

ESCUELA POLITÉCNICA NACIONAL

FACULTAD DE INGENIERÍA DE SISTEMAS

**DESARROLLO DE UNA APLICACIÓN WEB PARA LA GESTIÓN DE
PRUEBAS ACCESIBLES PARA PERSONAS NO VIDENTES**

**TRABAJO DE TITULACIÓN PREVIO A LA OBTENCIÓN DEL TÍTULO DE
INGENIERO EN SISTEMAS INFORMÁTICOS Y DE COMPUTACIÓN**

FARID SAUD ROLLERI

faridsaud93@gmail.com

DIRECTOR: ING. ENRIQUE ANDRÉS LARCO AMPUDIA, MSc.

andres.larco@epn.edu.ec

Quito, abril 2017

DECLARACIÓN

Yo, Farid Saud Rolleri, declaro bajo juramento que el trabajo aquí descrito es de mi autoría; que no ha sido previamente presentada para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración cedo mis derechos de propiedad intelectual correspondientes a este trabajo, a la Escuela Politécnica Nacional, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normatividad institucional vigente.

Farid Saud Rolleri

CERTIFICACIÓN

Certifico que el presente trabajo fue desarrollado por Farid Saud Rolleri, bajo mi supervisión.

Ing. Enrique Andrés Larco Ampudia, MSc.

DIRECTOR DE PROYECTO

AGRADECIMIENTOS

A mis padres, mis ejemplos a seguir en esta aventura llamada vida. Les agradezco eternamente por los valores inculcados, el apoyo constante que me han brindado durante toda mi vida y los consejos recibidos.

A Andrés Larco, mi director, profesor y gran amigo. Gracias por el apoyo académico dado a través de tus conocimientos, por el apoyo moral brindado a través de tus consejos como humano, por el tiempo y pasión invertida en este proyecto, para que este satisfaga en mayor medida la necesidad existente de las personas no videntes.

A todos mis profesores, por sus conocimientos impartidos, los cuales han sido un aporte valioso para mi desarrollo profesional y académico.

A Ivana, mi novia, por ser parte importante de mi vida, siempre brindándome su apoyo y amor.

DEDICATORIA

A la memoria de mi padre.

CONTENIDO

Resumen	1
Capítulo 1 - Situación actual del problema	2
1.1 Discapacidad visual	2
1.2 Proceso de rendir pruebas	3
1.2.1 Estructura de una prueba	4
1.2.2 Escenarios de pruebas	5
1.3 El proceso de rendir pruebas y la discapacidad visual	6
1.3.1 Análisis	7
1.3.2 Resultados	8
1.4 Datos	8
1.4.1 Nacionales	9
1.4.2 Internacionales	10
1.5 Encuesta inicial	12
1.5.1 Estructura de la encuesta	12
1.5.1.1 Introducción	12
1.5.1.2 Preguntas	13
1.5.2 Resultados	14
1.5.3 Análisis de resultados	17
1.6 Soluciones existentes	17
Capítulo 2 - Solución planteada	21
2.1 Descripción general	21
2.2 Justificación	22
2.3 Tipos de reactivos a implementarse	24
2.3.1 Análisis	25
2.3.1.1 Opción múltiple	25
2.3.1.2 Verdadero o falso	25
2.3.1.3 Respuesta corta	26
2.3.1.4 Respuesta larga	26

2.3.1.5	Completar	27
2.3.1.6	Emparejamiento	27
2.3.2	Resultados	28
2.4	Arquitectura de la aplicación	29
2.4.1	Conceptos claves	29
2.4.1.1	Protocolo HTTP	29
2.4.1.2	Servicios web RESTful	31
2.4.1.3	SOA	32
2.4.1.4	Patrón arquitectónico MVC	32
2.4.1.5	SPA	33
2.4.1.6	Arquitectura de una aplicación web	35
2.4.2	Diseño de la arquitectura de la aplicación	35
2.4.2.1	Arquitectura diseñada	35
2.5	Tecnologías a utilizar	38
2.5.1	Lenguajes	38
2.5.1.1	JavaScript	38
2.5.1.2	HTML5	38
2.5.1.3	CSS3	39
2.5.2	Frameworks	39
2.5.2.1	Bootstrap	39
2.5.2.2	AngularJS	39
2.5.2.3	Sails	40
2.5.2.4	MochaJS	40
2.5.3	Entornos de ejecución	41
2.5.3.1	NodeJS	41
2.5.4	Servicios en la Nube	41
2.5.4.1	Google Cloud Platform	41
2.5.4.2	Google App Engine	42
2.5.4.3	Google Cloud SQL	43
2.5.5	Base de datos	44
2.5.5.1	MySQL	44
2.5.6	APIs	44
2.5.6.1	Google Speech Synthesis	44
2.5.6.2	Google Speech Recognition	45

2.5.7	Herramientas	45
2.5.7.1	Atom	45
2.5.8	Repositorios de código	46
2.5.8.1	GitHub	46
2.6	Mapeo de las tecnologías a utilizar con la arquitectura de la aplicación	46
2.7	Requisitos de la aplicación	49
2.7.1	Cliente	49
2.7.1.1	Hardware	49
2.7.1.2	Software	50
2.7.2	Servidor web	50
2.7.2.1	Hardware	50
2.7.2.2	Software	50
2.7.3	Servidor de base de datos	50
2.7.3.1	Hardware	50
2.7.3.2	Software	50
2.8	Metodología de desarrollo	51
2.8.1	Justificación	51
2.8.2	Descripción	53
2.8.2.1	Roles	53
2.8.2.2	Eventos	54
2.8.2.3	Artefactos	57
2.8.2.4	Reglas	58
Capítulo 3 - Desarrollo de la solución		59
3.1	Generación del Product Backlog	59
3.2	Designación de roles Scrum	61
3.3	Definición de roles de la aplicación	61
3.4	Consideraciones	61
3.5	Sprint 1	63
3.5.1	Sprint Planning	63
3.5.1.1	Sprint Goal	63
3.5.1.2	Sprint Backlog	63
3.5.2	Ejecución del Sprint	63
3.5.3	Sprint Review	65

3.5.4	Sprint Retrospective	65
3.6	Sprint 2	66
3.6.1	Sprint Planning	66
3.6.1.1	Sprint Goal	66
3.6.1.2	Sprint Backlog	66
3.6.2	Ejecución del Sprint	66
3.6.3	Sprint Review	68
3.6.4	Sprint Retrospective	68
3.7	Sprint 3	69
3.7.1	Sprint Planning	69
3.7.1.1	Sprint Goal	69
3.7.1.2	Sprint Backlog	69
3.7.2	Ejecución del Sprint	69
3.7.3	Sprint Review	76
3.7.4	Sprint Retrospective	76
3.8	Sprint 4	77
3.8.1	Sprint Planning	77
3.8.1.1	Sprint Goal	77
3.8.1.2	Sprint Backlog	77
3.8.2	Ejecución del Sprint	77
3.8.3	Sprint Review	86
3.8.4	Sprint Retrospective	86
3.9	Sprint 5	87
3.9.1	Sprint Planning	87
3.9.1.1	Sprint Goal	87
3.9.1.2	Sprint Backlog	87
3.9.2	Ejecución del Sprint	87
3.9.3	Sprint Review	96
3.9.4	Sprint Retrospective	97
3.10	Sprint 6	97
3.10.1	Sprint Planning	97
3.10.1.1	Sprint Goal	97
3.10.1.2	Sprint Backlog	97
3.10.2	Ejecución del Sprint	97

3.10.3 Sprint Review	103
3.10.4 Sprint Retrospective	103
Capítulo 4 - Resultados	104
4.1 Roadmap de pruebas realizadas	104
4.2 Resultados obtenidos en la toma de pruebas con el prototipo funcional	105
4.3 Resultados obtenidos en la toma de pruebas con la aplicación parcial	109
4.4 Resultados obtenidos en la toma de pruebas con la aplicación final . .	111
4.5 Trabajos futuros	116
Capítulo 5 - Conclusiones y recomendaciones	117
5.1 Conclusiones	117
5.2 Recomendaciones	118
Siglas y Acrónimos	120
Glosario	123
Bibliografía	132
Anexos	137
Anexo A - Documentos generados	137
A.1 Encuesta inicial realizada	137
A.2 Lista de soluciones tecnológicas existentes agrupadas por tipo de dis- capacidad visual	137
A.3 Plan del proyecto de titulación	137
A.4 Encuesta sobre el prototipo funcional	137
A.5 Resultados obtenidos de la prueba del prototipo funcional en el even- to Hábitat-EPN	137
A.6 Manual de usuario	137
A.7 Manual de despliegue	137
A.8 Acta de reunión con el Instituto Mariana de Jesús	137
A.9 Encuesta sobre la aplicación final	137
A.10 Resultados obtenidos de la prueba de la aplicación final	137
A.11 Documento técnico JSDoc de la descripción de las funciones existen- tes en los controladores	137

Anexo B - Código de las aplicaciones desarrolladas	138
B.1 Prototipo funcional	138
B.2 Aplicación final	138
Anexo C - Modelos	139
C.1 Modelo conceptual de datos	139
C.2 Modelo físico de datos	139
C.3 Script de creación de la base de datos	139
Anexo D - Resultados de pruebas	140
D.1 Pruebas unitarias	140
D.2 Pruebas de calidad de código	140
D.3 Calidad externa - Facilidad de uso - ISO/IEC 25000	140
Anexo E - SCRUM	141
E.1 Product Backlog	141
E.2 Sprint 1	149
E.2.1 Sprint Backlog	149
E.2.2 Sprint Review	151
E.3 Sprint 2	151
E.3.1 Sprint Backlog	151
E.3.2 Sprint Review	153
E.4 Sprint 3	153
E.4.1 Sprint Backlog	153
E.4.2 Sprint Review	158
E.5 Sprint 4	161
E.5.1 Sprint Backlog	161
E.5.2 Sprint Review	167
E.6 Sprint 5	170
E.6.1 Sprint Backlog	170
E.6.2 Sprint Review	176
E.7 Sprint 6	179
E.7.1 Sprint Backlog	179
E.7.2 Sprint Review	183
Anexo F - Capturas de pantallas de las aplicaciones desarrolladas	186

F.1	Prototipo funcional	186
F.2	Aplicación final	189
F.2.1	Módulo Usuario	189
F.2.2	Módulo Curso	192
F.2.3	Módulo Prueba	199
F.2.4	Módulo Pregunta	206

ÍNDICE DE FIGURAS

1.1 Entradas y salidas presentes en una prueba	4
1.2 Resultados obtenidos en la pregunta 1 de la encuesta inicial	15
1.3 Resultados obtenidos en la pregunta 2 de la encuesta inicial	15
1.4 Resultados obtenidos en la pregunta 3 de la encuesta inicial	16
1.5 Resultados obtenidos en la pregunta 4 de la encuesta inicial	16
2.1 Forma como se adapta la solución planteada al momento de rendir una prueba	24
2.2 Movimientos requeridos para desplazarse por un reactivo de tipo opción múltiple	25
2.3 Movimientos requeridos para desplazarse por un reactivo de tipo verdadero o falso	26
2.4 Movimientos requeridos para desplazarse por un reactivo de tipo respuesta corta	26
2.5 Movimientos requeridos para desplazarse por un reactivo de tipo respuesta larga	27
2.6 Movimientos requeridos para desplazarse por un reactivo de tipo completar	27
2.7 Movimientos requeridos para desplazarse por un reactivo de tipo emparejamiento	28
2.8 Estructura de un mensaje HTTP	29
2.9 Estructura de línea de inicio (start-line) de un mensaje HTTP	30
2.10 Patrón arquitectónico MVC	33
2.11 Comparativa del ciclo de vida de una SPA y el ciclo de vida de una aplicación web tradicional	34
2.12 Arquitectura general de una aplicación web (Cliente-Servidor)	35
2.13 Arquitectura de la aplicación a desarrollarse	37
2.14 Mapeo de las tecnologías utilizadas con la arquitectura de la aplicación	47
2.15 Elementos que conforman Scrum y sus interacciones	53

2.16 Roles que conforman el Scrum Team	55
3.1 Burn-Up Sprint 1	65
3.2 Burn-Up Sprint 2	68
3.3 Burn-Up Sprint 3	76
3.4 Burn-Up Sprint 4	86
3.5 Burn-Up Sprint 5	96
3.6 Burn-Up Sprint 6	103
4.1 Roadmap de pruebas realizadas con la aplicación	104
4.2 Resultados obtenidos en la pregunta 1 de la encuesta Hábitat-EPN .	107
4.3 Resultados obtenidos en la pregunta 4 de la encuesta Hábitat-EPN .	107
4.4 Resultados obtenidos en la pregunta 5 de la encuesta Hábitat-EPN .	108
4.5 Resultados obtenidos en la pregunta 6 de la encuesta Hábitat-EPN .	108
4.6 Resultados obtenidos en la pregunta 7 de la encuesta Hábitat-EPN .	109
4.7 Resultados obtenidos en la pregunta 1 de la encuesta de la aplicación final	113
4.8 Resultados obtenidos en la pregunta 4 de la encuesta de la aplicación final	114
4.9 Resultados obtenidos en la pregunta 5 de la encuesta de la aplicación final	114
4.10 Resultados obtenidos en la pregunta 6 de la encuesta de la aplicación final	115
4.11 Resultados obtenidos en la pregunta 7 de la encuesta de la aplicación final	115
F.1 Interfaz gráfica de la historia de usuario PR1 tarea PR1-1	186
F.2 Cont. Interfaz gráfica de la historia de usuario PR1 tarea PR1-1	187
F.3 Interfaz gráfica de la historia de usuario PR1 tarea PR1-7	187
F.4 Cont. Interfaz gráfica de la historia de usuario PR1 tarea PR1-7	188
F.5 Interfaz gráfica de la historia de usuario MU1 tarea MU1-1	189
F.6 Cont. Interfaz gráfica de la historia de usuario MU1 tarea MU1-1	190
F.7 Interfaz gráfica de la historia de usuario MU2 tarea MU2-1	190
F.8 Interfaz gráfica de la historia de usuario MU3 tarea MU3-1	191
F.9 Cont. Interfaz gráfica de la historia de usuario MU3 tarea MU3-1	191

F.10	Interfaz gráfica de la historia de usuario MC1 tarea MC1-1	192
F.11	Interfaz gráfica de la historia de usuario MC2 tarea MC2-1	193
F.12	Interfaz gráfica de la historia de usuario MC3 tarea MC3-1	193
F.13	Cont. Interfaz gráfica de la historia de usuario MC3 tarea MC3-1	194
F.14	Interfaz gráfica de la historia de usuario MC4 tarea MC4-1	194
F.15	Cont. Interfaz gráfica de la historia de usuario MC4 tarea MC4-1	195
F.16	Interfaz gráfica de la historia de usuario MC5 tarea MC5-1	195
F.17	Cont. Interfaz gráfica de la historia de usuario MC5 tarea MC5-1	196
F.18	Interfaz gráfica de la historia de usuario MC6 tarea MC6-1	196
F.19	Cont. Interfaz gráfica de la historia de usuario MC6 tarea MC6-1	197
F.20	Interfaz gráfica de la historia de usuario MC7 tarea MC7-1	197
F.21	Cont. Interfaz gráfica de la historia de usuario MC7 tarea MC7-1	198
F.22	Interfaz gráfica de la historia de usuario MC8 tarea MC8-1	198
F.23	Interfaz gráfica de la historia de usuario MP1 tarea MP1-1	199
F.24	Interfaz gráfica de la historia de usuario MP2 tarea MP2-1	200
F.25	Interfaz gráfica de la historia de usuario MP3 tarea MP3-1	200
F.26	Cont. Interfaz gráfica de la historia de usuario MP3 tarea MP3-1	201
F.27	Interfaz gráfica de la historia de usuario MP4 tarea MP4-1	201
F.28	Cont. Interfaz gráfica de la historia de usuario MP4 tarea MP4-1	202
F.29	Interfaz gráfica de la historia de usuario MP5 tarea MP5-1	202
F.30	Cont. Interfaz gráfica de la historia de usuario MP5 tarea MP5-1	203
F.31	Interfaz gráfica de la historia de usuario MP6 tarea MP6-1	203
F.32	Interfaz gráfica de la historia de usuario MP7 tarea MP7-1	204
F.33	Interfaz gráfica de la historia de usuario MP8 tarea MP8-1	204
F.34	Cont. Interfaz gráfica de la historia de usuario MP8 tarea MP8-1	205
F.35	Interfaz gráfica de la historia de usuario MR1 tarea MR1-1	206
F.36	Cont. Interfaz gráfica de la historia de usuario MR1 tarea MR1-1	207
F.37	Cont. Interfaz gráfica de la historia de usuario MR1 tarea MR1-1	207
F.38	Interfaz gráfica de la historia de usuario MR2 tarea MR2-1	208
F.39	Cont. Interfaz gráfica de la historia de usuario MR2 tarea MR2-1	208
F.40	Cont. Interfaz gráfica de la historia de usuario MR2 tarea MR2-1	209
F.41	Interfaz gráfica de la historia de usuario MR3 tarea MR3-1	209
F.42	Interfaz gráfica de la historia de usuario MR4 tarea MR4-1	210

ÍNDICE DE TABLAS

1.1	Resultado del análisis del uso de la visión para los escenarios de pruebas mencionados	9
1.2	Personas con alguna discapacidad permanente por más de un año . . .	9
1.3	Personas menores a 12 años con alguna discapacidad permanente por más de un año	10
1.4	Personas con discapacidad visual (no videntes) en el Ecuador	10
1.5	Personas menores a 12 años con discapacidad visual (no videntes) en el Ecuador	10
1.6	Personas con discapacidad visual en el mundo divididas en rango de edades	11
1.7	Soluciones genéricas para la actividad: Acceso al computador	18
1.8	Soluciones genéricas para la actividad: Representación de información de gráficos	18
1.9	Soluciones genéricas para la actividad: Lectura	19
1.10	Soluciones genéricas para la actividad: Tomar notas	19
1.11	Soluciones genéricas para la actividad: Movilización	19
1.12	Soluciones genéricas para la actividad: Escritura	20
1.13	Soluciones genéricas para la actividad: Cálculo matemático	20
2.1	Caracterización de los tipos de reactivos	28
2.2	Mapeo entre las operaciones CRUD y los métodos HTTP	32
3.1	Designación de roles Scrum	61
4.1	Resultados la toma de pruebas con el prototipo funcional en el evento Hábitat-EPN	106
4.2	Resultados de la toma de pruebas con la versión parcial de la aplicación	111
E.1	Product Backlog	142
E.2	Cont. Product Backlog	143
E.3	Cont. Product Backlog	144

E.4 Cont. Product Backlog	145
E.5 Cont. Product Backlog	146
E.6 Cont. Product Backlog	147
E.7 Cont. Product Backlog	148
E.8 Sprint Backlog 1	150
E.9 Review de las historias de usuario y tareas del Sprint 1	151
E.10 Sprint Backlog 2	152
E.11 Review de las historias de usuario y tareas del Sprint 2	153
E.12 Sprint Backlog 3	154
E.13 Cont. Sprint Backlog 3	155
E.14 Cont. Sprint Backlog 3	156
E.15 Cont. Sprint Backlog 3	157
E.16 Review de las historias de usuario y tareas del Sprint 3	159
E.17 Cont. Review de las historias de usuario y tareas del Sprint 3	160
E.18 Sprint Backlog 4	162
E.19 Cont. Sprint Backlog 4	163
E.20 Cont. Sprint Backlog 4	164
E.21 Cont. Sprint Backlog 4	165
E.22 Cont. Sprint Backlog 4	166
E.23 Review de las historias de usuario y tareas del Sprint 4	168
E.24 Cont. Review de las historias de usuario y tareas del Sprint 4	169
E.25 Sprint Backlog 5	171
E.26 Cont. Sprint Backlog 5	172
E.27 Cont. Sprint Backlog 5	173
E.28 Cont. Sprint Backlog 5	174
E.29 Cont. Sprint Backlog 5	175
E.30 Review de las historias de usuario y tareas del Sprint 5	177
E.31 Cont. Review de las historias de usuario y tareas del Sprint 5	178
E.32 Sprint Backlog 6	180
E.33 Cont. Sprint Backlog 6	181
E.34 Cont. Sprint Backlog 6	182
E.35 Review de las historias de usuario y tareas del Sprint 6	184
E.36 Cont. Review de las historias de usuario y tareas del Sprint 6	185

RESUMEN

A lo largo de la vida, a una persona se le evalúa sus conocimientos a través de pruebas. Estas pruebas generalmente son realizadas de forma física en papel presentando limitaciones para que personas no videntes puedan ser evaluadas. En la actualidad en el Ecuador, existen un total de 186 117 personas que poseen una discapacidad visual permanente, es decir, son no videntes. A nivel mundial, este número asciende a 1,42 millones de personas.

El presente proyecto integrador propone como solución el desarrollo de una aplicación web que permite a las personas no videntes rendir pruebas. La aplicación se desarrolló utilizando como metodología de desarrollo a SCRUM, la API Google Web Speech para transformar texto a voz y viceversa, y los servicios de la plataforma de la nube de Google para alojar la aplicación.

Para demostrar el funcionamiento de la aplicación y la satisfacción de los requerimientos, se realizaron pruebas a personas videntes y no videntes, utilizando distintos escenarios para cada uno de ellas. Los resultados fueron satisfactorios, permitiendo tanto a personas videntes como no videntes rendir pruebas utilizando la aplicación.

CAPÍTULO 1

SITUACIÓN ACTUAL DEL PROBLEMA

Este capítulo tiene como objetivo justificar el problema existente de las personas no videntes al momento de rendir pruebas. Se responderán las preguntas como:

- ¿Existe realmente un problema relacionado con el proceso de rendir pruebas cuando la persona que rinde dicha prueba, es una persona no vidente? Ver sección 1.3.
- ¿Cuántas personas se ven afectadas? Ver sección 1.4.
- ¿Es un problema solamente local o también internacional? Ver sección 1.4.
- ¿Existen soluciones ya existentes para este problema? Ver sección 1.6.

1.1 DISCAPACIDAD VISUAL

Para poder entender el término discapacidad visual, es necesario primero conocer la definición de discapacidad, la cual es la siguiente: «Dicho de una persona que padece una disminución física, sensorial o psíquica que la incapacita total o parcialmente para el trabajo o para otras tareas ordinarias de la vida» [1].

Para la Organización Nacional de Ciegos Españoles (ONCE) la discapacidad visual son «las diferentes patologías y alteraciones oculares que pueden reducir en diversos grados o anular la entrada de esta información visual imprescindible para nuestro desempeño diario y bienestar. En este sentido, cuando se habla en general de ceguera o deficiencia visual se está refiriendo a condiciones caracterizadas por

una limitación total o muy seria de la función visual» [2]. Según el grado de falta de visión y el origen de este, la discapacidad visual se puede definir en los siguientes tres tipos [3].

- ❑ **Baja visión:** Cuando la agudeza visual del individuo varía entre rangos de 20/70 a 20/200 (legalmente ciegos) después de las mejores correcciones posibles (lentes u operaciones), o donde el campo visual posee un ángulo de 50 grados o menos [3].
- ❑ **Ceguera:** Cuando el individuo es capaz de percibir únicamente luz o inclusive nada, o dicho individuo es incapaz de usar eficientemente su visión residual [3].
- ❑ **Cerebral o cortical:** Cuando el individuo tiene dificultades interpretando la información visual debido a una condición neurológica donde se presentan problemas en la conexión visual de este [3].

1.2 PROCESO DE RENDIR PRUEBAS

A lo largo de su vida, una persona por distintas situaciones posibles, se enfrenta ante la necesidad de rendir algún tipo de prueba. Esta necesidad se presenta en una persona, indiferente de si sufre o no algún tipo de discapacidad, y las pruebas a rendir pueden ser debido a sus estudios, trabajo, etc.

El objetivo de una prueba es evaluar los conocimientos adquiridos por una persona. Para esto, la persona evaluada debe de alguna u otra forma mostrar estos conocimientos al ente evaluador. Este proceso dependerá del tipo de prueba. De forma general, el proceso de rendir una prueba viene dado por un conjunto de elementos de entrada y salida, estos se aprecian en la Figura 1.1. Los elementos de entrada son toda la información que percibe la persona evaluada para lograr un entendimiento correcto de la prueba, estos son: instrucciones, preguntas, etc. Los elementos de salida son los conocimientos que la persona evaluada plasma sobre la prueba, los cuales son evaluados posteriormente por el ente evaluador.

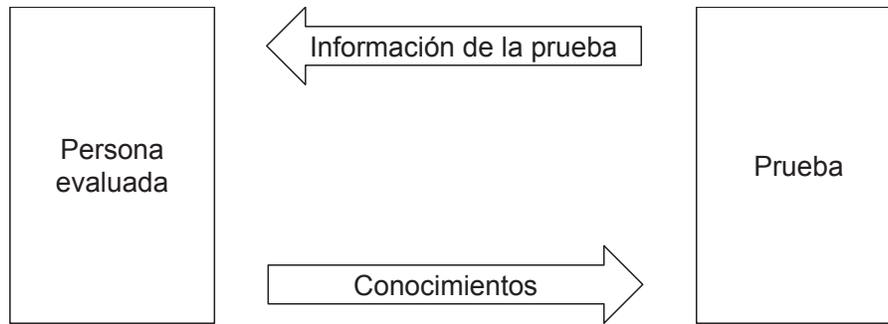


Figura 1.1: Entradas y salidas presentes en una prueba.

1.2.1 ESTRUCTURA DE UNA PRUEBA

Generalmente, una prueba está formada por un conjunto de reactivos (preguntas y opciones disponibles) donde la sumatoria del valor obtenido por el estudiante en cada reactivo, es el valor obtenido por dicho estudiante en la prueba. Adicional, según el tipo de reactivo, el formato de las opciones disponibles para esa pregunta varía. El Massachusetts Institute of Technology (MIT) establece un conjunto de reactivos [4]:

- Opción múltiple:** Conformado por un enunciado y un conjunto finito de opciones disponibles.
- Verdadero o falso:** Conformado por un enunciado y dos opciones posibles: verdadero o falso.
- Respuesta corta:** Conformado por un enunciado y un espacio disponible para una respuesta corta.
- Respuesta larga:** Conformado por un enunciado y un espacio disponible para una respuesta larga.

Adicional a estos tipos de reactivos, el Instituto de Evaluación e Ingeniería Avanzada (IEIA) determina dos tipos de reactivos adicionales [5].

- Completar:** Conformado por un enunciado con espacios en blanco, donde los espacios en blanco representan el texto que la persona evaluada debe escribir

dado un espacio disponible para esto o seleccionar de un conjunto finito de opciones disponibles.

- ❑ **Emparejamiento:** Conformado por dos columnas, donde los enunciados de la columna de la izquierda deben ser emparejados con enunciados de la columna de la derecha.

1.2.2 ESCENARIOS DE PRUEBAS

Según los escenarios de pruebas, la forma como se transmiten los elementos de entrada y salida antes mencionados varían. Se presentan algunos escenarios de pruebas y la forma como se transmiten estos elementos.

- ❑ Prueba escrita en un medio físico como el papel.
 - ✧ **Entrada:** La información de la prueba es percibida a través de la lectura del texto existente en el papel.
 - ✧ **Salida:** Los conocimientos son plasmados en la prueba seleccionando o escribiendo las respuestas que se consideran correctas, usando un bolígrafo o lápiz, en el medio correspondiente, en este caso el papel.
- ❑ Prueba escrita a través del uso de un computador: Para este escenario se considera que la persona evaluada no usa un lector de pantalla como puede ser Job Access With Speech (JAWS).
 - ✧ **Entrada:** La información de la prueba es percibida a través de la lectura del texto existente a través de la interfaz ofrecida por el dispositivo electrónico, generalmente un monitor.
 - ✧ **Salida:** Los conocimientos son plasmados en la prueba seleccionando o escribiendo las respuestas que considere correctas a través de una interfaz ofrecida por el dispositivo electrónico, por lo general ratón y teclado, en el medio correspondiente, el cual puede ser un LMS como el caso de Moodle.

☐ Prueba oral presencial.

- ✧ **Entrada:** La información de la prueba es percibida a través de la escucha de algún tipo de sonido, el sonido es transmitido directamente entre el ente evaluador y la persona evaluada.
- ✧ **Salida:** Los conocimientos son plasmados a través del dictado o habla de las respuestas que considere correctas, la voz es transmitida directamente al ente evaluador.

☐ Prueba oral virtual.

- ✧ **Entrada:** La información de la prueba es percibida a través de la escucha de algún tipo de audio, el audio es transmitido a través de una interfaz existente en un dispositivo electrónico entre el ente evaluador y la persona evaluada.
- ✧ **Salida:** Los conocimientos son plasmados a través del dictado o habla de las respuestas que considere correctas haciendo uso de la voz, por medio de una interfaz ofrecida por algún dispositivo electrónico, por lo general un micrófono, al ente evaluador.

Una persona sin algún tipo de discapacidad puede rendir sin problema alguno una prueba en los escenarios antes mencionados, puesto que la forma como transmite sus conocimientos es accesible para esta. En este escenario, lo único que llega a necesitar dicha persona para rendir la prueba son los conocimientos. En el caso de una persona con algún tipo de discapacidad, esta podrá o no rendir correctamente estos tipos de pruebas, siempre que la prueba le sea lo suficientemente accesible o no a dicha persona.

1.3 EL PROCESO DE RENDIR PRUEBAS Y LA DISCAPACIDAD VISUAL

Se procede a hacer un análisis para determinar si es posible o no para un individuo con ceguera total rendir pruebas frente a los escenarios antes mencionados.

1.3.1 ANÁLISIS

- ❑ Prueba escrita en un medio físico como el papel.
 - ❖ **Entrada:** Para percibir la información de la prueba, la persona evaluada requiere del uso de la visión para poder leer el texto en el papel, a menos que el papel tenga relieve con braille.
 - ❖ **Salida:** Para plasmar los conocimientos en la prueba, la persona evaluada requiere del uso de la visión para observar la respuesta a seleccionar, a menos que el papel tenga relieve con braille.

- ❑ Prueba escrita a través del uso de un computador.
 - ❖ **Entrada:** Para percibir la información de la prueba, la persona evaluada necesita hacer uso de la visión para poder leer el texto existente en la interfaz de salida del dispositivo electrónico, como un monitor, a menos que la persona evaluada esté usando algún tipo de lector de pantalla, y la prueba puede ser leída por este.
 - ❖ **Salida:** Para plasmar los conocimientos en la prueba, la persona evaluada necesita hacer uso de la visión para observar la respuesta a seleccionar, a menos que la persona evaluada esté usando algún tipo de lector de pantalla, y las posibles respuestas puedan ser leídas por este.

- ❑ Prueba oral presencial.
 - ❖ **Entrada:** Para percibir la información de la prueba, la persona evaluada requiere hacer uso de la audición para escuchar el sonido con dicha información, el cual es transmitido directamente desde el ente evaluador a la persona evaluada. Para esta acción la persona evaluada no necesita utilizar su visión.
 - ❖ **Salida:** Para plasmar los conocimientos en la prueba, la persona evaluada necesita hacer uso del habla para transmitir la información al ente evaluador. Para esta acción la persona evaluada no requiere utilizar su visión.

□ Prueba oral virtual.

- ✧ **Entrada:** Para percibir la información de la prueba, la persona evaluada necesita hacer uso de la audición para escuchar el sonido que contiene dicha información, el cual es producido por alguna interfaz de salida existente en el dispositivo electrónico. Para esta acción la persona evaluada no necesita utilizar su visión, a menos que para iniciar la secuencia de reproducción del audio, requiera de alguna interacción en donde sí necesite la visión, por ejemplo dar clic en algún lugar específico .
- ✧ **Salida:** Para plasmar los conocimientos en la prueba, la persona evaluada necesita hacer uso del habla para producir un sonido, el cual contiene los conocimientos a transferir a la prueba, que sea captado por alguna interfaz de entrada existente en el dispositivo electrónico. Para esta acción la persona evaluada no requiere utilizar su visión, a menos que para iniciar la secuencia de grabación del sonido, requiera de alguna interacción en donde sí necesite la visión, por ejemplo dar clic en algún lugar específico .

1.3.2 RESULTADOS

A partir del análisis realizado, se podría concluir que las personas con una discapacidad visual total (ceguera), bajo los escenarios antes mencionados, presentarían problemas para rendir las pruebas de tipo escrita, ya sea en papel o a través de un dispositivo electrónico. Un resumen de los resultados obtenidos a partir de este análisis se presenta en la Tabla 1.1.

1.4 DATOS

En esta sección se muestran los datos relacionados con las personas con algún tipo de discapacidad, especialmente con las personas con discapacidad visual existentes a nivel nacional e internacional.

Tabla 1.1: Resultado del análisis del uso de la visión para los escenarios de pruebas mencionados.

Escenario o tipo de prueba	Requiere visión para obtener información de la prueba (entrada)	Requiere visión para plasmar los conocimientos en la prueba (salida)	Resultado: Puede rendir la prueba sin problema alguno
Prueba escrita en un medio físico como el papel	Sí	Sí	No
Prueba escrita a través del uso de un computador	Sí	Sí	No
Prueba oral presencial	No	No	Sí
Prueba oral virtual	Posiblemente	Posiblemente	Posiblemente

1.4.1 NACIONALES

En el Ecuador, según el Instituto Nacional de Estadísticas y Censos (INEC), existen un total de 816 156 personas que poseen algún tipo de discapacidad permanente por más de un año, de las cuales 186 117 poseen una discapacidad visual total (ceguera)[6], los cuales presentarían problemas al momento de rendir las pruebas de tipo escrito, ya sea en un medio físico como el papel o en algún dispositivo electrónico, bajo los escenarios mencionados anteriormente.

Los datos referentes a las personas que poseen alguna discapacidad permanente por más de un año a nivel nacional se presentan en la Tabla 1.2, y cuando dichas personas son menores a doce años se presentan en la Tabla 1.3.

Tabla 1.2: Personas con alguna discapacidad permanente por más de un año [6].

Área	Sí	No	No responde	Total
Urbano	492 910	7 993 281	604 595	9 090 786
Rural	323 246	4 553 521	515 946	5 392 713
Total	816 156	12 546 802	1 120 541	14 483 499

Tabla 1.3: Personas menores a 12 años con alguna discapacidad permanente por más de un año [7].

Área	Sí	No	No responde	Total
Urbano	58 247	2 136 619	124 692	2 319 558
Rural	42 836	1 415 770	151 075	1 609 681
Total	101 083	3 552 389	275 767	3 929 239

Los datos referentes a las personas no videntes a nivel nacional se presentan en la Tabla 1.4, y cuando dichas personas son menores a doce años se presentan en la Tabla 1.5

Tabla 1.4: Personas con discapacidad visual (no videntes) en el Ecuador[6].

Área	Sí	Se ignora	Total
Urbano	113 208	56 790	169 998
Rural	72 909	31 939	104 848
Total	186 117	88 729	274 846

Tabla 1.5: Personas menores a 12 años con discapacidad visual (no videntes) en el Ecuador[7].

Área	Sí	Se ignora	Total
Urbano	10 344	11 178	21 522
Rural	6 737	7 671	14 408
Total	17 081	18 849	35 930

1.4.2 INTERNACIONALES

Según la Organización Mundial de la Salud (OMS), en su último informe acerca de las personas con discapacidad visual, existen 285 millones de personas con algún tipo de discapacidad visual, de las cuales 39 millones son totalmente ciegas [8]. Por consiguiente, a nivel mundial tendríamos 39 millones de personas que presentarían problemas al momentos de rendir las pruebas de tipo escrito, ya sea en un

medio físico como el papel o en algún dispositivo electrónico, bajo los escenarios mencionados anteriormente.

Los datos referentes a las personas con algún tipo de discapacidad visual, incluyendo las no videntes, a nivel internacional se reflejan en la Tabla 1.6.

Tabla 1.6: Personas con discapacidad visual en el mundo divididas en rango de edades [8].

Edad	Personas no videntes (en millones)	Personas con baja visión (en millones)	Total
0-14	1,421	17,518	18,939
15-49	5,784	74,463	80,248
50+	32,16	154,043	186,203
Todas las edades	39,365	246,024	285,389

Considerando que:

- ❑ Actualmente existen 7,5 billones de personas en el mundo [9].
- ❑ 420 millones de personas hablan español a nivel mundial [10], es decir el 5,6 % de la población.
- ❑ 39 millones de personas son no videntes, es decir el 0,52 % de la población.
- ❑ El porcentaje de personas no videntes se mantiene igual independiente del idioma hablado.

Se obtendría que el porcentaje de la población que habla español y es no vidente sería:

$$0,52\% * 5,6\% = 0,02912\% \quad (1.1)$$

Es decir, existen 2,184 millones de personas que hablan español y son no videntes, las cuales presentarían problemas al momento de rendir las pruebas bajo los escenarios antes mencionados.

1.5 ENCUESTA INICIAL

Para obtener más información acerca de la accesibilidad con la que cuentan las personas no videntes en el país, se desarrolló una encuesta. Esta encuesta fue desarrollada en los idiomas español e inglés utilizando Google Forms, y enviada en el mes de abril del 2016, a todas las instituciones definidas por el Consejo Nacional para la Igualdad de Discapacidades (CONADIS) que tienen relación con las discapacidades [11]. Para ver la encuesta revisar anexo A.1.

Para el desarrollo de la encuesta se utilizó la escala Likert en ciertas preguntas. La escala Likert ofrece un conjunto de posibles opciones calificadas dentro de un rango. Según el tipo de reactivo se puede especificar el nivel en el que se está de acuerdo con alguna afirmación, reactivo o pregunta. Estos niveles pueden reflejar niveles de aceptación, de probabilidad, de deseo, de prioridad, entre otros [12].

1.5.1 ESTRUCTURA DE LA ENCUESTA

La encuesta se encuentra dividida en dos secciones: introducción y preguntas. La sección de introducción cuenta con la presentación de la persona que realiza la encuesta, el objetivo y carácter de la misma. La sección de preguntas, como su nombre lo indica, contiene las preguntas utilizadas para obtener la información deseada.

1.5.1.1 Introducción

Contiene el siguiente texto:

Mi nombre es Farid Saud, estudiante de la Escuela Politécnica Nacional (Quito - Ecuador). La siguiente encuesta es de carácter anónimo, el objetivo es recolectar información acerca de la realidad nacional e internacional de la accesibilidad que tienen las personas no videntes con respecto al uso de la tecnología, específicamente a la accesibilidad que presentan los estudiantes no videntes al momento de rendir pruebas y

exámenes, ya que se pretende desarrollar un sistema para esto.

My name is Farid Saud, student of the National Polytechnic School (Quito - Ecuador). This is an anonymous survey, the purpose of the survey is to collect data about the national and international reality about the accessibility that people with visual impairments have from the use of the technology, more specifically about the accessibility that people with visual impairments have when they are taking tests, because we want to develop a system for that purpose.

1.5.1.2 Preguntas

Se desarrollaron las siguientes preguntas, donde el símbolo * significa obligatorio:

- ¿Usted conoce o ha escuchado anteriormente de accesibilidad? / Have you heard about accessibility? *

 - ✧ Sí / Yes
 - ✧ No

- En su lugar de trabajo o estudios, una persona con discapacidades visuales puede realizar sus actividades con total facilidad. / In your workplace or school/university/college a person with visual impairment can perform all of his/her activities with total ease. *

 - ✧ 1. Totalmente en desacuerdo / Strongly disagree
 - ✧ 2. En desacuerdo / Disagree
 - ✧ 3. Ni de acuerdo ni en desacuerdo / Neither agree nor disagree
 - ✧ 4. De acuerdo / Agree
 - ✧ 5. Totalmente de acuerdo / Strongly agree

- ¿Usted cree que debería existir un software (aplicación, programa) que facilite a los estudiantes con discapacidades visuales rendir pruebas y exámenes? /

Do you believe that it should exist a software (app, program) that makes easier for the student with visual impairment to take tests? *

✧ Sí / Yes

✧ No

En caso de ser estudiante. ¿En su centro de estudio (universidad, colegio, escuela) se cuenta con algún software (aplicación, programa) o herramienta que facilite a los estudiantes con discapacidades visuales rendir pruebas y exámenes? / In case you are an student. Your school/college/university have some software (app, program) or tool that makes easier to students with visual impairment to take tests?

✧ Sí / Yes

✧ No

Comentarios y Sugerencias. / Comments and Suggestions. Tu respuesta

✧ Respuesta abierta

1.5.2 RESULTADOS

La encuesta fue enviada a 258 instituciones, a pesar de esto, se obtuvo un total de 11 respuestas. Esto se debe en parte a que la información de contacto, existente en el documento ofrecido por la CONADIS, en ciertas instituciones no es la actual.

Los resultados que se presentan a continuación, son presentados considerando el orden de las preguntas definidas anteriormente en la subsección 1.5.1.2.

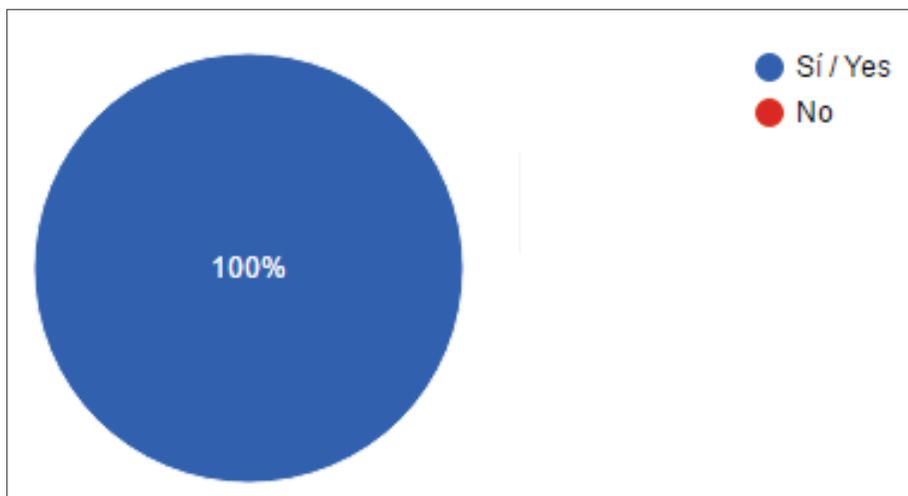


Figura 1.2: Resultados obtenidos en la pregunta: ¿Usted conoce o ha escuchado anteriormente de accesibilidad?

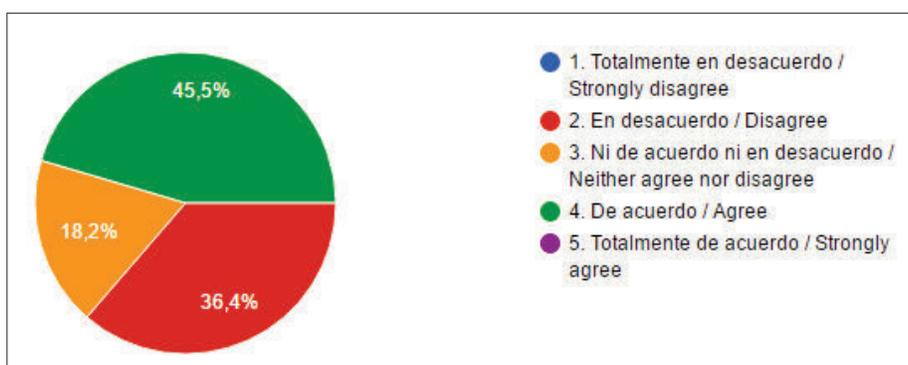


Figura 1.3: Resultados obtenidos en la pregunta: En su lugar de trabajo o estudios, una persona con discapacidades visuales puede realizar sus actividades con total facilidad.

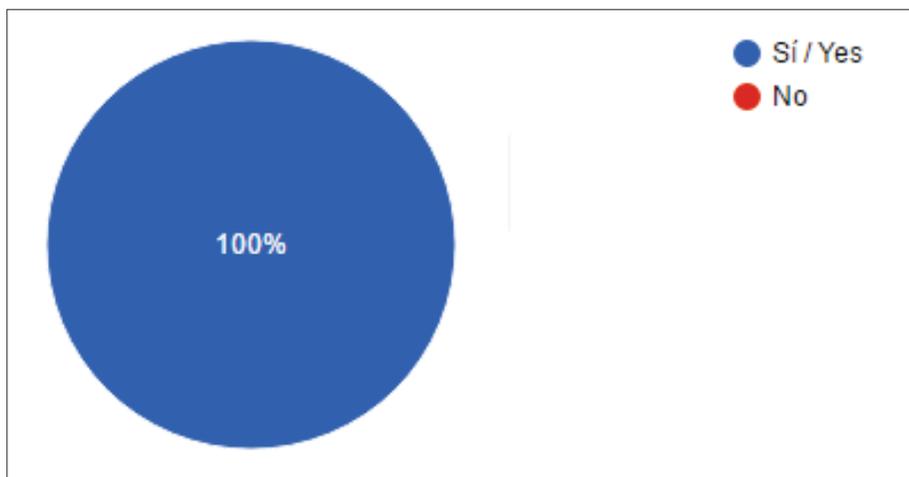


Figura 1.4: Resultados obtenidos en la pregunta: ¿Usted cree que debería existir un software (aplicación, programa) que facilite a los estudiantes con discapacidades visuales rendir pruebas y exámenes?

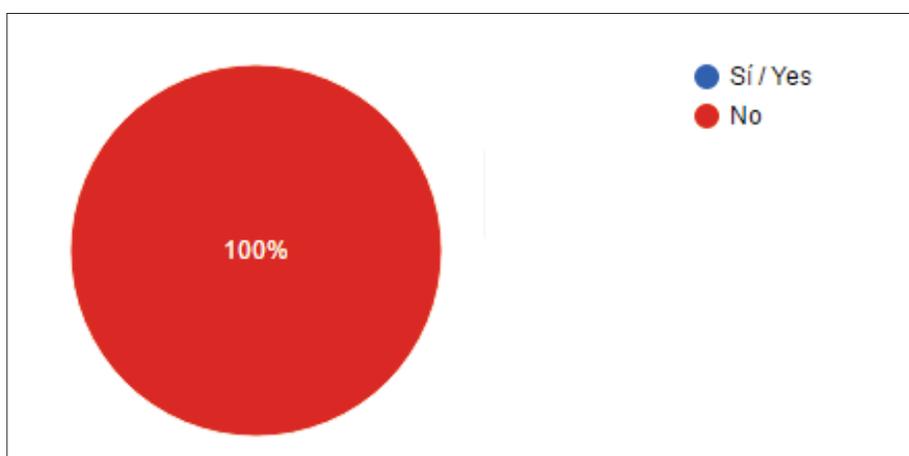


Figura 1.5: Resultados obtenidos en la pregunta: En caso de ser estudiante. ¿En su centro de estudio (universidad, colegio, escuela) se cuenta con algún software (aplicación, programa) o herramienta que facilite a los estudiantes con discapacidades visuales rendir pruebas y exámenes?

Las respuestas obtenidas en la pregunta abierta: Comentarios y Sugerencias; fueron las siguientes:

Uno de los programas que desarrolla nuestra Institución es el de Inclusión Educativa, las dificultades que presentan los estudiantes ciegos y con baja visión al momento de rendir sus pruebas no se ha podido remediar en su totalidad y cuando se trata de brindar el apoyo más adecuado,

el maestro regular no tiene el conocimiento y herramientas necesarias.

Existe desconocimiento sobre un software aplicado a la educación en personas no videntes.

Crear una aplicación exclusiva para invidentes con lector de pantalla claro y sencillo.

Como profesional con discapacidad visual he sentido muchas limitaciones en el momento de rendir las pruebas de los diversos cursos del MI-DEDUC y también presentan dificultades por la modalidad de selección de respuestas.

1.5.3 ANÁLISIS DE RESULTADOS

Todas las instituciones encuestadas conocen del término accesibilidad, sin embargo no todas las instituciones ofrecen las facilidades en sus instalaciones para que las personas con discapacidad visual puedan realizar sus actividades sin problema alguno. Adicionalmente, todos los encuestados concuerdan en que se necesita un software que permita a las personas no videntes dar pruebas, y que en su institución no existe software alguno.

1.6 SOLUCIONES EXISTENTES

Existe un estudio realizado por el Wisconsin Assistive Technology Initiative (WATI) [3] relacionado a las tecnologías de asistencias para personas que tienen baja visión o son no videntes. En este estudio, en base a la actividad que se desea realizar, proponen soluciones genéricas ordenadas jerárquicamente en base a su efectividad. En las Tablas 1.7 a la 1.13 se presentan esas soluciones genéricas agrupadas por actividad, entra mas lejos se encuentre la solución de la actividad será mas efectiva.

Tabla 1.7: Soluciones genéricas para la actividad: Acceso al computador [3].

Acceso al computador
Esquema de colores
Funcionalidades grandes del sistema operativo
Amplificación integrada
Amplificación completamente funcional
Amplificación con lector de pantalla
Lector de pantalla
Lector de pantalla con braille

Tabla 1.8: Soluciones genéricas para la actividad: Representación de información de gráficos [3].

Representación de información de gráficos
Formatos agrandados
CCTV
Modelos u objetos
Gráficos táctiles
Gráficos táctil-audio

Tabla 1.9: Soluciones genéricas para la actividad: Lectura [3].

Lectura
Lentes
Filtro de colores
Tabla de inclinación
Impresiones grandes
Amplificador óptico
Amplificador electrónico
CCTV
Monóculo
CCTV con cámara de distancia
Texto auditivo
Software de lectura
Braille electrónico

Tabla 1.10: Soluciones genéricas para la actividad: Tomar notas [3].

Toma de notas
Pizarra y marcador
Cinta o dispositivo digital de grabación
Software de grabación
Braille electrónico

Tabla 1.11: Soluciones genéricas para la actividad: Movilización [3].

Movilización
Bastón
Monóculo
Compras braille/hablador
Dispositivo de viaje electrónico
Dispositivo GPS

Tabla 1.12: Soluciones genéricas para la actividad: Escritura [3].

Escritura
Bolígrafo de alto contraste
Dispositivo portátil para el procesamiento de palabras
Escritura con soporte de audio
Escritor de braille
Escritura con soporte de braille
Braille electrónico
Reconocimiento de voz

Tabla 1.13: Soluciones genéricas para la actividad: Cálculo matemático [3].

Cálculo matemático
Impresiones grandes de herramientas de medidas
Calculadora con teclas grandes
Dispositivos de medición táctil
Ábaco
Calculadora parlante
Formas o modelos geométricos en 3D & 2D
Representación táctil TIAF

Para información de la lista de las soluciones tecnológicas existentes agrupadas por tipo de discapacidad visual y solución genérica, revisar el anexo A.2.

CAPÍTULO 2

SOLUCIÓN PLANTEADA

Este capítulo tiene como objetivo proponer y justificar una solución al problema planteado en el capítulo anterior. Responderá a las preguntas como:

- ❑ ¿Qué solución se propuso? Ver sección 2.1.
- ❑ ¿Cómo la solución propuesta solventa el problema existente? Ver sección 2.2.
- ❑ ¿Qué arquitectura tiene la aplicación propuesta? Ver sección 2.4.
- ❑ ¿Qué tecnologías son usadas tanto a nivel de hardware como software para soportar a la aplicación? Ver sección 2.5.
- ❑ ¿Cómo estas tecnologías encajan con la arquitectura definida? Ver sección 2.6.

2.1 DESCRIPCIÓN GENERAL

Como se mencionó en el capítulo anterior, el problema de rendir pruebas por parte de las personas no videntes se resume en la falta de accesibilidad para estas personas, debido a la necesidad de utilizar en mayor o menor medida el sentido de la visión al momento de obtener información de la prueba y transferir los conocimientos a esta.

Se propuso como solución, una aplicación web que hace uso de los siguientes elementos:

- ❑ **Un sintetizador de voz:** Encargado de transformar el texto a voz, para hablarle a la persona no vidente las acciones disponibles y el texto existente en la prueba.
- ❑ **Controles sencillos e intuitivos:** Estos controles son utilizados por parte de la persona que rinde la prueba, para navegar en la aplicación y seleccionar las opciones que considera correctas.

El alcance de la solución propuesta no tuvo por objetivo cumplir con todos los lineamientos planteados por la World Wide Web Consortium (W3C), puesto que en el caso de que así fuera, estaríamos limitando la solución únicamente a las personas no videntes que poseen un lector de pantalla y conocen el funcionamiento de los lineamientos establecidos por esta. El objetivo principal es permitir a cualquier persona no vidente rendir pruebas, independiente si esta persona conoce los lineamientos antes mencionados o posee un lector de pantalla.

2.2 JUSTIFICACIÓN

La razón por la cual se decidió desarrollar una aplicación web y no una aplicación nativa o híbrida se debe a las siguientes condiciones que existían y a los objetivos que se deseaban abarcar:

❑ **Condiciones:**

- ✧ El acceso a Internet es requerido independiente del tipo de aplicación que se vaya a desarrollar, puesto que la información de las pruebas se encontrarán almacenadas en un servidor.
- ✧ El autor del presente proyecto integrador, posee conocimientos y experiencia desarrollando aplicaciones web.

❑ **Objetivos:**

- ✧ Se deseaba abarcar la mayor cantidad de usuarios finales posibles.

Las aplicaciones web ofrecen ventajas sobre las aplicaciones nativas o híbridas. Entre estas ventajas se tienen:

- ❑ No requiere instalador y es independiente del sistema operativo del cliente, requiere únicamente un navegador web y acceso a Internet. Cuando se trabaja con aplicaciones nativa o híbridas: se requiere realizar una instalación previa, el código que ejecutará la aplicación del cliente dependerá de las librerías que posee el sistema operativo y el lenguaje de programación que puede compilar este.
- ❑ Se maneja una sola versión de la aplicación, la versión que reside en el servidor. Cuando se trabaja con aplicaciones nativas, se manejan varias versiones dependiendo de la cantidad de clientes existentes y las versiones que tienen estos instaladas.

Aplicando la solución al análisis realizado en el capítulo anterior, que se muestra en la Figura 1.1, se obtiene como resultado la Figura 2.1. Con esto, se justifica que la solución planteada sí permite a una persona no vidente rendir pruebas. A continuación el detalle de la solución:

- ❑ **Entrada:** La información de la prueba es percibida a través del sonido reproducido por el parlante o audífonos conectados al computador, el sonido es producido originalmente por el sintetizador de voz. En esta acción se hace uso del sentido de la audición.
- ❑ **Salida:** La transferencia del conocimiento por parte de la persona evaluada se realiza a través de la selección de la respuesta correcta usando los controles definidos por medio del teclado. Cabe recalcar que por cada movimiento del usuario, existe una respuesta de audio por parte del computador. En esta acción se hace uso del sentido del tacto.

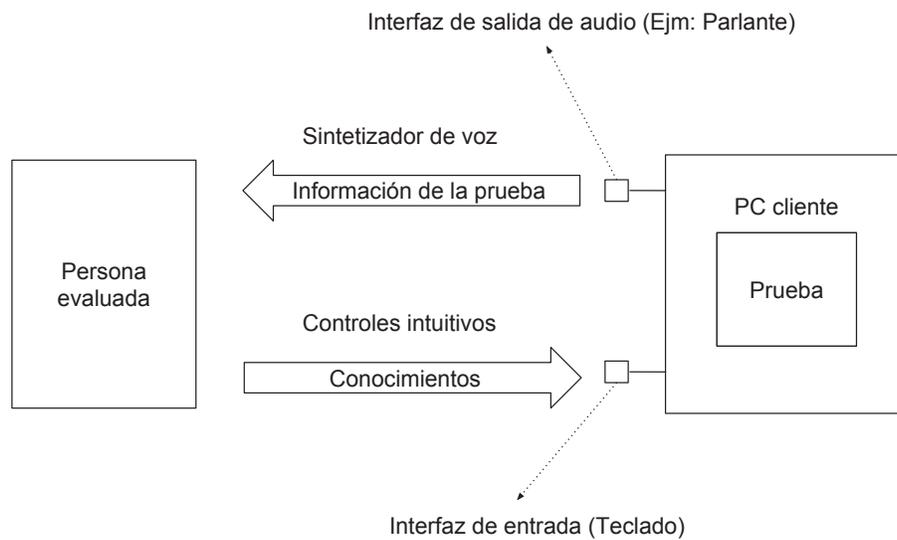


Figura 2.1: Forma como se adapta la solución planteada al momento de rendir una prueba.

2.3 TIPOS DE REACTIVOS A IMPLEMENTARSE

En el capítulo anterior, en la sección 1.2.1, se definió los tipos de reactivos que existen para una prueba. En esta sección se procede a justificar los tipos de reactivos que fueron implementados, tomando como referencia las siguientes condiciones:

- ❑ El reactivo debía ser un reactivo objetivo. Esto para poder evaluar automáticamente la totalidad de la prueba y generar un puntaje para esta cuando una persona evaluada rinda una prueba.

- ❑ La navegación completa del reactivo no debía incluir más de un cambio de dirección en el movimiento. Esto para que la navegación por toda la prueba sea simple.

2.3.1 ANÁLISIS

2.3.1.1 Opción múltiple

Como se muestra en la Figura 2.2, solo existe un cambio de movimiento para desplazarse por todo el reactivo, ya sea de abajo hacia arriba o de arriba hacia abajo. Es un reactivo objetivo puesto que no depende de la subjetividad del ente evaluador, ya que se basa en la selección de la opción correcta, es decir si se seleccionó la opción correcta se obtiene el puntaje correspondiente al reactivo, caso contrario no.

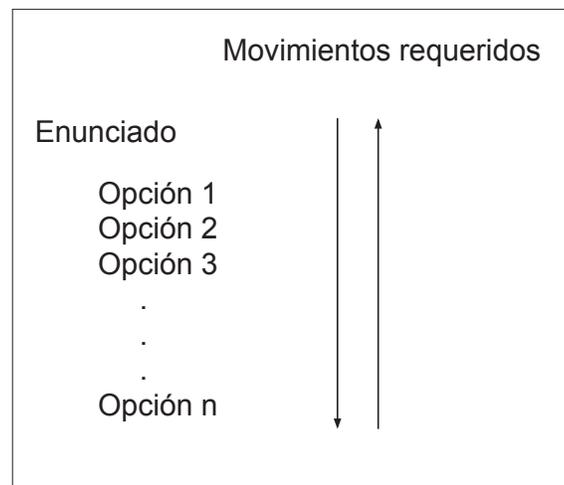


Figura 2.2: Movimientos requeridos para desplazarse por un reactivo de tipo opción múltiple.

2.3.1.2 Verdadero o falso

Como se muestra en la Figura 2.3, sucede lo mismo que en el reactivo opción múltiple.

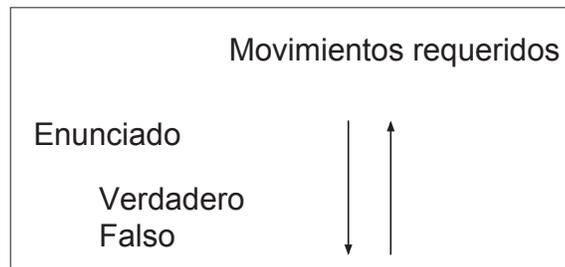


Figura 2.3: Movimientos requeridos para desplazarse por un reactivo de tipo verdadero o falso.

2.3.1.3 Respuesta corta

Como se muestra en la Figura 2.4, solo existe un cambio de movimiento para desplazarse por todo el reactivo, ya sea de abajo hacia arriba o de arriba hacia abajo. Es un reactivo subjetivo, puesto que puede haber una cantidad infinita de posibles respuestas dada por la persona evaluada, y para determinar si es o no una respuesta correcta, esta debe ser analizada por el ente evaluador.

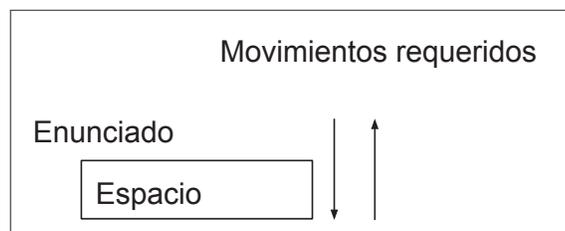


Figura 2.4: Movimientos requeridos para desplazarse por un reactivo de tipo respuesta corta.

2.3.1.4 Respuesta larga

Como se muestra en la Figura 2.5, sucede lo mismo que en el reactivo de respuesta corta.

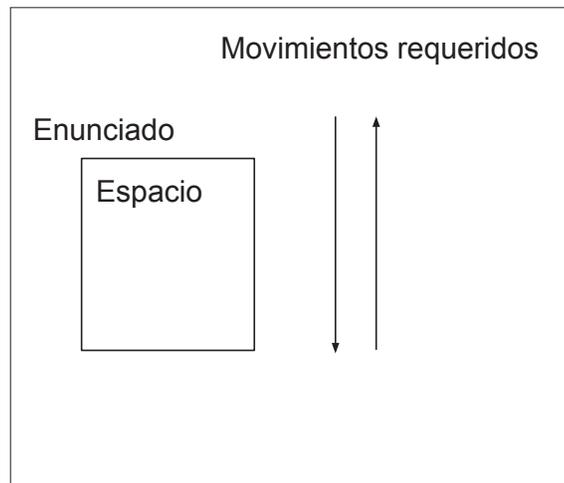


Figura 2.5: Movimientos requeridos para desplazarse por un reactivo de tipo respuesta larga.

2.3.1.5 Completar

Como se muestra en la Figura 2.6, sucede lo mismo que en el reactivo opción múltiple.

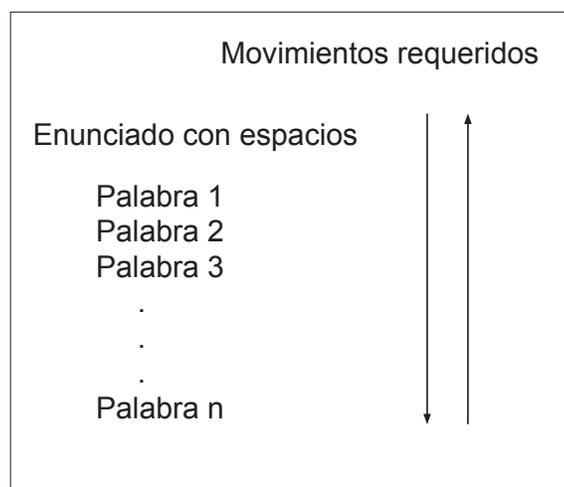


Figura 2.6: Movimientos requeridos para desplazarse por un reactivo de tipo completar.

2.3.1.6 Emparejamiento

Como se muestra en la Figura 2.7, existen hasta tres cambios de movimiento para desplazarse por todo el reactivo, ya sea de abajo hacia arriba o de arriba hacia abajo

para desplazarse entre los enunciados disponibles en la columna, o de izquierda hacia derecha o de derecha hacia izquierda para desplazarse de un enunciado de una columna a un enunciado de otra columna. Es un reactivo objetivo puesto que no depende de la subjetividad del ente evaluador, ya que se basa en la selección de la opción correcta, es decir si se seleccionó la opción correcta se obtiene el puntaje correspondiente al reactivo, caso contrario no.

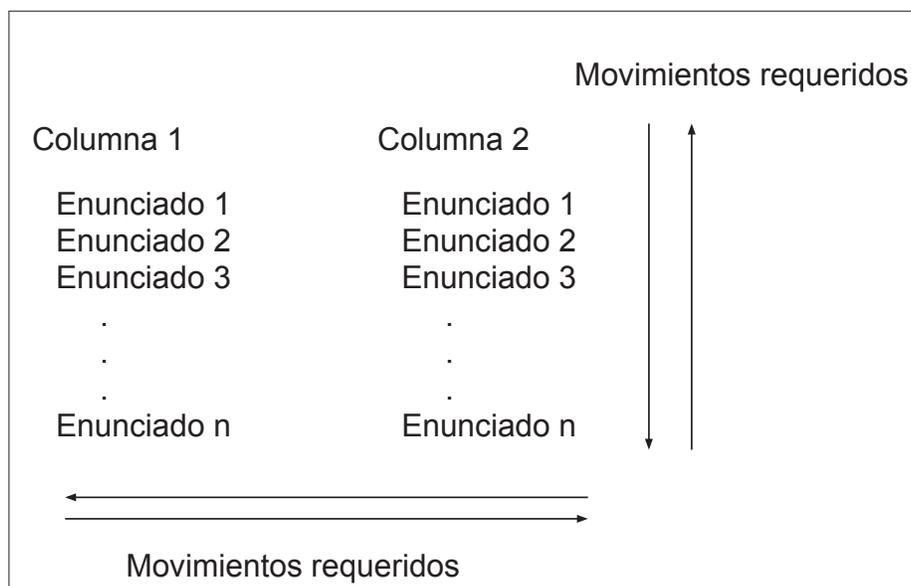


Figura 2.7: Movimientos requeridos para desplazarse por un reactivo de tipo emparejamiento.

2.3.2 RESULTADOS

Tabla 2.1: Caracterización de los tipos de reactivos.

Tipo de reactivo	Número de cambio de direcciones	Es objetiva
Opción múltiple	1	Sí
Verdadero o falso	1	Sí
Respuesta corta	1	No
Respuesta larga	1	No
Completar	1	Sí
Emparejamiento	3	Sí

Los resultados obtenidos se reflejan en la Tabla 2.1. A partir de estos y las consideraciones establecidas al inicio de esta sección, los reactivos implementados fueron:

- Opción múltiple
- Verdadero o falso
- Completar

2.4 ARQUITECTURA DE LA APLICACIÓN

2.4.1 CONCEPTOS CLAVES

2.4.1.1 Protocolo HTTP

Es un protocolo definido en el Requests for Comments (RFC) 2616 por la Internet Engineering Task Force (IETF) en el año de 1999. Este organismo lo define como «protocolo de nivel de aplicación para sistemas distribuidos, colaborativos, información hipermedia» [13]. Adicional, este protocolo es genérico y sin estado [13].

La versión 2.0 se encuentra definida en el RFC 7540, el cual según la IETF «habilita un uso más eficiente de los recursos de la red y una percepción reducida de la latencia al introducir una compresión en el campo cabecera y permitiendo múltiples intercambios concurrentes en la misma conexión» [14].

Los mensajes cuando se utiliza el protocolo Hypertext Transfer Protocol (HTTP) siguen la estructura definida en el RFC 7230, ver Figura 2.8.

```

HTTP-message      =  start-line
                    *( header-field CRLF )
                    CRLF
                    [ message-body ]

```

Figura 2.8: Estructura de un mensaje HTTP [15].

Un mensaje HTTP puede ser la petición del cliente al servidor o la respuesta del

servidor al cliente. La diferencia en la sintaxis entre estos dos tipos de mensajes varía únicamente en la línea de inicio (start-line) [15]. La línea de inicio puede ser una línea de petición (request-line) o una línea de estado (status-line) dependiendo de si se trata de una petición del cliente o la respuesta del servidor respectivamente. Esta estructura se puede apreciar en la Figura 2.9.

```
request-line    =    method SP request-URI SP HTTP-version CRLF
status-line    =    HTTP-version SP status-code SP reason-phrase CRLF
```

Figura 2.9: Estructura de línea de inicio (start-line) de un mensaje HTTP [15].

La línea de petición (request-line) está formada por el método (method), un espacio simple (SP), a quien se le realiza la petición (request-URI), otro espacio simple (SP), la versión HTTP usada en el mensaje (HTTP-version), y un retorno de carro y salto de línea (CRLF) [15].

La línea de estado (status-lines) está formada por la versión HTTP usada en el mensaje (HTTP-version), un espacio simple (SP), el código de estado (status-code), otro espacio simple (SP), una frase textual posiblemente vacía describiendo el código de estado (reason-phrase), y un retorno de carro y salto de línea (CRLF) [15].

La IETF define un conjunto de métodos [13] que pueden ser usados en la línea de petición (request-line) de una petición HTTP. Estos son textualmente los siguientes:

- ❑ **OPTIONS:** «Representa una petición por información acerca de las opciones de comunicación disponibles en la cadena petición (request) / respuesta (response) identificadas» [13].
- ❑ **GET:** «Obtiene cualquier información (en la forma de una entidad) que es identificada con el Uniform Resource Identifier (URI)» [13].
- ❑ **HEAD:** «Idéntico al método GET a excepción de que el servidor no debe devolver el cuerpo del mensaje (message-body) en la respuesta» [13].

- ❑ **POST:** «Solicita que el servidor de origen acepte la entidad encapsulada en la petición como un nuevo subordinado de los recursos identificados por el URI de la petición en la línea de la petición (request-line)» [13].
- ❑ **PUT:** «Solicita que la entidad encapsulada en la petición sea almacenada en el URI de la petición» [13].
- ❑ **DELETE:** «Solicita que el servidor de origen elimine el recurso identificado por el URI de la petición» [13].
- ❑ **TRACE:** «Invoca un loop-back remoto en la capa de aplicación del mensaje de la petición» [13].
- ❑ **CONNECT:** «Se reserva el método para uso con un proxy que puede dinámicamente cambiarse de túnel» [13], utilizado en la conexión HTTP con el servidor proxy.

2.4.1.2 Servicios web RESTful

Los servicios web RESTful son servicios web aplicando la arquitectura Representational State Transfer (REST). Según Oracle, REST es definido como «estilo arquitectónico que especifica restricciones, las cuales, aplicadas a un servicio web, inducen a las propiedades deseadas que permiten a los servicios trabajar de la mejor forma en la web» [16]. Tanto los datos y funcionalidad son considerados recursos, los cuales son accedidos haciendo uso del URI. Estos servicios siguen los siguientes principios [16]:

- ❑ **Identificación de los recursos a través del URI:** Todos los recursos son identificados a través de su URI, el cual permite un espacio de direccionamiento global para los recursos y descubrimiento de servicios [16].
- ❑ **Interfaz uniforme:** Los recursos son manipulados utilizando cuatro operaciones definidas: PUT, GET, POST, y DELETE [16].

- ❑ **Mensajes auto-descriptivos:** Los recursos son separados de su representación para que el contenido pueda ser accedido en una variedad de formatos [16]. Ver Tabla 2.2.
- ❑ **Interacciones con estados a través de hipervínculos:** Cada interacción con algún recurso es sin estado, sin embargo se pueden mantener los estados enviándolos en el mensaje usando distintas técnicas de intercambio de estados como uso de cookies, campos ocultos, etc [16].

Cuando se crean los servicios web, en este caso, servicios web RESTful, existe una relación de correspondencia entre los métodos HTTP existentes y las operaciones Create, Read, Update, Delete (CRUD). Esta relación se describe en la Tabla 2.2.

Tabla 2.2: Mapeo entre las operaciones CRUD y los métodos HTTP.

Operación CRUD	Método HTTP
Create	POST
Read	GET
Update	PUT
Delete	DELETE

2.4.1.3 SOA

Según la W3C, Service-Oriented Architecture (SOA) se define como «conjunto de componentes los cuales pueden ser invocados, y cuyas descripción de las interfaces pueden ser publicadas y descubiertas» [17].

2.4.1.4 Patrón arquitectónico MVC

Según Microsoft, Model-View-Controller (MVC) es un «patrón arquitectónico que divide a la aplicación en tres componentes principales: el modelo, la vista, y el controlador» [18].

- ❑ **Modelo:** «Es la parte de la aplicación que implementada la lógica para el dominio de los datos de la aplicación» [18]. Los datos de la aplicación pueden estar almacenados en una base de datos, archivos, etc.
- ❑ **Vista:** «Es la parte de la aplicación que despliega o muestra la interfaz de usuario de la aplicación» [18].
- ❑ **Controlador:** «Es la parte de la aplicación que maneja las interacciones del usuario, trabaja con el modelo, y finalmente selecciona una vista para ser renderizada y mostrar la interfaz de usuario» [18].

MVC ofrece bajo acoplamiento entre sus componentes, permitiendo que cada uno de estos componentes, se centre en aspectos distintos como la lógica de la interfaz de usuario (vista), la lógica de las entradas o inputs (controlador), y la lógica de negocio (modelo) [18]. Gracias a este bajo acoplamiento, el trabajo se puede paralelizar, permitiendo que exista un desarrollador o grupo de desarrolladores en componentes distintos.

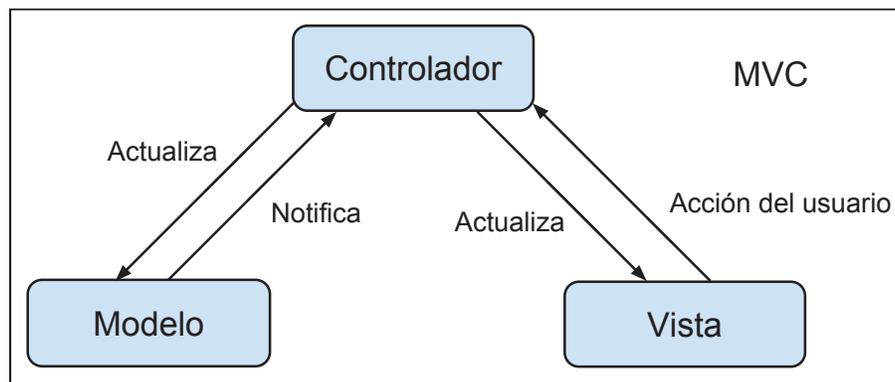


Figura 2.10: Patrón arquitectónico MVC.

2.4.1.5 SPA

Según Microsoft, una Single Page Application (SPA) se define como «aplicaciones web que cargan una sola página HyperText Markup Language (HTML) y actualiza dinámicamente la página a medida que el usuario interactúa con la aplicación» [19].

Para simular la navegación, la aplicación actualiza la barra de navegación a medida que el usuario navega por esta.

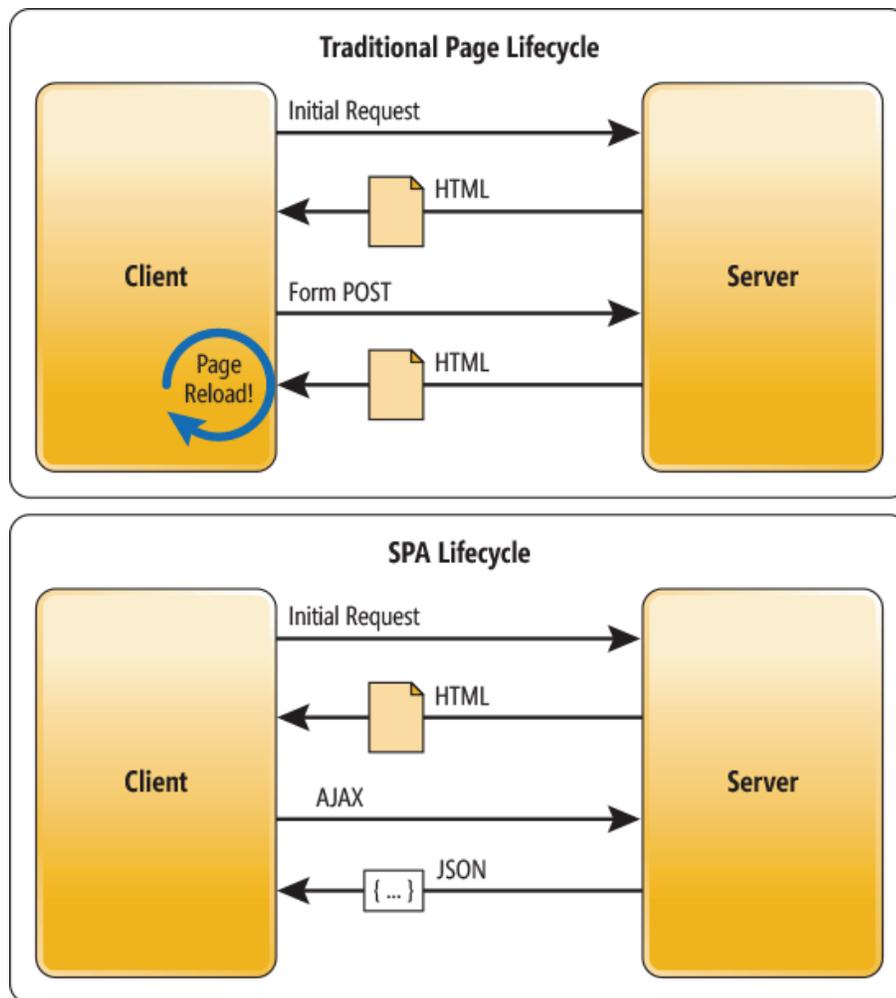


Figura 2.11: Comparativa del ciclo de vida de una SPA y el ciclo de vida de una aplicación web tradicional [19].

En una aplicación web tradicional:

- Por cada solicitud por parte del cliente al servidor web, este envía una página web completa con todos sus elementos (HTML, CSS, CSS).
- La navegación por parte del usuario se ve interrumpida cada vez que este carga una nueva página.

En una SPA:

- ❑ En la primera solicitud se cargan todos los elementos (HTML, CSS, CSS) de las distintas vistas que conforman la aplicación. En las peticiones posteriores, resultantes de la interacción del usuario con la aplicación, se envían por parte del servidor únicamente los datos solicitados.
- ❑ La navegación por parte del usuario no se ve interrumpida cuando se navega de una vista a otra, pero debe esperar a la carga de datos para una correcta visualización de esta.

2.4.1.6 Arquitectura de una aplicación web

De forma general una aplicación web posee una arquitectura cliente-servidor, donde el cliente realiza solicitudes al servidor usando el protocolo HTTP y el servidor responde a estas solicitudes usando el mismo protocolo.

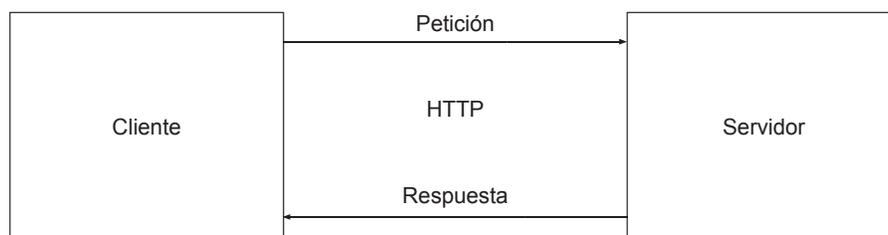


Figura 2.12: Arquitectura general de una aplicación web (Cliente-Servidor).

2.4.2 DISEÑO DE LA ARQUITECTURA DE LA APLICACIÓN

2.4.2.1 Arquitectura diseñada

Para el diseño de la arquitectura se consideraron los conceptos antes mencionados. La arquitectura diseñada se puede observar en la Figura 2.13. Esta se encuentra conformada por tres componentes principales:

- ❑ **Servidor de base de datos:** Almacena los datos de la aplicación web, en estructuras de datos generadas a partir del modelo de datos.

- **Servidor web:** Se encarga de recibir, procesar, y responder a todas las peticiones HTTP del cliente. Se aplicó sobre este servidor, el patrón arquitectónico MVC.
 - ✧ **Modelo(s):** Realiza las operaciones CRUD correspondientes sobre la base de datos.
 - ✧ **Controlador(es):** Publica los servicios web RESTful, y en base al servicio web y la funcionalidad asociada a este, interactúa con el modelo y responde la petición HTTP correspondiente. Adicional, el controlador genera la vista que contiene a la SPA.
 - ✧ **Vista:** contiene a la SPA.

- **Cliente:** Dispositivo que utiliza el usuario final para interactuar con la aplicación web. La interacción la realiza a través del navegador web, donde la primera interacción resulta en la carga de la vista inicial que contiene a la SPA, y las próximas interacciones son realizadas en las vistas proporcionadas por la SPA. Sobre esta se aplica el patrón arquitectónico MVC y queda constituida por los siguientes componentes:
 - ✧ **Vista(s):** Transfiere las interacciones del usuario final al controlador, y muestra los datos del modelo, proporcionados por el controlador, al usuario final.
 - ✧ **Modelo(s):** Son usualmente objetos de datos, que contienen la información que es mostrada al usuario final.
 - ✧ **Controlador(es):** Es el encargado de transformar las interacciones del usuario final sobre la vista, en peticiones HTTP que consumen los servicios web RESTful ofrecidos por el servidor web y procesar las respuestas por parte de este. Adicional, en base a las interacciones del usuario sobre la vista de la SPA, y los servicios web RESTful consumidos y respuestas obtenidas, renderiza o modifica la vista correspondiente de la SPA o actualiza el modelo de esta.

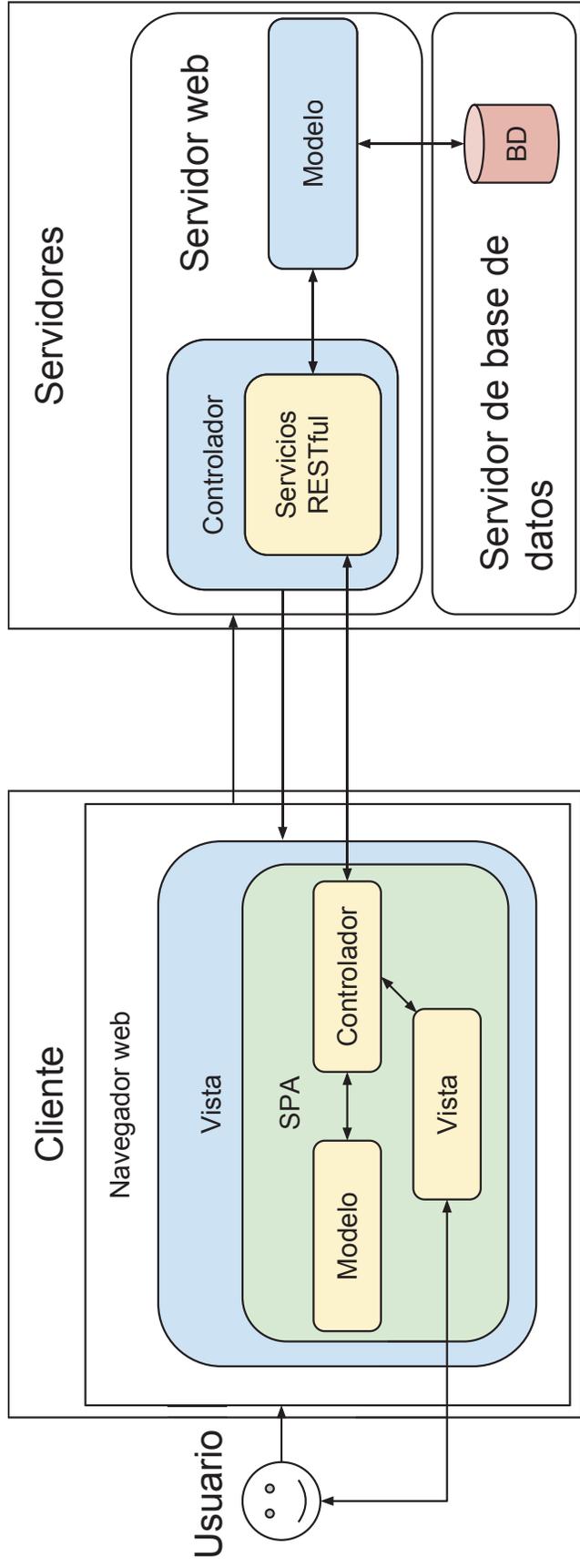


Figura 2.13: Arquitectura de la aplicación a desarrollarse.

2.5 TECNOLOGÍAS A UTILIZAR

2.5.1 LENGUAJES

2.5.1.1 JavaScript

Según Mozilla, JavaScript es definido como «Lenguaje ligero e interpretado, orientado a objetos con funciones de primera clase, más conocido como el lenguaje de script para páginas web, pero también usado en muchos entornos sin navegador» [20]. Anteriormente, JavaScript era usado únicamente a nivel del cliente por el explorador web, hoy en día también es usado a nivel del servidor por servidores como NodeJS.

Las razones por las cuales se eligió este lenguaje de programación:

- Es implementado por todos los navegadores web modernos [20].
- Se puede reutilizar código ya que se utiliza el mismo lenguaje de programación a nivel de cliente y a nivel de servidor.
- Permite implementar funcionalidad a nivel de cliente en páginas web.

2.5.1.2 HTML5

Según la W3C, HTML es definido como «Lenguaje estándar utilizado para crear páginas web y sus elementos forman los bloques de construcción de todas las páginas web» [21]. Su última versión es HTML5.

Las razones por las cuales se eligió este lenguaje:

- Es considerado un lenguaje estándar por la W3C [21].
- Permite definir la estructura de una página web.
- Presenta mejoras y nuevos elementos con respecto a sus versiones anteriores [22].

2.5.1.3 CSS3

Según Mozilla, Cascading Style Sheets (CSS) es definido como «lenguaje utilizado para describir la presentación de documentos HTML o XML» [23].

Las razones por las cuales se eligió este lenguaje:

- Es considerado un lenguaje estándar por la W3C [23].
- Permite definir la apariencia (estilo) de una página web.

2.5.2 FRAMEWORKS

2.5.2.1 Bootstrap

Bootstrap es un framework que utiliza HTML, JavaScript (JS), y CSS para el desarrollo de páginas web adaptables a la pantalla del dispositivo.

Las razones por las cuales se eligió este framework:

- Permite desarrollar páginas web que se adaptan a la pantalla del dispositivo.
- Experiencia que posee el desarrollador utilizando este framework.

2.5.2.2 AngularJS

Según Google, AngularJS es definido como «Framework estructural para aplicaciones web dinámicas» [24]. Este framework es desarrollado por Google.

Las razones por las cuales se eligió este framework:

- Permite generar código HTML dinámicamente.
- Permite aplicar el patrón arquitectónico MVC a nivel del cliente.
- Realiza una vinculación entre el modelo y la vista, cualquier cambio en el modelo se refleja automáticamente en la vista.

- Experiencia que posee el desarrollador utilizando este framework.

2.5.2.3 Sails

Sails es definido como «Framework MVC más popular para NodeJS, diseñado para emular patrones de MVC con soporte para los requerimientos de aplicaciones modernas utilizando una Application Programming Interface (API) escalables basadas en datos y una arquitectura orientada a servicios» [25].

Las razones por las cuales se eligió este framework:

- Tiene implementado el patrón arquitectónico MVC [25].
- Es el framework MVC más popular de NodeJS [25].
- Ofrece una arquitectura basada en servicios [25].
- Experiencia que posee el desarrollador utilizando este framework.
- Como se basa en NodeJS se puede reutilizar en el servidor cualquier código JS desarrollado a nivel de cliente.

2.5.2.4 MochaJS

MochaJS es un framework de JS que permite el desarrollo y ejecución de pruebas utilizando NodeJS.

Las razones por las cuales se eligió este framework:

- Permite el desarrollo de pruebas unitarias.
- Permite el desarrollo de pruebas a servicios web RESTful con la ayuda de librerías como supertest.
- Se requiere un framework para la prueba de las historias de usuario.

2.5.3 ENTORNOS DE EJECUCIÓN

2.5.3.1 NodeJS

NodeJS es definido como «un entorno de ejecución para JS construido con el motor de JavaScript V8 de Chrome» [26].

Las razones por las cuales se eligió este entorno de ejecución:

- ❑ Posee el ecosistema más grande de librerías de código abierto en el mundo [26].
- ❑ Una vez instalado, permite la ejecución de código programado en JS, habilitando con esto el uso de este a nivel del servidor web.
- ❑ Usa un modelo de operaciones E/S sin bloqueo y orientado a eventos [26], permitiendo con esto la ejecución de código asíncrono.
- ❑ Experiencia que posee el desarrollador utilizando este entorno de ejecución.

2.5.4 SERVICIOS EN LA NUBE

2.5.4.1 Google Cloud Platform

Google Cloud Platform es la plataforma de la nube de Google, es el nombre que lleva la plataforma que contiene a todos los servicios ofrecidos por Google relacionados con la Nube.

Las razones por las cuales se eligió esta plataforma de la nube:

- ❑ Se necesita una plataforma de la nube que permita desarrollar la aplicación web de este documento, así se definió en el anexo A.3. La razón principal de esta decisión (usar la nube) era para permitir escalabilidad de la aplicación, y que el desarrollador se preocupe únicamente del desarrollo de la aplicación

mas no de la infraestructura de esta. Esto se consigue gracias a esta plataforma, según Google [27].

- ❑ Ofrece Platform as a Service (PAAS), Infrastructure as a Service (IAAS), Software as a Service (SAAS).
- ❑ Ofrece una variedad de servicios dentro de su nube, entre ellos bases de datos MySQL.
- ❑ Reputación de la empresa Google.

2.5.4.2 Google App Engine

Google App Engine es el PAAS ofrecido por Google para el hosteo de aplicaciones. Según Gartner, PAAS se define como «ofrecimiento de una colección de servicios de infraestructura (middleware) para aplicaciones que incluyen plataforma para aplicaciones, integración, manejo de los procesos de negocio y servicios de base de datos» [28].

Para el despliegue de la aplicación en la nube se contaba con dos opciones: versión IAAS, y versión PAAS de Google. Las razones por la cual se eligió la versión PAAS frente a la versión IAAS de Google se describe a continuación.

- ❑ Utilizar Google Compute Engine, la versión IAAS de Google. Este ofrece instancias (servidores) en las cuales hay que realizar todas las instalaciones y configuraciones necesarias a nivel del sistema operativo, es análogo a una máquina virtual. El manejo de la infraestructura a nivel de hardware es manejado por Google, y a nivel de software por el desarrollador. Se pueden decidir las especificaciones de la instancia manualmente o que escalen automáticamente estas especificaciones.
- ❑ Utilizar Google App Engine, la versión PAAS de Google. Simplemente se aloja la aplicación en la plataforma ofrecido por este, sin preocupaciones del sistema operativo, configuraciones o instalaciones necesarias. El manejo de

la infraestructura tanto a nivel de hardware y software de la instancia, es manejada por Google. Se puede elegir las especificaciones de la instancia ya sea de forma manual o que escalen automáticamente.

Las razones por las cuales se eligió la versión PAAS de Google frente a otras en el mercado:

- Compatibilidad con NodeJS, sobre la cual se implementa el servidor web.
- Reconocimiento de la empresa Google.
- Es parte del ecosistema de Google Cloud Platform.
- Las demás razones mencionadas en la justificación de Google Cloud Platform.

2.5.4.3 Google Cloud SQL

Es la versión de MySQL ofrecida por Google Cloud Platform. Google define a Google Cloud SQL como «servicio de base de datos completamente administrada que permite una fácil configuración, mantenimiento, manejo, y administración de bases de datos relacionales MySQL» [29]. Esta base de datos es ofrecida como servicios a través de una instancia que contiene a esta.

Las razones por las cuales se la eligió este servicio de base de datos:

- Reputación de la empresa proveedora del servicio (Google).
- Facilidad de configuración, mantenimiento, manejo, y administración de la base de datos [29].
- Es la base de datos relacional ofrecida por Google Cloud Platform.

2.5.5 BASE DE DATOS

2.5.5.1 MySQL

Según Oracle, MySQL es «la base de datos de código abierto más popular del mundo» [30]. MySQL es ofrecido como servicio en Google Cloud Platform.

Las razones por las cuales se eligió esta base de datos son:

- Es una base de datos relacional, por consiguiente se puede manejar el concepto de Tablas y entidades, y utilizar álgebra relacional sobre esta.
- Es la base de datos de código abierto más utilizada del mundo.
- Posee versiones gratuitas como MySQL Community Edition.
- Experiencia del desarrollador utilizando MySQL y desarrollando base de datos relacionales.

2.5.6 APIS

2.5.6.1 Google Speech Synthesis

Google Speech Synthesis es una API de Google que se encarga de la síntesis del habla, es decir, de transformar el texto a voz. Esta API es parte de la API Web Speech de Google. Puede ser consumida con soporte total en el navegador de Google Chrome versión 33 o más y parcialmente en el navegador Safari para iOS7 [31].

Las razones por las cuales se eligió esta API son:

- Se necesitaba un componente que se encargue de transformar el texto a voz.
- Reputación de la empresa desarrolladora (Google) de la API.
- Es totalmente gratis.

- Disponible localmente (sin necesidad de uso de Internet) el idioma inglés, y utilizando Internet una variedad de idiomas, incluido el español.
- Permite la configuración de varios parámetros como la velocidad de la voz, acento, etc.

2.5.6.2 Google Speech Recognition

Google Speech Recognition es una API de Google que se encarga del reconocimiento de voz, es decir, de transformar la voz a texto. Esta API es parte de la API Web Speech de Google. Al igual que el sintetizador de voz de Google, puede ser consumida con soporte total en el navegador de Google Chrome versión 33 o más y parcialmente en el navegador Safari para iOS7 [31].

Las razones por las cuales se eligió esta API son:

- Permite transformar la voz a texto, lo que es usado para el inicio de sesión en la aplicación.
- Es totalmente gratis.
- Reputación de la empresa desarrolladora (Google) de la API.

2.5.7 HERRAMIENTAS

2.5.7.1 Atom

Atom es un editor de texto de código abierto configurable, con soporte para varios lenguajes de programación, desarrollado bajo la licencia MIT.

Las razones por las cuales se eligió este editor de texto son:

- Posee una gran variedad de paquetes que permiten su configuración y personalización en base a los gustos del desarrollador.

- Existen versiones para distintos sistemas operativos como Windows, Linux, y OS X.
- Experiencia de uso por parte del desarrollador.

2.5.8 REPOSITARIOS DE CÓDIGO

2.5.8.1 GitHub

GitHub es una plataforma que ofrece un repositorio de código para el manejo del versionamiento de este y trabajo colaborativo, basado en Git.

Las razones por las cuales se eligió esta plataforma son:

- Se requería de un repositorio de código que permita el versionamiento de este, debido a las ventajas que estos repositorios ofrecen.
- Ofrece un cliente con una interfaz gráfica amigable y fácil de usar.
- Posee una versión gratuita.
- Experiencia del desarrollador utilizando esta plataforma.

2.6 MAPEO DE LAS TECNOLOGÍAS A UTILIZAR CON LA ARQUITECTURA DE LA APLICACIÓN

Como se muestra en la Figura 2.13, la arquitectura de la aplicación desarrollada está dividida en cliente y servidores. El mapeo de las tecnologías antes mencionadas con la arquitectura de la aplicación se muestra en la Figura 2.14, y se describe a continuación.

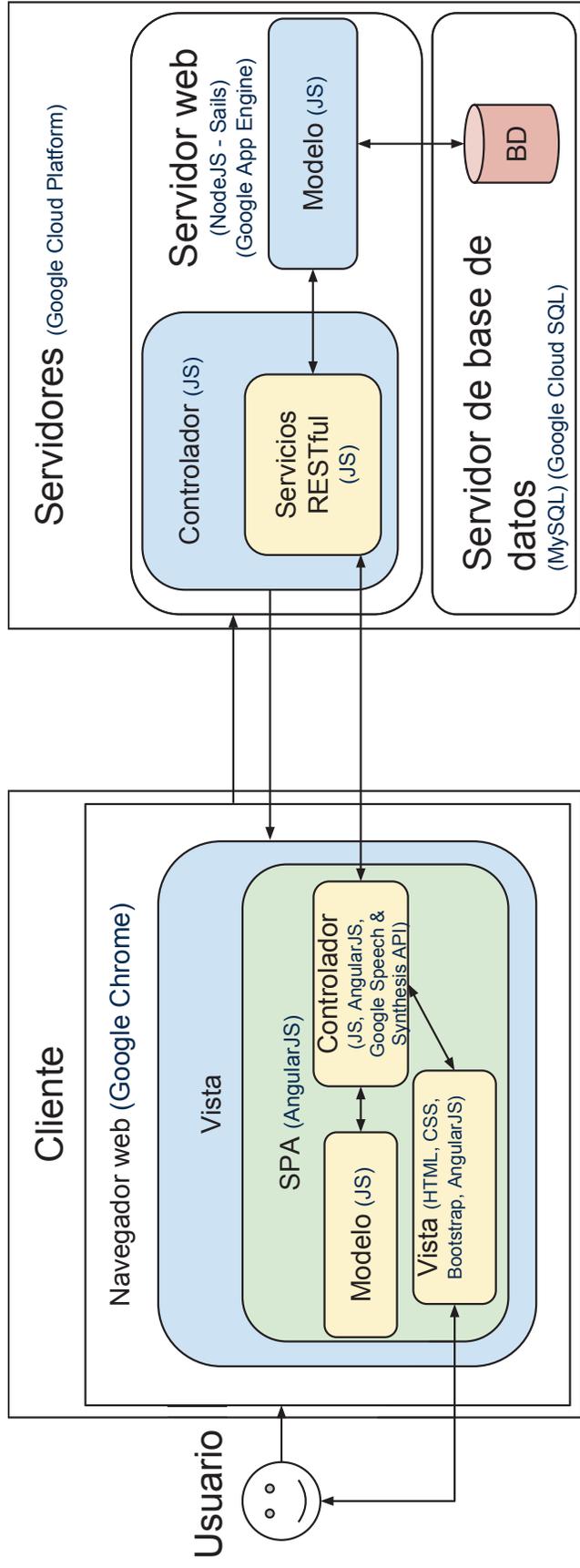


Figura 2.14: Mapeo de las tecnologías utilizadas con la arquitectura de la aplicación.

A nivel de:

❑ Servidor web

- ❖ La infraestructura requerida por este servidor dada por Google App Engine.
- ❖ NodeJS es el entorno de ejecución del servidor web.
- ❖ Sobre el entorno de ejecución se encuentra Sails, que es el framework sobre el cual se desarrolla la aplicación web considerando el patrón arquitectónico MVC y la arquitectura basada en servicios que ofrece este.
- ❖ La aplicación web en el servidor web, incluyendo los servicios web RESTful, controladores y modelos, es programada utilizando JS.
- ❖ Los servicios web RESTful se prueban utilizando las pruebas desarrolladas en JS en el framework de MochaJS.

❑ Servidor de base de datos

- ❖ La infraestructura requerida por este servidor viene dada por Google Cloud SQL.
- ❖ Google Cloud SQL implementa como Database Management System (DBMS) a MySQL.
- ❖ El acceso a la base de datos por parte del servidor web, se lo realiza utilizando sockets de Linux ofrecidos por Google Cloud SQL.

❑ **Cliente:**

- ❖ La vista generada por el servidor web la primera vez que se accede a la aplicación, es una SPA. Esta se desarrolla utilizando AngularJS, que permite separar a la SPA en modelos, vistas y controladores.
- ❖ Los controladores están programados utilizando JS como lenguaje de programación, en conjunto con el framework de AngularJS. En estos se consumen los servicios web RESTful ofrecidos por el servidor web, utilizando AngularJS. Adicional, el consumo de las APIs Google Speech

Recognition y Google Speech Synthesis son realizadas en estos controladores.

- ✧ Las vistas están desarrolladas utilizando: HTML para definir la estructura de la vista, CSS el estilo de estas, AngularJS para la generación de HTML dinámicamente, y Bootstrap para permitir la adaptabilidad de la vista a la pantalla del dispositivo que esta usando el usuario.
- ✧ Los modelos son representados utilizando la notación JavaScript Object Notation (JSON) en el controlador, y enlazados a la vista gracias a AngularJS.

2.7 REQUISITOS DE LA APLICACIÓN

A continuación, se presentan los requisitos tanto a nivel de hardware y software requeridos por la aplicación.

2.7.1 CLIENTE

2.7.1.1 Hardware

- Personal Computer (PC) con las características suficientes de procesador, memoria y disco para soportar el sistema operativo de elección.
- Parlantes o audífonos.
- Micrófono (opcional).
- Teclado.
- Ratón (opcional).
- Acceso a Internet constante, sin intermitencias, para la mejor experiencia de usuario posible.

2.7.1.2 Software

- Cualquier sistema operativo que soporte Google Chrome 33 o superior.
- Google Chrome 33 o superior.

2.7.2 SERVIDOR WEB

2.7.2.1 Hardware

- Toda la infraestructura del servidor viene proveída por Google App Engine. Esta infraestructura escala en base a la cantidad de peticiones.

2.7.2.2 Software

- Google App Engine ofrece todos los requisitos a nivel de software requeridos por el servidor, ya que este ofrece a NodeJS, el cual es requerido por todos los módulos del framework Sails y de la aplicación web.

2.7.3 SERVIDOR DE BASE DE DATOS

2.7.3.1 Hardware

- Toda la infraestructura del servidor viene proveída por Google Cloud SQL. Esta infraestructura escala en base a la cantidad de peticiones.

2.7.3.2 Software

- Google Cloud SQL provee de todos los requisitos de software requeridos por el servidor, incluido MySQL.

2.8 METODOLOGÍA DE DESARROLLO

2.8.1 JUSTIFICACIÓN

Para seleccionar la metodología de desarrollo a utilizar primero se necesitaba decidir entre un enfoque tradicional o ágil para el desarrollo de software. Se decidió utilizar un enfoque ágil por lo siguiente:

- La solución planteada es una aplicación web. Interesaba en mayor medida desarrollar una aplicación de calidad, que una documentación extensa relacionada a esta. Para lograr lo mencionado se debe:
 - ✧ Dedicar la mayor cantidad de los recursos posibles al desarrollo de las funcionalidades de la aplicación, mas no a su documentación.
 - ✧ Ver la calidad del producto, y eso se logra a través de las funcionalidades de este.
 - ✧ Generar documentos de calidad concretos que van a producir productos de calidad.

Dos de los doce principios de software ágil reflejan esto. Estos principios son:

- ✧ «El software funcionando es la medida principal de progreso» [32].
- ✧ «La simplicidad -arte de maximizar la cantidad de trabajo no realizado- es esencial» [32].

El manifiesto ágil refleja esto a través de:

- ✧ «Software funcionando sobre documentación extensiva» [33].

- Se tenía planeado realizar pruebas a distintas versiones de la aplicación para realizar los cambios que satisfagan en mayor medida, la necesidad de las personas no videntes a rendir pruebas, es decir los requerimientos iban a cambiar.

Uno de los doce principios de software ágil refleja esto. Este principio es:

- ✧ «Recibir cambios en los requerimientos, inclusive en etapas tardías del desarrollo» [32].

El manifiesto ágil refleja esto a través de:

- ✧ «Responder a cambios sobre seguir un plan» [33].

Una vez seleccionado el enfoque, el siguiente paso era seleccionar la metodología de desarrollo. Entre las principales metodologías de desarrollo ágil se tenían [32]:

- Scrum.
- Extreme Programming (XP).

Las razones por las cuales se eligió Scrum sobre XP son las siguientes:

- Scrum es más utilizada que XP [34].
- El desarrollador de la aplicación posee más conocimientos y experiencia trabajando con Scrum que con XP.
- Es posible, con ambas metodologías, abordar el problema de desarrollar una aplicación web que permita rendir pruebas a las personas no videntes.

Existe controversia entre si Scrum es o no una metodología. Algunos definen a Scrum como metodología [32], y otros como un framework [35]. Los coautores de Scrum, lo definen como «framework dentro del cual las personas pueden abordar problemas adaptativos complejos, mientras que productivamente y creativamente se entregan productos del mayor valor posible» [35].

Para propósitos de este documento, es indistinto si Scrum es una metodología de desarrollo o framework. Scrum define un proceso a seguir, con roles y artefactos, y considerando la definición de Scrum mencionada anteriormente, es posible concluir que Scrum permite abarcar el problema mencionado en este documento y es lo que realmente importa.

2.8.2 DESCRIPCIÓN

Scrum está formado por un conjunto de elementos: roles, artefactos, eventos y reglas. Las reglas definen la interacción y relación entre los demás elementos [35]. Esto se muestra en la Figura 2.15.

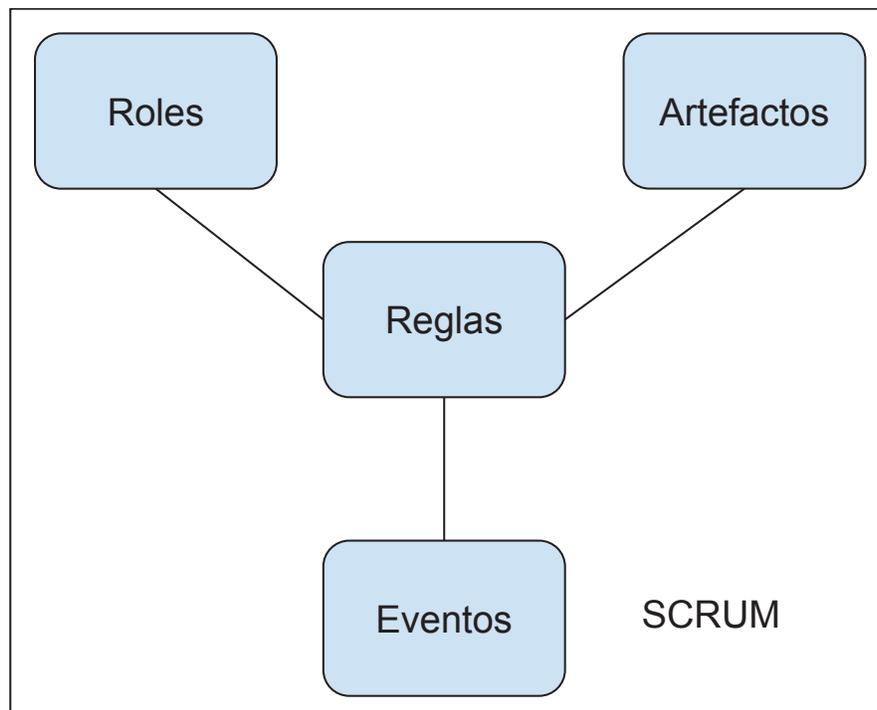


Figura 2.15: Elementos que conforman Scrum y sus interacciones.

2.8.2.1 Roles

El Scrum Team (equipo Scrum) está conformado por tres roles [35], ver Figura 2.16. Estos son:

- ❑ **Product Owner:** Rol responsable de maximizar el valor de un producto, mediante el manejo incremental y expresando las expectativas funcionales y de negocio de un producto hacia el Development Team [36].
 - ✧ Solamente este rol puede administrar el Product Backlog [35].
 - ✧ Es responsable y rinde cuentas del Product Backlog [35].

- ✧ Este rol es adoptado por una sola persona, más no por un comité [35].
 - ✧ Para el éxito de este rol, la organización debe respetar las decisiones tomadas por este [35].
- **Scrum Master:** Rol responsable de guiar, enseñar y asistir al Scrum Team de lograr un entendimiento correcto del uso de Scrum [36].
- ✧ Asegura que el Scrum Team se mantenga bajo las reglas y teoría de Scrum [35].
 - ✧ Ayuda a las personas fuera del Scrum Team a entender Scrum [35].
 - ✧ Es un pseudo-lider del Scrum Team [35].
- **Development Team (equipo desarrollador):** Rol responsable de administrar, organizar y realizar todo el trabajo requerido para crear un Incremento del producto cada Sprint [36].
- ✧ Solo este rol se encarga de realizar un Incremento [35].
 - ✧ Son autorizados por la organización para organizar y administrar su propio trabajo [35].
 - ✧ El Scrum Team es lo suficientemente pequeño para permanecer ágil y lo suficientemente grande como para lograr un trabajo significativo en el Sprint [35].

2.8.2.2 Eventos

- **Sprint:** Evento de duración de treinta días, o menos, en el cual un Incremento es creado, y sirve como contenedor para otros eventos y actividades Scrum. Los Sprints deben ser ejecutados consecutivamente sin intermitencias [36].
- ✧ Está conformado por Sprint Planning, Daily Scrums, el trabajo realizado, Sprint Review, y el Sprint Retrospective (retrospectiva del Sprint) [35].
 - ✧ Cada Sprint puede ser considerado un proyecto [35].

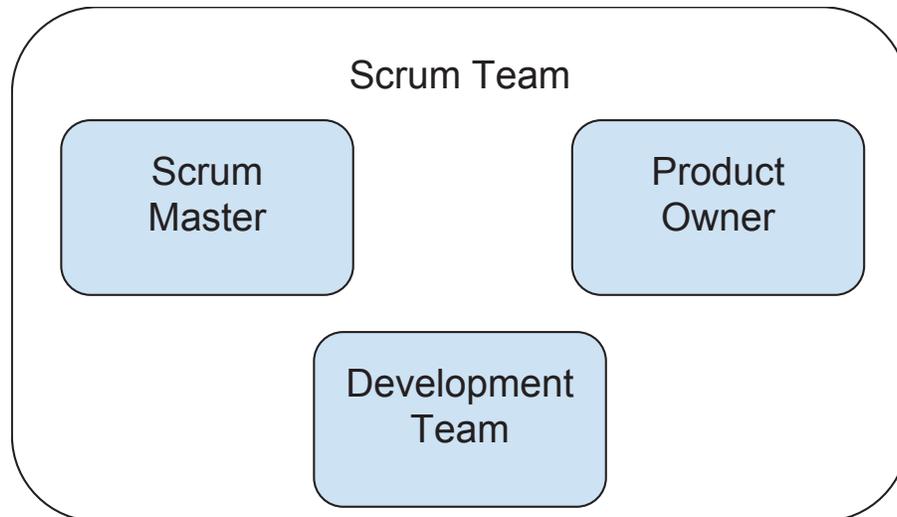


Figura 2.16: Roles que conforman el Scrum Team.

- ✧ Un Sprint es ejecutado inmediatamente después de la finalización de otro [35].
 - ✧ No se realizan cambios que pueden poner en peligro el Sprint Goal (meta del Sprint) [35].
 - ✧ Un Sprint puede ser cancelado antes del tiempo establecido para este, y solo lo puede cancelar el Product Owner [35].
- **Sprint Planning:** Evento de duración de ocho horas, o menos, el cual da comienzo a un Sprint. Se inspecciona el trabajo del Product Backlog que es el más importante para incluirlo en el Sprint Backlog [36].
- ✧ El Scrum Master se asegura que los eventos tengan lugar, y los que atienden a estos eventos entiendan su rol [35].
 - ✧ Se define en el Sprint Planning que Incremento puede ser entregado, y como el trabajo necesario para entregar este Incremento es alcanzado [35].
 - ✧ El objetivo que se desea alcanzar en el Sprint Planning, a través de la implementación de los ítem del Product Backlog seleccionados, es lo que se conoce como Sprint Goal [35].
- **Daily Scrum:** Evento de duración de quince minutos, o menos, para sincroni-

zar y planear, por parte del Development Team, el siguiente día de trabajo de desarrollo durante el Sprint [36]. Si existen cambios, estos deben reflejarse en el Sprint Backlog.

- ✧ El Development Team utiliza el Daily Scrum para inspeccionar el progreso hacia el Sprint Goal [35].
- ✧ El Scrum Master se asegura que el Daily Scrum se ejecute, y que solamente el Development Team sea el participante en esta [35].
- ✧ Toma lugar todos los días a la misma hora y en el mismo lugar para reducir la complejidad [35].

□ **Sprint Review:** Evento de duración de cuatro horas, o menos, realizado al final del Sprint para inspeccionar el Incremento y adaptar el Product Backlog de ser necesario [35].

- ✧ Participan el Scrum Team y los interesados (stakeholders) claves invitados por el Product Owner [35].
- ✧ El Development Team discute que se realizó con éxito y que problemas tuvieron [35].
- ✧ El Development Team demuestra el Incremento realizado y responde a las preguntas que se realicen acerca de este.
- ✧ El Product Owner explica que elementos del Product Backlog se completaron y cuales no.

□ **Sprint Retrospective:** Evento de duración de tres horas, o menos, realizado después del Sprint Review y antes del Sprint Planning, para inspeccionar el Sprint anterior y desarrollar un plan de mejoras de ser necesario [36].

- ✧ El Scrum Master se asegura que el evento sea ejecutado, y que cada participante del evento entienda el propósito de este [35].
- ✧ Participa todo el Scrum Team [35].

2.8.2.3 Artefactos

- ❑ **Product Backlog:** Lista ordenada de todo lo que pueda ser necesitado por el producto, es decir, todos los requerimientos [35].
 - ✧ El Product Owner es el responsable del contenido y ordenamiento del Product Backlog [35].
 - ✧ Representa el trabajo futuro a realizarse [35].
 - ✧ Puede cambiar [35].

- ❑ **Herramientas para evaluar el progreso hacia la meta:** Conjunto de herramientas utilizadas para evaluar el trabajo restante hacia la meta [35].
 - ✧ Dependiendo del proyecto y las prácticas aplicadas a este, las herramientas a utilizar pueden variar. Algunos ejemplos de estas son gráficas Burn-Down, Burn-UP, y flujos acumulativos [35].

- ❑ **Sprint Backlog:** Subconjunto de ítems del Product Backlog, que contienen los requerimientos a ser implementados y generar un Incremento.
 - ✧ Solamente el Development Team puede cambiar su Sprint Backlog durante el Sprint [35].
 - ✧ Si se presenta nuevo trabajo, este debe agregarse al Sprint Backlog por el Development Team [35].
 - ✧ El Sprint Backlog refleja en tiempo real, el trabajo que se está realizando y se espera cumplir en el Sprint [35].

- ❑ **Incremento:** Es la suma de todos los ítems del Product Backlog implementados durante un Sprint agregados a los Incrementos existentes [35], [36].
 - ✧ El Product Owner es el encargado de decidir si se libera o no el Incremento [35].
 - ✧ La suma de todos los Incrementos forman un producto [36].

2.8.2.4 Reglas

En la guía de Scrum [35] no describen estas reglas, puesto que, en este documento y la guía de Scrum, las reglas son explicadas en la descripción de los demás elementos que conforman Scrum (roles, artefactos y eventos).

CAPÍTULO 3

DESARROLLO DE LA SOLUCIÓN

Este capítulo tiene como objetivo mostrar las tareas realizadas para poder desarrollar la aplicación web planteada. Se responderán las preguntas como:

- ❑ ¿Qué historias de usuario se realizaron? Ver anexo E.1.
- ❑ ¿Cómo se distribuyó el trabajo realizado? Ver los Sprint Backlogs presentes en el anexo E.
- ❑ ¿Cuánto tiempo y esfuerzo tomó desarrollar la aplicación? Ver los Sprint Review presentes en este capítulo.

3.1 GENERACIÓN DEL PRODUCT BACKLOG

El Product Backlog generado se muestra en la Figura E.1. Para la generación de este se tomaron las siguientes consideraciones:

- ❑ Se presentan todos los requerimientos, incluidos los que surgieron al momento del desarrollo de la aplicación, esto para presentar de forma más clara todas las funcionalidades que presenta la aplicación desarrollada.
- ❑ Los ítems del Product Backlog se presentan en el orden en el que fueron desarrollados, basado en su importancia.
- ❑ El desarrollador representa al autor de este documento y desarrollador de la aplicación.

- ❑ El plan de pruebas está conformado por:
 - ❖ 163 pruebas unitarias a todos los servicios HTTP RESTful desarrollados. Estas se realizaron al final de cada Sprint, resultando: 40 en el Sprint 3, 29 en el Sprint 4, 82 en el Sprint 5 y 12 en el Sprint 6. Estas se encuentran en el Product Backlog con los códigos TEx. Los resultados obtenidos se muestran en el anexo D.1.
 - ❖ Pruebas de calidad del código fuente, las cuales fueron realizadas al final del último Sprint. Estas se muestran en el anexo D.2.
 - ❖ Resultados obtenidos al aplicar la norma ISO/IEC 25000 con respecto a la calidad externa, característica facilidad de uso. Estos se muestran en el anexo D.3.

- ❑ La importancia viene dada por un número que puede tomar valores de 0-99, siendo 0 lo más importante y 99 lo menos importante.

- ❑ Las historias de usuario se encuentran en el Product Backlog en el anexo E.1. Estas vienen definidas por las columnas *Como*, *Yo quiero*, y *con el fin de*.

- ❑ Las historias de usuario con ID:
 - ❖ PRx hacen referencia al prototipo funcional.
 - ❖ APx hacen referencia a los roles.
 - ❖ BDx hacen referencia a la base de datos.
 - ❖ CLx hacen referencia a Google Cloud.
 - ❖ LOx hacen referencia al ambiente de desarrollo.
 - ❖ MUx hacen referencia al módulo usuario.
 - ❖ TEx hacen referencia a las pruebas unitarias realizadas.
 - ❖ TCx hacen referencia a las pruebas de la calidad del código fuente.
 - ❖ Tlx hacen referencia a la aplicación de la norma ISO/IEC 2500.
 - ❖ TSx hacen referencia a las tareas necesaria para generar el documento JSDoc.

- ✧ MCx hacen referencia al módulo curso.
- ✧ MPx hacen referencia al módulo prueba.
- ✧ MRX hacen referencia al módulo pregunta.
- ✧ MAx hacen referencia a los manuales.
- ✧ ADx hacen referencia a historias de usuarios adicionales que no entran en las categorías anteriores.

3.2 DESIGNACIÓN DE ROLES SCRUM

La designación de los roles Scrum se muestra en la Tabla 3.1.

Tabla 3.1: Designación de roles Scrum.

Rol	Persona
Scrum Master	Farid Saud
Product Owner	Andrés Larco
Development Team	Farid Saud

3.3 DEFINICIÓN DE ROLES DE LA APLICACIÓN

La aplicación desarrollada cuenta con un total de dos roles:

- ❑ **Profesor:** Es el análogo de un profesor real, y tiene acceso a las funcionalidades definidas en el Product Backlog, donde el actor es un Profesor.
- ❑ **Estudiante:** Es el análogo de un estudiante real, y tiene acceso a las funcionalidades definidas en el Product Backlog, donde el actor es un Estudiante.

3.4 CONSIDERACIONES

Para la asignación de las historias de usuario en los distintos Sprints y el desarrollo de este, se consideró lo siguiente:

- ❑ La duración del Sprint es de dos semanas.

- ❑ El Development Team, cuenta con ocho horas diarias no fijas, es decir, pueden existir días en los que se trabaje más o menos.

- ❑ En cada Sprint, de ser posible, desarrollar un módulo completo.

- ❑ En el Sprint Backlog, las historias de usuario deben ser divididas en tareas más pequeñas, estas deben tener el mayor detalle posible. Esto con la finalidad de que el Sprint Backlog sea autónomo y posea el detalle suficiente para su realización.

- ❑ No se considera el evento Daily Scrum, debido a que los asistentes a este evento, Scrum Master y Development Team, son la misma persona (ver Tabla 3.1).

- ❑ La herramienta para evaluar el progreso hacia la meta (uno de los artefactos de Scrum) seleccionada fue Burn-Up debido a que permite observar la progresión de los recursos invertidos. Adicional, esta permite al final del Sprint verificar si el tiempo estimado fue mayor, menor o igual al realmente invertido.

- ❑ En el Sprint Backlog se debe incluir el tiempo real invertido por cada tarea, debido a que al momento de la redacción de este documento ya se cuenta con esta información, y así se evita repetir la misma tarea en varios Sprint Backlog.

- ❑ En la ejecución de los Sprints que se presenta a continuación, no se detalla el funcionamiento de las distintas vistas, esto se detalla en el manual de usuario en el anexo A.6. En la ejecución se muestra en que parte del código es realizada cada tarea.

3.5 SPRINT 1

3.5.1 SPRINT PLANNING

3.5.1.1 Sprint Goal

El objetivo del Sprint es desarrollar un prototipo funcional para la toma de pruebas, que permita obtener la retroalimentación por parte de las personas asistentes al evento Habitat-EPN que utilicen a este.

3.5.1.2 Sprint Backlog

El Sprint Backlog 1 se presenta en la Tabla E.8, y está constituido por la historia de usuario con ID PR1.

3.5.2 EJECUCIÓN DEL SPRINT

Las tareas realizadas en este Sprint fueron:

□ PR1-1:

- ✧ La interfaz se puede apreciar en las Figuras F.1 y F.2.
- ✧ El código HTML se encuentra en el archivo *./assets/html/angular/views/Prototype/test.html* en el anexo B.1.

□ PR1-2:

- ✧ El código JS se encuentra en el archivo *./assets/js/angular/controllers/Prototype/testController.js* en el anexo B.1.

□ PR1-3:

- ✧ El código JS se encuentra en el archivo *./assets/js/angular/controllers/Prototype/testController.js* en el anexo B.1.

❑ PR1-4:

- ❖ El código JS se encuentra en el archivo *./assets/js/angular/controllers/Prototype/testController.js* en el anexo B.1.

❑ PR1-5:

- ❖ El código JS se encuentra en el archivo *./assets/js/angular/controllers/Prototype/testController.js* en el anexo B.1.

❑ PR1-6:

- ❖ La encuesta desarrollada en conjunto con sus resultados se encuentran en el anexo A.4.

❑ PR1-7:

- ❖ La interfaz se puede apreciar en las Figuras F.3 y F.4.
- ❖ El código HTML se encuentra en el archivo *./assets/html/angular/views/Prototype/review.html* en el anexo B.1.

❑ PR1-8:

- ❖ El código JS se encuentra en el archivo *./assets/js/angular/controllers/Prototype/reviewController.js* en el anexo B.1.

❑ PR1-9:

- ❖ El código JS se encuentra en el archivo *./assets/js/angular/controllers/Prototype/reviewController.js* en el anexo B.1.

La Figura 3.1 representa la gráfica Burn-Up de este Sprint. Esta refleja el tiempo invertido acumulado durante todos los días del Sprint.

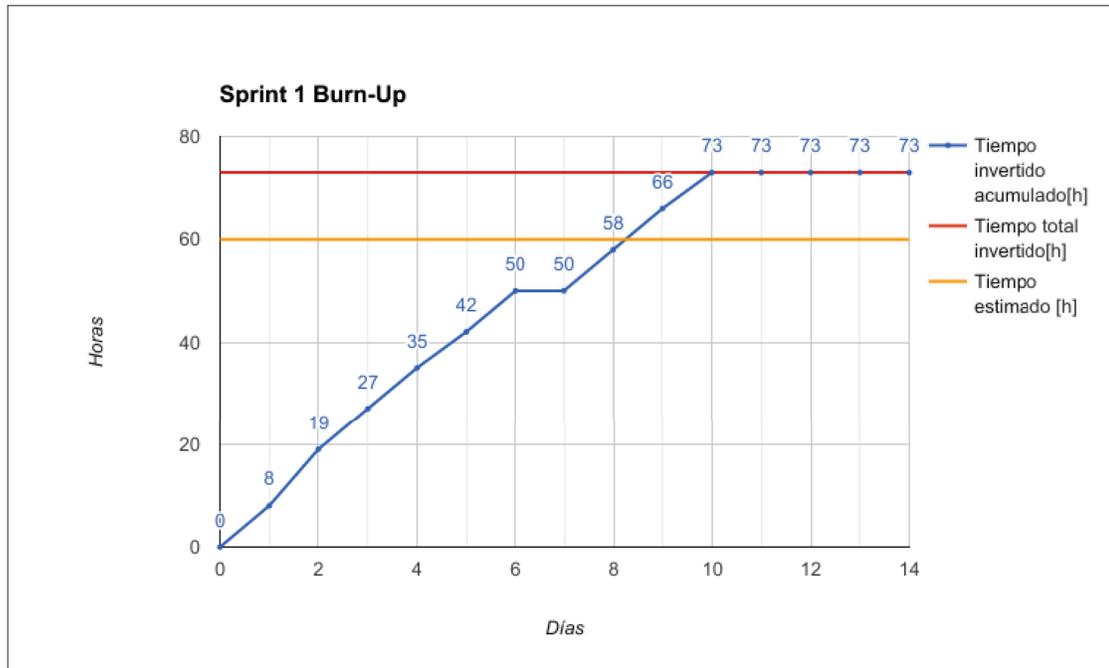


Figura 3.1: Burn-Up Sprint 1.

3.5.3 SPRINT REVIEW

Se completaron la totalidad de las historias de usuario y sus tareas con un total de 73 horas invertidas en el Sprint, esto se aprecia en la Tabla E.9.

3.5.4 SPRINT RETROSPECTIVE

El tiempo total invertido durante el Sprint fue superior al tiempo total estimado, existiendo una diferencia de 13 horas. Se presentaron dificultades que requirieron más tiempo en las tareas: PR1-1, PR1-2, PR1-3, PR1-5, PR1-8 y PR1-9.

3.6 SPRINT 2

3.6.1 SPRINT PLANNING

3.6.1.1 Sprint Goal

El objetivo del Sprint es familiarizarse con los elementos tecnológicos ofrecidos por Google Cloud Platform y dejar preparado las bases del proyecto para el desarrollo y despliegue en los ambientes respectivos.

3.6.1.2 Sprint Backlog

El Sprint Backlog 2 se presenta en la Tabla E.10, y está constituido por las historias de usuario con ID: BD1, BD2, CL1, CL2, CL3 y LO1.

3.6.2 EJECUCIÓN DEL SPRINT

Las tareas realizadas en este Sprint fueron:

☐ BD1-1:

- ✧ El modelo conceptual de datos generado se puede observar en el anexo C.1.

☐ BD1-2:

- ✧ El modelo físico de datos generado se puede observar en el anexo C.2.

☐ BD2-1:

- ✧ El script de creación de la base de datos se puede observar en el anexo C.3.

☐ CL1-1:

✧ Esta tarea no genera entregable alguno.

□ **CL2-1:**

✧ La librería necesaria para realizar el manejo de todos los proyectos de Google Cloud SQL, incluyendo el despliegue en Google App Engine, es el SDK de Google Cloud, el cual se encuentra disponible en la siguiente url: <https://cloud.google.com/sdk/>.

□ **CL2-2:**

✧ Los pasos a seguir para crear un proyecto en Google App Engine se pueden apreciar en el anexo A.7.

□ **CL2-3:**

✧ Los pasos a seguir para crear una instancia de base de datos en Google Cloud SQL se pueden apreciar en el anexo A.7.

□ **LO1-1:**

✧ Esta tarea no genera entregable alguno, las librerías necesarias ya se encontraban instaladas previamente.

La Figura 3.2 representa la gráfica Burn-Up de este Sprint. Esta refleja el tiempo invertido acumulado durante todos los días del Sprint.

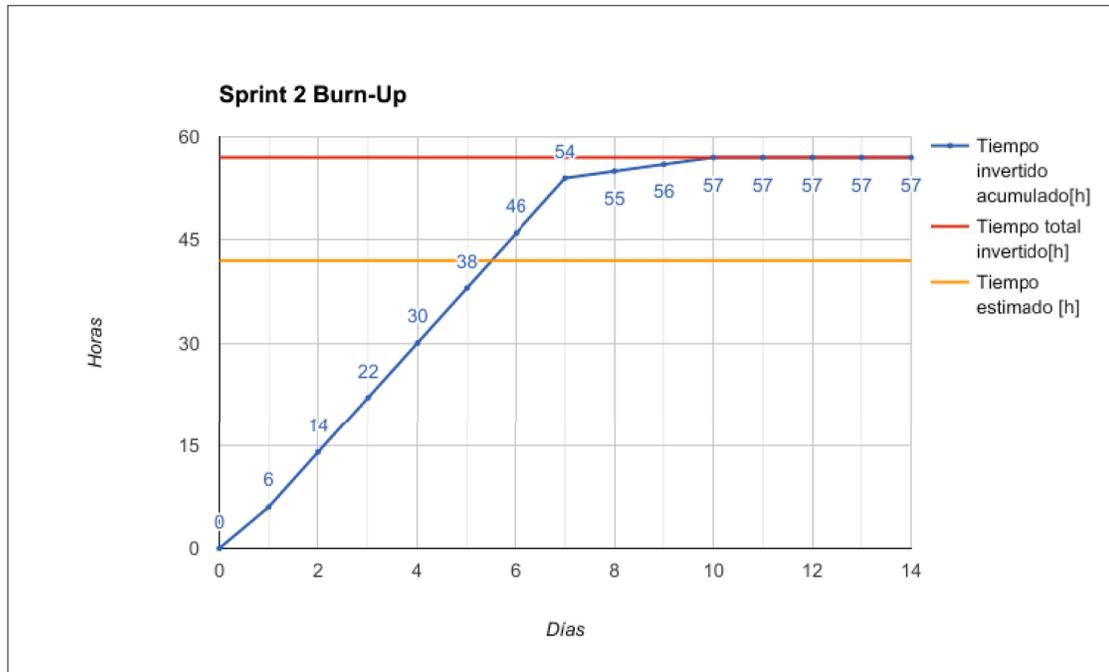


Figura 3.2: Burn-Up Sprint 2.

3.6.3 SPRINT REVIEW

Se completaron la totalidad de las historias de usuario y sus tareas con un total de 57 horas invertidas en el Sprint, esto se aprecia en la Tabla E.11.

3.6.4 SPRINT RETROSPECTIVE

El tiempo total invertido durante el Sprint fue superior al tiempo total estimado, existiendo una diferencia de 15 horas. Se presentaron dificultades que requirieron más tiempo en las tareas: BD1-1, CL1-1 y CL2-1.

3.7 SPRINT 3

3.7.1 SPRINT PLANNING

3.7.1.1 Sprint Goal

El objetivo del Sprint es desarrollar todas las funcionalidades del módulo usuario, y asegurar la calidad de estas con sus respectivas pruebas.

3.7.1.2 Sprint Backlog

El Sprint Backlog 3 se presenta en la Tabla E.12, y está constituido por las historias de usuario con ID: MU1, MU2, MU3, MU4, MU5, MU6 y TE1.

3.7.2 EJECUCIÓN DEL SPRINT

Las tareas realizadas en este Sprint fueron:

❑ MU1-1:

- ✧ La interfaz se puede apreciar en las Figuras F.5 y F.6.
- ✧ El código HTML se encuentra en el archivo `./assets/html/angular/views/User/register.html` en el anexo B.2.

❑ MU1-2:

- ✧ El código JS se encuentra en la función `register` del archivo `./api/controllers/UserController.js` en el anexo B.2.

❑ MU1-3:

- ✧ El código JS se encuentra en la función `register` del archivo `./api/controllers/UserController.js` en el anexo B.2.

❑ MU1-4:

- ✧ El algoritmo de encriptación utilizado es Bcrypt, debido a que ya existía una implementación en NodeJS, y Open Web Application Security Project (OWASP) lo considera entre los mejores algoritmos para la encriptación en una vía de las claves [37].

❑ **MU1-5:**

- ✧ El código JS se encuentra en la función *register* del archivo *./api/controllers/UserController.js* en el anexo B.2.

❑ **MU1-6:**

- ✧ El código JS se encuentra en la función *register* del archivo *./api/controllers/UserController.js* en el anexo B.2.

❑ **MU1-7:**

- ✧ El código JS se encuentra en el archivo *./config/routes.js* en el anexo B.2.
- ✧ El servicio se encuentra publicado en la ruta */user/register* con el método HTTP post.

❑ **MU1-8:**

- ✧ El código JS se encuentra en el archivo *./assets/js/angular/controllers/User/register.js* en el anexo B.2.

❑ **MU1-9:**

- ✧ El código JS se encuentra en el archivo *./assets/js/angular/controllers/User/register.js* en el anexo B.2.

❑ **MU1-10:**

- ✧ El código JS se encuentra en el archivo *./assets/js/angular/controllers/User/register.js* en el anexo B.2.

❑ **MU2-1:**

- ✧ La interfaz se puede apreciar en la Figura F.7.

- ✧ El código HTML se encuentra en el archivo `./assets/html/angular/views/home.html` en el anexo B.2.

❑ **MU2-2:**

- ✧ El código JS se encuentra en la función `login` del archivo `./api/controllers/UserController.js` en el anexo B.2.

❑ **MU2-3:**

- ✧ El código JS se encuentra en la función `login` del archivo `./api/controllers/UserController.js` en el anexo B.2.

❑ **MU2-4:**

- ✧ El código JS se encuentra en el archivo `./config/routes.js` en el anexo B.2.
- ✧ El servicio se encuentra publicado en la ruta `/login` con el método HTTP `post`.

❑ **MU2-5:**

- ✧ El código JS se encuentra en el archivo `./assets/js/angular/controllers/home.js` en el anexo B.2.

❑ **MU2-6:**

- ✧ El código JS se encuentra en el archivo `./assets/js/angular/controllers/home.js` en el anexo B.2.

❑ **MU2-7:**

- ✧ El código JS se encuentra en el archivo `./assets/js/angular/controllers/home.js` en el anexo B.2.

❑ **MU3-1:**

- ✧ Las interfaces gráficas se pueden apreciar en las Figuras F.8 y F.9.
- ✧ El código HTML se encuentra en el archivo `./assets/html/angular/views/home.html` en el anexo B.2.

❑ MU3-2:

- ❖ El código JS se encuentra en la función *check* del archivo *./api/controllers/UserController.js* en el anexo B.2.

❑ MU3-3:

- ❖ El código JS se encuentra en la función *check* del archivo *./api/controllers/UserController.js* en el anexo B.2.

❑ MU3-4:

- ❖ El código JS se encuentra en el archivo *./config/routes.js* en el anexo B.2.
- ❖ El servicio se encuentra publicado en la ruta */user/checkUser* con el método HTTP post.

❑ MU3-5:

- ❖ El código JS se encuentra en la función *updateSecurityInfo* del archivo *./api/controllers/UserController.js* en el anexo B.2.

❑ MU3-6:

- ❖ El código JS se encuentra en la función *updateSecurityInfo* del archivo *./api/controllers/UserController.js* en el anexo B.2.

❑ MU3-7:

- ❖ El código JS se encuentra en la función *updateSecurityInfo* del archivo *./api/controllers/UserController.js* en el anexo B.2.

❑ MU3-8:

- ❖ El código JS se encuentra en el archivo *./config/routes.js* en el anexo B.2.
- ❖ El servicio se encuentra publicado en la ruta */user/updateSecurityInfo* con el método HTTP post.

❑ MU3-9:

- ✧ El código JS se encuentra en el archivo `./assets/js/angular/controllers/User/recover.js` en el anexo B.2.

❑ **MU3-10:**

- ✧ El código JS se encuentra en el archivo `./assets/js/angular/controllers/User/recover.js` en el anexo B.2.

❑ **MU3-11:**

- ✧ El código JS se encuentra en el archivo `./assets/js/angular/controllers/User/recover2.js` en el anexo B.2.

❑ **MU3-12:**

- ✧ El código JS se encuentra en el archivo `./assets/js/angular/controllers/User/recover2.js` en el anexo B.2.

❑ **MU3-13:**

- ✧ El código JS se encuentra en el archivo `./assets/js/angular/controllers/User/recover2.js` en el anexo B.2.

❑ **MU4-1:**

- ✧ El código JS se encuentra en la función `login` del archivo `./api/controllers/UserController.js` en el anexo B.2.

❑ **MU4-2:**

- ✧ El código JS se encuentra en la función `login` del archivo `./api/controllers/UserController.js` en el anexo B.2.

❑ **MU4-3:**

- ✧ El código JS se encuentra en el archivo `./assets/js/angular/controllers/home.js` en el anexo B.2.

❑ **MU4-4:**

- ✧ El código JS se encuentra en el archivo *./assets/js/angular/controllers/home.js* en el anexo B.2.

❑ **MU4-5:**

- ✧ El código JS se encuentra en el archivo *./assets/js/angular/controllers/home.js* en el anexo B.2.

❑ **MU4-6:**

- ✧ El código JS se encuentra en el archivo *./assets/js/angular/controllers/home.js* en el anexo B.2.

❑ **MU4-7:**

- ✧ El código JS se encuentra en el archivo *./assets/js/angular/controllers/home.js* en el anexo B.2.

❑ **MU4-8:**

- ✧ El código JS se encuentra en el archivo *./assets/js/angular/controllers/home.js* en el anexo B.2.

❑ **MU4-9:**

- ✧ El código JS se encuentra en el archivo *./assets/js/angular/controllers/home.js* en el anexo B.2.

❑ **MU5-1:**

- ✧ El código HTML se encuentra en el archivo *./views/homepage.ejs* en el anexo B.2.

❑ **MU5-2:**

- ✧ La funcionalidad fue implementada directamente en el código HTML que se encuentra en el archivo *./views/homepage.ejs* en el anexo B.2.

❑ **MU6-1:**

- ✧ El código JS se encuentra en el archivo `./api/policias/isLoggedIn.js` en el anexo B.2.

❑ **MU6-2:**

- ✧ La política no se aplica a los servicios REST HTTP relacionados al módulo de usuario.

❑ **MU6-3:**

- ✧ El código JS se encuentra en el archivo `./config/policias.js` en el anexo B.2.

❑ **T1-1:**

- ✧ El código JS se encuentra en el escenario *Register user* en el archivo `./tests/RestServices.js` en el anexo B.2.
- ✧ Los resultados de las pruebas se muestran en el anexo D.1.

❑ **T1-2:**

- ✧ El código JS se encuentra en el escenario *Login user* en el archivo `./tests/RestServices.js` en el anexo B.2.
- ✧ Los resultados de las pruebas se muestran en el anexo D.1.

❑ **T1-3:**

- ✧ El código JS se encuentra en los escenarios *Check user data for password and pin change* y *Change password and pin* en el archivo `./tests/RestServices.js` en el anexo B.2.
- ✧ Los resultados de las pruebas se muestran en el anexo D.1.

La Figura 3.3 representa la gráfica Burn-Up de este Sprint. Esta refleja el tiempo invertido acumulado durante todos los días del Sprint.

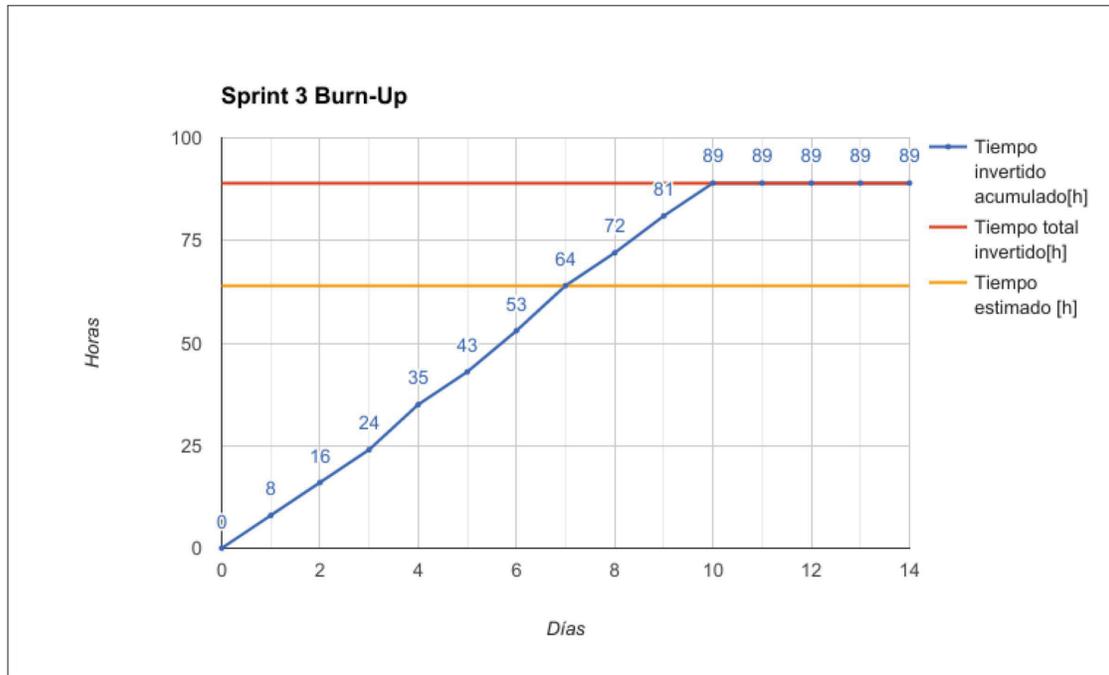


Figura 3.3: Burn-Up Sprint 3.

3.7.3 SPRINT REVIEW

Se completaron la totalidad de las historias de usuario y sus tareas con un total de 87 horas invertidas en el Sprint, esto se aprecia en la Tabla E.16.

3.7.4 SPRINT RETROSPECTIVE

El tiempo total invertido durante el Sprint fue superior al tiempo total estimado, existiendo una diferencia de 24 horas. Se presentaron dificultades que requirieron más tiempo en las tareas: MU1-4, MU3-3, MU3-7, MU3-11, MU4-1, MU4-2, MU4-3, MU4-4 y MU4-5.

3.8 SPRINT 4

3.8.1 SPRINT PLANNING

3.8.1.1 Sprint Goal

El objetivo del Sprint es desarrollar todas las funcionalidades del módulo curso, y asegurar la calidad de estas con sus respectivas pruebas.

3.8.1.2 Sprint Backlog

El Sprint Backlog 4 se presenta en la Tabla E.18, y está constituido por las historias de usuario con ID: MC1, MC2, MC3, MC4, MC5, MC6, MC7, MC8, y TE1.

3.8.2 EJECUCIÓN DEL SPRINT

Las tareas realizadas en este Sprint fueron:

❑ MC1-1:

- ✧ La interfaz gráfica se puede apreciar en la Figura F.10.
- ✧ El código HTML se encuentra en el archivo `./assets/html/angular/views/-Course/register.html` en el anexo B.2.

❑ MC1-2:

- ✧ El código JS se encuentra en la función `register` del archivo `./api/controllers/CourseController.js` en el anexo B.2.

❑ MC1-3:

- ✧ El código JS se encuentra en la función `register` del archivo `./api/controllers/CourseController.js` en el anexo B.2.

❑ MC1-4:

- ✧ El código JS se encuentra en el archivo *./config/routes.js* en el anexo B.2.
- ✧ El servicio se encuentra publicado en la ruta */course/register* con el método HTTP post.

❑ **MC1-5:**

- ✧ El código JS se encuentra en el archivo *./assets/js/angular/controllers/-Course/register.js* en el anexo B.2.

❑ **MC1-6:**

- ✧ El código JS se encuentra en el archivo *./assets/js/angular/controllers/-Course/register.js* en el anexo B.2.

❑ **MC1-7:**

- ✧ El código JS se encuentra en el archivo *./assets/js/angular/controllers/-Course/register.js* en el anexo B.2.

❑ **MC2-1:**

- ✧ La interfaz gráfica se puede apreciar en la Figura F.11.
- ✧ El código HTML se encuentra en el archivo *./assets/html/angular/views/-Course/registerStudent.html* en el anexo B.2.

❑ **MC2-2:**

- ✧ El código JS se encuentra en la función *registerStudent* del archivo *./api/controllers/CourseController.js* en el anexo B.2.

❑ **MC2-3:**

- ✧ El código JS se encuentra en la función *registerStudent* del archivo *./api/controllers/CourseController.js* en el anexo B.2.

❑ **MC2-4:**

- ✧ El código JS se encuentra en el archivo *./config/routes.js* en el anexo B.2.

- ✧ El servicio se encuentra publicado en la ruta `/course/register` con el método HTTP post.

❑ **MC2-5:**

- ✧ El código JS se encuentra en el archivo `./assets/js/angular/controllers/-Course/registerStudent.js` en el anexo B.2.

❑ **MC2-6:**

- ✧ El código JS se encuentra en el archivo `./assets/js/angular/controllers/-Course/registerStudent.js` en el anexo B.2.

❑ **MC2-7:**

- ✧ El código JS se encuentra en el archivo `./assets/js/angular/controllers/-Course/registerStudent.js` en el anexo B.2.

❑ **MC3-1:**

- ✧ Las interfaces gráficas se pueden apreciar en las Figuras F.12 y F.13.
- ✧ El código HTML se encuentra en los archivos `./assets/html/angular/views/Course/register.html` y `./assets/html/angular/views/Course/clone.html` en el anexo B.2.

❑ **MC3-2:**

- ✧ El código JS se encuentra en la función `clone` del archivo `./api/controllers/CourseController.js` en el anexo B.2.

❑ **MC3-3:**

- ✧ El código JS se encuentra en la función `getCoursesByTeacher` del archivo `./api/controllers/CourseController.js` en el anexo B.2.

❑ **MC3-4:**

- ✧ El código JS se encuentra en el archivo `./config/routes.js` en el anexo B.2.

- ✧ El servicio se encuentra publicado en la ruta */course/byTeacher* con el método HTTP post.

❑ **MC3-5:**

- ✧ El código JS se encuentra en la función *clone* del archivo *./api/controllers/CourseController.js* en el anexo B.2.

❑ **MC3-6:**

- ✧ El código JS se encuentra en el archivo *./config/routes.js* en el anexo B.2.
- ✧ El servicio se encuentra publicado en la ruta */course/clone* con el método HTTP post.

❑ **MC3-7:**

- ✧ El código JS se encuentra en el archivo *./assets/js/angular/controllers/Course/clone.js* en el anexo B.2.

❑ **MC3-8:**

- ✧ El código JS se encuentra en el archivo *./assets/js/angular/controllers/Course/clone.js* en el anexo B.2.

❑ **MC3-9:**

- ✧ El código JS se encuentra en el archivo *./assets/js/angular/controllers/Course/clone.js* en el anexo B.2.

❑ **MC4-1:**

- ✧ La interfaz gráfica se puede apreciar en la Figura F.15.
- ✧ El código HTML se encuentra en el archivo *./assets/html/angular/views/Course/clone.html* en el anexo B.2.

❑ **MC4-2:**

- ✧ El código JS se encuentra en la función *getCourseById* del archivo *./api/controllers/CourseController.js* en el anexo B.2.

❑ MC4-3:

- ❖ El código JS se encuentra en el archivo `./config/routes.js` en el anexo B.2.
- ❖ El servicio se encuentra publicado en la ruta `/course/getByld` con el método HTTP post.

❑ MC4-4:

- ❖ El código JS se encuentra en la función `editCourse` del archivo `./api/controllers/CourseController.js` en el anexo B.2.

❑ MC4-5:

- ❖ El código JS se encuentra en la función `editCourse` del archivo `./api/controllers/CourseController.js` en el anexo B.2.

❑ MC4-6:

- ❖ El código JS se encuentra en el archivo `./config/routes.js` en el anexo B.2.
- ❖ El servicio se encuentra publicado en la ruta `/course/update` con el método HTTP post.

❑ MC4-7:

- ❖ El código JS se encuentra en el archivo `./assets/js/angular/controllers/Course/edit.js` en el anexo B.2.

❑ MC4-8:

- ❖ El código JS se encuentra en el archivo `./assets/js/angular/controllers/Course/edit.js` en el anexo B.2.

❑ MC4-9:

- ❖ El código JS se encuentra en el archivo `./assets/js/angular/controllers/Course/edit.js` en el anexo B.2.

❑ MC4-10:

- ✧ El código JS se encuentra en el archivo *./assets/js/angular/controllers/-Course/edit.js* en el anexo B.2.

❑ **MC5-1:**

- ✧ La interfaz gráfica se puede apreciar en las Figuras F.16 y F.17.
- ✧ El código HTML se encuentra en el archivo *./assets/html/angular/views/-Teacher/home.html* en el anexo B.2.

❑ **MC5-2:**

- ✧ El código JS se encuentra en la función *deleteCourse* del archivo *./api/controllers/CourseController.js* en el anexo B.2.

❑ **MC5-3:**

- ✧ El código JS se encuentra en la función *deleteCourse* del archivo *./api/controllers/CourseController.js* en el anexo B.2.

❑ **MC5-4:**

- ✧ El código JS se encuentra en el archivo *./config/routes.js* en el anexo B.2.
- ✧ El servicio se encuentra publicado en la ruta */course/delete* con el método HTTP post.

❑ **MC5-5:**

- ✧ El código JS se encuentra en el archivo *./assets/js/angular/controllers/-Teacher/home.js* en el anexo B.2.

❑ **MC5-6:**

- ✧ El código JS se encuentra en el archivo *./assets/js/angular/controllers/-Teacher/home.js* en el anexo B.2.

❑ **MC5-7:**

- ✧ El código JS se encuentra en el archivo *./assets/js/angular/controllers/-Teacher/home.js* en el anexo B.2.

❑ MC6-1:

- ❖ Las interfaces gráficas se pueden apreciar en las Figuras F.18 y F.19.
- ❖ El código HTML se encuentra en los archivos `./assets/html/angular/views/Course/homeStudent.html` y `./assets/html/angular/views/Course/homeTeacher.html` en el anexo B.2.

❑ MC6-2:

- ❖ El código JS se encuentra en los archivos `./assets/js/angular/controllers/Teacher/home.js` y `./assets/js/angular/controllers/Student/home.js` en el anexo B.2.

❑ MC7-1:

- ❖ La interfaz gráfica se puede apreciar en las Figuras F.20 y F.21.
- ❖ El código HTML se encuentra en los archivos `./assets/html/angular/views/Teacher/home.html` y `./assets/html/angular/views/Students/home.html` en el anexo B.2.

❑ MC7-2:

- ❖ El código JS se encuentra en las funciones `getCoursesByTeacher` y `getCoursesByStudent` del archivo `./api/controllers/CourseController.js` en el anexo B.2.

❑ MC7-3:

- ❖ El código JS se encuentra en las funciones `getCoursesByTeacher` y `getCoursesByStudent` del archivo `./api/controllers/CourseController.js` en el anexo B.2.

❑ MC7-4:

- ❖ El código JS se encuentra en el archivo `./config/routes.js` en el anexo B.2.
- ❖ El servicio se encuentra publicado en las rutas `/course/byStudent` con el método HTTP post y `/course/byTeacher` con el método HTTP post.

❑ MC7-5:

- ❖ El código JS se encuentra en los archivos `./assets/js/angular/controllers/Teacher/home.js` y `./assets/js/angular/controllers/Student/home.js` en el anexo B.2.

❑ MC8-1:

- ❖ La interfaz gráfica se puede apreciar en la Figura F.22.
- ❖ El código HTML se encuentra en el archivo `./assets/html/angular/views/Course/homeTeacher.html` en el anexo B.2.

❑ MC8-2:

- ❖ El código JS se encuentra en la función `deleteStudent` del archivo `./api/controllers/CourseController.js` en el anexo B.2.

❑ MC8-3:

- ❖ El código JS se encuentra en la función `deleteStudent` del archivo `./api/controllers/CourseController.js` en el anexo B.2.

❑ MC8-4:

- ❖ El código JS se encuentra en el archivo `./config/routes.js` en el anexo B.2.
- ❖ El servicio se encuentra publicado en la ruta `/course/deleteStudent` con el método HTTP post.

❑ MC8-5:

- ❖ El código JS se encuentra en el archivo `./assets/js/angular/controllers/Course/homeTeacher.js` en el anexo B.2.

❑ MC8-6:

- ❖ El código JS se encuentra en el archivo `./assets/js/angular/controllers/Course/homeTeacher.js` en el anexo B.2.

❑ TE2-1:

- ✧ El código JS se encuentra en el escenario *Course register* en el archivo *./tests/RestServices.js* en el anexo B.2.

- ✧ Los resultados de las pruebas se muestran en el anexo D.1.

□ TE2-2:

- ✧ El código JS se encuentra en el escenario *Register a student in a course* en el archivo *./tests/RestServices.js* en el anexo B.2.

- ✧ Los resultados de las pruebas se muestran en el anexo D.1.

□ TE2-3:

- ✧ El código JS se encuentra en el escenario *Clone course* en el archivo *./tests/RestServices.js* en el anexo B.2.

- ✧ Los resultados de las pruebas se muestran en el anexo D.1.

□ TE2-4:

- ✧ El código JS se encuentra en el escenario *Edit course* en el archivo *./tests/RestServices.js* en el anexo B.2.

- ✧ Los resultados de las pruebas se muestran en el anexo D.1.

□ TE2-5:

- ✧ El código JS se encuentra en el escenario *Delete course* en el archivo *./tests/RestServices.js* en el anexo B.2.

- ✧ Los resultados de las pruebas se muestran en el anexo D.1.

□ TE2-6:

- ✧ El código JS se encuentra en los escenarios *Get courses by teacher* y *Get courses by student* en el archivo *./tests/RestServices.js* en el anexo B.2.

- ✧ Los resultados de las pruebas se muestran en el anexo D.1.

□ TE2-7:

- ✧ El código JS se encuentra en el escenario *Delete student from course* en el archivo `./tests/RestServices.js` en el anexo B.2.
- ✧ Los resultados de las pruebas se muestran en el anexo D.1.

La Figura 3.4 representa la gráfica Burn-Up de este Sprint. Esta refleja el tiempo invertido acumulado durante todos los días del Sprint.

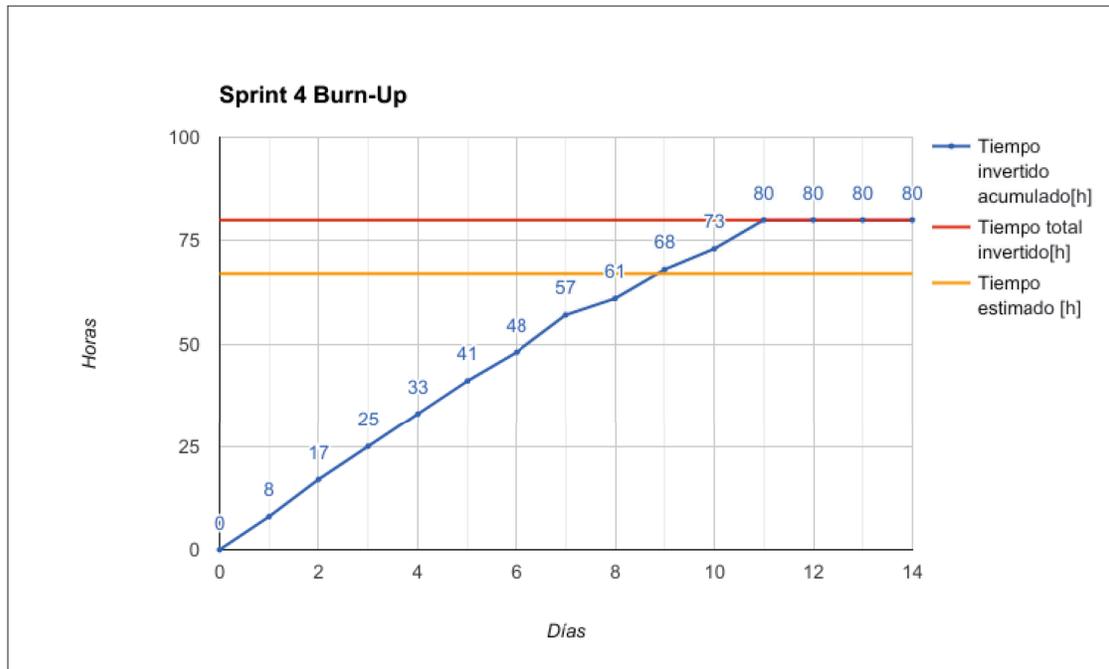


Figura 3.4: Burn-Up Sprint 4.

3.8.3 SPRINT REVIEW

Se completaron la totalidad de las historias de usuario y sus tareas con un total de 80 horas invertidas en el Sprint, esto se aprecia en la Tabla E.23.

3.8.4 SPRINT RETROSPECTIVE

El tiempo total invertido durante el Sprint fue superior al tiempo total estimado, existiendo una diferencia de 13 horas. Se presentaron dificultades que requirieron más tiempo en las tareas: MC2-3, MC3-1, MC3-2, MC3-3, MC3-5, MC4-5, MC5-3, MC5-6 y MC7-2.

3.9 SPRINT 5

3.9.1 SPRINT PLANNING

3.9.1.1 Sprint Goal

El objetivo del Sprint es desarrollar todas las funcionalidades del módulo prueba, y asegurar la calidad de estas con sus respectivas pruebas.

3.9.1.2 Sprint Backlog

El Sprint Backlog 5 se presenta en la Tabla E.25, y está constituido por las historias de usuario con ID: MP1, MP2, MP3, MP4, MP5, MP6, MP7, MP8, y TE3.

3.9.2 EJECUCIÓN DEL SPRINT

Las tareas realizadas en este Sprint fueron:

❑ MP1-1:

- ✧ La interfaz gráfica se puede apreciar en la Figura F.23.
- ✧ El código HTML se encuentra en el archivo `./assets/html/angular/views/Test/register.html` en el anexo B.2.

❑ MP1-2:

- ✧ El código JS se encuentra en la función `register` del archivo `./api/controllers/TestController.js` en el anexo B.2.

❑ MP1-3:

- ✧ El código JS se encuentra en la función `register` del archivo `./api/controllers/TestController.js` en el anexo B.2.

❑ MP1-4:

- ✧ El código JS se encuentra en el archivo *./config/routes.js* en el anexo B.2.
- ✧ El servicio se encuentra publicado en la ruta */test/register* con el método HTTP post.

❑ **MP1-5:**

- ✧ El código JS se encuentra en el archivo *./assets/js/angular/controllers/-Test/register.js* en el anexo B.2.

❑ **MP1-6:**

- ✧ El código JS se encuentra en el archivo *./assets/js/angular/controllers/-Test/register.js* en el anexo B.2.

❑ **MP1-7:**

- ✧ El código JS se encuentra en el archivo *./assets/js/angular/controllers/-Test/register.js* en el anexo B.2.

❑ **MP2-1:**

- ✧ La interfaz gráfica se puede apreciar en la Figura F.24.
- ✧ El código HTML se encuentra en el archivo *./assets/html/angular/views/-Test/edit.html* en el anexo B.2.

❑ **MP2-2:**

- ✧ El código JS se encuentra en la función *getTestById* del archivo *./api/controllers/TestController.js* en el anexo B.2.

❑ **MP2-3:**

- ✧ El código JS se encuentra en el archivo *./config/routes.js* en el anexo B.2.
- ✧ El servicio se encuentra publicado en la ruta */test/getTestById* con el método HTTP post.

❑ **MP2-4:**

- ✧ El código JS se encuentra en la función *edit* del archivo *./api/controllers/TestController.js* en el anexo B.2.

❑ **MP2-5:**

- ✧ El código JS se encuentra en la función *getTestById* del archivo *./api/controllers/TestController.js* en el anexo B.2.

❑ **MP2-6:**

- ✧ El código JS se encuentra en el archivo *./config/routes.js* en el anexo B.2.
- ✧ El servicio se encuentra publicado en la ruta */test/update* con el método HTTP post.

❑ **MP2-7:**

- ✧ El código JS se encuentra en el archivo *./assets/js/angular/controllers/Test/edit.js* en el anexo B.2.

❑ **MP2-8:**

- ✧ El código JS se encuentra en el archivo *./assets/js/angular/controllers/Test/edit.js* en el anexo B.2.

❑ **MP2-9:**

- ✧ El código JS se encuentra en el archivo *./assets/js/angular/controllers/Test/edit.js* en el anexo B.2.

❑ **MP2-10:**

- ✧ El código JS se encuentra en el archivo *./assets/js/angular/controllers/Test/edit.js* en el anexo B.2.

❑ **MP3-1:**

- ✧ La interfaz gráfica se puede apreciar en las Figuras F.25 y F.26.
- ✧ El código HTML se encuentra en los archivos *./assets/html/angular/views/Teacher/home.html* y *./assets/html/angular/views/Course/homeTeacher.html* en el anexo B.2.

❑ MP3-2:

- ❖ El código JS se encuentra en la función *deleteTest* del archivo *./api/controllers/TestController.js* en el anexo B.2.

❑ MP3-3:

- ❖ El código JS se encuentra en la función *deleteTest* del archivo *./api/controllers/TestController.js* en el anexo B.2.

❑ MP3-4:

- ❖ El código JS se encuentra en el archivo *./config/routes.js* en el anexo B.2.
- ❖ El servicio se encuentra publicado en la ruta */test/delete* con el método HTTP post.

❑ MP3-5:

- ❖ El código JS se encuentra en los archivos *./assets/js/angular/controllers/Teacher/home.js* y *./assets/js/angular/controllers/Course/homeTeacher.js* en el anexo B.2.

❑ MP3-6:

- ❖ El código JS se encuentra en los archivos *./assets/js/angular/controllers/Teacher/home.js* y *./assets/js/angular/controllers/Course/homeTeacher.js* en el anexo B.2.

❑ MP3-7:

- ❖ El código JS se encuentra en los archivos *./assets/js/angular/controllers/Teacher/home.js* y *./assets/js/angular/controllers/Course/homeTeacher.js* en el anexo B.2.

❑ MP4-1:

- ❖ La interfaz gráfica se puede apreciar en las Figuras F.27 y F.28.

- ✧ El código HTML se encuentra en los archivos *./assets/html/angular/views/Teacher/home.html* y *./assets/html/angular/views/Students/home.html* en el anexo B.2.

❑ **MP4-2:**

- ✧ El código JS se encuentra en las funciones *getTestsByStudent* y *getTestsCreatedByUser* del archivo *./api/controllers/TestController.js* en el anexo B.2.

❑ **MP4-3:**

- ✧ El código JS se encuentra en las funciones *getTestsByStudent* y *getTestsCreatedByUser* del archivo *./api/controllers/TestController.js* en el anexo B.2.

❑ **MP4-4:**

- ✧ El código JS se encuentra en las funciones *getTestsByStudent* y *getTestsCreatedByUser* del archivo *./api/controllers/TestController.js* en el anexo B.2.

❑ **MP4-5:**

- ✧ El código JS se encuentra en el archivo *./config/routes.js* en el anexo B.2.
- ✧ El servicio se encuentra publicado en las rutas */test/createdByUser* con el método HTTP post y */test/byStudent* con el método HTTP post.

❑ **MP4-6:**

- ✧ El código JS se encuentra en los archivos *./assets/js/angular/controllers/Teacher/home.js* y *./assets/js/angular/controllers/Student/home.js* en el anexo B.2.

❑ **MP5-1:**

- ✧ La interfaz gráfica se puede apreciar en las Figuras F.29 y F.30.

- ✧ El código HTML se encuentra en los archivos *./assets/html/angular/views/Course/homeStudent.html* y *./assets/html/angular/views/Course/homeTeacher.html* en el anexo B.2.

❑ **MP5-2:**

- ✧ El código JS se encuentra en las funciones *getTestsByCourseByStudent* y *getTestsByCourseByTeacher* del archivo *./api/controllers/TestController.js* en el anexo B.2.

❑ **MP5-3:**

- ✧ El código JS se encuentra en las funciones *getTestsByCourseByStudent* y *getTestsByCourseByTeacher* del archivo *./api/controllers/TestController.js* en el anexo B.2.

❑ **MP5-4:**

- ✧ El código JS se encuentra en las funciones *getTestsByCourseByStudent* y *getTestsByCourseByTeacher* del archivo *./api/controllers/TestController.js* en el anexo B.2.

❑ **MP5-5:**

- ✧ El código JS se encuentra en el archivo *./config/routes.js* en el anexo B.2.
- ✧ El servicio se encuentra publicado en las rutas */test/byCourseByTeacher* con el método HTTP post y */test/byCourseByStudent* con el método HTTP post.

❑ **MP5-6:**

- ✧ El código JS se encuentra en los archivos *./assets/js/angular/controllers/Course/teacherHome.js* y *./assets/js/angular/controllers/Course/studentHome.js* en el anexo B.2.

❑ **MP6-1:**

- ✧ La interfaz gráfica se puede apreciar en la Figura F.31.

- ✧ El código HTML se encuentra en el archivo *./assets/html/angular/views/-Test/studentTest.html* en el anexo B.2.

❑ **MP6-2:**

- ✧ El código JS se encuentra en la función *getTestByldForStudent* del archivo *./api/controllers/TestController.js* en el anexo B.2.

❑ **MP6-3:**

- ✧ El código JS se encuentra en la función *getTestByldForStudent* del archivo *./api/controllers/TestController.js* en el anexo B.2.

❑ **MP6-4:**

- ✧ El código JS se encuentra en el archivo *./config/routes.js* en el anexo B.2.
- ✧ El servicio se encuentra publicado en la ruta */test/getTestByldForStudent* con el método HTTP post.

❑ **MP6-5:**

- ✧ El código JS se encuentra en la función *registerTakenTest* del archivo *./api/controllers/TestController.js* en el anexo B.2.

❑ **MP6-6:**

- ✧ El código JS se encuentra en la función *registerTakenTest* del archivo *./api/controllers/TestController.js* en el anexo B.2.

❑ **MP6-7:**

- ✧ El código JS se encuentra en el archivo *./config/routes.js* en el anexo B.2.
- ✧ El servicio se encuentra publicado en la ruta */test/registerTakenTest* con el método HTTP post.

❑ **MP6-8:**

- ✧ El código JS se encuentra en el archivo *./assets/js/angular/controllers/-Test/studentTest.js* en el anexo B.2.

❑ MP6-9:

- ❖ El código JS se encuentra en el archivo *./assets/js/angular/controllers/-Test/studentTest.js* en el anexo B.2.

❑ MP6-10:

- ❖ El código JS se encuentra en el archivo *./assets/js/angular/controllers/-Test/studentTest.js* en el anexo B.2.

❑ MP6-11:

- ❖ El código JS se encuentra en el archivo *./assets/js/angular/controllers/-Test/studentTest.js* en el anexo B.2.

❑ MP6-12:

- ❖ El código JS se encuentra en el archivo *./assets/js/angular/controllers/-Test/studentTest.js* en el anexo B.2.

❑ MP7-1:

- ❖ La interfaz gráfica se puede apreciar en la Figura F.32.
- ❖ El código HTML se encuentra en el archivo *./assets/html/angular/views/-Test/studentReview.html* en el anexo B.2.

❑ MP7-2:

- ❖ El código JS se encuentra en el archivo *./assets/js/angular/controllers/-Test/studentReview.js* en el anexo B.2.

❑ MP7-3:

- ❖ El código JS se encuentra en el archivo *./assets/js/angular/controllers/-Test/studentReview.js* en el anexo B.2.

❑ MP8-1:

- ❖ La interfaz gráfica se puede apreciar en las Figuras F.33 y F.34.

- ✧ El código HTML se encuentra en el archivo *./assets/html/angular/views/Test/register.html* en el anexo B.2.

❑ **MP8-2:**

- ✧ El código JS se encuentra en el archivo *./assets/js/angular/controllers/Test/listAll.js* en el anexo B.2.

❑ **MP8-3:**

- ✧ El código JS se encuentra en el archivo *./assets/js/angular/controllers/Test/listAll.js* en el anexo B.2.

❑ **TE3-1:**

- ✧ El código JS se encuentra en el escenario *Register test* en el archivo *./tests/RestServices.js* en el anexo B.2.
- ✧ Los resultados de las pruebas se muestran en el anexo D.1.

❑ **TE3-2:**

- ✧ El código JS se encuentra en el escenario *Update test* en el archivo *./tests/RestServices.js* en el anexo B.2.
- ✧ Los resultados de las pruebas se muestran en el anexo D.1.

❑ **TE3-3:**

- ✧ El código JS se encuentra en el escenario *Delete test* en el archivo *./tests/RestServices.js* en el anexo B.2.
- ✧ Los resultados de las pruebas se muestran en el anexo D.1.

❑ **TE3-4:**

- ✧ El código JS se encuentra en los escenarios *Get tests created by user* y *Get tests by student* el archivo *./tests/RestServices.js* en el anexo B.2.
- ✧ Los resultados de las pruebas se muestran en el anexo D.1.

❑ **TE3-5:**

- ✧ El código JS se encuentra en los escenarios *Get tests by course and student* y *Get tests by course and teacher* el archivo *./tests/RestServices.js* en el anexo B.2.
- ✧ Los resultados de las pruebas se muestran en el anexo D.1.

□ TE3-6:

- ✧ El código JS se encuentra en los escenarios *Register test taken* y *Get test by id for student* el archivo *./tests/RestServices.js* en el anexo B.2.
- ✧ Los resultados de las pruebas se muestran en el anexo D.1.

La Figura 3.5 representa la gráfica Burn-Up de este Sprint. Esta refleja el tiempo invertido acumulado durante todos los días del Sprint.

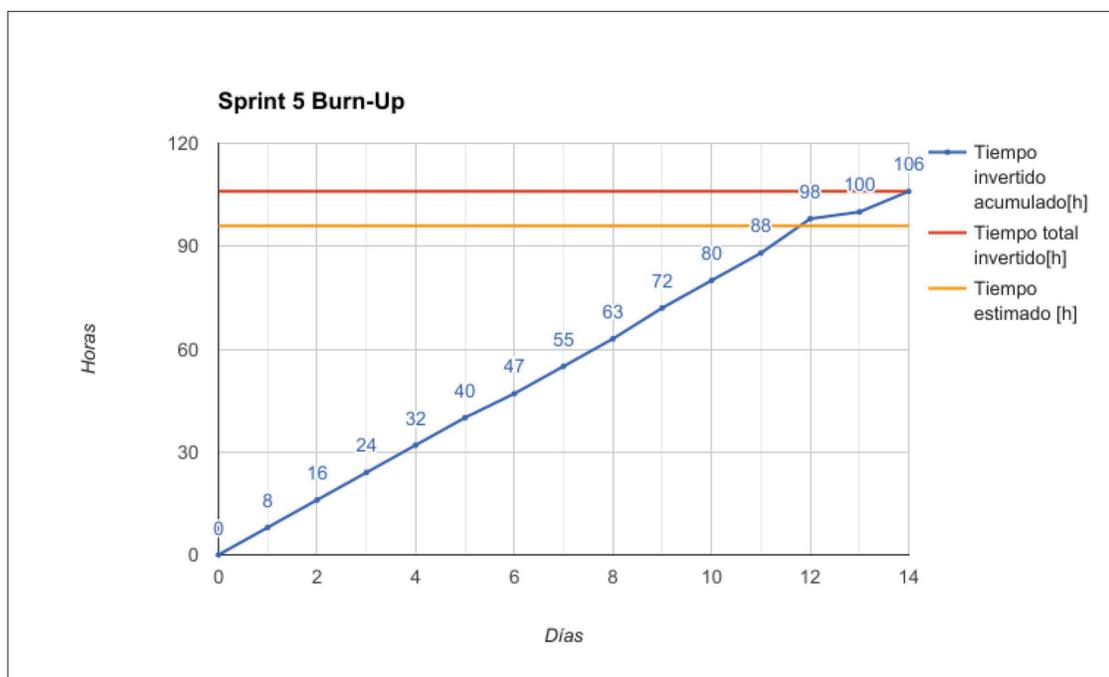


Figura 3.5: Burn-Up Sprint 5.

3.9.3 SPRINT REVIEW

Se completaron la totalidad de las historias de usuario y sus tareas con un total de 106 horas invertidas en el Sprint, esto se aprecia en la Tabla E.30.

3.9.4 SPRINT RETROSPECTIVE

El tiempo total invertido durante el Sprint fue superior al tiempo total estimado, existiendo una diferencia de 10 horas. Se presentaron dificultades que requirieron más tiempo en las tareas: MP1-1, MP1-2, MP1-3, MP2-2 y MP6-6.

3.10 SPRINT 6

3.10.1 SPRINT PLANNING

3.10.1.1 Sprint Goal

El objetivo del Sprint es desarrollar todas las funcionalidades del módulo pregunta, asegurar la calidad de estas con sus respectivas pruebas y generar la documentación técnica JSDoc del código, los manuales de usuario y de despliegue de la aplicación.

3.10.1.2 Sprint Backlog

El Sprint Backlog 6 se presenta en la Tabla E.32, y está constituido por las historias de usuario con ID: MR1, MR2, MR3, MR4, TE4, TC1, TI1, JS1, AD1, MA1, y MA2.

3.10.2 EJECUCIÓN DEL SPRINT

Las tareas realizadas en este Sprint fueron:

□ MR1-1:

- ✧ La interfaz gráfica se puede apreciar en las Figuras F.35, F.36, y F.37.
- ✧ El código HTML se encuentra en el archivo `./assets/html/angular/views/-Test/register.html` en el anexo B.2.

❑ MR1-2:

- ❖ El código JS se encuentra en las funciones *register* y *edit* del archivo *./api/controllers/TestController.js* en el anexo B.2.

❑ MR1-3:

- ❖ El código JS se encuentra en las funciones *register* y *edit* del archivo *./api/controllers/TestController.js* en el anexo B.2.

❑ MR1-4:

- ❖ El código JS se encuentra en los archivos *./assets/js/angular/controllers/Test/register.js* y *./assets/js/angular/controllers/Test/edit.js* en el anexo B.2.

❑ MR1-5:

- ❖ El código JS se encuentra en los archivos *./assets/js/angular/controllers/Test/register.js* y *./assets/js/angular/controllers/Test/edit.js* en el anexo B.2.

❑ MR2-1:

- ❖ La interfaz gráfica se puede apreciar en las Figuras F.38, F.39 y F.40.
- ❖ El código HTML se encuentra en el archivo *./assets/html/angular/views/Test/edit.html* en el anexo B.2.

❑ MR2-2:

- ❖ El código JS se encuentra en la función *getTestByld* del archivo *./api/controllers/TestController.js* en el anexo B.2.

❑ MR2-3:

- ❖ El código JS se encuentra en el archivo *./config/routes.js* en el anexo B.2.
- ❖ El servicio se encuentra publicado en la ruta */test/getTestByld* con el método HTTP post.

❑ MR2-4:

- ❖ El código JS se encuentra en las funciones *register* y *edit* del archivo *./api/controllers/TestController.js* en el anexo B.2.

❑ MR3-1:

- ❖ La interfaz gráfica se puede apreciar en las Figuras F.41.
- ❖ El código HTML se encuentra en los archivos *./assets/html/angular/views/Test/register.html* y *./assets/html/angular/views/Test/edit.html* en el anexo B.2.

❑ MR3-2:

- ❖ El código JS se encuentra en la función *delete* del archivo *./api/controllers/QuestionController.js* en el anexo B.2.

❑ MR3-3:

- ❖ El código JS se encuentra en la función *delete* del archivo *./api/controllers/QuestionController.js* en el anexo B.2.

❑ MR3-4:

- ❖ El código JS se encuentra en el archivo *./config/routes.js* en el anexo B.2.
- ❖ El servicio se encuentra publicado en la ruta */question/delete* con el método HTTP post.

❑ MR3-5:

- ❖ El código JS se encuentra en los archivos *./assets/js/angular/controllers/Test/resgister.js* y *./assets/js/angular/controllers/Test/edit.js* en el anexo B.2.

❑ MR3-6:

- ❖ El código JS se encuentra en los archivos *./assets/js/angular/controllers/Test/resgister.js* y *./assets/js/angular/controllers/Test/edit.js* en el anexo B.2.

❑ MR3-7:

- ❖ El código JS se encuentra en los archivos `./assets/js/angular/controllers/Test/resgister.js` y `./assets/js/angular/controllers/Test/edit.js` en el anexo B.2.

❑ MR4-1:

- ❖ La interfaz gráfica se puede apreciar en la Figura F.42.
- ❖ El código HTML se encuentra en el archivo `./assets/html/angular/views/Test/edit.html` en el anexo B.2.

❑ MR4-2:

- ❖ El código JS se encuentra en la función `cloneQuestion` del archivo `./api/controllers/QuestionController.js` en el anexo B.2.

❑ MR4-3:

- ❖ El código JS se encuentra en la función `cloneQuestion` del archivo `./api/controllers/QuestionController.js` en el anexo B.2.

❑ MR4-4:

- ❖ El código JS se encuentra en el archivo `./config/routes.js` en el anexo B.2.
- ❖ El servicio se encuentra publicado en la ruta `/test/question/clone` con el método HTTP post.

❑ MR4-5:

- ❖ El código JS se encuentra en el archivo `./assets/js/angular/controllers/Question/clone.js` en el anexo B.2.

❑ MR4-6:

- ❖ El código JS se encuentra en el archivo `./assets/js/angular/controllers/Question/clone.js` en el anexo B.2.

❑ MR4-7:

- ❖ El código JS se encuentra en el archivo `./assets/js/angular/controllers/Question/clone.js` en el anexo B.2.

❑ MR4-8:

- ❖ El código JS se encuentra en el archivo `./assets/js/angular/controllers/Question/clone.js` en el anexo B.2.

❑ TE4-1:

- ❖ El código JS se encuentra en el escenario *Clone question* en el archivo `./tests/RestServices.js` en el anexo B.2.
- ❖ Los resultados de las pruebas se muestran en el anexo D.1.

❑ TE4-2:

- ❖ El código JS se encuentra en el escenario *Delete question* en el archivo `./tests/RestServices.js` en el anexo B.2.
- ❖ Los resultados de las pruebas se muestran en el anexo D.1.

❑ TC1-1:

- ❖ La herramienta elegida para realizar el análisis de la calidad del código fuente de la aplicación fue Codacy, porque es un SAAS en el cual se agrega el repositorio de Github con el código del proyecto y realiza el análisis a todos los commits existentes de la aplicación. Adicional porque permite personalizar la lista de patrones aplicados al análisis del código fuente, el cual incluye patrones de compatibilidad, errores, rendimiento y estilo de código para los lenguajes CSS, JS, SQL, entre otros.

❑ TC1-2:

- ❖ Los resultados obtenidos se muestran en el anexo D.2.

❑ TI1-1:

- ❖ Los resultados obtenidos se muestran en el anexo D.3.

□ JS1-1:

- ✧ Los comentarios realizados se pueden apreciar en todos los archivos JS existentes en la carpeta `./api/controllers/`.

□ JS1-2:

- ✧ El documento generado corresponde al anexo A.11.

□ AD1-1:

- ✧ El nombre definido para la aplicación fue *Tests for all*, que en español significa pruebas para todos. Esto en referencia a que se trata de incluir a la mayor cantidad de personas, incluyendo las no videntes.

□ MA1-1:

- ✧ El documento generado corresponde al anexo A.7.

□ MA2-1:

- ✧ El documento generado corresponde al anexo A.6.

La Figura 3.6 representa la gráfica Burn-Up de este Sprint. Esta refleja el tiempo invertido acumulado durante todos los días del Sprint.

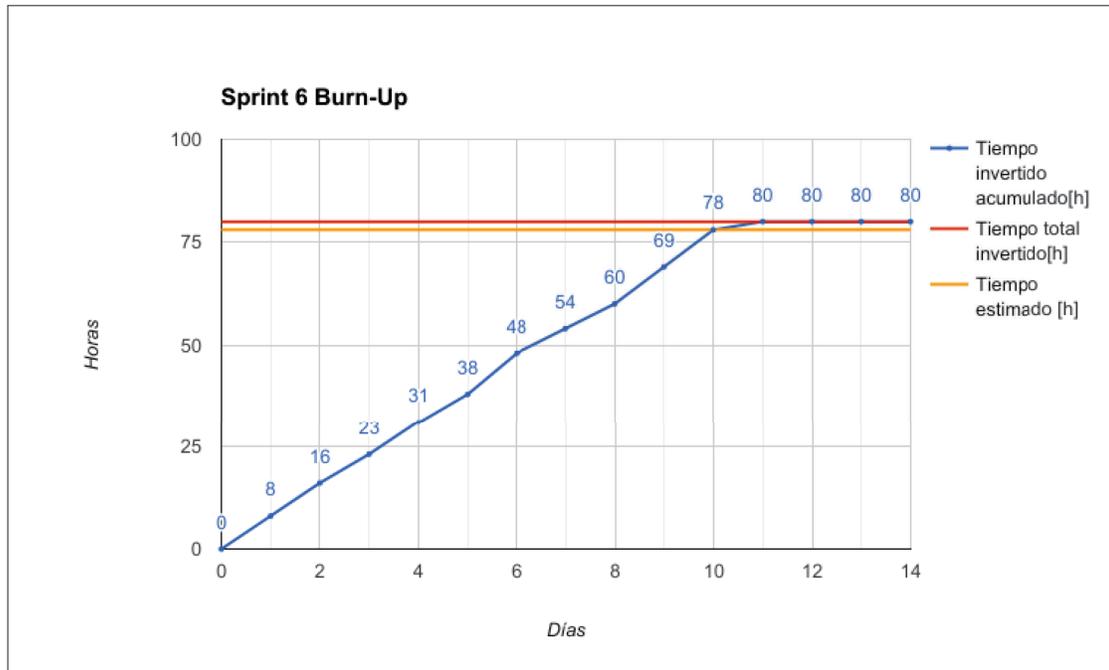


Figura 3.6: Burn-Up Sprint 6.

3.10.3 SPRINT REVIEW

Se completaron la totalidad de las historias de usuario y sus tareas con un total de 80 horas invertidas en el Sprint, esto se aprecia en la Tabla E.35.

3.10.4 SPRINT RETROSPECTIVE

El tiempo total invertido durante el Sprint fue superior al tiempo total estimado, existiendo una diferencia de 2 horas. Se presentaron dificultades que requirieron más tiempo en las tareas: MR1-3, MR1-4, MR2-2 y MR4-3.

CAPÍTULO 4

RESULTADOS

Este capítulo tiene como objetivo mostrar los resultados obtenidos cuando se realizó la toma de pruebas usando las distintas versiones de la aplicación. Los resultados referentes a las pruebas para asegurar a nivel técnico la calidad del producto se muestran en todos los anexos D y sus ejecuciones fueron descritas en el capítulo anterior. Se responderán las preguntas como:

- ❑ ¿Qué resultados se obtuvieron de las pruebas realizadas?
- ❑ ¿Es posible para una persona no vidente, rendir pruebas utilizando la aplicación?

4.1 ROADMAP DE PRUEBAS REALIZADAS

Para validar la funcionalidad de la aplicación y refinar esta, se realizaron pruebas utilizando distintas versiones de la aplicación, a medida que evolucionaba esta. Las pruebas realizadas se muestran en la figura 4.1.

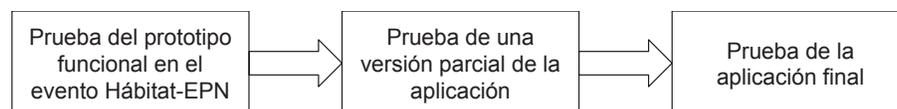


Figura 4.1: Roadmap de pruebas realizadas con la aplicación.

El objetivo del prototipo funcional era recopilar la mayor cantidad de retroalimentación posible por parte de los usuarios, de como hasta ese momento, se tenía la

funcionalidad de toma de pruebas. A partir de esta retroalimentación, se incluyeron los cambios en la aplicación, y se realizó la toma de pruebas a personas no videntes. Con la retroalimentación del uso de las personas no videntes de la aplicación desarrollada, se implementaron mejoras y cambios en los controles, dando como resultado la aplicación final. Finalmente, se tomaron pruebas con la aplicación final para validar el funcionamiento de esta.

4.2 RESULTADOS OBTENIDOS EN LA TOMA DE PRUEBAS CON EL PROTOTIPO FUNCIONAL

Se presentó el prototipo funcional desarrollado en el evento paralelo al evento Hábitat, llamado Hábitat-EPN, el cual fue organizado por la Escuela Politécnica Nacional (EPN). Para la retroalimentación del uso de la aplicación se consideró los resultados obtenidos por los participantes en la prueba, y una encuesta que se les realizó a estos.

La toma de pruebas utilizando el prototipo funcional se realizó bajo el siguiente escenario:

- La prueba fue rendida sin monitor, para asegurar de que en caso de la persona que rinde la prueba no tenga deficiencia visual, simule la situación de una persona no vidente.
- El audio fue transmitido utilizando audífonos para evitar que la persona evaluada escuche sonidos no provenientes de la aplicación, ya que el evento fue realizado al aire libre.
- La prueba fue rendida utilizando una PC de escritorio.

La prueba tomada en el prototipo constó de seis preguntas, de las cuales, dos eran de opción múltiple, dos de verdadero y falso, y dos de completar el espacio en blanco. La última pregunta no tenía respuesta correcta y era utilizada para recabar la opinión de los usuarios. Estas preguntas fueron:

1. ¿Cuál de las siguientes no es una región del Ecuador? 1) Costa, 2) Amazonía, 3) Arábica, 4) Sierra.
2. ¿Cuál de los siguientes animales es usado en el escudo nacional? 1) Cóndor, 2) Rana, 3) Águila, 4) Perro.
3. La mitad del mundo se encuentra en Ecuador. 1) Verdadero, 2) Falso.
4. La capital de *espacio en blanco* es Quito. 1) Ecuador, 2) Colombia, 3) Perú, 4) Chile.
5. Las islas *espacio en blanco* se encuentran en la región insular del Ecuador. 1) Filipinas, 2) Galápagos, 3) Hawái, 4) Malvinas.
6. ¿Encuentra de utilidad a la aplicación? 1) Verdadero, 2) Falso.

Se contó con la participación de 33 personas, las cuales rindieron la prueba utilizando la aplicación y realizaron la encuesta. Los resultados obtenidos en la prueba realizada pueden ser constatados en el anexo A.5. Estos fueron los siguientes:

Tabla 4.1: Resultados la toma de pruebas con el prototipo funcional en el evento Hábitat-EPN.

Pregunta	Opción 1	Opción 2	Opción 3	Opción 4	Total
1	0	0	31	1	32
2	29	2	0	0	31
3	31	1	-	-	32
4	32	0	0	0	32
5	0	31	0	0	31
6	30	1	-	-	31

Considerando las respuestas no seleccionadas como incorrectas, se obtuvieron un total de 154 respuestas correctas seleccionadas de 165, dando como resultado un acierto del 93,33%. Por consiguiente, es factible para una persona no vidente, rendir pruebas utilizando el prototipo funcional desarrollado. Adicional, en la última pregunta que 30 de 33 personas, es decir el 90,9% ve de utilidad a la aplicación.

Toda la información relacionada a la encuesta, incluyendo los resultados, se pueden observar en el anexo A.4. Las preguntas más importantes con sus resultados son las siguientes:

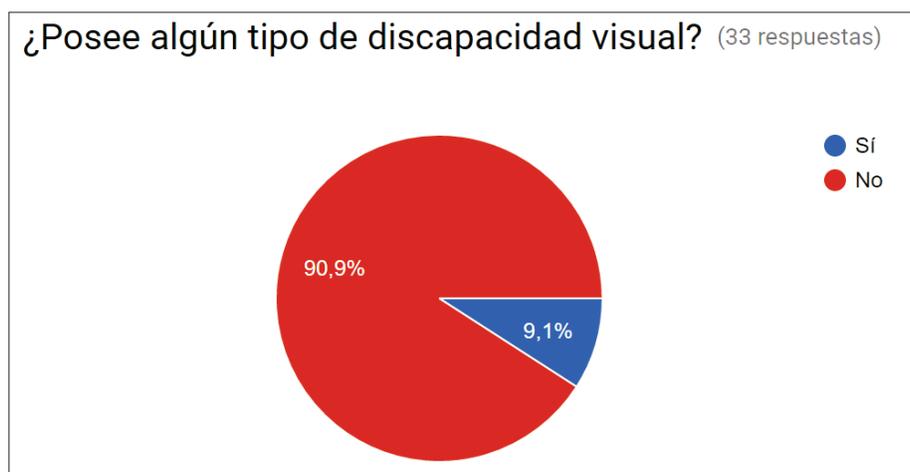


Figura 4.2: Resultados obtenidos en la pregunta: ¿Posee algún tipo de discapacidad visual?

Para la información respecto al uso de la aplicación, presentados en las figuras 4.3, 4.4, 4.5; 1 es *se necesita un cambio total* y 5 es *perfecto*.

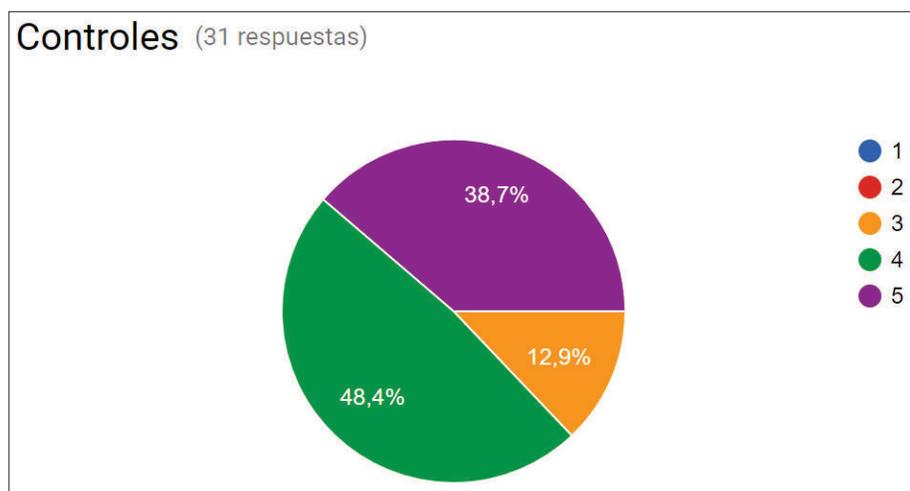


Figura 4.3: Resultados obtenidos en la pregunta: Información respecto al uso de la aplicación - Controles.

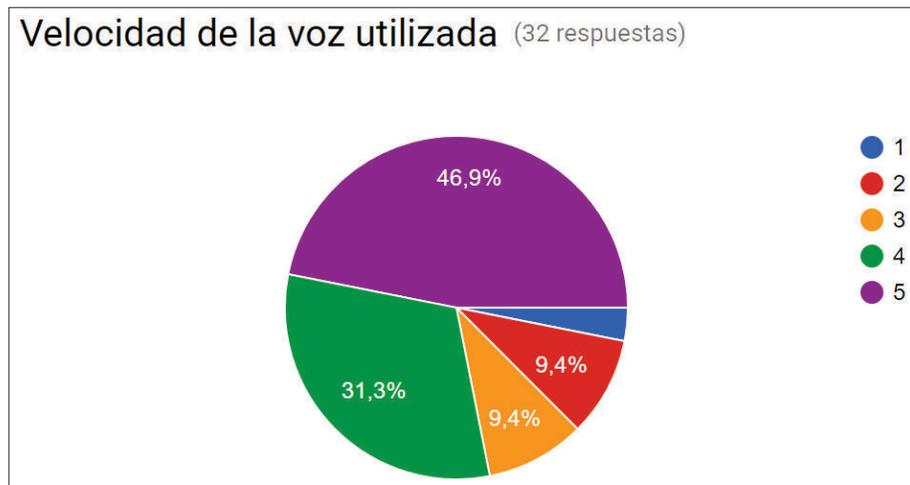


Figura 4.4: Resultados obtenidos en la pregunta: Información respecto al uso de la aplicación - Velocidad de la voz utilizada.

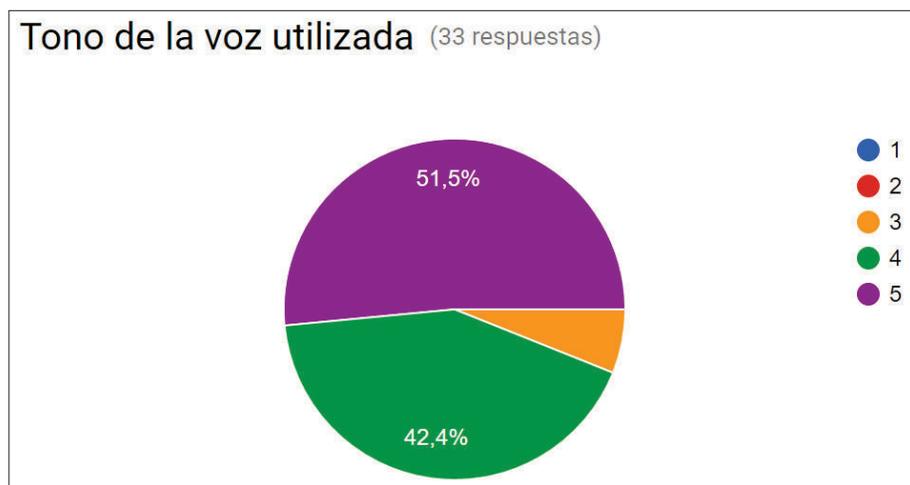


Figura 4.5: Resultados obtenidos en la pregunta: Información respecto al uso de la aplicación - Tono de la voz utilizada.

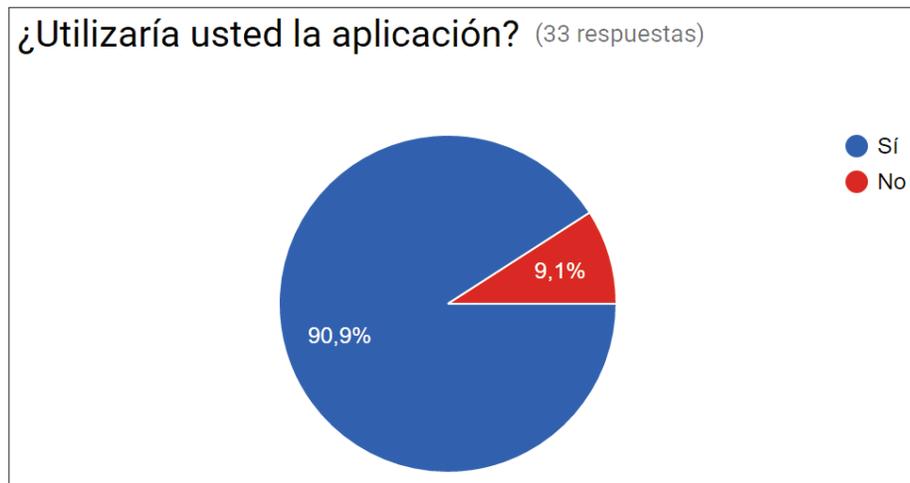


Figura 4.6: Resultados obtenidos en la pregunta: ¿Utilizaría usted la aplicación?

Estos resultados reflejan que:

- El 9,1 % de los encuestados refleja algún tipo de discapacidad visual.
- Con aspectos a los elementos del prototipo funcional considerados para evaluación: controles, velocidad de la voz, y tono de la voz; más del 75 % de los encuestados están satisfecho con este. Sin embargo el elemento que más críticas recibió fue la velocidad de la voz utilizada.
- El 90,9 % de los encuestados utilizaría la aplicación.

A partir de la retroalimentación ofrecida, se implementaron los siguientes cambios:

- Que con las teclas + y - sea posible cambiar la velocidad de la voz.
- Cambios en los controles para que las flechas izquierda y derecha sean exclusivas para las preguntas, y flecha arriba y abajo para opciones.

4.3 RESULTADOS OBTENIDOS EN LA TOMA DE PRUEBAS CON LA APLICACIÓN PARCIAL

A partir de la retroalimentación obtenida en el evento Hábitat-EPN, se realizaron cambios en la aplicación que incluyeron cambios en los controles y la configuración

de la velocidad de la voz por parte del usuario. Con estos cambios ya realizados, se deseaba probar la aplicación, para lo cual, se estableció una reunión con la CONADIS, en la cual nos dieron referencias del Instituto Mariana de Jesús.

El Instituto Mariana de Jesús es un instituto de educación para personas con discapacidades, con especialidad en la discapacidad visual. La evidencia de la reunión se presenta en el anexo 4.2. En el instituto se logró tomar una prueba con la aplicación al estudiante Wilson Cajamarca con número de cédula 1401080427, el cual presentaba una discapacidad visual total (no vidente). Adicional a esta prueba tomada, se tomó otra prueba a un ex estudiante de este instituto, Jorge Bassante con número de cédula 0801431420, el cual también presentaba una discapacidad visual total (no vidente).

La toma de pruebas utilizando el prototipo funcional se realizó bajo el siguiente escenario:

- Se tomó la prueba en un ambiente cerrado, sin ruido externo, utilizando los parlantes de la PC.
- La prueba fue rendida utilizando una PC portátil.

La prueba tomada con la aplicación (hasta ese momento) constó de seis preguntas, de las cuales, dos eran de opción múltiple, dos de verdadero y falso, y dos de completar el espacio en blanco. La última pregunta no tenía respuesta correcta y era utilizada para recabar la opinión de los usuarios. Estas preguntas fueron:

1. Uno más uno es igual a: 1) Tres, 2) Dos, 3) Uno, 4) Cuatro.
2. Cuatro menos tres es igual a: 1) Dos, 2) Tres, 3) Uno, 4) Nueve.
3. Las regiones del Ecuador son: costa, *espacio en blanco*, amazonia. 1) Sierra, 2) Argentina, 3) Bogotá, 4) Lima.
4. Quito es una *espacio en blanco* del Ecuador. 1) Región, 2) País, 3) Ciudad, 4) Continente.
5. La capital de Ecuador es Quito. 1) Verdadero, 2) Falso.

6. ¿Te gustó la aplicación? 1) Verdadero, 2) Falso.

Tabla 4.2: Resultados de la toma de pruebas con la versión parcial de la aplicación.

Pregunta	Opción 1	Opción 2	Opción 3	Opción 4	Total
1	0	2	0	0	2
2	0	0	2	0	2
3	2	0	0	0	2
4	0	0	2	0	2
5	2	0	-	-	2
6	2	0	-	-	2

A pesar de que el universo al que se le tomó la prueba estuvo constituido únicamente por dos personas, estas al ser no videntes, ofrecen resultados reales de la aplicación desarrollada. Con los resultados obtenidos se puede apreciar que ambas personas pudieron rendir la prueba respondiendo con certeza el 100 % de las preguntas, logrando con esto una calificación perfecta.

A partir de la retroalimentación ofrecida por ellos, se implementaron los siguientes cambios:

- Que con la tecla CTRL sea posible pausar y reanudar el audio.
- Indicar la totalidad de las opciones disponibles por pregunta.

4.4 RESULTADOS OBTENIDOS EN LA TOMA DE PRUEBAS CON LA APLICACIÓN FINAL

Se realizó la toma de una prueba para validar el funcionamiento de la aplicación final con todos los cambios, a los estudiantes pertenecientes a dos cursos (Arquitectura de Computadores y Calidad de Software) del Ing. Andrés Larco. Para la retroalimentación del uso de la aplicación se consideraron los resultados obtenidos por los participantes en la prueba, y una encuesta que se les realizó a estos.

La toma de pruebas utilizando la aplicación final se realizó bajo el siguiente escenario:

- La prueba fue rendida sin monitor, para asegurar de que en caso de la persona que rinde la prueba no tenga deficiencia visual, simule la situación de una persona no vidente.
- El audio fue transmitido utilizando parlantes o audífonos dependiendo del gusto de la persona evaluada.
- La prueba fue rendida utilizando una PC de escritorio.

La prueba tomada con la aplicación final constó de seis preguntas, de las cuales, dos eran de opción múltiple, dos de verdadero y falso, y dos de completar el espacio en blanco. La última pregunta no tenía respuesta correcta y era utilizada para recabar la opinión de los usuarios. Estas preguntas fueron:

1. ¿Cuál de las siguientes no es una región del Ecuador? 1) Sierra, 2) Arábica, 3) Amazonía, 4) Costa.
2. La mitad del mundo se encuentra en Ecuador. 1) Verdadero, 2) Falso.
3. Quito es la capital de Ecuador. 1) Verdadero, 2) Falso.
4. La capital de *espacio en blanco* es Quito. 1) Ecuador, 2) Colombia, 3) Perú, 4) Chile.
5. Las islas *espacio en blanco* se encuentran en la región insular del Ecuador. 1) Filipinas, 2) Galápagos, 3) Hawái, 4) Malvinas.
6. ¿Recomendarías esta aplicación? 1) Sí, 2) No.

Se contó con la participación de 44 personas, las cuales rindieron la prueba utilizando la aplicación y realizaron la encuesta. Los resultados obtenidos en la prueba realizada pueden ser constatados en el anexo A.10. Estos fueron los siguientes:

- Curso de Calidad de Software:**

- ✧ Participaron 21 estudiantes.
- ✧ El puntaje promedio de la prueba fue de 0.94 sobre 1.
- ✧ 19 estudiantes obtuvieron el puntaje de 1.
- ✧ 1 estudiante obtuvo un puntaje de 0.
- ✧ 1 estudiante obtuvo un puntaje de 0,83.

□ **Curso de Arquitectura de Computadores:**

- ✧ Participaron 23 estudiantes.
- ✧ El puntaje promedio de la prueba fue de 1 sobre 1.
- ✧ 23 estudiantes obtuvieron el puntaje de 1.

A partir de los datos recopilados de la prueba rendida por los estudiantes, se puede concluir que es factible para una persona no vidente, rendir pruebas utilizando la aplicación final.

Toda la información relacionada a la encuesta, incluyendo los resultados, se pueden observar en el anexo A.9. Las preguntas más importantes con sus resultados son las siguientes:



Figura 4.7: Resultados obtenidos en la pregunta: ¿Posee algún tipo de discapacidad visual?.

Para la información respecto al uso de la aplicación, presentados en las figuras 4.8, 4.9, 4.10; 1 es *se necesita un cambio total* y 5 es *perfecto*.

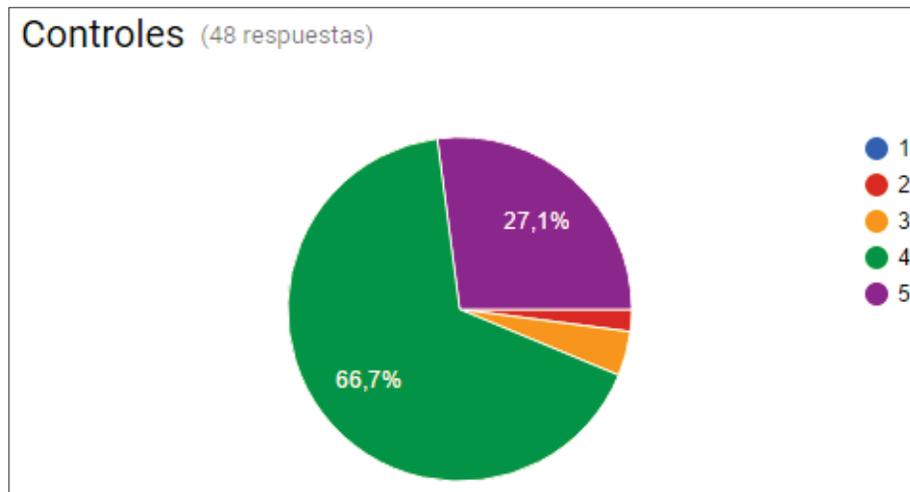


Figura 4.8: Resultados obtenidos en la pregunta: Información respecto al uso de la aplicación - Controles.



Figura 4.9: Resultados obtenidos en la pregunta: Información respecto al uso de la aplicación - Velocidad de la voz utilizada.



Figura 4.10: Resultados obtenidos en la pregunta: Información respecto al uso de la aplicación - Tono de la voz utilizada.

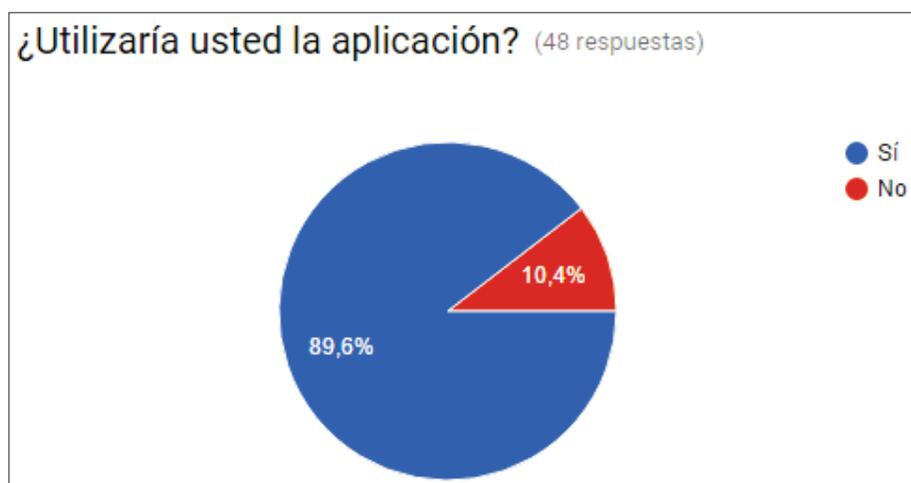


Figura 4.11: Resultados obtenidos en la pregunta: Información respecto al uso de la aplicación - ¿Utilizaría usted la aplicación?

Estos resultados reflejan que:

- El 8,7 % de los encuestados reflejaba algún tipo de discapacidad visual.
- El 93,7 % de los encuestados está conforme con los controles de la aplicación.
- El 89,4 % de los encuestados utilizaría la aplicación.

4.5 TRABAJOS FUTUROS

Se podrían implementar mas funcionalidades a la aplicación final, estas podrían ser:

- Permitir el manejo de varios idiomas para las pruebas, entre ellos inglés, italiano, portugués. Ya que estos idiomas ya se encuentran en la Web Speech API de Google.
- Implementación de los controles para rendir pruebas, cuando el dispositivo que utiliza el usuario es un teléfono móvil.

CAPÍTULO 5

CONCLUSIONES Y RECOMENDACIONES

5.1 CONCLUSIONES

- ❑ El 100% de las personas no videntes que utilizaron la aplicación, pudieron rendir una prueba a satisfacción.
- ❑ Los tipos de preguntas que pueden ser resueltas por personas no videntes utilizando la aplicación desarrollada, considerando como máximo número de cambios de dirección uno y que sea una pregunta objetiva, son preguntas de opción múltiple, de verdadero o falso, y de completar el espacio en blanco.
- ❑ En todas las pruebas rendidas por personas sin utilizar la visión a través de las distintas versiones de la aplicación, se obtuvieron puntajes promedios superiores al 93%, asegurando con esto que es posible para una persona, tanto vidente o no vidente, rendir pruebas sin utilizar la visión a través de la aplicación desarrollada.
- ❑ El módulo pruebas, al poseer las operaciones CRUD, facilita la creación y actualización de pruebas por parte de los docentes, adicionalmente, facilita la calificación al hacerlo de una manera automática.
- ❑ El uso de Scrum garantizó el desarrollo metodológico de la aplicación, con lo cual se obtuvo un total de seis Sprints, con 42 historias de usuarios completadas, invirtiendo 483 horas de desarrollo de la aplicación, sin contar el tiempo invertido en el desarrollo del presente documento.

- ❑ La reutilización de código es parte del desarrollo de cualquier tipo de aplicación, en este caso puntual, se reutilizó el Web Speech API de Google para realizar la síntesis del habla (texto a voz) y reconocimiento de la voz (voz a texto).
- ❑ Para que la aplicación desarrollada funcione correctamente se debe utilizar la versión 33 de Google Chrome o superior.
- ❑ La infraestructura en la cual fue desarrollada la aplicación es el servicio PaaS Google App Engine ofrecido por Google, el cual escala la cantidad de recursos asignados a la aplicación en base al número de peticiones que existen sobre la misma, con lo cual el desarrollador se dedica solo a programar y no se preocupa del dimensionamiento del hardware.
- ❑ La presente aplicación cuenta con una conexión segura, debido a que todas las aplicaciones desplegadas en Google App Engine, ofrecen por defecto una conexión HTTP sobre SSL, es decir HTTPS.
- ❑ Los servicios de la nube no son gratuitos, el mayor costo facturado por estos servicios es el proveniente de la base de datos, es decir, de Google Cloud SQL, superiores inclusive al de Google App Engine.
- ❑ La herramienta para evaluar el progreso hacia la meta Burn-Up permite detectar si existió un cambio entre el tiempo planificado para el Sprint, y el tiempo realmente invertido, logrando con esto precedentes para las estimaciones en los Sprints futuros.
- ❑ Al poseer el mismo lenguaje de programación, tanto a nivel de servidor como de cliente, como es el caso cuando se utilizaba NodeJS a nivel del servidor, la reutilización de código es directa, sin necesidad de cambios.

5.2 RECOMENDACIONES

- ❑ La conexión a Internet utilizada por los estudiantes debe ser estable para evitar demoras entre las acciones del estudiante y el audio que estas acciones

generan.

- ❑ La aplicación se encuentra actualmente en los servidores de Google App Engine, a través de un cambio de propietario la EPN podría mantener la aplicación sin cambios adicionales necesarios.
- ❑ El estudiante debe estar familiarizado con la posición de las teclas en el teclado, para evitar con esto que presione teclas no deseadas.
- ❑ Se puede utilizar Scrum no únicamente en proyectos de desarrollo de software, sino también en proyectos de otra índole.
- ❑ Recordar que en Scrum existen varias herramientas para evaluar el progreso hacia la meta como es el caso de Burn-Up, no únicamente Burn-Down.
- ❑ El uso de plataformas existentes en la nube, como el caso de Google Cloud Platform, debe ser enseñando con mayor profundidad en la carrera.
- ❑ Utilizar un repositorio para el manejo de versiones como Github, para llevar un estado de todas las versiones de la aplicación.

SIGLAS Y ACRÓNIMOS

A | C | D | E | H | I | J | M | O | P | R | S | U | W | X

A

API Application Programming Interface. 40, 44, 45, 48, 116, 120, 125, 129, *Glosario*: Application Programming Interface

C

CONADIS Consejo Nacional para la Igualdad de Discapacidades. 12, 14, 110, 120, *Glosario*: Consejo Nacional para la Igualdad de Discapacidades

CRUD Create, Read, Update, Delete. 32, 36, 120, *Glosario*: Create, Read, Update, Delete

CSS Cascading Style Sheets. 34, 35, 39, 49, 101, 120, *Glosario*: Cascading Style Sheets

D

DBMS Database Management System. 48, 120, *Glosario*: Database Management System

E

EPN Escuela Politécnica Nacional. 105, 120, *Glosario*: Escuela Politécnica Nacional

H

HTML HyperText Markup Language. 33–35, 38, 39, 49, 63, 64, 69, 71, 74, 77–80, 82–84, 87–89, 91–95, 97–100, 120, 130, *Glosario*: HyperText Markup Language

HTTP Hypertext Transfer Protocol. 29, 30, 32, 35, 36, 60, 70–72, 75, 78–84, 88–93, 98–100, 121, 130, *Glosario: Hypertext Transfer Protocol*

I

IAAS Infrastructure as a Service. 42, 121, *Glosario: Infrastructure as a Service*

IEIA Instituto de Evaluación e Ingeniería Avanzada. 4, 121, *Glosario: Instituto de Evaluación e Ingeniería Avanzada*

IETF Internet Engineering Task Force. 29, 30, 121, *Glosario: Internet Engineering Task Force*

INEC Instituto Nacional de Estadísticas y Censos. 9, 121, *Glosario: Instituto Nacional de Estadísticas y Censos*

J

JAWS Job Access With Speech. 5, 121, *Glosario: Job Access With Speech*

JS JavaScript. 38–41, 48, 63, 64, 69–75, 77–96, 98–102, 121, 127, *Glosario: JavaScript*

JSON JavaScript Object Notation. 49, 121, *Glosario: JavaScript Object Notation*

M

MIT Massachusetts Institute of Technology. 4, 121, *Glosario: Massachusetts Institute of Technology*

MVC Model-View-Controller. 32, 33, 36, 40, 121, 129, *Glosario: Model-View-Controller*

O

OMS Organización Mundial de la Salud. 10, 121, *Glosario: Organización Mundial de la Salud*

ONCE Organización Nacional de Ciegos Españoles. 2

OWASP Open Web Application Security Project. 70, 122, 123, *Glosario*: Open Web Application Security Project

P

PAAS Platform as a Service. 42, 43, 122, 125, *Glosario*: Platform as a Service

PC Personal Computer. 49, 105, 110, 112, 122, *Glosario*: Personal Computer

R

REST Representational State Transfer. 31, 48, 122, *Glosario*: Representational State Transfer

RFC Requests for Comments. 29, 122, *Glosario*: Requests for Comments

S

SAAS Software as a Service. 42, 101, 122, *Glosario*: Software as a Service

SOA Service-Oriented Architecture. 32, 122, *Glosario*: Service-Oriented Architecture

SPA Single Page Application. 33, 34, 36, 48, 122, *Glosario*: Single Page Application

U

URI Uniform Resource Identifier. 30, 31, 122, *Glosario*: Uniform Resource Identifier

W

W3C World Wide Web Consortium. 22, 32, 38, 39, 122, *Glosario*: World Wide Web Consortium

WATI Wisconsin Assistive Technology Initiative. 17, 122, *Glosario*: Wisconsin Assistive Technology Initiative

WWW World Wide Web. 122, 130, *Glosario*: World Wide Web

X

XP Extreme Programming. 52, 122, *Glosario*: Extreme Programming

GLOSARIO

A | B | C | D | E | G | H | I | J | M | N | O | P | R | S | U | W

A

AngularJS

Framework estructural para aplicaciones web dinámicas [24].. 39, 48, 49

Application Programming Interface

Conjunto de directivas que pueden ser usadas para tener una pieza de software funcionando con algunas otras [38].. 40, 120

Atom

Editor de texto de código abierto, configurable con soporte para varios lenguajes de programación.. 45

B

Base de datos

Colección de datos interrelacionados almacenados juntos en uno o mas archivos computarizados [39].. 48

Bcrypt

Algoritmo para la encriptación en una vía de claves, recomendado por OWASP [37].. 70

Bootstrap

Framework para el desarrollo de páginas web responsivas y adaptables al móvil.. 39, 49

C

Cascading Style Sheets

Lenguaje utilizado para describir la presentación de documentos HTML o XML [23].. 39, 120

Consejo Nacional para la Igualdad de Discapacidades

Institución que se encarga de formular, transversalizar, observar, realizar el seguimiento y la evaluación de las políticas públicas en discapacidades, en todo el territorio nacional, en todos los niveles de gobierno y en los ámbitos público y privado; con el fin de asegurar la plena vigencia y el ejercicio de los derechos de las personas con discapacidad y sus familias; promoviendo, impulsando, protegiendo y garantizando el respeto al derecho de igualdad y no discriminación, de acuerdo a la Constitución de la República, los Instrumentos Internacionales vigentes, la Ley Orgánica de Discapacidades y de los Consejos Nacionales para la Igualdad.. 12, 120

Create, Read, Update, Delete

Representa las operaciones básicas; crear, leer, actualizar y eliminar; que se pueden realizar sobre una base de datos.. 32, 120

D

Daily Scrum

Evento de Scrum con duración de quince minutos, o menos, para sincronizar y planear, por parte del Development Team, el siguiente día de trabajo de desarrollo durante el Sprint [36]. Si existen cambios, estos deben reflejarse en el Sprint Backlog.. 55, 56

Database Management System

Producto usado para el almacenamiento y organización de datos que tienen usualmente formatos y estructuras definidas [40].. 48, 120

Development Team

Rol de Scrum responsable de administrar, organizar y realizar todo el trabajo requerido para crear un Incremento del producto cada Sprint [36].. 53, 56, 57

E

Escuela Politécnica Nacional

Universidad pública del Ecuador clase A.. 105, 120

Extreme Programming

Es uno de algunos procesos ágiles [41].. 52, 122

G

Github

Plataforma que ofrece un repositorio de código para el manejo del versionamiento de este y trabajo colaborativo, basado en Git.. 46

Google App Engine

PAAS ofrecido por Google para el hosteo de aplicaciones.. 42, 48, 50, 67

Google Cloud Platform

Plataforma de la nube de Google, es el nombre que lleva la plataforma que contiene a todos los servicios ofrecidos por Google relacionados con la Nube.. 41, 66

Google Cloud SQL

Servicio de base de datos completamente administrada que permite una fácil configuración, mantenimiento, manejo, y administración de bases de datos relacionales MySQL [29].. 43, 48, 50, 67

Google Speech Recognition

API de Google que se encarga del reconocimientos de voz.. 45, 48

Google Speech Synthesis

API de Google que se encarga de la síntesis del habla.. 44, 49

H

HyperText Markup Language

Lenguaje estándar utilizado para crear páginas web y sus elementos forman los bloques de construcción de todas las páginas web [21].. 33, 120

Hypertext Transfer Protocol

Protocolo de nivel de aplicación para sistemas distribuidos, colaborativos, información hipermedia [13].. 29, 121

I**Infrastructure as a Service**

Ofrecimiento de los recursos de computo, complementados con capacidades de almacenamiento y red, que son hospedados por el proveedor de servicios y ofrecidos en base a la demanda a los consumidores [42].. 42, 121

Instituto de Evaluación e Ingeniería Avanzada

Órgano Evaluador Externo ante la Dirección General de Profesiones y Órgano Evaluador ante COPEEMS del Sistema Nacional De Bachillerato.. 4, 121

Instituto Nacional de Estadísticas y Censos

Órgano rector de la estadística nacional y el encargado de generar las estadísticas oficiales del Ecuador para la toma de decisiones en la política pública.. 9, 121

Internet Engineering Task Force

Comunidad internacional abierta conformada por diseñadores de red, operadores, vendedores, e investigadores preocupados por la evolución de la arquitectura de Internet y la funcionamiento de este [43].. 29, 121

J**JavaScript**

Lenguaje ligero e interpretado, orientado a objetos con funciones de primera clase, más conocido como el lenguaje de script para páginas web, pero también usado en muchos entornos sin navegador [20].. 38, 39, 121

JavaScript Object Notation

Formato ligero para el intercambio de datos basado en texto e independiente del lenguaje [44].. 49, 121

Job Access With Speech

Lector de pantalla para Windows. 5, 121

M**Massachusetts Institute of Technology**

Instituto que se encarga de la investigación de nuevos conocimientos y educación de los estudiantes en ciencia, tecnologías y otras áreas que mejor servirán a la nación y el mundo en el siglo.. 4, 121

MochaJS

Framework de JS que permite el desarrollo y ejecución de pruebas utilizando NodeJS.. 40, 48

Model-View-Controller

Patrón de arquitectura.. 32, 121

MySQL

Base de datos de código abierto más popular del mundo [30].. 44, 50

N**NodeJS**

Entorno de ejecución para JavaScript construido con el motor de JavaScript V8 de Chrome [26].. 40, 41, 48, 50, 70, 127, 129

O**Open Web Application Security Project**

Organización mundial sin fines de lucros dedicada a mejorar la seguridad de software.. 70, 122

Organización Mundial de la Salud

Autoridad directiva y coordinadora en asuntos de sanidad internacional en el sistema de las Naciones Unidas.. 10, 121

P**Personal Computer**

Microcomputador destinado principalmente para el uso autónomo por parte de un individuo [39].. 49, 122

Platform as a Service

Ofrecimiento de una colección de servicios de infraestructura (middleware) para aplicaciones que incluyen plataforma para aplicaciones, integración, manejo de los procesos de negocio y servicios de base de datos [28].. 42, 122

Product Backlog

Lista ordenada ordenada de todo lo que pueda ser necesitado por el producto, es decir, todos los requerimientos [35].. 53, 55–57

Product Owner

Rol de Scrum responsable de maximizar el valor de un producto, mediante el manejo incremental y expresando las expectativas funcionales y de negocio de un producto hacia el Development Team [36].. 53, 55–57

Protocolo

Conjunto de convenciones que gobiernan la interacción de procesos, dispositivos, y otros componentes dentro de un sistema [39].. 29

R**Reactivo**

Elemento de una prueba formado por una pregunta y un conjuntos de opciones disponibles o espacio para escribir la respuesta.. 24

Reactivo objetivo

Reactivo que contiene al menos una opción correcta, y el criterio de aceptación es una acción objetivo como la selección.. 24

Representational State Transfer

Estilo arquitectónico que especifica restricciones, la cuales, aplicadas a un servicio web, inducen a las propiedades deseadas que permiten a los servicios trabajar de la mejor forma en la web [16].. 31, 122

Requests for Comments

Serie de documentos que contienen notas técnicas y organizacionales acerca del Internet [45].. 29, 122

S**Sails**

Framework MVC más popular para NodeJS, diseñado para emular patrones de MVC con soporte para los requerimientos de aplicaciones modernas utilizando una API escalables basadas en datos y una arquitectura orientada a servicios [25].. 40, 48, 50

Scrum

Framework dentro del cual las personas pueden abordar problemas adaptativos complejos, mientras que productivamente y creativamente se entregan productos del mayor valor posible [35].. 52–54, 58, 129

Scrum Master

Rol de Scrum responsable de guiar, enseñar y asistir al Scrum Team y sus alrededores de lograr un entendimiento correcto del uso de Scrum [36].. 54–56

Service-Oriented Architecture

Conjunto de componentes los cuales pueden ser invocados, y cuyas descripción de las interfaces pueden ser publicadas y descubiertas [17].. 32, 122

Servicio web

Aplicaciones cliente y servidor que se comunican a través de la World Wide Web (WWW) usando el protocolo HTTP [46].. 31, 32, 36, 40

Single Page Application

Aplicaciones web que cargan una sola página HTML y actualiza dinámicamente la página a medida que el usuario interactúa con la aplicación [19].. 33, 122

Software as a Service

Software que es de propiedad, entregado y manejado remotamente por uno o más proveedores [47].. 42, 122

Sprint

Evento de Scrum con duración de treinta días, o menos, en el cual un Incremento es creado, y sirve como contenedor para otros eventos y actividades Scrum. Los Sprints deben ser ejecutados consecutivamente sin intermitencias [36].. 54–57, 60, 62–68, 103

Sprint Backlog

Subconjunto de ítems del Product Backlog, que contienen los requerimientos a ser implementados y generar un Incremento.. 55–57

Sprint Planning

Evento de Scrum con duración de ocho horas, o menos, el cual da comienzo a un Sprint. Se inspecciona el trabajo del Product Backlog que es el más importante para incluirlo en el Sprint Backlog [36].. 54–56

Sprint Retrospective

Evento de Scrum con duración de tres horas, o menos, realizado después del Sprint Review y antes del Sprint Planning, para inspeccionar el Sprint anterior y desarrollar un plan de mejoras de ser necesario [36].. 54, 56

Sprint Review

Evento de Scrum con duración de cuatro horas, o menos, realizado al final del Sprint para inspeccionar el Incremento y adaptar el Product Backlog de ser necesario [35].. 54, 56

U**Uniform Resource Identifier**

Cadena de caracteres que identifica el recurso sobre el cual aplicar la petición HTTP [13].. 30, 122

W**Wisconsin Assistive Technology Initiative**

Organización fundado por Departamento de Instrucción Pública de Wisconsin, el cual se encarga de proveer entrenamiento y soporte en tecnologías de asistencia dentro de la configuración educacional de Wisconsin.. 17, 122

World Wide Web

Sistema interconectado de páginas web públicas accesibles a través de Internet [48].. 122, 130

World Wide Web Consortium

Comunidad internacional que desarrolla estándares que aseguran el crecimiento de la Web a largo plazo.. 22, 122

BIBLIOGRAFÍA

- [1] RAE, *DLE: discapacitado, da - Diccionario de la lengua española - Edición del Tricentenario*. dirección: <http://dle.rae.es/?id=DrrzNuK> (visitado 12-01-2017).
- [2] ONCE, *Discapacidad visual - aspectos generales*. dirección: <http://www.once.es/new/servicios-especializados-en-discapacidad-visual/discapacidad-visual-aspectos-generales> (visitado 03-02-2017).
- [3] P. Jaroslaw Wiazowski, «Chapter 12 - Assistive Technology for Students who are Blind or Have Low Vision», Wisconsin Department of Public Instruction, inf. téc., 2009, págs. 01-30. dirección: <http://www.wati.org/content/supports/free/pdf/Ch12-Vision.pdf>.
- [4] MIT, *MIT Center for Academic Excellence: Mastering Tests - Different Types of Tests*. dirección: <http://web.mit.edu/uaap/learning/test/testtypes.html> (visitado 19-01-2017).
- [5] Instituto de Evaluación e Ingeniería Avanzada, «Manual de Elaboración de Ítems», inf. téc., 2012. dirección: <http://www.minerd.gob.do/sitios/pnacionales/Documentos%20Pruebas%20Nacionales/MANUAL%20ELABORACION%20DE%20ITEMS%20PRUEBAS%20NACIONALES.pdf>.
- [6] Instituto Nacional de Estadísticas y Censos, «Población por condición de discapacidad, según provincia, cantón, parroquia y área de empadronamiento.», inf. téc., 2010.
- [7] —, «Población menor a 12 años por condición de discapacidad, según provincia, cantón, parroquia y área de empadronamiento», inf. téc., 2010.
- [8] World Health Organization, «GLOBAL DATA ON VISUAL IMPAIRMENTS 2010», inf. téc., 2012. dirección: <http://www.who.int/blindness/GLOBALDATAFINALforweb.pdf>.
- [9] Dadax, *Worldometers - real time world statistics*. dirección: <http://www.worldometers.info/> (visitado 17-01-2017).

- [10] Statista, *The most spoken languages worldwide*. dirección: <https://www.statista.com/statistics/266808/the-most-spoken-languages-worldwide/> (visitado 17-01-2017).
- [11] Consejo Nacional para la Igualdad de Discapacidades (CONADIS), «REGISTRO NACIONAL DE DISCAPACIDADES - INSTITUCIONES», inf. téc., 2014. dirección: http://www.consejodiscapacidades.gob.ec/wp-content/uploads/downloads/2014/05/registro%7B%5C_%7Dnacional%7B%5C_%7Ddiscapacidades%7B%5C_%7Dinstituciones.pdf.
- [12] W. M. Vagias, «Likert-Type Scale Response Anchors», dirección: <http://www.peru.edu/oira/wp-content/uploads/sites/65/2016/09/Likert-Scale-Examples.pdf>.
- [13] IETF, *HTTP protocol specification*. dirección: <https://www.ietf.org/rfc/rfc2616.txt> (visitado 19-01-2017).
- [14] —, *Hypertext Transfer Protocol Version 2 (HTTP/2)*. dirección: <https://tools.ietf.org/html/rfc7540> (visitado 23-01-2017).
- [15] —, *Hypertext Transfer Protocol (HTTP/1.1): Message Syntax and Routing*. dirección: <https://tools.ietf.org/html/rfc7230> (visitado 23-01-2017).
- [16] Oracle, *What Are RESTful Web Services?*, 2013. dirección: <http://docs.oracle.com/javase/6/tutorial/doc/gijqy.html> (visitado 23-01-2017).
- [17] W3C, *Web Services Glossary*. dirección: <https://www.w3.org/TR/ws-gloss/> (visitado 23-01-2017).
- [18] Microsoft, *ASP.NET MVC Overview*. dirección: [https://msdn.microsoft.com/en-us/library/dd381412\(v=vs.108\).aspx](https://msdn.microsoft.com/en-us/library/dd381412(v=vs.108).aspx) (visitado 19-01-2017).
- [19] —, *ASP.NET - Single-Page Applications: Build Modern, Responsive Web Apps with ASP.NET*. dirección: <https://msdn.microsoft.com/en-us/magazine/dn463786.aspx> (visitado 23-01-2017).
- [20] Mozilla Developer Network, *JavaScript | MDN*. dirección: <https://developer.mozilla.org/es/docs/Web/JavaScript> (visitado 23-01-2017).
- [21] W3C, *HTML*. dirección: <https://www.w3.org/html/> (visitado 23-01-2017).

- [22] Mozilla Developer Network, *HTML5 - HTML | MDN*. dirección: <https://developer.mozilla.org/es/docs/HTML/HTML5> (visitado 24-01-2017).
- [23] —, *CSS | MDN*. dirección: <https://developer.mozilla.org/es/docs/Web/CSS> (visitado 23-01-2017).
- [24] Google, *AngularJS: Developer Guide: Introduction*. dirección: <https://docs.angularjs.org/guide/introduction> (visitado 24-01-2017).
- [25] The Sails Company, *Sails.js | Realtime MVC Framework for Node.js*. dirección: <http://sailsjs.com/> (visitado 24-01-2017).
- [26] Node.js Foundation, *Node.js*. dirección: <https://nodejs.org/es/> (visitado 24-01-2017).
- [27] Google, *Building Scalable and Resilient Web Applications on Google Cloud Platform*. dirección: <https://cloud.google.com/solutions/scalable-and-resilient-apps> (visitado 24-01-2017).
- [28] Gartner, *PaaS - Platform as a Service - Gartner IT Glossary*. dirección: <http://www.gartner.com/it-glossary/platform-as-a-service-paas/> (visitado 24-01-2017).
- [29] Google, *Google Cloud SQL Documentation | Cloud SQL | Google Cloud Platform*. dirección: <https://cloud.google.com/sql/docs/> (visitado 24-01-2017).
- [30] Oracle, *MySQL | La base de datos de código abierto más popular*. dirección: <https://www.oracle.com/lad/mysql/index.html> (visitado 24-01-2017).
- [31] Google, *Web apps that talk - Introduction to the Speech Synthesis API*. dirección: https://developers.google.com/web/updates/2014/01/Web-apps-that-talk-Introduction-to-the-Speech-Synthesis-API%7B%5C#%7Dbrowser%7B%5C_%7Dsupport (visitado 24-01-2017).
- [32] A. Stellman y J. Greene, *Learning Agile - UNDERSTANDING SCRUM, XP, LEAN, AND KANBAN*, ISBN: 978-1-449-33192-4.
- [33] Agile Alliance, *Manifesto for Agile Software Development*. dirección: <http://agilemanifesto.org/> (visitado 25-01-2017).

- [34] VersionOne, «State of Agile», inf. téc., 2015. dirección: <https://versionone.com/pdf/VersionOne-10th-Annual-State-of-Agile-Report.pdf>.
- [35] K. Schwaber y J. Sutherland, «The Scrum Guide», inf. téc., 2016.
- [36] Scrum, *Scrum Glossary*. dirección: <https://www.scrum.org/Resources/Scrum-Glossary> (visitado 25-01-2017).
- [37] OWASP, *Password Storage Cheat Sheet*, 2016. dirección: https://www.owasp.org/index.php/Password%7B%5C_%7DStorage%7B%5C_%7DCheat%7B%5C_%7DSheet (visitado 03-02-2017).
- [38] Mozilla Developer Network, *API - Glossary | MDN*. dirección: <https://developer.mozilla.org/en-US/docs/Glossary/API> (visitado 24-01-2017).
- [39] IEEE, «ISO/IEC/IEEE 24765:2010 Systems and software engineering — Vocabulary», inf. téc., 2010.
- [40] Gartner, *DBMS (database management system) - Gartner IT Glossary*. dirección: <http://www.gartner.com/it-glossary/dbms-database-management-system/> (visitado 25-01-2017).
- [41] Don Wells, *Extreme Programming: A Gentle Introduction*. dirección: <http://www.extremeprogramming.org/> (visitado 25-01-2017).
- [42] Gartner, *IaaS - Infrastructure as a Service - Gartner IT Glossary*. dirección: <http://www.gartner.com/it-glossary/infrastructure-as-a-service-iaas/> (visitado 24-01-2017).
- [43] IETF, *About the IETF*. dirección: <https://www.ietf.org/about/> (visitado 19-01-2017).
- [44] ECMA, «Standard ECMA-404 The JSON Data Interchange Format», inf. téc., 2013. dirección: <http://www.ecma-international.org/publications/files/ECMA-ST/ECMA-404.pdf>.
- [45] IETF, *Request for Comments (RFC)*. dirección: <https://www.ietf.org/rfc.html> (visitado 22-01-2017).
- [46] Oracle, *What Are Web Services?*, 2013. dirección: <http://docs.oracle.com/javasee/6/tutorial/doc/gijvh.html> (visitado 23-01-2017).

- [47] Gartner, *SaaS - Software As A Service - Gartner IT Glossary*. dirección: <http://www.gartner.com/it-glossary/software-as-a-service-saas/> (visitado 24-01-2017).
- [48] Mozilla Developer Network, *World Wide Web*. dirección: https://developer.mozilla.org/en-US/docs/Glossary/World%7B%5C_%7DWide%7B%5C_%7DWeb (visitado 23-01-2017).

ANEXO A

DOCUMENTOS GENERADOS

- A.1 ENCUESTA INICIAL REALIZADA**
- A.2 LISTA DE SOLUCIONES TECNOLÓGICAS EXISTENTES AGRUPADAS POR TIPO DE DISCAPACIDAD VISUAL**
- A.3 PLAN DEL PROYECTO DE TITULACIÓN**
- A.4 ENCUESTA SOBRE EL PROTOTIPO FUNCIONAL**
- A.5 RESULTADOS OBTENIDOS DE LA PRUEBA DEL PROTOTIPO FUNCIONAL EN EL EVENTO HÁBITAT-EPN**
- A.6 MANUAL DE USUARIO**
- A.7 MANUAL DE DESPLIEGUE**
- A.8 ACTA DE REUNIÓN CON EL INSTITUTO MARIANA DE JESÚS**
- A.9 ENCUESTA SOBRE LA APLICACIÓN FINAL**
- A.10 RESULTADOS OBTENIDOS DE LA PRUEBA DE LA APLICACIÓN FINAL**
- A.11 DOCUMENTO TÉCNICO JSDOC DE LA DESCRIPCIÓN DE LAS FUNCIONES EXISTENTES EN LOS CONTROLADORES**

ANEXO B

CÓDIGO DE LAS APLICACIONES DESARROLLADAS

B.1 PROTOTIPO FUNCIONAL

B.2 APLICACIÓN FINAL

ANEXO C

MODELOS

C.1 MODELO CONCEPTUAL DE DATOS

C.2 MODELO FÍSICO DE DATOS

C.3 SCRIPT DE CREACIÓN DE LA BASE DE DATOS

ANEXO D

RESULTADOS DE PRUEBAS

D.1 PRUEBAS UNITARIAS

D.2 PRUEBAS DE CALIDAD DE CÓDIGO

D.3 CALIDAD EXTERNA - FACILIDAD DE USO - ISO/IEC 25000

ANEXO E

SCRUM

E.1 PRODUCT BACKLOG

Tabla E.1: Product Backlog.

ID	Tema	Tipo	Como	Yo quiero	con el fin de	Notas	Importancia
PR1	Prototipo funcional	Trabajo técnico	Desarrollador	desarrollar un prototipo funcional que permita rendir una prueba predefinida	obtener feedback por parte de las personas asistentes al evento Hábitat-EPN.	-Recopilar la información del feedback a través de encuestas	0
BD1	Base de datos	Trabajo técnico	Desarrollador	generar el modelo de datos que sustenta a la aplicación	entender la estructura de datos que sustenta a la aplicación.	-Generar el modelo conceptual de datos en Power Designer. -Generar el modelo físico de datos en Power Designer.	1
BD2	Base de datos	Trabajo técnico	Desarrollador	generar el script SQL	crear la base de datos en el DBMS	-Generar el script SQL de creación de la base de datos.	1
CL1	Ambiente de producción	Adquisición de conocimientos	Desarrollador	entender el funcionamiento de Google Cloud Platform	conocer qué requisitos son necesarios para realizar el despliegue de la aplicación en los servidores de Google.	N/A	1
CL2	Ambiente de producción	Trabajo técnico	Desarrollador	configurar el ambiente de producción	poder realizar el despliegue de la aplicación en los servidores de Google.	-Crear el proyecto en Google App Engine -Crear la instancia de base de datos en Google Cloud SQL	1
LO1	Ambiente de desarrollo	Trabajo técnico	Desarrollador	configurar el ambiente de desarrollo	poder desarrollar la aplicación web.	-Instalar librerías y herramientas necesarias para el desarrollo de la aplicación (de ser necesario).	1
MU1	Módulo Usuario	Funcionalidad	Estudiante/Profesor	crear una cuenta con mis datos	ingresar al sistema.	-Validar el formato de los datos. -Almacenar los datos en la base de datos. -Existir una vista para el usuario. -Encriptar los datos sensibles. -Notificar al usuario en caso de error o éxito.	2

Tabla E.2: Cont. Product Backlog.

ID	Tema	Tipo	Como	Yo quiero	con el fin de	Notas	Importancia
MU2	Módulo Usuario	Funcionalidad	Estudiante/Profesor	iniciar sesión escribiendo mi información	ingresar al sistema.	-Validar el formato de los datos. -Existir una vista para el usuario. -Validar la información ingresada por el usuario con la información existente en la base de datos. -Notificar al usuario en caso de error o éxito.	2
MU3	Módulo Usuario	Funcionalidad	Estudiante/Profesor	cambiar mi contraseña	recuperar mi cuenta en el caso de que olvide mi información.	-Validar el formato de los datos. -Existir una vista para el usuario. -Validar la información ingresada por el usuario con la información existente en la base de datos -Notificar al usuario en caso de error o éxito.	2
MU4	Módulo Usuario	Funcionalidad	Estudiante/Profesor	iniciar sesión utilizando la voz	ingresar al sistema.	-Validar el formato de los datos. -Existir una vista para el usuario. -Validar la información ingresada por el usuario con la información existente en la base de datos -Notificar al usuario en caso de error o éxito.	2
MU5	Módulo Usuario	Funcionalidad	Estudiante/Profesor	cerrar sesión	salir del sistema.	-Existir una vista para el usuario.	2
MU6	Módulo Usuario	Funcionalidad	Desarrollador	aplicar la política de estar iniciada la sesión	evitar que personas no autorizadas, utilicen los servicios REST HTTP.	-Existir una vista para el usuario.	2
TE1	Pruebas	Trabajo técnico	Desarrollador	desarrollar pruebas para las historias de usuario MUX	validar el funcionamiento de estas historias de usuario	-Generar pruebas a los servicios HTTP de estas historias. -Las pruebas deben validar todas las bifurcaciones posibles.	2
MC1	Módulo Curso	Funcionalidad	Profesor	crear un curso	guardar la información del curso.	-Existir una vista para el usuario. -Notificar al usuario en caso de error o éxito. -Almacenar los datos en la base de datos.	3

Tabla E.3: Cont. Product Backlog.

ID	Tema	Tipo	Como	Yo quiero	con el fin de	Notas	Importancia
MC2	Módulo Curso	Funcionalidad	Profesor	registrar estudiantes en el curso	agrupar a un conjunto de estudiantes.	-Validar el formato de los datos. -Existir una vista para el usuario. -Notificar al usuario en caso de error o éxito. -Almacenar los datos en la base de datos.	3
MC3	Módulo Curso	Funcionalidad	Profesor	clonar un curso	crear otro curso con la misma información y pruebas.	-Existir una vista para el usuario. -Notificar al usuario en caso de error o éxito. -Almacenar los datos en la base de datos. -Clonar las pruebas pertenecientes al curso.	3
MC4	Módulo Curso	Funcionalidad	Profesor	editar un curso	actualizar la información de este.	-Existir una vista para el usuario. -Notificar al usuario en caso de error o éxito. -Almacenar los nuevos datos en la base de datos.	3
MC5	Módulo Curso	Funcionalidad	Profesor	eliminar un curso	no visualizar la información de este.	-Existir una vista para el usuario. -Notificar al usuario en caso de error o éxito. -Eliminar los datos en la base de datos.	3
MC6	Módulo Curso	Funcionalidad	Estudiante/Profesor	visualizar la información del curso	saber que la información de este.	-Existir una vista para el usuario.	3
MC7	Módulo Curso	Funcionalidad	Estudiante/Profesor	visualizar todos los cursos	saber que en qué cursos estoy registrado o saber los cursos que yo creé	-Existir una vista para el usuario.	3
MC8	Módulo Curso	Funcionalidad	Profesor	eliminar un estudiante del curso al que pertenece	sacar al estudiante del curso	-Existir una vista para el usuario. -Notificar al usuario en caso de error o éxito. -Eliminar los datos en la base de datos.	3

Tabla E.4: Cont. Product Backlog.

ID	Tema	Tipo	Como	Yo quiero	con el fin de	Notas	Importancia
TE2	Pruebas	Trabajo técnico	Desarrollador	desarrollar pruebas para las historias de usuario MCx	validar el funcionamiento de estas historias de usuario	-Generar pruebas a los servicios HTTP de estas historias. -Las pruebas deben validar todas las bifurcaciones posibles.	3
MP1	Módulo Prueba	Funcionalidad	Profesor	crear una prueba	evaluar a los estudiantes.	-Existir una vista para el usuario. -Notificar al usuario en caso de error o éxito. -Almacenar los datos en la base de datos. -Registrar la prueba a todos los estudiantes pertenecientes al curso.	4
MP2	Módulo Prueba	Funcionalidad	Profesor	editar una prueba	actualizar la información de este.	-Existir una vista para el usuario. -Notificar al usuario en caso de error o éxito. -Almacenar los nuevos datos en la base de datos.	4
MP3	Módulo Prueba	Funcionalidad	Profesor	eliminar una prueba	no visualizar la información de este.	-Existir una vista para el usuario. -Notificar al usuario en caso de error o éxito. -Eliminar los datos en la base de datos.	4
MP4	Módulo Prueba	Funcionalidad	Estudiante/Profesor	conocer la lista de todas las pruebas	saber todas las pruebas que se/me están tomando, están en ejecución y se tomaron.	-Existir una vista para el usuario. -Agrupar las pruebas por tipo. -Existir controles en el teclado que me permitan navegar por la vista (solo siendo Estudiante). -Transformar el texto a voz de los componentes por los cuales el usuario está navegando (solo siendo Estudiante).	4

Tabla E.5: Cont. Product Backlog.

ID	Tema	Tipo	Como	Yo quiero	con el fin de	Notas	Importancia
MP5	Módulo Prueba	Funcionalidad	Estudiante/Profesor	conocer la lista de todas las pruebas pertenecientes al curso	saber todas las pruebas que se están tomando, están en ejecución y se tomaron en ese curso.	<ul style="list-style-type: none"> -Existir una vista para el usuario. -Agrupar las pruebas por tipo. -Existir controles en el teclado que me permitan navegar por la vista (solo siendo Estudiante). -Transformar el texto a voz de los componentes por los cuales el usuario está navegando (solo siendo Estudiante). 	4
MP6	Módulo Prueba	Funcionalidad	Estudiante	rendir una prueba	plasmear mis conocimientos y obtener un puntaje.	<ul style="list-style-type: none"> -Existir una vista para el usuario. -Almacenar los datos de las preguntas seleccionadas en la base de datos. -Generar un puntaje para la prueba rendida. -Existir controles en el teclado que me permitan navegar por la vista. -Transformar el texto a voz de los componentes por los cuales el usuario está navegando. 	4
MP7	Módulo Prueba	Funcionalidad	Estudiante	obtener una retroalimentación de la prueba rendida	saber en qué preguntas acerté y en cuales no.	<ul style="list-style-type: none"> -Existir una vista para el usuario. -Almacenar los datos de las preguntas seleccionadas en la base de datos. -Generar un puntaje para la prueba rendida. -Existir controles en el teclado que me permitan navegar por la vista. -Transformar el texto a voz de los componentes por los cuales el usuario está navegando. 	4
MP8	Módulo Prueba	Funcionalidad	Profesor	clonar una prueba	copiar todas las preguntas de esta en una nueva prueba.	<ul style="list-style-type: none"> -Existir una vista para el usuario. -Notificar al usuario en caso de error o éxito. -Almacenar los datos de la nueva prueba en la base de datos. 	4

Tabla E.6: Cont. Product Backlog.

ID	Tema	Tipo	Como	Yo quiero	con el fin de	Notas	Importancia
TE3	Pruebas	Trabajo técnico	Desarrollador	desarrollar pruebas para las historias de usuario MPx	validar el funcionamiento de estas historias de usuario	-Generar pruebas a los servicios HTTP de estas historias. -Las pruebas deben validar todas las bifurcaciones posibles.	4
MR1	Módulo Pregunta	Funcionalidad	Profesor	crear una pregunta con sus opciones en una prueba	poder evaluar el estudiante en un tópico específico.	-Existir una vista para el usuario. -Almacenar los datos de la pregunta de la prueba en la base de datos.	5
MR2	Módulo Pregunta	Funcionalidad	Profesor	editar una pregunta de una prueba	cambiar el texto de la pregunta y sus opciones disponibles.	-Existir una vista para el usuario. -Almacenar los nuevos datos de la pregunta de la prueba en la base de datos.	5
MR3	Módulo Pregunta	Funcionalidad	Profesor	eliminar una pregunta de una prueba	no visualizar la información de esta.	-Existir una vista para el usuario. -Notificar al usuario en caso de error o éxito. -Eliminar los datos en la base de datos.	5
MR4	Módulo Pregunta	Funcionalidad	Profesor	clonar una pregunta de una prueba	copiar y registrar la información de esta en una prueba	-Existir una vista para el usuario. -Notificar al usuario en caso de error o éxito. -Almacenar los datos de la pregunta clonada en la base de datos.	5
TE4	Pruebas	Trabajo técnico	Desarrollador	desarrollar pruebas para las historias de usuario MRx	validar el funcionamiento de estas historias de usuario	-Generar pruebas a los servicios HTTP de estas historias. -Las pruebas deben validar todas las bifurcaciones posibles.	5
TC1	Pruebas	Trabajo técnico	Desarrollador	probar la calidad del código fuente	validar la calidad de este	-Generar un documento con los resultados obtenidos	6
TI1	Pruebas	Trabajo técnico	Desarrollador	probar la facilidad de uso de la aplicación	determinar en el grado que se cumple la característica de la norma ISO/IEC 25000	-Generar un documento con los resultados obtenidos	6

Tabla E.7: Cont. Product Backlog.

ID	Tema	Tipo	Como	Yo quiero	con el fin de	Notas	Importancia
JD1	Document o técnico	Trabajo técnico	Desarrollador	crear un documento técnico con la descripción de las funciones de los controladores de la aplicación	facilitar el entendimiento de todas las funciones existentes en la aplicación	-Generar un documento JSDoc	6
AD1	Nombre APP	Otros	Desarrollador	definir el nombre de la aplicación desarrollada	identificar a la aplicación	-Nombre debe ser en inglés	6
MA1	Manual	Trabajo técnico	Desarrollador	generar manual de despliegue	explicar los pasos a seguir para realizar el despliegue de la aplicación en los servidores de Google.	N/A	6
MA2	Manual	Trabajo técnico	Desarrollador	generar manual de usuario	explicar cómo utilizar la aplicación desarrollada.	N/A	6

E.2 SPRINT 1

E.2.1 SPRINT BACKLOG

E.2.2 SPRINT REVIEW

Tabla E.9: Review de las historias de usuario y tareas del Sprint 1.

ID Historia	ID Tarea	Completada	Criterio de aceptación
PR1	PR1-1	x	Rendir una prueba, al seleccionar las opciones correctas y enviar estos resultados, debe comenzar la retroalimentación con el puntaje obtenido y las opciones seleccionadas vs las correctas
PR1	PR1-2	x	
PR1	PR1-3	x	
PR1	PR1-4	x	
PR1	PR1-5	x	
PR1	PR1-6	x	
PR1	PR1-7	x	
PR1	PR1-8	x	
PR1	PR1-9	x	

E.3 SPRINT 2

E.3.1 SPRINT BACKLOG

E.3.2 SPRINT REVIEW

Tabla E.11: Review de las historias de usuario y tareas del Sprint 2.

ID Historia	ID Tarea	Completada	Criterio de aceptación
BD1	BD1-1	x	Que existan los modelos mencionados
BD1	BD1-2	x	
BD2	BD2-1	x	Que exista el script mencionado
CL1	CL1-1	x	Ninguno
CL2	CL2-1	x	Que exista el proyecto en Google App Engine y la instancia de base de datos Google Cloud SQL
CL2	CL2-2	x	
CL2	CL2-3	x	
LO1	LO1-1	x	Ninguno

E.4 SPRINT 3

E.4.1 SPRINT BACKLOG

E.4.2 SPRINT REVIEW

Tabla E.16: Review de las historias de usuario y tareas del Sprint 3.

ID Historia	ID Tarea	Completada	Criterio de aceptación
MU1	MU1-1	x	Que se almacenen en la base de datos los datos ingresados por el usuario a través de la GUI
MU1	MU1-2	x	
MU1	MU1-3	x	
MU1	MU1-4	x	
MU1	MU1-5	x	
MU1	MU1-6	x	
MU1	MU1-7	x	
MU1	MU1-8	x	
MU1	MU1-9	x	
MU1	MU1-10	x	
MU2	MU2-1	x	Que pueda ingresar al sistema con los datos de un usuario registrado
MU2	MU2-2	x	
MU2	MU2-2	x	
MU2	MU2-3	x	
MU2	MU2-4	x	
MU2	MU2-5	x	
MU2	MU2-6	x	Que pueda cambiar la contraseña, preguntas de seguridad y pin actuales a partir del ingreso de las preguntas de seguridad anteriores con sus respectivas respuestas
MU3	MU3-1	x	
MU3	MU3-2	x	
MU3	MU3-3	x	
MU3	MU3-4	x	
MU3	MU3-5	x	
MU3	MU3-6	x	
MU3	MU3-7	x	
MU3	MU3-8	x	
MU3	MU3-9	x	
MU4	MU4-1	x	Que pueda ingresar al sistema dictando el PIN generado cuando se registro el usuario
MU4	MU4-2	x	
MU4	MU4-3	x	
MU4	MU4-4	x	
MU4	MU4-5	x	
MU4	MU4-6	x	
MU4	MU4-7	x	
MU4	MU4-8	x	
MU4	MU4-9	x	
MU4	MU4-10	x	
MU5	MU5-1	x	Que pueda salir

Tabla E.17: Cont. Review de las historias de usuario y tareas del Sprint 3.

ID Historia	ID Tarea	Completada	Criterio de aceptación
MU5	MU5-2	x	del sistema y no quede la sesión del usuario activa
MU6	MU6-1	x	Que solo se puedan consumir los servicios REST HTTP si cumple con la política definida
MU6	MU6-2	x	
MU6	MU6-3	x	
TE1	TE1-1	x	Que se pasen todas las pruebas diseñadas para validar el funcionamiento de los servicios REST HTTP del módulo Usuario
TE1	TE1-2	x	
TE1	TE1-3	x	
TE1	TE1-4	x	
TE1	TE1-5	x	
TE1	TE1-6	x	

E.5 SPRINT 4

E.5.1 SPRINT BACKLOG

E.5.2 SPRINT REVIEW

Tabla E.23: Review de las historias de usuario y tareas del Sprint 4.

ID Historia	ID Tarea	Completada	Criterio de aceptación
MC1	MC1-1	x	Que se almacenen en la base de datos los datos ingresados por el usuario a través de la GUI
MC1	MC1-2	x	
MC1	MC1-3	x	
MC1	MC1-4	x	
MC1	MC1-5	x	
MC1	MC1-6	x	
MC1	MC1-7	x	
MC2	MC2-1	x	Que se almacene en la base de datos la relación del o los estudiantes con el curso, incluyendo todas las pruebas generadas en el curso, ingresando el email del estudiante por el usuario a través de la GUI
MC2	MC2-2	x	
MC2	MC2-3	x	
MC2	MC2-4	x	
MC2	MC2-5	x	
MC2	MC2-6	x	
MC2	MC2-7	x	
MC3	MC3-1	x	Que se copie toda la información en un nuevo curso del curso seleccionado para clonación, inclusive las pruebas de este
MC3	MC3-2	x	
MC3	MC3-3	x	
MC3	MC3-4	x	
MC3	MC3-5	x	
MC3	MC3-6	x	
MC3	MC3-7	x	
MC3	MC3-8	x	
MC3	MC3-9	x	
MC4	MC4-1	x	Que se almacenen en la base de datos los nuevos datos ingresados por el usuario a través de la GUI
MC4	MC4-2	x	
MC4	MC4-3	x	
MC4	MC4-4	x	
MC4	MC4-5	x	
MC4	MC4-6	x	
MC4	MC4-7	x	
MC4	MC4-8	x	
MC4	MC4-9	x	
MC4	MC4-10	x	
MC5	MC5-1	x	

Tabla E.24: Cont. Review de las historias de usuario y tareas del Sprint 4.

ID Historia	ID Tarea	Completada	Criterio de aceptación
MC5	MC5-2	x	Que se elimine de la base de datos al curso seleccionado por el usuario a través de la GUI
MC5	MC5-3	x	
MC5	MC5-4	x	
MC5	MC5-5	x	
MC5	MC5-6	x	
MC5	MC5-7	x	
MC6	MC6-1	x	Que se muestre en la GUI la información del curso
MC6	MC6-2	x	
MC7	MC7-1	x	Que se muestre en la GUI la información de todos los cursos
MC7	MC7-2	x	
MC7	MC7-3	x	
MC7	MC7-4	x	
MC7	MC7-5	x	
MC8	MC8-1	x	Que se elimine de la base de datos la relación del estudiante seleccionado por el usuario a través de la GUI con el curso, incluyendo los datos de las pruebas rendidas seleccionado por el usuario a través de la GUI
MC8	MC8-2		
MC8	MC8-3	x	
MC8	MC8-4	x	
MC8	MC8-5	x	
MC8	MC8-6	x	
TE2	TE2-1	x	Que se pasen todas las pruebas diseñadas para validar el funcionamiento de los servicios REST HTTP del módulo Curso
TE2	TE2-2	x	
TE2	TE2-3	x	
TE2	TE2-4	x	
TE2	TE2-5	x	
TE2	TE2-6	x	
TE2	TE2-7	x	

E.6 SPRINT 5

E.6.1 SPRINT BACKLOG

E.6.2 SPRINT REVIEW

Tabla E.30: Review de las historias de usuario y tareas del Sprint 5.

ID Historia	ID Tarea	Completada	Criterio de aceptación
MP1	MP1-1	x	Que se almacenen en la base de datos los datos ingresados por el usuario a través de la GUI
MP1	MP1-2	x	
MP1	MP1-3	x	
MP1	MP1-4	x	
MP1	MP1-5	x	
MP1	MP1-6	x	
MP1	MP1-7	x	
MP2	MP2-1	x	Que se almacenen en la base de datos los nuevos datos ingresados por el usuario a través de la GUI
MP2	MP2-2	x	
MP2	MP2-3	x	
MP2	MP2-4	x	
MP2	MP2-5	x	
MP2	MP2-6	x	
MP2	MP2-7	x	
MP2	MP2-8	x	
MP2	MP2-9	x	
MP2	MP2-10	x	
MP3	MP3-1	x	Que se elimine el registro de la prueba y los datos relacionados a esta en la base de datos, de la prueba seleccionada a través de la GUI por el usuario
MP3	MP3-2	x	
MP3	MP3-3	x	
MP3	MP3-4	x	
MP3	MP3-5	x	
MP3	MP3-6	x	
MP3	MP3-7	x	
MP4	MP4-1	x	Que se muestre en la GUI la información de todas las pruebas
MP4	MP4-2	x	
MP4	MP4-3	x	
MP4	MP4-4	x	
MP4	MP4-5	x	
MP4	MP4-6	x	
MP5	MP5-1	x	Que se muestre en la GUI la información de todas las pruebas pertenecientes al curso
MP5	MP5-2	x	
MP5	MP5-3	x	
MP5	MP5-4	x	
MP5	MP5-5	x	
MP5	MP5-6	x	
MP6	MP6-1	x	
MP6	MP6-2	x	
MP6	MP6-3	x	

Tabla E.31: Cont. Review de las historias de usuario y tareas del Sprint 5.

ID Historia	ID Tarea	Completada	Criterio de aceptación
MP6	MP6-4	x	Que se pueda rendir la prueba sin utilizar la visión por parte del usuario
MP6	MP6-5	x	
MP6	MP6-6	x	
MP6	MP6-7	x	
MP6	MP6-8	x	
MP6	MP6-9	x	
MP6	MP6-9	x	
MP6	MP6-10	x	
MP6	MP6-11	x	
MP6	MP6-12	x	
MP7	MP7-1	x	Que se pueda conocer en que preguntas se equivoco el usuario y porque, sin utilizar la visión por parte de este
MP7	MP7-2	x	
MP7	MP7-3	x	
MP8	MP8-1	x	Que se copie toda la información en una nueva prueba de la prueba seleccionado para clonación, inclusive las opciones de esta
MP8	MP8-2	x	
MP8	MP8-3	x	
TE3	TE3-1	x	Que se pasen todas las pruebas diseñadas para validar el funcionamiento de los servicios REST HTTP del módulo Curso
TE3	TE3-2	x	
TE3	TE3-3	x	
TE3	TE3-4	x	
TE3	TE3-5	x	
TE3	TE3-6	x	

E.7 SPRINT 6

E.7.1 SPRINT BACKLOG

E.7.2 SPRINT REVIEW

Tabla E.35: Review de las historias de usuario y tareas del Sprint 6.

ID Historia	ID Tarea	Completada	Criterio de aceptación	Responsable
MR1	MR1-1	x	Que se cree una pregunta con sus opciones a través de la GUI de registrar una prueba. Los datos serán almacenados en la base de datos cuando se guarden los cambios en la prueba	Farid
MR1	MR1-2	x		Farid
MR1	MR1-3	x		Farid
MR1	MR1-4	x		Farid
MR1	MR1-5	x		Farid
MR2	MR2-1	x	Que se edite la información de las preguntas utilizando la GUI por parte del usuario. Los datos serán actualizados en la base de datos cuando se guarden los cambios en la prueba	Farid
MR2	MR2-2	x		Farid
MR2	MR2-3	x		Farid
MR2	MR2-4	x		Farid
MR3	MR3-1	x	Que se elimine la información de la base de datos de la pregunta seleccionada a través de la GUI por parte del usuario	Farid
MR3	MR3-2	x		Farid
MR3	MR3-3	x		Farid
MR3	MR3-4	x		Farid
MR3	MR3-5	x		Farid
MR3	MR3-6	x		Farid
MR3	MR3-7	x		Farid
MR4	MR4-1	x	Que se copie la información en la base de datos de una pregunta seleccionada por el usuario a través de la GUI	Farid
MR4	MR4-2	x		Farid
MR4	MR4-3	x		Farid
MR4	MR4-4	x		Farid
MR4	MR4-5	x		Farid
MR4	MR4-6	x		Farid
MR4	MR4-7	x		Farid

TE4	TE4-1	x	Que se pasen todas las pruebas diseñadas para validar el funcionamiento de los servicios REST HTTP del módulo Pregunta	Farid
TE4	TE4-2	x		Farid
TC1	TC1-1	x	Que exista un documento con los resultados	Farid
TC1	TC1-2	x		Farid
TI1	TI1-1	x	Que exista el documento con los resultados	Farid
JS1	JS1-1	x	Que exista el documento JSDoc	Farid
JS1	JS1-2	x		Farid
AD1	AD1-1	x	Que exista el nombre de la aplicación	Farid
MA1	MA1-1	x	Que exista el manual de despliegue	Farid
MA2	MA2-1	x	Que exista el manual de usuario	Farid

Tabla E.36: Cont. Review de las historias de usuario y tareas del Sprint 6.

ANEXO F

CAPTURAS DE PANTALLAS DE LAS APLICACIONES DESARROLLADAS

F.1 PROTOTIPO FUNCIONAL

Figura F.1: Interfaz gráfica historia de usuario PR1 tarea PR1-1.

Pregunta 1: Cuál de las siguientes no es una región del Ecuador

Opcion 1: costa
Opcion 2: amazonía
Opcion 3: arábica
Opcion 4: sierra

Pregunta 2: Cuál de los siguientes animales es usado en el escudo nacional

Opcion 1: cóndor
Opcion 2: rana
Opcion 3: águila
Opcion 4: perro

Pregunta 3: La mitad del mundo se encuentra en Ecuador

Opcion 1: Verdadero
Opcion 2: Falso

Figura F.2: Cont. Interfaz gráfica historia de usuario PR1 tarea PR1-1.

Pregunta 4: La capital de. espacio en blanco. es Quito

Opcion 1: Ecuador
Opcion 2: Colombia
Opcion 3: Perú
Opcion 4: Chile

Pregunta 5: Las islas. espacio en blanco. se encuentran en la región insular del Ecuador

Opcion 1: Filipinas
Opcion 2: Galápagos
Opcion 3: Hawái
Opcion 4: Malvinas

Pregunta 6: Encuentra de utilidad a la aplicación

Opcion 1: Verdadero
Opcion 2: Falso

ENVIAR

Figura F.3: Interfaz gráfica historia de usuario PR1 tarea PR1-7.

Pregunta 1 Cuál de las siguientes no es una región del Ecuador

Respuesta seleccionada: costa
Respuesta correcta: arábica

Pregunta 2 Cuál de los siguientes animales es usado en el escudo nacional

Respuesta seleccionada: cóndor
Respuesta correcta: cóndor

Pregunta 3: La mitad del mundo se encuentra en Ecuador

Respuesta seleccionada: Verdadero
Respuesta correcta: Verdadero

Figura F.4: Cont. Interfaz gráfica historia de usuario PR1 tarea PR1-7.

Pregunta 4: La capital de. espacio en blanco. es Quito

Respuesta seleccionada: Ecuador

Respuesta correcta: Ecuador

Pregunta 5: Las islas. espacio en blanco. se encuentran en la región insular del Ecuador

Respuesta seleccionada: Filipinas

Respuesta correcta: Galápagos

F.2 APLICACIÓN FINAL

F.2.1 MÓDULO USUARIO

Figura F.5: Interfaz gráfica de la historia de usuario MU1 tarea MU1-1.

T4A Home Cuenta ▾

Email
faridsaud@yahoo.com

Clave
.....

Confirmar clave
.....

Nombres
farid

Apellidos
saud

Pasaporte/ID
0123456789

País
Ecuador ▾

Rol ?
 Estudiante
 Profesor

Pregunta de seguridad 1:
Nombre de una mascota ▾

Figura F.6: Cont. Interfaz gráfica de la historia de usuario MU1 tarea MU1-1.

.....

Confirmar clave

.....

Nombres

farid

Apellidos

saud

Pasaporte/ID

0123456789

Pais

Ecuador

Rol ?

Estudiante

Profesor

Pregunta de seguridad 1:

Nombre de una mascota

1

Pregunta de seguridad 2:

Lugar de nacimiento de la madre

2

REGISTRAR

Figura F.7: Interfaz gráfica de la historia de usuario MU2 tarea MU2-1.

T4A Home Cuenta

Tests 4 All

Generador de pruebas para personas no vidente.

Proyecto de titulación creado por el estudiante Farid Saud Roller

Iniciar Sesión

Email

faridsaud@yahoo.com

Clave

.....

Rol ?

Profesor

CREAR CUENTA INICIAR SESIÓN

He olvidado mi contraseña o pin

Figura F.8: Interfaz gráfica de la historia de usuario MU3 tarea MU3-1.

The screenshot shows a web interface with a blue header bar. On the left, it says 'T4A'. On the right, there are links for 'Home' and 'Cuenta'. The main content area contains a registration form with the following fields:

- Email: A text input field with a small icon on the right.
- Pregunta de seguridad 1: A dropdown menu.
- Pregunta de seguridad 2: A dropdown menu.
- A blue button labeled 'SIGUIENTE' at the bottom right.

Figura F.9: Cont. Interfaz gráfica de la historia de usuario MU3 tarea MU3-1.

The screenshot shows the same web interface as Figure F.8, but with the second step of the registration form. The fields are:

- Nueva contraseña: A text input field with a small icon on the right.
- Repetir nueva contraseña: A text input field with a small icon on the right.
- Two checked checkboxes: 'Generar nuevo pin' and 'Cambiar preguntas de seguridad'.
- Pregunta de seguridad 1: A dropdown menu with the text 'Nombre de una mascota'.
- Pregunta de seguridad 2: A dropdown menu with the text 'Lugar de nacimiento de la madre'.
- A blue button labeled 'SIGUIENTE' at the bottom right.

F.2.2 MÓDULO CURSO

Figura F.10: Interfaz gráfica de la historia de usuario MC1 tarea MC1-1.

T4A Home Cuenta >

Nombre

Prueba

Descripción

Descripción de prueba

CLONAR REGISTRAR

Figura F.11: Interfaz gráfica de la historia de usuario MC2 tarea MC2-1.

The screenshot shows a web interface with a blue header bar. On the left, it says 'T4A'. On the right, there are links for 'Home' and 'Cuenta'. Below the header, there is a form with the label 'Correo Electrónico' and the text 'test2@test.com'. A red box highlights a blue button labeled 'REGISTRAR'.

Figura F.12: Interfaz gráfica de la historia de usuario MC3 tarea MC3-1.

The screenshot shows a web interface with a blue header bar. On the left, it says 'T4A'. On the right, there are links for 'Home' and 'Cuenta'. Below the header, there are two input fields: 'Nombre' and 'Descripción'. At the bottom, there are two buttons: 'CLONAR' (highlighted with a red box) and 'REGISTRAR'.

Figura F.13: Cont. Interfaz gráfica de la historia de usuario MC3 tarea MC3-1.

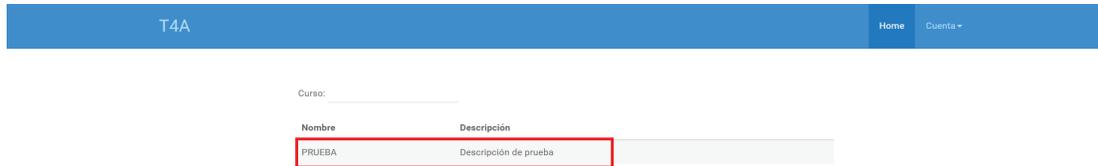


Figura F.14: Interfaz gráfica de la historia de usuario MC4 tarea MC4-1.

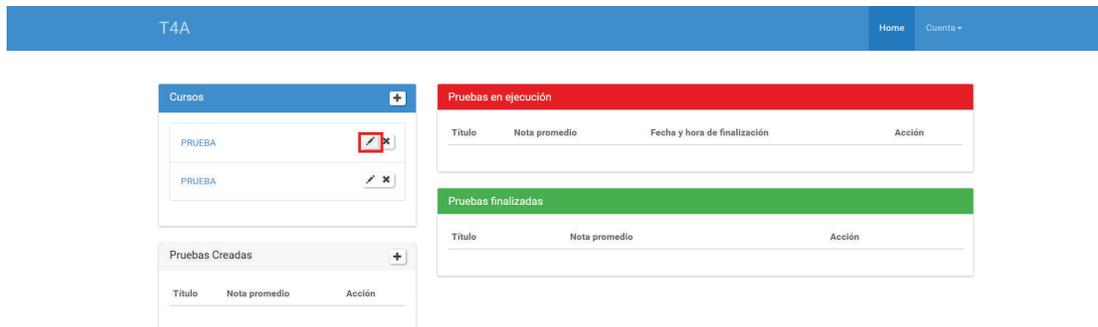


Figura F.15: Cont. Interfaz gráfica de la historia de usuario MC4 tarea MC4-1.

T4A Home Cuenta ▾

Nombre
Nuevo nombre PRUEBA

Descripción
Nueva descripción de prueba

GUARDAR

Figura F.16: Interfaz gráfica de la historia de usuario MC5 tarea MC5-1.

T4A Home Cuenta ▾

Cursos +

PRUEBA ✎ ✕

PRUEBA ✎ ✕

Pruebas en ejecución

Título	Nota promedio	Fecha y hora de finalización	Acción

Pruebas finalizadas

Título	Nota promedio	Acción

Pruebas Creadas +

Título	Nota promedio	Acción

Figura F.17: Cont. Interfaz gráfica de la historia de usuario MC5 tarea MC5-1.

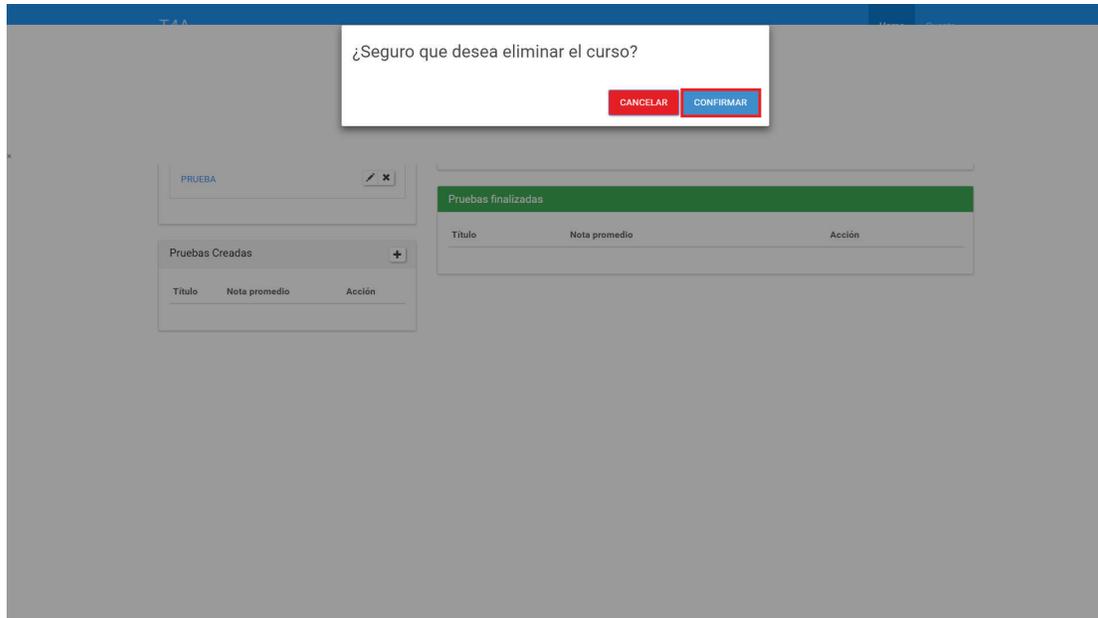


Figura F.18: Interfaz gráfica de la historia de usuario MC6 tarea MC6-1.



Figura F.19: Cont. Interfaz gráfica de la historia de usuario MC6 tarea MC6-1.

T4A Home Cuenta ▾

Curso seleccionado

PRUEBA

Descripción de prueba

Pruebas Creadas +

Título	Nota promedio	Acción

Estudiantes +

Nombre	Email	Acción
null null	test2@test.com	✕

Pruebas en ejecución

Título	Nota promedio	Fecha y hora de finalización	Acción

Pruebas finalizadas

Título	Nota promedio

Figura F.20: Interfaz gráfica de la historia de usuario MC7 tarea MC7-1.

T4A Home Cuenta ▾

Cursos +

PRUEBA ✎ ✕

Pruebas Creadas +

Título	Nota promedio	Acción

Pruebas en ejecución

Título	Nota promedio	Fecha y hora de finalización	Acción

Pruebas finalizadas

Título	Nota promedio	Acción

Figura F.21: Cont. Interfaz gráfica de la historia de usuario MC7 tarea MC7-1.

T4A Home Cuenta ▾

Cursos

PRUEBA

Pruebas pendientes

Título	Intentos restantes	Nota actual X/10	Fecha y hora de finalización

Pruebas finalizadas

Título	Nota

Figura F.22: Interfaz gráfica de la historia de usuario MC8 tarea MC8-1.

T4A Home Cuenta ▾

Curso seleccionado

PRUEBA
Descripción de prueba

Pruebas Creadas +

Título	Nota promedio	Acción

Estudiantes +

Nombre	Email	Acción
null null	test2@test.com	✘
test5 test5	test5@test.com	✘

Pruebas en ejecución

Título	Nota promedio	Fecha y hora de finalización	Acción

Pruebas finalizadas

Título	Nota promedio

F.2.3 MÓDULO PRUEBA

Figura F.23: Interfaz gráfica de la historia de usuario MP1 tarea MP1-1.

T4A Home Cuenta ▾

Curso:
PRUEBA

Título:
prueba

Descripción:
descripción prueba

Número de intentos:
3

Fecha de inicio:
09/02/2017 12:00 AM

Fecha de finalización:
12/02/2017 12:00 AM

CLONAR PRUEBA GUARDAR

Preguntas de opción múltiple

+

Figura F.24: Interfaz gráfica de la historia de usuario MP2 tarea MP2-1.

T4A
Home Cuenta ▾

Título:
nuevo nombre prueba

Descripción:
nueva descripción prueba

Número de intentos:
2

Fecha de inicio: 10/02/2017 12:00 AM Fecha de finalización: 12/03/2017 12:00 AM

GUARDAR

Preguntas de opción múltiple
+

Preguntas de verdadero/falso
+

Figura F.25: Interfaz gráfica de la historia de usuario MP3 tarea MP3-1.

T4A
Home Cuenta ▾

Cursos +

PRUEBA
✎ ✕

Pruebas Creadas +

Título	Nota promedio	Acción
prueba	NA	✎ ✕

Pruebas en ejecución

Título	Nota promedio	Fecha y hora de finalización	Acción

Pruebas finalizadas

Título	Nota promedio	Acción

Figura F.26: Cont. Interfaz gráfica de la historia de usuario MP3 tarea MP3-1.

T4A Home Cuenta ▾

Curso seleccionado

PRUEBA
Descripción de prueba

Pruebas Creadas +

Título	Nota promedio	Acción
prueba	NA	

Estudiantes +

Nombre	Email	Acción
null null	test2@test.com	
test5 test5	test5@test.com	

Pruebas en ejecución

Título	Nota promedio	Fecha y hora de finalización	Acción

Pruebas finalizadas

Título	Nota promedio

Figura F.27: Interfaz gráfica de la historia de usuario MP4 tarea MP4-1.

T4A Home Cuenta ▾

Cursos +

PRUEBA

Pruebas Creadas +

Título	Nota promedio	Acción
prueba	NA	

Pruebas en ejecución

Título	Nota promedio	Fecha y hora de finalización	Acción
prueba	0	2017-09-02 00:00:00	

Pruebas finalizadas

Título	Nota promedio	Acción

Figura F.28: Cont. Interfaz gráfica de la historia de usuario MP4 tarea MP4-1.

T4A Home Cuenta ▾

Cursos

PRUEBA

Pruebas pendientes

Título	Intentos restantes	Nota actual X/10	Fecha y hora de finalización
prueba	1 de 1		2017-09-02 00:00:00

Pruebas finalizadas

Título	Nota

Figura F.29: Interfaz gráfica de la historia de usuario MP5 tarea MP5-1.

T4A Home Cuenta ▾

Curso seleccionado

PRUEBA
Descripción de prueba

Pruebas en ejecución

Título	Nota promedio	Fecha y hora de finalización	Acción
prueba	0	2017-09-02 00:00:00	<input type="button" value="x"/>

Pruebas finalizadas

Título	Nota promedio

Pruebas Creadas +

Título	Nota promedio	Acción
prueba	NA	<input type="button" value="x"/>

Estudiantes +

Nombre	Email	Acción
null null	test2@test.com	<input type="button" value="x"/>
test5 test5	test5@test.com	<input type="button" value="x"/>

Figura F.30: Cont. Interfaz gráfica de la historia de usuario MP5 tarea MP5-1.

The screenshot displays the user history interface for MP5 task MP5-1. It features a blue navigation bar at the top with 'T4A' on the left, 'Home' in the center, and 'Cuenta' on the right. Below the navigation bar, the interface is divided into three main sections:

- Curso (Course):** A white box with a blue header containing the text 'PRUEBA' and 'Descripción de prueba'.
- Pruebas pendientes (Pending tests):** A red-bordered box with a red header. It contains a table with the following data:

Título	Intentos restantes	Fecha y hora de finalización
prueba	1 de 1	2017-09-02 00:00:00
- Pruebas finalizadas (Completed tests):** A green-bordered box with a green header. It contains a table with the following data:

Título	Nota

Figura F.31: Interfaz gráfica de la historia de usuario MP6 tarea MP6-1.

The screenshot displays the user history interface for MP6 task MP6-1. It features a blue navigation bar at the top with 'T4A' on the left, 'Home' in the center, and 'Cuenta' on the right. Below the navigation bar, the interface shows three question sections:

- Pregunta 1: Enunciado nuevo**
 - Opcion 1 de 2: opcion 2 nueva
 - Opcion 2 de 2: opcion 1 nueva
- Pregunta 2: enunciado nueva**
 - Opcion 1 de 2: verdadero
 - Opcion 2 de 2: falso
- Pregunta 3: enunciado 1 nueva. espacio en blanco. enunciado 2 nueva. espacio en blanco. enunciado 3 nueva**
 - Opcion 1 de 2: a, b, c nueva
 - Opcion 2 de 2: e,d,f nueva

At the bottom of the interface, there is a button labeled 'ENVIAR'.

Figura F.32: Interfaz gráfica de la historia de usuario MP7 tarea MP7-1.

The screenshot shows a user interface for task MP7-1. At the top, there is a blue header with 'T4A' on the left and 'Home' and 'Cuenta' on the right. Below the header, the content is organized into three sections, each representing a question:

- Pregunta 1: Enunciado nuevo**
 - Respuesta seleccionada: opcion 2 nueva
 - Justificación respuesta seleccionada: justificacion 2 nueva
 - Respuesta correcta: opcion 1 nueva
 - Justificación respuesta correcta: justificacion 1 nueva
- Pregunta 2: enunciado nueva**
 - Respuesta seleccionada: falso
 - Respuesta correcta: verdadero
 - Justificación respuesta correcta: justificacion
- Pregunta 3: enunciado 1 nueva. espacio en blanco. enunciado 2 nueva. espacio en blanco. enunciado 3 nueva**
 - Respuesta seleccionada: e,d,f nueva
 - Justificación respuesta seleccionada: justificacion nueva
 - Respuesta correcta: a, b, c nueva

Figura F.33: Interfaz gráfica de la historia de usuario MP8 tarea MP8-1.

The screenshot shows a user interface for task MP8-1. At the top, there is a blue header with 'T4A' on the left and 'Home' and 'Cuenta' on the right. Below the header, there is a 'Curso:' label followed by a text input field. Below that is a table with three columns: 'Curso', 'Título', and 'Descripción'. The table contains two rows of data, with the first row highlighted by a red border.

Curso	Título	Descripción
PRUEBA	prueba	descripción prueba
PRUEBA	prueba	prueb

Figura F.34: Cont. Interfaz gráfica de la historia de usuario MP8 tarea MP8-1.

T4A Home Cuenta ▾

Curso: ▾

Título:
prueba

Descripción:
descripción prueba

Número de intentos:
3

Fecha de inicio:
09/02/2017 12:00 AM

Fecha de finalización:
12/02/2017 12:00 AM

CLONAR PRUEBA GUARDAR

Preguntas de opción múltiple

+
Preguntas de opción múltiple (6)

F.2.4 MÓDULO PREGUNTA

Figura F.35: Interfaz gráfica de la historia de usuario MR1 tarea MR1-1.
Preguntas de opción múltiple

Pregunta 1

Enunciado: Ponderación: X:1
Enunciado 1

Opción 1: Respuesta correcta
opcion 1

Justificación opción 1:
justificacion 1

Opción 2: Respuesta correcta
opcion 2

Justificación opción 2:
justificacion 2

[AGREGAR OPCIÓN](#)

[ELIMINAR PREGUNTA](#) [CLONAR PREGUNTA](#)

[+](#)

Figura F.36: Cont. Interfaz gráfica de la historia de usuario MR1 tarea MR1-1.

justificacion 2

AGREGAR OPCIÓN

ELIMINAR PREGUNTA CLONAR PREGUNTA

+

Preguntas de verdadero/falso

Pregunta 1

Enunciado:
enunciado

Opción correcta: Ponderación: X:1 Justificación:
justificacion

Verdadero
 Falso

ELIMINAR PREGUNTA CLONAR PREGUNTA

+

Preguntas de completar

+

Figura F.37: Cont. Interfaz gráfica de la historia de usuario MR1 tarea MR1-1.

Pregunta 1

Enunciados: Ponderación: X:1
enunciado 1 1

enunciado 2

enunciado 3

AGREGAR ENUNCIADO

Opción 1: Respuesta correcta
a, b, c

Justificación 1:
justificacion

Opción 2: Respuesta correcta
e,d,f

Justificación 2:
justificacion

AGREGAR OPCIÓN

ELIMINAR PREGUNTA CLONAR PREGUNTA

Figura F.38: Interfaz gráfica de la historia de usuario MR2 tarea MR2-1.

Preguntas de opción múltiple

Pregunta 1

Enunciado: Enunciado nuevo Ponderación: X:1
1

Opción 1: opcion 2 nueva Respuesta correcta

Justificación opción 1: justificacion 2 nueva

Opción 2: opcion 1 nueva Respuesta correcta

Justificación opción 2: justificacion 1 nueva

[AGREGAR OPCIÓN](#)

[ELIMINAR PREGUNTA](#) [CLONAR PREGUNTA](#)

Figura F.39: Cont. Interfaz gráfica de la historia de usuario MR2 tarea MR2-1.

[+](#)

Preguntas de verdadero/falso

Pregunta 1

Enunciado: enunciado nueva

Opción correcta: Verdadero Falso Ponderación: X:1
1 Justificación: justificacion

[ELIMINAR PREGUNTA](#) [CLONAR PREGUNTA](#)

[+](#)

Preguntas de completar

Pregunta 1

Enunciados: enunciado 1 nueva Ponderación: X:1
1

enunciado 2 nueva

Figura F.40: Cont. Interfaz gráfica de la historia de usuario MR2 tarea MR2-1.

Pregunta 1

Enunciados:

enunciado 1 nueva

enunciado 2 nueva

enunciado 3 nueva

AGREGAR ENUNCIADO

Opción 1:

a, b, c nueva Respuesta correcta

Justificación 1:

justificacion nueva

Opción 2:

e,d,f nueva Respuesta correcta

Justificación 2:

justificacion nueva

AGREGAR OPCIÓN

ELIMINAR PREGUNTA CLONAR PREGUNTA

Figura F.41: Interfaz gráfica de la historia de usuario MR3 tarea MR3-1.

AGREGAR OPCIÓN

ELIMINAR PREGUNTA CLONAR PREGUNTA

+

Preguntas de verdadero/falso

Pregunta 1

Enunciado:

enunciado nueva

Opción correcta:

Verdadero Falso

Ponderación: X:1 1

Justificación:

justificacion

ELIMINAR PREGUNTA CLONAR PREGUNTA

+

Preguntas de completar

Pregunta 1

Figura F.42: Interfaz gráfica de la historia de usuario MR4 tarea MR4-1.

Curso	Titulo	Descripción
PRUEBA	prueba	descripción prueba
PRUEBA	prueba	prueb