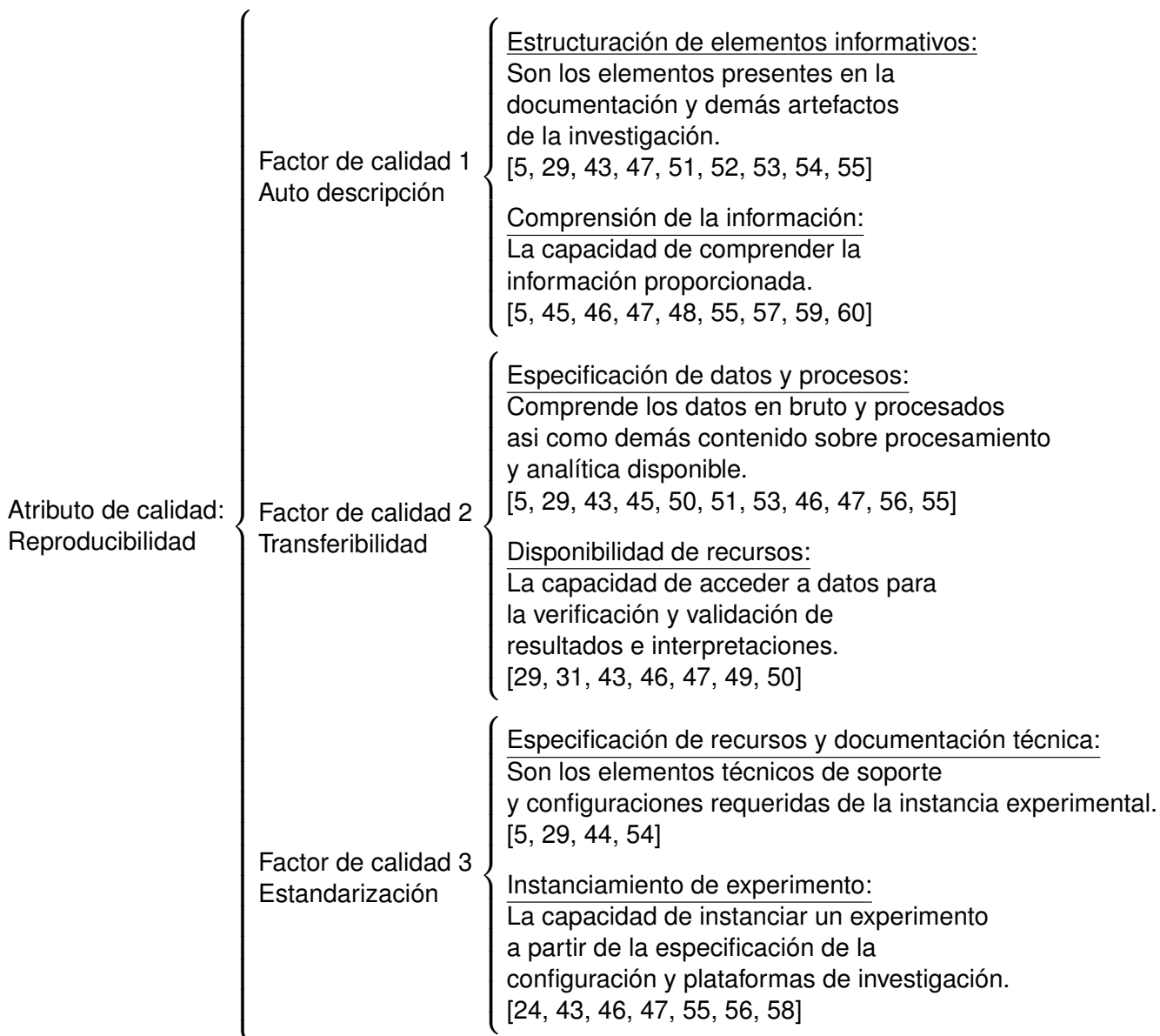


Figura 3.5: Métricas

3.2.5. Muestras

Para este experimento se contará con la participación de dos tipos de participantes: estudiantes y expertos. Como se especificó anteriormente (Sección 3.2.2), los primeros participarán en la ejecución el experimento; mientras que los segundos aportarán con

el proceso de revisión de resultados. De esta forma se espera reducir cualquier tipo de sesgo sobre los resultados. Estos participantes obedecen a las siguientes características de muestreo (Cuadro 3.7).

3.2.6. Tipo de diseño de experimento

La selección de diseño experimental incluye la selección de los grupos de sujetos participantes y su esquema de asignación. También aquí se define cuantos tratamientos se empelarán en el análisis. Los detalles del diseño se especifican en el Cuadro 3.8.

3.2.7. Recolección de datos

La recolección de datos se soportará con la herramienta para gestión de conocimiento Obsidian. Esta herramienta utiliza notas escritas con lenguaje Markdown. En cada nota se registrarán actividades y novedades, así como enlaces a los diferentes artefactos generados para el proceso experimental, por ejemplo, materiales, documentación e instrumentos de evaluación. Es necesario establecer qué protocolos se deben seguir para documentar la ejecución del experimento. A continuación, se detallan las nomenclaturas, actividades y protocolos.

Definición de nomenclatura (Actividades, materiales y anexos)

Los artefactos serán codificados de la siguiente manera:

```
<tipo><###>-<breve_descripción>[.<ext>]
```

Donde:

<tipo> corresponde a tipo de artefacto que se está registrando (Cuadro 3.9)

<###> es la enumeración del artefacto

<breve_descripción> especifica un nombre sobre el contenido del artefacto

[.<ext>] se especifica la extensión del artefacto si se trata de un archivo digital

Definición de actividades de recolección

La gestión de actividades se llevará a cabo bajo un esquema de tablero Kanban que especificará un código, la actividad, la fase de ejecución y la fecha tentativa tal como se muestra en el Cuadro 3.10. Cabe recalcar que el proceso cíclico de diseño y ejecución han permitido que las fechas y detalle de las actividades vayan ajustándose en el tiempo. En Obsidian se definió una plantilla para el detalle de actividad se ingresa el detalle de la

actividad, resultados esperados, locación donde se llevará a cabo la actividad, materiales, participantes y se listan posibles novedades (ver Código 3.1).

Protocolos para registro de novedades

Las novedades relacionadas a cualquier actividad serán registradas utilizando una plantilla en la que se detalla el tipo de actividad, el nivel de impacto estimado, y el tipo de amenaza a la validez que se considera a fin de que pueda ser registrada y considerada en el análisis e interpretación de resultados. La plantilla sigue un esquema en el que se especifican el problema, impacto y contingencias; de igual manera en la cabecera YAML se ingresa tipo de novedad, nivel de impacto, fecha y amenaza (ver Código 3.2).

Protocolos de validación de recolección

El proceso de validación de la recolección de datos para esta experimentación contempla dos etapas. En la etapa de inducción se validará a través de la asistencia de los sujetos a las sesiones de entrenamiento y se fomentará la participación en clase y fuera de ella a través de tareas grupales e individuales. En la etapa de recolección se utilizará la plataforma Moodle para asegurar la entrega oportuna de los artefactos generados. Adicionalmente, se culminó la recolección cargando, registrando y enlazando los artefactos a las distintas actividades en la plataforma Obsidian, siguiendo la nomenclatura proporcionada (ver Cuadro 3.9 y Anexos 1 y 2).

3.2.8. Instrumentos

Los instrumentos utilizados en este experimento corresponden a dos tipos: archivos generados como material para capacitación o de recopilación de información; y archivos ubicados en línea como formularios de encuestas digitales. La lista de instrumentos disponibles en la bóveda del experimento y en línea se muestran en el Cuadro 3.11.

3.3. Recolección de datos

La fase de recolección de datos inicia con la actividad ACT-001 en donde se plantean las primeras tareas para validar la recepción de lo enseñado a los sujetos participantes. En esta etapa se debe especificar: (i) si existe información privada cuyo uso deba ser autorizado, (ii) cómo asegurar la motivación de los participantes y (iii) cómo finalmente reportar las novedades encontradas durante la ejecución. A continuación, se detallan estos puntos.

Cuadro 3.6: Perfiles de revisores

Revisor	Experiencia (Años)	Campo de investigación en IS	País
1	6	Aprendizaje & Desarrollo	Ecuador
2	24	Testeo & Requerimientos	España
3	32	Testeo	México
4	17	Testeo	México
5	16	Replicación	Uruguay
6	22	Replicación	Chile
7	8	Replicación	Paraguay
8	9	Desarrollo	Ecuador
9	12	Seguridad	Ecuador
10	12	Testeo	Ecuador
11	13	Testeo	Ecuador

```
1 ---
2 tag: detalle-actividad
3 ---
4 ### Actividad
5 Descripción detallada de la actividad
6 ### Resultados esperados
7 - Resultados que se esperan alcanzar al realizar esta actividad
8 ### Locación
9 Ubicación física, geográfica o virtual
10 ### Material
11 - Listado de materiales creados para esta actividad
12 ### Participantes
13 - Listado de personas que intervienen en la actividad
14 ### Novedades
15 - Listado de novedades o incidentes
```

Código 3.1: Plantilla de detalle de actividad

Cuadro 3.7: Muestra

Participantes	Organización	Muestreo
Estudiantes	Agrupada aleatoria	Conveniente (quinto nivel)
Expertos	Estratificada	Sistemática (Por experiencia)

Cuadro 3.8: Características del diseño del experimento

Ítem	Descripción
Factor	Tipo de soporte de investigación (Modelo, Libro base)
Sujeto	Experimentadores no iniciados
Objeto	Experimento
Tratamiento	Desarrollo de la planificación de un experimento utilizando una herramienta de soporte
Variables	Auto descripción, Transferibilidad, Estandarización
Grupos	Dos: Grupo A con modelo y Grupo B con libro base
Asignación	Aleatoria

3.3.1. Autorización para recolección de datos

No se requiere solicitar autorización para publicar los resultados o datos relevantes de los participantes del proceso de experimentación.

3.3.2. Motivación de los participantes

El proceso será llevado a cabo a través de la participación de estudiantes y expertos en el tema.

Estudiantes: La motivación de los estudiantes será a través de una recompensa

Cuadro 3.9: Nomenclatura de artefactos

Tipo	Significado
MAT	Material utilizado para actividad
BRT	Artefactos en bruto recabados o generados durante una actividad
PRO	Artefactos procesados o transformados
SCR	Scripts o código utilizado para procesar artefactos
RES	Resultados extraídos
ACT	Actividad
NVD	Novedad

Cuadro 3.10: Lista de actividades de recolección y previas

textbfCódigo	Etapa	Actividad	Fecha
ACT001	Inducción	Presentación - Procesos de experimentación	2022-06-14
ACT002	Inducción	Repaso - Procesos de experimentación	2022-06-16
ACT003	Inducción	Presentación - Reproducibilidad	2022-06-17
ACT003	Inducción	Tarea - Planificación de experimento	2022-06-21
ACT004	Inducción	Repaso - Reproducibilidad	2022-06-23
ACT005	Inducción	Retroalimentación - Planificación de experimento y reproducibilidad	2022-06-23
ACT006	Inducción	Taller - Criterios de planificación y reproducibilidad	2022-06-28
ACT007	Inducción	Discusión - Criterios de planificación y reproducibilidad	2022-06-30
ACT008	Recolección	Ejecución de taller - Grupo A y B	2022-07-05
ACT009	Análisis	Selección de expertos y envío de instrumentos generados	2022-07-24
ACT010	Análisis	Recolección de instrumentos evaluados	2022-07-30
ACT011	Análisis	Evaluación de instrumentos de revisión	2022-07-25
ACT012	Análisis	Procesamiento de resultados	2022-08-08

académica. El experimento es parte de una asignatura por aprobar “Investigación en Ingeniería de Software”. Posiblemente, esto garantice un esfuerzo real sobre el éxito de las metas propuestas en los talleres.

Expertos: Por otro lado, los expertos serán invitados a colaborar con el proceso de revisión. Los expertos considerados forman parte de grupos de investigación en el área, y han participado previamente en otras investigaciones de los integrantes de este trabajo. Por lo tanto, existe un grado de compromiso que podría motivar el reconocimiento de su participación y aporte a los temas de investigación en el área.

3.3.3. Artefactos recolectados

Se presenta en el Cuadro 3.12 los artefactos recolectados por cada actividad ejecutada. En los Anexos 6, 7, 8 y 9 se destacan los artefactos a evaluación y que corresponden a las actividades 8 al 10.

Código	Nombre	Actividad	Tipo
MAT001	Presentación experimento	ACT001	En línea
MAT002	Planificacion_experimental.pdf	ACT001	En bóveda
MAT003	Presentación reproducibilidad	ACT003	En línea
MAT004	Artículo1.pdf	ACT006	En bóveda
MAT005	Artículo2.pdf	ACT006	En bóveda
MAT007	Presentación grupo B.pdf	ACT008	En bóveda
MAT008	Instrucciones grupo A.pdf	ACT008	En bóveda
MAT009	Modelo de reproducibilidad	ACT008	En línea
MAT010	Encuesta experimentadores novatos	ACT008	En línea
MAT011	Aspectos de una investigación	ACT011	En línea
MAT012	Revisión de planificación	ACT011	En línea

Cuadro 3.11: Instrumentos generados para el experimento

Cuadro 3.12: Artefactos recolectados

Actividad	Artefacto
ACT002	BRT001-Revisión_de_conceptos_sobre_experimentos.pdf BRT002-Revision_de_conceptos_sobre_experimentos.xlsx
ACT003	BRT003-TAREA8- Grupo1-Planificacion.docx BRT004-TAREA8-GRUPO2 -PLANIFICACION-EXPERIMENTO.docx BRT006-TAREA8-Planificacion -Experimento-Grupo3.docx BRT005-TAREA8-Planificacion de experimento-Grupo4.docx
ACT005	BRT007-Maycol Tituaña Planificación de experimento.docx BRT008-tarea9.docx BRT010-TAREA9_VENEGAS_CAMILA.docx BRT011-TAREA9- Correccion-Planificacion-Grupo3.docx BRT012-TAREA9- LOPEZ-CORRECCION-PLANIFICACION- EXPERIMENTO-REPRODUCIBILIDAD.docx BRT013-TAREA9- Observaciones del experimento-Grupo4-Eduardo Mortensen.docx

Continuación de la tabla 3.12

Actividad	Artefacto
	BRT014-TAREA9- Revision sobre la planificación-GallardoJuan.docx
	BRT015-TAREA9- Revisión sobre la planificación-Otuna Shirley.docx
	BRT016-TAREA9- Revisión-Planificación-Experimento-CoronelAndrés.docx
	BRT017-TAREA9- Revision-Planificacion-Experimento-Grupo3.pdf
	BRT018-Tarea9- Revisión-Planificación-Experimento-MalteMarcelo.docx
	BRT019-TAREA9- Revision-Planificación-YepezChristopher.docx
	BRT020-TAREA9- RUIZ-JERICO-REVISION-PLANIFICACION.docx
ACT006	BRT021-Tarea 10 - Evaluación de la planificación de experimentos.pdf
	BRT022-TAREA10 - GRUPO2.xlsx.pdf
	BRT023-TAREA10.pdf
	BRT024-TAREA10-Evaluación de la planificación_Grupo4.pdf
ACT007	BRT025-Evaluacion_de_articulos-1656622345531.xlsx
	BRT026-Evaluacion_de_articulos-1656622352217.pdf
ACT008	BRT027-Percepción estudiantes.csv
	BRT028-A01.pdf
	BRT029-A02.pdf
	BRT030-A03.pdf
	BRT031-A04.pdf
	BRT032-A05.pdf
	BRT033-A11.pdf
	BRT034-A12.pdf
	BRT035-A14.pdf

Continuación de la tabla 3.12

Actividad	Artefacto
	BRT036-B06.pdf
	BRT037-B07.pdf
	BRT038-B08.pdf
	BRT039-B09.pdf
	BRT040-B10.pdf
	BRT041-B13.pdf
	BRT042-B15.pdf
ACT010	BRT043-Aspectos de una investigación.csv
	BRT044-Revisión de una planificación de investigación.csv
ACT011	SRC001-percepción_estudiantes.R
	SRC002-aspectos_de_una_investigación.R
	SRC003-Revisión_de_una_planificación_de_investigación.R

3.3.4. Reporte de novedades

Finalmente, conforme se ejecutaron las actividades del experimento se registraron cinco novedades, listadas en el Cuadro 3.13. En el detalle de amenazas se especificó las estrategias de contingencia o de supervisión de los efectos causados en relación con las amenazas a la validez.

3.4. Análisis de datos

En esta etapa se consideró varios análisis. Primero, se recopiló datos sobre la percepción de los estudiantes respecto a su tarea de planificación. Segundo, se recopiló datos de los evaluadores en relación con su perspectiva de reproducibilidad. Estos dos puntos fueron evaluados mediante estadística descriptiva. Finalmente, se evaluó la efectividad de la herramienta aplicando una prueba de hipótesis coherente al tipo de datos que fueron recopilados. En cada punto se presenta los datos y los procesos de tratamiento siguiendo un protocolo de validación referencial explicado en el Cuadro 3.14.

Cuadro 3.13: Reporte de novedades

Código	Incidente	Amenaza
NVD006	No se exigió que se utilice la herramienta de soporte de forma obligatoria	Conclusión, Interna
NVD001	Estudiantes ausentes en actividades de inducción	Conclusión
NVD005	Un documento no cumple con parámetros	Interna
NVD004	Problemas para acceder al material de soporte - libro	Interna
NVD002	Esta clase se dictó de manera tele presencial a través de Zoom	Interna

Cuadro 3.14: Protocolo de validación

Etapas	Actividades
Análisis	Validar la recepción de los documentos y la participación de los expertos. Validar la consistencia de datos. Validar la exclusión de documentos que puedan causar interferencia. Validar la recolectar la revisión de expertos a través de una plataforma Google Forms. Validar la creación un script en R para automatizar la transformación de datos .

3.4.1. Sensibilidad de estudiantes al modelo

Los estudiantes, experimentadores no iniciados, al comenzar el proceso contaban con un grado básico de conocimiento sobre métodos de investigación, principalmente enfocados a la encuesta. Durante el proceso de inducción, los estudiantes fueron instruidos en los conceptos de experimentación y reproducibilidad.

Durante las sesiones de trabajo se hizo hincapié sobre lo fundamental que es considerar aspectos de calidad desde la concepción de un experimento, por lo que se planteó considerar a la reproducibilidad desde las etapas de planificación. Esto permitiría a los estudiantes planificar un proceso de experimentación sin llegar a instanciarlo. Sin embargo, se consideró que a través de la planificación es posible plasmar muchos elementos importantes que deberán seguirse durante la ejecución y que podrán influenciar varios aspectos de reproducibilidad.

Durante el aprendizaje y ejecución de experimentos, los investigadores generalmente

están expuestos a literatura que se enfoca principalmente en conceptos e indicaciones sobre la experimentación. Por este motivo se consideró interesante analizar las diferencias entre aplicar una herramienta de soporte tradicional (libro) en contraste con el empleo de nuestro artefacto (modelo propuesto).

Como resultado de ejecutar la actividad ACT008 se recopiló el archivo BRT027 que incluye los datos sobre la ejecución de la planificación solicitada desde la perspectiva de los experimentadores "no iniciados". En este archivo se destacan datos sobre:

- Su percepción sobre el uso de las distintas herramientas: Modelo (propuesta) y Libro (material tradicional).
- Su nivel de seguridad para realizar la tarea, sin el uso de herramientas de soporte proporcionada.
- Su nivel de seguridad para ejecutar la tarea, con el empleo de la herramienta de soporte proporcionada.

Los datos fueron tratados de la siguiente manera siguiendo el script SCR001-percepción_estudiantes.R en donde se realiza la carga del archivo, renombrado de columnas y se realiza una corrección sobre el tipo de herramienta que usó el estudiante (ver Código 3.3).

Con los datos recabados se computó la diferencia de seguridad que tuvo el estudiante para poder cumplir la tarea solicitada sin el uso de ninguna herramienta, así como la seguridad con el uso de la herramienta (ver Código 3.4).

Adicionalmente se recabó información sobre la utilidad percibida de la herramienta. El Código 3.5 muestra el análisis estadístico tomando en consideración la mediana de los datos, debido a que se trata de pocos datos no paramétricos.

3.4.2. Expectativa de reproducibilidad de los expertos

Los expertos, quienes revisan los documentos generados sobre planificación de experimentos, cuentan con experiencia en la ejecución de procesos de investigación experimental. Ellos son ajenos a las propuestas y demás hallazgos que se han formulado sobre el modelo de reproducibilidad propuesto. Se espera que ellos evalúen los documentos guiados por varios elementos categóricos; sin embargo, en esta evaluación ellos aplicarán sus propias valoraciones según sus experiencias y exigencias para garantizar que no exista un sesgo debido a algún proceso de inducción. Por esta razón se aspira conocer la apreciación de los revisores respecto a los factores que propician a la reproducibilidad según nuestra investigación.

Como resultado de la actividad ACT009 se recopiló el archivo BRT028 que incluye los criterios de los expertos sobre la reproducibilidad. En este archivo se destacan datos sobre:

```

1 ---
2 tipo_novedad: [incidente, interferencia, lección]
3 nivel_impacto: Alto Medio Bajo
4 fecha: YYYY-MM-DD
5 tipo_amenaza: [constructo, conclusión, interna, externa]
6 ---
7 ### Problema
8 Describa el problema
9 ### Impacto
10 - Describir posibles impactos
11 ### Contingencia
12 - Describir contingencias en caso de haberla

```

Código 3.2: Plantilla de detalle de novedad

```

1 library(tidyverse)
2 library(rstatix)
3 library(ggpubr)
4 library(dplyr)
5 library(ggplot2)
6
7 #Definir working directory
8 #setwd("../Planificación experimental//04 Análisis de datos")
9
10 #Carga de archivos
11 estudiantes <-
12   read.csv("adjuntos//BRT027-Percepción estudiantes.csv", encoding = "UTF-8")
13
14 #Renombrado de columnas
15 names(estudiantes) <- c(
16   "ts",
17   "Autor",
18   "Herramienta",
19   "Seguridad_sin",
20   "Seguridad_con",
21   "Nivel_utilidad",
22   "Aporta_ayuda",
23   "Descripción_ayuda",
24   "Aporta_dificultad",
25   "Descripción_dificultad")
26
27 #Corrección de tipo de herramienta usada
28 estudiantes[which(estudiantes$Autor == "Mateo Andrade"), 3] = "Libro"

```

Código 3.3: Tratamiento del archivo CVS sobre percepción de estudiantes

```

1 estudiantes %>%
2   group_by(Herramienta) %>%
3   summarise(
4     n = n(),
5     Utilidad = mean(Nivel_utilidad),
6     Seguridad_sin = mean(Seguridad_sin),
7     Seguridad_con = mean(Seguridad_con),
8     Diferencia_seg = Seguridad_con - Seguridad_sin
9   )
10
11 Herramienta   n  Utilidad Seguridad_ sin Seguridad_con Diferencia_seg
12 <chr>        <int>  <dbl>      <dbl>      <dbl>      <dbl>
13 1 Libro         7    4.43        3.43         4    0.571
14 2 Modelo        8    4.25        3.75         3.88    0.125

```

Código 3.4: Seguridad percibida por estudiantes con/sin herramienta

```

1 estudiantes %>%
2   group_by(Herramienta) %>%
3   get_summary_stats(Nivel_utilidad, type = "median_iqr")
4
5 Herramienta variable          n median  iqr
6 <chr>        <chr>          <dbl> <dbl> <dbl>
7 1 Libro      Nivel_utilidad    7     5     1
8 2 Modelo     Nivel_utilidad    8     4     1

```

Código 3.5: Utilidad de las herramientas percibida por estudiantes

- Su opinión sobre aspectos que propician la auto descripción de una investigación.
- Su opinión sobre aspectos que propician la transferibilidad de una investigación.
- Su opinión sobre aspectos que propician la estandarización de una investigación.
- Su opinión sobre los tres factores que propician la reproducibilidad de una investigación.

Los datos fueron tratados de la siguiendo el script SCR002-percepción_expertos.R en donde se realizan el renombrado de columnas (Código 3.6) y adicionalmente se computa nuevas variables para realizar los cálculos de las “relevancias” de los expertos (Código 3.7).

3.4.3. Pruebas de hipótesis

Ya que los expertos evaluarán las planificaciones presentadas por los estudiantes, se diseñó un formulario para recopilar las características de una investigación relacionadas a la auto descripción, transferibilidad y estandarización es un estudio. En este análisis se debe considerar que el tipo de análisis se ajuste al tipo de dato generado de la recopilación de los resultados de la revisión que los expertos realizaron a los documentos sujetos de experimentación. Para esto se han formulado tres grupos de preguntas relacionadas a los tres factores de reproducibilidad de la siguiente forma:

- Respecto a la auto descripción se considera la identificación de:

- Problemas de investigación
- Objetivos de investigación
- Contexto de investigación
- Preguntas e Hipótesis de investigación
- Parámetros de investigación
- Descripción de sujetos
- Descripción de muestra
- Diseño de experimento
- Definición de instrumentos
- Definición de meta datos de estudio

- Respecto a la transferibilidad se considera la identificación de:

- Protocolos de gestión
- Ubicación y organización de datos
- Procesamiento de incidentes
- Ubicación y gestión de incidentes
- Procesos análisis
- Procesos de empaquetado
- Descripción de repositorios
- Instrucciones de navegabilidad
- Esquemas de codificación y etiquetas
- Glosario de términos


```

1 library(ggplot2)
2 library(dplyr)
3
4 #Definir working directory
5 #setwd("../Planificación experimental//04 Análisis de datos")
6
7 # Carga de archivos
8 expertos <-
9   read.csv("adjuntos//BRT043-Aspectos investigación.csv", encoding = "UTF-8")
10
11 #Renombrado de columnas
12 names(expertos) <- c(
13   "ts", "Experto",
14   "AD_estructura",
15   "AD_navegabilidad",
16   "AD_estandarización",
17   "AD_otras",
18   "T_canales",
19   "T_repositorios",
20   "T_gestión_conocimiento",
21   "T_reportes",
22   "T_otras",
23   "E_revistas",
24   "E_plataformas",
25   "E_infraestructuras",
26   "E_arquitecturas",
27   "E_otras",
28   "Relevancia_AD",
29   "Relevancia_T",
30   "Relevancia_E",
31   "Reproducibilidad_personal",
32   "Reproducibilidad_grupo",
33   "Reproducibilidad_otros")
34
35 #Convertir clase de las columnas a numéricas
36 cols <- c(3:5,7:10,12:15,17:22)
37 expertos[,cols] <-
38   apply(expertos[,cols], 2, function(x) as.numeric(as.character(x)))

```

Código 3.6: Tratamiento del archivo CVS sobre opinión de expertos

```

1  #revisión de factores
2  cs <- data.frame(tipo = character(), ad = numeric(), t = numeric(), e = numeric())
3  cs <- rbind(cs, list(
4    tipo = "Calificación\nindirecta",
5    ad = mean(c(
6      mean(expertos$AD_estructura),
7      mean(expertos$AD_navegabilidad),
8      mean(expertos$AD_estandarización)
9    )),
10   t = mean(c(
11     mean(expertos$T_canales),
12     mean(expertos$T_repositorios),
13     mean(expertos$T_gestión_conocimiento),
14     mean(expertos$T_reportes)
15   )),
16   e = mean(c(
17     mean(expertos$E_revistas),
18     mean(expertos$E_plataformas),
19     mean(expertos$E_infraestructuras),
20     mean(expertos$E_arquitecturas)
21   ))
22   ), stringsAsFactors = FALSE)
23
24  cs <- rbind(cs, list(
25    tipo = "Calificación\ndirecta",
26    ad = mean(expertos$Relevancia_AD),
27    t = mean(expertos$Relevancia_T),
28    e = mean(expertos$Relevancia_E)
29   ), stringsAsFactors = FALSE)

```

Código 3.7: Preparación de dataframes para análisis de datos

- Datos y mecanismos de contacto
- Respecto a la estandarización se considera la identificación de:
 - Uso de recursos adicional
 - Descripción de herramientas y tecnologías
 - Descripción sobre el uso de repositorios
 - Descripción sobre el despliegue del estudio

Las preguntas antes descritas fueron formuladas con una escala Likert. Esto significa que los resultados, al tratarse de escalas, son ordinales en naturaleza y por lo tanto los datos se consideran como “No paramétricos” [69, 70]. Las escalas consideradas van desde “No identificable” a “Muy fácilmente identificable”. Adicionalmente por el tamaño de las muestras no es posible definir la distribución, aunque se puede realizar una prueba de normalidad de los datos. En base a estas características se tomó como prueba de hipótesis las pruebas de Wilcoxon.

Como resultado de la actividad ACT010 se recopiló el archivo BRT-044 que incluye la revisión de cada documento. En este archivo se cubren datos sobre las características presentadas en la sección de métricas, principalmente evaluando la presencia o consideración explícita de los siguientes elementos:

- Datos relevantes sobre la investigación
- Procedimientos relevantes sobre la investigación
- Descripción de artefactos relevantes sobre la investigación
- Descripción de repositorios de la investigación
- Descripción de tecnologías y herramientas relevantes sobre la investigación
- Uso de tecnologías y herramientas estandarizadas en la investigación

Para preparar el conjunto de datos de análisis se ejecutaron los procesamientos especificados en el Código 3.8. Sin embargo, para poder realizar el análisis estadístico se computaron columnas adicionales con promedios por factores de reproducibilidad y grupos, ejecutando el Código 3.9.

Resultado de la prueba de hipótesis

Para las pruebas de hipótesis se empleó el test de Wilcoxon (ver Código 3.10) sobre cada categoría de factor de reproducibilidad propuesta. Como resultado del análisis estadístico se determina que los valores no son menores que 0.05 lo que indica un resultado no significativo. Consecuentemente se acepta la hipótesis nula y se rechaza la hipótesis alternativa dando como resultado que **no existe diferencia en la reproducibilidad de un experimento al emplear un modelo basado en factores de reproducibilidad en contraste con un proceso tradicional.**

```

1 library(tidyverse)
2 library(rstatix)
3 library(ggpubr)
4 library(dplyr)
5 library(ggplot2)
6 library(gridExtra)
7 library(cowplot)
8
9 #Definir working directory
10 #setwd("../Planificación experimental//04 Análisis de datos")
11
12 # Carga de archivos
13 revision <-
14   read.csv("adjuntos//BTR044-Revisión planificación.csv", encoding = "UTF-8")
15
16 #Renombrado de columnas
17 names(revision) <- c(
18   "ts", "Experto", "Documento", "Acuerdo", "Problema",
19   "Objetivos_preguntas", "Contexto", "Hipotesis", "Parametros",
20   "Sujetos", "Muestras_seleccion", "Diseño", "Instrumentos",
21   "Metadatos", "Protocolos", "Ubicacion", "Procesos_incidentes",
22   "Ubicacion_incidentes", "Procesos_analisis", "Procesos_empaquetado",
23   "Repositorios", "Instrucciones", "Codificacion_etiquetas", "Glosario",
24   "Contacto", "Recursos_adicionales", "Informacion_herramientas",
25   "Uso_repositorios", "Despliegue_instancia", "Comprension",
26   "Esfuerzo", "Tecnologias", "Toma_decisiones", "Autodescrita",
27   "Transferible", "Estandarizado")
28
29 #Cambio de etiquetas de LIKERT a valores
30 revision[revision=="No identificable" | revision=="Totalmente en desacuerdo" ]=0
31 revision[revision=="Difícil" | revision=="Algo en desacuerdo"]=1
32 revision[revision=="Normal" | revision=="Indeciso"]=2
33 revision[revision=="Fácil" | revision=="Algo de acuerdo"]=3
34 revision[revision=="Muy fácil" | revision=="Totalmente de acuerdo"]=4
35
36 #Convertir clase de las columnas a numéricas
37 cols <- c(5:ncol(revision))
38 revision[,cols] <-
39   apply(revision[,cols], 2, function(x) as.numeric(as.character(x)))

```

Código 3.8: Preparación de datos para análisis

```

1 #Promedios por factor de reproducibilidad
2 revision$ad <- rowMeans(revision[,c(5:14)], na.rm=TRUE)
3 revision$t <- rowMeans(revision[,c(15:25)], na.rm=TRUE)
4 revision$e <- rowMeans(revision[,c(26:28)], na.rm=TRUE)
5
6 #Agregación de grupos de documentos
7 revision$Grupo <-
8   ifelse(
9     startsWith(revision$Documento,"A"),
10    "Modelo",
11    "Libro"
12  )

```

Código 3.9: Preparación de datos para pruebas de hipótesis

```

1 > revision %>% wilcox_test(ad ~ Grupo) %>% add_significance()
2 # A tibble: 1 x 8
3   .y. group1 group2  n1  n2 statistic  p p.signif
4 1 ad  Libro  Modelo  11  11    66.5 0.717 ns
5
6 > revision %>% wilcox_test(t ~ Grupo) %>% add_significance()
7 # A tibble: 1 x 8
8   .y. group1 group2  n1  n2 statistic  p p.signif
9 1 t   Libro  Modelo  11  11    57.5 0.869 ns
10
11 > revision %>% wilcox_test(e ~ Grupo) %>% add_significance()
12 # A tibble: 1 x 8
13   .y. group1 group2  n1  n2 statistic  p p.signif
14 1 e   Libro  Modelo  11  11    46.5 0.368 ns

```

Código 3.10: Pruebas de hipótesis con Wilcoxon-test

Carpeta Definición	{	Carpeta de adjuntos Elementos de definición Material complementario bibliográfico del marco teórico
Carpeta Planificación	{	Carpeta de adjuntos Elementos de planificación Biografía y datos de participantes del experimento
Carpeta Recolección	{	Carpeta de adjuntos Especificación de procesos de recolección y validación
Carpeta Análisis	{	Carpeta de adjuntos Especificación de procesos de análisis y validación Scripts y código de análisis automático (archivos R)
Carpeta Empaquetado	{	Carpeta de adjuntos Especificación de empaquetado Interpretaciones, amenazas e interferencias, lecciones aprendidas
Carpeta Anexos	{	Carpeta de plantillas: incidentes, participantes, actividades Carpeta de detalle de actividades Tablero Kanban para la gestión las actividades

Archivo “Léeme primero” que resume y explica la navegabilidad

Figura 3.6: Parámetros posibles

3.5. Empaquetado de resultados

El empaquetado de resultados resume los principales hallazgos de la investigación como son las interpretaciones de los resultados, la identificación de amenazas, las interferencias de los datos y la identificación de lecciones aprendidas. Estos elementos serán tratados en el capítulo siguiente. Como parte del empaquetado de resultados, se pone en explica el esquema de organización de la información y los mecanismos de acceso a la investigación.

3.5.1. Esquemas de organización

El proceso de experimentación fue organizado con la herramienta Obsidian y en lenguaje Markdown. Se organizaron carpetas para organizar las distintas “notas” con la información sobre la experimentación. Las carpetas fueron esquematizadas a partir de la taxonomía presentada en [4] de la siguiente forma (Figura 3.6).

3.5.2. Navegación

La plataforma elegida permite la navegación y exploración del contenido a través de enlaces de tipo “wikylinks” que permite la especificación de conexiones de entrada o salida entre los distintos elementos de la “bóveda” del experimento. Este mecanismo permite acceder a una vista de grafo, misma que se muestra en la Figura 3.7. En esta vista se destacan las interrelaciones entre los datos, así como varios tipos de “conocimiento” por grupo de colores, como ejemplo: artefactos generados (café), revisores (celeste), novedades de tipo amenaza (rojo), entre otras. Este esquema facilita el entendimiento y exploración del estudio.

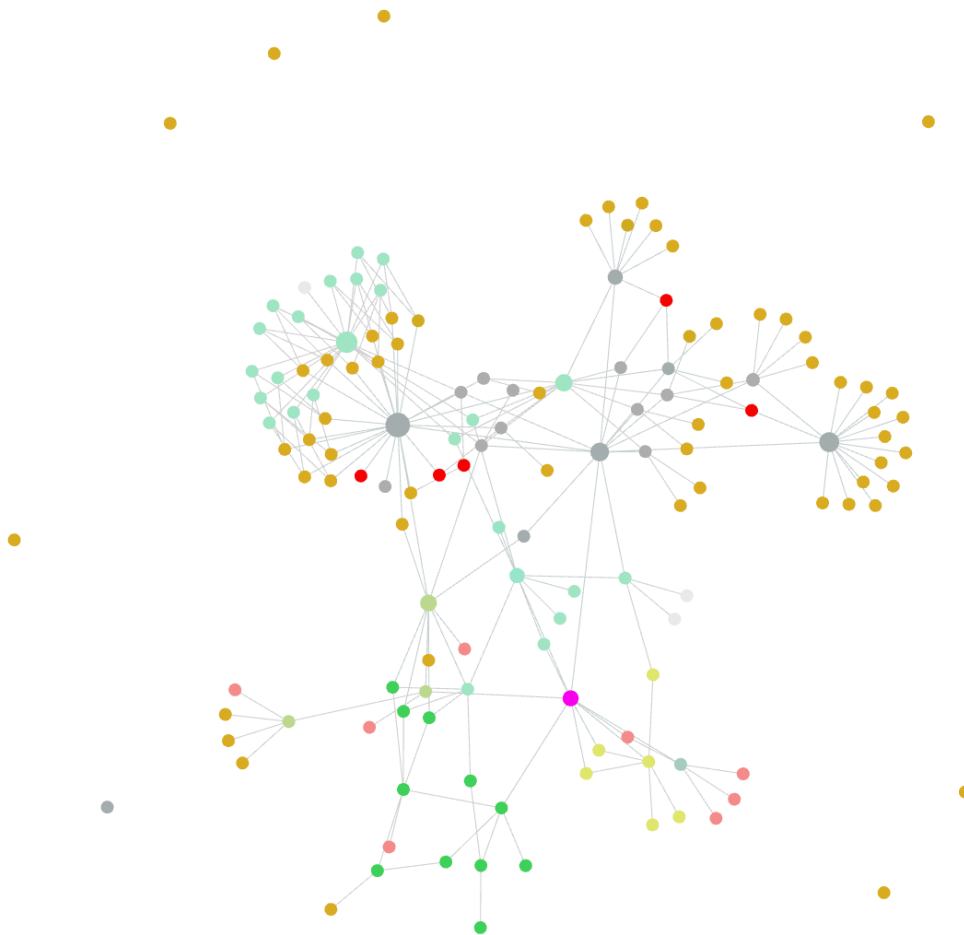


Figura 3.7: Red grafo de bóveda de investigación

3.5.3. Acceso

El acceso a la bóveda será distribuido a través de un repositorio público en GitLab. El repositorio puede ser accedido en <https://gitlab.com/educeav/modelo-de-reproducibilidad> y

al tratarse de un repositorio público puede ser accedido, clonado o bifurcarlo. Para acceder al contenido se debe abrir la “bóveda” con obsidian.md.

3.6. Interpretación de resultados

3.6.1. El impacto del modelo en los usuarios

Al realizar el análisis estadístico sobre la utilidad de la herramienta según la perspectiva de los sujetos de experimentación se pudo determinar que existió poca diferencia entre las opiniones de los usuarios usando ambas herramientas. Examinando la media, el modelo planteado fue menos valorado que el método de soporte tradicional (literatura sobre experimentación y experimentación de experimentos), posiblemente por el tiempo de familiarización que se dio sobre la herramienta, que fue de cinco minutos. Aunque se cree que la herramienta fue sencilla de interpretar, el modelo no se trataba de una herramienta convencional como un libro, sino que se trataba de un modelo de intenciones. Sin embargo, el análisis mostró que no existe una diferencia significativa sobre la valoración entre ambas herramientas de soporte (Figura 3.8).

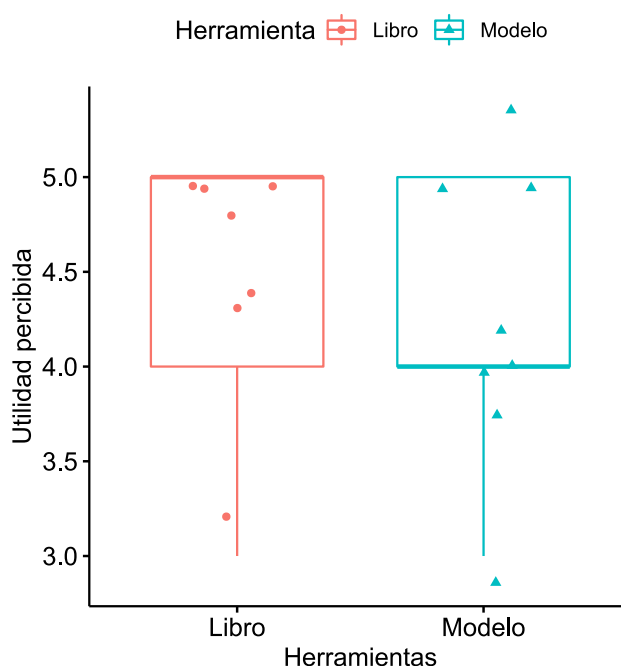


Figura 3.8: Diferencia entre utilidades percibidas

3.6.2. La perspectiva de los expertos

Aquellos que han practicado la investigación están conscientes sobre la importancia y la necesidad de la reproducibilidad. Pese a ello, aún existen muchas investigaciones que hablan sobre esta temática y su impacto en las actividades científicas. En esta investigación se aprovechó la disponibilidad de los expertos para entender como ellos visualizarían los factores de reproducibilidad que fueron introducidos (auto descripción, transferibilidad y estandarización). Esta indagación se realizó de dos formas: la primera a través de las características de cada factor, y preguntado de forma directa; dando como resultado lo observado en la Figura 3.9.

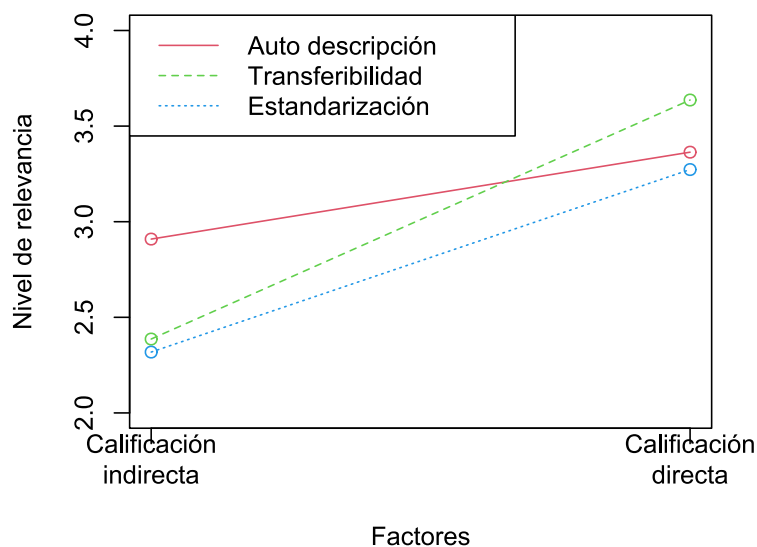


Figura 3.9: Relevancia de los factores de calidad

Los expertos fueron indagados a través de preguntas directas para valorar la importancia de los tres factores dando como resultado que la transferibilidad es más valorado, seguido de la auto descripción y muy de cerca, la estandarización de los elementos de una investigación. Sin embargo, al realizar la misma indagación de forma indirecta, todas las valoraciones decaen en términos de relevancia. Inclusive el orden de éstas cambia, siendo más importante la auto descripción seguida muy de cerca por la transferibilidad y la estandarización. Este resultado coincide mucho con las prácticas recabadas en la literatura investigada donde se menciona el énfasis que los autores de una investigación hacen a los mecanismos de reporte (artículos científicos) y descuidando otros factores que mejorarían la reproducibilidad.

Los expertos también han recomendado otros elementos adicionales sobre la reproducibilidad como, por ejemplo:

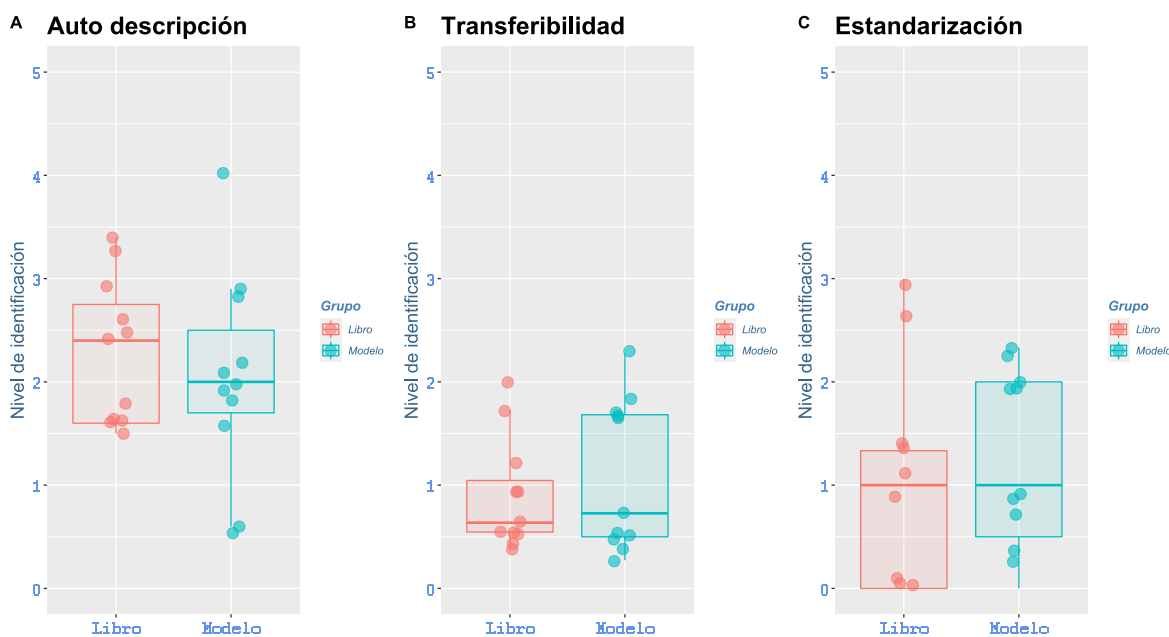


Figura 3.10: Relevancia de los factores de calidad

■ Auto descripción:

- Describir el área de investigación y el problema a resolver
- Uso de nomenclatura homogénea en las figuras y tablas.
- Trazabilidad a través de la descripción detallada de la investigación

■ Transferibilidad:

- Participación directa en la realización
- Identificar congresos, revistas que estén alineadas a la investigación

■ Estandarización:

- Ninguna

3.6.3. La efectividad del modelo

A través de la experimentación y la definición de hipótesis, el modelo fue probado a través de la prueba de Wilcoxon debido a la naturaleza de los datos y al tipo de variables recabadas. A través de la prueba de hipótesis se acepta la hipótesis nula, sin embargo, se puede analizar cómo se ha desenvuelto la aplicación del modelo. La Figura 3.10 muestra la comparativa en el uso del modelo y el libro en los tres factores de reproducibilidad. En las gráficas se puede observar que no existe un cambio significativo, reforzando lo explorado en la prueba del Wilcoxon; sin embargo, la transferibilidad y la estandarización mejoran según la perspectiva de los expertos.

Adicionalmente se realizaron las mismas pruebas por características de forma particular y se notó dos elementos que se destacan de forma significativa (ver Código 3.11 y

Figura 3.11). En los componentes de empaquetado y repositorios se nota una diferencia estadística significativa. El proceso de empaquetado incluye los aspectos necesarios para la comprensión de una investigación. Esto conlleva a especificar adecuadamente tanto los hallazgos finales del estudio, así como presentar la estructura de navegación para la trazabilidad de ellos. Por otro lado, la especificación de los repositorios, también incluida en el empaquetado, debe ser analizada para establecer los niveles de acceso y privacidad, en caso de que el estudio incluya información crítica o confidencial. Finalmente, las ventajas de tener repositorios avanzados, por ejemplo, los basados en git, es la capacidad no solo del despliegue, sino también de su capacidad de controlar los cambios. Esto cambios son mejores aprovechados si los formatos de archivos son de tipo “abierto”. Esta característica permite automatizar el proceso de comparaciones entre distintas ramas (familias) de una investigación.

```

1 > revision %>% wilcox_test(Procesos_empaquetado ~ Grupo) %>% add_significance()
2 # A tibble: 1 x 8
3   .y.      group1 group2    n1    n2 statistic      p p.signif
4   <chr>    <chr> <chr> <int> <int>    <dbl> <dbl> <chr>
5 1 Procesos_empaquetado Libro Modelo    11    11     29.5 0.0374 *
6
7 > revision %>% wilcox_test(Uso_repositorios ~ Grupo) %>% add_significance()
8 # A tibble: 1 x 8
9   .y.      group1 group2    n1    n2 statistic      p p.signif
10  <chr>    <chr> <chr> <int> <int>    <dbl> <dbl> <chr>
11 1 Uso_repositorios Libro Modelo    11    11     31 0.0456 *

```

Código 3.11: Pruebas adicionales de características con diferencia significativa

3.7. Amenazas a la validez

A continuación, se presentan las distintas amenazas que se han identificado durante la concepción y ejecución de esta investigación. Primero se consideraron las posibles amenazas de constructo, que hacen referencia a los factores y fundamentos que considerados para la formulación del modelo. Seguidamente se plantearon las amenazas internas, afines a las relaciones entre variables dependientes e independientes. Se continuó con las amenazas a la conclusión estadística, y se finalizó con las amenazas externas sobre la generalización de los efectos observados.

La reproducibilidad es un problema muy difícil de especificar ya que depende de las capacidades de todos los componentes de los elementos de comunicación (el mensaje, el medio, el transmisor y el receptor). Las investigaciones, en este contexto, deben ser capaces de actuar en todos estos componentes a la vez o ser capaces de aislar y enfocarse en

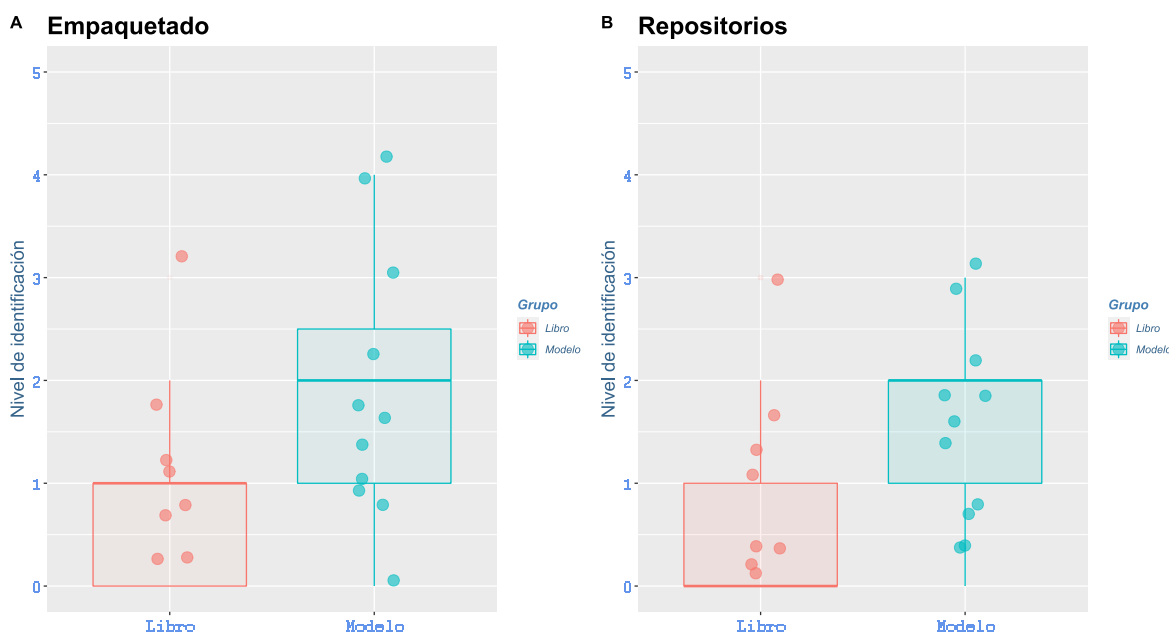


Figura 3.11: Relevancia de dos características de calidad

cada uno de ellos. Por un lado, si se tratan todos los componentes a la vez, el proceso de investigación se torna muy costoso en recursos y en tiempo. Por otro lado, una evaluación independiente de uno de los componentes podría dejar de lado variables de comunicación que impactarían los resultados.

Para minimizar las amenazas de constructo, se optó por especificar un modelo de madurez en donde se identificó elementos del proceso que estaban ausentes o pocos especificados en el proceso de la reproducibilidad. Seguidamente se definió posibles factores de calidad para entender de mejor manera la reproducibilidad. El primer riesgo puede aparecer en la formulación de estos factores; sin embargo, el foco de la investigación es respecto a características o elementos habilitantes, dando como resultado que los factores son una agrupación que permite tratar de forma abstracta un grupo de habilitantes. Otro elemento que ayudó a seleccionar las variables del modelo y proceso es fue por la aplicación de la metodología DS.

Por el lado de las amenazas internas, el riesgo más grande fue escoger adecuadamente un grupo de experimentación. Por un lado, se tienen a expertos que podrían presentar un sesgo importante al “dejarse” guiar por un esquema nuevo. Pero, tener personas con nociones nulas de investigación podrían perjudicar el desarrollo correcto del experimento dejando de lado los aspectos que se desean evaluar. En este proceso se contó con la participación de un grupo de estudiantes que se encontraban en un rango medio de conocimientos, que permitió aplicar la experimentación reproducible. Sin embargo, hay que considerar que sus motivaciones pudieron no ser tan altas para comprender realmente la naturaleza de la propuesta. De este grupo, no se presentó un estudiante, sin embargo, esto

no fue un factor de amenaza por tratarse de un grupo grande de estudiantes. Finalmente, otro factor de amenaza fue la creación y diseño de los instrumentos (materiales de inducción, pruebas y formularios). Para disminuir esta amenaza, los instrumentos fueron revisados y discutidos tanto en un ámbito conceptual como estructural. Durante la ejecución de las actividades se destacaron las siguientes novedades:

- **Estudiantes ausentes en sesiones (NVD001).** Aunque existieron ausencias, existieron varios mecanismos para reforzar vacíos de conocimiento como talleres grupales e individuales durante varias sesiones.
- **Cambio de modalidad de inducción (NVD002).** Las sesiones eran en modalidad presencial, pero por circunstancias ajenas, se cambiaron a tele presencial. Como respuesta se utilizó una plataforma de participación para mantener la retroalimentación activa de las sesiones de trabajo.
- **Problemas para acceder a materiales en taller (NVD004).** Uno de los grupos tuvo problemas para acceder a los materiales por un periodo menor de cuatro minutos. Esto restaría tiempo a su proceso de desarrollo; sin embargo, los estudiantes fueron capaces de terminar antes de tiempo su tarea.

En relación con las amenazas de conclusión estadística se tomó en cuenta el tipo de distribución de los datos, el tipo de datos y la cantidad de la muestra. Wilcoxon es el método de pruebas más apropiado para este tipo de características. El nivel de confianza que consideramos es del 95%. A más del proceso de análisis estadístico existe, novedades respecto al análisis del resultado, entre los que destacan los siguientes:

- **Documento descartado (NVD005).** Se revisó un documento antes de ser enviado a los revisores. Este documento no cumplía con los parámetros de revisión al no seguir los pasos especificados en la tarea. Pese a esta novedad, este documento no influye en los datos analizados.
- **Obligatoriedad de la herramienta (NVD005).** En las sesiones de taller para la elaboración de la planificación se mencionó de forma implícita el uso de las herramientas de soporte. De esta manera pudo existir la posibilidad de que los estudiantes hubieran hecho caso omiso al uso de una herramienta adicional por las restricciones de tiempo y alcance. Se mencionó que la encuesta final se trataba sobre el uso de la herramienta. Los datos encuestados parecen demostrar que si se usó tanto el libro como el modelo.

En cuanto a las amenazas externas, los datos pueden ser generalizados ya que la evaluación de los instrumentos no dependió de la revisión de los mismos creadores de la propuesta sino de la participación de terceros, expertos ajenos al grupo de esta investigación, así como de sus hallazgos y objetivos. Adicionalmente, la revisión se realizó aplicando un mecanismo ciego o anonimización.

3.8. Discusión

3.8.1. El énfasis a la reproducibilidad de experimentos

La ISE es un componente de la disciplina de la IS que ha tomado más presencia con relación al impacto para generar prácticas de ingeniería basada en evidencia científica. Consecuentemente las prácticas y problemáticas que conlleva la generación de conocimiento científico (es decir, conocimiento validado y generalizable) también se hacen evidentes. Aunque, la reproducibilidad es un atributo que se considera arraigado a cualquier investigación, lo cierto es que medir y especificar los habilitantes dista mucho de ser una práctica común.

Para lograr un verdadero entendimiento de la reproducibilidad, como atributo de calidad de investigaciones, es necesario empezar a evaluar este atributo como se lo hace en otros productos cotidianos. También es importante valorar el impacto de una investigación replicada o reproducida sobre la formación del conocimiento científico y no confundir el alcance y la filosofía del término “investigación original”. Quizá la reproducción de estudios debería ser tomada y propiciada como método de entrada al mundo científico en los programas de estudio de investigación y así poder generar artículos de impacto en revistas “locales”. Este proceso no solo haría énfasis en la necesidad de un estudio reproducible sino en la comprensión sobre lo que significa un estudio de calidad.

3.8.2. El destino de las propuestas

Continuando con la mejora de los “procesos de reproducibilidad”, es posible encontrar una vasta cantidad de propuestas en una exploración de literatura. A la par de las propuestas, también existen artículos que evalúan y recomiendan mejoras a estas propuestas. Se cree que estas evaluaciones dan a entender que las propuestas son “cerradas” y por ende no son flexibles a aplicaciones distintas, o se está confundiendo un modelo de proceso con la instancia de aplicación. Posiblemente el resultado del diseño o mala interpretación actúa como barreras al tratar de acoger nuevas propuestas, componente social que podría ser analizado.

Muchas de estas propuestas no son acogidas más allá de su aplicación en los grupos de investigación donde se originan. De hecho, existen múltiples plataformas web cuyos sitios están dados de baja. La colaboración entre grupos de investigación parece no ser lo suficientemente comprometida para mejorar los factores de reproducibilidad que prometen las herramientas. El foco se sigue manteniendo en el reporte, como producto tangible y necesario para competir en el ámbito académico-investigativo.

3.8.3. La crisis de la reproducibilidad

El proceso metodológico para la creación del conocimiento científico se apoya en la reproducibilidad. Se ha explorado el tema de “crisis de reproducibilidad” a la imposibilidad o complicaciones al tratar de acceder a datos, procesos y decisiones contempladas o generadas en un estudio, comprometiendo la verificación y validación de los datos. Existen varios problemas arraigados a este fenómeno, como por ejemplo el fraude, sesgo y la falsificación de los hallazgos. Existen autores que le refieren a esta crisis como la “narrativa de la crisis” [18]. La narrativa es entonces el foco al creciente problema de las prácticas de reproducibilidad y sus efectos inherentes.

Es cierto que los efectos adversos de la falta de reproducibilidad están presentes, pero representan un porcentaje insignificante sobre la totalidad de la producción científica. Incluso con el problema aún en boga, existe mucha investigación de calidad. El punto es entender a la reproducibilidad como una cuestión de calidad y no solamente como un proceso de investigación. Cuando un estudio es de calidad, consecuentemente tendrá características que le doten de propiedades únicas como la capacidad de verificar o validar sus hallazgos. Estas cualidades deberán ser consideradas desde la concepción del estudio, y por lo tanto requerirán un costo o esfuerzo adicional. Podría decirse que la comunidad de investigación entonces se enfrenta a una “crisis de calidad” ante la abrumadora generación de artículos.

Capítulo 4

Conclusiones y trabajos futuros

En esta sección se presenta lecciones aprendidas sobre la ejecución de la presente investigación y la problemática de la reproducibilidad; de igual manera se presentan las conclusiones sobre el diseño de la propuesta de solución, la utilidad del modelo y sobre la evaluación de este. Finalmente se resume las principales aportaciones del trabajo y se presentan posibles líneas de investigación futura que surgen a partir de los hallazgos encontrados en relación con la madurez de las prácticas de reproducibilidad de experimentos en ingeniería de software.

4.1. Lecciones aprendidas

La reproducibilidad es un tópico de gran interés en las comunidades científicas, como la de ISE. En este aspecto, los investigadores se preocupan de la agregación de mecanismos que incrementen la reproducibilidad. Adicionalmente, las revistas científicas abordan estas temáticas año tras año. Sin embargo, la reproducibilidad sigue siendo poco valorada desde el punto de vista operativo. La creencia de que la reproducibilidad es inherente a todo estudio hace que la problemática (baja reproducción de estudios) siga presente en la actualidad.

Las propuestas de reproducibilidad siguen creciendo en el tiempo. La tarea de encontrar un proceso ideal para mejorar la reproducibilidad causa el efecto contrario; es decir, incrementa la cantidad de alternativas que son difíciles de abarcar por los investigadores o grupos de investigación. De igual forma, el enfoque a procesos holísticos crea una barrera de usabilidad y aplicabilidad a las propuestas. El objetivo entonces es enfocarse en propuestas modulares que se orienten a mecánicas operativas específicas del proceso de investigación reproducible.

Al analizar el proceso de reproducibilidad con un modelo de madurez apunta a que el proceso como tal, carece de aspectos que permitan mapear las distintas necesidades

de investigación con herramientas especializadas, no al dominio de investigación, pero sí al contexto del problema. Como resultado de la investigación, se cree que avanzar en la obtención de más alternativas no aporta a la mejora; por el contrario, las directrices de experimentación deberían ser maduras como para llegar a tener esquemas o marcos de trabajo absolutamente definidos como en otras disciplinas científicas.

Los investigadores tienen patrones de trabajo arraigados desde su aprendizaje y que han sido moldeados tanto por el dominio de sus investigaciones como por sus equipos de trabajo. Esto provoca sesgos metodológicos y tecnológicos sobre la forma de trabajar y de gestionar la información. En entornos donde se prioriza la internalización de trabajo, no llega a ser un problema, pero sí cuando se trata de exteriorizar una investigación y sus hallazgos. En contraste con investigadores que se están iniciando en procesos de experimentación, podrán adoptar nuevas estrategias sin resistencia al cambio.

La reproducibilidad es un atributo de calidad de una investigación. Por lo tanto, debe contar con factores y habilitantes que puedan ser medidas. Es importante seguir indagando en estos componentes. Se ha propuesto que la reproducibilidad contempla el grado en que una investigación se describe a través de los distintos medios de distribución; también que se debe tomar en cuenta cómo se transfiere su información y se brinda acceso a los distintos elementos que conforman una investigación; y finalmente, la especificación y empleo de mecanismos estandarizados reducirán el esfuerzo al explorar los estudios. Los factores de reproducibilidad, sea cuales fueren, deben considerarse, planificarse e implementarse, desde la concepción de un estudio que se espera que sea lo más reproducible posible.

Los expertos parecen estar conscientes de la importancia de la reproducibilidad en los estudios; sin embargo, aún existen problemas al tratar de comprender estudios planteados por otros investigadores. Es posible que exista una marcada diferencia entre la comprensión de los habilitantes de la reproducibilidad y la aplicación de estrategias operativas y procedimentales para incluirlas dentro del desarrollo de una investigación. Esto genera una brecha entre la teoría y la práctica, dado que integrar elementos para incrementar las oportunidades de reproducibilidad, aumenta también el esfuerzo y complejidad de conducir una investigación.

4.2. Conclusiones

Este proyecto de investigación busca mejorar la calidad de la experimentación en IS mediante el uso de una herramienta enfocada a la selección de estrategias que satisfagan las necesidades de los investigadores. El propósito de perseguir esta mejora es la de facilitar los procesos de replicación interna y externa. Para alcanzar este objetivo, el proceso de investigación se orientó con el proceso Design Science (Sección 1.4). Este proceso permite la formulación de un artefacto basado en procesos de investigación, tanto para las partes

iniciales de la construcción del artefacto hasta su evaluación.

Para abordar este proceso fue necesario indagar en la problemática de la reproducibilidad. Se aplicaron distintos métodos para aprender sobre la temática, como lo es las revisiones de literatura, básica y sistemática. Con estos métodos se encontró que este tema se ha tratado desde los años 2000s a partir de la llamada “Crisis de la reproducibilidad”. Se recopiló datos que muestran las innumerables propuestas que pretenden mejorar la reproducibilidad que van desde procesos completos, plataformas tecnológicas, marcos de trabajo, recomendaciones y guías, y esquemas de trabajo complejos (Figura 1.10). Se complementó la indagación a través de las entrevistas a diversos experimentadores que denotaron la tendencia al uso de herramientas y mecanismos informales de gestión y comunicación (Sección 1.2).

El proceso de creación del artefacto partió del análisis de un modelo de madurez. Se escogió el PSM para analizar de mejor manera los componentes que forman parte de la solución. PSM contempla el análisis de necesidades y de problemas presentes en un contexto de experimentación (Secciones 2.2.1 y 2.2.2). También se determinó que no existe una a base de conocimiento sobre soluciones, y por ende, mecanismo de selección de herramientas. Esto significa que el proceso de experimentación no es lo suficientemente maduro (Figura 1.3). Este hallazgo orientó el análisis y desarrollo de la solución hacia estos elementos ausentes, generando un modelo que incorpore las herramientas o estrategias con características que puedan brindar un factor de evaluación.

El diseño del modelo de reproducibilidad

Como parte de la propuesta se determinó que el modelo debe incluir un mecanismo de análisis de herramientas basados en factores de calidad. Los factores de calidad incorporados fueron tres: auto descripción, transferibilidad y estandarización (Figura 2.4). Dado que los factores propuestos contemplan las necesidades y problemáticas del contexto de la reproducibilidad es posible asociar éstas a una serie de catálogo de herramientas y así permitir la adopción de aquellas que más se ajusten a los requerimientos de un contexto experimental.

Para esquematizar este modelo se utilizó un meta modelo de procesos basado en estrategias e intenciones. Este esquema de organización permite trabajar en múltiples niveles de interés. Se formuló tres niveles extensibles. El primer nivel estratégico se enfoca en los factores de reproducibilidad. El segundo nivel estratégico se enfoca en los componentes de utilidad. Un tercer nivel abarca herramientas. Los niveles estratégicos subsecuentes pueden denotar particularidades de las herramientas o de los dominios de aplicación. De esta manera se brinda la característica de expresividad al modelo a poder ajustarse a diversos dominios de aplicación. La naturaleza del modelo le otorga

comprensibilidad y coherencia al poder brindar al usuario la capacidad de elegir las estrategias que más le convengan, a diferencia de otras propuestas donde se debe aplicar un proceso muy riguroso.

Utilidad del modelo para la reproducibilidad

Para consolidar la completitud del proceso de investigación experimental, el modelo consideró diversos elementos característicos y genéricos. Para la evaluación de éste, se empleó el método de experimentación que fue guiado por los principios de la propuesta planteada, enfatizando las diferentes actividades que permitieron gestionar el conocimiento y transferibilidad de manera correcta

En la gestión de conocimiento se consideró el uso de una plataforma libre que permite el enlace de distintos elementos de conocimiento en forma de grafo, habilitando la trazabilidad como necesidad del contexto de reproducibilidad. Adicionalmente, la herramienta se soporta en el uso de archivos de texto plano que es ideal para realizar comparaciones diferenciales de su contenido. Otra ventaja obtenida es la capacidad de realizar operaciones sobre estos archivos, de tal manera que la información pueda ser procesable y computable, si se tratasen de familias de experimentos.

Por otro lado, la transferibilidad fue lograda a través de la incorporación, no solo de la documentación sino también por la especificación de notas que incorporaron lenguaje YAML. Esto sumado a la navegabilidad de los enlaces (de entrada y de salida) y la capacidad de cómputo de las notas de conocimiento, brindan a un usuario la posibilidad de encontrar información relevante a sus necesidades de investigación (Figura 3.7).

El modelo de estrategias permitió considerar varios aspectos que de otra manera hubiesen pasado desapercibidos. Aunque no se habían explorado todas las herramientas existentes de las prácticas comunes de investigación experimental, brindó pautas para poder explorar y caracterizar herramientas que pueden ser utilizadas para mejorar la utilización de repositorios y actividades de empaquetado de experimentos.

Evaluación del modelo de reproducibilidad

Para la evaluación del modelo se realizó una planificación de un experimento, tomando en cuenta que los factores de reproducibilidad deben ser contemplados en las etapas tempranas de la concepción de un estudio. El experimento contó con la participación de estudiantes de la asignatura de “Investigación en ingeniería de software” impartida en la Universidad de las Fuerzas Armadas. Estos estudiantes fueron seleccionados por contar con los conocimientos básicos sobre investigación a los cuales se introdujo los aspectos (conceptuales y logísticos) básicos sobre la experimentación y reproducibilidad. Al finalizar

el proceso de inducción, realizaron una planificación de experimento que debía considerar aspectos de reproducibilidad. Estos documentos fueron evaluados por un grupo de expertos con experiencia en temas de reproducibilidad y ajenos a especificación de la propuesta.

En la evaluación se determinó que no existe una mejora significativa cuando se aplica el modelo, en contraste con el uso de una herramienta de ayuda convencional, como un libro demostrando el cumplimiento de la hipótesis nula descrita en la Figura 3.4. Sin embargo, existe una mejora en los aspectos de transferibilidad y estandarización que merecen ser analizados con más profundidad (Figura 3.9): el empaquetado y el uso de repositorios. Dado que los datos recopilados corresponden a una muestra pequeña, es factible extender esta investigación a más escenarios para aplicar métodos de comprobación de hipótesis más robustos como T-test para análisis de muestras mayores.

4.3. Aportaciones

Durante la investigación de este estudio se han generado conocimientos y publicaciones que aportan al área de investigación en ingeniería de software empírica, mismos que se listan a continuación:

Conocimiento científico generado

1. Sistematización del problema de reproducibilidad con enfoque a la gestión de conocimiento, motivaciones y comunicación
2. Categorización de herramientas por formalidad y especialización
3. Identificación de nivel de madurez de las actividades de experimentación y elementos presentes
4. Formulación del marco conceptual de factores de reproducibilidad
5. Elaboración de modelo estrategias e intenciones de reproducibilidad

Publicación de artículos

6. The replication of experiments in software engineering, a dilemma associated with knowledge generation en CIbSE 2017 - XX
7. Ibero-American Conference on Software Engineering el 2017)
8. Resources for reproducibility of experiments in empirical software engineering: Topics derived from a secondary study en IEEE Accessthis link is disabled el 2020
9. The paradox of reproducibility improvement in empirical research in software engineering: Thoughts derived from a case study en Revista Iberica de Sistemas e Tecnologias de Informacaotthis el 2020

4.4. Líneas de investigación futura

La investigación sobre reproducibilidad aún está en boga por las implicaciones sobre el conocimiento científico y como práctica de investigación. Los resultados de esta investigación desvelan que existe una falta de madurez al momento de formar una base de conocimiento sobre un problema-solución. El proceso de análisis de alternativas no es formal y está ligado a juicios de valor o la costumbre. Es necesario profundizar en los elementos de madurez del contexto de la reproducibilidad en general.

Adicionalmente, los resultados estadísticos muestran una variación en el efecto del modelo propuesto sobre la efectividad de mejora de la reproducibilidad. Sería de gran valor mejorar la interpretación analizando más experimentos y consolidar el proceso y tomando en cuenta las lecciones aprendidas sobre proceso. Pueden ser igualmente interesante validar el efecto sobre sujetos con diferentes niveles de experiencia.

Pese a la conclusión, se cree que el modelo si tiene una aplicación válida en el proceso de mejora de la reproducibilidad. Posiblemente se pueda complementar los demás niveles estratégicos y realizar la exploración a un dominio específico de investigación, ya sea para formalizar su proceso o para recabar información sobre las alternativas problema-solución.

En los resultados se notó un cambio significativo sobre dos aspectos: el empaquetado y los repositorios. Aunque estos son elementos que comúnmente son considerados en un estudio, puede decirse que no es considerado como parte de una arquitectura de investigación cuando se plantea un estudio. Estudiar el efecto que implica considerar y estructurar estos dos componentes arquitectónicos resulta de gran utilidad para mejorar la madurez del proceso experimental reproducible.

Bibliografía

- [1] Grigorov, I., Elbæk, M., Rettberg, N., y Davidson, J. Winning horizon2020 with open science? Technical report, FP7 FOSTER & FP7 OpenAIRE+ 2015, 01 2015.
- [2] Tekinerdogan, B., Mehmet, A., Hikmet, A., y Bicer, V. *A Comparative Analysis of Software Engineering with Mature Engineering Disciplines Using a Problem-Solving Perspective*. Information Science Reference, 2011.
- [3] Wiernga, R. *Design Science Methodology for Information Systems and Software Engineering*. Springer, 1 edición, 2014.
- [4] Goulão, M. y Brito e Abreu, F. Modeling the experimental software engineering process. *QUATIC 2007 - 6th International Conference on the Quality of Information and Communications Technology*, págs. 77–90, 10 2007.
- [5] Becker, J., Beverungen, D., Breuker, D., Dietrich, H., y Rauer, H. Guidelines for reporting productivity studies—a review of the reproducibility of data envelopment analysis in the service sector. *Int. J. Services and Operations Management*, 16(3):407–425, 2013.
- [6] Peng, R. Reproducible Research in Computational Science. *Science*, 334(6060):1226–1227, 2011.
- [7] Rodríguez, F., Pérez, A., Batomeus, I., y Varela, S. Ciencia reproducible: qué, por qué, cómo. *Ecosistemas*, 25(2):83–92, 2016.
- [8] Wohlin, C., Runeson, P., Höst, M., Ohlsson, M., Regnell, B., y Wesslén, A. *Experimentation in Software Engineering*. Springer, 2012.
- [9] Mikkonen, T., Lassenius, C., Männistö, T., Oivo, M., y Järvinen, J. Continuous and collaborative technology transfer: Software engineering research with real-time industry impact. *Information and Software Technology*, 95:34–45, 2018.
- [10] Zhang, L., Tian, J., Jiang, J., Liu, Y., Py, M., y Yue, T. Empirical research in software engineering — A literature survey. *JOURNAL OF COMPUTER SCIENCE AND TECHNOLOGY*, 35:876–899, 2018.

- [11] Juristo, N. y Gómez, O. S. *Replication of Software Engineering Experiments*, págs. 60–88. Springer-Verlag, 01 2011.
- [12] da Silva, F., Suassuna, M., França, A., Grubb, A., Gouveia, T., Monteiro, C., y dos Santos. I. Replication of empirical studies in software engineering research: a systematic mapping study. *Empir Software Eng*, 19:501–557, 2014.
- [13] Fonseca Carrera, E. R. *Conceptualización e Infraestructura para la Investigación Experimental en Ingeniería del Software*. PhD thesis, Universidad Politécnica de Madrid, 2014.
- [14] Gundersen, O. The fundamental principles of reproducibility. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 379(2197), Mar 2021.
- [15] García-Bullé, S. ¿qué es la replicabilidad y por qué está en crisis? - observatorio: Instituto para el futuro de la educación, Abr 2021.
- [16] The National Academies of Sciences, E. y Medicine. *Reproducibility and Replicability in Science*. The National Academies Press, Washington, DC, 2019.
- [17] Humphreys, L., Lewis, Neil A, J., Sender, K., y Won, A. S. Integrating qualitative methods and open science: Five principles for more trustworthy research. *Journal of Communication*, 71(5):855–874, 08 2021.
- [18] Fanelli, D. Is science really facing a reproducibility crisis, and do we need it to? *Proceedings of the National Academy of Sciences*, 115:2628–2631, 2018.
- [19] Rodríguez-Pérez, G., Robles, G., y González-Barahona, J. M. Reproducibility and credibility in empirical software engineering: A case study based on a systematic literature review of the use of the szz algorithm. *Information and Software Technology*, 99:164–176, 2018.
- [20] Anchundia, C., Sanchez-Gordon, S., Cuadros, A., y Fonseca, C. La paradoja de la mejora de reproducibilidad en la investigación empírica en Ingeniería de Software: Reflexiones derivadas de un estudio de caso. *Iberian Journal of Information Systems and Technologies*, págs. 545–557, 2020.
- [21] Bausell, B. *The Problem with Science: The Reproducibility Crisis and What to do About It*. Oxford University Press, 1 edición, 2021.
- [22] Anchundia, C. La replicación de experimentos en ingeniería de software, un dilema asociado con la generación de conocimiento. *CibSE 2017 - XX Ibero-American Conference on Software Engineering*, págs. 34–41, 2017.

- [23] Crick, T., Hall, B., e Ishtiaq, S. "Can i implement your algorithm?": A model for reproducible research software. *ArXiv*, 1(1407.5981), 07 2014.
- [24] Dittrich, Y. What does it mean to use a method? towards a practice theory for software engineering. *Information and Software Technology*, 70:220–231, 02 2016.
- [25] Wilson, B. Implementing reproducible research. *Journal of Statistical Software*, 61, 10 2014.
- [26] Garousi, V., Petersen, K., y Özkan, B. Challenges and best practices in industry-academia collaborations in software engineering: A systematic literature review:. *Information and Software Technology*, 79, 07 2016.
- [27] Tekinerdogan, B. y Mehmet, A. Integrating the Concepts of Synthesis in the Software Architecture Design Process. *Society for Design and Process Science*, 7(4):1–11, 2006.
- [28] Perry, D., Porter, A., y Votta, L. Empirical Studies of Software Engineering: A Roadmap. *ICSE '00: Proceedings of the Conference on The Future of Software Engineering*, pág. 345–355, 2000.
- [29] Rozier, K. y Rozier, E. Reproducibility, Correctness, and Buildability: the Three Principles for Ethical Public Dissemination of Computer Science and Engineering Research. *Proc. IEEE Int. Symp. Ethics Sci. Technol. Eng.*, pág. 1–13, 2014.
- [30] Anchundia, C. y Fonseca, C. Resources for reproducibility of experiments in empirical software engineering: Topics derived from a secondary study. *IEEE Accessthis link is disabled*, 8:8992–9004, 2020.
- [31] Kitchenham, B., Al-Khilidar, H., Ali, M., Berry, M., Cox, K., Keung, J., Kurniawati, F., Staples, M., Zhang, H., y Zhu, L. Evaluating guidelines for reporting empirical software engineering studies. *Empir Software Eng*, 13:97–121, 2008.
- [32] Méndez-Romero, R. Qualitative data analysis with atlas.ti, por susanne friese. *Qualitative Research in Education*, 5:226, 06 2016.
- [33] Humble, A. y Radina, M. *How Qualitative Data Analysis Happens: Moving Beyond "Themes Emerged"*. Routledge, 01 2019.
- [34] Mendes, E., Wohlin, C., Felizardo, K., y Kalinowski, M. When to update systematic literature reviews in software engineering. *Journal of Systems and Software*, 167, 2020. Cited By :21.
- [35] Li, Z. Stop building castles on a swamp! the crisis of reproducing automatic search in evidence-based software engineering. En *2021 IEEE/ACM 43rd International*

Conference on Software Engineering: New Ideas and Emerging Results (ICSE-NIER), págs. 16–20, 2021.

- [36] Díaz, E., Panach, J. I., Rueda, S., y Distanto, D. A family of experiments to generate graphical user interfaces from bpmn models with stereotypes. *Journal of Systems and Software*, 173, 2021. Cited By :1.
- [37] Press, N. A. *How people learn brain, mind, experience, and school*. National Academy Press, Washington, D.C, expanded ed. edición, 2000.
- [38] Nehm, R. y Ridgway, J. What do experts and novices “see” in evolutionary problems? *Evo Edu Outreach*, 2011.
- [39] AL-Badareen, A. B., Selamat, M. H., A. Jabar, M., Din, J., y Turaev, S. Software quality models: A comparative study. En *Software Engineering and Computer Systems*, págs. 46–55, Berlin, Heidelberg, 2011. Springer Berlin Heidelberg.
- [40] ISO Central Secretary. System and software quality requirements and evaluation. Standard ISO/IEC 25000:2014, International Organization for Standardization, Geneva, CH, 2014.
- [41] A., C. Mapeos sistemáticos de literatura asequibles y reproducibles aplicando gestión del conocimiento y minería de procesos. *Universidad de las Fuerzas Armadas - ESPE*, 2019.
- [42] Leedy, P., Ormrod, J., y Johnson, L. *Practical research: planning and design*. Pearson, NY, NY, 12th edición, 2019.
- [43] de Oliveira, F., Torkar, R., y Machado, P. An initiative to improve reproducibility and empirical evaluation of software testing techniques. *2015 IEEE/ACM 37th IEEE International Conference on Software Engineering*, págs. 575–578, 2015.
- [44] Community, T. T. W. *The Turing Way: A handbook for reproducible, ethical and collaborative research*. Zenodo, July 2022.
- [45] Shull, F., Basili, V., Carver, J., J., M., Travassos, G., Medoca, M., y Fabbri, S. Replicating software engineering experiments: Addressing the tacit knowledge problem. *Proceedings of the 2002 International Symposium on Empirical Software Engineering (ISESE'02)*, 2002.
- [46] Mendonca, M., Madolando, J., Oliviera, M., Carver, J., Fabbri, S., Shull, F., Travassos, G., y Höhn. E, Basili, V. A framework for software engineering experimental replications. *13th IEEE International Conference on Engineering of Complex Computer Systems*, págs. 203–212, 2008.

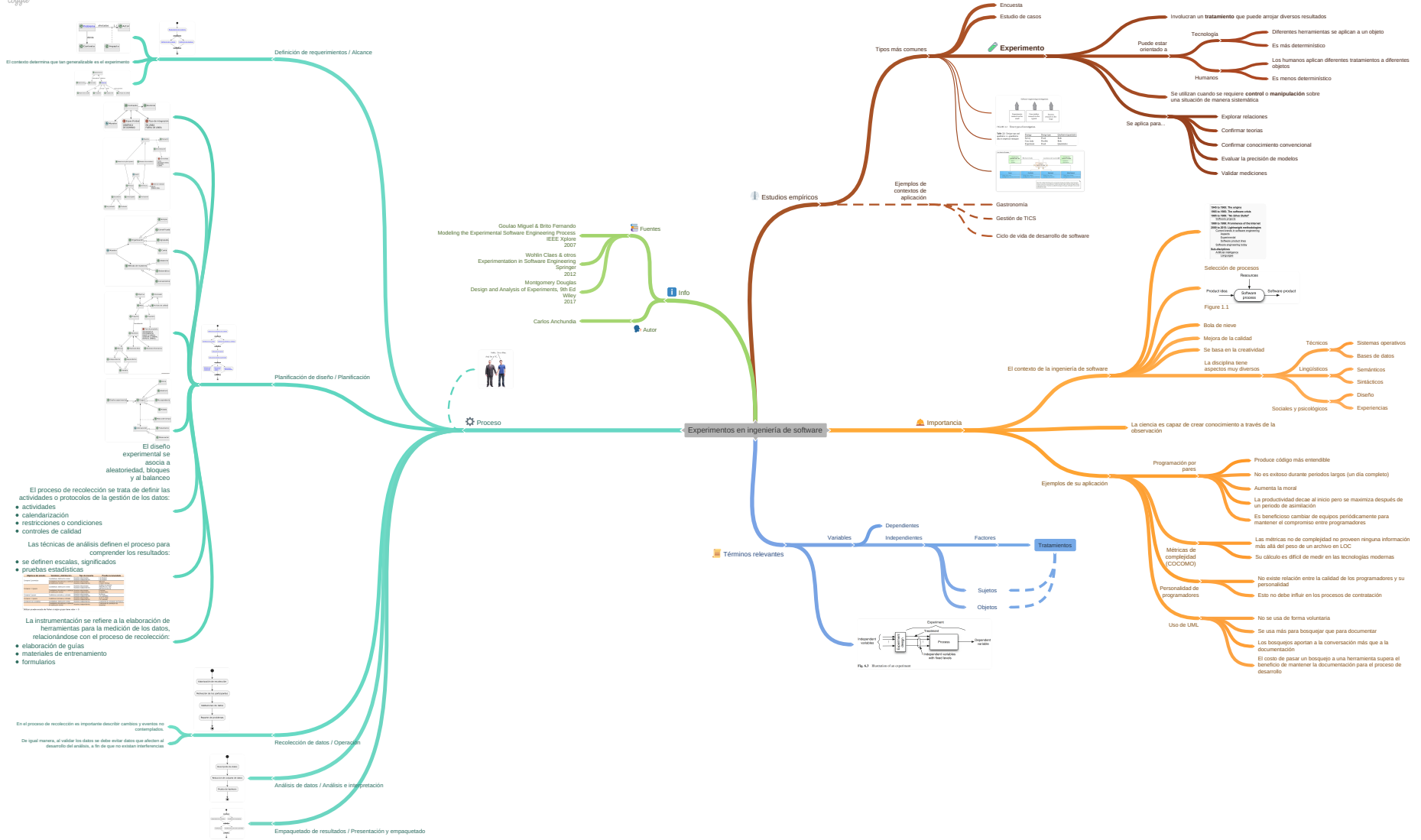
- [47] Shull, F., Mendonca, M., Basili, V., Carver, J., Maldonado, J., Fabbri, S., Travassos, G., y Ferreira, M. Knowledge-sharing issues in experimental software engineering. *Empirical Software Engineering*, págs. 111–137, 2004.
- [48] Lung, J., Aranda, J., Easterbrook, S., y Wilson, G. On the difficulty of replicating human subjects studies in software engineering. *Proceedings - International Conference on Software Engineering*, págs. 191–200, 01 2008.
- [49] Sandelin, T. y Vierimaa, M. Empirical studies in esernet. *Lecture Notes in Computer Science*, 2765:39–54, 01 2003.
- [50] Araújo, I., Silva, W., Nunes, J., y Neto, F. Arrestt: A framework to create reproducible experiments to evaluate software testing techniques. *1st Brazilian Symposium*, págs. 1–10, 09 2016.
- [51] Craze, D., Mendonça, M., Basili, V., Shull, F., y Jino, M. Extracting information from experimental software engineering papers. *XXVI International Conference of the Chilean Society of Computer Science (SCCC'07)*, págs. 105 – 114, 12 2007.
- [52] Solari, M. Identifying experimental incidents in software engineering replications. *2013 ACM / IEEE International Symposium on Empirical Software Engineering and Measurement*, págs. 213–222, 2013.
- [53] Biffi, S., Serral, E., Winkler, D., Dieste, O., y Juristo, N. Replication data management: Needs and solutions. *2013 ACM / IEEE International Symposium on Empirical Software Engineering and Measurement*, págs. 233–242, 2013.
- [54] Franca, B. y Travassos, G. Experimentation with dynamic simulation models in software engineering: planning and reporting guidelines. *Empi Software Eng*, 2015.
- [55] Fonseca, L., Seaman, C., y Soares, S. Describing what experimental software engineering experts do when they design their experiments - a qualitative study. *2017 ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM)*, págs. 211–216, 11 2017.
- [56] Solari, M., Vegas, S., y Juristo, N. Content and structure of laboratory packages for software engineering experiments. *Information and Software Technology*, págs. 64–79, 2018.
- [57] Juristo, N. y Vegas, S. Design patterns in software maintenance: An experiment replication at upm. *2011 Second International Workshop on Replication in Empirical Software Engineering Research*, págs. 7–14, 2011.
- [58] Miller, J. Replicating software engineering experiments: a poisoned chalice or the holy grail. *Information and Software Technology*, 47:233–244, 2005.

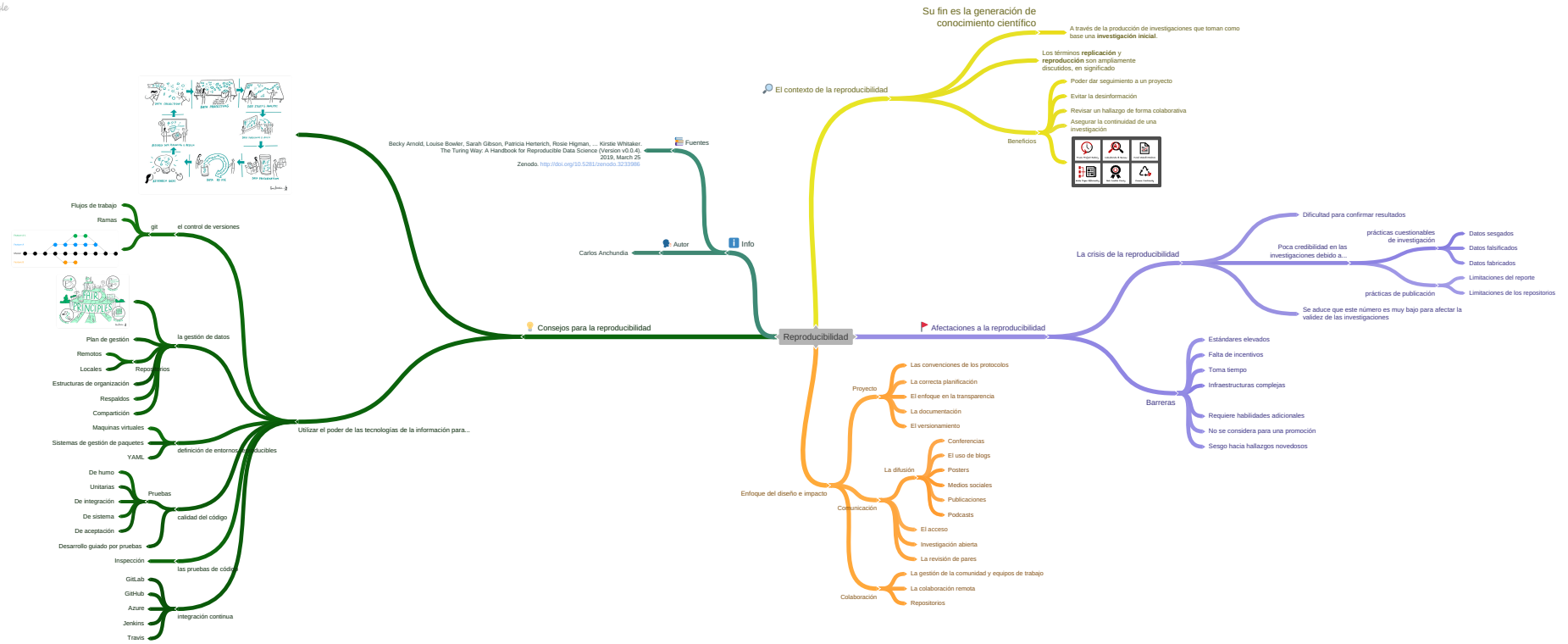
- [59] Ferreira, W., Baldassarre, M., Soares, S., Cartaxo, B., y Visaggio, G. A comparative study of model-driven approaches for scoping and planning experiments. *2013 27th Brazilian Symposium on Software Engineering*, págs. 78–87, 06 2017.
- [60] C. de Magalhaes, C., Silva, F., y de Souza Santos, R. Investigations about replication of empirical studies in software engineering: Preliminary findings from a mapping study. *ACM International Conference Proceeding Series*, 05 2014.
- [61] Rolland, C. Modeling the requirements engineering process, 1993.
- [62] Rolland, C., Prakash, N., y Benjamen, A. A multi-model view of process modelling. *Requirements Engineering, Springer Verlag*, págs. 169–187, 1099.
- [63] Harmsen, F., Brinkkemper, S., y Oei, J. Situational method engineering for information system project approaches. *IFIP Transactions A: Computer Science and Technology*, págs. 169–194, 01 1994.
- [64] Hommes, B. y Van Reijswoud, V. Assessing the quality of business process modelling techniques. *System Sciences, 2000. Proceedings of the 33rd Annual Hawaii International*, págs. 10 pp. vol.1, 02 2000.
- [65] Rey, A. *Libro Electrónico Multimedial: Evaluación de la Calidad de la Tecnología Educativa*. Universidad de Santander, 2015.
- [66] Montgomery, D. *Design and Analysis of Experiments*. Wiley, 2017.
- [67] Salman, I., Misirli, A. T., y Juristo, N. Are students representatives of professionals in software engineering experiments? En *2015 IEEE/ACM 37th IEEE International Conference on Software Engineering*, volume 1, págs. 666–676, 2015.
- [68] Creswell, J. W. y Creswell, J. D. *Research design: Qualitative, quantitative, and mixed methods approaches*. SAGE Publications, Thousand Oaks, CA, 5 edición, 2017.
- [69] James, C. y Rocco, P. Resolving the 50-year debate around using and misusing likert scales. *MEDICAL EDUCATION*, 42:1150–1152, 2008.
- [70] Norman, G. Likert scales, levels of measurement and the “laws” of statistics. *Advances in health sciences education : theory and practice*, 15:625–32, 02 2010.
- [71] Cantor, G. Ueber unendliche, lineare Punktmannichfaltigkeiten I. *Mathematische Annalen*, 15:1–7, 1879.

Anexos

Anexo 1

Material preparado para la inducción sobre experimentos y reproducibilidad.





Becky Arnold, Louise Bowler, Sarah Gibson, Patricia Hertenich, Rosie Highman, ... Kirstie Whitaker.
 The Turing Way: A Handbook for Reproducible Data Science (Version v0.0.4),
 2019, March 25
 Zenodo. <https://doi.org/10.5281/zenodo.3233986>

Puentes

Info

Autor

Carlos Anchunda

Su fin es la generación de conocimiento científico

A través de la producción de investigaciones que toman como base una investigación inicial.

Los términos replicación y reproducción son ampliamente discutidos, en significado

Poder dar seguimiento a un proyecto

Evitar la desinformación

Revisar un hallazgo de forma colaborativa

Asegurar la continuidad de una investigación



Afectaciones a la reproducibilidad

Proyecto

Las convenciones de los protocolos

La correcta planificación

El enfoque en la transparencia

La documentación

El versionamiento

La difusión

Conferencias

El uso de blogs

Posters

Medios sociales

Publicaciones

Podcasts

Comunicación

El acceso

Investigación abierta

La revisión de pares

Colaboración

La gestión de la comunidad y equipos de trabajo

La colaboración remota

Repositorios

La crisis de la reproducibilidad

Dificultad para confirmar resultados

Poca credibilidad en las investigaciones debido a...

prácticas cuestionables de investigación

Datos sesgados

Datos fabricados

Limitaciones del reporte

prácticas de publicación

Limitaciones de los repositorios

Se aduce que este número es muy bajo para afectar la validez de las investigaciones

Barreras

Estándares elevados

Falta de incentivos

Toma tiempo

Infraestructuras complejas

Requiere habilidades adicionales

No se considera para una promoción

Sesgo hacia hallazgos novedosos

Anexo 2

Material preparado para comprobar el aprendizaje y refuerzo sobre los temas presentados.

¿Cuáles son las fases del CVDS?

- 1 Aplicación
- 2 Votación (0)
- 3 Resultado

1 analisis de requisitos, diseno, codificacion, pruebas, mantenimiento

1 Requerimientos
Diseño
Implementación

1 Requisitos, análisis, diseño, implementación, pruebas, mantenimiento

1 Planificacion
Análisis
Diseño

1 Analisis y requerimientos
Diseño
Implementación

1 Determinación de requerimientos
Diseño del sistema
Desarrollo de Software

1 analisis
diseño

1 Planificacion, Analisis,
Implementación, Pruebas

1 Análisis
Diseño
Implementación



¿Qué impactos y problemas existen?

- 1 Aplicación
- 2 Votación (0)
- 3 Resultado

1 Impacto: Diferentes sistemas para usuarios (software y hardware)
Problemas: Aumento de precios

1 - Diferentes SO
- Compatibilidad, Rendimiento, Velocidad

1 Problemas de compatibilidad

1 Uso del sistema operativo basado en Unix

1 Problema: Compatibilidad
Impacto: precio

1 -Puede darse una baja aceptación del usuario
- La compatibilidad con otros

1 Problema: sistema operativo diferente
Impacto : Precio

1 Problema: Necesidad de desarrollar un software multiples veces debido al uso de funciones

1 Problema: dificultad de comprensión de los requerimientos.

1



¿Qué participantes podemos considerar?

1 Aplicación

2 Votación (0)

3 Resultado

1 Las computadoras	1 Computadoras Desarrolladores	1 Mac y Pc
1 Computadoras	1 Computadoras	1 Desarrolladores
1 desarrolladores	1 computadores	1 computadoras, desarrolladores

👏 10

¿De qué forma podemos agrupar las muestras?

1 Aplicación Todavía no hay ideas.

2 Votación (0)

3 Resultado

👏 0

¿Qué podríamos medir?

- 1 Aplicación
- 2 Votación (0)
- 3 Resultado

1 Tiempos de respuesta de algun proceso

1 Las funcionalidades

1 Calidad, tiempo

1 -Velocidad, capacidad de almacenamiento
- Tiempo de ejecución

1 cantidad de codigo

1 velocidad de ejecución,

1 Porcentaje de excepciones entre sistemas operativos

1 Velocidad de respuesta en los procesos

1 Velocidad de ejecución, almacenamiento y respuesta

1

1

¿Cómo podemos definir para proceso de recolección?

- 1 Aplicación
- 2 Votación (0)
- 3 Resultado

1 Encuestas

1 Encuestas y cuestionarios a un grupo determinado.

1 Analizando de forma directa al sujeto y el comportamiento de la computadora ante el proceso que

1 Los ordenadores participantes deben tener las mismas características

1 Cuando: Cuando quiero sacar un software al mercado para ambos Mac/PC

1 Analizar población y establecer técnica de recolección, análisis de la data y conclusiones

1 Observando al objeto que será estudiado con el fin de que se determinen sus características

1 Restricción: aplicaciones que esten disponibles para ambos sistemas operativos

1 Agrupar las computadoras de acuerdo a su SO
Determinar los tiempos de

1

1

¿Qué instrumentos podemos considerar?

- 1 Aplicación Todavía no hay ideas.
- 2 Votación (0)
- 3 **Resultado**

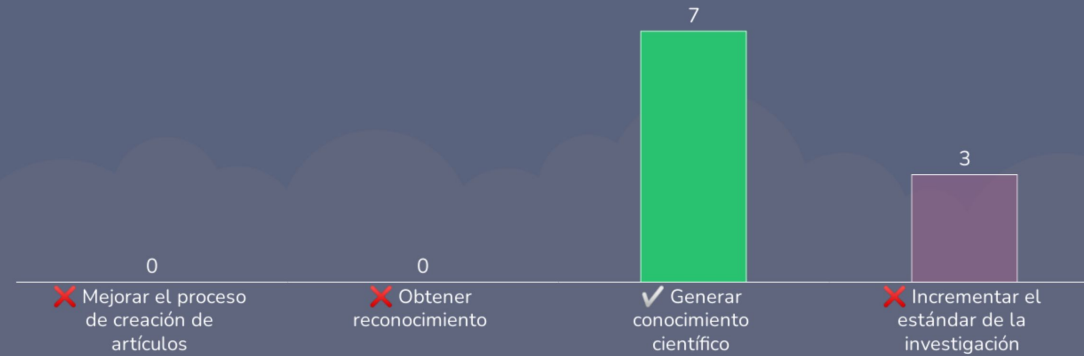
👏 0

¿Cómo podemos registrar problemas?

- 1 Aplicación Todavía no hay ideas.
- 2 Votación (0)
- 3 **Resultado**

👏 0

¿Cuál es el propósito de poder reproducir o replicar experimentos?



👍 1 🙄 17 🙋 10

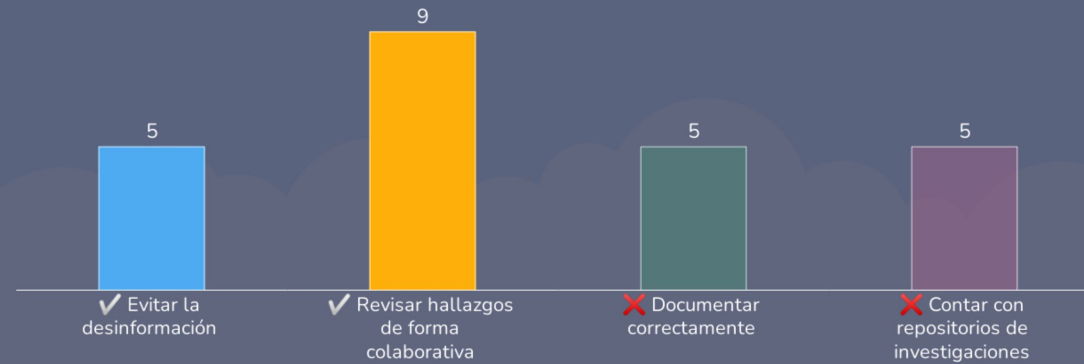
Leaderboard

13 jugadores

1	👤 Shirley Otuna	82p +82
2	👍 Andrés López	81p +81
3	😊 Michelle Cantuña	77p +77
4	😎 Maycol Tituaña	76p +76
5	😊 Jerico Ruiz	62p +62

👍 1 🙄 17

Seleccione dos beneficios de poder reproducir una investigación



👍 1 🙄 17 🖱️ 12

13 jugadores

1	Shirley Otuna	170p +88
2	Michelle Cantuña	161p +84
3	Marcelo Malte	135p +75
4	Andrés López	81p +0
5	Maycol Tituaña	76p +0

👍 1 🙄 17

La crisis de la reproducibilidad se refiere a...



👍 1 🗳️ 17 🙋 13

Leaderboard

13 jugadores

1	👤 Shirley Otuna	262p +92
2	😊 Michelle Cantuña	250p +89
3	👍 Andrés López	163p +82
4	😎 Maycol Tituaña	160p +84
5	😬 Christopher Yepez	144p +73



👍 1 🗳️ 17

Empajere los enfoques de la reproducibilidad con sus prácticas o ejen

Proyecto	1	A	Gestionar la difusión a través de blogs
Comunicación	2	B	Gestionar habilidades adicionales
Colaboración	3	C	Gestionar una comunidad y equipos
		D	Gestionar la documentación

👍 1 🤔 17 🙋 11

Leaderboard

13 jugadores

1	🤗 Michelle Cantuña	303p +53
2	👤 Shirley Otuna	262p +0
3	👍 Andrés López	163p +0
4	🕶️ Maycol Tituaña	160p +0
5	🤖 Christopher Yep..	144p +0

👍 1 🤔 17

Empareje la aplicación de TICS para la mejora de reprod

Control de versiones 1

Gestión de datos 2

Entornos reproducibles 3

A Inspecciones

B Git


C Máquinas virtuales o archivos Yaml

D Estructurar repositorios y directorios

👍 1 🤔 17 🙌 13

Leaderboard

13 jugadores

1	 Michelle Cantuña	376p +73
2	 Shirley Otuna	326p +64
3	 Andrés López	244p +81
4	 Marcelo Quiroga	221p +86
5	 Jerico Ruiz	216p +83

Michelle Cantuña

376 puntos

👍 1 🤔 17

Anexo 3

Tarea #8 de forma grupal sobre la planificación de un experimento.



Investigación de Ingeniería de Software – NRC: 8506

GRUPO 1

Integrantes:

Andrade Peñafiel Mateo Nicolás (C)

Venegas Torres Camila Vanessa

Tituaña Tupiza Maycol Estalin

Carrera: Software

Fecha: 21/06/2022

Objetivo:

Realizar la planificación de un experimento enfocado en determinar si un PC o una MAC son mejores para el desarrollo de software.

Introducción:

Dado el avance tecnológico en la industria del desarrollo de software, se ha visto la necesidad de emplear computadores con características avanzadas que brinden un mejor entorno de desarrollo para los programadores, dado el caso se tiene a dos marcas importantes en el mercado tal como: PC y Mac, las dos desarrolladas en un alto nivel, sin embargo se tiene ciertas dudas sobre cual es mejor en ciertos aspectos, por consiguiente se procede con el siguiente experimento que propone comparar las dos marcas en diferentes tipos de software.

Definición de parámetros de contexto

Contexto: Desarrollo de software.

Dominio: Las PC y Mac de diferentes características y especificaciones.

Muestra: Conjunto de computadores que se evaluarán que cumplan ciertas características.

Identificación de sujetos

Sujeto: Los desarrolladores que van a comprobar el rendimiento, la compatibilidad y velocidad que tienen estos ordenadores en el desarrollo de software.

Objeto: Los ordenadores que contienen varias especificaciones que puedan ampliar un rango de estudio.

Definición de hipótesis y variables

Las computadoras Mac poseen mayor rendimiento que las PC cuando nos referimos al desarrollo de software, además de ser más estables y proveer al programador de más seguridad frente a virus.

Selección de muestra

Nos referimos no solo al grupo de desarrolladores, sino también al grupo de ordenadores Mac y PC que serán evaluados. Todos estos pertenecientes al campo del desarrollo de software.

Selección del diseño experimental

Utilizaremos el método de observación donde pondremos a prueba a diferentes computadoras con similares características donde probaremos el mismo programa en los computadores, y evaluaremos el tiempo de ejecución y velocidad del programa en el ordenador.

VARIABLES OBSERVADAS: Tiempo de ejecución, Tiempo de respuesta,

RELACIÓN ENTRE ELEMENTOS: Relación entre el tiempo de ejecución y la nitidez y funcionamiento del programa en la computadora que está siendo evaluada.

Definición del proceso de recolección

Actividades a realizar:

1. De acuerdo a la muestra obtenida se deberá obtener ordenadores de Mac y PC de última o décima generación.
2. Los ordenadores deben poseer las mismas características como: memoria RAM, disco duro o sólido, tarjeta gráfica, procesador etc.
3. Se deberá instalar en los ordenadores todo lo necesario para el desarrollo de software como: IDEs de programación, base de datos, tecnologías, frameworks etc.
4. Los desarrolladores ocuparan los ordenadores para desarrollar un software en específico, todos ellos comenzarán al mismo tiempo.
5. Los observadores deberán tomar nota de todo lo que se les presente a los participantes como: problemas que tuvieron, tiempo de finalización, tiempo de ejecución del software etc.
6. Por último se realizará el análisis de todos los datos obtenidos en este caso a través de una tabla comparativa para poder ver cual de los dos ordenadores es mejor para el desarrollo de software.

Restricciones

Las comparaciones de rendimiento deben ser entre ordenadores de la misma generación y tener características similares.

Selección de técnicas de análisis

Ejecución de una Tabla Comparativa de los tiempos de ejecución de cada programa en los distintos ordenadores de PC y Mac que están siendo evaluados.

Selección de instrumentación

Un cuestionario que será interpretado en una hoja de excel en la que compararemos mediante ecuaciones que evalúen el tiempo de ejecución de algoritmos la reacción y eficiencia de los ordenadores en las distintas pruebas a las que serán sometidas.

ESPE

UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA



DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN

CARRERA:

ING. DE SOFTWARE

NRC:

8506

ASIGNATURA:

INVESTIGACIÓN EN INGENIERÍA DE SOFTWARE

TEMA:

PLANIFICACIÓN DE EXPERIMENTO

GRUPO N°: 2

FECHA:

21/06/2022

ÍNDICE

- 1. INTRODUCCIÓN**
- 2. OBJETIVOS**
- 3. DESARROLLO**
 - 3.1 DEFINICIÓN DE PARÁMETROS DE CONTEXTO**
 - 3.2.1 IDENTIFICACIÓN DE SUJETOS**
 - 3.2.2 DEFINICIÓN DE HIPÓTESIS Y VARIABLES**
 - 3.3 SELECCIÓN DE MUESTRA**
 - 3.4 SELECCIÓN DEL DISEÑO EXPERIMENTAL**
 - 3.5.1 DEFINICIÓN DEL PROCESO DE RECOLECCIÓN**
 - 3.5.2 SELECCIÓN DE TÉCNICAS DE ANÁLISIS**
 - 3.5.3 SELECCIÓN DE INSTRUMENTACIÓN**
- 4. CONCLUSIONES**
- 5. REFERENCIAS**

1. INTRODUCCIÓN

En base a los últimos conocimientos impartidos en las clases de investigación de ingeniería de software que se enfocan a la realización de experimentos para la adquisición de nuevos conocimientos en base a un tema específico.

Con todo esto en mente se ha planteado la realización del planteamiento de un experimento que determine si cual es el mejor implemento tecnológico para el desarrollo de software si una PC o una MAC. Basándonos tanto en tiempos de respuesta y otros puntos que puedan decir cual de los dos es el mejor en los aspectos a tratar.

2. OBJETIVOS

- Planificación de las bases del experimento y todos los aspectos a experimentar.
- Determinar los aspectos más favorables de cada parte, tanto de PC como de MAC.

3. DESARROLLO

3.1 DEFINICIÓN DE PARÁMETROS DE CONTEXTO

En base a la teoría vista y el contexto de experimento que se ha planteado tenemos dos parámetros a estudiar y a ser relacionados. Por un lado tenemos nuestros objetos de experimentación que serán los diferentes dispositivos tecnológicos de la rama de computadoras que usen el sistema operativo de Windows y por otra parte las que usan MacOs.

Estos objetos se verán puestos a prueba en diferentes programas y aplicaciones que identificamos como nuestro parámetro de artefactos que sean enfocados al desarrollo de software que nos permitan identificar cual de estos dos dispositivos tiene ventajas el uno sobre el otro, identificando así el más óptimo e indicado para desarrolladores.

3.2.1 IDENTIFICACIÓN DE SUJETOS

Los sujetos que se identificaron para este experimento basado en el contexto establecido son los Desarrolladores, ya que estos pueden realizar su trabajo tanto en sistemas operativos MacOs como en Windows. También las computadoras en si las Pc y Mac y el conjunto de programas o software que se van probar en los diferentes SO para poder comprobar las diferentes características.

3.2.2 DEFINICIÓN DE HIPÓTESIS Y VARIABLES

3.2.2.1 Hipótesis

Hipótesis Alternativa: El rendimiento, compatibilidad de un programa específico cambia si se ejecuta en los sistemas operativos MacOs y Windows.

Hipótesis Nula: El rendimiento, compatibilidad de un programa específico no cambia si se ejecuta en los sistemas operativos MacOS y Windows

3.2.2.2 Variables

Variable dependiente: Tiempo de respuesta de los diferentes procesos del programa o software dependiendo en que sistema operativo se encuentre ejecutándose. La velocidad de ejecución.

Variable independiente: Las funcionalidades que tiene el programa en los diferentes sistemas operativos.

3.3 SELECCIÓN DE MUESTRA

La muestra en este experimento son las computadoras que se pueden organizar de forma que se encuentren categorizadas por sus características, en Mac las memorias ram mínimo 8GB, procesador mínimo Intel Core i5 de doble núcleo y 1.8 GHz en computadoras Windows 8 GB de memoria ram como mínimo para 64 bits, procesador Intel Core i3-10100TE y 2.42 GHz.

3.4 SELECCIÓN DEL DISEÑO EXPERIMENTAL

Se seleccionó un diseño experimental de grupo Aleatorio debido a que las aplicaciones seleccionadas para realizar las pruebas experimentales pueden ser cualquiera, lo que se pretende es medir el tiempo de ejecución de dichos programas y la intervención es de tipo Observación ya que se comparará el tiempo de ejecución entre MacOS y Windows para así determinar cual es mejor para el desarrollo de software.

3.5.1 DEFINICIÓN DEL PROCESO DE RECOLECCIÓN

A continuación se definen las actividades que nos ayudarán al proceso de recolección:

Cuándo: Cuando se desea conocer un análisis de cuál sistema operativo será el mejor para una empresa o para un uso personal.

Restricciones: Los dispositivos deberán contar con similares características con respecto al almacenamiento o generación del procesador.

Cómo: Ejecutar el programa mediante el código y guardar el tiempo de ejecución.

3.5.2 SELECCIÓN DE TÉCNICAS DE ANÁLISIS

Para la selección de técnicas de análisis se ve involucrada una parte de estadística para comprender los resultados, en este caso los tiempos de ejecución, ya que por cada sistema operativo, se realizará un promedio y entre estas dos variables cuantitativas la prueba escogida es t de Student según (Rodó, 2022) "Es una distribución de probabilidad que estima el valor de la media de una muestra pequeña extraída de una población que sigue una distribución normal y de la cual no conocemos su desviación típica". Esta distribución estadística es la recomendada ya que funciona de gran manera en una muestra pequeña cómo es el caso, además de comprar únicamente dos valores.

3.5.3 SELECCIÓN DE INSTRUMENTACIÓN

Utilizaremos un formulario en el cual podremos evaluar cuantitativamente los tiempos obtenidos de cada una de las repeticiones en el experimento. Se ejecutará por lo menos diez veces para tener nuestra muestra y poder empezar a tabular datos, se tomará en cuenta cada uno de los programas evaluados en los dos sistemas operativos tal que sean compatibles y multiplataforma. Después de eso se podrá calificar aspectos de carácter cualitativo como podría ser el diseño visual, experiencia de usuario, etc.

4. CONCLUSIONES

Se logró determinar los aspectos favorables y desfavorables en cada uno de los aspectos tomados en cuenta a la hora de establecer la planificación del experimento. Se concluye que la experimentación fue una técnica adecuada para poder comparar los dos OS.

Se logró desarrollar una planificación de un experimento en base a dos sistemas operativos.

5. REFERENCIAS

Carlos .A.(2022). Experimentos en ingeniería de Software

<https://coggle.it/diagram/YqdOlrdxQl8Lc0r/t/experimentos-en-ingenier%C3%ADa-de-software/480f7554f8826df791a4d10e65d8ada33cb8485f3dfbaeb30024496d7f46c826>

Rodó, P. (2022). *Distribución t de Student*. Economipedia.

<https://economipedia.com/definiciones/distribucion-t-de-student.html>



**DEPARTAMENTO EN CIENCIAS DE LA
COMPUTACIÓN
CARRERA DE INGENIERÍA DE SOFTWARE**

**INVESTIGACIÓN EN LA INGENIERÍA DE
SOFTWARE**

NRC: 8506

Planificación de Experimento

INTEGRANTES:

Michelle Cantuña

Andrés Coronel

Erick Riascos

Christopher Yépez

DOCENTE:

Ingeniero Efrain R. Fonseca C.

FECHA: 2022/06/20

SANGOLQUÍ-ECUADOR

ÍNDICE

OBJETIVOS	3
OBJETIVO GENERAL	3
OBJETIVOS ESPECÍFICOS	3
DESARROLLO	3
Definición de requerimientos/ Alcance	3
1.1. Planteamiento del problema	3
1.2. Definición contexto	3
1.3. Definición de objetivos	4
Planificación de diseño	4
2.1. Definición de parámetros de contexto	4
2.2. Identificación de sujetos	4
2.3. Definición de hipótesis y variables	5
2.4. Selección de muestra	5
Alien Aurora R6 PC:	5
MacBook Pro:	5
2.5. Selección del diseño experimental	6
2.6. Definición del proceso de recolección	6
2.7. Selección de técnicas de análisis	6
2.8. Selección de instrumentación	6
CONCLUSIONES	8
RECOMENDACIONES	8
REFERENCIAS	8

OBJETIVOS

OBJETIVO GENERAL

Realizar la planificación de un experimento enfocado en determinar si un PC o una MAC son mejores para el desarrollo de software mediante el seguimiento de los pasos establecidos en fuentes bibliográficas.

OBJETIVOS ESPECÍFICOS

- Reconocer el procedimiento para realizar un experimento de manera estructurada y organizada
- Planificar un experimento correctamente, utilizando la información presentada y desarrollada.

DESARROLLO

1. Definición de requerimientos/ Alcance

1.1. Planteamiento del problema

Es necesario conocer qué tipo de dispositivos entre PC y MAC permite un mejor desarrollo para la creación de software, esto principalmente por las diferencias constantes que se presentan entre MAC y diferentes dispositivos y su propia creación de un sistema para MAC.

Problema

Determinar qué tipo de dispositivo entre MAC y PC permite a los desarrolladores de software una mejor creación de productos software mediante el cálculo de diferentes cualidades cuantitativas en la eficiencia de creación.

Actores

Desarrolladores de Software.

Impacto

Permitir mayor eficiencia en la creación de nuevo software y su desarrollo gracias a la utilización de dispositivos eficientes en los ámbitos tecnológicos y de funcionalidad.

1.2. Definición contexto

Los desarrolladores de software han creado diferentes tipos de productos software a lo largo del tiempo, pero hace dos décadas la compañía Apple sacó sus propios productos con un sistema operativo distinto, que pretendía tener su propio estilo y diseño, lo que motivó a muchos programadores a desarrollar en MAC.

La mayor diferencia que otorga MAC es su sistema operativo de Apple MAC OS, que pretende ser una competencia con windows, y linux. Su simplicidad y facilidad de uso

hace que sea más intuitivo de usar que otros sistemas operativos y su funcionalidad con productos Apple hace que utilizar como viene predeterminado sea la mejor forma de utilizar estos productos (Balli & Korukoğlu, 2009).

Por ello, los programadores se han preguntado realmente qué es mejor usar, una MAC o un PC con diferente sistema operativo y desarrollo propio. Es necesario conocer las ventajas y desventajas de MAC comparado a los PCs y obtener resultados cuantitativos sobre las diferencias tecnológicas entre los dispositivos.

1.3. Definición de objetivos

Para definir los objetivos es necesario reconocer que realiza cada objetivo, su razón de ser, a qué apunta, que realiza, como lo realiza y quien se encarga del objetivo.

Objetivos

Definir cuál dispositivo PC o MAC tiene mayor eficiencia en su tiempo de respuesta, funcionalidad y velocidad de ejecución para reconocer el mejor entre ellos mediante un experimento realizado por un grupo de estudio.

Reconocer las diferencias entre PC y MAC en ámbitos tecnológicos de compatibilidad para comprender las dificultades de producir software y obtener el dispositivo que permite mayor compatibilidad, mediante un experimento de reproducción de software realizado por un grupo de estudio.

2. Planificación de diseño

2.1. Definición de parámetros de contexto

El desarrollo de software es un campo que tiene que ver con el conjunto de actividades informáticas que tiene como funcionalidad la creación, arquitectura, despliegue y la compatibilidad del software (Leboso & Gironés, 2016). Por tanto el software son los programas o conjuntos de instrucciones que le dicen que hacer a una computadora, es por esa razón que es necesario realizar un experimento que nos permita determinar si una PC o Mac es mejor para poder desarrollar el software.

Como se sabe esto es uno de los dilemas de mayor relevancia que poseen los desarrolladores de software, al momento de no saber que sistema operativo usar, sin embargo el más popular a nivel mundial, por otro lado algunos eligen Mac, es por eso analizar a fondo cada uno de estos a través de parámetros como : seguridad, calidad, tiempo de respuesta, compatibilidad (Maida & Pacienza, 2015).

2.2. Identificación de sujetos

Los sujetos son: Mac, PC, desarrolladores.

2.3. Definición de hipótesis y variables

Ho: El 75% de desarrolladores de software no prefieren PC que Mac para el desarrollo de software porque tiene compatibilidad absoluta y hace el mejor uso de los recursos.

Ha: El 75% de desarrolladores de software prefieren PC que Mac para el desarrollo de software porque tiene compatibilidad absoluta y hace el mejor uso de los recursos.

Variable independiente: El 75% de desarrolladores de software prefieren PC que Mac para el desarrollo de software

Variable dependiente: Tiene compatibilidad absoluta y hace el mejor uso de los recursos.

2.4. Selección de muestra

Para realizar un experimento comparativo se tomará el muestreo de varios equipos tanto MAC como PC, dichos equipos deberán contar con aproximadamente las mismas características para lograr hacer una comparativa más eficiente y sin sesgos entre ellos:

Alien Aurora R6 PC:

Características:

- 4.2 Ghz 4-core i7
- 16Gb 2666Mhz DDR4
- NVIDIA GTX 1070 8Gb DDR5
- Precio: 1.600 dólares

MacBook Pro:

Características

- 3.1 Ghz 4-core i7
- 16Gb 2133 Mhz DDR3
- Pasion Pro 560 4Gbram
- Precio: 3.399 dólares

Añadido a esto un conjunto de IDE's ocupados por desarrolladores para realizar las respectivas pruebas tales como:

- Netbeans
- Visual Studio 2022

- CodeBlock

2.5. Selección del diseño experimental

Se seleccionará un diseño experimental de un grupo único debido a que ya se establecieron los programas y equipos que se van a utilizar en las pruebas experimentales, lo que pretende medir el tiempo de ejecución del software establecido en los equipos establecidos, para la determinación cual es mejor estadísticamente.

2.6. Definición del proceso de recolección

Cuándo: Cuando se desea conocer un análisis de qué equipos son mejores para el uso en diferentes ámbitos ya sea laboral o personal.

Restricciones: los equipos deberán contar con las mismas características aproximadamente para evitar los sesgos y obtener un resultado más fiable.

Como: Ejecutar los diferentes IDE's de desarrollo para comparar los tiempos de compilación y ejecución

2.7. Selección de técnicas de análisis

En este caso se van a tener variables cuantitativas (velocidades de respuesta, excepciones generadas por el SO, etc.) con distribución normal, donde se va a buscar comparar promedios, por tanto, se utilizará la técnica de análisis de t de student para estos datos específicos.

A su vez, se tendrá en cuenta que hay variables cualitativas (usabilidad del SO) nominales y ordinales, donde se busca comparar 2 grupos de trabajo (aquellos que usan PC y aquellos que usan Mac) en muestras independientes, por tanto se utilizará chi-cuadrado.

2.8. Selección de instrumentación

Se utilizarán formularios que permitirán medir cuantitativamente los resultados obtenidos de la medición de las variables en los casos de que estas sean cuantitativas (número de excepciones presentadas en un programa dependiendo de si se utiliza PC o Mac, Velocidad de respuesta entre PC o Mac, etc.) de esta forma también se podrá utilizar para cuantificar elementos cualitativos (la UX del sistema, la UI del sistema, etc.)

Por ejemplo para la medición de una variable cuantitativa (en este caso el ejemplo propuesto es la velocidad de respuesta en diversas aplicaciones) se puede usar la tabla o formulario debajo para posteriormente utilizar una T de student.

Parámetro de medición	Velocidad de respuesta en el aplicativo X durante el tiempo de ejecución en ms	
	Nº	PC (ms)
1	50	54
2	42	42
3	60	65
4	50	55
5	52	52
6	46	50
7	65	65
8	32	40
9	52	56
10	51	59
Promedio	50	54

En el caso de pruebas cuantitativas se puede utilizar una encuesta luego de tratar de evaluar la variable correspondiente (en este ejemplo se utilizará la evaluación de la experiencia de usuario o UX luego de utilizar la PC o Mac)

1. ¿Cuál es el SO que utilizó?
 - PC
 - Mac

2. ¿Qué grado de responsividad siente que tiene el SO?
 - Muy baja
 - Baja
 - Moderada
 - Alta
 - Muy alta

3. ¿En qué grado siente que es simple navegar utilizando el SO?

- Muy bajo
 - Bajo
 - Moderado
 - Alto
 - Muy alto
4. ¿En qué grado considera que el SO es fácil de utilizar?
- Muy bajo
 - Bajo
 - Moderado
 - Alto
 - Muy alto
5. ¿En qué grado considera que el SO facilitó su experiencia con el desarrollo de software?
- Muy bajo
 - Bajo
 - Moderado
 - Alto
 - Muy alto

CONCLUSIONES

- Se siguió la estructura para la planificación de un experimento, verificando en las fuentes bibliográficas de tal manera que nos permitió realizar una investigación de manera eficaz y sistemática. Esto se puede observar claramente en el desarrollo donde se identificó claramente el problema, definición de parámetros, hipótesis, muestra y proceso de recolección del problema correspondiente.
- Se logró hacer la planificación de un experimento que se encuentra enfocado en determinar si un PC o una MAC son mejores para el desarrollo de software mediante el seguimiento de los pasos establecidos en fuentes bibliográficas.

RECOMENDACIONES

- Se recomienda hacer una revisión a profundidad de las definiciones dadas en clase y apoyarse de material bibliográfico para una correcta realización de la planificación.
- Realizar una revisión de material estadístico ya que se utilizara en gran medida.

REFERENCIAS

Leboso, L., & Gironés, J. (2016). —*Estudio de la herramienta 'RAD Studio' para el desarrollo de aplicaciones multiplataforma en Android, iOS, Mac y Windows*—. RiuNet.

Retrieved June 20, 2022, from

<https://riunet.upv.es/bitstream/handle/10251/74660/LEBOSO%20-%20Estudio%20de%20la%20herramienta%20%27RAD%20Studio%27%20para%20el%20desarrollo%20de%20aplicaciones%20multiplataform....pdf?sequence=3&isAllowed=y>

Maida, E., & Pacienza, J. (2015). *Metodologías de desarrollo de software*. Repositorio Institucional UCA. Retrieved June 20, 2022, from

<https://repositorio.uca.edu.ar/bitstream/123456789/522/1/metodologias-desarrollo-software.pdf>

Ballı, S., & Korukoğlu, S. (2009). Operating System Selection Using Fuzzy AHP and TOPSIS Methods. *Mathematical and Computational Applications*, 14(2), 119–130.

<https://doi.org/10.3390/mca14020119>



DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN

CARRERA

Ingeniería de Software

INGENIERO

Ing. Efrain Rodrigo Fonseca Carrera

INTEGRANTES

Gallardo Alava Juan Esteban
Montensen Franco Eduardo Antonio
Otuna Rojano Shirley Stefania
Perez Jacome Marco Santiago

MATERIA

Investigación en Ingeniería de Software

FECHA

2022/06/21

SEMESTRE Y NRC

S-I MAYO – SEPT 22 – 4322

1. DEFINICIÓN DE PARÁMETROS DE CONTEXTO

1.1. Contexto del problema:

Siendo Windows es la plataforma más popular a nivel mundial, tanto para el usuario común como para los programadores, existen personas que optan por utilizar un Mac, e incluso recomiendan este tipo de computadora por encima de las que poseen Windows. Por supuesto, cada sistema tiene sus beneficios y falencias

Por lo cual es importante conocer lo mejor que nos puede ofrecer cada uno de estos sistemas operativos.

¿Para el desarrollo de software es mejor una PC o Mac?

1.2. Método de muestreo: Para el siguiente experimento hemos escogido el método aleatorio para el muestreo.

2. IDENTIFICACIÓN DE SUJETOS

Para este experimento hemos seleccionado como sujeto a los desarrolladores de software, los cuáles pueden ser junior, semi senior y senior.

3. DEFINICIÓN DE HIPÓTESIS Y VARIABLES

3.1. Hipótesis Nula

Estabilidad

Algunos PC son inestables, algunas veces se bloquean, hay que apagarlos y reiniciarlos. Esto es cada vez menos frecuente, pero muchas personas han perdido información a causa de esto. Los computadores Mac rara vez se traban o bloquean.

3.2. Hipótesis Alternativa

- H0: algunos PC son inestables, algunas veces se bloquean, hay que apagarlos y reiniciarlos. Esto es cada vez menos frecuente, pero muchas personas han perdido información a causa de esto. Los computadores Mac rara vez se traban o bloquean.
- H1: algunos PC son estables, algunas veces no se bloquean, no hay que apagarlos o reiniciarlos. Los computadores Mac siempre se traban o bloquean.

Estas hipótesis son planteadas a base de la hipótesis nula, es decir, son respuestas alternativas a la hipótesis nula que pretenden demostrar su falsedad. Este tipo de hipótesis se acepta o no se acepta.

4. SELECCIÓN DE MUESTRA

4.1. Muestra no probabilística o dirigida

La elección de los elementos depende de causas relacionadas con las características del trabajo y del investigador o de quien constituye la muestra. El procedimiento no es

mecánico ni se hace con fórmulas de probabilidad, depende del proceso de toma de decisiones de una persona o de un grupo y, por supuesto, tiende a estar sesgadas.

Para este experimento seleccionaremos dos de las muestras no probabilistas que se adapte mejor a la situación y con la que se pueda obtener mejores resultados.

4.1.1. Sujetos voluntarios

Para determinar si un PC o una MAC son los mejores para el desarrollo de software, seleccionaremos a varios voluntarios que tengan experiencia con el uso de ambos de PC y MAC, para así determinar si son los mejores en cuanto a estabilidad y otros aspectos.

4.1.2. Expertos

Personas que opinen sobre el experimento. Ya son especialistas en el tema están más aptos a dar conocimientos acerca de este tema que es el determinar si un PC o una MAC son los mejores para el desarrollo de software.

5. SELECCIÓN DEL DISEÑO EXPERIMENTAL

El diseño experimental se asocia a aleatoriedad, bloques y al balanceo.

En este caso, se propone usar el Paradigma Experimental, si bien el paradigma experimental tiene sus orígenes en la física o la química, en la actualidad diversas disciplinas emplean la experimentación como método de obtención de nuevo conocimiento. Este conocimiento resultante puede ser entonces aplicado de manera confiable por profesionales (por ejemplo, ingenieros químicos o físicos, etc.) para resolver problemas que ocurren en la práctica de su disciplina.

La estructura básica de un experimento se conforma por dos tipos de variables conocidas también como factores (o variables independientes) y variables de respuesta (o variables dependientes). Los factores son aquellas variables que podemos manipular o controlar en un experimento, mientras que las variables de respuesta son variables que analizamos para observar el efecto que producen los cambios en los factores de interés.

En este caso de estudio, utilizaremos un proceso de experimentación genérico para obtener los resultados:

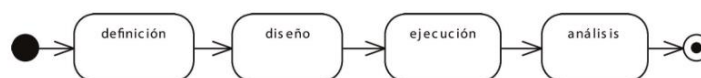


Gráfico 1. Proceso de experimentación genérico (Gómez et al., 2013).

Esto nos permitirá especificar de manera general los aspectos principales que debemos tener en cuenta para nuestro experimento a realizar.

Dependiendo del número de aspectos que podemos considerar como lo son:

- La Frecuencia de funcionamiento

- Número de núcleos e hilos de proceso
- Memoria caché
- Arquitectura del procesador

El diseño deberá contemplar la experimentación de estas 4 características, a fin de obtener un resultado efectivo.

En este caso, el instrumento de experimentación propuesto sería el llamado “Overclock”

5.1. Overclock

El overclocking es una técnica utilizada para aumentar la frecuencia de reloj de un componente electrónico, haciendo que esta sea superior a la que ofrece de serie por su fabricante. La frecuencia de reloj son los hercios del ordenador, los ciclos por segundo, e indican la velocidad a la que un ordenador realiza sus operaciones más básicas.

Cada ordenador tiene algunos componentes como la CPU (el procesador) o la GPU o tarjeta gráfica, que tienen una frecuencia de reloj concreta establecida por el fabricante. Por ejemplo, si coges unos procesadores como los últimos lanzados por Intel, estos tienen una frecuencia de reloj base de 3,70 GHz, y una frecuencia de reloj máxima de 5,30 GHz.

6. DEFINICIÓN DEL PROCESO DE RECOLECCIÓN

Con base a la experimentación, los aspectos a tomar en cuenta a la hora de contrastar la información serán los resultados de la experimentación, podremos evidenciar, mediante los procedimientos de Overclocking, la diferencia entre distintos tipos de procesadores en cuanto a rendimiento y mantenibilidad.

Según Cinebench (Cinebench es un conjunto de pruebas multiplataforma del mundo real que evalúa las capacidades de hardware de su computadora)

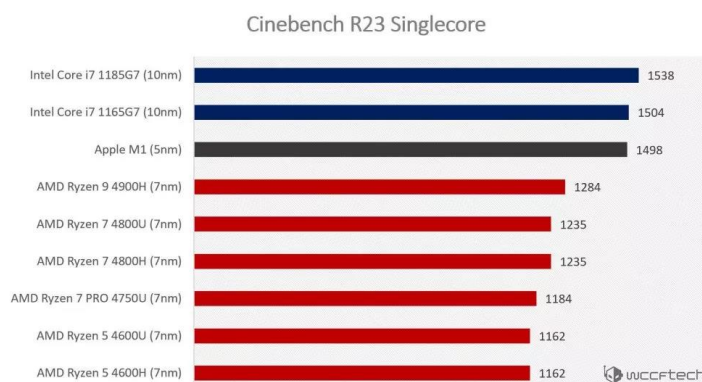


Gráfico 2. Cinebench R23 Singlecore

El Chip Apple M1 obtiene 1498 puntos en la prueba de un solo núcleo en Cinebench R23, como podemos ver en la gráfica de más arriba. Los chips x86 de AMD, que están un nodo por detrás y no tienen velocidades de reloj muy agresivas (prefieren el rendimiento multinúcleo como veremos en seguida) obtienen una puntuación de hasta 1284 puntos; por otro lado, el Apple M1 es fácilmente superado por los procesadores de Intel incluso aunque están dos nodos por detrás en cuanto a proceso de fabricación. (Alonso, 2020)

7. SELECCIÓN DE TÉCNICAS DE ANÁLISIS

Redes Neuronales

En este caso seleccionamos las redes neuronales, ya que, las redes neuronales están formadas por capas de nodos, que contienen una capa de nodos de entrada, una o varios nodos consecutivos al de entrada y finalmente tienen una capa de nodos de salida. Cada nodo, o neurona artificial, se conecta a otro y tiene un peso y un umbral asociados. Si la salida de un nodo individual está por encima del valor de umbral especificado, dicho nodo se activa y envía datos a la siguiente capa de la red. De lo contrario, no se pasan datos a la siguiente capa de la red. Esto permitiendo hacer una evaluación completa de las variables que vamos a emplear.

8. SELECCIÓN DE INSTRUMENTACIÓN

Big data y Apache Storm

Es un instrumento de análisis con gran potencial de procesamiento de datos sin distinción de su lenguaje programático. Procesando en tiempo real grandes cantidades de información de forma continua y constante en lapsos muy cortos de tiempo. Siendo útil en nuestro caso, este tipo de tecnología de Big Data se utiliza en plataformas de tiempo real, redes sociales cuya frecuencia de datos requieren analizarse en el momento que estén compartiendo al momento que estén sucediendo. Esta herramienta también nos permite procesar y detectar errores de los datos emitidos por motores de búsqueda y el comportamiento que tengan. Los usuarios en las redes sociales o en el uso de algún producto o servicio, arrojando información necesaria, eficaz y procesada instantáneamente.

REFERENCIAS

Martín del Brío, B., & Serrano Cinca, C. (2019). Fundamentos de redes neuronales artificiales: hardware y software. Scire: Representación Y organización Del Conocimiento, 1(1), 103–125. <https://doi.org/10.54886/scire.v1i1.1036>

Alonso, R. (2020, noviembre 18). Las CPUs Intel y AMD x86 Mobile baten al Apple M1 en Cinebench. HardZone. <https://hardzone.es/noticias/procesadores/intel-amd-cpu-x86-apple-m1-cinebench/>

Experimentos en ingeniería de software - Coggle Diagram. (s/f). Coggle.it. Recuperado el 21 de junio de 2022, de <https://coggle.it/diagram/YqdOlrdxQl8Lc0r/t/experimentos-en-ingenier%C3%ADa-de-software/480f7554f8826df791a4d10e65d8ada33cb8485f3dfbaeb30024496d7f46c826>

Fernández, Y. (2020, julio 7). Overclock: qué es, qué ventajas ofrece y qué desventajas puede tener. Xataka.com; Xataka. <https://www.xataka.com/basics/overclock-que-que-ventajas-ofrece-que-desventajas-puede-tener>

Anexo 6

Documentos generados sobre el tratamiento experimental

Código: A01

ESPE
UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA



DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN

CARRERA:

ING. DE SOFTWARE

NRC:

8506

ASIGNATURA:

INVESTIGACIÓN EN INGENIERÍA DE SOFTWARE

TAREA 11:

EXPERIMENTO Y REPRODUCIBILIDAD

NOMBRE:

██████████

FECHA:

05/07/2022

Código: A01

Índice

1. Desarrollo.....	3
1.1 Definición de Parámetros de Contexto.....	3
1.1.1 Identificación de sujetos.....	3
1.2 Selección de muestra	4
1.3 Selección del diseño experimental.....	4
1.3.1 Definición del proceso de recolección.....	4
1.3.2 Selección de técnicas de Análisis	5
1.3.3 Selección de Instrumentación.....	5
1.4 Empaquetado.....	6
Mejoras para aumentar la reproducibilidad.....	7
1. Control de Versiones.....	7
2. Gestión de Datos	7
3. Integración continua.....	7

Código: A01

1. Desarrollo

1.1 Definición de Parámetros de Contexto

Los desarrolladores de aplicaciones se analizarán mediante comparaciones de esfuerzo las cuales verán puestos a prueba elaborando un experimento de desempeño, tomando en cuenta el esfuerzo para desarrollar una aplicación entre el pair programming y el solo programming.

Se tomará en cuenta métricas como el tiempo de desarrollo, versionamiento, cantidad de código y velocidad ágil la cual avanza según su metodología.

Se llevará un calendario también para el control de las fechas de desarrollo y el tiempo empleado en cada una, para que después se pueda analizar con más facilidad el tiempo de desarrollo, en comparación con el repositorio Git Hub que también llevará el registro del tiempo empleado.

En base al contexto de experimento que se ha planteado tenemos dos parámetros a estudiar y a ser relacionados. Por un lado, tenemos nuestros objetos de experimentación que serán programadores puestos en cuestión y por otro lado el desempeño de los desarrolladores.

1.1.1 Identificación de sujetos

Los sujetos que se identificaron para este experimento basado en el contexto establecido son los desarrolladores senior de aplicaciones, para no tener inconvenientes se necesita experiencia en la práctica de pair programming o en el solo programming, ya que el desarrollador es el que se verá puesto en cuestión para calificar su esfuerzo en ambas situaciones y metodologías ágiles de desarrollo de Software.

1.1.2 Definición de hipótesis y variables

1.1.2.1 Hipótesis

Hipótesis Nula: El esfuerzo para desarrollar una aplicación no cambia, utilizando la practica de solo programming y pair programming.

Hipótesis Alternativa: El rendimiento, compatibilidad de un programa específico cambia si se ejecuta en los sistemas operativos MacOs y Windows.

Código: A01

1.1.2.2 Variables

Variable dependiente: Tiempo de desarrollo de la aplicación de los programadores estudiados.

Variable independiente: Tipo de práctica para el desarrollo de la aplicación, sea solo programming o pair programming.

1.2 Selección de muestra

La muestra en este experimento son los desarrolladores senior a los cuales se evaluará su esfuerzo principalmente en función al tiempo, como métricas secundarias a evaluar sería el versionamiento establecido en Git Hub, el cumplimiento del calendario establecido, calidad de código y la velocidad ágil de la aplicación desarrollada.

1.3 Selección del diseño experimental

Se seleccionó un diseño experimental determinístico a que el tiempo empleado en el desarrollo de software en pares o solo, podría ser cualquier tiempo. Lo que se pretende es medir el tiempo de desarrollo de los sujetos estudiados y las diferencias presentadas en cada una de las evaluaciones del experimento para así determinar con qué práctica se obtiene mejores resultados, cual es mas reproducible y

1.3.1 Definición del proceso de recolección

A continuación, se definen las actividades que nos ayudarán al proceso de recolección:

En el caso de pair programming y solo programming para la práctica del desarrollo de la aplicación se establece un tiempo de desarrollo el cual será registrado para la recolección de datos, también se establece un repositorio dentro de Git Hub para poder subir los documentos, tiempos de desarrollo y resultados obtenidos. La carpeta de creación del proyecto deberá llevar un .git que establecerá la dependencia del repositorio y el .README que nos servirá para saber como leer los datos y analizarlos correctamente.

Código: A01

Cuando: Cuando se desea conocer un análisis de cuál sistema operativo será el mejor para una empresa o para un uso personal.

Restricciones: Los dispositivos deberán contar con iguales características para así tener una comparación de datos con respecto al desempeño del desarrollador sin que se vea afectado por factores de componentes del computador.

Cómo: Desarrollar el software planteado con la práctica seleccionada (SP o PP) y todo tendrá registro en el repositorio de Git Hub especificado, conteniendo como dato principal el tiempo de desarrollo.

1.3.2 Selección de técnicas de Análisis

Para la selección de técnicas de análisis se ve involucrada una parte de estadística para comprender los resultados, en este caso los tiempos de desarrollo principalmente, en el mismo análisis también se tomará en cuenta la cantidad de código desarrollada, la velocidad ágil en la cual va avanzando en la metodología ágil, número de versiones desarrolladas y la cantidad de errores solucionados.

“Es una distribución de probabilidad que estima el valor de la media de una muestra pequeña extraída de una población que sigue una distribución normal y de la cual no conocemos su desviación típica” (Hook .S , 2015). Esta distribución estadística es la recomendada ya que funciona de gran manera en una muestra pequeña como es el caso, además de comparar únicamente dos valores.

1.3.3 Selección de Instrumentación

Se procederá a recolectar los datos obtenidos del repositorio Git Hub para sacar los tiempos de desarrollo diarios y la cantidad de versionamientos realizados del proyecto, para comparar la eficiencia de código entre dos personas o una sola. Para más facilidad de acceso al tiempo, se procederá a sacar un reporte del calendario establecido al inicio del proyecto de desarrollo en el cual podremos evaluar cuantitativamente los tiempos obtenidos de cada una de las pruebas del experimento. Se deberá hacer por lo menos una vez para poder empezar a tabular datos, se tomará en cuenta que cada uno de los programas desarrollados deberán estar funcionando de igual manera y siguiendo el mismo objetivo inicial de la aplicación desarrollada. Después de eso se podrá calificar varios aspectos de carácter cualitativo como las métricas en una hoja de cálculo

Código: A01

1.4 Empaquetado

Para empaquetar los resultados obtenidos se realizará un artículo científico de carácter experimental, con cada característica de la aplicación desarrollada, se procederá a extraer los datos recopilados en el repositorio Git Hub, tanto cada versión como los reportes de actividad del programador y las horas en las que se trabajó, también tendrá el calendario de trabajo establecido al inicio.

Se incluye los datos evaluados, la síntesis y análisis de datos desarrollados en el experimento y por último se procederá a confirmar si la hipótesis se comprueba y si los objetivos se cumplen.

Código: A01

Mejoras para aumentar la reproducibilidad

1. Control de Versiones

El uso de un gestor de control versiones ayuda y facilita en el proceso de un trabajo colectivo, tanto como la organización de un proyecto y todo su desarrollo. Se manejará un gestor de versiones de los mas usados en el mundo laboral, GitHub.

2. Gestión de Datos

La gestión de datos manejará repositorio de GitHub como gestor de control de versiones, y reportes de la aplicación al ser accesible y compatible para todos, para así poder obtener los datos y sea más facil el análisis cuando haga falta. Google Calendar para llevar el registro del calendario. Git Hub para determinar la velocidad ágil del proyecto.

3. Integración continua

GitHub se utilizaría para mantener una integración continua del proceso del desarrollo del producto software y el trabajo colaborativo de todos los miembros del equipo desarrollador.

Código: A02

NRC: 8506

Pair Programming vs Solo Programming

1. Definición

La programación por partes (Pair Programming) ha sido una práctica que ofrece muchos beneficios, pero también es comprendida y criticada desde el punto de vista operativo. Se desea plantear un experimento para comparar el esfuerzo de desarrollar una aplicación entre la práctica de la programación por pares (Pair Programming) en comparación con programación tradicional (Solo programming).

2. Diseño del experimento

Para el desarrollo de este experimento se deberá tomar en cuenta a ambos grupos, aquellos desarrolladores partitidos del "Pair Programming" y aquellos del "Solo Programming". Para establecer las variables que se tomarán en cuenta en nuestro análisis habrá que validar cada una de las variables específicas que afectarán al experimento, como pueden ser:

- Tiempo de desarrollo
- Extensión del proyecto
- Número de colaboradores
- Especialización en la tecnología utilizada para el proyecto

Teniendo en cuenta estas variables, el mejor experimento práctico que podemos llevar a cabo será una medición del tiempo de desarrollo utilizado por ambos grupos para concluir con un proyecto, ambos grupos utilizando los métodos propios de "Pair Programming" y "Solo Programming".

Código: A02

El proyecto planteado será realizar una plataforma Web, que conste de:

- Inicio de Sesión
- Portal/Dashboard
- Gestión de Usuarios

Lenguajes de programación, Frameworks, entre otros recursos para el desarrollo, serán de libre elección para los participantes.

También, se recomendará la configuración del entorno donde desarrollarán el trabajo, en este caso "Pair Programming" cuenta con Software especializado para realizar esta actividad, siendo el caso del IDE "Visual Studio 2019", como ambiente de desarrollo, que también cuenta con la funcionalidad "Live Share" que permitirá trabajar de manera colaborativa con distintos participantes, dentro de un mismo proyecto.

Así, ambos equipos manejarán un sistema de control de versiones (GIT) en la plataforma Github, para llevar una trazabilidad de los avances de ambos equipos a lo largo del desarrollo del proyecto.

Los equipos deberán realizar un reporte de cambios continuos, utilizando "commits" para marcar su progreso con el proyecto asignado.

De la misma forma, ambos equipos tendrán un tiempo límite de 2 horas, estimado para verificar el avance de ambos equipos durante la fase de desarrollo.

Con objetivo de análisis, al finalizar se llevará a cabo una revisión del desarrollo y gestión de ambos equipos con base a la norma ISO 12207.

3. Recolección

Para la recolección de datos se creará un repositorio público en la plataforma "Github", se configurará una "Rama" independiente para cada uno, también, se usará la matriz de

Código: A02

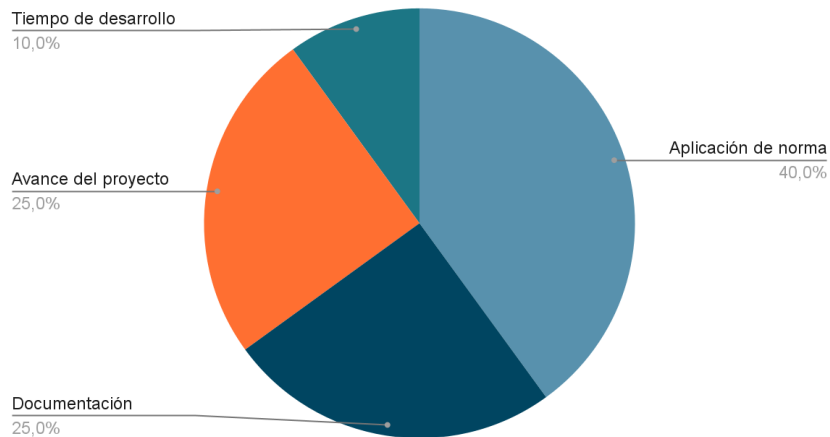
puntuación evaluando el desarrollo con base a la norma ISO 12207. Se solicitará a los participantes evidencien las metodologías que aplicaron individualmente y la eficacia a la hora de priorizar actividades.

4. Análisis

Para llevar a cabo el análisis de los resultados tendremos en cuenta no solo el tiempo de desarrollo de ambos equipos, sino también el cumplimiento con los estándares de calidad que todo Software debe ofrecer al consumidor, para esto, se tomará en cuenta los parámetros que nos establece la norma ISO 12207 para calificar el ciclo de vida que cumplió el Software desarrollado, así como el cumplimiento con el proyecto en el plazo establecido.

El siguiente gráfico demostrará la matriz de evaluación que será utilizada.

Valoración de Resultados



Código: A02

5. Empaquetado

Los resultados de la investigación estarán disponibles públicamente en su repositorio en Github, a su vez, utilizaremos una carpeta en la nube de "OneDrive" para recopilar los informes detallando la metodología empleada por los equipos, observaciones y, anexo, los resultados del análisis de la investigación.

Código: A03 Universidad de las Fuerzas Armadas ESPE

Nombre: [REDACTED]

NRC: 8506

Planificación de Experimento

Actividad Plantear un experimento para comparar el esfuerzo de desarrollar una aplicación entre la práctica de la programación por pares en comparación con programación tradicional

INDICE

Contenido

Introducción	2
Definición	2
Planteamiento del Problema	2
Definición Contexto	2
Definición Objetivos	2
Definición general del problema	3
Definición de parámetros de contexto	3
Identificación de sujetos	3
Definición de hipótesis y variables	3
Diseño	3
Selección de muestra	3
Selección del diseño experimental	3
Definición del proceso de recolección	3
Selección de técnicas de análisis	5
Selección de instrumentación	5
Recolección	5
Autorización de Recolección	6
Validaciones de Datos	6
Problemas reportados.....	6
Análisis	6
Descripción de datos	6
Reducción de conjunto de datos	6
Comprobación de Hipótesis	6
Empaquetado.....	6
Identificación de amenazas e Interferencias	6
Interpretación de resultados.....	7
Identificación de Lecciones aprendidas.....	7
Reporte de Resultados	7
Bibliografía	7

Comparación del esfuerzo generado entre la programación tradicional y programación por pares.

Introducción

Diferentes técnicas de desarrollo de software han surgido en los últimos años, estas buscan mejorar el desarrollo mediante calidad o procedimiento, generando alternativas a la programación tradicional.

Entre estas, se encuentra la técnica ágil programación por pares (Pair Programming) que es un tipo de programación donde 2 desarrolladores programan por turnos en un tipo de

En la actualidad, la programación por pares es altamente reconocida en términos generales, principalmente por la estabilidad que crea para proyectos de software y su desarrollo.

Por ello, a pesar del éxito que ha proporcionado, es necesario evaluar el efecto consecuente en los desarrolladores y programadores que utilizan esta técnica, principalmente en comparación con la programación tradicional, para reconocer si existirán problemas graves en los desarrolladores y comprender si es una técnica que proporciona utilidad a una empresa.

Definición

Planteamiento del Problema

Definición Contexto

El panorama general se ha enfocado en como afecta a la empresa o cliente el uso de diferentes tipos de técnicas de desarrollo de software, que circunstancias generan o que efectos tienen en sus productos y tiempos. Pero además de ello, es necesario conocer que ocurre en los desarrolladores al utilizar diferentes técnicas, principalmente para conocer si existen posibles problemas que ocurra al utilizar estas técnicas para el cliente, y reconocer si estas técnicas son factibles y recomendadas para los programadores.

El experimento se desea llevar a diferentes tipos de técnicas de desarrollo de software, enfocándose en los desarrolladores y cual es el efecto que proporciona utilizar estas técnicas. Por ello el contexto planteado será definido en el efecto producido al utilizar las técnicas de desarrollo de software en un grupo de desarrolladores.

Definición Objetivos

- Reconocer cuantitativamente el esfuerzo necesario por los desarrolladores al utilizar las técnicas de desarrollo de software tradicional y de programación por pares de manera que sea reconocible la técnica que produce mayores inconvenientes a los desarrolladores de software.
- Identificar los puntos el esfuerzo producido al utilizar las técnicas de desarrollo de software de programación por pares y tradicional, para fragmentar el esfuerzo y

Código: A03 Universidad de las Fuerzas Armadas ESPE

comparar con mayor claridad el tipo de efectos que reproduce el uso de estas técnicas en los desarrolladores.

Definición general del problema

Los efectos producidos por un esfuerzo sobreestimado en los programadores, así como su rendimiento son perjudiciales para sus clientes, provocando fallas que pueden producir a proyectos de software fallidos. Por ello, es necesario conocer que ocurre en los desarrolladores de software al utilizar las técnicas de desarrollo de software, con análisis en la programación tradicional con la técnica ágil programación por pares. Esto principalmente para reconocer que tipo de técnica produce un mayor esfuerzo en los programadores, que puede producir efectos adversos en su eficiencia y trabajo productivo, así como actitud general y rendimiento.

Definición de parámetros de contexto

Identificación de sujetos

Los sujetos a evaluar son:

Los desarrolladores activos de software, que utilizan los dos tipos de técnicas de desarrollo de software: Programación tradicional, Programación por pares.

Definición de hipótesis y variables

Ho: El 75% de los desarrolladores de software no produce un esfuerzo significativo al utilizar la técnica de desarrollo por pares, en comparativa con la técnica de desarrollo tradicional

Hi: El 75% de los desarrolladores de software produce un esfuerzo significativo al utilizar la técnica de desarrollo por pares, en comparativa con la técnica de desarrollo tradicional

Variable Independiente: Desarrolladores de Software

Variable dependiente: Esfuerzo Realizado

Diseño

Selección de muestra

La muestra utilizada, será un grupo aleatorio equivalente entre programadores que utilicen programación por pares, y programación tradicional de 50 programadores. Se dividirán 25 programadores de programación pro pares y 25 programadores de programación tradicional.

El grupo debe ser de desarrolladores experimentados con proyectos que tengan una misma planificación en su duración y estén enfocados a un área similar en este caso se tomará en cuenta 4 meses de trabajo en proyectos de mediana calidad de desarrollo de aplicaciones web. De esta manera se evitan problemas externos que sean derivables a la cuantificación del esfuerzo.

Selección del diseño experimental

Para la selección del diseño experimental hay que tomar en cuenta diferentes parámetros

Definición del proceso de recolección

En el proceso de recolección, se tomará en cuenta diferentes contextos:

Código: A03 Universidad de las Fuerzas Armadas ESPE

Actividades

Recolección de evaluación externa

La recolección de datos externos que muestre el desarrollador al producir software se realizará mediante el uso de un grupo de control, que maneje una matriz de evaluación con diferentes puntos mostrando reacciones comunes a un aumento de esfuerzo o sobreesfuerzo en el desarrollo de software.

Para ello se plantea que, en el periodo de 4 meses, semanalmente se plantee una visita del grupo de control a los programadores seleccionados, este grupo revisará el comportamiento de los desarrolladores y evaluará que consecuencias externas se demuestran en su esfuerzo utilizando una matriz con parámetros cuantificables.

Se plantean parámetros, con puntos internos a definir para la evaluación de datos. Los diferentes parámetros a calificar que se plantean son:

- Efectos físicos de aumento de esfuerzo general (Pérdida de cabello, Movimientos erráticos en extremidades, Tics nerviosos, Efectos de ansiedad)
- Efectos cognitivos de aumento de esfuerzo general. (Pérdida de memoria, problemas de socialización, falta de sueño, otros)
- Efectos emocionales de aumento de esfuerzo (Ataques de emociones, ira, enojo, tristeza, otros)
- Efectos de producción (manejo simple del producto, falta de motivación)

Recolección de evaluación interna

Para la recolección interna, se plantea el uso de encuestas de satisfacción personales para los programadores y desarrolladores. Para ello, semanalmente los programadores deberán llenar una encuesta, que muestre diferentes parámetros de evaluación interna, principalmente se plantean:

- Estado de ánimo
- Evaluación general de esfuerzo
- Problemas personales
- Errores grupales

Se recolectará la información para comparar los resultados obtenidos entre las diferentes técnicas.

Recolección de Producto Finalizado

Finalmente, se plantea una recolección del producto finalizado al utilizar las diferentes técnicas de desarrollo de software, para luego su comparativa, esto principalmente para comprobar si el esfuerzo realizado creó errores en su producción.

Para la evaluación se utilizará técnicas de evaluación de software general, en este caso, para el desarrollo de software de web, se planteará una revisión de código presentado, el uso de test para el producto, así como el control y calidad general.

Código: A03 Universidad de las Fuerzas Armadas ESPE

Calendarización

Se planificará en un horario calendarizado de 4 meses, donde se pueda distinguir con mayor claridad los patrones y efectos que produce el esfuerzo en los desarrolladores, además de recolecciones semanales de información por parte de un grupo de control de 6 a 8 personas.

Restricciones

El conocimiento del experimento debe afectar en lo mínimo a los desarrolladores, por ello se evitará el reporte de resultados, y la interacción entre el grupo de control y los desarrolladores externamente.

Además, se buscará que no existan filtraciones de resultados, y que se mantenga un tipo de desarrollo constante y con horarios, para que no existan riesgos externos.

Controles de Calidad

Para mayor calidad de recolección de datos, se plantea el uso de herramientas tecnológicas para el guardado de datos, así como el uso de tecnologías como Excel, para mayor facilidad de tipeo y recolección.

Para un aumento de calidad, también debe existir facilidad de obtención de datos, por lo que se utilizara Google Forms para las encuestas, para que los datos sean directamente manejados por variables claras y no exista manipulación de información.

Selección de técnicas de análisis

Para el análisis del desarrollo y resultados del experimento, se planificará una matriz de recolección de datos.

Esta matriz tendrá los parámetros ya mostrados con una escala de evaluación del 1 al 10, donde 1 es el menor y 10 el mayor.

En el desarrollo de la encuesta, se tomará puntos de evaluación del 1 al 5, para comprender los resultados se buscará que 1 sea menor y 5 mayor.

En el desarrollo del esquema comparativo, se revisarán el resultado producido por los desarrolladores en sus productos software, mediante el uso de parámetros que evalúen el cumplimiento de enfoques de calidad, como limpieza en el código, versionamiento, uso de herramientas externas, entre otros.

Selección de instrumentación

Para ello, se realizará grupos de control que tengan que revisar a los desarrolladores, encuestas de satisfacciones semanales de los desarrolladores y un cuadro comparativo del software obtenido.

Los datos se verán guardados en un repositorio privado que mantenga la información guardada con sus respectivas fechas y tiempos de realización.

Recolección

En el proceso de recolección, es necesario conocer diferentes puntos que permitan mayor validez y estándar en su proceso. Por ello para la planificación es necesario conocer diferentes características

Código: A03 Universidad de las Fuerzas Armadas ESPE

Autorización de Recolección

Para la estandarización, se proporcionará un escrito firmado, que valide que se obtendrá información constante de los desarrolladores y su comportamiento para la evaluación y reconocimiento de efectos del esfuerzo.

Además, se pedirá autorizaciones a las empresas o clientes que desarrollen el software, sobre el conocimiento de los desarrolladores general del producto software, así como posibles interrupciones calendarizadas por grupos de control.

Validaciones de Datos

Para la validación de datos, se utilizará rotaciones en los grupos de control, que permitan que los datos no se vean manipulados.

Se validará la información obtenida semanalmente de los desarrolladores, mediante su firma y certificación de su presencia en el desarrollo del producto software utilizando la técnica de desarrollo.

Problemas reportados

Para el control de problemas, se tendrá en cuenta diferentes problemáticas posibles en la recolección de datos. Principalmente la ausencia de desarrolladores o su falta de interés por la experimentación será registrada en un documento semanal de problemas surgidos.

En el documento, también se encontrarán posibles problemas del experimento tales como problemas en el producto, en la empresa, entre otros con la posibilidad de añadir más opciones.

Finalmente, se tomará en cuenta estos problemas para el desarrollo de riesgos en el análisis de los resultados.

Análisis

Descripción de datos

Los datos obtenidos en los 4 meses de realización del experimento serán debidamente transcritos cada semana, y recopilados mediante el uso de versionamiento.

Se plantea el uso de técnicas de guardado de datos como Obsidian y para el mantenimiento de la información se plantea el uso de GitHub en un repositorio privado para el experimento.

Reducción de conjunto de datos

Comprobación de Hipótesis

Se comprobará la hipótesis obtenida, esto mediante el uso de técnicas de análisis de datos. En este caso se utilizará un cuadro comparativo estructural, con los

Empaquetado

Identificación de amenazas e Interferencias

Luego de obtener el análisis claro de los resultados, se planteará un documento que identifique que tipo de amenazas o interferencias generales tuvo el experimento, mostrando que ocurrió en cada versionamiento de la semana, así como planteamiento de como evitar las amenazas e interferencias obtenidas.

Código: A03 Universidad de las Fuerzas Armadas ESPE

Interpretación de resultados

Para la interpretación de resultados, se utilizará tecnologías de análisis de datos, tales como lenguaje R y Matlab, para reconocer los resultados obtenidos y crear gráficos y estructuras más comprensibles para futuros investigadores.

Identificación de Lecciones aprendidas

Mediante el desarrollo interno, se puede producir conclusiones de que información se ha obtenido en general para el desarrollo experimental, con una clara estructura que aporte a futuras investigaciones en este tipo de temáticas.

Reporte de Resultados

Se plantea el desarrollo de un documento de Latex, que será producido en idioma inglés y español, que será divulgado en una revista científica, principalmente Jedisktra, utilizando su formato y esquema estructurado de reporte.

Además, el desarrollo de los resultados será previamente validados y verificados por un grupo externo del experimento, a fin de conocer posibles errores de redacción y manejo. Y comprobar si los resultados obtenidos son validados.

Bibliografía

Se referenciarán todos los datos necesarios y recolectado de la planificación. Se utilizara herramientas como Mendelev para el referenciado y se mantendrá una estructura de referencias tipo APA.

Código: A04



**“UNIVERSIDAD DE LAS FUERZAS
ARMADAS ESPE “**

**INVESTIGACIÓN EN LA INGENIERÍA DE
SOFTWARE**

Nombre: [REDACTED]

NRC: 8506

Fecha: 05/07/2022

Planificación de un experimento para comparar el esfuerzo del desarrollo de una aplicación desempeñando pair programming VS solo programming.

Código: A04

TABLA DE CONTENIDO:

Planteamiento del Problema:	3
Objetivos:	3
Muestra:	4
Sujetos:.....	4
Objetos/Artefactos:.....	4
Técnica de muestreo:	4
Hipótesis	5
Análisis de Variables:	5
Identificación de variables.....	5
Instrumentos de Recolección y Análisis:	5
CRONOGRAMA:	6
Empaquetado de Resultados:	6

Código: A04

Planteamiento del Problema:

El siguiente experimento requiere hacer una comparación del esfuerzo que toma a los desarrolladores crear una aplicación de alta calidad mediante un enfoque de desarrollo basado en pair programming comparado con el solo programming

Si se analiza en las últimas décadas del desarrollo de software, la programación ha sido enseñada y podría decirse que diseñada para que una sola persona la realice con ayuda del computador, esto se refiere a que un solo programador es capaz de crear sistemas y software de diferentes tipos; la cuestión en esta premisa es, ¿qué tan bueno sería este software?, será un código entendible, cumplirá con estándares de calidad u orden, todas las mencionadas son preguntas que deben plantearse para conocer si esta práctica está dando buenos resultados o no.

Objetivos:

General

Comparar las ventajas y desventajas de aplicar pair programming en el desarrollo de una aplicación, mediante la ejecución de un experimento que permita el esfuerzo realizado por los sujetos con este mecanismo en el desarrollo.

Específicos:

- Aplicar pair programming en un grupo de desarrolladores para medir la calidad de código y desarrollo.
- Identificar los impactos positivos o negativos de la pair programming y la solo programming, a partir de una revisión de datos arrojados por el experimento.
- Medir el esfuerzo del programador a partir del mecanismo pair programming frente al solo programming.

Para empezar con la ejecución de este experimento primero se debe identificar a los posibles sujetos e instrumentos que se deben incluir en el mismo; para ello se debe hacer un estudio de la muestra y lo que la conforma.

En este caso la muestra debe ser tomado de un conjunto de sujetos que tengan un nivel medio/alto de conocimiento de programación, que ya hayan tenido experiencia programando y creando sistemas antes.

El escenario debe ser tomado de una empresa de desarrollo, que posea cierta cantidad de empleados preparados y que manejen herramientas TI

Código: A04

Muestra:

Equipo de desarrolladores de 2 a 3 años de experiencia en el mundo de la programación, deben poseer conocimiento en diversos lenguajes de programación y haber realizado otros proyectos en el pasado.

Sujetos:

A continuación, se detalla los sujetos implicados para la ejecución del experimento:

- Desarrollador
- Investigador

Objetos/Artefactos:

- Computadora (última generación, procesador de 8 núcleos y 16GB RAM)
- IDE, framework, repositorio (para que se vaya guardando el proyecto)

Técnica de muestreo:

Para poder proceder con la elección de los candidatos, se debe aplicar un muestreo aleatorio dentro del mismo equipo de programadores.

Se tiene que escoger 2 grupos como se describe a continuación:

1er Grupo: Pair Programming

3 Parejas de programadores, ubicados en una misma computadora con un mismo proyecto.

2do Grupo: Solo Programming

3 Desarrolladores, ubicadas cada uno en una computadora individual, con un mismo proyecto.

Cabe recalcar que para todos los grupos el proyecto será el mismo, el tiempo de desarrollo estimado será el mismo de igual manera.

Código: A04

Hipótesis

El uso de la técnica pair programming disminuye el esfuerzo de los programadores en el desarrollo de una aplicación, así como mejora la calidad del producto software.

Análisis de Variables:

Identificación de variables

En el siguiente experimento van a intervenir algunas variables que aportarán con la medición del esfuerzo empleado por el programador en el proceso de desarrollo de la aplicación.

- **Tiempo:** la siguiente variable será medida en días y horas en las que el proyecto este terminado,
- **Errores de Código:** Se hará un análisis e inspección de código limpio para medir errores de los programadores.
- **Emociones:** Se estará observando los sentimientos y actitudes (frustración, felicidad, estrés, tranquilidad), que vayan presentando los programadores durante el proceso de desarrollo.

Se debe enfatizar que lo que se va a estudiar es el esfuerzo de los sujetos en el desarrollo, antes que la funcionalidad al 100% de la aplicación, sin embargo igualmente se debe integrar técnicas de análisis para verificar la calidad del producto software final.

Instrumentos de Recolección y Análisis:

Una vez se haya realizado el experimento se debe poseer y aplicar técnicas de análisis de datos dependiendo de las variables dadas.

A continuación, se detalla lo necesario:

- En este caso todos los datos recogidos deben ser almacenados en un repositorio GITHUB, que facilite la revisión de los mismos.
- El tiempo empleado para el desarrollo, será analizado mediante el CRONOGRAMA establecido.
- Los participantes harán preguntas o reclamos a su jefe de proyecto y responderán ENCUESTAS de cómo se sintieron cada vez que tengan una sesión de trabajo.

Cabe recalcar que para cada grupo se hará un carpeta de información relevante, donde constarán los datos necesarios para medir sus capacidades y progresos.

Así mismo se establece un cronograma de como se estaría efectuando la aplicación y análisis del experimento.

Código: A04

CRONOGRAMA:

ACTIVIDAD	FECHA	Actividad	Fecha
Selección Muestra	Enero 2023	Análisis y control del proceso de desarrollo por grupos	1 Marzo 2023
Adquisición instrumentos (computadoras, programas)	Enero 2023	Recolección de datos generales	Marzo 2023
División y Creación de Repositorios por grupo en GITHUB	Febrero 2023	Recolección de comportamientos de los sujetos en la etapa de desarrollo	Marzo 2023
Inicio de Experimento	20 Febrero 2023	Finalización del proyecto	1 Abril 2023
1er Push del proyecto en GitHub (README)	1 Marzo 2023	Inspección y análisis de resultados	Abril 2023

Empaquetado de Resultados:

Los resultados de calidad, tiempo e impactos positivos y negativos, basado en las variables, se darán a conocer mediante el repositorio de GitHub de cada grupo, así como también se hará una presentación e informe de los datos obtenidos para cada grupo y variable, el repositorio será público para que otros investigadores puedan acceder a los resultados, tanto como a los proyectos de software realizados.

No obstante, se debe tomar en cuenta los riesgos en la interpretación de los resultados, como es el caso del grupo 1 donde se tendrá parejas de programadores, ya que en este caso un desarrollador podría haber aportado más que otro, y podría haber recaído en un solo programming, por ende se debe detallar bien las reglas aplicadas en este experimento.

Código: A05

PROGRAMACION EN PARES VS PROGRAMACION CONVENCIONAL

Planificación de experimentación

Código: A05

Índice

Índice.....	2
1. Planteamiento del problema.....	3
1.1. Definición de contexto	3
1.2. Objetivos	3
1.2.1. Objetivo general	3
1.2.2. Objetivos específicos	3
2. Definición de parámetros de contexto	3
2.1. Identificación de los sujetos	3
2.2. Hipótesis.....	3
2.3. Selección de la muestra.....	4
2.4. Selección del diseño experimental	4
3. Recolección de datos.....	4
3.1. Autorización de recolección de datos.....	4
3.2. Validación de datos.....	4
3.3. Reporte de problemas.....	4
4. Análisis de datos.....	4
4.1 Descripción de los datos obtenidos.....	4
4.2 Pruebas de hipótesis	4
5. Empaquetado de resultados	5

Código: A05

1. Planteamiento del problema

1.1. Definición de contexto

La programación por pares es una práctica muy criticada, que a la vez ofrece diferentes beneficios, sin embargo, existen oposiciones debido a que consigo trae diferentes dudas, principalmente siendo, ¿es lo más óptimo a comparación de la programación habitual?, ya que en esta práctica se ocupan dos personas para realizar un mismo trabajo en conjunto, para las empresas o contratistas esto puede significar una baja en la eficiencia de sus recursos y procesos de desarrollo.

1.2. Objetivos

1.2.1. Objetivo general

Determinar que metodología es mejor entre el pair programming y la programación habitual analizando las diferencias mediante un experimento comparativo entre ventajas y desventajas de cada metodología para determinar cuál es la que se adapta de mejor manera a las necesidades de las diferentes empresas y Contratistas.

1.2.2. Objetivos específicos

- Determinar las ventajas que tiene el pair programming en comparativa a la programación habitual
- Comparar las desventajas entre las metodologías tomando en cuenta diferentes parámetros (tiempo, errores, recursos).

2. Definición de parámetros de contexto

2.1. Identificación de los sujetos

Dos programadores certificados en Java.

2.2. Hipótesis

El pair programming presenta más ventajas que desventajas a comparación de la programación habitual adaptándose de mejor manera a la solvencia de problemas de las empresas/clientes

Ho: El pair programming presenta más desventajas que ventajas a comparación de la programación habitual adaptándose de mejor manera a la solvencia de problemas de las empresas/clientes

Variable independiente: El pair programming presenta más ventajas que desventajas a comparación de la programación habitual

Variable dependiente: adaptándose de mejor manera a la solvencia de problemas de las empresas/clientes

Código: A05

2.3. Selección de la muestra

Se requiere a dos desarrolladores certificados en java con dos computadores.

2.4. Selección del diseño experimental

El diseño experimental se basará en un grupo único seleccionado, donde se tendrá que desarrollar un mismo proyecto que deberá ser de una robustez media, desarrollado en java dos veces aplicando cada metodología con los mismos programadores y equipos, para obtener finalmente dos proyectos finales, pudiendo comparar entre ellos los resultados de cada metodología, sin embargo, también es necesario compararlas en el desarrollo aplicando una matriz comparativa que nos ayude a cuantificar las características.

El instrumento para ocupar es una matriz evaluativa comparativa donde los parámetros dominantes son tiempo de desarrollo, errores finales, errores en desarrollo, costos, calidad del proyecto final, mantenibilidad.

3. Recolección de datos

3.1. Autorización de recolección de datos.

Los desarrolladores deben dar su consentimiento escrito para poder ocupar los datos profesionales y de los proyectos que realizaron

3.2. Validación de datos.

Se realizará una recolección de datos desde diferentes puntos de vista para que haya mas veracidad y evitar posibles alteraciones de datos provocados por alguna mala practica

3.3. Reporte de problemas.

Todos los problemas deberán ser reportados con especificaciones para evitar una alteración de los resultados.

4. Análisis de datos

4.1 Descripción de los datos obtenidos

Los datos recolectados deben estar estrictamente detallados basándose en la matriz comparativa para poder realizar un análisis de una manera más eficiente añadido deberá tener especificaciones de recolección tales como hora y fecha, y observaciones que pudo tener el recolector que aporte al análisis.

4.2 Pruebas de hipótesis

Se analiza finalmente la hipótesis basada en el valor cuantitativo obtenido de la matriz comparativa, entre la programación en parejas y la convencional, determinando si se acepta o rechaza la hipótesis

Código: A05

5. Empaquetado de resultados

Los dos proyectos finales serán subidos a un repositorio publico en la nube que sea de dominio público, los informes conjunto a la matriz comparativa se subirán a Google drive abierto al acceso privado compartido, donde constara el link al repositorio donde se encuentran ubicados los proyectos para que puedan corroborar los datos recolectados, añadido a esto en los informes constara la información de contacto de los elaboradores del proyecto por si surge una duda del experimento o de los datos recolectados para poder comunicarse con la persona en específico ya que los documentos compartidos constan con un historial de edición de tal manera que podemos ver los cambios que ha realizado cualquiera de las personas finalmente el link de los informes serán distribuidos a los interesados.

Código: A11

Tabla de contenido

Introducción	2
Objetivo	3
Definición de parámetros de contexto	3
Contexto del problema	3
Método de muestreo	3
Identificación de sujetos	3
Definición de hipótesis y variables	3
Selección de muestra	4
Selección del diseño experimental	4
Definición del proceso de recolección	4
Selección de técnicas de análisis	4
Métricas de evaluación	3
Informe de resultados	3
Selección de instrumentación	4
Conclusiones	4
Referencias	5

Código: A11

Comparación del esfuerzo de desarrollo de una aplicación entre la práctica de la programación por pares (Pair programming) en comparación con programación tradicional (Solo programming)

Universidad de las Fuerzas Armadas ESPE
Av. General Rumiñahui s/n, 171-52318
Sangolquí, Ecuador

Resumen. - Este experimento se desarrolla debido al interés general que se tiene sobre los orígenes de la programación por pares y la programación tradicional; además de que al conocer más sobre estos tipos de programación nos ayudara a comprender la comparación de esfuerzo en base al desarrollo de una aplicación.

Este experimento surge frente a la necesidad de estudiar todo lo relacionado en la programación por pares y tradicional, así que se buscara la obtención de una extensa información a través de información obtenida por medio de búsquedas en diferentes sitios web.

Abstract. - This experiment is developed due to the general interest in the origins of pair programming and traditional programming; In addition, knowing more about these types of programming will help us understand the comparison of these based on the development of an application.

This experiment arises from the need to study everything related to traditional and pair programming, so it will seek to obtain extensive information through information obtained through searches on different websites.

1. INTRODUCCIÓN

El desarrollo de software es una actividad que debe realizarse con el apoyo de algún enfoque metodológico que facilite la organización de tareas, el trabajo colaborativo y el entendimiento de los requerimientos de las aplicaciones. No existe un enfoque que sea universalmente reconocido como el mejor para todos los escenarios, por lo que las diversas formas de trabajo deben evaluarse para elegir la más indicada de acuerdo con las necesidades de cada proyecto y cada equipo de personas.

Para poder empezar con este experimento tenemos que conocer el ¿Qué? ¿Cómo? de estos tipos de programación y así comparar al final que tipo de programación requiere más esfuerzo en base al desarrollo de una aplicación.

Primero empezamos por la programación por pares, es una práctica que consiste en que dos programadores juntan sus conocimientos y esfuerzos para trabajar en un mismo sitio. El objetivo es que uno de los programadores pueda realizar una tarea que el otro programador no está realizando durante ese momento. En este caso, la persona que está haciendo la programación recibe el nombre de “controlador”, mientras que la persona que lo dirige recibe el nombre de “navegador”.

Código: A11

La programación tradicional es un proceso manual, es decir, una persona o un programador crea el programa. Pero sin que nadie programe la lógica, uno tiene que formular manualmente o codificar reglas.

2. OBJETIVO

Comparar el esfuerzo de desarrollo de una aplicación que se realiza en cada una de las programaciones, en este caso programación por pares (Pair programming) y programación tradicional (Solo Programming).

3. DEFINICIÓN DE PARAMETROS DE CONTEXTO

3.1. Contexto del problema

A cuestión general se necesita saber qué tipo de programación es más esencial al momento de crear un software y el esfuerzo que se va a requerir para la elaboración de este, es por eso que se debe realizar la comparación entre los dos tipos de programación y al final llegar a entender el ¿Por qué? de que el lenguaje seleccionado es el mejor.

3.2. Método de muestreo

Para el siguiente experimento el mejor método de muestreo es el método aleatorio para el muestreo.

4. IDENTIFICACIÓN DE SUJETOS

Para este experimento se selecciona como sujetos a los desarrolladores de software, los mismos que pueden ser junior, semi senior y senior.

5. DEFINICIÓN DE HIPÓTESIS Y VARIABLES

5.1. Hipótesis Nula

Tiempo de trabajo

Los dos tipos de programación pueden parecer rápidos y ágiles de aplicar, la diferencia está en el tiempo, en programación por pares al trabajar dos programadores cada uno desempeña un rol y en cierto periodo de tiempo cambian de roles y pueden ir corrigiendo errores, acortando el tiempo de que al final salga algún error en la aplicación; en programación tradicional trabaja una sola persona haciendo el cumplimiento de todos los roles aplicando mas tiempo dado que al final puede haber uno o varios errores al final de la aplicación.

5.2. Hipótesis Alternativa

- H0: mientras el driver trabaja implementando una parte de la solución, el navegador puede trabajar en una investigación con el objetivo de mejorar la solución que se está desarrollando.
- H1: mientras el driver no trabaja implementando una parte de la solución, el navegador no puede trabajar en una investigación con el objetivo de mejorar la solución que se está desarrollando.

Código: A11

Estas hipótesis son planteadas a base de la hipótesis nula, es decir, son respuestas alternativas a la hipótesis nula que pretenden demostrar su falsedad. Este tipo de hipótesis se acepta o no se acepta.

6. SELECCIÓN DE MUESTRA

En la muestra tenemos un aplicativo puede ser sobre cualquier tema en específico, suponiendo en este caso una aplicación de suma de matrices, donde se pide programar por los dos tipos de programación por pares y tradicional.

7. SELECCIÓN DEL DISEÑO EXPERIMENTAL

Para este experimento como diseño experimental tenemos los diferentes aspectos que van a estar en el experimento, en este caso podemos usar un informe o artículo donde se englobe todos los resultados del problema planteado en un inicio.

8. DEFINICIÓN DEL PROCESO DE RECOLECCIÓN

En base a los tipos de programación tenemos diversos parámetros que podemos aplicar para selección de técnicas de análisis haciendo el trabajo de comparación de esfuerzo mas sencillo.

9. SELECCIÓN DE TÉCNICAS DE ANÁLISIS

9.1. Métricas de evaluación

Para la selección de instrumentación lo podemos hacer en base una métrica de evaluación de ambos tipos de programación, donde cada parámetro de evaluación está conformado según el tipo de programación.

9.2. Informe de resultados

Al final de la evaluación por medio de un informe se obtienen los resultados de cual tipo de programación es mejor porque se requiere menos esfuerzo al momento de desarrollar una aplicación, cabe recalcar que los resultados son en base a números y observaciones de cada parámetro.

10. SELECCIÓN DE INSTRUMENTACIÓN

Como instrumentación tenemos al aplicativo que se menciono anterior mente y en base a este se desarrolla todo el experimento y los resultados que se vayan a obtener.

11. CONCLUSIONES

- Se puede concluir que de cierto modo ninguno de los tipos de programación es el ideal, dado que cada uno tiene sus pros y contra al momento de desarrollar una aplicación, solo varían en pequeños detalles haciendo que programación tradicional (Solo programming) se vea como la mas efectiva al momento de desarrollar un aplicativo por la parte esencial que es la optimización del tiempo de trabajo haciendo que se requiera menos esfuerzo.

Código: A11

Referencias

Assembler Institute. (2022, 1 junio). *Pair programming: ¿en qué consiste?* Recuperado 5 de julio de 2022, de <https://assemblerinstitute.com/blog/que-es-el-pair-programming/>

Gonzalez, L. (2020, 19 agosto). *Diferencia entre Machine Learning y la Programación*

Tradicional. Aprende IA. Recuperado 5 de julio de 2022, de

<https://aprendeia.com/diferencia-entre-machine-learning-y-la-programacion-tradicional/#:~:text=La%20programaci%C3%B3n%20tradicional%20es%20un,formular%20manualmente%20o%20codificar%20reglas.>

Slattery, T. N. (2020, 19 mayo). *Pros y contras de la programación en pares para la*

automatización de red. ComputerWeekly.es. Recuperado 5 de julio de 2022, de

<https://www.computerweekly.com/es/consejo/Pros-y-contras-de-la-programacion-en-pares-para-la-automatizacion-de-red>

Código: A12



DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN

CARRERA

Ingeniería de Software

INGENIERO

Ing. Efrain Rodrigo Fonseca Carrera

INTEGRANTE

[REDACTED]

MATERIA

Investigación en Ingeniería de Software

FECHA

2022/07/05

SEMESTRE Y NRC

S-I MAYO – SEPT 22 – 4322

INDICE

1.1 DEFINICIÓN DE REQUERIMIENTOS

1.2 PLANIFICACIÓN DISEÑO

1.3 RECOLECCIÓN DE DATOS OPERACIÓN Y ANALISIS DE DATOS E INTERPRETACIÓN

Código: A12

1.3 EMPAQUETADO DE RESULTADOS

1.1 DEFINICIÓN DE REQUERIMIENTOS

1.1.1 PROBLEMA Y CONTEXTO

comparar el esfuerzo de desarrollar una aplicación entre la práctica de la programación por pares (Pair programming) en comparación con programación tradicional (Solo programming).

1.1.2 DEFINICIÓN DE VARIABLES

PAIR PROGRAMING

Es una técnica de desarrollo de software ágil, en la que dos desarrolladores se unen en una sola computadora. Las dos personas trabajan juntas para diseñar, codificar y probar historias de usuarios donde es el navegador el otro el conductor.

SOLO PROGRAMING

Es la técnica más común ya que consiste en que solo existe un desarrollador que genera el código y supervisa el mismo

1.1.3 DEFINICIÓN ACTORES E IMPACTO

NAVEGADOR: Aquel que revisa la pantalla y monitorea al conductor evaluando su código

CONDUCTOR: Aquel que codifica en el ordenador y van realizando el programa

1.1.4 HIPOTESIS

H: La programación por partes demuestra ser más exitosa que la programación sola ya que en sí genera calidad del software, aunque no es útil para ciertos proyectos con plazos de tiempo cortos

H1: La programación en pares genera mejor calidad de código

H2: La programación en pares es costosa y tardía, no es útil para trabajos con poco plazo de tiempo

H3: La programación sola es una técnica ampliamente conocida y usada genera más seguridad

1.2 PLANIFICACIÓN DISEÑO

1.2.1 Variables de estudio

En este caso estudiaremos varios aspectos generalmente usados como es la productividad, tiempo del proyecto, calidad y otros aspectos entre ambos aspectos

Código: A12

PRODUCTIVIDAD: automatización de ciertas tareas repetitivas, lo cual permite optimizar el desarrollo de algunos procesos

CALIDAD: todo el conjunto de cualidades que lo caracterizan determinando su eficiencia y utilidad, satisfaciendo las necesidades tanto implícitas como explícitas del cliente.

1.2.2 metodo de estudio

Ajustes de investigación. Este método de investigación utilizado en los cuatro casos proyectos es el enfoque de estudio de caso controlado. El enfoque combina aspectos de experimentos, estudios de casos e investigación-acción, y está especialmente diseñado para el estudio de metodologías ágiles. En resumen, en síntesis se busca llevar a cabo un proyecto que tiene un prioridad de entregar un producto final funcional a un cliente, en configuración cercana a la industria, donde los datos de medición son constantemente recopilados para comentarios rápidos, mejora de procesos e investigación

Recopilación y validación de datos.

Dado que todos los proyectos de casos tenían objetivos comerciales y de investigación igualmente importantes, la recopilación de datos fue diseñado para ser lo más efectivo y extenso posible, pero aun así consumir mínimamente los recursos de personal. La evidencia empírica se ha recopilado de múltiples fuentes de datos para obtener múltiples medidas del mismo fenómeno para mejorar la validez, confiabilidad y credibilidad del proyecto

1.3 RECOLECCIÓN DE DATOS OPERACIÓN Y ANALISIS DE DATOS E INTERPRETACIÓN

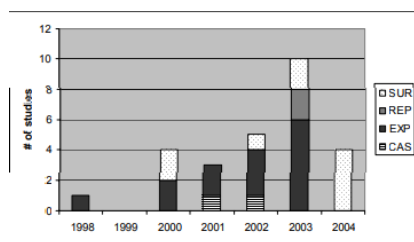


Figura 1. Analisis entre programación en pares y programación sola

El primer estudio empírico reportado sobre programación en pares fue

Código: A12

publicado en 1998 y los estudios más recientes han sido fechados años antes de escribir este artículo (junio 2004). La Figura 1 ilustra la distribución anual de diferentes tipos de estudios empíricos encontrados en la literatura, que se incluyeron en la revisión. Los estudios se han organizado en función de sus enfoques de investigación a las siguientes categorías: estudios de casos (CAS), experimentos (EXP), encuestas (SUR) e informes de experiencia (REPS). Un creciente interés de investigación hacia la programación en pareja puede verse claramente en la figura. Sin embargo, cabe señalar, que mientras que la Figura 1 ilustra las tendencias en los tipos y cantidades de estudios empíricos, no es completamente integral, es decir, estudios centrados en el uso de la programación en pareja con fines educativos en los escenarios universitarios no han sido explorados a fondo.

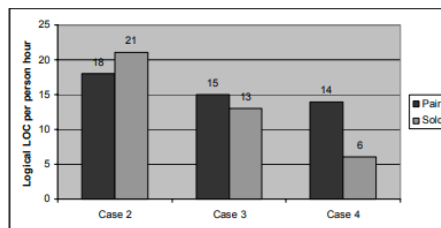


Figura 2. Productividad total de la programación en tres casos de estudio

La figura 2 muestra la productividad total de los pares y solos. La programación en los tres casos de proyectos parece que no hay regularidad entre la productividad de diferentes programas estilos: en el caso dos, la programación en solitario tiene un poco más alta productividad que la programación en pareja, en el caso tres la situación es invertida, y en el caso cuatro, la programación de pares ha cambiado sustancialmente. Mayor productividad que la programación en solitario.

Código: A12

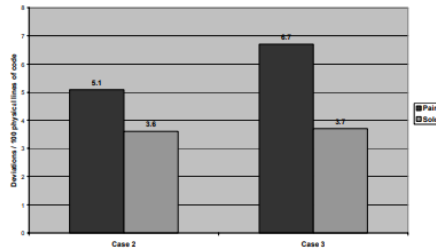


Figura 3. Densidad de las desviaciones estándar de codificación en caso proyectos 2 y 3

. Densidad de las desviaciones estándar de codificación nos sirve para identificar la calidad del software

En la figura 3 muestra la densidad de las desviaciones de los estándares de codificación en casos dos y por 100 líneas físicas de código. Se puede ver que en ambos proyectos, la densidad de desviación fue mucho mayor para el par programación que para la programación en solitario.

1.3 EMPAQUETADO DE RESULTADOS

Trabajos citados

Hulkko, H. (2004). *A Multiple Case Study on the Impact of Pair and Solo programming*.

Obtenido de

https://dl.acm.org/doi/pdf/10.1145/1062455.1062545?casa_token=Uqwz-RcCcyYAAAAA:LPh6P6OgGjh8G209OHF8IbXb6c31VdrjYWD2LeXyx83QdZEY5IysHTGkfbFsXZNFwN7XGTESqoO6WA

Código: A14



Investigación de Ingeniería de Software – NRC: 8506

Autor:



Carrera: Software

Fecha: 05/07/2022

Código: A14

ÍNDICE

- 1. Definición de Requerimientos**
 - 1.1 Definición de problema**
 - 1.2 Objetivos**
 - 1.3 Introducción**
- 2. Planificación del diseño**
 - 2.1 Definición de parámetros de contexto**
 - 2.2 Identificación de sujetos**
 - 2.3 Definición de hipótesis y variables**
 - 2.4 Selección de muestra**
 - 2.5 Selección del diseño experimental**
 - 2.6 Definición del proceso de recolección**
 - 2.7 Restricciones**
 - 2.8 Selección de técnicas de análisis**
 - 2.9 Selección de instrumentación**
- 3. Recolección de datos**
- 4. Análisis de datos**
- 5. Empaquetado de resultados**

Código: A14

1. Definición de Requerimientos

1.1 Definición de problema

El problema que existe es el desconocer cuál de las dos practicas (Pair programming y Solo programming) es mejor para los desarrolladores al momento de crear una aplicación.

1.2 Objetivos

Determinar las diferencias que existe entre la práctica de programación por pares y la programación tradicional, mediante un experimento para comparar el esfuerzo que tienen los desarrolladores al crear una aplicación.

1.3 Introducción

Existen varias prácticas de programación al momento de querer desarrollar un software, esta investigación se enfocará en la Pair programming y Solo programming para poder ver las ventajas y desventajas que cada una tiene, con la finalidad de determinar cuál de las dos practicas es mejor emplearla.

2. Planificación del diseño

2.1 Definición de parámetros de contexto

Contexto: Los estudiantes del Departamento de Ciencias de la Computación de la Universidad de las Fuerzas Armadas ESPE.

Dominio: Las prácticas de pair programming y solo programming las ventajas y desventajas que posee cada una de ellas.

Muestra: Los participantes serán los programadores que cumplan ciertas características que se especificara a continuación.

Código: A14

2.2 Identificación de sujeto

Sujeto: Los desarrolladores que tengan las siguientes características:

- Mínimo estar en 5 semestre.
- Tener conocimiento sobre las prácticas de programación.
- Saber manejar al menos 3 lenguajes de programación.
- Saber manejar el IDE en este caso NetBeans.

Objeto: Los ordenadores que tienen las siguientes características:

- Sistema operativo Windows
- Memoria RAM 16
- Disco duro 1 TB
- Tener instalado el IDE NetBeans.

2.3 Definición de hipótesis y variables

Hipótesis Nula

- Los desarrolladores emplean el mismo esfuerzo al utilizar la practica de Pair programming o Solo programming.

Hipótesis Alternativas

- Los desarrolladores emplean menor esfuerzo al utilizar la práctica de Pair programming.
- Los desarrolladores emplean menor esfuerzo al utilizar la práctica de Solo programming.

Código: A14

2.4 Selección de muestra

Para sacar la muestra de la población se lo hará a través de la fórmula del tamaño de la muestra, en este caso nuestra población será todos los estudiantes del Departamento de Ciencias de la Computación de la Universidad de las Fuerzas Armadas ESPE que cumplan las características mostradas anteriormente, obteniendo así una muestra aceptable para realizar el experimento.

2.5 Selección del diseño experimental

Utilizaremos el método de observación donde pondremos a prueba a las dos prácticas de programación, donde se podrá ver el esfuerzo que tienen los participantes al momento de desarrollar un software, además de tomar datos importantes como tiempo de desarrollo y número de errores.

VARIABLES OBSERVADAS: Tiempo de desarrollo de la aplicación, número de errores encontrados.

2.6 Definición del proceso de recolección

Actividades para realizar:

1. De acuerdo con la muestra obtenida se deberá seleccionar a los participantes que cumplan las características mencionadas.
2. Los ordenadores deben poseer las mismas características como: memoria RAM, disco duro o sólido, tarjeta gráfica, procesador etc.
3. Se deberá instalar en los ordenadores todo lo necesario para el desarrollo de software en este caso el IDE NetBeans donde se desarrollará la aplicación.

Código: A14

4. Los desarrolladores ocuparán los ordenadores para desarrollar un software en específico, todos ellos comenzarán al mismo tiempo y emplearán una sola practica programación.
5. Los observadores deberán tomar nota de todo lo que se les presente a los participantes como: tiempo de finalización, numero de errores en el código, el esfuerzo que tuvieron los participantes etc.
6. Por último, se realizará el análisis de todos los datos obtenidos en este caso a través de una tabla comparativa para poder ver cuál de las dos practicas es la mejor de emplear en el proceso de desarrollo del software.
7. Todos los documentos que se generen serán subidos al repositorio Google Drive

2.7 Restricciones

- No debe existir una amistad previa entre los participantes, para poder tener buenos resultados.
- Los pares deben ser formados de manera aleatoria.
- La practica de programación debe ser asignada de forma aleatoria.

2.8 Selección de técnicas de análisis

Ejecución de una Tabla Comparativa de todas las variables observadas entre las dos prácticas.

Código: A14

2.9 Selección de instrumentación

Emplear una encuesta en la cual se evalúe el tiempo de desarrollo, el esfuerzo de los participantes, el número de errores encontrados entre las dos prácticas. Además, dicha encuesta será realizada en la herramienta de Google Forms.

3. Recolección de datos

En este caso al utilizar el método de observación, los observadores deberán estar tomando nota de todo lo que sucede en el experimento, estos datos pueden ser guardados en un documento Excel y ser subido a un repositorio para que todos los involucrados puedan acceder a dicha información.

También se ejecutará la encuesta a los participantes para obtener la información sobre el esfuerzo que tuvieron al realizar la aplicación con la practica de programación que se les designo.

4. Análisis de datos

Los datos pueden ser analizados atravésé de una tabla comparativa para poder ver las diferencias que existen entre estas dos practicas de programación, además de cuantificar los datos y poder representarlos en un diagrama de barras o pastel. Esta información se la puede subir o tener en el repositorio Google Drive.

5. Empaquetado de resultados

Se debe generar un informe estructurado, donde se pueda mostrar los resultados alcanzados de la investigación y este se encontrará en el repositorio Google Drive.

Código: B06



ESPE
UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA

**DEPARTAMENTO EN CIENCIAS DE LA
COMPUTACIÓN
CARRERA DE INGENIERÍA DE SOFTWARE**

**INVESTIGACIÓN EN LA INGENIERÍA DE
SOFTWARE**

NRC: 8506

Taller Planificación de Experimento

AUTOR:



DOCENTE:

Ingeniero Efrain R. Fonseca C.

FECHA: 2022/07/05

SANGOLQUÍ-ECUADOR

Código: B06

Introducción	3
Diseño del experimento	3
Meta	3
Participantes	3
Material del experimento	4
Tareas del experimento	4
Variables y parámetros	4
Variable independiente	4
Variables dependientes	4
Parámetros	5
Hipótesis	5
Diseño del experimento	5
Anexos	6
3.1 Repositorio de preguntas de LeetCode utilizados	6
3.2 Cuestionario de esfuerzo requerido para realizar el desarrollo del software	6
Referencias	7

Código: B06

1. Introducción

El desarrollo de software es un proceso complejo que amerita de una correcta identificación de los errores para evitar retrasos en el futuro. De esta manera, se puede realizar el mismo utilizando varias metodologías de desarrollo en concreto, una de las metodologías más ampliamente utilizadas es la de programación a pares o *pair programming*, a su vez podemos encontrar la contraparte de este en el *solo programming* o programación estándar donde se realiza el desarrollo por una sola persona

1.1 Pair Programming

De acuerdo con Gillis, A. en 2021 es una metodología derivada del *Extreme Programming (XP)* donde dos programadores trabajan juntos en un solo computador. Estas dos personas trabajan juntas para, diseñar, programar y testear el código. Idealmente en la programación a pares es necesario que los dos programadores tengan habilidades de programación similares.

2. Diseño del experimento

De acuerdo con lo planteado por Wohlin, C en 2012 la taxonomía correspondiente del proceso de experimentación de software sigue esta estructura:

2.1 Meta

Analizar la metodología Pair Programming vs Solo Programming

Con el propósito de comparar los efectos

Respecto al esfuerzo requerido en el desarrollo

Desde el punto de vista de los investigadores

En el contexto de estudiantes de ingeniería de software de la universidad de las fuerzas armadas ESPE desarrollando software en Python

2.2 Participantes

Ya que se desea estudiar el esfuerzo requerido en el proceso de desarrollo de software utilizando *Pair Programming* se considerarán ingenieros en software con habilidades de codificación similares en el lenguaje de programación Python. Se va a separar a los

Código: B06

participantes en dos partes un grupo que se enfocará en desarrollar un software haciendo uso de *Pair Programming* y otro que realizará el mismo software, pero haciendo uso de *Solo Programming*. Estos se separarán, basados en un tratamiento al azar, separándolos en grupos sin ninguna consideración.

2.3 Material del experimento

Se va a utilizar principalmente dos materiales (véase anexos para mayor detalle):

- Un cuestionario que se enfoque en dos cosas: (1) evaluar la facilidad con la que se consideró que se realizó el software y (2) que tan agradable consideró el desarrollador el proceso de desarrollar software en la metodología correspondiente (véase anexo 4.2)
- Problemas de programación encontrados en LeetCode para evaluar las capacidades de desarrollo (véase anexo 4.1)

2.4 Tareas del experimento

Durante el experimento los participantes realizarán las siguientes tareas: (1) resolver preguntas del sitio LeetCode hasta completarlas o en su defecto hasta que se rindan, (2) rellenar un cuestionario luego de realizar los programas correspondientes de la experiencia de desarrollo con respecto a sus respectivas metodologías.

2.5 Variables y parámetros

2.5.1 Variable independiente

La metodología de desarrollo utilizada: en este caso se limita a dos *Pair Programming* y *Solo Programming*

2.5.2 Variables dependientes

Esfuerzo requerido para realizar el software: este se obtendrá a partir de el cuestionario planteado al final del proceso de desarrollo de software, siendo una variable cuantitativa, que representará a que tan cómodos se sintieron los participantes en el proceso de desarrollo del software. Este se obtendrá haciendo un cuestionario donde el resultado del esfuerzo es el promedio de todas las preguntas realizadas.

Código: B06

2.5.3 Parámetros

Los factores a tomar en cuenta deben ser: la falta de experiencia de los desarrolladores con el problema planteado, la interacción entre las parejas.

2.6 Hipótesis

H_0 : No hay diferencia en el esfuerzo requerido para desarrollar software utilizando Pair programming en comparación a Solo Programming.

H_1 : Hay diferencia en el esfuerzo requerido para desarrollar software utilizando Pair programming en comparación a Solo Programming.

2.7 Diseño del experimento

En este proceso se realizará la asignación al azar de los participantes al grupo de estudio de Pair Programming o de Solo Programming. De esta manera se realizará el siguiente diseño:

	Metodología de desarrollo	
	Pair Programming	Solo Programming
Esfuerzo requerido para desarrollar el software después de hacer la codificación	G1	G2

Este es el caso de “Un factor con dos tratamientos” descrito por Wohlin, C en 2012 por tanto se utilizará una técnica de análisis *T de Student* para extraer los resultados relevantes del esfuerzo promedio requerido para desarrollar el software dependiendo de la metodología utilizada.

Código: B06

3. Anexos

3.1 Repositorio de preguntas de LeetCode utilizados

- Add Two Numbers (<https://leetcode.com/problems/add-two-numbers/>)
- Longest Substring Without Repeating Characters (<https://leetcode.com/problems/longest-substring-without-repeating-characters/>)
- Longest Palindromic Substring (<https://leetcode.com/problems/longest-palindromic-substring/>)
- Zigzag Conversion (<https://leetcode.com/problems/zigzag-conversion/>)
- Reverse Integer (<https://leetcode.com/problems/reverse-integer/>)

3.2 Cuestionario de esfuerzo requerido para realizar el desarrollo del software

Grupo: _____

Responda las preguntas planteadas a continuación donde 1 es poco de acuerdo y 5 es muy de acuerdo:

1. El esfuerzo que utilice fue proporcional al de los problemas planteados

1	2	3	4	5

2. Fui capaz de analizar y resolver los problemas sin ningún tipo de problemática externa

1	2	3	4	5

3. (En caso de haber trabajado con algún compañero) Considero que mi compañero proporciono ideas que mitigaron el esfuerzo del proceso de desarrollo

1	2	3	4	5

Código: B06

4. Referencias

- GENERO BOCCO, M. A. R. C. E. L. A., CRUZ LEMUS, J. A., & PIATTINI VELTHUIS, M. A. R. I. O. (2015). Métodos de investigación en ingeniería del software. " EDICIONES DE LA U LTDA".
- Gillis, L. (2021). Pair Programming. Recuperado de: <https://www.techtargget.com/searchsoftwarequality/definition/Pair-programming>
- Wohlin, C., Runeson, P., Höst, M., Ohlsson, M. C., Regnell, B., & Wesslén, A. (2012). Experimentation in software engineering. Springer Science & Business Media.

Código: B07

ESPE
UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA



DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN

CARRERA:

ING. DE SOFTWARE

NRC:

8506

ASIGNATURA:

INVESTIGACIÓN EN INGENIERÍA DE SOFTWARE

TAREA 9:

PLANIFICACIÓN EXPERIMENTO PAIR PROGRAMMING VS SOLO PROGRAMMING

NOMBRE: ██████████

FECHA:

05/07/2022

Código: B07

Índice

1. Alcance	3
2. Planificación	3
2.1. Selección de contexto	3
2.2. Selección de sujetos.....	3
2.3. Selección de variables.....	3
2.4. Formulación de Hipótesis.....	4
2.5. Elección del diseño.....	4
2.6. Instrumentación	4
2.7. Evaluación de la Validez	4
3. Operación.....	4
3.1. Preparación	4
3.2. Ejecución	5
3.3. Validación de los Datos.....	5
4. Análisis e Interpretación	5
5. Presentación y Difusión	5

Código: B07

1. Alcance

Analizar la técnica de desarrollo ágil programación por pares y la programación en solitario con el propósito de evaluar con respecto a la efectividad desde el punto de vista de los desarrolladores en el contexto de los estudiantes de niveles superiores de la Ingeniería de Software de la Universidad de las Fuerzas Armadas ESPE.

2. Planificación

2.1. Selección de contexto

El contexto en el cuál se desarrollará este experimento con el fin de obtener resultados más generales van a ser los estudiantes de niveles superiores (7mo Nivel) de la Ingeniería de Software de la Universidad de las Fuerzas Armadas ESPE ya que ellos conocen y han aplicado estas técnicas de programación por pares en trabajos grupales y de igual forma con la programación en solitario mediante la ejecución de tareas individuales durante su carrera y serían de gran utilidad para conocer sus opiniones ya sean estas a favor o en contra.

Aunque esta sería una situación ideal para reproducir el experimento, también se lo podría realizar en niveles inferiores (2do Nivel), aunque los estudiantes aún no tengan el conocimiento o experiencia suficiente programando, ya van a verse involucrados con la carrera y esto le podría ser beneficioso para el desarrollo de la misma.

2.2. Selección de sujetos

En el experimento participarán 25 alumnos de séptimo nivel de la materia de Aseguramiento de la Calidad de Software de la Universidad de las Fuerzas Armadas ESPE, los cuales cuentan ya con la experiencia necesaria en la programación, a tal nivel de tener un software totalmente listo para medir su calidad, de esta forma evitamos el temor de la validez del experimento, ya que todos cuentan con los mismos conocimientos.

2.3. Selección de variables

Las variables a continuación describen se aplican tanto para la técnica de programación por pares como para la programación en solitario, para de esta forma compararlas y evaluarlas adecuadamente

2.3.1. Variables Dependientes

Tiempo de desarrollo de un software: se mide en horas de trabajo

Número de errores encontrados: se mide con un número entero

Cantidad de líneas de código/hora: se mide en el número de líneas escritas en cada hora de trabajo desarrollando el software.

Código: B07

2.3.2. Variable Independiente

Programa que cumple determinada función, sin importar como este construido.

2.4. Formulación de Hipótesis

Se formularon las siguientes hipótesis considerando las dos variables dependientes propuestas:

- H_0 : La eficiencia de un programa no cambia a pesar de ser desarrollada mediante pair programming o solo programming.
- H_1 : Es mejor aplicar la programación por pares ya que se detectan a tiempo los errores encontrados.

2.5. Elección del diseño

El diseño seleccionado para este experimento ha sido el de Balanceo o equilibrado ya que este nos permitirá tratar con el mismo número de sujetos como con los programas desarrollados por los mismos, además de favorecer y simplificar el análisis estadístico entre las variables dependientes. Se tomará en cuenta un tratamiento, ya que solo se cuenta con una variable independiente, y a esta se le aplicará 3 criterios que son las variables dependientes que son el tiempo, número de errores, y el número de líneas de código/hora.

2.6. Instrumentación

Los medios para realizar la experimentación son los siguientes:

- Objetos experimentales: En este caso se tomará en cuenta el código escrito de cada programa.
- Guías: Se utilizará una descripción del proceso que cada sujeto deberá realizar, de forma general es lo siguiente:

Programar en cada una de las técnicas.

Los sujetos deben realizar un seguimiento de las líneas de código escritas cada hora.

Los sujetos deben realizar un seguimiento de las horas trabajadas

Los sujetos deben realizar un seguimiento del número de errores al final de cada día de trabajo.

2.7. Evaluación de la Validez

La forma de confiar en los resultados del experimento será de validez interna debido a la forma en que se estructuro a los sujetos.

3. Operación

3.1. Preparación

Los sujetos involucrados deben estar preparados y motivados para colaborar con la experimentación

Código: B07

3.2. Ejecución

Realizado también en horas clase, para observar y modificar el comportamiento de los sujetos

3.3. Validación de los Datos

Para la validación de datos se describió los cambios.

4. Análisis e Interpretación

Análisis estadístico de los datos recolectados, tiempo, errores y horas trabajadas.

5. Presentación y Difusión

Identificación de lecciones aprendidas.

Código: B08



**DEPARTAMENTO EN CIENCIAS DE LA
COMPUTACIÓN
CARRERA DE INGENIERÍA DE SOFTWARE**

**INVESTIGACIÓN EN LA INGENIERÍA DE
SOFTWARE**

NRC: 8506

Planificación de Experimento

NOMBRE:



DOCENTE:

Ingeniero Efraín R. Fonseca C.

FECHA: 2022/07/2022

SANGOLQUÍ-ECUADOR

Código: B08

1. Definición de requerimientos /Alcance

1.1. Planteamiento del problema

En el desarrollo del software es necesario que se realice a través de un método de tal manera que facilite al desarrollador es por eso que debe elegir cual usar en el momento indicado, es por eso que es necesario realizar una comparación entre Pair Programming vs Solo Programming, de tal manera que encontremos el esfuerzo que requiere cada una de ellas a la hora de hacer una aplicación Web.

Problema: Determinar qué tipo de metodología permite a los desarrolladores de software una mejor creación de aplicaciones de software a través del uso de una de ellas.

Actores: Los desarrolladores de software deben ser capaces de crear una aplicación Web en parejas y solos.

Impacto: Brinda una mayor eficiencia a la hora de crear una aplicación, ya que al usar una de estas metodologías, puede reducir tiempo y esfuerzo.

1.2. Definición contexto

La programación en pares consiste en que dos personas trabajan conjuntamente en un mismo programa en el cual el uno se encarga de escribir el código es decir es el conductor, mientras que la otra persona se encarga de revisar cada línea para comprobar si hay errores. Mientras que la programación sola, quiere decir que la misma persona es la que se encarga de todo en el código.

Es por esa razón que al pasar los años en la Ingeniería de Software algunos prefieren trabajar con una metodología y otros no, esto se debe muchas veces debido a los requerimientos del proyecto, o porque los desarrolladores en base de su experiencia optan por trabajar con dicha metodología.

1.3. Definición de objetivos

Definir cual metodología Pair Programming o Solo Programming requiere de un mayor esfuerzo a la hora de realizar una aplicación Web, funcionalidad, tiempo, que permita reconocer el mejor entre ellos, a través de un experimento por un grupo de estudio.

Analizar las diferencias entre Pair Programming vs Solo Programming para comprender las dificultades que se presentan a la hora de realizar una aplicación de software.

2. Planificación de diseño

2.1. Definición de parámetros de contexto

El experimento consiste en realizar una aplicación Web para el centro Médico de la Universidad de las Fuerzas Armadas Espe, de tal manera que los estudiantes de software de quinto semestre , tendrán que realizarlo en los laboratorios de computo, lo cual en un plazo de tiempo se hará dicha

Código: B08

aplicación en parejas, mientras que en otro lapso tendrán que hacerlos solos, con el fin de que deberán presentar la aplicación, y posterior a ello responder un survey que permita determinar con cual método se quedan.

2.2. Identificación de sujetos

Los sujetos son los estudiantes de software que cursan el quinto semestre en Universidad de las Fuerzas Armadas Espe.

2.3. Definición de hipótesis y variables

Ha: El 75% de desarrolladores de software prefieren Pair Programming que Solo Programming para el desarrollo de aplicaciones de software porque requiere de menos esfuerzo, tiempo y hace el mejor uso de los recursos.

Ho: El 75% de desarrolladores de software no prefieren Pair Programming que Solo Programming para el desarrollo de aplicaciones de software porque requiere de menos esfuerzo, tiempo y hace el mejor uso de los recursos.

Variable dependiente: Requiere de menos esfuerzo, tiempo y hace el mejor uso de los recursos.

Variable independiente: El 75% de desarrolladores de software prefieren Pair Programming que Solo Programming para el desarrollo de aplicaciones de software.

2.4. Selección de muestra

Para realizar este experimento se tomará estudiantes de quinto semestre de la Universidad de las Fuerzas Armadas Espe, de forma aleatoria, es decir un total de 14 estudiantes, en el cual en un lapso de tiempo van a realizar una aplicación Web para el centro Médico de la universidad en donde lo van a realizar solos, posterior a ello deberán realizar la misma aplicación pero en parejas, en este caso se tomara en cuenta la experiencia de los estudiantes para formar las parejas de forma equilibrada.

2.5. Selección del diseño experimental

Como se mencionó el punto anterior se va a tomar a 14 estudiantes de forma aleatoria, sin embargo, hay que recalcar que en los laboratorios de computo, se va a instalar Visual Studio Code, para que solo trabajen en el desarrollo de la aplicación Web para el centro médico.

2.6. Definición del proceso de recolección

Código: B08

Cuando: En dos días los estudiantes deberán realizar la aplicación solos, posterior a ello, tendrán dos días mas para hacerlo en parejas.

Restricciones: Los equipos que van a trabajar los estudiantes deben contar con las mismas características y con el mismo programa instalado con el fin de evitar sesgos y obtener un resultado más confiable.

Como: Los estudiantes van a realizar una aplicación Web para el centro medico de la ESPE, en donde van a usar los dos métodos, cuando finalice ello van a responder un survey que permita identificar con que método se quedan.

2.7. Selección de técnicas de análisis

Para este experimento se tiene variables cuantitativas como es el esfuerzo, funcionalidad, entre otras, entonces para ello necesitamos de una técnica que permita hacer la comparación ente estos métodos que se van usar es por eso que se ha optado usar el T de Student con datos especificos, y por otro lado se va a realizar un survey que también nos permita obtener dichos datos.

2.8. Selección de instrumentación

Una vez que haya finalizado la aplicación Web con las dos metodologías se va hacer un survey para que puedan evaluar cada ítem, en donde cada una de estas preguntas tendrán una escala de puntuación de 1-5 en donde uno es la puntuación más baja, que significa “poco satisfactorio”, y cinco es la más alta, que significa “muy satisfactorio”. A continuación, se muestra una tabla que tiene los ítems que se basan las preguntas para cada uno de los métodos.

N° Pregunta	Ítem a Evaluar
1	¿Cuál es la calificación total del método usado?
2	Facilidad que obtuvo al momento de implementar el método
3	Experiencia que tuvo con el método
4	Velocidad en finalizar la aplicación Web
5	Grado de comunicación
6	Errores que se presentaron al momento de implementar la aplicación
7	Calidad en la aplicación

Código: B08

8	Grado de confianza en la aplicación
9	Grado de esfuerzo que puso en la aplicación
10	Satisfacción con el trabajo terminado

Reproducibilidad

Para este experimento usaría la aplicación Nextflow, en donde permite a los usuarios hacer la aplicación desde cero y en tiempo real, es decir en las mismas condiciones, entonces aquí en esta herramienta se encontrarán los parámetros a tomar en cuenta para que se trabaje en parejas y solo, de tal forma que cada usuario pueda identificar que metodología se le hace más fácil emplear al momento de desarrollar una aplicación Web , por otro lado la encuesta y el paper realizado , se puede visualizar en GutHub con el fin de que tengan acceso a lo que ya se ha realizado. Por otro lado, se puede crear un blog que permita hacer foros en donde se hagan debates acerca de Pair Programming vs Solo Programming.

Código: B09

ESPE
UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA



DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN

CARRERA:

ING. DE SOFTWARE

NRC:

8506

ASIGNATURA:

INVESTIGACIÓN EN INGENIERÍA DE SOFTWARE

TEMA:

PLANIFICACIÓN DE EXPERIMENTO



FECHA:

05/07/2022

Código: B09

ÍNDICE

Table of Contents

1. INTRODUCCIÓN.....	3
2. DESARROLLO	3
2.1 PLANTEAMIENTO DEL PROBLEMA	3
2.2 DEFINICION DEL CONTEXTO	3
2.3 DEFINICIÓN DE OBJETIVOS	3
Objetivo General.....	3
Objetivo Específico.....	3
2.4 IDENTIFICACIÓN DE SUJETOS	4
2.5 FORMULACIÓN DE HIPÓTESIS Y VARIABLES	4
Hipótesis nula	4
Hipótesis Alternativa.....	4
Variables	4
2.6 SELECCIÓN DEL DISEÑO EXPERIMENTAL	4
2.7 SELECCIÓN DE INSTRUMENTACIÓN.....	4
2.8 DESCRIPCIÓN DE DATOS.....	5
2.9 VALIDACIÓN DE DATOS.....	5
2.10 REPRODUCIBILIDAD	5
2.10.1 CONTRLOL DE VERSIONES	5
2.10.2 GESTIÓN DE DATOS	5
2.10.3 PRUEBAS DE CÓDIGO	5
CONCLUSIONES.....	5

Código: B09

1. INTRODUCCIÓN

La programación en pares es una técnica ágil de desarrollo de software en la que dos programadores trabajan juntos en una estación de trabajo. Uno, el controlador, escribe el código mientras que el otro, el observador o navegador revisa cada línea de código a medida que se ingresa. Los dos programadores intercambian roles con frecuencia [1].

La programación tradicional o también la programación individual es como su nombre lo indica una sola persona es la que esta programando sin recibir ninguna retroalimentación o corrección de otra persona externa al momento de dicha actividad.

2. DESARROLLO

2.1 PLANTEAMIENTO DEL PROBLEMA

La programación por pares ha sido una práctica que es muy beneficiosa en algunos aspectos, pero también incomprendida y criticada desde el punto de vista operativo, al momento de desarrollar aplicaciones con este tipo de programación. Cada desarrollador tiene diferente percepción comparando con la programación individual.

2.2 DEFINICION DEL CONTEXTO

El esfuerzo que conlleva un desarrollador al momento del desarrollo de aplicaciones web con el uso de la práctica de programación por pares y con el uso de la programación tradicional, existen diferentes formas de asimilarlo y de generar resultados con dichas prácticas.

2.3 DEFINICIÓN DE OBJETIVOS

Objetivo General

Planificar el diseño de un experimento para comparar el esfuerzo de desarrollar una aplicación web entre la práctica de programación por pares en comparación con la programación tradicional.

Objetivo Específico

Desarrollar toda la planificación del experimento enfocado en los dos tipos de programación.

Código: B09

2.4 IDENTIFICACIÓN DE SUJETOS

Como sujetos se decidió escoger a los alumnos de la universidad de las fuerzas armadas ESPE del Departamento de Ciencias de la Computación, un total de 40 alumnos que cursan el quinto nivel de la carrera de ingeniería de software, considerando el conocimiento adquirido en su trayecto universitario hasta dicho nivel es suficiente para que los sujetos puedan entender y aplicar las prácticas de la programación por pares y programación tradicional. Los sujetos deberán ser voluntarios y seleccionados a conveniencia.

2.5 FORMULACIÓN DE HIPÓTESIS Y VARIABLES

Hipótesis nula

El esfuerzo de los desarrolladores no cambia si se aplica la práctica de programación por pares o programación tradicional.

Hipótesis Alternativa

El esfuerzo de los desarrolladores cambia si se aplica la práctica de programación por pares o programación tradicional.

Variables

Variable dependiente: El Esfuerzo que cada desarrollador aplica al usar la programación por pares o programación tradicional considerando su efectividad y eficiencia.

Variable independiente: El Desarrollo de las aplicaciones web.

2. 6 SELECCIÓN DEL DISEÑO EXPERIMENTAL

Se seleccionó un diseño experimental de grupo único debido a que las aplicaciones web para realizar el desarrollo experimental pueden ser aplicaciones "single page" (una sola página) o "multiple page" (varias páginas), lo que se pretende es medir la eficiencia, efectividad con el esfuerzo que realizan al desarrollar dichas páginas. La intervención es de tipo Observación ya que se comparará el esfuerzo que se aplican en las diferentes prácticas de programación que los diferentes sujetos realizan y determinar con practicas es la que menor esfuerzo conlleva.

2.7 SELECCIÓN DE INSTRUMENTACIÓN

Se preparo el objeto experimental que consta del desarrollo de aplicaciones web y el tipo de aplicación web que se va a realizar, separados por dos grupos equivalentes. También se

Código: B09

diseñó las tareas que deberían realizar. Como se muestra en la tabla 1 y la tabla 2 respectivamente.

Durante la instrumentación también se desarrolló un pequeño cuestionario después de realizado el experimento con preguntas de selección múltiple y así poder obtener la percepción de cada grupo de desarrolladores sobre las prácticas de programación.

Tareas	Resumen de la tarea	Tipo de practica	Máxima Puntuación /5
T1	Realización del Front-end	Programación por pares	4 puntos
T2	Realización del Back-end	Programación por pares	5 puntos
T3	Conexión a una Base de datos	Programación por pares	5 puntos

Tabla 1. Resumen de las tareas de los desarrolladores Grupo 1. Single Page

Tareas	Resumen de la tarea	Tipo de practica	Máxima Puntuación /5
T1	Realización del Front-end	Programación tradicional	4 puntos
T2	Realización del Back-end	Programación tradicional	5 puntos
T3	Conexión a una Base de datos	Programación tradicional	5 puntos

Tabla 1. Resumen de las tareas de los desarrolladores Grupo 2. Single Page

2.8 DESCRIPCIÓN DE DATOS

Todos los datos obtenidos a través de la encuesta y las tareas específicas de los diferentes grupos son tomadas en cuenta para un análisis estadístico.

2.9 VALIDACIÓN DE DATOS

Después de la ejecución del experimento se recolecta todos los datos en una tabla diseñada con el fin de otorgar un puntaje y para su análisis respectivo.

2.10 REPRODUCIBILIDAD

2.10.1 CONTRLLO DE VERSIONES

El control de versiones que se usara en este experimento es GIT

2.10.2 GESTIÓN DE DATOS

La gestión que se tiene es a través de los repositorios ya sea local y remotamente.

2.10.3 PRUEBAS DE CÓDIGO

Las pruebas que se realizan son las Unit Test a todas las funcionalidades de la aplicación web

CONCLUSIONES

Se planifico todo el desarrollo del experimento con el problema dado específicamente.

Código: B09

Se logró determinar los aspectos favorables en cada uno de los aspectos tomados en cuenta a la hora de establecer la planificación del experimento.

REFERENCIAS

[1] Wikipedia contributors. (2022, May 4). Pair programming. Wikipedia, The Free Encyclopedia. https://en.wikipedia.org/w/index.php?title=Pair_programming&oldid=1086158771

Wohlin, C. (n.d.). Experimentation in Software Engineering.

Bocco, M. (n.d.). Metodos de investigacion en la Ingenieria de Software.

Código: B10



DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN

CARRERA:

ING. DE SOFTWARE

NRC:

8506

ASIGNATURA:

INVESTIGACIÓN EN INGENIERÍA DE SOFTWARE

TEMA:

PLANIFICACIÓN DE EXPERIMENTO: PROGRAMACIÓN EN PAREJAS VERSUS
PROGRAMACIÓN INDIVIDUAL

ESTUDIANTE:

████████████████████

FECHA:

05/07/2022

Código: B10

PROGRAMACIÓN EN PAREJAS VERSUS PROGRAMACIÓN INDIVIDUAL

1. INTRODUCCIÓN

El presente documento da a conocer dos métodos usados dentro del desarrollo ágil de software, por un lado, tenemos la programación en parejas (pair programming) que consiste en trabajar en el mismo equipo dos programadores de forma conjunta. Uno de ellos (el conductor) escribe el código, mientras que el otro (observador) lo supervisa. Estos roles se van alternando para tener distintas visiones de lo que se está realizando. Por otro lado, tenemos la programación tradicional que tiene como principal objetivo la planificación estructurada de un desarrollo y que muchas veces es llevada a cabo por una sola persona en términos de codificación.

2. DESARROLLO

2.1. DEFINICIÓN DE ALCANCE

Analizar los dos métodos planteados en nuestra temática principal, con el propósito de evaluar en términos de productividad y la calidad del producto final dentro del contexto de desarrollo de software, cuál de los dos métodos es mejor para programar.

2.2. SELECCIÓN DE CONTEXTO

Con el objetivo de obtener los mejores resultados posibles, esta experimentación debe ser ejecutada en proyectos reales, más específicamente dentro de empresas de desarrollo de software en un cierto grupo que se dedique únicamente a la codificación para así lograr una clara diferenciación dentro del ámbito de programación en pares y la programación tradicional a la que están habituados.

2.3. FORMULACION DE HIPOTESIS

“La programación en parejas es más eficiente y se obtienen mejores resultados debido al mayor apoyo y revisión que se tiene de parte de dos sujetos, cada uno con puntos de vista diferentes que llegan a congeniar y dar soluciones de manera más eficaz que si lo hicieran de manera individual”.

2.4. IDENTIFICACIÓN DE SUJETOS

Los sujetos que se identifican para esta experimentación son los desarrolladores, ya que estos son los que llevaran a cabo la codificación de un software tanto individual como por pares. Por otra parte, también se pueden llegar a evaluar los tiempos de desarrollo para verificar cuál de los dos métodos es más eficaz, así mismo una vez finalizado el desarrollo del software solicitado se puede evaluar la calidad del código para conocer su eficiencia y la satisfacción que pueden tener los clientes con dicho software.

2.5. IDENTIFICACIÓN DE VARIABLES

Variable independiente: la variable independiente más precisa para este tipo de experimento es el tiempo, se puede llegar a solicitar una fecha límite establecida para un programa pequeño que nos indique método de programación tiene mejores resultados en el tiempo establecido.

Código: B10

Variable dependiente: la variable más notable en este caso es la calidad del software, el que tan bien se encuentre desarrollado, lo agradable que pueda ser a primera vista y la calidad del código que tendrá es la que más cambios posee al delimitar el tiempo de la experimentación.

2.6. SELECCIÓN DEL DISEÑO

Se selecciona en este caso un testeo a un grupo aleatorio de desarrolladores de software, por un lado, se tendrá a dos programadores que utilizaran el método de programación en parejas y se lo comparara frente a un solo programador, ambas partes desarrollaran un pequeño software que pueda manejar el inventario de una librería; desarrollando esto en un plazo máximo de dos semanas dentro de un lenguaje de programación específico como lo puede ser Python. Bajo estas características se observará y analizará el software culminado en cuando a la calidad y la satisfacción que estos generan en el cliente. De manera más técnica se analiza la calidad del código para conocer cuál de los dos softwares se encuentra en un estándar más alto y ver si la influencia de la metodología de programación en el grupo de la pareja resulta con estándares de calidad más altos que la programación individual del otro grupo.

2.7. DEFINICIÓN DEL PROCESO DE RECOLECCIÓN

Como se comentaba dentro de la selección del diseño, una vez se desee conocer los resultados de la experimentación será con la culminación del desarrollo de un software pequeño con las restricciones propuestas de un plazo de tiempo no muy extenso y con la definición de uso de un solo lenguaje de programación, el cual debe ser conocido por ambos grupos estudiados.

2.8. SELECCIÓN DE TECNICAS DE ANALISIS E INSTRUMENTACION

Para llevar a cabo un buen control y un análisis mas profundo de la calidad del código estos dos grupos deben llevar a cabo un control de versiones notable dentro del repositorio de GitHub que pueda ser visible en todo momento. De esta manera se podrá analizar la calidad del código y el como afecta al primer grupo el tener a dos programadores que van rotando sus roles de conductor y observadores de manera diaria para notar todos los cambios que se llevan durante este proceso. En base a todo esto se usarán diferentes formularios que nos ayuden a analizar este experimento en dos partes. Por un lado, tenemos la manera técnica, este formulario llevara a cabo un análisis cuantitativo en cuanto a tiempos de ejecución, tiempo de desarrollo y los cambios notorios a lo largo de las diferentes versiones presentadas por cada grupo. Mientras que de otra manera se tendrá un formulario que mida de una manera más estadística como es la satisfacción de un grupo de no mas de 100 usuarios que nos indiquen que software tiene mejor calidad en casos visuales, de eficiencia y usabilidad que nos permitan determinar cuál de los dos softwares tiene mejor acogida.

2.9. VALIDACION DE LOS DATOS

Finalmente, para la validación de todos los datos obtenidos tanto de manera técnica como de manera de satisfacción de usuarios, todos estos datos pueden ser recopilados y analizados de manera estadística por medio de un entorno de desarrollo estadístico como lo puede ser Rstudio, donde se puede representar de manera gráfica cuál de los métodos ha obtenido mejores resultados frente al tema técnico como a la calidad que le otorga un grupo de usuarios de sus mismos programas.

Código: B10

3. CONCLUSIONES

Se a planteado un experimento que nos permite realizar una comparación de dos métodos de programación, como lo es la tradicional y la programación en pares. De tal manera que ambos grupos estudiados sean escogidos de manera aleatoria con condiciones específicas como el que ambos grupos sean dedicados al desarrollo, el tiempo que estos tendrá para ello y el lenguaje de programación utilizado.

Código: B13



Investigación de Ingeniería de Software – NRC:
8506

PAIR PROGRAMMING VS SOLO PROGRAMMING



Carrera: Software

Fecha: 05/07/2022

Código: B13

INDICE

1. Definición del Alcance
 - 1.1 Objetivo
 - 1.2 Introducción

2. Planificación del diseño
 - 2.1 Selección del Contexto
 - 2.2 Formulación de hipótesis
 - 2.3 Selección de variables
 - 2.4 Selección de sujetos
 - 2.5 Elección del diseño
 - 2.6 Instrumentación

3. Recolección de Datos
 - 3.1 Motivación de los participantes
 - 3.3 Validación de datos
 - 3.4 Reporte de Problemas

Código: B13

PAIR PROGRAMMING VS SOLO PROGRAMMING

DEFINICION DE ALCANCE

- Objetivo

Desarrollar un experimento de avance eficaz en tiempos y resultados que refleje las ventajas de desarrollar aplicaciones web aplicando (Pair Programming), frente a las características que presenta el método tradicional (Solo Programming). Este estudio es efectuado al ser el Pair Programming una práctica incomprendida y criticada desde un punto de vista operativo. Estudiantes de la ESPE desarrollaran este estudio en el que los sujetos de estudio serán grupos de estudiantes de Software de niveles superiores que realicen proyectos usando los dos métodos. Emplearemos un gran enfoque en identificar como el desarrollo de software es alterado según el método usado.

- Introducción

En 2000, Kent Beck popularizó la noción de programación en pareja y argumentó que esta práctica podría mejorar la calidad del software y el aprendizaje del desarrollador, y no afectaría negativamente la duración, en comparación con la programación en solitario.

La programación en solitario es probablemente la forma más popular de hacer las cosas. Suele ser la forma más fácil desde el punto de vista logístico y las personas sienten que pueden hacer su mejor trabajo cuando trabajan solas. Pueden trabajar a su propio ritmo, pueden desarrollar sus propias soluciones, probar cosas y soluciones nuevas. Por otra parte, algunas personas no se sienten cómodas pidiendo ayuda cuando se atascan en un problema determinado. Para esas personas, la siguiente forma de programar puede ser muy beneficiosa. En cambio, la programación en pares es una forma de desarrollar software en la que dos desarrolladores trabajan juntos en una máquina para desarrollar una pieza de software. Esta forma de trabajar fomenta la colaboración desde el primer momento. Dos desarrolladores se ven “obligados” a trabajar juntos, a hablar entre ellos para terminar su trabajo. Al hacer la programación en pareja, un desarrollador escribirá el código mientras que el otro observará y revisará ese código. Esto a veces puede aumentar la cantidad de horas que se requieren para terminar una tarea, ya que solo una persona está escribiendo código, pero, si se hace correctamente, el resultado puede ser un código mejor y con menos errores, lo que a cambio reducirá la cantidad de tiempo necesario para volver atrás, revisar, cambiar el código y corregirlo, una vez que se realiza la prueba. Dos cabezas piensan mejor que una parece el lema principal de la programación en pareja. Dos ideas diferentes, dos formas diferentes de abordar un problema parece siempre una buena idea. Pero a veces puede ser perjudicial para la tarea en cuestión. A la hora de elegir a las personas que van a trabajar juntas es importante intentar juntar personas que sean compatibles.

Código: B13

PLANIFICACIÓN DEL DISEÑO

- Selección del Contexto



El experimento se desarrollará en un ambiente en el que estudiantes de Software de niveles superiores estén desarrollando sistemas geográficos. La revisión y análisis de estos será online mediante el uso de repositorios compartidos, en donde se visualizará y supervisará los avances de los proyectos. Se desea obtener resultados específicos en cuanto a cada etapa del desarrollo de los sistemas y su variación según el método usado ya sea (Pair Programming o Solo Programming).

- Formulación de hipótesis

Ho: Pair Programming es un método ineficaz en el desarrollo de sistemas informáticos cuando los desarrolladores son inexpertos ya que produce confusión entre los desarrolladores involucrados, por lo que retrasa tiempo de presentación y eficacia del sistema.

H1: Pair Programming es un método solvente en el desarrollo de sistemas informáticos siempre que los desarrolladores tengan algo de experiencia de trabajo con este método, logrando así mejoras en tiempo y efectividad en el desarrollo.

- Selección de variables

VARIABLES INDEPENDIENTES	Cantidad y complejidad de requisitos del sistema en desarrollo.	
VARIABLES DEPENDIENTES	Calidad y tiempos de desarrollo de en cada fase del ciclo de desarrollo del software.	

Código: B13

- Selección de sujetos



Los sujetos implicados en el experimento son estudiantes de Software de las Universidades ESPE, UDLA y UIDE que estén cursando 6to, 7vo y 8vo semestre.

Como características para los sujetos de nuestro experimento buscamos que aparte de los conocimientos adquiridos durante su tiempo universitario posean un Nivel de inglés alto (C2), certificaciones en Desarrollo Web, Análisis de Datos en Python, además de conocimientos y experiencia en Bases de Datos NOSQL.



Código: B13

- Elección del diseño

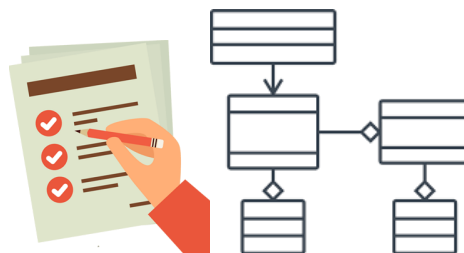
METODO DE ANALISIS – BALANCEO

<p>Los desarrolladores involucrados en el equipo de (Pair Programming) tendrán la misma complejidad de requisitos y tiempos limites que los que estén efectuando (Solo Programming).</p>	<p>Al todos los desarrolladores subir sus avances a un repositorio donde se puede analizar los avances, se evaluará el tratamiento y avance de los sistemas, así como la efectividad y calidad de los mismos.</p>	<p>Buscamos con este método determinar las diferencias que cada método produce en el sistema en cada etapa, para posteriormente analizar el sistema final.</p>																								
 <table border="1"> <caption>Solo vs pair programming</caption> <thead> <tr> <th>Metric</th> <th>1 Pair</th> <th>Solo</th> </tr> </thead> <tbody> <tr> <td>productivity</td> <td>60</td> <td>40</td> </tr> <tr> <td>Coding skills</td> <td>50</td> <td>65</td> </tr> <tr> <td>Debugging skills</td> <td>60</td> <td>40</td> </tr> <tr> <td>Use of new tools</td> <td>20</td> <td>10</td> </tr> <tr> <td>Shorten development time</td> <td>10</td> <td>15</td> </tr> <tr> <td>Explore test case</td> <td>10</td> <td>15</td> </tr> <tr> <td>Parameters</td> <td>10</td> <td>15</td> </tr> </tbody> </table>	Metric	1 Pair	Solo	productivity	60	40	Coding skills	50	65	Debugging skills	60	40	Use of new tools	20	10	Shorten development time	10	15	Explore test case	10	15	Parameters	10	15		
Metric	1 Pair	Solo																								
productivity	60	40																								
Coding skills	50	65																								
Debugging skills	60	40																								
Use of new tools	20	10																								
Shorten development time	10	15																								
Explore test case	10	15																								
Parameters	10	15																								

- Instrumentación

OBJETOS EXPERIMENTALES

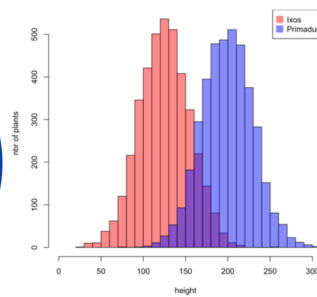
- ❖ Los requisitos de cada sistema a desarrollar
- ❖ Los UML desarrollados como estructura del sistema



Código: B13

INSTRUMENTOS ESTADISTICOS DE MEDICION

- ❖ Rstudio servirá para medir la satisfacción de los sistemas evaluados dado el tiempo de desarrollo y compararlos según (Pair-Solo Programming)



RECOLECCIÓN DE DATOS

- Motivación de los participantes



Incentivos económicos que ayudaran para financiar futuras especializaciones de los estudiantes.

Código: B13



Acceso de cursos para lograr certificaciones en el área de programación

- Validación de datos

La validación de datos se dará mediante la ayuda de maestros especializados en Programación que evaluarán los sistemas desarrollados y maestros en estadística que dominan las métricas de evaluación y las herramientas usadas en el experimento.

- Reporte de Problemas

Los problemas en la ejecución de este experimento deberán ser documentados y archivados en un repositorio en el que se tendrán apuntes de los avances, problemas y novedades de todos los sistemas que están siendo evaluados.

Código: B15

DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN

CARRERA

Ingeniería de Software

INGENIERO

Ing. Efrain Rodrigo Fonseca Carrera

NOMBRE

MATERIA

Investigación en Ingeniería de Software

FECHA

2022/06/27

SEMESTRE Y NRC

S-I MAYO – SEPT 22 – 4322

Código: B15

ÍNDICE

- 1. DEFINICIÓN DE PARÁMETROS DE CONTEXTO**
 - 1.1 Contexto del problema:
 - 1.2 Método de muestreo
- 2. IDENTIFICACIÓN DE SUJETOS**
- 3. DEFINICIÓN DE VARIABLES**
 - 3.1 Variable independiente
 - 3.2 Variable Dependiente
- 4. SELECCIÓN DE MUESTRA**
 - 4.1 Muestra no probabilística o dirigida
 - 4.1.1 Sujetos voluntarios
 - 4.1.2 Expertos
- 5. SELECCIÓN DEL DISEÑO EXPERIMENTAL**
- 6. SELECCIÓN DE INSTRUMENTACIÓN**
- 7. DEFINICIÓN DEL PROCESO DE RECOLECCIÓN**
- 8. TÉCNICAS DE ANÁLISIS**
- 9. REFERENCIAS BIBLIOGRÁFICAS**

Código: B15

1. DEFINICIÓN DE PARÁMETROS DE CONTEXTO

1.1. Contexto del problema:

El desarrollo de un software nuevo no es un proceso sencillo. Dependiendo del tamaño del programa, se deberá tener en cuenta una gran cantidad de posibles coyunturas, funciones y cuestiones problemáticas. Incluso los desarrolladores de software más expertos pueden llegar a desorientarse.

Por lo cual es importante conocer la mejor metodología para aquellos que van a programar varios procesos.

El **pair programming o programación en parejas** especifica que siempre haya dos personas trabajando al mismo tiempo en el código y que, en la medida de lo posible, se sienten juntas. Una se encarga de escribir el código y la otra de supervisarlo en tiempo

Código: B15

real. Al mismo tiempo, están constantemente intercambiando impresiones: debaten problemas, encuentran soluciones y desarrollan ideas creativas.

Por otro lado, tenemos la programación individual donde como su nombre lo dice solo se enfoca en la programación de una sola persona. Donde requiere más tiempo y mucha más concentración de quién lo hace. Ralentizando ciertos procesos, pero mejorando de cierta manera la calidad del software final

¿Para el desarrollo de software es mejor programar en parejas o hacerlo de manera individual?

1.2. Método de muestreo: Para el siguiente experimento se ha escogido el método aleatorio para el muestreo.

2. IDENTIFICACIÓN DE SUJETOS

Para este experimento hemos seleccionado como sujeto a los desarrolladores de software, los cuáles pueden ser junior, semi senior y senior.

Reproducibilidad

Para que nuestro experimento tenga la capacidad de ser reproducible he visto importante la implementación de nuevos “sujetos” que ayuden a la expansión y continuidad del experimento.

Dichos sujetos pueden ser ayudantes de investigación de software, o a su vez expertos en Pc’s que no necesariamente programen pero que ocupen aplicaciones y nos ayuden a comprobar la eficiencia de las mismas en nuestro contexto

3. DEFINICIÓN DE VARIABLES

3.1. Variable independiente

Metodologías de programación.

Las metodologías de investigación nos ayudarán a comprender y a desarrollar de mejor manera software con enfoque principalmente en la calidad del mismo.

3.2. Variable dependiente

Calidad de software

Es importante conocer que a pesar de que se tiene una infinidad de productos de software estos en su mayoría carecen de calidad. Es por eso la importancia de cuál metodología es la mejor para que los programadores pueden producir mejores productos de calidad.

Código: B15

4. SELECCIÓN DE MUESTRA

4.1. Muestra no probabilística o dirigida

La elección de los elementos depende de causas relacionadas con las características del trabajo y del investigador o de quien constituye la muestra. El procedimiento no es mecánico ni se hace con fórmulas de probabilidad, depende del proceso de toma de decisiones de una persona o de un grupo y, por supuesto, tiende a estar sesgadas.

Para este experimento seleccionaremos dos de las muestras no probabilísticas que se adapte mejor a la situación y con la que se pueda obtener mejores resultados.

4.1.1. Sujetos voluntarios

Para determinar si es mejor programar en parejas o hacerlo de manera individual, seleccionaremos a varios voluntarios que tengan experiencia con la programación de las dos maneras que nos puedan aportar experiencias.

4.1.2. Expertos

Personas que opinen sobre el experimento. Ya son especialistas en el tema están más aptos a dar conocimientos acerca de este tema sobre si es mejor programar en parejas o hacerlo de manera individual.

5. SELECCIÓN DEL DISEÑO EXPERIMENTAL

La aleatoriedad se discierne a todo retroceso cuyo proceso es previsible más que en razón de la inversión del azar.

Por lo cual para este diseño he decidido aplicar aleatorización para poder asignar de manera aleatoria tratamiento a los sujetos que con anterioridad describimos.

Programación por pares
<ul style="list-style-type: none">•La programación por pares en cursos introductorios ayuda a que los alumnos obtengan calificaciones más altas en cursos avanzados (Smith et al., 2018)•La preparación individual aunada a la programación en parejas es benéfica para evitar el bajo desempeño académico de los alumnos (Lee et al., 2016)
<ul style="list-style-type: none">•La programación por pares es una herramienta útil que mejora la comunicación entre los estudiantes, la comprensión de los temas y, además, desencadena nuevas ideas (Du et al., 2015)•La programación en parejas mejora los niveles de comunicación y reduce el tiempo en el que se escribe el código (Saltz & Shamsurin, 2017)•El desarrollo de <i>software</i> no tiene que llevarse a cabo todo el tiempo entre dos personas. Se pueden intercalar actividades en solitario y en pares (Swamidurai & Umphress, 2015)•La programación por pares puede usarse ocasionalmente, en especial en las partes más complicadas del <i>software</i> (Meyer, 2018)

Fuente: scielo/programación por pares

Código: B15

6. SELECCIÓN DE INSTRUMENTACIÓN

Para este experimento se preparará una serie de evaluaciones que nos servirán como instrumento para nuestra experimentación.

Dichas evaluaciones serán realizadas con el fin de ser aplicables a nuestros sujetos y poder de esa manera determinar qué metodología es la mejor para el incremento de la calidad del software.

Los sujetos las realizarán y los datos o información obtenido nos servirá para la toma de decisiones y conclusiones a la problemática planteada.

7. DEFINICIÓN DEL PROCESO DE RECOLECCIÓN

Con base a la experimentación, los aspectos a tomar en cuenta a la hora de contrastar la información serán los resultados de la experimentación, podremos evidenciar, mediante los procedimientos de prueba o evaluación, la diferencia entre distintos tipos de metodologías al momento de programar.

Al momento de realizar las pruebas hemos decidido aplicar Formularios de contacto en un sitio web, los cuales nos ayudaran al ordenamiento y a la recolección de datos de nuestro experimento

8. SELECCIÓN DE TÉCNICAS DE ANÁLISIS

Aprendizaje de reglas de asociación

Se trata de un conjunto de técnicas de análisis de datos empleadas para descubrir relaciones interesantes entre variables en grandes bases de datos. La generación y prueba de posibles reglas es el resultado de la aplicación de algoritmos.

Esta técnica nos servirá de apoyo para analizar los datos que obtuvimos con anterioridad y determinar de mejor manera la experimentación de nuestra hipótesis planteada.

REFERENCIAS

- **Wohlin, C., Gustavsson, A., Host, M., Mattsson, C.: A framework for technology introduction in software organizations. In: Proceedings of the Conference on Software Process Improvement, Brighton, pp. 167–176 (1996)**

Código: B15

- Zendler, A.: A preliminary software engineering theory as investigated by published experiments. *Empir. Softw. Eng.* 6, 161–180 (2001). doi: <http://dx.doi.org/10.1023/A:1011489321999>
- Experimentos en ingeniería de software - Coggle Diagram. (s/f). Coggle.it. Recuperado el 21 de junio de 2022, de <https://coggle.it/diagram/YqdOlndxQl8Lc0r/t/experimentos-en-ingenier%C3%ADa-de-software/480f7554f8826df791a4d10e65d8ada33cb8485f3dfbaeb30024496d7f46c826>

Anexo 7

Resultado de encuesta sobre apreciación de los estudiantes a las herramientas planteadas.

Control de planificación de experimento

15 respuestas

[Publicar análisis](#)

Ingrese su nombre

15 respuestas

Jeremy Joel Cadena Díaz

Michelle Cantuña

Marco Pérez

Andrés Coronel

Camila Venegas

Maycol Tituaña

Mateo Andrade

ERICK RIASCOS

Andrés López

Shirley Otuna

Jericó Ruiz

Juan Gallardo

Marcelo Malte

Eduardo Antonio Mortensen Franco

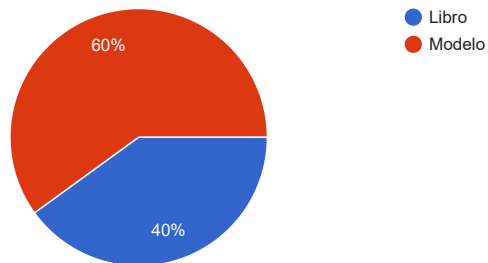
Christopher Daniel Yépez Xhandi



¿Cuál herramienta de soporte al proceso de planificación ha utilizado?

 Copiar

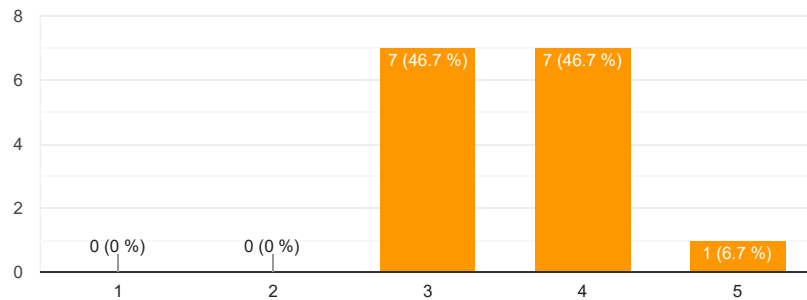
15 respuestas



¿Cuán seguro se considera para planificar un experimento reproducible?

 Copiar

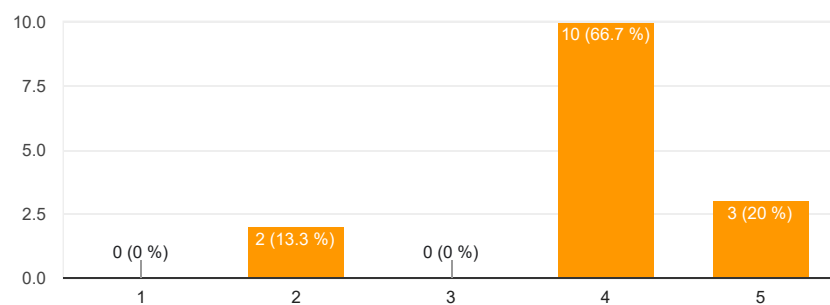
15 respuestas



¿Cuán seguro se considera para planificar un experimento reproducible apoyado por la herramienta de soporte proporcionada?

 Copiar

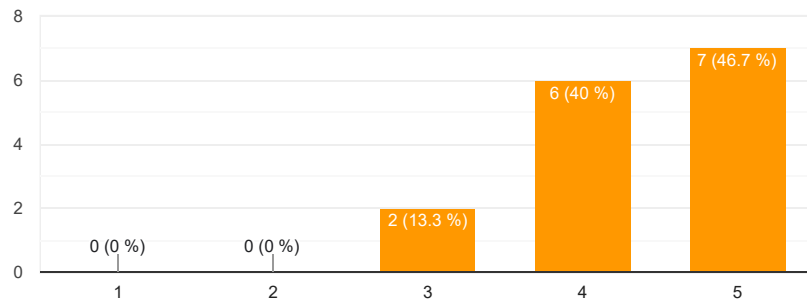
15 respuestas



¿Le resulta útil el uso de la herramienta?

 Copiar

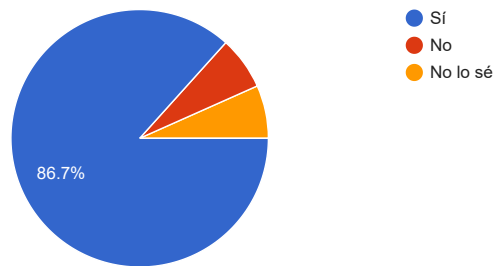
15 respuestas



¿Cree que la herramienta le ayudo a obtener una planificación de calidad?

 Copiar

15 respuestas



Brevemente describa cómo le ayudó la herramienta

14 respuestas

Me ayudo a tener un enfoque mas preciso acerca de la planificación de un experimento y las partes que este conlleva.

A definir claramente que parámetros debo tomar en cuenta para hacer la planificación del experimento.

Gestionar la información de una manera eficaz

Me ayudo a delimitar claramente la estructura

La herramienta fue como un standard para establecer la planificación de un proyecto

A tener una mejor organización y que nuestra planificación sea mas detallada y de calidad.

Me dio las pautas fundamentales para realizar la planificación de un experimento.

Ayuda como guia con recomendaciones para poder realizar un trabajo reproducible

Describía cada una de las etapas, además de tener ejemplos en los cuales basarse para obtener una planificación de calidad.

A tener una idea mas clara para poder realizar la planificación

Facilidad de entender cada uno de los aspectos, aunque tenga carácter científico.

Me ayudo para encontrar información de apoyo para el taller realizado

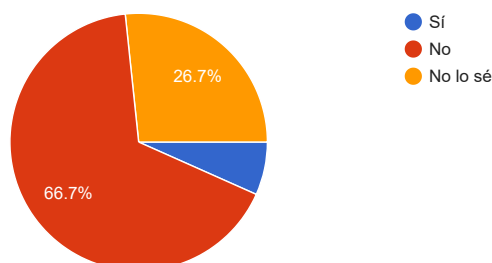
Me ayudo en la planificación del experimento y como se debe desarrollar cada punto

Me ayudo a comprender por partes como realizar, con una explicación concisa y metódica

¿Cree que la herramienta pudo haber entorpecido el desarrollo de la planificación?

 Copiar

15 respuestas



Brevemente describa cómo le afecto la herramienta

11 respuestas

No me afecto en nada.

Me dio una guía clara de que hacer y una explicación de lo mismo

Existen muchos datos a tomar en cuenta por lo que puede ser confuso entender todo completamente.

No presente inconvenientes.

Hubo mucha teoría en ocasiones lo que me confundía, además sería bueno mas ejemplos.

Afecto de manera positiva dado soporte al momento de realizar la planificación

Yo considero que la herramienta no afecto de desarrollo de la planificación

No afecto el uso de la herramienta

No afectó

Me afecto positivamente al momento de tener como guía, como planificar un experimento

Me aportó con información útil, pero la información tan concisa puede provocar errores de comprensión

Google no creó ni aprobó este contenido. [Denunciar abuso](#) - [Condiciones del Servicio](#) - [Política de Privacidad](#)

Google Formularios



Anexo 8

Resultado de encuesta sobre apreciación de los expertos hacia los factores de reproducibilidad propuestos.

Aspectos de una investigación

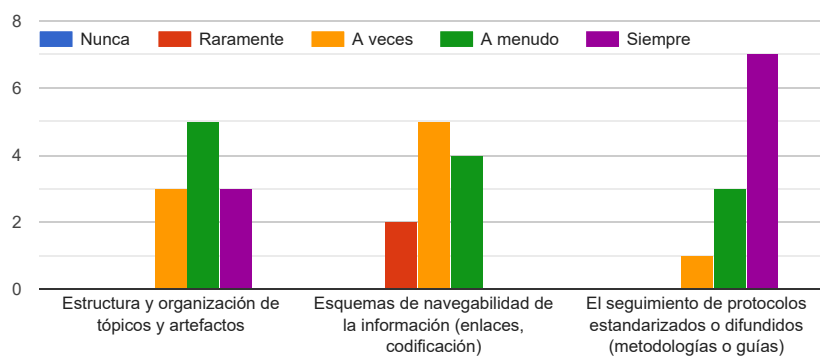
11 respuestas

[Publicar análisis](#)

Auto descripción de una investigación

¿Con qué frecuencia usted aplica estos elementos para conseguir la auto descripción en su investigación?

 Copiar



¿Qué otro elemento aplica para que su investigación pueda ser auto descrita?

5 respuestas

Repositorios digitales

Describir el área de investigación y el problema a resolver

Uso de nomenclatura homogénea en en las figuras y tablas.

Documentación sobre instrumentos, materiales, objetos entre otros que se van a utilizar en la investigación

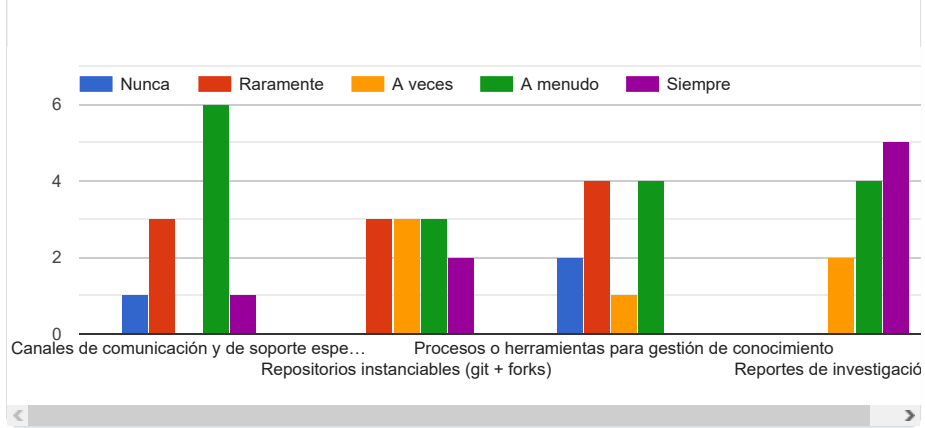
Trazabilidad a través de la descripción detallada de la investigación

Transferibilidad de una investigación



¿Con qué frecuencia usted aplica estos elementos para conseguir la transferibilidad en su investigación?

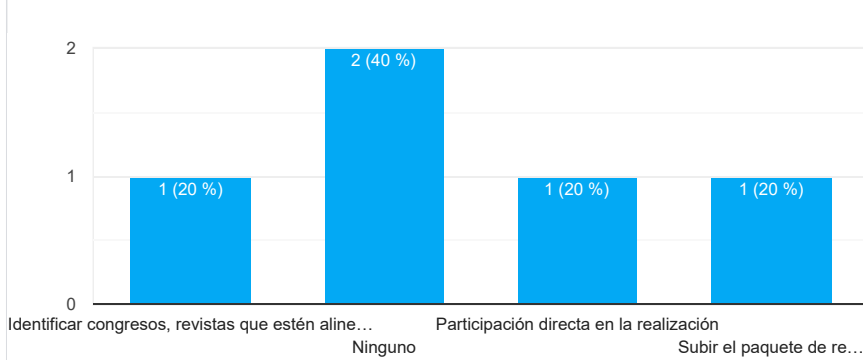
[Copiar](#)



¿Qué otro elemento aplica para que su investigación sea transferible?

[Copiar](#)

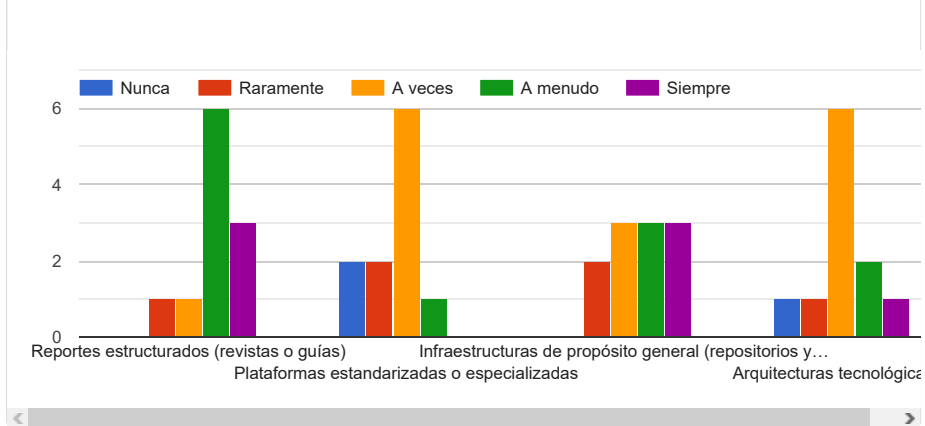
5 respuestas



Estandarización en una investigación

¿Con qué frecuencia usted aplica estos elementos para estandarizar los procesos de su investigación?

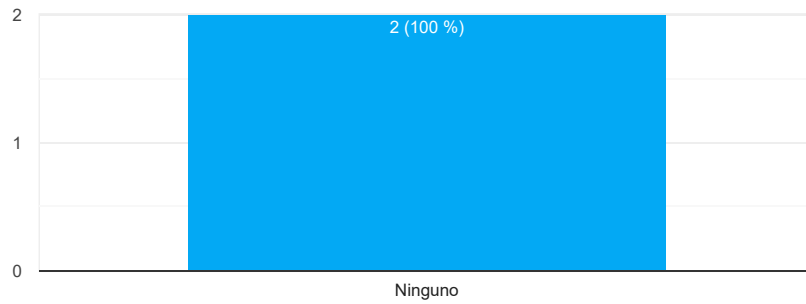
[Copiar](#)



¿Qué otro elemento aplica para que su investigación sea estandarizada?

 Copiar

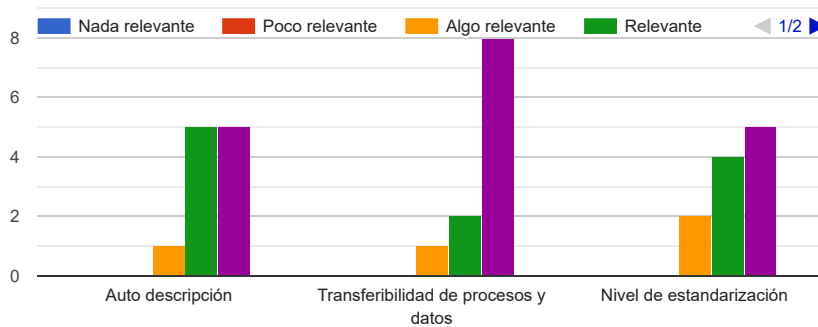
2 respuestas



Reproducibilidad

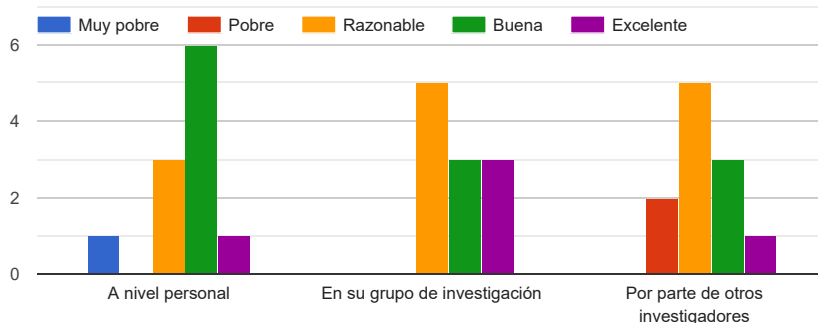
¿Cuán relevantes cree usted que son estos factores en la reproducibilidad de una investigación?

 Copiar



¿En qué medida considera que la investigación, en los ámbitos indicados a continuación, es reproducible?

 Copiar



Anexo 9

Resultado de encuesta sobre la evaluación de expertos a los documentos generados en el experimento.

Revisión de la planificación de una investigación experimental

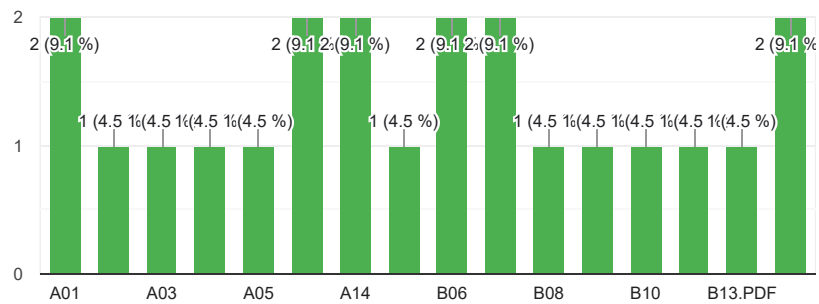
22 respuestas

[Publicar análisis](#)

Ingrese el código del documento provisto para su evaluación

[Copiar](#)

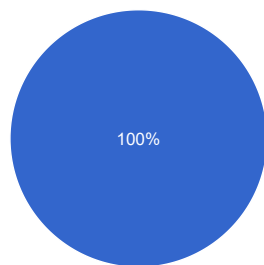
22 respuestas



Para continuar, asegúrese de haber leído todos los documentos a revisar que fueron proporcionados en el correo electrónico.

[Copiar](#)

22 respuestas



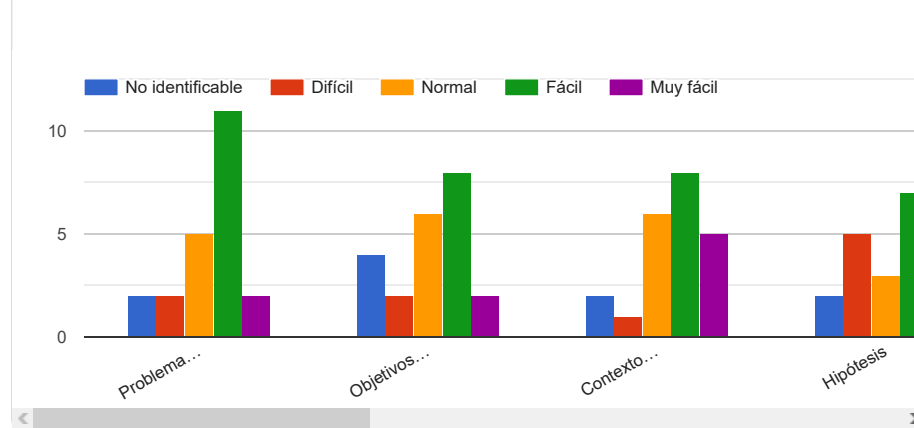
- He leído todos los documentos provistos y he llenado el formulario "Aspectos de una investigación" (Formulario Azul)
- Aún no he leído los documentos provistos o no he llenado el formulario "Aspectos de una investigación"

Revisión



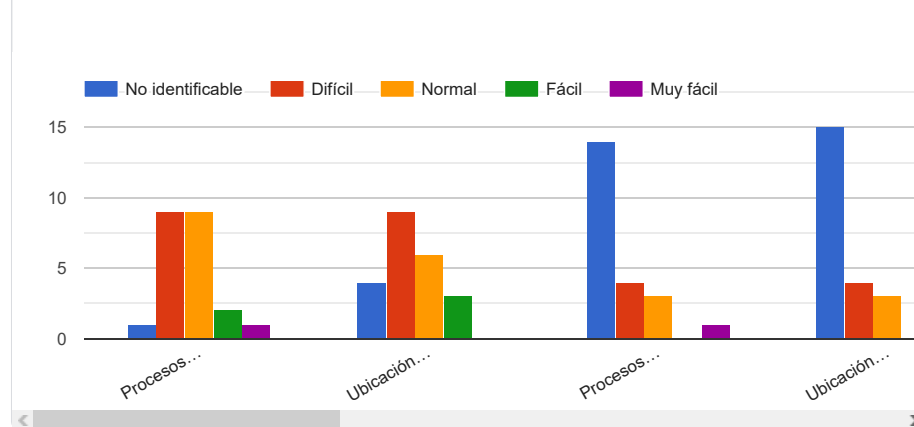
Una investigación debe incorporar muchos elementos básicos ¿Qué grado de dificultad le supuso identificar este tipo de información en el documento?

 Copiar



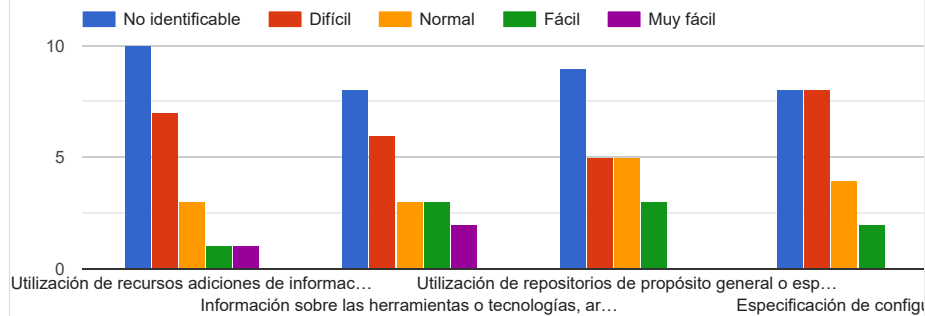
Una investigación presenta mucha información tácita que debe hacerse explícita, principalmente concerniente a procesos ¿Qué grado de dificultad le supuso identificar este tipo de información en el documento?

 Copiar



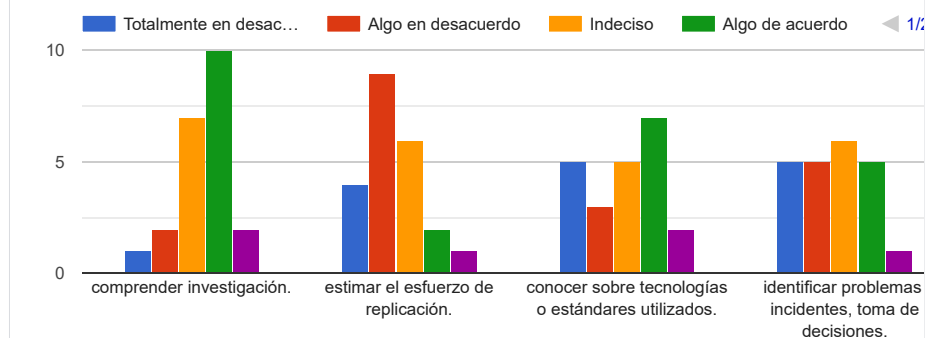
Una investigación se apoya en recursos tecnológicos adicionales que facilitan el desarrollo de la misma ¿Qué grado de dificultad le supuso identificar este tipo de elementos en el documento?

[Copiar](#)



La planificación de una investigación debe considerar varios aspectos de calidad, destacando la reproducibilidad. En función de lo planificado en el documento, considera que la investigación incluye detalles de información para...

[Copiar](#)



Considera que la planificación revisada podría aportar de forma positiva a que la investigación sea:

[Copiar](#)



Anexo 10

Lista de estudios primarios

#	Código	Repositorio	Año	Título
1	LD0002	ACM	2006	Evaluating Guidelines for Empirical Software Engineering Studies
2	LD0008	ACM	2008	A Framework for Software Engineering Experimental Replications
3	LD0009	ACM	2004	Knowledge-Sharing Issues in Experimental Software Engineering
4	LD0010	ACM	2016	ARRESTT: A Framework to Create Reproducible Experiments to Evaluate Software Testing Techniques
5	LD0014	ACM	2016	Experimentation with Dynamic Simulation Models in Software Engineering: Planning and Reporting Guidelines
6	LD0031	ACM	2005	Replicating Software Engineering Experiments: A Poisoned Chalice or the Holy Grail
7	LD0052	IEEEExplore	2008	An Environment to Support Large Scale Experimentation in Software Engineering
8	LD0073	IEEEExplore	2012	A TraceLab-based solution for creating, conducting, and sharing feature location experiments
9	LD0111	IEEEExplore	2007	Extracting Information from Experimental Software Engineering Papers
10	LD0161	IEEEExplore	2007	Modeling the Experimental Software Engineering Process
11	LD0187	IEEEExplore	2017	Describing What Experimental Software Engineering Experts Do When They Design Their Experiments - A Qualitative Study

#	Código	Repositorio	Año	Título
12	LD0189	IEEEExplore	2002	Experience from replicating empirical studies on prediction models
13	LD0207	IEEEExplore	2004	Infrastructure support for controlled experimentation with software testing and regression testing techniques
14	LD0379	IEEEExplore	2014	Reproducibility, correctness, and buildability: The three principles for ethical public dissemination of computer science and engineering research
15	LD0475	IEEEExplore	2013	Replication Data Management: Needs and Solutions – An Initial Evaluation of Conceptual Approaches for Integrating Heterogeneous Replication Study Data
16	LD0511	IEEEExplore	2015	An Initiative to Improve Reproducibility and Empirical Evaluation of Software Testing Techniques
17	LD0745	IEEEExplore	2005	Genetic algorithms to support software engineering experimentation
18	LD0758	IEEEExplore	2013	Identifying Experimental Incidents in Software Engineering Replications
19	LD0890	IEEEExplore	2002	Replicating software engineering experiments: addressing the tacit knowledge problem
20	LD0912	IEEEExplore	2005	Reporting guidelines for controlled experiments in software engineering
21	LD0992	IEEEExplore	2008	On the difficulty of replicating human subjects studies in software engineering
22	LD1014	IEEEExplore	2011	Design Patterns in Software Maintenance: An Experiment Replication at UPM - Experiences with the RESER'11 Joint Replication Project

#	Código	Repositorio	Año	Título
23	LD1047	IEEEExplore	2011	Design Patterns in Software Maintenance: An Experiment Replication at Brigham Young University
24	LD1126	ScienceDirect	2018	Content and structure of laboratory packages for software engineering experiments
25	LD1284	Scopus	2018	Empirical studies omit reporting necessary details: A systematic literature review of reporting quality in model based testing
26	LD1291	Scopus	2017	Describing What Experimental Software Engineering Experts Do When They Design Their Experiments-A Qualitative Study
27	LD1332	Scopus	2017	A comparative study of model-driven approaches for scoping and planning experiments
28	LD1392	Scopus	2016	ARRESTT-A framework to create reproducible experiments to evaluate software testing techniques
29	LD1395	Scopus	2016	An External Replication on the Effects of Test-driven Development Using a Multi-site Blind Analysis Approach
30	LD1477	Scopus	2015	Investigations about replication of empirical studies in software engineering: A systematic mapping study
31	LD1479	Scopus	2015	Mechanisms to characterize context of empirical studies in software engineering
32	LD1546	Scopus	2014	A process-oriented environment for the execution of software engineering experiments

#	Código	Repositorio	Año	Título
33	LD1595	Scopus	2013	Guidelines for reporting productivity studies - A review of the reproducibility of data envelopment analysis in the service sector
34	LD1663	Scopus	2012	Using configuration management and product line software paradigms to support the experimentation process in software engineering
35	LD1665	Scopus	2012	Design patterns in software maintainace: An experiment replication at UPM: Experiences with the RESER'11 joint replication project
36	LD1698	Scopus	2010	Replication of defect prediction studies: Problems, pitfalls and recommendations
37	LD1708	Scopus	2010	On the effectiveness of screen mockups in requirements engineering: Results from an internal replication
38	LD1875	Springer Link	2005	Supporting Controlled Experimentation with Testing Techniques: An Infrastructure and its Potential Impact
39	LD1961	Springer Link	2014	Reporting experiments to satisfy professionals' information needs
40	LD1966	Springer Link	2003	Empirical Studies in ESERNET

Anexo 11

Lista de estudios al 2022

#	Repositorio	Año	Título
1	Scopus	2021	A family of experiments to generate graphical user interfaces from BPMN models with stereotypes
2	Scopus	2020	Standing on shoulders or feet? An extended study on the usage of the MSR data papers
3	Scopus	2020	OCAM: Out-of-core coordinate descent algorithm for matrix completion
4	IEEEExplore	2021	Stop Building Castles on a Swamp! The Crisis of Reproducing Automatic Search in Evidence-Based Software Engineering
5	IEEEExplore	2021	Empirical Comparison of Search Heuristics for Genetic Improvement of Software
6	IEEEExplore	2022	Racking Buggy Files: New Efficient Adaptive Bug Localization Algorithm
