

# **ESCUELA POLITÉCNICA NACIONAL**

**FACULTAD DE INGENIERÍA DE SISTEMAS**

**DESARROLLO DE UNA APLICACIÓN WEB PARA LA  
DETERMINACIÓN DE CRITERIOS DE RELEVANCIA APLICADOS  
A REVISIONES DOCUMENTALES DE INGENIERIA**

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

**RONNIE MARCELO NIETO MOROCHO**

**Ronnie.nieto01@epn.edu.ec**

**DIRECTOR: PhD. MARCO OSWALDO SANTÓRUM GAIBOR**

**marco.santorum@epn.edu.ec**

**CO-DIRECTORA: MSc. MAYRA CARRIÓN**

**mayra.carrion@epn.edu.ec**

**Quito, julio de 2021**

## **CERTIFICACIÓN**

Certificamos que el presente trabajo fue desarrollado por Ronnie Marcelo Nieto Morocho bajo nuestra supervisión.

---

**PhD. Marco Santórum**

**DIRECTOR DE PROYECTO**

---

**MSc. Mayra Carrión**

**CO-DIRECTORA DE PROYECTO**

## **DECLARACIÓN**

Yo, Ronnie Marcelo Nieto Morocho, 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.

---

**Ronnie Marcelo Nieto Morocho**

## **DEDICATORIA**

Este trabajo de titulación está dedicado a mi madre, sin su esfuerzo, perseverancia y apoyo incondicional esta meta no se habría alcanzado.

## **AGRADECIMIENTO**

A mi madre, que siempre estuvo en cada paso de esta etapa con su amor incondicional, por ser el pilar de mi vida y enseñarme que a pesar de las adversidades solo depende de nosotros el forjar nuestro propio camino.

A mis hermanos Karen y Gabriel cuya preocupación y ejemplo me motivaban a continuar.

A mi abuelita Mercedes, por su cariño incondicional a lo largo del camino.

A mis tutores de mi trabajo de titulación, Marco Santórum y Mayra Carrión, por su guía y colaboración en este proyecto.

A mis amigos, por el apoyo brindado en los momentos más difíciles y en las horas de alegría. Especialmente a Anderson que me demostró lo que es la verdadera amistad. A ustedes, gracias totales.

A Tania por su amistad, apoyo y consejo.

A Emillie por su apoyo que me motivo a seguir en las instancias finales.

## ÍNDICE DE CONTENIDO

CERTIFICACIÓN.....	I
DECLARACIÓN.....	II
DEDICATORIA.....	III
AGRADECIMIENTO.....	IV
ÍNDICE DE CONTENIDO.....	V
ÍNDICE DE FIGURAS.....	VIII
ÍNDICE DE TABLAS.....	IX
RESUMEN.....	XI
ABSTRACT.....	XII
CAPÍTULO 1.....	1
1.1. Introducción.....	1
1.2. Descripción del problema.....	2
1.3. Objetivos.....	2
1.3.1. Objetivo general.....	2
1.3.2. Objetivos específicos.....	3
1.4. Marco Teórico.....	3
1.4.1. Frecuencia de Término – Frecuencia Inversa de Documento (TF-IDF).....	3
1.4.2. Lógica Matemática.....	5
1.4.3. Metodología para la inferencia de criterios.....	8
1.5. Scrum.....	11
1.6. Resumen del capítulo.....	12
CAPÍTULO 2.....	14
2.1. Metodología.....	14
2.2. Arquitectura, herramientas y recursos utilizados.....	14
2.2.1. Patrón Modelo Vista Controlador (MVC).....	14
2.2.2. Arquitectura general de la aplicación.....	15
2.2.3. Elección de herramientas y recursos.....	16
2.3. Fase inicial de Scrum.....	17

2.3.1.	Definición de requerimientos.....	17
2.3.2.	Identificación de roles.....	18
2.3.3.	Historias de usuario.....	19
2.3.4.	Product Backlog.....	24
2.4.	Desarrollo de la aplicación.....	26
2.4.1.	Sprint 0.....	26
2.4.2.	Sprint 1.....	32
2.4.3.	Sprint 2.....	37
2.5.	Caso de estudio.....	44
2.6.	Fase de lanzamiento.....	50
2.7.	Resumen del capítulo.....	52
CAPÍTULO 3.....		54
Evaluación y Resultados.....		54
3.1.	Pruebas de funcionalidad.....	54
3.1.1.	Módulo de carga de archivos PDF.....	54
3.1.2.	Módulo de carga de criterios y sinónimos.....	56
3.1.3.	Módulo de resultados.....	59
3.1.4.	Módulo de usuario.....	60
3.2.	Pruebas de usabilidad.....	61
3.2.1.	Primera fase.....	61
3.2.2.	Segunda fase.....	62
3.2.3.	Tercera fase.....	65
3.2.4.	Cuarta fase.....	67
3.3.	Resumen del capítulo.....	69
CAPÍTULO 4.....		71
4.1.	Conclusiones.....	71
4.2.	Perspectivas.....	72
4.3.	Recomendaciones.....	73
REFERENCIAS BIBLIOGRÁFICAS.....		74

ANEXOS ..... 75



## ÍNDICE DE FIGURAS

Figura 1. Arquitectura MVC.....	15
Figura 2: Arquitectura de la aplicación.....	16
Figura 3: Esquema de base de datos .....	29
Figura 4: Burndown Chart Sprint 0.....	31
Figura 5: Interfaz para el registro de usuario .....	31
Figura 6: Interfaz de información del usuario .....	32
Figura 7: Burndown Chart Sprint 1.....	36
Figura 8: Interfaz para la carga de archivos Pdf.....	37
Figura 9: Interfaz para la carga de criterios y sinónimos.....	37
Figura 10: Burndown Chart Sprint 2.....	42
Figura 11: Interfaz para la selección de la variable k .....	43
Figura 12: Interfaz de resultados .....	43
Figura 13: Interfaz de descripción de la metodología.....	45
Figura 14: Carga de un archivo pdf.....	46
Figura 15: Carga de varios archivos pdf.....	46
Figura 16: Interfaz que muestra los archivos pdf cargados de los grupos P y Q .....	47
Figura 17: Interfaz que muestra los criterios y sinónimos cargados .....	48
Figura 18: Interfaz para seleccionar la forma de calcular la variable k .....	49
Figura 19: Interfaz de resultados .....	49
Figura 20: Promedio porcentual por pregunta, encuesta CSUQ.....	66
Figura 21: Promedio por escala .....	67
Figura 22: Resultados pregunta 1 encuesta CSUQ .....	68
Figura 23: Resultados pregunta 16 encuesta CSUQ .....	68
Figura 24: Resultados pregunta 5 encuesta CSUQ .....	69
Figura 25: Resultados pregunta 7 encuesta CSUQ .....	69

## ÍNDICE DE TABLAS

Tabla 1: Tabla de verdad para los operadores lógicos. ....	6
Tabla 2: Tabla de patrones lógicos de caracterización .....	7
Tabla 3: Matriz de frecuencia absoluta .....	9
Tabla 4: Matriz de frecuencia normalizada .....	9
Tabla 5: Matriz de frecuencia inversa del documento .....	9
Tabla 6: Matriz de instanciación.....	10
Tabla 7: Matriz de comportamiento.....	10
Tabla 8: Herramientas y Recursos.....	16
Tabla 9: Historia de usuario épica 1.....	18
Tabla 10: Historia de usuario épica 2 .....	18
Tabla 11: Historia de usuario épica 3 .....	18
Tabla 12: Roles de Scrum.....	19
Tabla 13: Historia de usuario HU01 .....	20
Tabla 14: Historia de usuario HU02.....	20
Tabla 15: Historia de usuario HU03.....	21
Tabla 16: Historia de usuario HU04 .....	21
Tabla 17: Historia de usuario HU05.....	22
Tabla 18: Historia de usuario HU06.....	22
Tabla 19: Historia de usuario HU07 .....	23
Tabla 20: Historia de usuario HU08.....	23
Tabla 21: Historia de usuario HU09.....	24
Tabla 22: Historia de usuario HU10 .....	24
Tabla 23: Product Backlog inicial .....	25
Tabla 24: Product Backlog final .....	25
Tabla 25: Sprint 0 Backlog .....	26
Tabla 26: Sprint 0 Review .....	30
Tabla 27: Sprint 1 Backlog .....	32
Tabla 28: Sprint 1 Review .....	35
Tabla 29: Sprint 2 Backlog .....	38
Tabla 30: Sprint 2 Review .....	41
Tabla 31: Retrospectiva del proyecto .....	51
Tabla 32: Casos de prueba módulo de carga de archivos PDF.....	54
Tabla 33: Casos de prueba módulo de carga de criterios y sinónimos .....	56
Tabla 34: Casos de prueba módulo de resultados.....	59
Tabla 35: Casos de prueba módulo de usuario.....	61

Tabla 36: Tareas para la evaluación de usabilidad .....	62
Tabla 37: Cuestionario para evaluar la usabilidad .....	64

## RESUMEN

El presente trabajo de titulación presenta una solución de software para la automatización de la metodología para la inferencia de criterios clave propuesto por Mayra Carrión. Mediante el estudio de la medida numérica TF-IDF (frecuencia de término y la frecuencia inversa del documento) y un conjunto de declaraciones lógicas, se efectúa la inferencia de criterios que se consideran necesarios y suficiente. El aplicativo fue desarrollado con un enfoque ágil como SCRUM, mientras que para el desarrollo de la aplicación web se usó el framework Django, para la implementación en producción se hizo uso de los servicios de AWS junto con el servidor Nginx. Para validar la usabilidad y funcionalidad del aplicativo se efectuaron dos pruebas con 6 participante de la Escuela Politécnica Nacional, entre ingenieros y tesisistas, quienes han trabajado previamente con el método de investigación automatizado en este proyecto. Los resultados obtenidos en las dos pruebas efectuadas evidencian que la funcionalidad de la aplicación es la esperada, y se obtuvo un promedio de 92.71% en las preguntas planteadas en la encuesta de usabilidad lo que permite concluir que la aplicación es amigable con el usuario, fácil de aprender y usable. Por lo tanto, se concluye que se cumplió el objetivo principal de este proyecto que consiste en el desarrollo e implementación en producción de una aplicación web que automatice el método de inferencia de criterios clave.

**PALABRAS CLAVE:** Aplicación web, SCRUM, TF-IDF, enfoque ágil, criterios de relevancia, revisiones documentales.

## ABSTRACT

The present degree work presents a software solution for the automation of the methodology for the inference of key criteria proposed by Mayra Carrión. Through the study of the numerical measure TF-IDF (frequency of term and the inverse frequency of the document) and a set of logical statements, the inference of criteria that are considered necessary and sufficient is made. The application was developed with an agile approach such as SCRUM, while for the development of the web application the Django framework was used, for the implementation in production the AWS services were used together with the Nginx server. To validate the usability and functionality of the application, two tests were carried out with 6 participants from the Escuela Politécnica Nacional, between engineers and thesis students, who have previously worked with the automated research method in this project. The results obtained in the two tests show that the functionality of the application is as expected, and an average of 92.71% was obtained in the proposed questions in the usability questionnaire, which allows to conclude that the application is user-friendly, easy to use, easy to learn and usable. Therefore, it is concluded that the main objective of this project was fulfilled, which consists of the development and implementation in production of a web application that automates the key criteria inference method.

**KEYWORDS:** Web application, SCRUM, TF-IDF, agile approach, relevance criteria, documentary reviews.

# CAPÍTULO 1

## 1.1. Introducción

La búsqueda y recuperación de información, llamado en inglés Information Search and Retrieval (ISR), consiste en buscar información en un documento, buscar documentos en sí mismos y también buscar metadatos que describen datos y bases de datos de textos, imágenes o audios que satisfagan la necesidad de información [1]. Este proceso inicia cuando el usuario realiza una consulta esperando obtener un resultado, un claro ejemplo de este proceso lo encontramos en los actuales motores de búsqueda de internet; la recuperación de consultas garantiza la continuación de la investigación y la generación de mejores soluciones para el problema. Así, para este trabajo el problema refinado se convierte en la tarea de buscar en un corpus los documentos y criterios que se puedan considerar más relevantes para lo que el usuario ingresa como consulta [2].

Debido a la disponibilidad generalizada y creciente de documentos de texto en formato electrónico, la búsqueda y recuperación de información se ha visto obligada a adaptarse a las nuevas tecnologías aumentando la importancia de utilizar métodos automáticos para analizar el contenido de los documentos de texto. Como resultado, la identificación y clasificación de la información que se puede considerar relevante para un investigador dentro de los documentos de texto en función de su contenido se ha vuelto imperativa [3].

El significado de la investigación científica enfatiza la recopilación sistemática, la interpretación y la evaluación de los datos obtenidos con el propósito de contribuir a la ciencia. Tomando la definición anterior de la investigación científica podemos englobar a la revisión documental y a la ISR como el proceso que permiten llevar a cabo la recopilación sistemática de datos. Bajo la interpretación de este concepto, se concibe el desarrollo de una aplicación web con el fin de que cualquier investigador que realice una investigación sistemática de literatura, posea una herramienta que le facilite y ahorre tiempo en la tarea de analizar e interpretar los datos mediante la ponderación de consultas y la determinación y selección de criterios de relevancia metodológica, dentro de un conjunto de documentos.

## **1.2. Descripción del problema**

Determinar la relevancia metodológica de criterios en una revisión documental conlleva un proceso manual cuya duración se ve influenciada no solo por el número de documentos incluidos en la revisión, sino también por el proceso utilizado para ponderar y determinar la relevancia de los criterios deseados. La automatización de este proceso se ve restringida debido a que requiere la interacción directa y constante por parte del investigador en tareas como la valoración de documentos, que solo puede llevarse a cabo por la experiencia y el criterio del investigador con el fin de recopilar solo aquellos documentos que presenten cierta relevancia dentro de la temática investigada. Pero la discriminación de documentos presente en la revisión documental representa solo la primera parte de este proceso, la selección de criterios de relevancia metodológica y la ponderación de estos constituyen los siguientes pasos a seguir. Al igual que en la valoración de documentos, la selección de criterios puede llevarse a cabo mediante el criterio de un profesional, así como a través del uso de métodos estadísticos, matemáticos u otros. La última tarea del proceso, que consiste en la ponderación de los criterios obtenidos, es la tarea en la cual se enfoca este trabajo. Esta tarea consiste en aplicar la medida numérica TF-IDF, dado una consulta/termino, con el fin de obtener una matriz booleana que será comparada con las tablas de verdad, y así usando la lógica matemática y la lógica proposicional poder inferir criterios suficientes y necesarios.

La solución implementada como una aplicación web, responde a la necesidad de los investigadores de poseer una herramienta a la que puedan acceder en cualquier momento y lugar sin la necesidad de instalar ningún programa. El desarrollo de nuevas tecnologías que permiten la creación de páginas web, mediante el uso de lenguajes que facilitan la manipulación de estructuras de datos y la ejecución de cálculos matemáticos de forma eficiente, constituyo otro factor para proponer una aplicación web para la solución del problema planteado.

## **1.3. Objetivos**

### **1.3.1. Objetivo general**

Desarrollar una aplicación web utilizando principios de desarrollo ágil con Scrum, para la determinación de criterios de relevancia mediante el uso combinado de la medida numérica TF-IDF, lógica matemática y lógica proposicional.

### 1.3.2. Objetivos específicos

- Estudiar la medida numérica TF-IDF para la ponderación de consultas, la determinación de criterios de relevancia en un corpus, y estudiar el uso de tablas de verdad presentes en la lógica proposicional y la lógica matemática para la inferencia de criterios suficientes y necesarios.
- Realizar pruebas de uso en cuanto a funcionalidad y usabilidad con el usuario final.
- Implementar la solución en un ambiente de producción.

## 1.4. Marco Teórico

### 1.4.1. Frecuencia de Término – Frecuencia Inversa de Documento (TF-IDF)

El término frecuencia de término – frecuencia inversa de documento (TF-IDF por sus siglas en inglés) es una medida numérica simple y eficaz para hacer coincidir palabras en una consulta con documentos que son relevantes para esa consulta, además puede devolver aquellos documentos que son altamente relevantes para una consulta en particular. En otras palabras, el enfoque TF-IDF captura la relevancia entre las palabras, los documentos de texto y las categorías particulares [3]. Si un usuario ingresara una consulta para un tema en particular, esta medida numérica permite encontrar documentos que contengan información relevante sobre la consulta. Cabe mencionar que la codificación de TF-IDF es sencilla, lo que la hace ideal para formar la base de algoritmos más complicados y sistemas de recuperación de consultas [2].

Esencialmente, TF-IDF determina la frecuencia relativa de las palabras en un documento específico comparado con la proporción inversa de dicha palabra a través del corpus de documentos. Mediante esta operación se puede intuir cuán relevante es una palabra para un documento específico. Las palabras que son comunes en un único documento o en un grupo pequeño de documentos tienden a tener un TF-IDF más alto en comparación a palabras como artículos o preposiciones [2].

TF-IDF es una combinación de dos términos, frecuencia de término TF (Term Frequency) y frecuencia inversa de documentos (Inverse Document Frequency). El primer término TF, se usa para describir el número de apariciones de una palabra dentro de un documento. Para este cálculo se debe tomar en cuenta que la longitud de los documentos, en referencia al número de palabras que poseen, nunca va a ser la misma; un corpus podrá contener documentos muy extensos y a su vez documentos relativamente pequeños, por lo que es una posibilidad que cualquier palabra aparezca con mayor frecuencia en un documento



extenso en comparación con un documento menos extenso. Para corregir cualquier desigualdad que este factor puede presentar, se aplicará en este trabajo una variante de la medida  $tf$  conocida como “doble normalización 0.5”, en la cual la frecuencia de cualquier palabra en un documento es dividida para la frecuencia del término más frecuente en el documento [4].

$$(1) \quad tf(t_i, d_j, a) = a + (1 - a) * \frac{f_{t,d}}{f_{\max(d_j)}}$$

Donde: la constante  $a=0.5$  es un valor que permite normalizar de la frecuencia del término.

$f_{t,d}$ , representa la frecuencia absoluta del término ( $t$ ) en el documento ( $d$ ).

$f_{\max(d_j)}$ , representa el termino con la máxima frecuencia dentro del documento.

Suponiendo que poseemos un documento D1 con un total de 1000 palabras, la palabra “politécnica” está presente en 10 ocasiones, y el termino más frecuente es “educación” con 50 coincidencias, la frecuencia de termino para la palabra “politécnica” será:

$$tf = 0.5 + 0.5 * \frac{10}{50} = 0.6$$

Cuando se calcula la frecuencia de un término de un documento, se puede observar que el algoritmo trata a todas las palabras por igual, sin embargo, no todas las palabras tienen la misma importancia, la preposición “para” no tiene la misma importancia para el usuario como el sustantivo “politécnica” sin embargo, esta preposición puede tener más ocurrencias dentro de un documento. Aquí es donde entra el término IDF, que se encarga de asignar un menor peso a las palabras frecuentes y asignar un peso mayor a las palabras menos frecuentes [4]. Así, la medida se obtiene aplicando el logaritmo al cociente de la división del número total de documentos por el número de documentos que contienen el término.

$$(2) \quad idf(t_i, D) = \log \frac{n(D)}{df_t}$$

Donde:  $n(D)$ , representa el total de documentos dentro de la colección y  $df_t$  representa el total de documentos en los que aparece dicho término.

Por ejemplo, si tenemos un conjunto de 10 documentos y el término “politécnica” está presente en 5 de esos documentos, la frecuencia inversa del documento se puede calcular como:

$$idf = \log\left(\frac{10}{5}\right) = 0.3010$$

La medida TF-IDF se calcula multiplicando TF por IDF

$$(3) \text{tfidf}(t_i, d_j, a, D) = \text{tf}(t_i, d_j, a) * \text{idf}(t_i, D)$$

Para nuestro ejemplo la medida TF-IDF, para el término “politécnica” en un determinado documento, se obtiene multiplicando los dos valores ya obtenidos:

$$\text{tfidf} = 0.6 * 0.3010 = 0.1806$$

#### **1.4.2. Lógica Matemática**

La lógica matemática es un subcampo de las matemáticas que explora las aplicaciones de la lógica formal a las matemáticas. La lógica matemática es la herramienta más eficaz que se tiene para razonar cosas que no se pueden comprender, lo que la convierte en una herramienta ideal para la Ciencia de la Computación [5].

Como base, la lógica matemática trata con conceptos matemáticos expresados usando sistemas lógicos formales. Estos sistemas comparten la propiedad común de considerar solo expresiones en un lenguaje formal fijo. Los sistemas de lógica proposicional y lógica de primer orden son los más estudiados, debido a su aplicabilidad a los fundamentos de las matemáticas.

El enfoque que se aborda consiste en crear una lista de axiomas que sean declaraciones con un valor de verdad sobre el modelo, y una lista de reglas de inferencia que permita derivar nuevas declaraciones verdaderas de los axiomas. Los axiomas y las reglas de inferencia juntas generan una teoría que consta de todas las afirmaciones que se pueden construir a partir de los axiomas mediante la aplicación de las reglas de inferencia [5].

La lógica proposicional [6] es la forma más simple de la lógica matemática, aquí las únicas afirmaciones que se consideran son proposiciones, es decir, aquellas afirmaciones que no contienen variables y se enmarcan en los criterios de verdad, proposiciones que son siempre verdaderas o siempre falsas. Aquí los modelos se tratan como una colección de proposiciones. Los sistemas de lógica proposicional incluyen además conectivas lógicas,

por lo que dentro de este tipo de lógica se puede analizar la inferencia lógica de proposiciones a partir de proposiciones, pero sin tener en cuenta la estructura interna de las proposiciones más simples [5].

Para realizar esta inferencia lógica se combinan las proposiciones con operadores o conectores lógicos. El conector implicación, también conocido como condicional, es considerado el más importante para pruebas, este conector representa una afirmación del tipo “si...entonces”. Si  $p$  implica  $q$ , entonces se puede escribir  $p \rightarrow q$  o  $p \Rightarrow q$ . La proposición  $p \rightarrow q$  es verdadera siempre que (a)  $p$  sea falso, o (b)  $q$  es verdadero (ver tabla 1) [5].

El segundo conector presente en este trabajo es el conector bicondicional, la única forma para que  $p \leftrightarrow q$  sea falsa es que un lado sea verdadero y un lado sea falso (ver tabla 1).

### Tablas de verdad

Para definir las operaciones lógicas sobre las proposiciones se utilizan tablas de verdad, las cuales dan para cualquier combinación de valores de verdad (verdadero o falso, que serán representados como 1 y 0) de las entradas, el valor de verdad de las salidas. Para este trabajo se basará en la concepción tradicional del análisis de la lógica, cuyo objetivo es determinar las condiciones necesarias suficientes de verdad de una proposición. Cada renglón de la tabla que hace verdadera la proposición en cuestión determina una condición suficiente para su verdad. En cada renglón, los valores asignados a cada variable proposicional determinan una condición necesaria del renglón [7].

A continuación, se muestra una tabla de verdad para los operadores lógicos:

<i>Tabla 1: Tabla de verdad para los operadores lógicos. <math>p</math></i>	$q$	$p \vee q$	$p \oplus q$	$p \wedge q$	$p \rightarrow q$	$p \leftrightarrow q$
0	0	0	0	0	1	1
0	1	1	1	0	1	0
1	0	1	1	0	0	0
1	1	1	0	1	1	1

Se puede considerar a cada fila de una tabla de verdad como un modelo para la lógica proposicional, debido a que las únicas cosas que se puede describir en la lógica proposicional son si las proposiciones son verdaderas o falsas. La construcción de una tabla de verdad se corresponde con la generación de todos los modelos posibles [5].

Esta concepción de tablas de verdad como posibles modelos es útil cuando se desea determinar cuándo una proposición particular es verdadera. Probar la veracidad de una proposición mediante el uso de tablas de verdad es una versión simple de la verificación de modelos. Esto funciona para la lógica proposicional porque la lista de modelos es igual a la lista de posibles combinaciones de valores de verdad para todas las proposiciones simples P, Q, etc. Se puede verificar que cada tabla de verdad construida funciona, al verificar que los valores de verdad de cada columna (que corresponden a alguna sub-expresión de lo que se intenta demostrar) se ajusten a los valores de verdad de las columnas anteriores de acuerdo con las reglas establecidas por la tabla de verdad que define la operación lógica apropiada [5].

Para este trabajo se establecen los valores lógicos de verdad de la tabla 2 con los valores de verdad de la lógica matemática antes mencionados; esta tabla será utilizada para determinar la relevancia de un criterio y para categorizarlo, este proceso se explica en el apartado referente a la metodología utilizada.

<i>Tabla 2: Tabla de patrones lógicos de caracterización.</i>		<b>Condición necesaria y no suficiente</b>	<b>Condición suficiente y no necesaria</b>	<b>Condición necesaria y suficiente</b>
<b>Valores de verdad</b>				
<b><i>p</i></b>	<b><i>q</i></b>	<b><math>q \rightarrow p</math></b>	<b><math>p \rightarrow q</math></b>	<b><math>p \leftrightarrow q</math></b>
0	0	1	1	1
0	1	0	1	0
1	0	1	0	0
1	1	1	1	1

A continuación, se describe cada una de las categorías en las que se puede encasillar un criterio:

- Necesario y no suficiente: se refiere a que el criterio aparece en todos los documentos que están en el contexto de estudio, y dicho criterio aparecerá al menos en uno o más documentos que no están en el contexto de estudio.
- Suficiente y no necesario: se refiere a que el criterio aparece en al menos uno o más documentos que están en el contexto de estudio, y dicho criterio no aparecerá en ninguno de los documentos que no están en el contexto de estudio.

- Necesario y suficiente: se refiere a que el criterio aparece en todos los documentos que están en el contexto de estudio, y dicho criterio no aparecerá en ninguno de los documentos que no están en el contexto de estudio.
- No necesario y no suficiente: se refiere a que el criterio aparece en al menos uno o más documentos que están en el contexto de estudio, y dicho criterio igualmente aparece en al menos uno o más documentos que no están en el contexto de estudio.
- Ninguno: se refiere al resto de categorías que pueden aparecer y no están especificadas en ninguna de las anteriores.

### **1.4.3. Metodología para la inferencia de criterios**

La metodología utilizada para la inferencia de criterio corresponde al proceso propuesto por Mayra Carrión en [8], a esta metodología la dividiremos en tres fases para una mayor comprensión de esta:

- La primera fase corresponde a la recolección de documentos relacionados con la temática a investigar y la selección de criterios y sus correspondientes sinónimos.
- La segunda fase se divide en dos procesos independientes, el proceso cualitativo: el cual se basa en la opinión del investigador para obtener la matriz booleana que será usada en la sección siguiente, y el proceso cuantitativo basado en el cálculo TF-IDF para obtener la matriz booleana.
- La tercera fase corresponde a la selección de los criterios de relevancia mediante la comparación de la matriz booleana y las tablas de verdad.

Esta metodología toma dos enfoques para realizar la inferencia de criterios, la opinión del investigador y el uso de la medida numérica TF-IDF, para este trabajo se busca automatizar la dos últimas fases específicamente el proceso cuantitativo correspondiente a la segunda fase de la metodología, es decir, aquel que usa la medida numérica TF-IDF.

De la primera fase de metodología se obtienen un conjunto de documentos y conjunto de criterios (matriz de análisis reducida), los cuales sirven de datos de entrada para la ejecución de la segunda fase de la metodología.

A continuación, se describen los pasos que se ejecutan en el proceso cuantitativo de la segunda fase:

El primer paso consiste en la obtención de la matriz de frecuencia absoluta (ver tabla 3), cuyo objetivo es especificar el número de ocurrencias de cada criterio en cada uno de los documentos.

Tabla 3: Matriz de frecuencia absoluta.	Documentos contexto de estudio 1			Documentos contexto de estudio 2		
	Documento 1	Documento 2	...	Documento 1	Documento 2	...
Término 1	$f_{t,d}$	$f_{t,d}$		$f_{t,d}$	$f_{t,d}$	
Término 2	$f_{t,d}$	$f_{t,d}$		$f_{t,d}$	$f_{t,d}$	
...	$f_{t,d}$	$f_{t,d}$		$f_{t,d}$	$f_{t,d}$	

El siguiente paso busca obtener la matriz de frecuencia normalizada (ver tabla 4), la cual se obtiene mediante la aplicación de la ecuación (1) mencionada en el apartado 1.4.1.

Tabla 4: Matriz de frecuencia normalizada.	Documentos contexto de estudio 1			Documentos contexto de estudio 2		
	Documento 1	Documento 2	...	Documento 1	Documento 2	...
Término 1	$tf(t_i, d_j, a)$	$tf(t_i, d_j, a)$		$tf(t_i, d_j, a)$	$tf(t_i, d_j, a)$	
Término 2	$tf(t_i, d_j, a)$	$tf(t_i, d_j, a)$		$tf(t_i, d_j, a)$	$tf(t_i, d_j, a)$	
...	$tf(t_i, d_j, a)$	$tf(t_i, d_j, a)$		$tf(t_i, d_j, a)$	$tf(t_i, d_j, a)$	

Posteriormente se obtiene la matriz de frecuencia inversa del documento (ver tabla 5), esta matriz permite expresar la relevancia de un término en la colección de documentos y se obtiene aplicando la ecuación (2) del apartado 1.4.1.

Tabla 5: Matriz de frecuencia inversa del documento.	Frecuencia Inversa del Documento
Término 1	$idf(t_i, D)$
Término 2	$idf(t_i, D)$
...	

A continuación, se obtiene la matriz de valor TF-IDF, mediante la multiplicación de los valores obtenidos de la frecuencia normalizada por los valores de la frecuencia inversa, como se muestra en la ecuación (3) del apartado 1.4.1.

Mediante la matriz de valor TF-IDF obtenida se establece la matriz booleana, para esto se especifica una variable  $k$ . Para construir la matriz booleana se compara cada uno de los valores de la matriz TF-IDF con la variable  $k$ , por lo tanto, si  $tfidf(t_i, d_j, a, D) \geq k$  se coloca un 1 en la matriz booleana caso contrario se coloca un 0.

La última fase de la metodología corresponde a la selección de los criterios de relevancia, el primer paso de esta fase consiste en elaborar la matriz de instanciación para cada criterio (ver tabla 6). La primera columna de esta matriz muestra los valores de la matriz booleana para un único criterio evaluado en cada uno de los documentos, la segunda columna se

construye con el conjunto de documentos, es decir, si un documento está en el contexto del caso de estudio se coloca un 1 y si no se encuentra en el contexto se coloca un 0.

<p><i>Tabla 6: Matriz de instanciación. Valores de matriz booleana para un único criterio, evaluado en cada documento</i></p>	<p><b>Documentos dentro del contexto (1)</b></p>
	<p><b>Documentos que no están en el contexto (0)</b></p>
	<p>0 v 1</p>
	<p>0 v 1</p>
<p>...</p>	<p>...</p>

El siguiente paso consiste en la elaboración de la matriz de comportamiento (ver tabla 7) que consta de tres columnas. La primera columna indica la presencia del criterio en los documentos, el valor 1 si se encuentra presente o el valor 0 si no se encuentra presente, esta columna representa la variable  $p$  de las tablas de verdad; la segunda columna muestra los valores relacionados al contexto de estudio y representa la variable  $q$ ; la tercera columna muestra el resultado de la comparación de cada par de valores, es decir, se busca en la matriz de instanciación el par de la tabla de verdad de valores lógicos, representados por la columna 1 y columna 2 de la matriz de comportamiento, si un par coincide se coloca un 1 y un 0 en el caso que no exista ninguna coincidencia.

<p><i>Tabla 7: Matriz de comportamiento. Valores lógicos</i></p>		<p><b>Criterios</b></p>
<p><b>El "criterio i" está presente en los documentos de entrada</b></p>	<p><b>Contexto evaluado</b></p>	<p><b>C1 ... Cn</b></p>
<p>0</p>	<p>1</p>	<p>1 v 0</p>
<p>0</p>	<p>0</p>	<p>1 v 0</p>
<p>1</p>	<p>1</p>	<p>1 v 0</p>
<p>1</p>	<p>0</p>	<p>1 v 0</p>

Dependiendo del resultado de la columna de criterios de la matriz de comportamiento (ver tabla 7) se categoriza el criterio según la tabla de patrones de categorización (ver tabla 2).

## **1.5. Scrum**

Scrum, es un marco de trabajo que busca dividir el trabajo en Sprints que puedan ser completados dentro de un mes o menos, tiempo en el cual se crea un incremento del producto potencialmente desplegable [9].

### **El equipo Scrum**

La función del equipo Scrum es entregar productos de forma iterativa e incremental, este trabajo permite mantener un contacto eficiente entre las partes interesadas además de maximizar la retroalimentación. El equipo scrum se encuentra conformado por:

- Propietario del Producto (Product Owner): la responsabilidad del propietario del producto es la de tener una visión clara de lo que se quiere construir, y transmitir esa visión al equipo scrum.
- Equipo de desarrollo (Development Team): son aquellos profesionales encargados de realizar la entrega de un incremento funcional que potencialmente puede ser implementado en producción al final de cada sprint.
- Scrum Máster: es el encargado de asegurarse que el proceso Scrum es seguido de forma adecuada por el equipo Scrum.

### **Eventos en Scrum**

Los eventos en Scrum son periodos limitados de tiempo, la mayoría predefinidos con el fin de crear regularidad y minimizar la necesidad de reuniones. Entre los eventos en Scrum se encuentran:

- El Sprint (Sprint): es el evento más importante y el corazón de Scrum, es un periodo de tiempo de un mes o menos durante el cual se crea un incremento del producto utilizable y potencialmente desplegable. Todos los sprints deben tener una duración consistente a lo largo de todo el proyecto. Cada nuevo sprint comienza inmediatamente finalizado el sprint anterior. Cada sprint tiene un objetivo de lo que se construirá, un plan, un diseño, el trabajo del equipo y el incremento del producto resultante.
- Planificación del Sprint (Sprint Planning): es el evento donde se planifica el trabajo a realizar en el sprint.
- Revisión del Sprint (Sprint Review): al final de cada sprint se lleva a cabo una revisión del sprint que permite inspeccionar el incremento y a modificar la pila del producto (Product Backlog) en el caso de ser necesario.



- Retrospectiva del Sprint (Sprint Restrospective): es el evento que le permite al equipo Scrum inspeccionarse con el fin de crear un plan de mejoras.

### **Artefactos en Scrum (Scrum Artifacts)**

Los artefactos de Scrum son los elementos que se generan del resultado de aplicar Scrum, entre los utilizados en este proyecto se encuentran:

- Pila de producto (Product Backlog): es una lista de todas las características, requisitos, funcionalidades, todo lo que se conoce del producto que podría ser implementado en el producto final. El primer desarrollo de la pila de producto solo refleja los requisitos conocidos hasta ese momento, siendo posible la modificación de la pila en el transcurso del desarrollo del proyecto [9].
- Pila del Sprint (Sprint Backlog): es una lista de las tareas que debe ejecutar el equipo Scrum en el próximo Sprint.

Además de los artefactos mencionados, se utilizaron dos artefactos que se describen a continuación, cuyo objetivo es facilitar el desarrollo de la pila de producto y la pila del sprint, además de poder mostrar gráficamente el desarrollo de las tareas dentro de un sprint:

- Historias de usuario: en el desarrollo de software una historia de usuario es una descripción en lenguaje natural de una o más características de un software. En el desarrollo ágil es una herramienta para recolectar la descripción de una funcionalidad del software desde la perspectiva del usuario final. Las historias de usuario permiten tener una idea simplificada de un requisito.
- Burndown Chart: es una herramienta visual que permite visualizar el trabajo restante en contraposición al tiempo. Para el trabajo restante, que se ubican en el eje vertical, se puede utilizar diferentes medidas como puntos de historias, días ideales, etc. Mientras que en el eje horizontal se muestran los sprint del proyecto. Esta herramienta se debe modificar al final de cada Sprint.

## **1.6. Resumen del capítulo**

La revisión documental de ingeniería conlleva un proceso extenso para el investigador, la metodología planteada por Mayra Carrión no es la excepción. La metodología estudiada consiste en la revisión de un corpus de documentos para categorizar un grupo de criterios claves, este proceso puede conllevar mucho tiempo para el investigador dependiendo del número de archivos pdf y el número de criterio seleccionados. El presente proyecto tiene como objetivo principal el desarrollo de una aplicación web utilizando principios de

desarrollo ágil con Scrum, para la determinación de criterios de relevancia, es decir, para automatizar la metodología planteada por Mayr Carrión. Mediante el estudio de la medida numérica TF-IDF para la ponderación de consultas y el uso de tablas de verdad para la inferencia de criterios suficientes y necesarios.

La medida numérica TF-IDF (frecuencia de termino – frecuencia inversa de documento) el primer término describe el número de apariciones de una palabra dentro de un documento, el segundo término permite asignar un menor peso a las palabras más frecuentes y asignar un peso mayor a las palabras menos frecuentes. Así, la medida permite determinar la relevancia entre las palabras, los documentos y las consultas particulares.

Las tablas de verdad junto a la lógica matemática permiten determinar la relevancia de un criterio y categorizarlo, la metodología estudiada presenta la siguiente clasificación de criterios:

- Necesario y no suficiente
- Suficiente y no necesario
- Necesario y suficiente
- No necesario y no suficiente
- Ninguno

La metodología utilizada para la inferencia de criterios propuesto por Mayra Carrión está dividida en tres fases:

- La primera fase corresponde a la recolección de documentos relacionados con la temática a investigar y la selección de criterios y sus correspondientes sinónimos.
- La segunda fase se divide en dos procesos independientes, el proceso cualitativo: el cual se base en la opinión del investigador, y el proceso cuantitativo basado en el cálculo TF-IDF, ambos procesos tienen como resultado una matriz booleana.
- La tercera fase corresponde a la selección de los criterios de relevancia mediante la comparación de la matriz booleana y las tablas de verdad.

## CAPÍTULO 2

### 2.1. Metodología

En el capítulo anterior, en la sección 1.5 se describe la metodología Scrum y los diferentes componentes de la metodología utilizados en el desarrollo de este proyecto. Dado que el objetivo general del proyecto es la creación de una aplicación web, se escogió la metodología ágil Scrum que se enfoca en el desarrollo iterativo e incremental, donde los requisitos y soluciones pueden evolucionar con el tiempo según la necesidad del proyecto.

### 2.2. Arquitectura, herramientas y recursos utilizados

#### 2.2.1. Patrón Modelo Vista Controlador (MVC)

La arquitectura utilizada en este proyecto es la arquitectura Modelo Vista Controlador (MVC), lo cual busca separar al código en tres capas diferentes interrelacionadas. Este modelo se ha vuelto muy utilizado en los frameworks de desarrollo web como es el caso de Django, framework con el que se desarrolló la aplicación de este proyecto.

- Modelo: esta capa contiene la esencia de la aplicación, es la encargada de trabajar con los datos. Dentro de esta capa tendremos las funciones que permitan extraer la información de los archivos pdf para aplicar la medida TF-IDF y la posterior comparación con las tablas de verdad.
- Vista: esta capa contiene las “ventanas” que permiten la interacción del usuario
- Controlador: el controlador es un objeto que permite manipular la vista, esta capa sirve de enlace entre las vistas y el modelo. Se podría decir que el controlador maneja las entradas, mientras que la vista maneja las salidas [10].

En la figura 1 se muestra como interactúan las tres capas de arquitectura MVC.

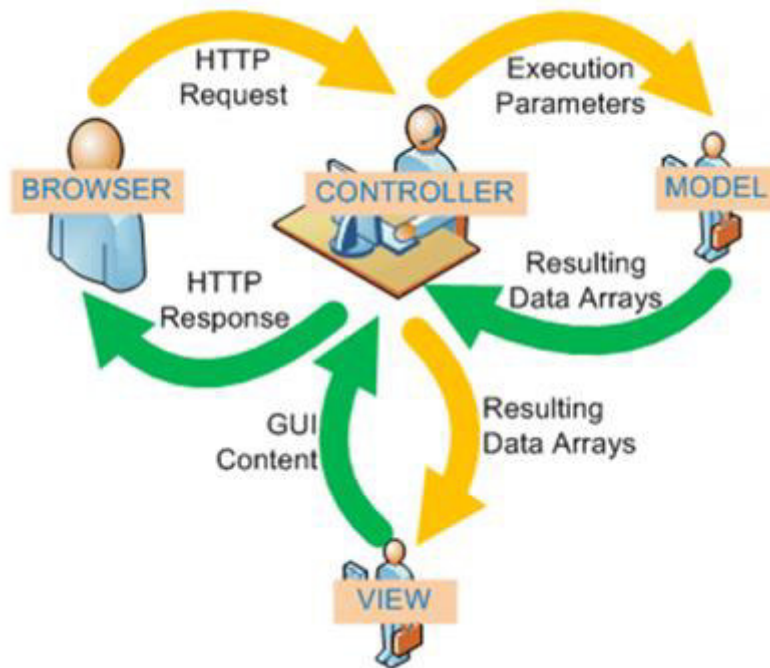


Figura 1: Arquitectura MVC [11].

### 2.2.2. Arquitectura general de la aplicación

Además del patrón MVC, al ser una aplicación web se encasillará en la arquitectura de este tipo de aplicaciones. En la figura 2 se muestra la arquitectura de la aplicación basada en una arquitectura web tradicional. Para esta aplicación, adicional a la base de datos que se encarga de almacenar la información de los usuarios, se utiliza espacio de memoria en el servidor como almacenamiento para los archivos pdf utilizados por los usuarios.

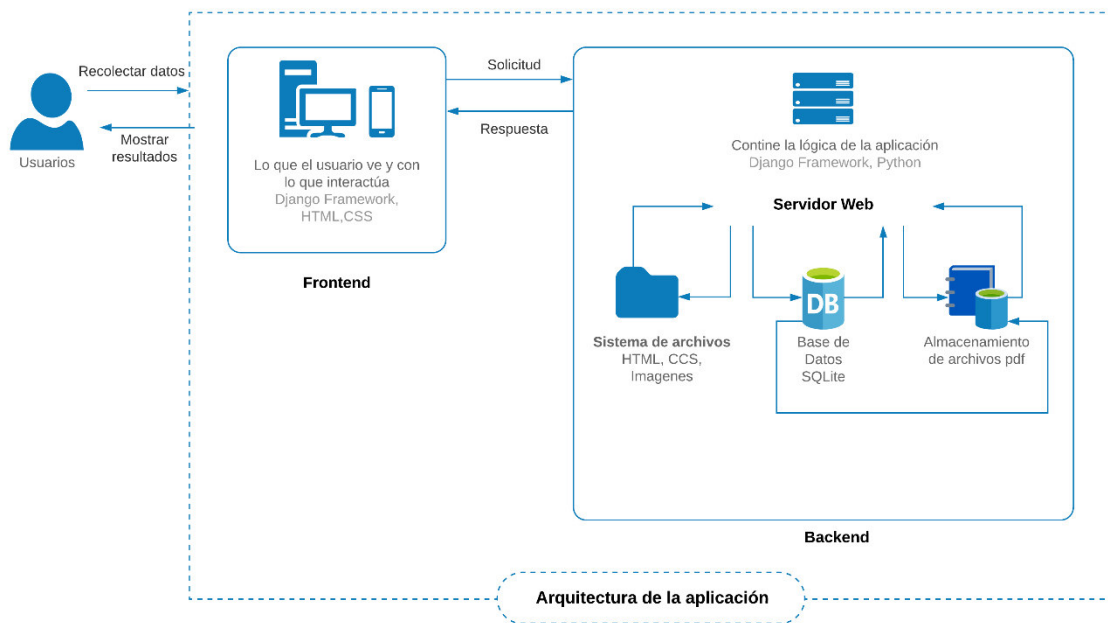


Figura 2: Arquitectura de la aplicación.

### 2.2.3. Elección de herramientas y recursos

Para el desarrollo de la aplicación web se utilizó el lenguaje Python mediante el framework Django. Para la implementación de la aplicación en un ambiente de producción se usaron de forma conjunta los servidores Gunicorn y Nginx.

A continuación, se presenta una descripción de cada una de las herramientas utilizadas.

Tabla 8: Herramientas y Recursos. Nombre	Descripción	Tipo
Python	Python es un lenguaje de programación interpretado de propósito general, de alto nivel. Creado en 1991 por Guido Van Rossum, la filosofía de Python enfatiza la legibilidad del código. Debido a que el código es compilado automáticamente a código de bytes y se ejecuta, Python es adecuado para su uso como lenguaje de scripting, lenguaje para implementación de aplicaciones web, etc [12].	Programación

Django	Django es un framework web de Python gratuito y de código abierto. Un framework web es un framework diseñado para soportar el desarrollo de aplicaciones web incluyendo servicios web, recursos web y API's web.	Programación
Gunicorn	Gunicorn 'Green Unicorn' es un servidor HTTP WSGI (Web Server Gateway Interface) de Python para UNIX. Este tipo de servidor esta desarrollado para la convención de llamadas para que los servidores web reenvíen solicitudes a aplicaciones o frameworks web escritos en el lenguaje de programación Python.	Servidor
Nginx	Nginx fue creado por Igor Sysoev y fue publicado por primera vez en 2004. Nginx es un servidor HTTP y proxy reverso gratuito, de código abierto y de alto rendimiento.	Servidor
AWS	Amazon Web Services (AWS) es una subsidiaria de Amazon que proporciona plataformas de computación en la nube y API bajo demanda.	Servidor

## 2.3. Fase inicial de Scrum

A continuación, se describe la fase inicial del proceso de scrum que corresponde al levantamiento de requerimientos, la identificación de los roles, y la creación de historias de usuario épicas, historias de usuario y el Product Backlog.

### 2.3.1. Definición de requerimientos

El objetivo de la aplicación es el de automatizar dos fases del proceso planteado en [8], como se describió en el apartado 1.4.3 de este documento. Para este fin se hizo uso de las historias de usuario épicas o Épicos, las cuales son historias muy amplias que aún no se han detallado por lo que no se pueden transformar en un incremento de producto. Estas

historias épicas se desglosan en historias de usuario más pequeñas [13]. A continuación, se describen las tres historias épicas planteadas:

*Tabla 9: Historia de usuario épica 1.*

<b>Historia de Usuario Épica</b>	<b>HUE01</b>
<b>Título:</b> Desarrollar el módulo de carga de información de archivos PDF	
<b>Descripción:</b> Como usuario, requiero poder cargar un aproximado de 70 archivos PDF y un promedio de 100 términos entre criterios y sus sinónimos asociados, para efectuar el proceso de determinación.	

*Tabla 10: Historia de usuario épica 2.*

<b>Historia de Usuario Épica</b>	<b>HUE02</b>
<b>Título:</b> Desarrollar el módulo de cálculo de criterios de relevancia.	
<b>Descripción:</b> Como usuario, requiero poder efectuar los cálculos que conllevan el proceso de determinación de criterios, utilizando los archivos PDF y los criterios ingresados en la aplicación.	

*Tabla 11: Historia de usuario épica 3.*

<b>Historia de Usuario Épica</b>	<b>HUE03</b>
<b>Título:</b> Desarrollar el módulo de administración de usuario.	
<b>Descripción:</b> Como usuario, requiero que se lleve un registro de los procesos realizados en la aplicación, es decir requiero guardar, borrar y modificar los procesos que se efectúen.	

### 2.3.2. Identificación de roles

La tabla 12 muestra los roles asignados a cada uno de los involucrado.

Tabla 12: Roles de Scrum.

Rol	Encargado	Descripción
Propietario del producto	Marco Santorum	Es el encargado de encaminar la visión del cliente hacia el equipo Scrum. Es el responsable de que la pila del producto (Product Backlog) se gestione de forma adecuada.
Scrum máster	Mayra Carrión	Permite la interacción entre el equipo de desarrollo y el dueño del producto con el fin asegurar que los objetivos planteados se alcancen alineados a las políticas y reglas de Scrum.
Equipo de desarrollo	Ronnie Nieto	Encargados de diseñar, desarrollar y realizar las pruebas correspondientes, al producto.

### 2.3.3. Historias de usuario

En el desarrollo de software, las historias de usuario son descripciones en lenguaje natural e informal de una o más características de un sistema de software [13]. Las historias de usuario son escritas desde el punto de vista de un usuario final y son de utilidad para que el equipo desarrollador comprenda el sistema y su contexto.

De acuerdo con los requerimientos planteados y las historias de usuario épicas, se definieron las siguientes historias de usuarios, usando el esquema planteado en [14]:



Tabla 13: Historia de usuario HU01.

Historia de Usuario		HU01
<b>Numero:</b> 1	<b>Usuario:</b> Administrador del sistema	
<b>Nombre historia:</b> Registro de los usuarios para el ingreso en el sistema.		
<b>Prioridad:</b> Alta	<b>Riesgo en desarrollo:</b> Alto	
<b>Iteración asignada:</b> 1	<b>Responsable:</b> Ronnie Nieto	
<p><b>Descripción:</b> Yo como administrador del sistema requiero que los usuarios se registren para poder utilizar la aplicación, con el fin de poder llevar un registro del uso de la aplicación.</p> <p><b>Descripción técnica:</b></p> <ul style="list-style-type: none"> <li>- Se creará un módulo de login que permita al usuario registrarse ingresando un correo electrónico, un nombre de usuario y una contraseña</li> </ul>		
<b>Observaciones:</b>		

Tabla 14: Historia de usuario HU02.

Historia de Usuario		HU02
<b>Numero:</b> 2	<b>Usuario:</b> Usuario del sistema	
<b>Nombre historia:</b> Definir la variable de sesión y estructura de la base de datos		
<b>Prioridad:</b> Alta	<b>Riesgo en desarrollo:</b> Alto	
<b>Iteración asignada:</b> 1	<b>Responsable:</b> Ronnie Nieto	
<p><b>Descripción:</b> Como usuario requiero poder mantener mi proceso actual (documentos y criterio) ingresados, cuando cambie de ordenador. Además, requiero poder acceder a procesos efectuados anteriormente.</p> <p><b>Descripción técnica:</b></p> <ul style="list-style-type: none"> <li>• Crear una estructura de sesión utilizando el Middleware de sesiones de Django para almacenar la referencia a los archivos pdf y los criterios ingresados.</li> <li>• Crear la estructura de la base de datos en SQLite utilizando el Middleware de modelos proporcionado por Django</li> </ul>		
<b>Observaciones:</b>		
La variable de sesión se elimina cuando el usuario cierra sesión, y se carga con los últimos datos registrados en la base de datos cuando el usuario inicia sesión.		

Tabla 15: Historia de usuario HU03.

Historia de Usuario		HU03
<b>Numero:</b> 3	<b>Usuario:</b> Usuario del sistema	
<b>Nombre historia:</b> Administrar procesos (guardar, borrar y modificar).		
<b>Prioridad:</b> Alta	<b>Riesgo en desarrollo:</b> Alto	
<b>Iteración asignada:</b> 1	<b>Responsable:</b> Ronnie Nieto	
<p><b>Descripción:</b> El usuario del sistema requiere guardar, borrar y modificar los procesos efectuados en la aplicación.</p> <p><b>Descripción técnica:</b></p> <ul style="list-style-type: none"> <li>– Mediante el uso de la variable de sesión en conjunto con la base de datos, se creará un módulo que le permita al usuario llevar un registro de los procesos efectuados, así como su modificación o eliminación.</li> </ul>		
<b>Observaciones:</b>		

Tabla 16: Historia de usuario HU04.

Historia de Usuario		HU04
<b>Numero:</b> 4	<b>Usuario:</b> Usuario del sistema	
<b>Nombre historia:</b> Cargar archivos PDF		
<b>Prioridad:</b> Alta	<b>Riesgo en desarrollo:</b> Alto	
<b>Iteración asignada:</b> 1	<b>Responsable:</b> Ronnie Nieto	
<p><b>Descripción:</b> Yo como usuario requiero poder cargar un promedio de 70 documentos en formato PDF. Requiero poder cargar varios documentos al mismo tiempo y en caso de cargar un único PDF, este se debe poder visualizar antes de realizar la carga del documento.</p> <p><b>Descripción técnica:</b></p> <ul style="list-style-type: none"> <li>• Procesar el archivo PDF, en el caso que sea necesario, con el fin de que se pueda extraer su contenido, previo a ser cargado a la aplicación.</li> <li>• Registrar al archivo PDF con un nombre único que impida confusión con otros archivos.</li> <li>• Registrar los archivos ingresados por el usuario en su variable de sesión.</li> </ul>		

**Observaciones:**

En el caso de archivos PDF encriptados con clave, el usuario deberá primero descryptar el archivo PDF de forma manual y posteriormente subir el archivo a la aplicación.

Tabla 17: Historia de usuario HU05.

Historia de Usuario		HU05
<b>Numero:</b> 5	<b>Usuario:</b> Usuario del sistema	
<b>Nombre historia:</b> Ingreso de criterios y sus sinónimos		
<b>Prioridad:</b> Alta	<b>Riesgo en desarrollo:</b> Medio	
<b>Iteración asignada:</b> 1	<b>Responsable:</b> Ronnie Nieto	
<p><b>Descripción:</b> Yo como usuario de la aplicación requiero poder ingresar un criterio (referencial) y sus respectivos sinónimos, sin un número límite de posibles términos para ingresar.</p> <p><b>Descripción técnica:</b></p> <ul style="list-style-type: none"> <li>– Registrar los criterios y sus respectivos sinónimos en la variable de sesión del usuario.</li> </ul>		
<b>Observaciones:</b>		

Tabla 18: Historia de usuario HU06.

Historia de Usuario		HU06
<b>Numero:</b> 6	<b>Usuario:</b> Usuario del sistema	
<b>Nombre historia:</b> Cálculo de criterios de relevancia.		
<b>Prioridad:</b> Alta	<b>Riesgo en desarrollo:</b> Alto	
<b>Iteración asignada:</b> 1	<b>Responsable:</b> Ronnie Nieto	
<p><b>Descripción:</b> Yo como usuario deseo determinar cuáles términos pueden considerarse de relevancia dentro de una revisión documental. Utilizando para este fin los archivos PDF y los criterios ingresados en el sistema.</p> <p><b>Descripción técnica:</b></p> <ul style="list-style-type: none"> <li>– La determinación de criterios de relevancia se efectúa mediante la aplicación de la medida numérica TF-IDF y la comparación de los resultados obtenidos con tablas de verdad.</li> </ul>		

<b>Observaciones:</b>

Tabla 20: Historia de usuario HU08.

Historia de Usuario		HU08
<b>Numero:</b> 8	<b>Usuario:</b> Usuario del sistema	
<b>Nombre historia:</b> Exportar resultados.		
<b>Prioridad:</b> Medio	<b>Riesgo en desarrollo:</b> Medio	
<b>Iteración asignada:</b> 1	<b>Responsable:</b> Ronnie Nieto	
<b>Descripción:</b> Yo como usuario de la aplicación requiero poder exportar los resultados obtenidos en un archivo Excel.		
<b>Descripción técnica:</b> <ul style="list-style-type: none"> <li>- El proceso de determinación mostrará como resultado seis tablas, las cuales se exportarán de forma de forma conjunta en un único archivo Excel estando cada una de las tablas en una hoja diferente del documento.</li> </ul>		
<b>Observaciones:</b>		

Las siguientes historias de usuario se añadieron en la ejecución de los sprints.

Historia de Usuario		HU09
<b>Numero:</b> 9	<b>Usuario:</b> Usuario del sistema	
<b>Nombre historia:</b> Seleccionar un método para realizar el cálculo de la variable $k$ .		
<b>Prioridad:</b> Medio	<b>Riesgo en desarrollo:</b> Medio	
<b>Iteración asignada:</b> 1	<b>Responsable:</b> Ronnie Nieto	
<b>Descripción:</b> Yo como usuario de la aplicación requiero poder calcular la variable $k$ , que permite efectuar el cálculo para la generación de la matriz booleana, de una de las siguientes formas: mediana, media, moda o varianza.		

<b>Descripción técnica:</b> <ul style="list-style-type: none"> <li>– Se creará un checkbox que permitirá al usuario seleccionar entre las tres opciones de cálculo de la variable <math>k</math></li> </ul>
<b>Observaciones:</b>

Tabla 21: Historia de usuario HU09.

Tabla 22: Historia de usuario HU10.

Historia de Usuario		HU10
<b>Numero:</b> 10	<b>Usuario:</b> Usuario del sistema	
<b>Nombre historia:</b> Mostrar gráficamente los resultados.		
<b>Prioridad:</b> Bajo	<b>Riesgo en desarrollo:</b> Bajo	
<b>Iteración asignada:</b> 1	<b>Responsable:</b> Ronnie Nieto	
<b>Descripción:</b> Yo como usuario de la aplicación requiero poder observar los resultados de forma gráfica, no únicamente las tablas resultantes.		
<b>Descripción técnica:</b> <ul style="list-style-type: none"> <li>– Se mostrará en un gráfico tipo pastel los resultados obtenidos.</li> </ul>		
<b>Observaciones:</b>		

#### 2.3.4. Product Backlog

El Product Backlog es una lista de todas las características que se deben incluir dentro del producto y es la única fuente de requisitos. El propietario del producto es la persona encargada del product backlog, así como de su contenido, la disponibilidad y el orden que debe tener cada ítem dentro de la lista. El product backlog es una lista dinámica que se puede ir modificando mientras avanza el desarrollo del proyecto, la primera versión del product backlog solo contiene los requisitos conocidos y mejor entendidos al principio del proyecto [9].

La construcción de producto Backlog, como se muestra en la siguiente tabla, se basa en las historias de usuario tomando en cuenta el riesgo de desarrollo y la prioridad de negocio de cada una de las historias de usuario.

La tabla 23 muestra el product backlog inicial.

Tabla 23: Product Backlog inicial.

<b>Product Backlog</b>			
<b>Historia de Usuario</b>	<b>Enunciado</b>	<b>Prioridad</b>	<b>Riesgo</b>
HU01	Registro de los usuarios para el ingreso en el sistema.	Alta	Alto
HU02	Definir la variable de sesión y estructura de la base de datos	Alta	Alto
HU03	Administrar procesos (guardar, borrar y modificar).	Alta	Alto
HU04	Cargar archivos PDF.	Alta	Alto
HU06	Cálculo de criterios de relevancia.	Alta	Alto
HU05	Ingreso de criterios y sus sinónimos.	Alta	Medio
HU08	Exportar resultados.	Medio	Medio
HU07	Omitir referencias en la búsqueda de criterios.	Medio	Bajo

La tabla 24 muestra el Product Backlog final.

Tabla 24: Product Backlog final.

<b>Product Backlog</b>			
<b>Historia de Usuario</b>	<b>Enunciado</b>	<b>Prioridad</b>	<b>Riesgo</b>
HU01	Registro de los usuarios para el ingreso en el sistema.	Alta	Alto
HU02	Definir la variable de sesión y estructura de la base de datos.	Alta	Alto
HU03	Administrar procesos (guardar, borrar y modificar).	Alta	Alto
HU04	Cargar archivos PDF.	Alta	Alto
HU06	Cálculo de criterios de relevancia.	Alta	Alto
HU05	Ingreso de criterios y sus sinónimos.	Alta	Medio
HU09	Seleccionar un método para realizar el cálculo de la variable $k$	Medio	Medio
HU08	Exportar resultados.	Medio	Medio

HU07	Omitir referencias en la búsqueda de criterios.	Medio	Bajo
HU10	Mostrar gráficamente los resultados.	Bajo	Bajo

## 2.4. Desarrollo de la aplicación

### 2.4.1. Sprint 0

#### Objetivos del Sprint

Desarrollar el módulo de administración de usuarios y de procesos.

#### Sprint Backlog

La tabla 25, muestra las historias de usuario y las tareas que se ejecutaron durante este sprint.

Tabla 25: Sprint 0 Backlog.

Código	Historia de Usuario	Tarea	Criterio de Aceptación
HU01	Registro de los usuarios para el ingreso en el sistema.	Crear un módulo para el registro e ingreso de usuarios.	Registro de un usuario en el sistema.
		Desarrollar la interfaz que permita visualizar la información del usuario.	Mostrar en la interfaz la información del usuario (correo, nombre de usuario y fecha del último ingreso a la aplicación). Además, mostrar los procesos registrados por el usuario.
HU02	Definir la variable de sesión y estructura de la base de datos	Crear la estructura de sesión usando el Middleware de sesiones de Django.	Registro de la información (pdf, criterios y sinónimos, cálculo de la variable k) ingresada por el usuario en la estructura de sesión.
		Crear la estructura de la base de datos para almacenar la información del usuario y los procesos realizados.	Registro de la información de un usuario y un proceso relacionado.
HU03	Administrar procesos (guardar, borrar y modificar).	Crear las funciones que permitan guardar,	Guardar, borrar y modificar un proceso.

		borrar y modificar un registro (proceso).	
--	--	---	--

### Requisitos completados del Sprint

Desarrollar el módulo de administración de usuarios y de procesos.

El objetivo del sprint 0 consiste en crear dos módulos, uno para el registro de los usuarios en la aplicación y el segundo para la administración de los procesos creados por el usuario en la aplicación, es decir, el módulo de administración de procesos le permitirá al usuario guardar, borrar y modificar los procesos efectuados en la aplicación. A continuación, se describe la ejecución de las tareas de las correspondientes historias de usuario:

- **Registro de los usuarios para el ingreso en el sistema:** La primera tarea consistió en la creación de un módulo para el registro de usuarios en el sistema, para esto se utilizó el sistema de autenticación otorgado por Django. Este sistema maneja tanto la autenticación como la autorización. Para registrarse en el sistema se desarrolló una interfaz donde el usuario debe ingresar un correo electrónico, un nombre de usuario único y una contraseña. Ingresados los datos correspondientes el usuario podrá acceder a la página de inicio de la aplicación.  
Para mostrar la información del usuario se desarrolló una interfaz accesible desde cualquier parte de la aplicación, esta interfaz consta de dos secciones: la primera corresponde a la información del usuario: correo electrónico, nombre de usuario y fecha del último ingreso a la aplicación; la segunda sección corresponde a los procesos registrados por el usuario, en esta sección se muestra una tabla con el nombre del proceso, la fecha en la que se creó el proceso, el estado del proceso (si el proceso se encuentra o no seleccionado y se están realizando modificaciones a este), y dos botones que permiten seleccionar y eliminar un proceso.
- **Definir la variable de sesión y estructura de la base de datos:** Una sesión es un intercambio temporal de información que ocurre entre una computadora y un usuario. Esta sesión se establece en un punto determinado y se termina en un punto posterior de tiempo. Las sesiones suelen tener estados, lo que implica que al menos una de las dos partes registre la información del estado actual de la sesión. Una variable de sesión es la estructura de datos que nos permite almacenar la información relacionada a la sesión. El framework de sesiones de Django permite almacenar y recuperar datos por visitante. Con este framework de sesión cada objeto HttpRequest, el cual es el primer argumento en cualquier función de Django, tendrá un atributo de sesión, que es un objeto similar a un diccionario [15].



La estructura de la variable de sesión definida para la aplicación posee la siguiente estructura:

Componentes de la variable de sesión:

- Archivos pdf grupo P: un arreglo de tuplas que contiene la referencia a la ubicación del archivo pdf y el nombre del pdf ingresado por el usuario.
- Archivos pdf grupo Q: un arreglo de tuplas que contiene la referencia a la ubicación del archivo pdf y el nombre del pdf ingresado por el usuario.
- Criterios y sus sinónimos: un arreglo que contiene los criterios y los sinónimos.
- Resultados: un arreglo con las matrices resultantes obtenidas del proceso de determinación.

A continuación, se muestra el esquema de la base de datos utilizada en la aplicación para el almacenamiento de la información del usuario y la información de los procesos realizados por el usuario.

## Esquema de base de datos

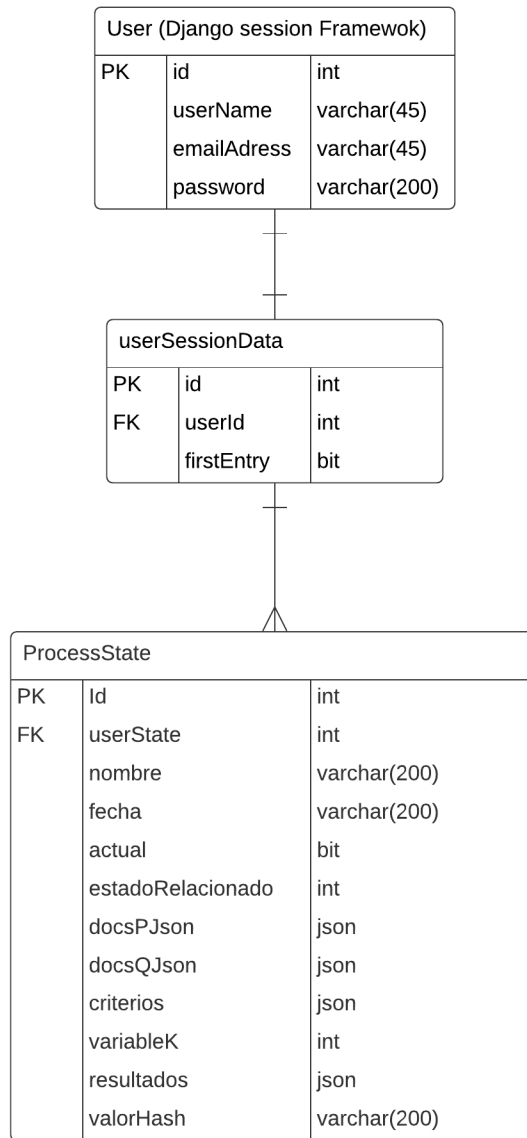


Figura 3: Esquema de base de datos.

- **Administrar procesos (guardar, borrar y modificar):** Con las estructuras definidas en las tareas previas, correspondientes a la variable de sesión y a la base de datos se procedió a crear las funciones que le permitan al usuario guardar, borrar y modificar los procesos realizados en la aplicación. La variable de sesión se usó como un almacenamiento intermedio entre la aplicación y la base de datos para agilizar los cálculos en la aplicación evitando efectuar muchas consultas hacia la base de datos. Para guardar un proceso el usuario únicamente ingresa un nombre

para el proceso, toda la información de la variable de sesión se almacena en la base de datos junto con el nombre ingresado y la fecha de creación. Para la eliminación de procesos el usuario debe seleccionar de la tabla de procesos, ubicada en la interfaz de información del usuario, el proceso que desee eliminar se le pedirá que confirme la operación y el proceso se eliminara de la base de datos junto con todos los archivos pdf relacionados a dicho proceso. La acción de modificar un proceso consiste en obtener los datos almacenados de un proceso seleccionado por el usuario, de la base de datos y almacenarlos en la variable de sesión; mediante el uso de una función hash podemos determinar si los datos del proceso fueron modificados, así el usuario podrá modificar el proceso seleccionado. Si se lo requiere el nombre del proceso puede ser modificando cuando el usuario registre los cambios efectuados en el proceso.

### **Sprint Review**

Se cumplió el objetivo del sprint que consistía en crear los módulos para el registro de usuarios y el módulo para la administración de procesos y de información. Como se puede observar en la tabla 26 los criterios de aceptación establecidos para este sprint se cumplieron.

<i>Tabla 26: Sprint 0 Review.</i> Código	Historia de Usuario	Criterio de Aceptación	Cumplido
HU01	Registro de los usuarios para el ingreso en el sistema.	Registro de un usuario en el sistema.	Si
		Mostrar en la interfaz la información del usuario (correo, nombre de usuario y fecha del ultimo ingreso a la aplicación). Además, mostrar los procesos registrados por el usuario.	Si
HU02	Definir la variable de sesión y estructura de la base de datos	Registro de la información (pdf, criterios y sinónimos, cálculo de la variable k) ingresada por el usuario en la estructura se sesión.	Si
		Registro de la información de un usuario y un proceso relacionado.	Si

HU03	Administrar procesos (guardar, borrar y modificar).	Guardar, borrar y modificar un proceso.	Si
------	---	---	----

### Sprint Retrospective

Los dos módulos plantados se desarrollaron satisfactoriamente. A continuación, se observa en la figura 4 el burndown chart que muestra el avance del trabajo durante el sprint. Como se puede observar en la figura, en la ejecución de las tareas correspondientes a la creación de la variable de sesión, existe un incremento en el tiempo que tomo ejecutar dichas tareas, esto debido a cambios que se presentaron en la ejecución del sprint referentes a la estructura en la que se almacenan los criterios y sus sinónimos.

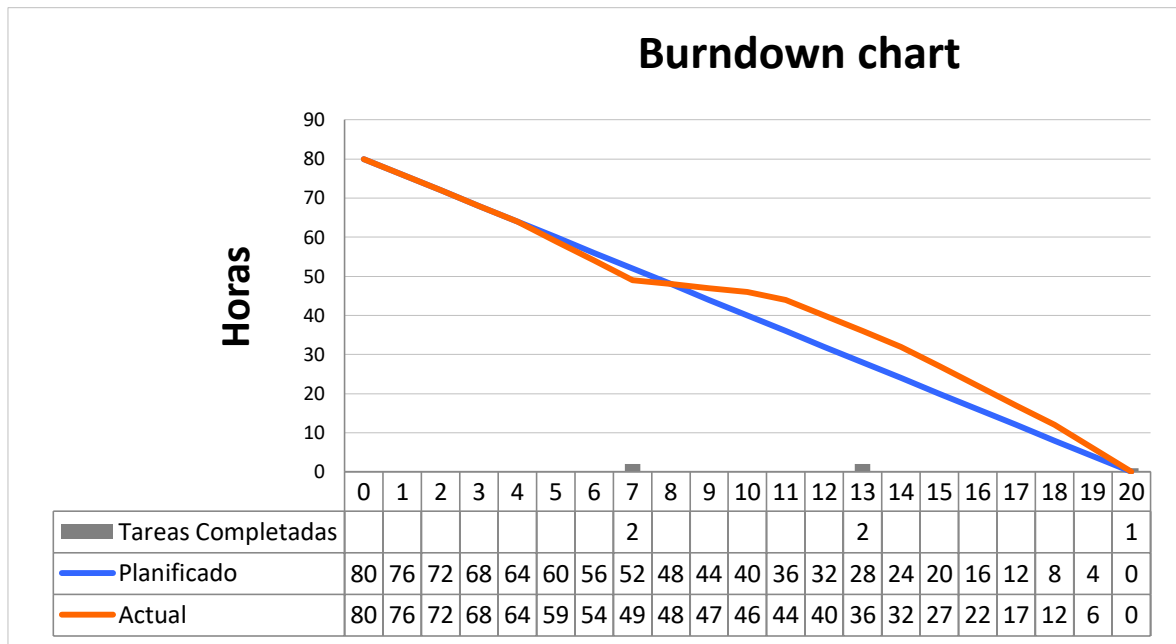


Figura 4: Burndown Chart Sprint 0.

A continuación, las figuras 5 y 6 presentan el avance obtenido durante el sprint.

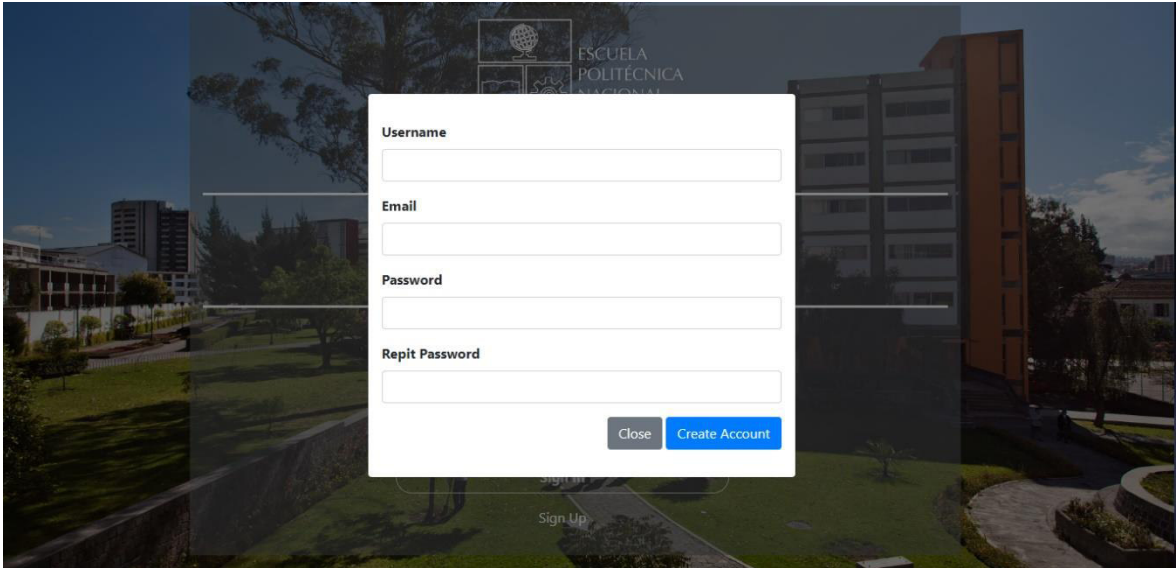


Figura 5: Interfaz para el registro de usuarios.

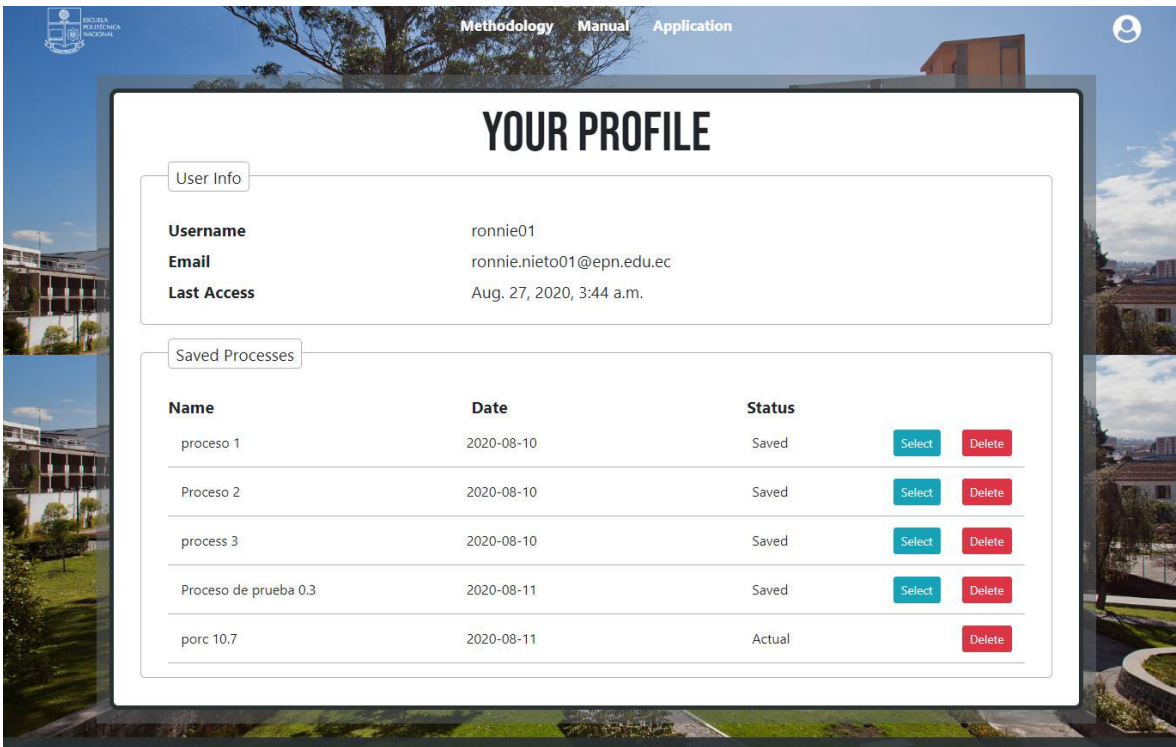


Figura 6: Interfaz de información del usuario.

## 2.4.2. Sprint 1

### Objetivos del Sprint

Desarrollar los módulos que permitan realizar la carga de archivos pdf y de criterios con sus correspondientes sinónimos.

### Sprint Backlog

La tabla 27, muestra la historia de usuario y las tareas que se ejecutaron durante este sprint.

Tabla 27: Sprint 1 Backlog.

Código	Historia de Usuario	Tarea	Criterio de Aceptación
HU04	Cargar archivos PDF	Desarrollar la interfaz que permita seleccionar los archivos PDF que se van a almacenar.	Mostrar en la interfaz el nombre del archivo PDF seleccionado o el total de archivos seleccionados para la carga.
		Desarrollar un visualizador de archivos para el caso en que se cargue solo un archivo PDF.	Mostrar en el visualizador el archivo PDF seleccionado por el usuario.
		Extraer la información del archivo PDF en una cadena de caracteres y almacenarlo en una estructura de datos (arreglo).	Registro de la cadena de caracteres correspondiente a cada archivo PDF, en el arreglo de datos.
		Registrar la ubicación y el nombre del archivo PDF ingresado dentro de la variable de sesión.	Actualización de la variable de sesión con el registro del nuevo archivo PDF ingresado.
HU05	Ingreso de criterios y sus sinónimos	Desarrollar la interfaz que permita ingresar un criterio con sus respectivos sinónimos.	Mostrar en la interfaz una tabla con el criterio y sus sinónimos ingresados.
		Registrar los criterios y sus sinónimos en la variable de sesión.	Actualización de la variable de sesión con el registro del criterio y sus sinónimos.

#### Requisitos completados del Sprint

El objetivo del sprint 1 consiste en desarrollar el módulo que permitirá realizar la carga de los criterios con sus sinónimos. A continuación, se describe la ejecución de las tareas de la siguiente historia de usuario:

- **Cargar archivos PDF:** La carga de archivos PDF constituye una de las tareas críticas dentro del desarrollo de la aplicación ya que no se limita únicamente a almacenar el archivo PDF dentro del servidor para que pueda ser utilizado dentro de la aplicación, esta tarea conlleva además el procesamiento del archivo, es decir,

una vez ingresado el archivo por el usuario la aplicación se encarga de extraer el texto y codificarlo de tal forma que el archivo pueda ser almacenado en una cadena de caracteres. Cabe mencionar que aquellos archivos en español deben conservar todos aquellos caracteres especiales propios del lenguaje como tildes, la letra (ñ), diéresis, entre otros. Previo al procesamiento de los archivos el usuario tiene la posibilidad de cargar los archivos de dos formas: la primera consiste en seleccionar varios archivos al mismo tiempo y cargarlos de forma simultánea; la segunda forma consiste en seleccionar un archivo a la vez y revisarlo mediante el visualizador de archivos PDF que se encuentra en la interfaz, para posteriormente realizar la carga del documento en el sistema, en esta alternativa de carga de archivos existe la posibilidad de modificar el nombre con el que se almacenará el archivo, lo que no ocurre en la carga simultánea en cuyo caso los archivos se almacenan con el nombre predefinido en el sistema del usuario.

Con el fin de evitar confusiones al realizar la búsqueda de los archivos PDF almacenados en el servidor, al momento de realizar la carga del documento se le añade al nombre un número que corresponde a la fecha y la hora que se realiza la carga de dicho archivo, añadiendo este número único se evita una posible confusión si existen varios archivos con el mismo nombre. Con este nuevo nombre se almacena el archivo en el servidor, y se guarda dentro de la variable de sesión del usuario una tupla que contiene una referencia a la ubicación del archivo y el nombre registrado por el usuario.

En cuanto a los archivos encriptados, cabe mencionar que la aplicación no desencripta aquellos documentos que contengan una contraseña, en esta situación el usuario debe desencriptar el archivo previo a su carga en la aplicación.

Para la carga, modificación (solo del nombre), y la eliminación de los archivos PDF se desarrolló en la aplicación dos ventanas independientes, una para cada grupo de documentos según su contexto. Para verificar que los archivos PDF se están cargando y almacenando de forma adecuada en la aplicación, se podrá observar en el visualizador de archivos PDF el documento seleccionado en el caso de que se realice la carga de un solo documento. En el caso en que se cargue más de un documento en la interfaz se podrá observar el número de documentos que se van a cargar a la aplicación. Además, se desarrolló una tercera ventana que permite observar en dos tablas, correspondientes a cada uno de los contextos, todos los documentos ingresados en la aplicación.

- **Ingreso de criterios y sus sinónimos:** Para el registro de criterios y sinónimos dentro de la interfaz de la aplicación se colocó dos TextArea, el primero corresponderá para el ingreso del criterio y el segundo TextArea permitirá al usuario ingresar un número indefinido de sinónimos. Con el propósito de distinguir un término de otro se debe ingresar un criterio por línea, así el salto de línea permitirá separar los términos para ser almacenados en un arreglo. Este arreglo de términos se almacena dentro de la variable de sesión del usuario.

Para verificar que los criterios y sus sinónimos se están almacenando de forma adecuado en la aplicación, se desplegara en la parte derecha de la pantalla una tabla con todos los criterios registrados por el usuario. Cada vez que se ingrese un nuevo criterio con sus respectivos sinónimos se añadirá a la tabla una nueva fila que contara con el número de criterio ingresado, el criterio principal y una lista con los sinónimos relacionados a dicho criterio. A su vez esta tabla contara con un botón que permita seleccionar el criterio y sus sinónimos para efectuar cualquier modificación deseada, y contara con un checkbox cuya finalidad es poder seleccionar uno o varios criterios para su eliminación.

### **Sprint Review**

Se cumplió el objetivo del sprint que consistía en desarrollar los módulos que permitan realizar la carga de los archivos pdf y de los criterios con sus sinónimos. Como se puede observar en la tabla 28 los criterios de aceptación establecidos para este sprint se cumplieron.

*Tabla 28: Sprint 1 Review.*

<b>Código</b>	<b>Historia de Usuario</b>	<b>Criterio de Aceptación</b>	<b>Cumplido</b>
HU04	Cargar archivos PDF	Mostrar en la interfaz el nombre del archivo PDF seleccionado o el total de archivos seleccionados para la carga.	Si
		Mostrar en el visualizador el archivo PDF seleccionado por el usuario.	Si
		Registro de la cadena de caracteres correspondiente a cada archivo PDF, en el arreglo de datos.	Si



		Actualización de la variable de sesión con el registro del nuevo archivo PDF ingresado.	Si
HU05	Ingreso de criterios y sus sinónimos	Mostrar en la interfaz una tabla con el criterio y sus sinónimos ingresados.	Si
		Actualización de la variable de sesión con el registro del criterio y sus sinónimos.	Si

### Sprint Retrospective

En la figura 7 se muestra el Burndown Chart del avance del sprint.

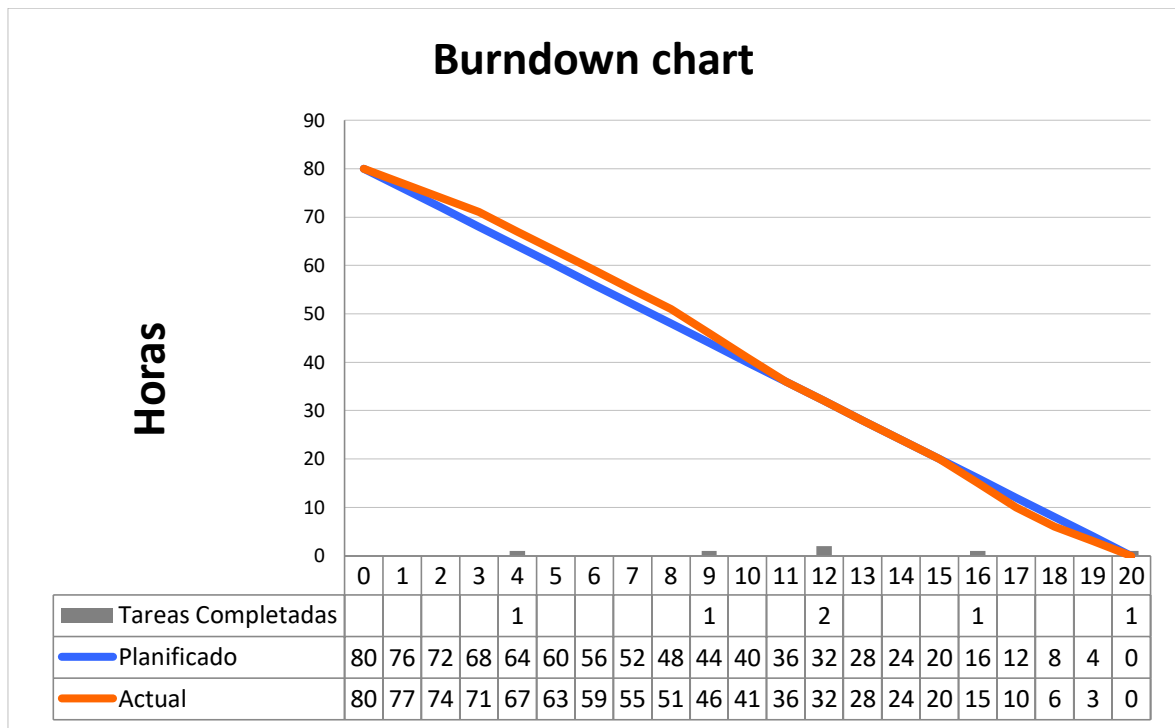


Figura 7: Burndown Chart Sprint 1.

A continuación, se presentan el avance obtenido durante el sprint.

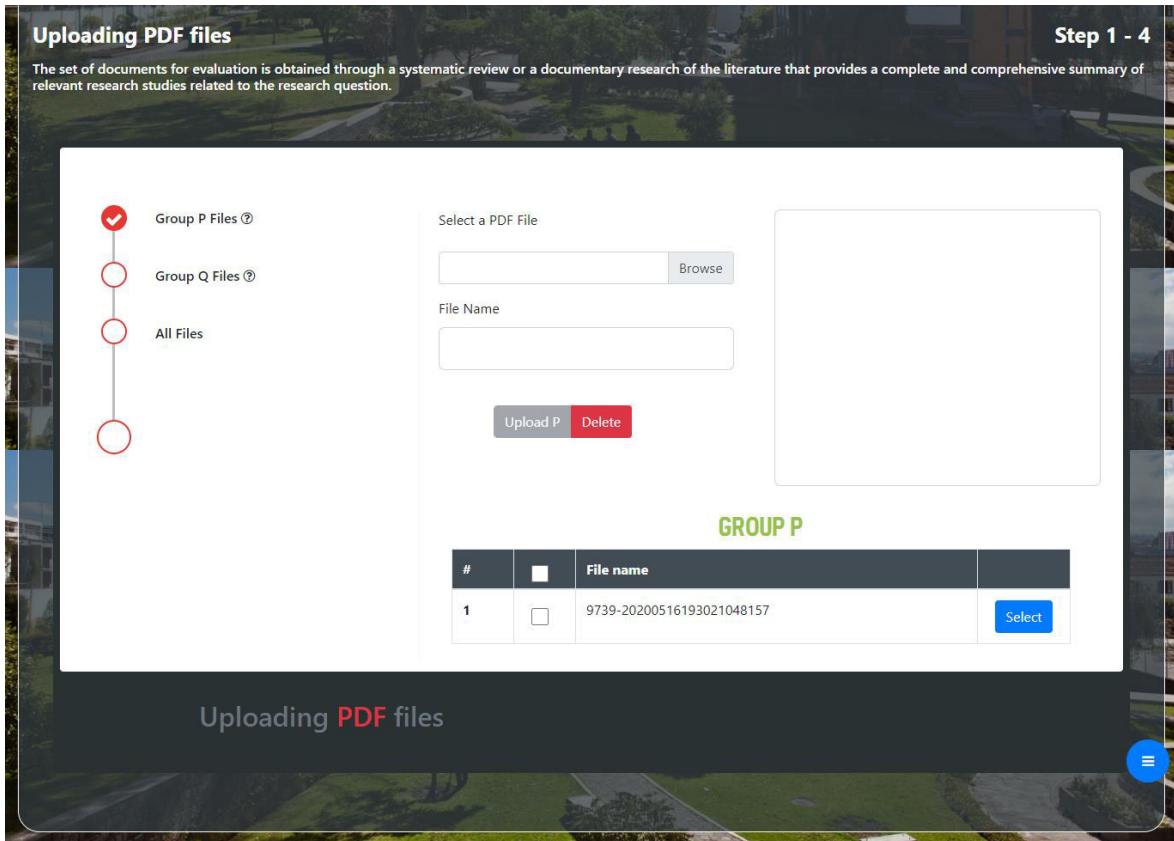


Figura 8: Interfaz para la carga de archivos PDF.

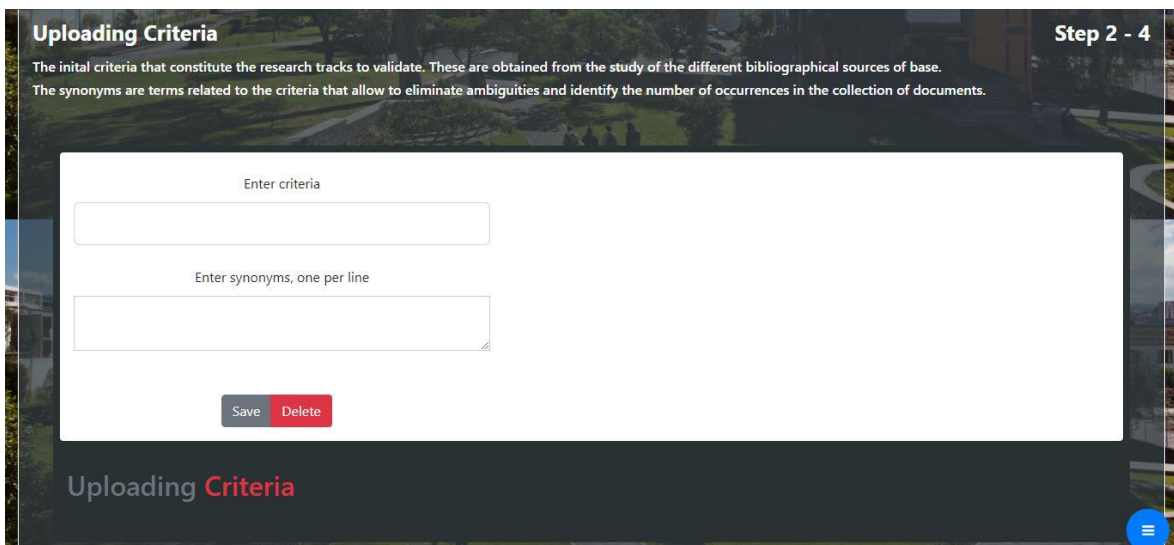


Figura 9: Interfaz para la carga de criterios y sinónimos.

### 2.4.3. Sprint 2

#### Objetivos del Sprint

Desarrollar el módulo de cálculo de criterios de relevancia y exponer los resultados obtenidos.

#### Sprint Backlog

La tabla 29, muestra las historias de usuario y las tareas que se ejecutaron durante este sprint. Además, en este sprint se generaron dos nuevas historias de usuario: HU09 y HU10, las cuales se añadieron en el Product Backlog.

Tabla 29: Sprint 2 Backlog.

Código	Historia de Usuario	Tarea	Criterio de Aceptación
HU07	Omitir referencias en la búsqueda de criterios.	Eliminar de la cadena de caracteres la sección correspondiente a las referencias.	La búsqueda de criterios dentro de un archivo pdf debe omitir la sección de referencias.
HU06	Cálculo de criterios de relevancia.	Obtener la matriz de frecuencia absoluta.	Mostrar en pantalla cada una de las matrices resultantes.
		Obtener la matriz resultante de aplicar la medida numérica TF-IDF.	
		Obtener la matriz booleana.	
		Comparar la matriz booleana con las tablas de verdad para determinar criterios de relevancia.	
HU08	Exportar resultados.	Crear una función que permita exportar las tablas resultantes del proceso en formato Excel.	Descarga del archivo Excel con los resultados obtenidos.
HU09	Seleccionar un método para realizar el cálculo de la variable $k$ .	Crear las funciones que permitan calcular la variable $k$ ya sea mediante la mediana, la media, la moda o la varianza.	Mostrar un <i>checkbox</i> que permite seleccionar entre las cuatro opciones: mediana, media, moda y varianza.
HU10	Mostrar gráficamente los resultados.	Crear un gráfico tipo pastel con los resultados de la tabla de categorización de criterios.	Mostrar el gráfico tipo pastel en la venta de resultados.

### Requisitos completados del Sprint

El objetivo del sprint 2 consiste en desarrollar el módulo que realizará los cálculos para obtener la relevancia de los criterios ingresados y mostrar los resultados obtenidos. A

continuación, se describe la ejecución de las tareas de las correspondientes historias de usuario:

- **Omitir referencias en la búsqueda de criterios:** Una de las tareas ejecutadas en el Sprint 0 corresponde a la extracción de la información de los archivos PDF para ser almacenada en una cadena de caracteres y posteriormente esta cadena ser almacenada en un arreglo. Previo a efectuar la búsqueda de cada uno de los términos dentro de las cadenas de caracteres, correspondientes a los archivos PDF ingresados, se debe tomar en cuenta que la sección de referencias dentro de los artículos científicos debe ser omitida ya que esta sección no forma parte de una manera directa de la investigación presentada en el artículo, por lo que contar las coincidencias encontradas en esta sección al realizar la búsqueda de los términos podría afectar los posteriores resultados que dependen de las coincidencias de los términos en los archivos PDF.

Para omitir la sección de referencias en cada una de las cadenas de caracteres, se realizó una división de la cadena principal en varias subcadenas que son almacenadas en una lista, siendo el término referencias o references en ingles el término separador. Al realizar este proceso existe la posibilidad de que la cadena principal se divida en más de dos subcadenas, al ser únicamente el resultado dos subcadenas existe la certeza que toda la sección de referencias va a ser omitida de la cadena original, en cambio al existir como resultado de la división más de dos subcadenas existe la posibilidad de que no toda la sección de referencias sea eliminada. En el segundo caso del proceso de eliminación, en el que existe más de una subcadena resultante, se elimina únicamente la última subcadena de la lista, suponiendo que la última coincidencia de los términos referencias o refereneses incluye la totalidad de dicha sección.

- **Cálculo de criterios de relevancia:** El objetivo de la aplicación es poder determinar los criterios de relevancia de una lista de criterios y sus sinónimos en un corpus de documentos. Para efectuar esta tarea se requiere que el usuario ingrese a la aplicación los documentos y los términos, en el caso de los archivos PDF se extrae su contenido en una cadena de caracteres para ser almacenado en dos arreglos dependiendo del contexto al que pertenece el documento (grupo P o grupo Q), de la misma forma todos los criterios y sus respectivos sinónimos son almacenadas en un arreglo.

Para obtener la relevancia de los criterios ingresados se efectúan cuatro procesos cuyo resultado o salida sirve como entrada del siguiente proceso. Así el primer proceso consiste en obtener la matriz de frecuencia absoluta, para conseguir este

resultado se itera por separado cada uno de los arreglos que contiene la información de los archivos PDF y a su vez se itera el arreglo de criterios con el fin de que en cada archivo se realice la búsqueda de los criterios y sus respectivos sinónimos, la suma del número de coincidencias referentes a un único criterio se almacena en una variable y posteriormente en un arreglo que representa la matriz de frecuencia absoluta.

Una vez obtenida la matriz de frecuencia absoluta el siguiente paso corresponde a calcular la mediana numérica TF-IDF para cada uno de los criterios ingresados, la primera parte de este proceso corresponde al cálculo de TF: como se muestra en la ecuación TF ( ver ecuación 1 del apartado 1.4.1), el cálculo requiere conocer el criterio con mayor coincidencia en cada uno de los documentos, por lo que se itera la matriz de tal forma que se compara el resultado de cada uno de los criterios en un único documento. Una vez encontrado el criterio con mayor coincidencia se almacena este valor en un nuevo arreglo que contendrá el criterio con mayor coincidencia por documento. Obtenido este arreglo el siguiente paso consiste en aplicar la fórmula TF iterando la matriz de frecuencia absoluta y el arreglo de criterios con la mayor concurrencia para obtener la matriz TF. La segunda parte del proceso corresponde al cálculo IDF, para esto se vuelve a iterar la matriz de concurrencia con el fin de determinar en cuantos documentos aparece un criterio, con este dato se ejecuta la ecuación IDF (ver ecuación 2 del apartado 1.4.1) y se obtiene un arreglo con los valores IDF para cada uno de los criterios. Finalmente se itera la matriz TF multiplicando cada uno de sus elementos con su valor correspondiente del arreglo IDF dando como resultado la matriz TF-IDF.

El siguiente proceso corresponde a la obtención de la matriz booleana, para este proceso se obtiene una variable  $k$  que corresponde a la media, mediana, moda o varianza de la matriz TF-IDF, según la selección realizada por el usuario. Obtenido este valor se itera la matriz TF-IDF y se compara cada uno de sus valores con la variable, si el valor de la matriz es mayor o igual a la variable se registra un 1, si el valor es menor a la variable se registra un 0.

Finalmente se compara la matriz booleana con las tablas de verdad, primero se crea un arreglo donde los documentos correspondientes al grupo P tienen un valor de 1 mientras que los documentos correspondientes al grupo Q tienen un valor de 0. Iterando en la matriz la fila correspondiente a cada uno de los criterios, se realiza la comparación con su similar en el arreglo de documentos; para registrar un 1 en el arreglo resultante debe coincidir con alguna de las tuplas de los valores de verdad, si no existe ninguna coincidencia se registra un 0. Este arreglo resultante se

compara con las tablas de verdad para obtener la relevancia de los criterios, en caso de no coincidir con ninguna de las tablas de verdad se coloca como valor “Ninguno”.

Para verificar que el proceso se está efectuando de forma adecuado, al finalizar la ejecución de los cálculos se despliega una ventana que contiene todas las matrices resultantes.

- **Exportar resultados:** Como resultado del proceso de determinación se obtiene seis matrices las cuales se despliegan en la interfaz de resultados como tablas, por requerimiento del usuario se podrá descargar un archivo Excel que contendrá una hoja de cálculo por cada una de las tablas resultantes.
- **Mostrar gráficamente los resultados:** El proceso de cálculo da como resultado seis tablas, la última tabla correspondiente a la categorización de los criterios se muestra en la venta de resultados mediante un gráfico tipo pastel, con la cantidad de criterios encontrados en cada una de las categorías existentes.
- **Seleccionar un método para realizar el cálculo de la variable k:** para la generación de la matriz booleana se utiliza una variable  $k$ , la cual se obtiene mediante la mediana, la media, la moda o la varianza de la matriz TF-IDF; para este fin se creó una ventana que contiene un *checkbox*, el usuario podrá seleccionar entre las cuatro opciones.

### Sprint Review

Se cumplió el objetivo del sprint que consistía en desarrollar el módulo de cálculo de los criterios de relevancia y mostrar los resultados obtenidos. Como se puede observar en la tabla 30 los criterios de aceptación establecidos para este sprint se cumplieron.

Tabla 30: Sprint 2 Review.

Código	Historia de Usuario	Criterio de Aceptación	Cumplido
HU07	Omitir referencias en la búsqueda de criterios.	La búsqueda de criterios dentro de un archivo pdf debe omitir la sección de referencias.	Si
HU06	Cálculo de criterios de relevancia.	Mostrar en pantalla cada una de las matrices resultantes.	Si
HU08	Exportar resultados.	Descarga de los archivos Excel con los resultados obtenidos.	Si

HU09	Seleccionar un método para realizar el cálculo de la variable $k$ .	Mostrar un checkbox que permite seleccionar entre las tres opciones: mediana, media y moda.	Si
HU10	Mostrar gráficamente los resultados.	Mostrar el gráfico tipo pastel en la venta de resultados.	Si

### Sprint Retrospective

Al observar la figura 10 correspondiente al burndown chart del sprint se puede notar que el desarrollo dentro del sprint tubo un tiempo de ejecución similar al estimado, existen pequeños intervalos donde el proceso se incrementó o disminuyo en ciertas tareas.

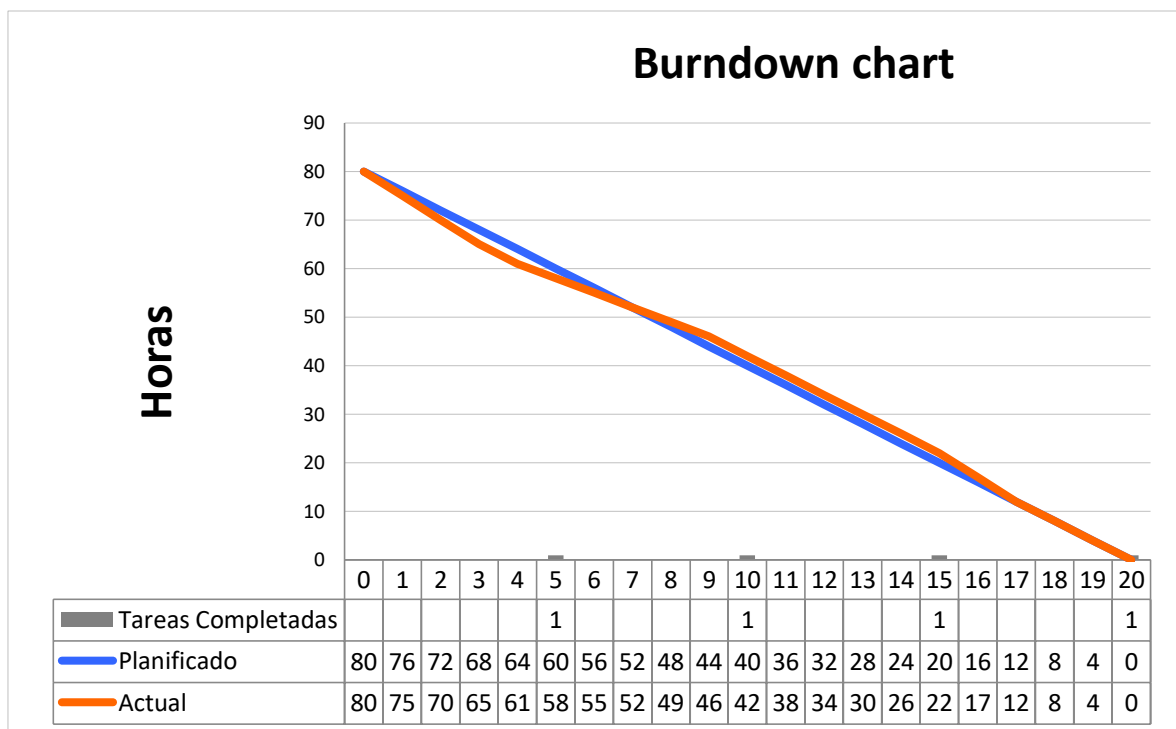


Figura 10: Burndown Chart Sprint 2.

A continuación, las figuras 11 y 12 presentan el avance obtenido durante el sprint.

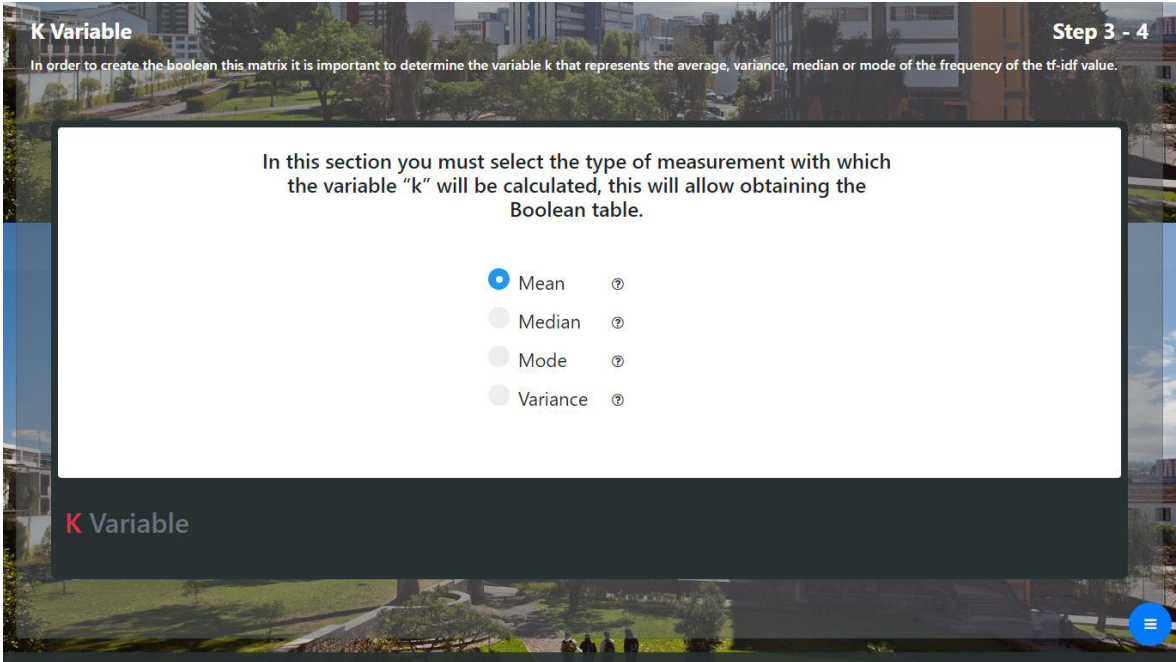


Figura 11: Interfaz para la selección de la variable k.

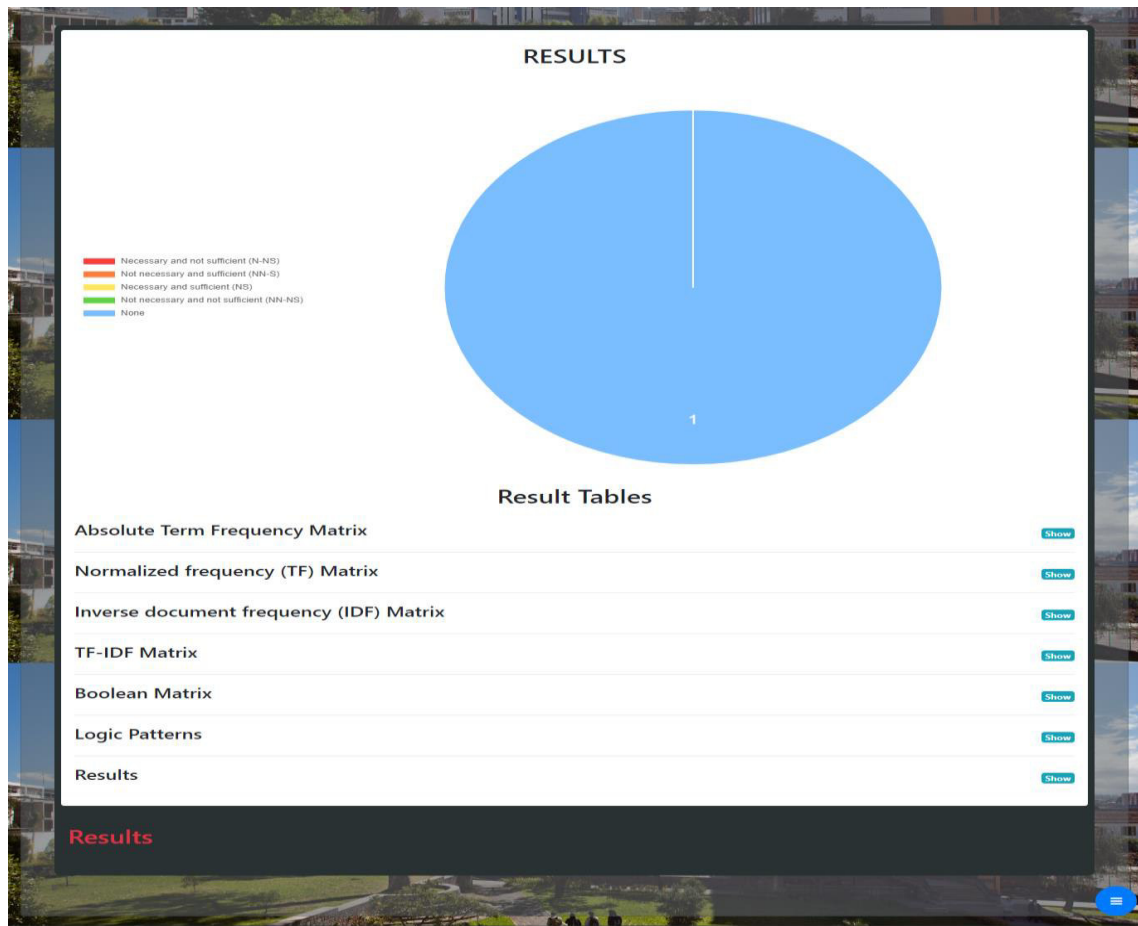


Figura 12: Interfaz de resultados.



## **2.5. Caso de estudio**

El uso de juegos serios como recurso pedagógico de apoyo al proceso de enseñanza y aprendizaje se ha convertido en una práctica recurrente gracias a la constante incursión de las tecnologías de la información en casi todos los ámbitos de la vida. Para diseñar juegos educativos serios que cumplan satisfactoriamente con su objetivo lúdico y pedagógico, es necesario conocer previamente cuáles son los aspectos fundamentales que se deben considerar en su diseño para obtener un producto de calidad.

Sin un diseño correcto del juego serio el alumno podría desmotivarse en su aprendizaje porque el juego no le permite experimentar un proceso de inmersión y motivación como el que se vive con los videojuegos comerciales.

El presente caso de estudio permite identificar qué criterios son clave cuando se diseñan juegos serios junto con la aplicación de metodologías tradicionales para el diseño de software, a través de la comparación de dos conjuntos de estudio entre los que se encuentran: las metodologías de desarrollo de software, sobre las cuales ya existe una sólida teoría y base conceptual; y metodologías serias de desarrollo de juegos, que son una tendencia emergente donde aún hay mucho por investigar.

### **Aplicación de la herramienta para el contexto de estudio.**

Esta sección describe el uso de la aplicación web en el caso de estudio mencionado. Al ingresar a la aplicación, el primer paso es crear una cuenta, un correo electrónico, un nombre de usuario único y una contraseña son los únicos requisitos solicitados.

Una vez creada la cuenta, se muestra la interfaz inicial con una descripción de la metodología implementada. Las interfaces de la metodología y el manual son obligatorias en el primer ingreso a la aplicación, en ingresos posteriores puede saltarse estas interfaces e ir a la aplicación.



Figura 13: Interfaz de descripción de la metodología.

Una vez se haya ingresado a la aplicación el primer paso consiste en cargar todos los documentos correspondientes a cada grupo de estudio (metodologías de desarrollo de software y metodologías para el diseño de juegos serios). Cada grupo de estudio está representado por el Grupo P y el Grupo Q respectivamente.

Solo se pueden cargar archivos PDF en la aplicación. Hay dos opciones para cargar archivos pdf. La primera opción es cargar un archivo a la vez, lo que permitirá ver el archivo y cambiar su nombre si es necesario. Ver figura 14.

La segunda opción es cargar varios archivos al mismo tiempo, en este caso solo se mostrará el número de archivos a cargar. Ver figura 15.



Figura 14: Carga de un archivo pdf.

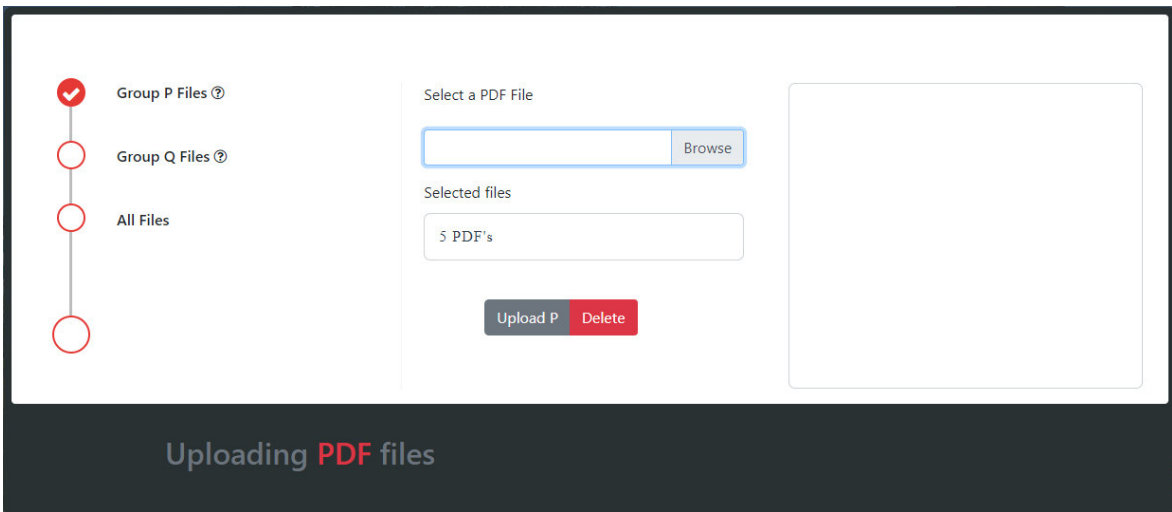


Figura 15: Carga de varios archivos pdf.

La sección de carga de archivos pdf contiene una interfaz que muestra todos los documentos cargados en la aplicación en su respectivo grupo de estudio. Ver figura 16.

Group P Files

Group Q Files

All Files

### GROUP P

#	File name
1	A serious game development process using competency - Saavedra
2	Agile Software Development Process Applied to the Serious Games Development for Children from 7 to 10 Years Old
3	Barbosa et al. - 2014 - A New Methodology of Design and Development of Serious Games-annotated
4	Cagatay - 2012 - A Methodological Approach For Serious Game Software Development An Application For Language Disorders-annotated
5	Cagatay, M. A methodological approach for serious game software development an application for language disorder
6	Cano et al. - 2016 - Towards a Methodology for Serious Games Design for Children with Auditory Impairments(2)-annotated
7	Cano et al. - 2016 - Towards a Methodology for Serious Games Design for Children with Auditory Impairments-annotated
8	De Lope et al. - 2015 - Design methodology for educational games based on interactive screenplays-annotated
9	EDoS An authoring environment for serious games design based on three models
10	EMERGO methodology and toolkit for efficient development of serious games in higher education
11	Jiménez-Hernández, Piattini, Revillagigedo - Tulais - 2016 - Methodology to construct educational video games in software engineering-annotated
12	López-Martínez, Miranda-Palma, González-Segura - 2013 - Una metodología para el análisis y diseño de un videojuego serio para combatir l-annotated
13	Najoua, Mohamed - 2018 - KASP A Cognitive-Affective Methodology for Designing Serious Learning Games-annotated
14	P. De Lope et al. - 2017 - Designing educational games key elements and methodological approach-annotated
15	Padilla - 2011 - Metodología Para El Diseño De Videojuegos Educativos Sobre Una Arquitectura Para El Análisis Del Aprendizaje Colaborati-annotated
16	Roungas, Dalpiaz - 2016 - A Model-driven Framework for Educational Game Design-annotated
17	Towards a Methodology for Serious Games Design for Children with Auditory Impairments
18	Using the DODDEL model to teach serious game design to novice designers
19	Cagatay, M. A 40-71
20	Cagatay, M. A 40-71-55-86

### GROUP Q

#	File name
1	Balduino - 2007 - Introduction to OpenUP (Open Unified Process)
2	Cockburn - 2001 - Agile Software Development
3	Compare el Proceso Unificado Esencial
4	El Proceso Unificado
5	GINDEV
6	Guía a Rational Unified Process
7	Hui et al. - 2015 - Compare Essential Unified Process (EssUP) with Rational Unified Process (RUP)
8	I. Jacobson, G. Booch - 2001 - The Unified Process
9	Ingeniería del software UN ENFOQUE PRÁCTICO
10	Introducción a OpenUP
11	KeSchwaber, Jeff - 2013 - La Guía de Scrum(5)
12	LA GUIA DE SCRUM
13	Las Metodologías de Cristal
14	MARCO DE SOLUCIONES MICROSOFT
15	Martínez - 2011 - Guía a Rational Unified Process
16	Methodologies of Agile Development like an Opportunity for the Engineering of Educative Software
17	Microsoft Solutions Framework
18	Morales et al. - 2010 - procesos de desarrollo para videojuegos
19	Pressman - 2012 - Ingeniería Del Software, un enfoque práctico
20	RACED
21	Running an Agile Software Development Project
22	SCRUM Y EXTREME GAME DEVELOPMENT EN UNA METODOLOGIA PARA DESARROLLO DE VIDEOJUEGOS EN ANDROID
23	SOMEMERVILLE
24	Una introducción al UML y al proceso unificado HUNT
25	Ze - 2014 - Microsoft Solutions Framework (MSF) for Agile Software Development

Uploading PDF files

Figura 16: Interfaz que muestra los archivos pdf cargados de los grupos P y Q.



El tercer paso consiste en seleccionar el método para calcular la variable k.

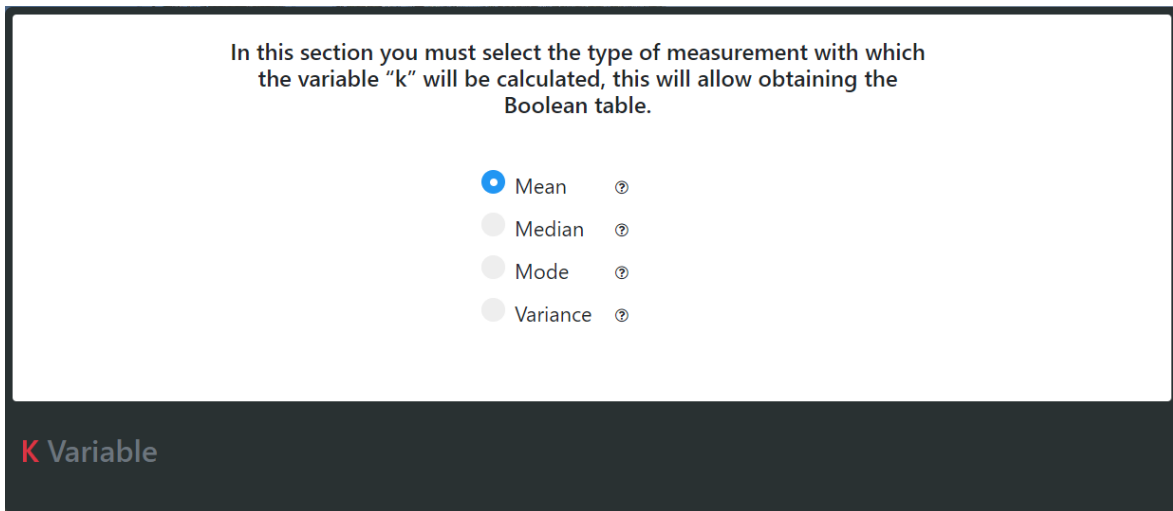


Figura 18: Interfaz para seleccionar la forma de calcular la variable k.

Los resultados del proceso se muestran en un gráfico tipo paste, en el lado izquierdo están las categorías en las que se puede categorizar un criterio, mientras que el gráfico muestra el número total de criterios y una lista de criterios por categoría. Ver figura 19.

En la parte inferior de la interfaz se encuentran todas las matrices obtenidas en el proceso.

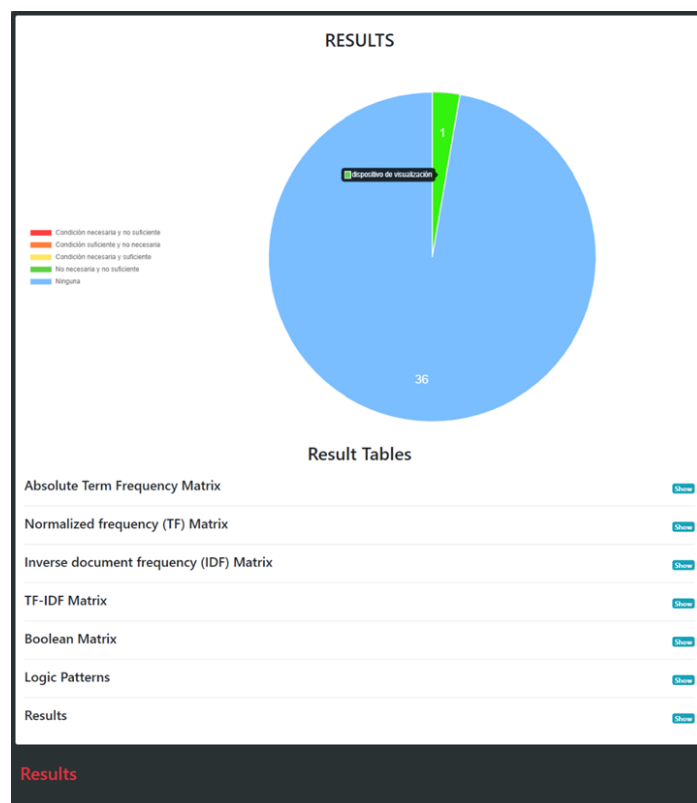


Figura 19: Interfaz de resultados.

Las matrices resultantes del caso de estudio obtenidas al usar el aplicativo se muestran en el Anexo 1: Resultados caso de estudio.

Con el uso de la aplicación se obtuvo el siguiente resultado: El criterio “Dispositivo de visualización” fue categorizado como “No es necesario ni suficiente”. El resto de los criterios no se incluyeron en ninguna categoría.

## **2.6. Fase de lanzamiento**

La fase de lanzamiento incluye la identificación de las lecciones aprendidas durante el proyecto y la implementación de la aplicación en un ambiente de producción.

### **Implementación en un ambiente de producción.**

Para la implementación de la aplicación en un ambiente de producción se utilizaron las siguientes herramientas: AWS, Gunicorn y Nginx. A continuación, se describe el proceso de implementación.

Para el alojamiento de la aplicación se optó por el uso del servicio EC2 (Amazon Elastic Compute Cloud) de AWS, ya que este servicio permite a los usuarios rentar computadoras virtuales para la ejecución de aplicaciones. Para este proyecto se configuró una máquina virtual o instancia con el sistema operativo Ubuntu Server 18.04 LTS de propósito general.

Una vez configurada la máquina virtual se realizó la instalación y configuración de Gunicorn y el servidor Nginx y la clonación del proyecto mediante git. Gunicorn se utilizó para servir la aplicación de forma local, mientras que para crear un entorno de producción se utilizó Nginx como un servidor proxy reverso, el cual procesa las solicitudes de la aplicación web para enrutarlas a través del servidor de aplicaciones Gunicorn evitando así que este servidor interactúe directamente con las peticiones del exterior. Finalmente se configuró un archivo Systemd Unit que permite la habilitación de un servicio para que la aplicación web se inicialice automáticamente.

Finalmente se registró el dominio key-criteria.com para identificar a la aplicación web desarrollada

### **Retrospectiva del proyecto**

En la tabla 31 se muestra la retrospectiva del proyecto.

Tabla 31: Retrospectiva del proyecto.

Pregunta	Respuesta
¿Qué salió bien?	<ul style="list-style-type: none"> <li>➤ El uso del framework Django facilitó la creación de los módulos de la aplicación, como, por ejemplo, el sistema de autenticación de Django facilitó la creación del módulo de autenticación de usuarios.</li> <li>➤ El uso de Python como lenguaje de desarrollo facilitó la extracción de la información y el procesamiento de esta.</li> </ul>
¿Qué salió mal?	<ul style="list-style-type: none"> <li>➤ La aplicación se renderiza de forma frecuente lo que impide un flujo continuo para el usuario al momento de utilizar la aplicación.</li> <li>➤ No existía un diseño inicial definido de la interfaz de la aplicación, lo que ocasionó cambios en la interfaz a lo largo del proyecto.</li> <li>➤ No se pudo programar una función que permita extraer la información de aquellos archivos pdf constituidos por imágenes.</li> </ul>
¿Qué obstáculos se presentaron?	<ul style="list-style-type: none"> <li>➤ La situación sanitaria global que se vivió en el transcurso del desarrollo de este proyecto impidió una adecuada ejecución de las pruebas de usabilidad.</li> <li>➤ Dificultad a la hora de extraer la información de los archivos pdf, ya que estos podían tener diferentes codificaciones o podían estar conformados por imágenes.</li> <li>➤ El lenguaje español contiene caracteres especiales tales como tildes, la letra ñ, entre otros, que impedían una comparación adecuada entre la información del archivo pdf y los términos ingresados por el usuario.</li> </ul>
Lecciones aprendidas	<ul style="list-style-type: none"> <li>➤ Crear un boceto inicial de las interfaces a implementar en la aplicación, con el fin de evitar cambios significativos a lo largo de la ejecución de proyectos.</li> <li>➤ El uso de frameworks web (Django), agilitan el desarrollo de aplicaciones permitiendo ahorrar recursos y tiempo.</li> </ul>



	<ul style="list-style-type: none"> <li>➤ Al extraer la información de los archivos pdf, se debe procesar la data mediante la codificación de la misma de tal forma que los caracteres especiales, puedan ser reconocidos o caso contrario que sean eliminados; esto con el fin de que la comparación entre la información contenida en los archivos pdf y los términos ingresados por el usuario se pueda efectuar de forma adecuado sin pérdida en la contabilización de coincidencias.</li> </ul>
--	---

## 2.7. Resumen del capítulo

En el capítulo 2 “Metodología” se presentan seis secciones. La primera y la segunda sección describen la metodología y las herramientas utilizadas en el desarrollo de la aplicación. La metodología utilizada es Scrum, mientras que las herramientas seleccionadas para el desarrollo fueron: Django como *framework* de desarrollo web, este framework está basado en Python. Para la implementación de la aplicación en producción se seleccionaron los servidores Nginx y Gunicorn, en conjunto con los servicios de *hosting* de AWS.

La tercera sección del capítulo presenta la fase inicial de scrum donde se definen los requerimientos de la aplicación en 3 historias de usuario épicas las cuales se desglosan en 10 historias de usuario. Dentro de la fase inicial se identifica los roles siendo: Marco Santorum como el propietario del producto, Mayra Carrión como el scrum máster y Marcelo Nieto como el equipo de desarrollo.

La cuarta sección del capítulo presenta el desarrollo de la aplicación en tres sprints. En el primer esprint se presenta el desarrollo del módulo de administración de usuarios y procesos, en el segundo sprint se desarrolló los módulos que permiten realizar la carga de archivos pdf y de criterios con sus respectivos sinónimos, el tercer esprint presenta el desarrollo del módulo que efectúa el cálculo de criterios de relevancia y la presentación de los resultados. Cada uno de los sprints presenta un *sprint review* que muestra el cumplimiento de los criterios de aceptación de cada una de las historias de usuario, el burndown chart que describe el tiempo de desarrollo en el sprint, y los gráficos de la aplicación correspondientes al avance alcanzado en el sprint.

La quinta sección del capítulo presenta el caso de estudio que se llevó a cabo, aquí describe la temática del caso de estudio y se presenta el uso de la aplicación desarrollada paso a paso.

Finalmente, la sexta sección del capítulo presenta la fase de lanzamiento de la aplicación, esta sección presenta la implementación de la aplicación en un ambiente de producción usando los servicios de AWS y los servidores Nginx y Unicorn. A su vez se presenta la retrospectiva del proyecto, lo que salió bien, lo que salió mal, los obstáculos que se presentaron y las lecciones aprendidas en el desarrollo del proyecto.

## CAPÍTULO 3

### Evaluación y Resultados

#### 3.1. Pruebas de funcionalidad

Para validar la funcionalidad de la aplicación se establecieron casos de prueba que permitan demostrar si se cumplen o no los requerimientos de desarrollo predefinidos. Los casos de prueba se establecieron a partir del formato propuesto en [16], el cual consiste en definir entradas y salidas esperadas en contra posición de las salidas obtenidas.

A continuación, se especifican los casos de estudio aplicados en cada uno de los módulos.

##### 3.1.1. Módulo de carga de archivos PDF

Para el módulo de carga de archivos PDF se definieron 9 casos de prueba y la tabla 32 muestra los resultados obtenidos y el cumplimiento de las diferentes funcionalidades.

Tabla 32: Casos de prueba módulo de carga de archivos PDF.

Código	Descripción del caso	Salida		Cumplimiento
		Esperada	Obtenida	
CPCA01	Verificar la interfaz del módulo de carga de archivos para el grupo P y Q.	Se muestra los recuadros con los componentes para la carga de archivos tanto para el grupo P como para el grupo Q.	Se muestra los recuadros con los componentes para la carga de archivos tanto para el grupo P como para el grupo Q.	Si
CPCA02	Verificar que el botón de carga este inhabilitado cuando no hay ningún archivo cargado.	El botón de carga está inhabilitado cuando no se seleccionado ningún archivo.	El botón de carga está inhabilitado cuando no se selecciona ningún archivo.	Si
CPCA03	Verificar que se despliegue el buscador de archivos cuando	Se muestra el buscador de archivos cuando	Se muestra el buscador de archivos cuando	Si

	se da clic sobre el botón Browse.	se da clic sobre el botón Browse.	se da clic sobre el botón Browse.	
CPCA04	Verificar que se muestra el nombre del archivo en el campo "File Name" cuando se selecciona un archivo.	Se muestra el nombre del archivo en el campo "File Name" cuando se selecciona el archivo.	Se muestra el nombre del archivo en el campo "File Name" cuando se selecciona el archivo.	Si
CPCA05	Verificar que se muestra el total de archivos en el campo "Selected files" cuando se seleccionan varios archivos.	Se muestra el total de archivos en el campo "Selected Files" cuando se seleccionan varios archivos.	Se muestra el total de archivos en el campo "Selected Files" cuando se seleccionan varios archivos.	Si
CPCA06	Verificar que se muestra en la tabla los archivos cargados por el usuario.	Se muestra en la tabla los archivos cargados por el usuario.	Se muestra en la tabla los archivos cargados por el usuario.	Si
CPCA07	Verificar que se muestre el archivo pdf y su nombre cuando se da clic sobre el botón "Select" en la tabla de archivos.	Se muestra el archivo pdf y su nombre cuando se da clic sobre el botón "Select" en la tabla de archivos.	Se muestra el archivo pdf y su nombre cuando se da clic sobre el botón "Select" en la tabla de archivos.	Si
CPCA08	Verificar que se marque la casilla de todos los archivos pdf cuando se	Se marcan las casillas de todos los archivos pdf cuando se selecciona el	Se marcan las casillas de todos los archivos pdf cuando se selecciona el	Si

	selecciona el primer check box de la tabla de archivos.	primer check box de la tabla de archivos.	primer check box de la tabla de archivos	
CPCA09	Verificar que se borren únicamente los archivos seleccionados (check box)	Se borran únicamente los archivos seleccionados.	Se borran únicamente los archivos seleccionados.	Si

**Observaciones:** Se presento un caso donde el usuario no podía seleccionar el archivo pdf de la tabla de archivos. Para solucionar el inconveniente se le pido al usuario que vuelva a cargar la página de la aplicación, una vez se volvió a cargar la página el problema se solucionó.

### 3.1.2. Módulo de carga de criterios y sinónimos

Para el módulo de carga de criterios y sinónimos se definieron 13 casos de prueba y la tabla 33 muestra los resultados obtenidos y el cumplimiento de las diferentes funcionalidades.

*Tabla 33: Casos de prueba módulo de carga de criterios y sinónimos.*

Código	Descripción del caso	Salida		Cumplimiento
		Esperada	Obtenida	
CPCC01	Verificar que se pueda ingresar un criterio en el campo "Enter Criteria".	Se puede ingresar un criterio en el campo "Enter Criteria".	Se puede ingresar un criterio en el campo "Enter Criteria".	Si
CPCC02	Verificar que se puede ingresar un sinónimo por línea en el campo "Enter synonyms, one per line".	Se puede ingresar un sinónimo por línea en el campo "Enter synonyms, one per line".	Se puede ingresar un sinónimo por línea en el campo "Enter synonyms, one per line".	Si

CPCC03	Verificar que se guarden el criterio y los sinónimos ingresados al dar clic en el botón "Save".	Se guarda el criterio y los sinónimos ingresados al dar clic en el botón "Save".	Se guarda el criterio y los sinónimos ingresados al dar clic en el botón "Save".	Si
CPCC04	Verificar que se muestre en la tabla de criterios, los criterios y sinónimos guardados.	Se muestre en la tabla de criterios, los criterios y sinónimos guardados.	Se muestre en la tabla de criterios, los criterios y sinónimos guardados.	Si
CPCC05	Verificar que se muestre en los campos correspondientes el criterio y los sinónimos cuando se da clic sobre el botón "Select" en la tabla de criterios.	Se muestre en los campos correspondientes el criterio y los sinónimos cuando se da clic sobre el botón "Select" en la tabla de criterios.	Se muestre en los campos correspondientes el criterio y los sinónimos cuando se da clic sobre el botón "Select" en la tabla de criterios.	Si
CPCC06	Verificar que se muestran los botones "Modify" y "New" cuando se da clic sobre el botón "Select" en la tabla de criterios.	Se muestran los botones "Modify" y "New" cuando se da clic sobre el botón "Select" en la tabla de criterios.	Se muestran los botones "Modify" y "New" cuando se da clic sobre el botón "Select" en la tabla de criterios.	Si
CPCC07	Verificar que se modifique el criterio o los	Se modifique el criterio o los sinónimos cuando	Se modifique el criterio o los sinónimos	Si

	sinónimos cuando se da clic sobre el botón "Modify".	se da clic sobre el botón "Modify".	cuando se da clic sobre el botón "Modify".	
CPCC08	Verificar que los campos de criterio y sinónimos se borran cuando se da clic sobre el botón "New".	Los campos de criterio y sinónimos se borran cuando se da clic sobre el botón "New".	Los campos de criterio y sinónimos se borran cuando se da clic sobre el botón "New".	Si
CPCC09	Verificar que se marque la casilla de todos los criterios cuando se selecciona el primer check box de la tabla de criterios.	Se marcan las casillas de todos los criterios cuando se selecciona el primer check box de la tabla de criterios.	Se marcan las casillas de todos los criterios cuando se selecciona el primer check box de la tabla de criterios.	Si
CPCC10	Verificar que se borren únicamente los criterios que hayan sido seleccionados (check box)	Se borran únicamente los criterios seleccionados.	Se borran únicamente los criterios seleccionados.	Si
CPCC11	Verificar que el botón "Next Step" se muestra únicamente cuando existen criterios registrados.	El botón "Next Step" se muestra únicamente cuando existen criterios registrados.	El botón "Next Step" se muestra únicamente cuando existen criterios registrados.	Si
CPCC12	Verificar que se pueda regresar	Se pueda regresar al módulo de carga	Se pueda regresar al	Si

	al módulo de carga de archivos pdf.	de archivos pdf al dar clic en el botón "Back"	módulo de carga de archivos pdf al dar clic en el botón "Back"	
CPCC13	Verificar que se pueda continuar al siguiente módulo.	Se pueda continuar al siguiente módulo dando clic sobre el botón "Next Step"	Se pueda continuar al siguiente módulo dando clic sobre el botón "Next Step"	Si

**Observaciones:** Se presento un caso donde el usuario al dar clic sobre el botón Select en la tabla de términos, no se cargaba la selecciona realizada. Para solucionar el inconveniente se le pido al usuario que vuelva a cargar la página de la aplicación, una vez se volvió a cargar la página el problema se solucionó.

### 3.1.3. Módulo de resultados

Para el módulo de resultados se definieron 6 casos de prueba y la tabla 34 muestra los resultados obtenidos y el cumplimiento de las diferentes funcionalidades.

*Tabla 34: Casos de prueba módulo de resultados.*

Código	Descripción del caso	Salida		Cumplimiento
		Esperada	Obtenida	
CPMR01	Verificar que se desplieguen cada una de las tablas resultantes.	Se despliegan cada una de las tablas resultantes al dar clic en los respectivos botones "Show".	Se despliegan cada una de las tablas resultantes al dar clic en los respectivos botones "Show".	Si
CPMR02	Verificar que se muestre le grafico tipo pastel con cada uno de los	Se muestre el grafico tipo pastel con cada uno de los criterios y su clasificación.	Se muestre el grafico tipo pastel con cada uno de los criterios y	Si



	critérios y su clasificación.		su clasificación.	
CPMR03	Verificar que se puedan exportar las tablas resultantes en un archivo Excel.	Se puedan exportar las tablas resultantes en un archivo Excel.	Se puedan exportar las tablas resultantes en un archivo Excel.	Si
CPMR04	Verificar que se despliegue el modal para guardar el proceso al dar clic en "New Process".	Se despliegue el modal para guardar el proceso al dar clic en "New Process".	Se despliegue el modal para guardar el proceso al dar clic en "New Process".	Si
CPMR05	Verificar que, en el modal para guardar el proceso se pueda ingresar el nombre en el campo "Process Name".	En el modal para guardar el proceso se pueda ingresar el nombre en el campo "Process Name".	En el modal para guardar el proceso se pueda ingresar el nombre en el campo "Process Name".	Si
CPCC06	Verificar en el modal que se guarde o modifique el proceso al dar clic en el botón "Save Changes"	En el modal se guarda o modifica el proceso al dar clic en el botón "Save Changes"	En el modal se guarda o modifica el proceso al dar clic en el botón "Save Changes"	Si

### 3.1.4. Módulo de usuario

Para el módulo de usuario se definieron 3 casos de prueba y la tabla 35 muestra los resultados obtenidos y el cumplimiento de las diferentes funcionalidades.

Tabla 35: Casos de prueba módulo de usuario.

Código	Descripción del caso	Salida		Cumplimiento
		Esperada	Obtenida	
CPMU01	Verificar que se muestren los procesos guardados por el usuario.	Se muestran los procesos guardados por el usuario.	Se muestran los procesos guardados por el usuario.	Si
CPMU02	Verificar que se puede cambiar entre procesos al dar clic en el botón "Select".	Se puede cambiar entre procesos al dar clic en el botón "Select".	Se puede cambiar entre procesos al dar clic en el botón "Select".	Si
CPMU03	Verificar que se puede eliminar un proceso al dar clic en el botón "Delete"	Se elimina un proceso al dar clic en el botón "Delete"	Se elimina un proceso al dar clic en el botón "Delete"	Si

## 3.2. Pruebas de usabilidad

Para la evaluación de usabilidad, se definió un protocolo que consta de cuatro fases:

- La primera fase consiste en definir el número de personas que serán parte del experimento para evaluar la usabilidad del sistema.
- La segunda fase conlleva la ejecución de la evaluación por parte de los participantes.
- La tercera fase consiste en presentar los resultados obtenidos en la evaluación.
- La cuarta fase comprende la discusión de los resultados obtenidos en la evaluación de usabilidad realizada.

### 3.2.1. Primera fase

En esta fase se define el número de personas encargadas de evaluar la aplicación. Para la ejecución de esta evaluación se convocó a 6 miembros de la Escuela Politécnica Nacional, entre ingenieros y tesisistas, quienes han trabajado previamente con el método de investigación automatizado en este proyecto. De acuerdo con Nielsen [17] para realizar la evaluación de un producto de software se debe contar únicamente con 5 personas para

poder identificar alrededor del 85% de todos los problemas de usabilidad que pueda presentar el producto. Debido a que la aplicación este enfocada para un único rol, el de los investigadores, no se requiere el uso de varios grupos de personas para evaluar las mismas tareas.

Al ser una aplicación web, la evaluación del sistema se realizó de forma remota.

### 3.2.2. Segunda fase

En la segunda fase se describe el protocolo que será usado por los participantes para realizar la evaluación de la aplicación. Las actividades llevadas a cabo se muestran a continuación:

- Consentimiento para efectuar el estudio y uso de información: En esta actividad se expone a los usuarios el objetivo del estudio que se va a llevar a cabo. Antes de participar en la evaluación se les solicite que llenen y firmen el documento de consentimiento. Esta actividad tiene una duración aproximada de 10.
- Presentación del funcionamiento de la aplicación: En esta actividad se da una introducción de la metodología automatizada, a su vez se realiza una exposición de las interfaces y las funcionalidades más importantes de la aplicación. Esta actividad tiene una duración aproximada de 10 minutos.
- Experimento: en esta actividad se definen las tareas específicas que van a ser evaluadas dentro de la aplicación. Los participantes son los encargados de ejecutar cada una las taras expuestas. Esta actividad tiene una duración aproximada de 60 minutos.

Para el protocolo se definieron las siguientes tareas, descritas en la tabla 36.

*Tabla 36: Tareas para la evaluación de la aplicación.*

<b>Tareas</b>	<b>Descripción</b>
Ingresar a la aplicación	Registrarse en la aplicación, completando los datos solicitados.
Seleccionar y guardar archivos pdf.	Seleccionar uno archivo pdf. Una vez seleccionado el archivo, guardar el archivo en la aplicación.
	Seleccionar varios archivos pdf. Una vez seleccionados los archivos, guardar los archivos en la aplicación.

Borrar un pdf.	Marcar el checkbox de uno o varios archivos pdf en la tabla y eliminarlos.
Modificar un pdf.	Seleccionar un archivo pdf de la tabla de archivos, modificar el nombre y guardar el archivo modificado.
Guardar un criterio con sinónimos.	Ingresar en los campos correspondientes un criterio con sus sinónimos y registrarlos en el sistema.
Borrar un criterio.	Marcar el checkbox de uno o varios criterios en la tabla y eliminarlos.
Modificar un criterio.	Seleccionar un criterio de la tabla de criterios, modificar el criterio o uno de sus sinónimos y guardar el criterio modificado.
<b>Proceso 1</b>	
Cagar la matriz booleana.	Descargar el template de la matriz booleana, llenar los campos correspondientes.
	Cargar en la aplicación el archivo modificado de la matriz booleana.
<b>Proceso 2</b>	
Seleccionar el tipo de medida numérica.	Seleccionar una opción entre las cuatro existentes para calcular la variable k.
<b>Resultados</b>	
Desplegar los resultados.	Desplegar cada una de las tablas resultantes del proceso.
Exportar resultados.	Descargar las tablas resultantes en un archivo Excel para su visualización.
Guardar el proceso.	Ingresar un nombre para el proceso y registrarlo en la aplicación.
<b>Administrativo</b>	
Ingresar al perfil del usuario.	Navegar en la barra de menú e ingresar al perfil del usuario.
Cambiar proceso.	Seleccionar un nuevo proceso entre la lista de procesos.

	Verificar que se realizó el cambio de procesos.
Borrar un proceso.	Eliminar un proceso entre los registrados en la aplicación.
Modificar y guardar el proceso actual.	Modificar un pdf o un criterio del actual proceso.
	Guardar el actual proceso modificando su nombre.
Crear un nuevo proceso.	Crear un proceso nuevo sin guardar el proceso actual.

- Responder el cuestionario: Una vez finalizadas las tareas de evaluación de la actividad anterior los participantes proceden a contestar una encuesta de usabilidad. Esta actividad tiene una duración aproximada de 10 minutos.

Para la encuesta de usabilidad se usó la propuesta por IMB en [18], conocida como Computer System Usability Questionnaire (CSUQ).

La encuesta aplicada en esta evaluación está conformada por 16 preguntas planteadas en el cuestionario CSUQ, y una pregunta adicional correspondiente a comentarios y sugerencias de los participantes. Las primeras 16 preguntas pueden ser valoradas en una escala de 1 a 7, donde 1 indica que el usuario está en total desacuerdo, y 7 indica que el usuario está totalmente de acuerdo. La tabla 37 presenta las preguntas de la encuesta.

*Tabla 37: Cuestionario para evaluar la usabilidad.*

N°	Pregunta	Escala						
		1	2	3	4	5	6	7
1	En general, estoy satisfecho con lo fácil que es usar este sistema.							
2	Es simple usar este sistema.							
3	Soy capaz de completar mi trabajo rápidamente usando este sistema.							
4	Me siento cómodo usando este sistema.							
5	Fue fácil aprender a usar este sistema.							
6	Creo que me volví productivo rápidamente usando este sistema.							

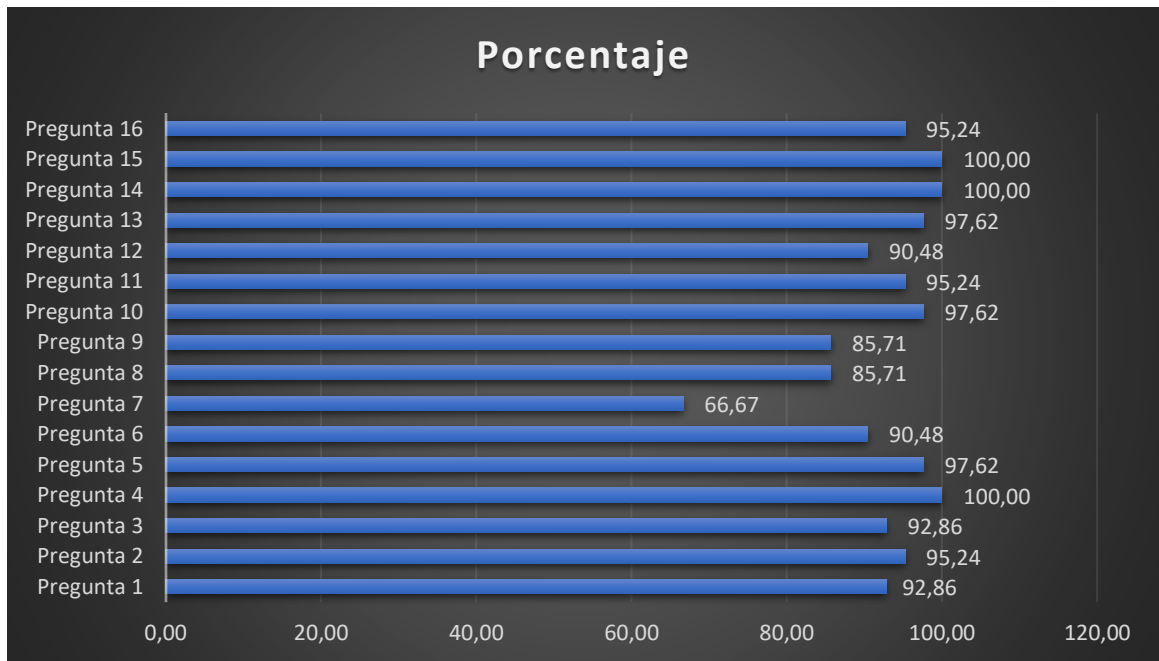
7	El sistema mostró mensajes de error que me indicaron claramente cómo solucionar problemas.							
8	Cada vez que cometía un error al utilizar el sistema, podía recuperarme fácil y rápidamente.							
9	La información (tal como ayuda en línea, mensajes en pantalla y otra documentación) proporcionada con este sistema es clara.							
10	Es fácil encontrar la información que necesitaba							
11	La información proporcionada con el sistema es eficaz para ayudarme a completar mi trabajo.							
12	La organización de la información en las pantallas del sistema es clara.							
13	La interfaz de este sistema es muy agradable.							
14	Me gustó usar la interfaz de este sistema.							
15	Este sistema tiene todas las funciones y capacidades que espero tener.							
16	En general, estoy satisfecho con este sistema							
17	Comentarios y sugerencias							

### 3.2.3. Tercera fase

Esta fase consiste en presentar los resultados obtenidos en la evaluación de usabilidad. Cabe mencionar que la encuesta fue realizada con un total de 6 participantes. La información sin procesar obtenida de la encuesta se encuentra en el Anexo 4: Respuestas encuesta de usabilidad.

#### Resultados Encuesta CSUQ

La figura 20 muestra los porcentajes promedio obtenidos en cada una de las preguntas de la encuesta CSUQ.

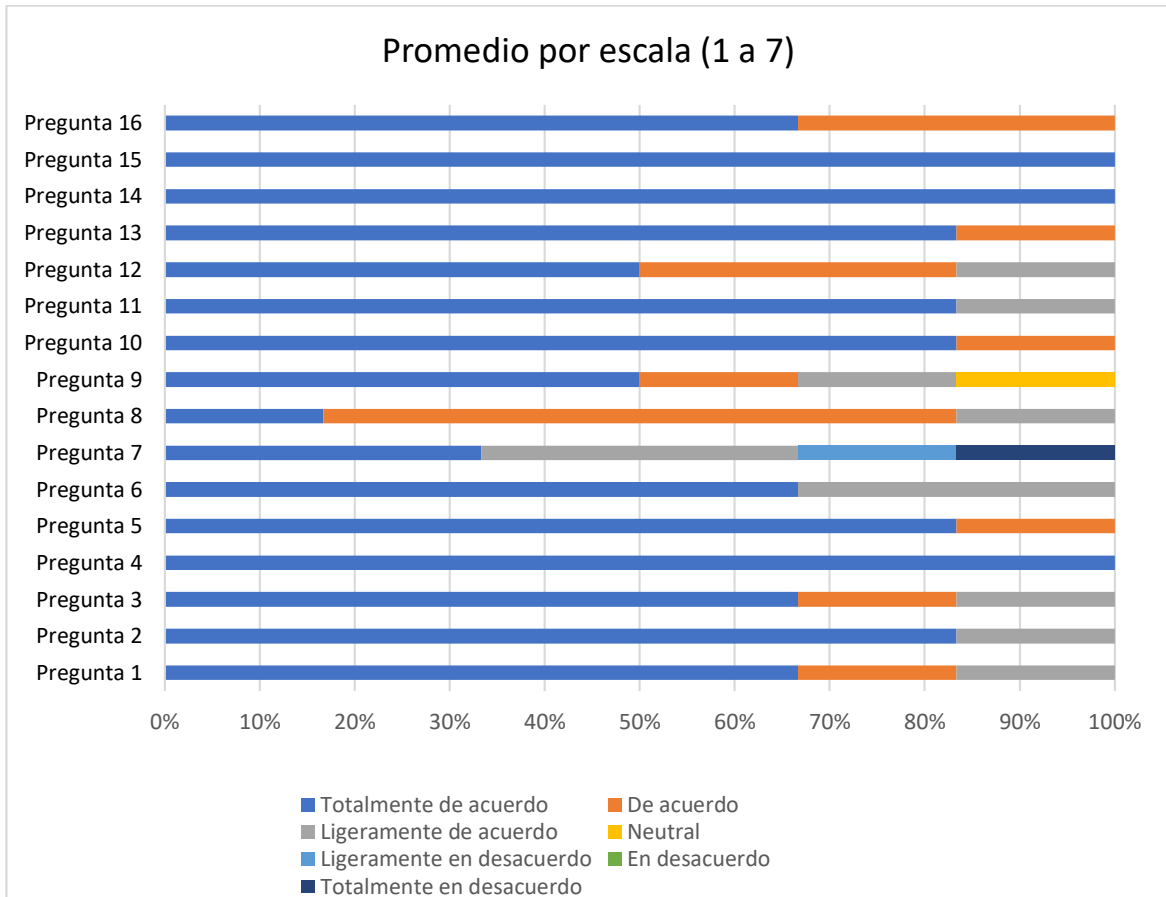


*Figura 20: Promedio porcentual por pregunta, encuesta CSUQ.*

Como se puede observar en la figura 20 existen tres preguntas con porcentaje del cien por ciento (100%) estas preguntas son: 4, 14 y 15. Mientras que la pregunta con el puntaje más bajo es la 7, con sesenta y seis por ciento (66.67%). Además, el porcentaje global de todas las preguntas planteadas en la encuesta de usabilidad es de 92.71%.

La figura 21 muestra el porcentaje de cada valor de la escala por pregunta. Así, la figura nos permite identificar que en 14 de las 16 preguntas el valor de la escala correspondiente a totalmente de acuerdo tuvo un porcentaje entre 50% y 100%. De igual forma se puede identificar que en la pregunta 7 existe un 16.7% de participantes que están totalmente en desacuerdo.

En el Anexo 5: Evidencia visual y grafica de usabilidad, se encuentra el Excel completo con la información procesada de las respuestas de los participantes por pregunta, junto con las gráficas obtenidas.



*Figura 21: Promedio por escala.*

### 3.2.4. Cuarta fase

Finalmente, en esta fase se discuten los resultados obtenidos en la evaluación de usabilidad. Para ver todos los diagramas resultantes, ver el Anexo 3: Tabulación de datos y graficas de usabilidad.

Concorde a los resultados obtenidos en la encuesta CSUQ, se concluye que la aplicación es usable ya que en las preguntas: “En general, estoy satisfecho con lo fácil que es usar este sistema” y “En general, estoy satisfecho con este sistema” mostradas en las figuras 22 y 23, tuvieron una aceptación mayor al 80%.



1. Overall, I am satisfied with how easy it is to use this system. En general, estoy satisfecho con lo fácil que es usar este sistema.

[Más detalles](#)

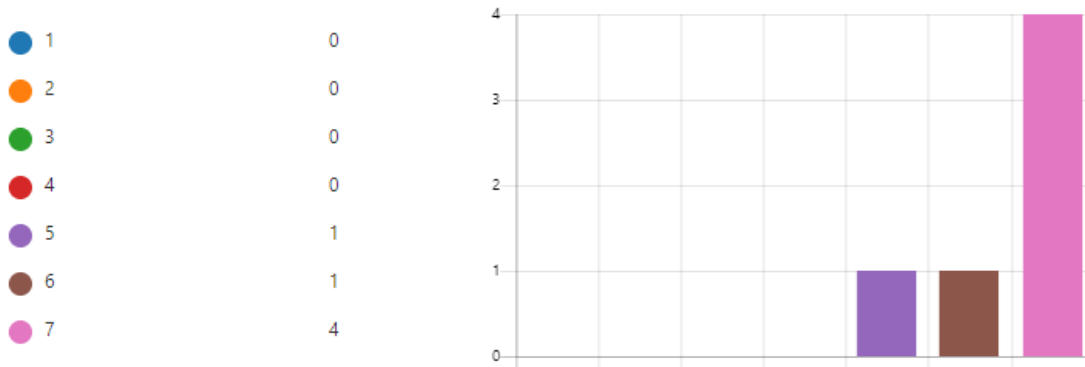


Figura 22: Resultados pregunta 1 encuesta CSUQ.

16. Overall, I am satisfied with this system. En general, estoy satisfecho con este sistema.

[Más detalles](#)

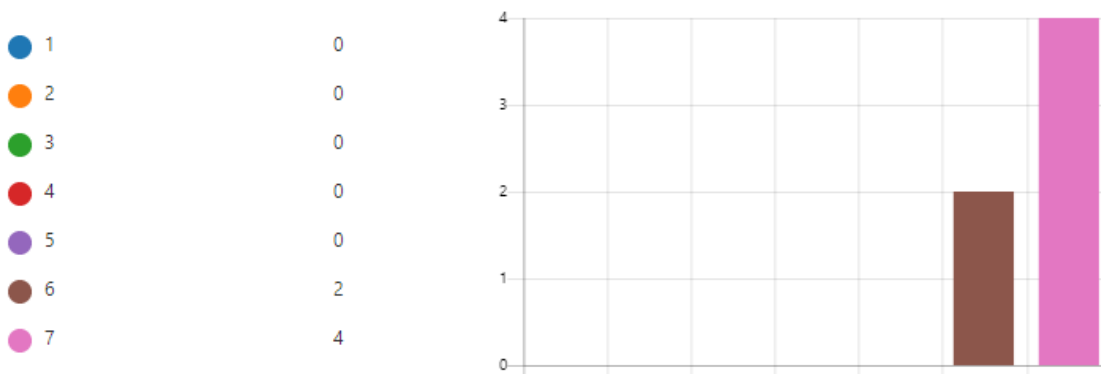


Figura 23: Resultados pregunta 16 encuesta CSUQ.

De igual forma en la pregunta 5: “Fue fácil aprender a usar este sistema”, que representa la curva de aprendizaje de la aplicación, se obtuvo un promedio de 97.62% lo que evidencia que para los participantes fue fácil aprender a usar la herramienta.

5. It was easy to learn to use this system. Fue fácil aprender a usar este sistema.

[Más detalles](#)

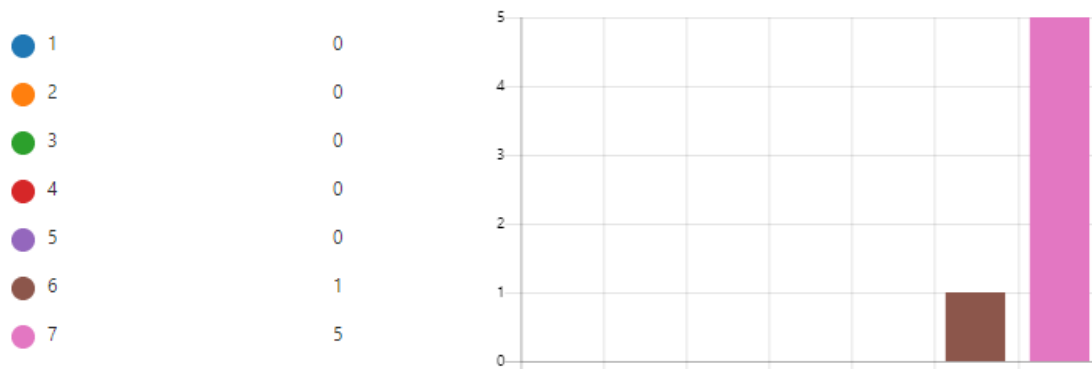


Figura 24: Resultados pregunta 5 encuesta CSUQ.

La pregunta 7: “El sistema mostró mensajes de error que me indicaron claramente cómo solucionar problemas”, presenta el porcentaje de aceptación más bajo de la encuesta (66.67%). La figura 25 muestra los resultados obtenidos en esta pregunta.

7. The system gives error messages that clearly tell me how to fix problems. El sistema mostró mensajes de error que me indicaron claramente cómo solucionar problemas.

[Más detalles](#)

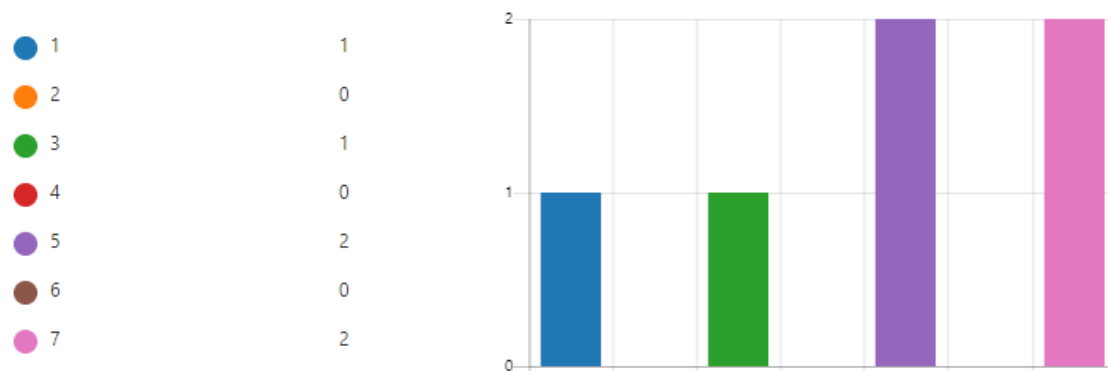


Figura 25: Resultados pregunta 7 encuesta CSUQ.

### 3.3. Resumen del capítulo

En el capítulo 3 “Evaluación y Resultados” se desarrolló la evaluación de funcionalidad y usabilidad de la aplicación desarrollada. La primera sección del capítulo presenta las pruebas de funcionalidad, aquí se presenta los casos de estudio planteados, el cumplimiento de estos y las observaciones encontradas para cada uno de los módulos de la aplicación.

La segunda sección del capítulo presenta las pruebas de usabilidad, para esta evaluación se empleó el cuestionario CSUQ propuesto por IBM. Aquí se presentó un protocolo de

cuatro fases: la primera fase consiste en definir el número de personas que serán parte de la prueba de usabilidad, se convocó a 6 miembros de la Escuela Politécnica Nacional para que participen en las pruebas, el rol planteado fue el de investigadores ya que los participantes trabajaron previamente con la metodología automatizada. La segunda fase conlleva la ejecución de la evaluación por parte de los participantes, en esta fase se entregó a los participantes los pasos que se ejecutarían en la prueba, una carta de consentimiento y un cuestionario que llenaron al finalizar la prueba. La tercera fase consiste en presentar los resultados obtenidos en la evaluación, aquí se presenta el porcentaje obtenido en cada una de las preguntas realizadas a los participantes. Finalmente la cuarta fase comprende la discusión de los resultados obtenidos en la evaluación de usabilidad realizada.

## CAPÍTULO 4

### 4.1. Conclusiones

- Se desarrolló de manera exitosa una aplicación web que automatiza el método planteado para la determinación de criterios de relevancia mediante el uso de un enfoque ágil como SCRUM. Utilizando los principios de desarrollo ágil se establecieron 10 historias de usuario (sección 2.3.4) distribuidas en 3 *sprints*, el avance logrado en cada uno de los *sprint* fue el esperado con un desarrollo de 20 días por *sprint* como se puede observar en los *burndownchart* correspondientes a cada *sprint* (Figura 4: *Burndown Chart Sprint 0*, Figura 7: *Burndown Chart Sprint 1*, Figura 10: *Burndown Chart Sprint 2*). En combinación con las historias de usuario se consiguió obtener los requerimientos necesarios para el desarrollo de la aplicación, entre los elementos más importantes están el manejo de los archivos pdf para la extracción de la información, así como la selección de las diferentes medidas numéricas para el cálculo de la variable  $K$ .

Se desarrollaron tres módulos que componen la automatización del método planteado: el módulo de carga de archivos pdf, módulo de carga de criterios y sinónimos y el módulo de resultados. Además, se desarrolló un módulo administrativo con el fin de permitirle al usuario gestionar todos los procesos efectuados en la aplicación.

- Gracias a los requerimientos obtenidos mediante la aplicación de las historias de usuario se logró desarrollar una aplicación más completa en cuanto a la automatización de la metodología presentada, ya que con esta aplicación se puede efectuar el proceso que utiliza la medida numérica TF-IDF para la obtención de la matriz booleana, así como el proceso cuya información de entrada es la matriz booleana basada en la opinión del investigador.
- El estudio de la medida numérica TF-IDF permitió entender como un término tiene más relevancia con respecto a otros en un corpus de documentos, la medida determina la frecuencia relativa de un término en un documento específico comparado con la proporción inversa de dicha palabra a través del corpus de documentos (sección 1.4.1). Mediante el estudio realizado de la medida numérica TF-IDF que permite ponderar los criterios analizados, en conjunto con las tablas de verdad y la lógica proposicional, se creó un marco de referencia que permitió un análisis a profundidad del funcionamiento de la metodología para su posterior implementación en código.

- El uso del framework web Django fue un gran acierto en el desarrollo de la aplicación, ya que permitió un desarrollo más ágil en cuanto a la seguridad e interfaces de la aplicación lo que permitió enfocarse más en la automatización del proceso.
- Para el proyecto se efectuaron dos pruebas una de funcionalidad y una de usabilidad. Las dos pruebas se llevaron a cabo con 6 miembros de la Escuela Politécnica Nacional quienes han trabajado previamente con la metodología automatizada. Para la prueba de funcionalidad se hizo el uso de casos de prueba basadas en las historias de usuario para los cuatro módulos de la aplicación, dado el resultado satisfactorio obtenidos en las pruebas de funcionalidad como se puede observar en las tablas correspondientes a los casos de prueba de cada uno de los módulos (Tabla 32: Casos de prueba módulo de carga de archivos PDF, Tabla 33: Casos de prueba módulo de carga de criterios y sinónimos, Tabla 34: Casos de prueba módulo de resultados, Tabla 35: Casos de prueba módulo de usuario), se concluyó que el producto final desarrollado en este proyecto logró satisfacer las expectativas de los usuarios, en cuanto a la automatización de los dos procesos que componen la metodología tratada.
- De la evaluación de usabilidad, basado en los resultados obtenidos en la encuesta CSUQ (sección 3.2.3), se llegó a la conclusión que la aplicación tiene un alto grado de usabilidad siendo los puntos más altos, la facilidad y la comodidad de los usuarios para utilizar el sistema, esto debido a que se obtuvo un promedio de 92.17% en el porcentaje global de todas las preguntas planteadas en la encuesta de usabilidad.
- Dentro de la encuesta de usabilidad CSUQ, la pregunta “El sistema mostró mensajes de error que me indicaron claramente cómo solucionar problemas.”, presentó el porcentaje de aceptación más bajo con 66.67%. Este resultado se debió por el error que se presentó un usuario debió al buscador que utilizó para abrir la aplicación.
- Finalmente, la aplicación se implementó de forma exitosa en ambiente de producción utilizando los servicios de AWS, en conjunto con el servidor Nginx y Unicorn. La aplicación en ambiente de producción se puede encontrar en la url [www.key-criteria.com](http://www.key-criteria.com)

## 4.2. Perspectivas

- La selección de criterios corresponde a uno de los pasos críticos dentro del método, realizar una selección adecuada de los criterios influye directamente en el cálculo

de la medida TF-IDF ya sea para obtener un resultado esperado o de forma contraria, dando la posibilidad de no obtener ningún tipo de resultado. Esta variación puede presentarse por la modificación de un único término, así como por la modificación de todo un conjunto de términos, por lo que se considera que a futuro se podría desarrollar un módulo comparativo dentro de un mismo proceso, con el fin de que el usuario pueda visualizar los cambios (en los criterios y sinónimos), y la variación en los resultados producto de dicho cambio.

- Siguiendo la línea del punto anterior la variación en los resultados obtenidos también puede ocurrir dependiendo del tipo de medida que se utilice para realizar el cálculo de la variable  $k$ , dentro del módulo comparativo propuesto en el punto anterior, se incluiría una comparación de resultados donde el cambio sea el tipo de medida seleccionada por el usuario.

### **4.3. Recomendaciones**

- Durante el proceso de desarrollo de la aplicación se llevaron a cabo cambios significativos en la interfaz de la aplicación lo que derivó en un incremento de las horas de desarrollo, se recomienda definir y dejar clara las interfaces y su funcionalidad a implementar, con el fin de evitar cambios a lo largo del desarrollo que conlleven más tiempo de programación. De igual forma es importante contar con retroalimentación por parte del producto Owner para que se pueda realizar a tiempo las correcciones pertinentes.
- Se recomienda hacer un estudio que permita determinar un promedio de documentos y criterios utilizados al llevar a cabo el método estudiado con el fin de mejorar las características tanto de procesador, memoria y almacenamiento del servidor en el caso que el uso de la aplicación aumente significativamente.
- Dado que la extracción de información de los archivos pdf se ve restringida en aquellos casos en que los archivos no poseen una codificación adecuada, no se permite su lectura o están compuestos por imágenes, se recomienda implementar OCR (Optical character reader) para la extracción de información. Ya que el principal inconveniente que presenta esta técnica radica en el consumo de memoria dentro del servidor, debido a la transformación en imágenes de todo el archivo pdf, se recurriría a esta técnica únicamente en estos casos especiales.

## REFERENCIAS BIBLIOGRÁFICAS

- [1] C. D. Manning, P. Raghavan, and H. Schütze, *An Introduction to Information Retrieval*. New York: Cambridge University Press, 2009.
- [2] J. Ramos, "Using TF-IDF to Determine Word Relevance in Document Queries," 2003.
- [3] Y. Zhang, L. Gong, and Y. Wang, "An improved TF-IDF approach for text classification," *J. Zhejiang Univ. Sci.*, vol. 6, no. 1, pp. 49–55, 2004, doi: 10.1631/jzus.2005.a0049.
- [4] S. Qaiser and R. Ali, "Text Mining: Use of TF-IDF to Examine the Relevance of Words to Documents," *Int. J. Comput. Appl.*, vol. 181, no. 1, pp. 25–29, 2018, doi: 10.5120/ijca2018917395.
- [5] J. Aspnes, *Notes on Discrete Mathematics*. 2018.
- [6] H. B. Enderton, *A Mathematical Introduction to Logic*, vol. 6, no. 2. London: Academic Press, 1972.
- [7] A. A. Barceló Aspeitia, "Introducción a la Lógica Intensional Lógica Temporal Proposicional," no. 1921, 2012, [Online]. Available: <http://www.filosoficas.unam.mx/~abarcelo/INTENSIONAL/2012/260312.pdf>.
- [8] M. Carrion, M. Santorum, J. Aguilar, and A. Pinaida, "Study to Infer Key Criteria for the Design of Serious Games," *2019 Int. Conf. Inf. Syst. Softw. Technol.*, pp. 63–70, 2019, doi: 10.1109/ici2st.2019.00016.
- [9] K. Schwaber and J. Sutherland, "La Guía de Scrum," 2017.
- [10] J. Deacon, "Model-View-Controller (MVC) Architecture," pp. 1–6, 2009.
- [11] "Modelo vista controlador (MVC)." <https://si.ua.es/es/documentacion/asp-net-mvc-3/1-dia/modelo-vista-controlador-mvc.html> (accessed Dec. 31, 2019).
- [12] D. Kuhlman, "A Python Book," pp. 1–227, 2013, [Online]. Available: <http://www.davekuhlman.org>.
- [13] M. Cohn, *User Stories Applied: For Agile Software Development*. Addison-Wesley Professional, 2004.
- [14] J. Menzinsky, Alexander, López, Gertrudis, Palacio, *Historias de Usuario*. 2018.
- [15] "How to use sessions." <https://docs.djangoproject.com/en/3.0/topics/http/sessions/> (accessed Dec. 31, 2019).
- [16] E. Serna Montoya, *Prueba funcional del software: un proceso de verificación constante*. Fondo Editorial ITM, 2013.
- [17] J. Nielsen, "Why You Only Need to Test with 5 Users." 2000.
- [18] I. Journal, H. Interaction, and J. R. L. Ibm, "Lewis , J . R . : IBM Computer Usability

Satisfaction Questionnaires: Psychometric Evaluation and Instructions for Use .  
International Journal of IBM Computer Usability Satisfaction Questionnaires :  
Psychometric Evaluation and Instructions for Use,” vol. 7, no. May, pp. 57–78, 1995.

## **ANEXOS**

- Anexo 1 – Resultados caso de estudio.
- Anexo 2 – Manual de usuario.
- Anexo 3 – Respuestas encuesta de usabilidad.
- Anexo 4 – Tabulación de datos y graficas de usabilidad.