

UNIVERSIDAD TÉCNICA DEL NORTE



Facultad de Ingeniería en Ciencias Aplicadas
Carrera de Ingeniería en Sistemas Computacionales

**DESARROLLO DE UN SISTEMA WEB PARA BÚSQUEDAS DE ALTO
RENDIMIENTO SOBRE UNA BASE DE DATOS DE NORMAS Y ESTÁNDARES
QUE SE APLICAN EN LAS ÁREAS DE INGENIERÍA MEDIANTE LA API LUCENE**

Trabajo de grado previo a la obtención del título de Ingeniero en Sistemas
Computacionales

Autor:

RICHARD FERNANDO TARUPI CALAN

Director:

MSC. MAURICIO REA

Ibarra – Ecuador

2019



UNIVERSIDAD TÉCNICA DEL NORTE

BIBLIOTECA UNIVERSITARIA

AUTORIZACIÓN DE USO Y PUBLICACIÓN A FAVOR DE LA UNIVERSIDAD TÉCNICA DEL NORTE

1. IDENTIFICACIÓN DE LA OBRA

En cumplimiento del Art. 144 de la Ley de Educación Superior, hago la entrega del presente trabajo a la Universidad Técnica del Norte para que sea publicado en el Repositorio Digital Institucional, para lo cual pongo a disposición la siguiente información:

DATOS DE CONTACTO		
CÉDULA DE IDENTIDAD:	0401734041	
APELLIDOS Y NOMBRES:	TARUPI CALAN RICHARD FERNANDO	
DIRECCIÓN:	EL ANGEL – BARRIO SAN FRANCISCO, CALLE PICHINCHA, N° 17-16	
EMAIL:	rftarupic1@utn.edu.ec	
TELÉFONO FIJO:	-----	TELÉFONO MÓVIL: 0984905902

DATOS DE LA OBRA	
TÍTULO:	DESARROLLO DE UN SISTEMA WEB PARA BÚSQUEDAS DE ALTO RENDIMIENTO SOBRE UNA BASE DE DATOS DE NORMAS Y ESTÁNDARES QUE SE APLICAN EN LAS ÁREAS DE INGENIERÍA MEDIANTE LA API LUCENE
AUTOR (ES):	TARUPI CALAN RICHARD FERNANDO
FECHA: DD/MM/AAAA	12/07/2019
SOLO PARA TRABAJOS DE GRADO	
PROGRAMA:	<input checked="" type="checkbox"/> PREGRADO <input type="checkbox"/> POSGRADO
TÍTULO POR EL QUE OPTA:	INGENIERO EN SISTEMAS COMPUTACIONALES
ASESOR /DIRECTOR:	MSc. MAURICIO REA

2. CONSTANCIAS

El autor (es) manifiesta (n) que la obra objeto de la presente autorización es original y se la desarrolló, sin violar derechos de autor de terceros, por lo tanto la obra es original y que es (son) el (los) titular (es) de los derechos patrimoniales, por lo que asume (n) la responsabilidad sobre el contenido de la misma y saldrá (n) en defensa de la Universidad en caso de reclamación por parte de terceros.

Ibarra, a los 12 días del mes de julio de 2019

EL AUTOR:



RICHARD FERNANDO TARUPI CALAN



UNIVERSIDAD TÉCNICA DEL NORTE
FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

CERTIFICACIÓN DIRECTOR DE TESIS

Por medio del presente yo MSc. Mauricio Rea, certifico que el Sr. RICHARD FERNANDO TARUPI CALAN, portador de la cédula de identidad Nro. 0401734041, ha trabajado en el desarrollo del proyecto de grado "DESARROLLO DE UN SISTEMA WEB PARA BÚSQUEDAS DE ALTO RENDIMIENTO SOBRE UNA BASE DE DATOS DE NORMAS Y ESTÁNDARES QUE SE APLICAN EN LAS ÁREAS DE INGENIERÍA MEDIANTE LA API LUCENE", previo a la obtención del título de Ingeniero en Sistemas Computacionales, lo cual ha realizado en su totalidad con responsabilidad.

Es todo cuanto puedo certificar en honor a la verdad.

Atentamente,

MSc. Mauricio Rea
DIRECTOR DE TESIS

Dedicatoria

El presente trabajo va dedicado:

A Dios, por ser mi guía y permitirme culminar esta importante etapa de mi vida profesional, que con su amor y bondad ha hecho de mí una persona fuerte y decidida siempre dispuesto a entregar lo mejor de mí mismo.

A mi madre **María Calán**, abuelito **Jorge Calán** y a toda mi familia con mucho cariño y gratitud por ser los pilares fundamentales para el cumplimiento de mis metas, por haber depositado su confianza en mí y por todo el esfuerzo y apoyo incondicional que han tenido conmigo durante mi formación profesional.

Richard Tarupí

Agradecimientos

A Dios, por su infinito amor y sabiduría que me ha dado todo para salir adelante.

A mi madre, María Calán, por su apoyo incondicional, por sus esfuerzos y entrega hacia mí, por ese cariño y sus consejos que me han ayudado en cada etapa de mi vida, tanto personal como profesional.

A mi abuelito, Jorge Calán, por sus valores y consejos que han hecho de mí una persona crítica y constructivista, una persona que no le teme al fracaso y lucha fervientemente por sus sueños y metas.

A toda mi familia, por depositar en mí su confianza y anhelos, por ser personas que me han dado siempre las fuerzas necesarias para seguir adelante y nunca rendirme ante ninguna dificultad.

A mi novia, Helen Ulloa, por su apoyo incondicional y su constante preocupación, quien ha sabido darme ánimos y la confianza necesaria para dar cumplimiento a las metas que me proponga.

A todos mis amigos, por ser una parte fundamental en esta etapa de formación profesional, por sus consejos y su amistad, que incondicionalmente han estado presentes en los buenos y malos momentos.

Finalmente, Mi más profundo agradecimiento a mi director de tesis, MSc. Mauricio Rea, por sus enseñanzas y dirección durante este proceso, quien me ha brindado su apoyo y gran dedicación en cada etapa del presente proyecto de titulación.

Richard Tarupí

Tabla de Contenido

INTRODUCCIÓN	1
Identificación del problema	1
Antecedentes	1
Situación Actual.....	1
Prospectiva	2
Planteamiento del Problema.....	2
Objetivos.....	3
Alcance.....	3
Justificación	5
CAPÍTULO 1 Marco Teórico.....	6
1.1. RECUPERACIÓN DE INFORMACIÓN (RI)	6
1.1.1. Introducción.....	6
1.1.2. Importancia de un Sistema de Recuperación de Información.....	7
1.1.3. Recuperación de Información y Recuperación de Datos.....	8
1.1.4. Fundamentos de la RI.....	9
1.2. MODELOS CLÁSICOS DE RECUPERACIÓN DE INFORMACIÓN.....	10
1.2.1. Introducción.....	10
1.2.2. Procesamiento de Documentos	10
1.2.3. Modelos de Recuperación de Información.....	12
1.2.4. Comparativa de los modelos	18
1.3. APLICACIONES JAVA ENTERPRISE EDITION (JEE)	18
1.3.1. Introducción.....	18
1.3.2. Servidor Web y Servidor de Aplicaciones	19
1.3.3. Estructura de las aplicaciones JEE.....	20
1.3.4. Java frente a otros lenguajes de programación	20
1.4. EVOLUCIÓN DE LOS SISTEMAS WEB CON RESPECTO A LA BÚSQUEDA DE INFORMACIÓN	21
1.5. API LUCENE.....	22
1.5.1. Introducción.....	22
1.5.2. Características Principales.....	23
1.5.3. Fases Principales de Lucene.....	23
1.5.4. Clases principales para la indización.....	24
1.5.5. Clases principales para la búsqueda.....	25
1.5.6. Proceso de indización y búsqueda con Lucene	25
1.5.7. Popularidad de Lucene frente a otras APIs para RI	26

1.6.	METODOLOGIAS PARA EL DESARROLLO DE SOFTWARE.....	26
1.6.1.	El Manifiesto Ágil	26
1.6.2.	Principios de Manifiesto Ágil	27
1.6.3.	Agilismo en desarrollo de software	28
1.6.4.	Comparativa entre metodologías tradicionales y metodologías ágiles	28
1.6.5.	SCRUM como marco de trabajo ágil	29
1.7.	NORMAS Y ESTÁNDARES APLICADOS EN LA INGENIERÍA DE SOFTWARE..	30
1.7.1.	Introducción	30
1.7.2.	Asociación de Maquinaria Computacional (ACM).....	31
1.7.3.	Instituto de Ingeniería de Software (SEI)	31
1.7.4.	Organización Internacional de Normalización (ISO).....	31
1.7.5.	Comisión Electrotécnica Internacional (IEC)	32
1.7.6.	Instituto de Ingenieros Eléctricos y Electrónicos (IEEE)	32
1.8.	DEFINICIÓN DE LA NORMA ISO/IEC 25010.....	33
1.8.1.	Introducción.....	33
CAPÍTULO 2 Desarrollo		35
2.1.	Definición del Proceso de Búsqueda de normas	35
2.2.	Definición de indicadores de calidad considerando la subcaracterística de Comportamiento Temporal dentro de la característica de Eficiencia de Desempeño de la norma ISO 25010.....	37
2.3.	Metodología de desarrollo.....	38
2.4.	Roles SCRUM.....	38
2.5.	Artefactos SCRUM.....	39
2.5.1.	Matriz de planificación	39
2.5.1.1.	Cartillas de Historias de Usuario	45
2.5.2.	Casos de Uso.....	52
2.5.3.	Diagrama Conceptual	54
2.5.4.	Arquitectura del Software	55
2.5.5.	Wireframe	56
2.6.	Pruebas de funcionamiento	58
2.6.1.	Plan de Pruebas	58
2.6.2.	Informe de Plan de Pruebas	62
2.7.	Implementación de la solución	63
CAPÍTULO 3 Validación		64
3.1.	Análisis e Interpretación de resultados.....	64
3.2.	Análisis de Impactos	72

CONCLUSIONES	73
RECOMENDACIONES.....	74
GLOSARIO DE TÉRMINOS.....	75
BIBLIOGRAFIA.....	77
ANEXOS	81
Anexo A: Levantamiento de Proceso de Búsqueda de Normas.....	81
Anexo B: Wireframes SBNI.....	87

Índice de Figuras

Fig. 1. Diagrama de ishikawa	2
Fig. 2. Arquitectura monolítica multicapa java EE	3
Fig. 3. Proceso SCRUM	4
Fig. 4. Proceso general de RI	9
Fig. 5. Procesamiento de documentos para RI	11
Fig. 6. Comparativa de popularidad 2018 - 2019 de lenguajes de programación a nivel mundial.....	20
Fig. 7. Comparativa de popularidad 2018 de lenguajes de programación en Ecuador	21
Fig. 8. Evolución de los sistemas web con respecto a la RI	21
Fig. 9. Uso de Lucene en una aplicación	22
Fig. 10. Clases usadas para la indización de documentos con Lucene.....	24
Fig. 11. Proceso de indización y búsqueda con Lucene.	25
Fig. 12. Comparativa de popularidad 2018 - 2019 de Lucene frente a otras APIs de RI a nivel mundial	26
Fig. 13. Modelo De Calidad Del Producto ISO/IEC 25010	34
Fig. 14. Diagrama BPMN del Proceso de Búsqueda de normas (con finalidad administrativa)	35
Fig. 15. Diagrama BPMN del Proceso de Búsqueda de normas (con finalidad de búsqueda)	36
Fig. 16. Estructura General de la norma ISO/IEC 25010:2011	37
Fig. 17. Caso de Uso 1 – Acceso al Sistema.....	52
Fig. 18. Caso de Uso 2 – Gestión de usuarios	52
Fig. 19. Caso de Uso 3 – Gestión de Normas	53
Fig. 20. Caso de Uso 4 – Búsqueda de normas	53
Fig. 21. Diagrama conceptual del sistema	54
Fig. 22. Diagrama de la arquitectura de software	55
Fig. 23. Árbol de páginas (árbol web) del sistema	56
Fig. 24. Wireframe página index.....	56
Fig. 25. Wireframe página que muestra información detallada de una norma específica	57
Fig. 26. Wireframe de la página de resultados de búsqueda	57
Fig. 27. Resultados generales de búsquedas con Lucene.....	67
Fig. 28. Resultados generales de búsquedas con JPQL en tags del sistema	70
Fig. 29. Comparativa de los buscadores con Lucene y TAGS	71

Índice de Cuadros

TABLA 1.1	Comparativa entre RD y RI	8
TABLA 1.2	Tabla básica de fichero inverso del Modelo Booleano.....	13
TABLA 1.3	Ejemplo de Matriz de términos y documentos en el Espacio Vectorial..	15
TABLA 1.4	Matriz de términos y documentos en el Espacio Vectorial con pesos...	16
TABLA 1.5	Comparativa entre Modelos clásicos de Recuperación de Información	18
TABLA 1.6	Comparativa entre metodologías tradicionales y metodologías ágiles..	28
TABLA 2.1	Objetivo 1 de pruebas funcionales del módulo administrativo.....	58
TABLA 2.2	Objetivo 2 de pruebas funcionales del módulo administrativo.....	58
TABLA 2.3	Objetivo 1 de pruebas funcionales del módulo operativo.....	59
TABLA 2.4	Objetivo 2 de pruebas funcionales del módulo operativo.....	59
TABLA 2.5	Objetivo 3 de pruebas funcionales del módulo operativo.....	59
TABLA 2.6	Objetivo 1 de pruebas funcionales del módulo de búsquedas.....	60
TABLA 2.7	Objetivo 2 de pruebas funcionales del módulo de búsquedas.....	60
TABLA 2.8	Objetivo 3 de pruebas funcionales del módulo de búsquedas.....	60
TABLA 2.9	Objetivo 1 de pruebas de seguridad.....	61
TABLA 2.10	Objetivo 2 de pruebas de seguridad.....	61
TABLA 2.11	Objetivo 2 de pruebas de seguridad.....	61
TABLA 2.12	Recurso humano del plan de pruebas.....	62
TABLA 2.13	Resultado de las pruebas del sistema.....	62
TABLA 2.14	Resultado de las pruebas de interfaz de usuario.....	62
TABLA 2.15	Resultado de las pruebas de seguridad.....	63
TABLA 3.1	Tiempos de respuesta y resultados en búsquedas de una sola palabra con Lucene.....	65
TABLA 3.2	Tiempos de respuesta y resultados en búsquedas de dos palabras con Lucene.....	65
TABLA 3.3	Tiempos de respuesta y resultados en búsquedas de tres palabras con Lucene.....	66
TABLA 3.4	Media de tiempos de respuesta con desviación estándar en búsquedas con Lucene.....	66
TABLA 3.5	Cantidad de resultados en búsquedas con Lucene.....	67
TABLA 3.6	Tiempos de respuesta y resultados en búsquedas de una sola palabra con JPQL en tags del sistema.....	68
TABLA 3.7	Tiempos de respuesta y resultados en búsquedas de dos palabras con JPQL en tags del sistema.....	68
TABLA 3.8	Tiempos de respuesta y resultados en búsquedas de tres palabras con JPQL en tags del sistema.....	69
TABLA 3.9	Media de tiempos de respuesta con desviación estándar en búsquedas con Lucene.....	69
TABLA 3.10	Cantidad de resultados en búsquedas con JPQL en tags del sistema..	70

Resumen

El presente proyecto proporciona una optimización en el proceso de búsqueda de información de normas o estándares que tanto docentes como estudiantes de las distintas áreas de ingenierías necesitan para cumplir con los requerimientos de sus proyectos de investigación.

La optimización de este proceso se realizó mediante un Sistema de Recuperación de Información, desarrollado con tecnología Java Enterprise Edition (JEE) y la API de Recuperación de Información Lucene, bajo la metodología de desarrollo SCRUM.

El Sistema de Búsqueda de Normas para Ingenierías incorpora características de un motor de búsquedas, tales como, indexación, procesamiento de información y búsquedas de alto rendimiento, mismas que son gestionadas por la API Lucene. Además, incorpora distintos métodos de búsqueda, como las búsquedas en el índice de Lucene, búsquedas por palabras claves y mediante la visualización de las familias de normas utilizando una estructura gráfica jerárquica.

La información de las normas o estándares que se utiliza en el presente proyecto se almacenan en una base de datos PostgreSQL, esta información paralelamente se almacena en archivos de índice gestionados por la API Lucene para su motor de búsquedas.

Palabras claves:

Optimización de Proceso de Búsquedas, API Lucene, Java Enterprise Edition, SCRUM, Normas y Estándares para Ingenierías.

Abstract

The present project provides an optimization in the process of searching for information of standards that both teachers and students of the different areas of engineering need to comply with the requirements of their research projects.

The optimization of this process was performed using an Information Retrieval System, developed with Java Enterprise Edition (JEE) technology and Information Retrieval API Lucene, under the SCRUM development methodology.

The Engineering Standards Search System incorporates features of a search engine, such as indexing, information processing and high-performance searches, which are managed by the Lucene API. In addition, it incorporates different search methods, such as searches in the Lucene index, searches for keywords and by visualization of families of standards using a hierarchical graphic structure.

The information of the standards used in the present project are stored in a PostgreSQL database, this information is stored in parallel in index files managed by the Lucene API for its search engine.

Keywords:

Search Process Optimization, Lucene API, Java Enterprise Edition, SCRUM, Standards for Engineering.

INTRODUCCIÓN

Identificación del problema

Antecedentes

Los estándares y normas han sido una herramienta importante a la hora de emprender un proyecto de calidad, sin embargo varios años atrás la búsqueda de información referente a estos estándares o normas no ha sido algo sencillo, ya que no ha existido una herramienta dedicada a este proceso y dichas búsquedas no han sido optimizadas para facilidad del usuario, dificultando de esta manera obtener información importante de normativas o estándares que se acoplen a las necesidades de los proyectos que tanto docentes como estudiantes desarrollan.

Actualmente, estas búsquedas se las realiza con un motor de búsqueda como Google, pero se ha encontrado la necesidad de una herramienta dedicada exclusivamente a normas y estándares referentes a ingeniería que nos proporcione una búsqueda de alto desempeño, sofisticada y flexible.

Estas búsquedas pueden ser optimizadas con la API de Lucene la cuál según The Apache Software Foundation (s. f.) afirma: “es un software de **recuperación de información**¹ (IR) de código abierto. Es una API flexible que permite añadir capacidades de **indexación**² y búsqueda a cualquier sistema que se esté desarrollando”.

Situación Actual

Se logra determinar que los docentes y estudiantes de ingenierías de la Universidad Técnica del Norte, no cuentan con una herramienta enfocada en la búsqueda de normas y estándares, que les permita encontrar información específica de cada norma, mediante una búsqueda de alto desempeño, flexible y amigable.

Tanto docentes como estudiantes realizan sus búsquedas de normas o estándares usando motores de búsqueda como Google, los cuales no son herramientas enfocadas a estas búsquedas por lo que no resulta óptimo.

¹ Recuperación de Información o RI hace referencia al área que consiste en aplicar un conjunto de métodos y técnicas para buscar y recuperar información de manera eficiente y que sea relevante para las necesidades de información.

² Indexar o Indizar se le conoce al proceso de ordenamiento de una serie de datos o información de acuerdo con un criterio en común a todos los datos, con la finalidad de facilitar la consulta y análisis de estos.

Prospectiva

El presente proyecto permitirá colaborar a los docentes y estudiantes de ingenierías en sus diferentes proyectos, mediante una herramienta que tenga como fin proporcionar información específica de las diferentes normas o estándares que se adapten a sus respectivos proyectos, mediante una búsqueda de alto desempeño.

Este sistema será indispensable a la hora de empezar un proyecto de calidad ya que, mediante este, los usuarios podrán obtener información relevante de una manera rápida.

Planteamiento del Problema

El proceso de selección de normas o estándares para su aplicación en proyectos que tanto docentes como estudiantes de ingeniería han desarrollado durante los últimos años, no ha sido satisfactoria debido a la falta de una herramienta enfocada en una búsqueda de alto desempeño que permita disponer de información importante de las diferentes normas o estándares que podrían ser aplicables a sus proyectos.

La Fig. 1 detalla las causas y efectos del problema encontrado en el presente trabajo de investigación.

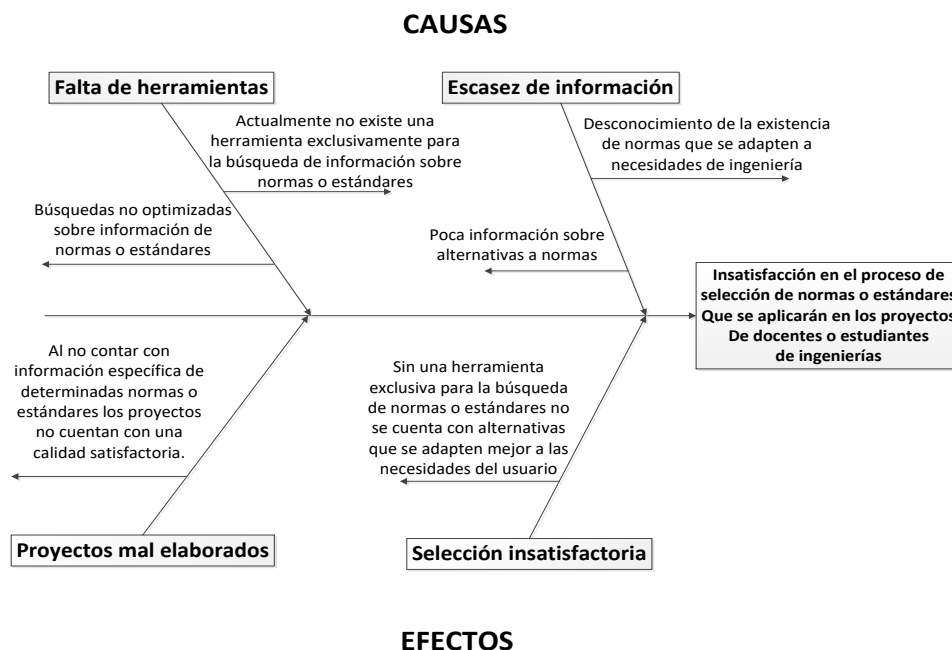


Fig. 1. Diagrama de ishikawa
Fuente: Propia

Objetivos

Objetivo General

Desarrollar un sistema web que incluya un motor de búsqueda de alto desempeño utilizando los métodos de indexación de información textual del API Lucene, sobre una base de datos de normas y estándares que se aplican en las áreas de ingeniería.

Objetivos Específicos

1. Establecer un marco de referencia sobre métodos de análisis y Recuperación de Información (RI) para la indexación y búsqueda de texto completo.
2. Desarrollar e implementar un sistema web que integre las capacidades de motor de búsqueda del API de Recuperación de Información Lucene para procesar una base de datos de normas y estándares enfocados a Ingeniería.
3. Validar los resultados

Alcance

El proyecto tiene como finalidad el desarrollo de un sistema web en un entorno de Java Enterprise con una base de datos en PostgreSQL y el uso de la API de recuperación de Información Lucene, que permita una búsqueda de texto completo sobre una base de datos de información acerca de normas y estándares enfocados a ingenierías.

La arquitectura usada en el proyecto es la arquitectura monolítica multicapa para Java Enterprise, la cual se muestra a continuación.

La Fig. 2 presenta la arquitectura utilizada en el presente proyecto.

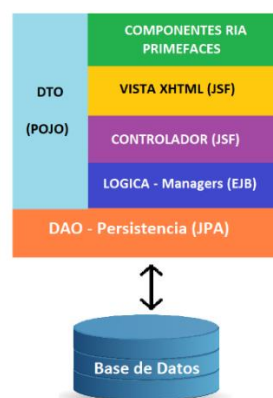


Fig. 2. Arquitectura monolítica multicapa java EE
Fuente: Propia

“De forma muy resumida, puede decirse que la arquitectura monolítica es aquella en la que el software se estructura de forma que todos los aspectos funcionales del mismo quedan acoplados y sujetos en un mismo programa” (VIEWNEXT S.A, 2018).

“Un sistema orientado a objetos multicapa está organizados en varias capas, cada una de las cuales contiene un conjunto de clases con responsabilidades relacionadas con la capa a la que pertenecen” (Usaola, 2017).

Además, se usará la metodología de desarrollo ágil **SCRUM**, la cual será de gran beneficio para que las actividades planificadas se desarrollen con efectividad.

La metodología SCRUM según Francia J. (2017) afirma: “es un proceso de gestión que reduce la complejidad en el desarrollo de productos para satisfacer las necesidades de los clientes”.

La Fig. 3 muestra el ciclo de vida o mejor conocido como el proceso de la metodología de desarrollo SCRUM.

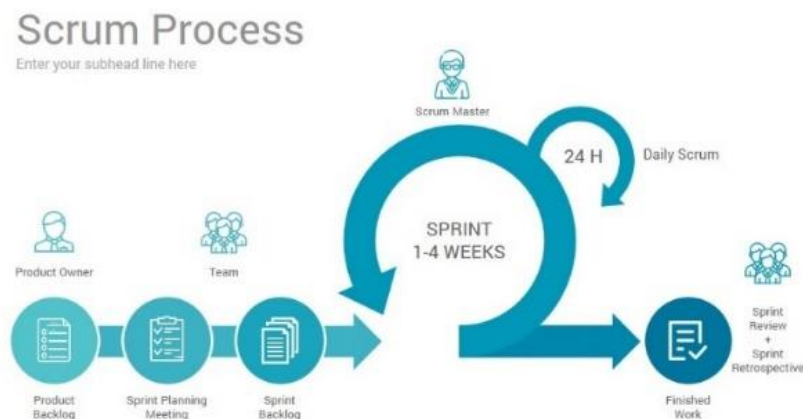


Fig. 3. Proceso SCRUM
Fuente: (Frechina, 2018)

Luego de un estudio se determinará el método estadístico que permita validar los resultados

Justificación

Político

El presente proyecto aporta para la mejora de la Calidad en la Educación lo cual a su vez apoya al Objetivo de Desarrollo Sostenible (ODS) 4: “EDUCACIÓN DE CALIDAD”, que, según la **CEPAL**³, N. (2015) busca: “Garantizar una educación inclusiva y equitativa de calidad y promover oportunidades de aprendizaje permanente para todos”.

Social

Este sistema web es una herramienta para búsquedas de alto desempeño en una base de datos de normas y estándares aplicables a ingenierías, lo que permitirá reducir el tiempo que tanto docentes como estudiantes de ingeniería emplean en la búsqueda de información de normas o estándares que serán implementadas en sus respectivos proyectos.

Tecnológico

Esta herramienta no solo permitirá a mencionados usuarios optimizar su tiempo y acelerar este proceso de búsqueda, sino también le permitirá tener a disposición la información más relevante de cada norma o familia de normas, mediante una búsqueda de texto completo, lo que mejora los resultados de sus búsquedas y por consiguiente tener la capacidad de seleccionar los estándares que se acoplen mejor a sus proyectos, obteniendo mejores resultados.

³ Una de las cinco comisiones regionales de las Naciones Unidas

CAPÍTULO 1

Marco Teórico

Son evidentes los cambios que han surgido en el desarrollo de los sistemas web con el pasar de los años, y el incremento masivo de información que éstas han generado a través de la Internet. Por lo que se ha presenciado una gran evolución en la manera de acceder a esta información y en la manera de optimización de estas búsquedas.

Las herramientas que faciliten u optimicen las búsquedas se han convertido en una parte esencial de todos los sistemas que trabajen con grandes cantidades de información y actualmente existe infinidad de estas herramientas que a la vez utilizan modelos tanto matemáticos como estadísticos, con la finalidad de tener una mayor satisfacción en los resultados a las necesidades de los usuarios.

En el presente capítulo se define teóricamente el proceso de búsquedas con la finalidad de conocer alternativas a búsquedas básicas que muchos de los sistemas web siguen utilizando, además de la definición teórica de las tecnologías usadas para el desarrollo del presente proyecto y la evolución que dichas tecnologías han tenido con respecto al proceso de búsquedas dentro de los sistemas web.

1.1. RECUPERACIÓN DE INFORMACIÓN (RI)

1.1.1. Introducción

El término recuperar hace referencia al hecho de volver a tener, por lo que recuperar información significaría volver a tener información que en un determinado tiempo hubo, es decir hace unos minutos o incluso años.

La Recuperación de Información (también conocida como RI o IR de su significado en inglés de *Information Retrieval*) RI de aquí en adelante, es la disciplina que tiene como propósito el estudio de la representación, la organización y el acceso eficiente a la información que se encuentra registrada en documentos, los cuales pueden ser texto o incluso archivos digitales. (Alonso & Vilares, s. f.).

El objetivo fundamental de la RI es, dada una necesidad de información y un conjunto de documentos, obtener los documentos relevantes para esa necesidad, ordenarlos en función del grado de relevancia, y presentarlos al usuario. Podemos definir relevancia como la medida de cómo un documento se ajusta a una consulta. (Madrid Gorelov, Zazo, Figuerola, & Berrocal, 2007, p. 2).

La RI nace de la problemática de sobrecarga de información, debido al gran volumen de información existente y la necesidad de los usuarios de obtener resultados de manera eficiente. (Bordignon & Tolosa Chacón, 2007).

Bordignon & Tolosa Chacón (2007) afirman: “La Recuperación de Información (RI) no es un área nueva, sino que se viene desarrollando desde finales de la década de 1950. Sin embargo, en la actualidad adquiere un rol más importante, debido al valor de la información” (p. 14).

La recuperación de información en el tiempo justo y la forma adecuada puede resultar en el éxito o fracaso de una operación, por lo que los sistemas de RI “son la pieza clave para resolver el problema de la búsqueda rápida y eficiente de información” (Madrid Gorelov et al., 2007).

1.1.2. Importancia de un Sistema de Recuperación de Información

Un punto fundamental en el área de la RI son los sistemas de RI (SRI como algunos autores lo mencionan, y como se utilizará indistintamente en el presente documento), quienes se encargan del correcto manejo de la información. Rodríguez Leyva, Viltres Sala, & Pons Flores (2016) afirman:

Los sistemas de recuperación de información o buscadores, son una fuente de acceso a la información que se encuentra distribuida en el mundo de la web, así como los servicios que en esta se brindan, su importancia y necesidad en el mundo de la navegación por internet está determinada debido a que son principalmente rastreadores de información que luego es almacenada y posteriormente accesible a través de interfaces y funcionalidades de búsqueda y ordenada según algoritmos matemáticas empleados para calcular la relevancia de los resultados. (p. 150).

A pesar de que los SRI se encargan del manejo correcto de la información, la función principal de estos no es la de entregar al usuario la información que este desee, sino determinar e indicar los documentos o resultados que son potencialmente relevantes según la necesidad de la búsqueda, estos resultados son ordenados mediante el uso de algoritmos matemáticos proporcionados por los distintos modelos de RI.

En la actualidad se han desarrollado diferentes sistemas de RI, los cuales han sido puestos al beneficio de miles de usuarios en el mundo entero a través de Internet. Entre los más importantes están los de los motores de búsqueda de internet como Google, Yahoo, etc.

1.1.3. Recuperación de Información y Recuperación de Datos

Al tratar sobre RI frecuentemente se suele confundir con la Recuperación de Datos (RD de aquí en adelante) al ser áreas que tienen un fin parecido. La diferencia entre estas áreas es por el hecho de utilizarse “para distinguir entre sistemas de RI y sistemas de gestión de bases de datos (SGBD)” (Madrid Gorelov et al., 2007).

Bordignon & Tolosa Chacón (2007) refieren lo siguiente, En la RD los objetos tratados contienen una estructura y representación definida, es decir, contienen una sintaxis y semántica no ambigua. Podemos tomar como ejemplo una tabla de clientes en una base de datos, esta tabla contiene diferentes columnas las cuales representan a los atributos de un cliente y cada fila representa a un cliente, con este ejemplo podemos notar que cada atributo de un cliente tiene su dominio ya definido y su semántica establecida. Al contrario, en la RI los objetos tratados son documentos de texto (por lo general) sin una estructura establecida.

En recuperación de datos se emplean preguntas altamente formalizadas, cuya respuesta es directamente la información deseada. En cambio, en recuperación de información las preguntas resultan difíciles de trasladar a un lenguaje normalizado (aunque existen lenguajes para la recuperación de información, son de naturaleza mucho menos formal que los empleados en los sistemas de bases de datos relacionales, por ejemplo) y la respuesta será un conjunto de documentos que probablemente contendrá lo deseado, con un evidente factor de indeterminación. (Martinez-Mendez, 2004, p. 2).

En la TABLA 1.1 se muestra una comparativa de varios aspectos entre la recuperación de datos y la recuperación de información.

TABLA 1.1
Comparativa entre RD y RI

	Recuperación de Datos	Recuperación de Información
Estructura	Información estructurada con semántica bien definida	Información semi o no estructurada
Resultados	Aciertos exactos	Aciertos parciales, lo mejor
Lenguaje de consulta	Fuertemente estructurado	Estructurado o natural
Especificación de consulta	Precisa	Imprecisa

Fuente: (Tolosa & Bordignon, s. f.)

Para la recuperación de datos es estrictamente necesario que se sepa lo que se busca ya que en este proceso se usa el operador LIKE como fuente de recuperación. Para una mejor comprensión se menciona un pequeño ejemplo comparando la RI y la RD:

Mediante una búsqueda se quiere obtener resultados relacionados a la “Calidad de Software”, por lo que en la recuperación de datos se tendría que estructura una consulta similar a la siguiente:

```
Select * from tbl_normas where norm_descr LIKE '%Calidad de Software%'
```

En la consulta anterior se deduce que los resultados obtenidos necesariamente deben tener la frase “Calidad de Software”, por lo que restringiría resultados como: “Calidad en el Software”, “Software de Calidad”, “Calidad para el Software”, etc. Mediante este pequeño ejemplo es fácil darse cuenta de que este tipo de consultas no satisfacen al usuario completamente y ante esto es necesario realizar una búsqueda más compleja la cual se realiza mediante la RI.

1.1.4. Fundamentos de la RI

La RI básicamente es un proceso en el que se puede distinguir tres elementos, los cuales según Cordina (1995) refiere son los siguientes:

- Una colección de ítems de información: estos pueden ser documentos registrados en una fuente de información, por ejemplo, una base de datos.
- Las consultas o preguntas de los usuarios: estas traducen las necesidades de información de los usuarios.
- La función de comparación entre los documentos y las consultas: el resultado de esta comparación son documentos relevantes que son presentados al usuario a fin de satisfacer sus necesidades.

En la Fig. 4. Se muestra el proceso general de la función que cumple la RI.

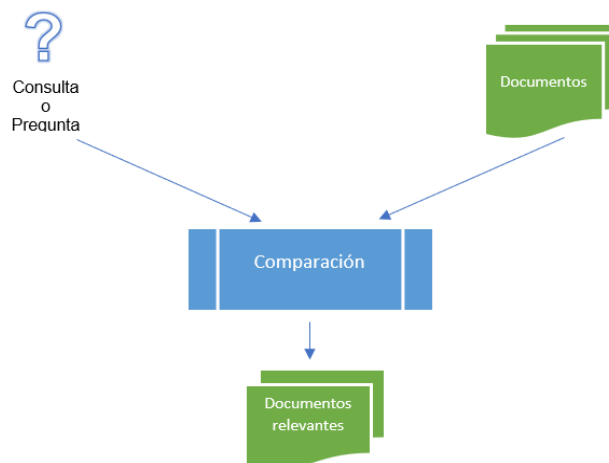


Fig. 4. Proceso general de RI
Autor: Propia
Fuente: (Cordina, 1995)

Se puede observar el proceso general de RI, en el que se destaca la importancia de una correcta comparación que debe existir para obtener mejores resultados, esta comparación se la realiza con un modelo de RI, los cuales se mencionan en el siguiente punto.

1.2. MODELOS CLÁSICOS DE RECUPERACIÓN DE INFORMACIÓN

1.2.1. Introducción

García Broncano (s. f.) refiere que, durante los últimos años la RI ha tomado gran relevancia y es uno de los aspectos que más importancia tiene en Internet, ya que la gran cantidad de información existente hace necesario herramientas que permitan al usuario encontrar información de una manera eficiente, y para ello el primer método usado para obtener cualquier tipo de información son los buscadores, al ser los encargados de realizar búsquedas avanzadas y de alto rendimiento.

La necesidad de las búsquedas de alto rendimiento para recuperar información relevante y de la misma manera rechazar lo irrelevante, hace necesario el estudio de varios modelos que permitan al usuario obtener los mejores resultados.

En la presente investigación se profundizará sobre los modelos clásicos de RI: modelo Booleano, modelo Probabilístico y modelo Vectorial o Espacio Vectorial. (García Broncano, s. f.). Es importante mencionar que para que los modelos de RI tengan éxito es necesario que los documentos tengan un procesamiento adecuado antes de su aplicación.

1.2.2. Procesamiento de Documentos

El procesamiento de documentos es una etapa fundamental para el correcto funcionamiento de los modelos de RI.

Se realiza preprocesamiento sobre las preguntas (consultas) y sobre los documentos (posibles ítems que satisfacen una consulta). Las preguntas pueden ser vistas como una expresión que deben satisfacer los documentos a recuperar o como otro documento que es cercano semánticamente al conjunto de documentos a recuperar. (Jaimes & Riveros, 2005).

En la Fig. 5 se muestra el procesamiento de documentos para RI con sus respectivas fases, las cuales son de gran importancia para una correcta recuperación de información.

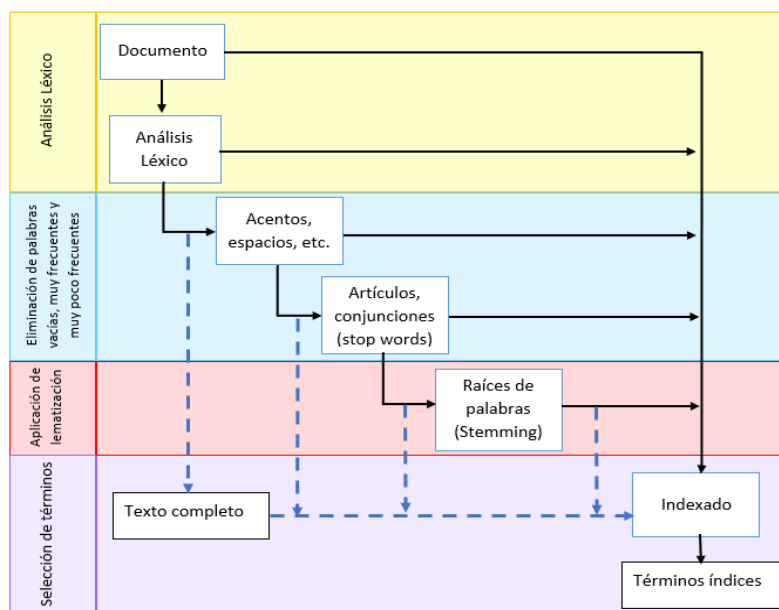


Fig. 5. Procesamiento de documentos para RI
 Autor: Propia
 Fuente: (ICT - Udlap, s. f.)

Madrid Gorelov et al. (2007) refiere que este proceso suele llevar la sucesión de los siguientes pasos:

- **Análisis léxico del texto:** Conversión texto de entrada en un conjunto de palabras.
- **Eliminación de palabras vacías, muy frecuentes y muy poco frecuentes:** Los términos que se eliminen deben ser los poco significativos como términos demasiado frecuentes o que son muy poco frecuentes.
- **Aplicación de lematización⁴ (stemming):** Consiste en encontrar un lema (stem) de los términos y no necesariamente debe tener significado. Tiene como finalidad eliminar variaciones morfo-sintácticas y obtener las raíces de los términos, sin prefijos ni sufijos. Por ejemplo, “Amigo” es el lema de amigos, amigas y amiguitos.
- **Selección de términos:** Términos (campos o fields) que se almacenarán en el documento y los términos en los que se podrá realizar una búsqueda. Los términos que han llegado a culminar este proceso se almacenarán en el índice o **fichero inverso**⁵ (fichero diccionario) y se repite el proceso para cada documento existente en la colección.

⁴ La lematización es un proceso lingüístico que tiene como finalidad hallar el lema correspondiente dada una forma flexionada. Llámesele lema a la forma que se acepta como representante de un conjunto de formas flexionadas de una misma palabra.

⁵ Proviene de la acción de indizar en donde se almacenan las relaciones existentes entre los términos y los documentos de los que se realiza la indización.

1.2.3. Modelos de Recuperación de Información

- **Modelo Booleano**

El modelo Booleano es un modelo de RI simple, basado en el álgebra booleana y la teoría de conjuntos, con la finalidad de manejar ecuaciones de lógica matemática.

Luque Rodríguez (2005) afirma:

La lógica matemática trata con proposiciones, elementos de circuitos de dos estados, etc., asociados por medio de operadores como Y, O, NO, EXCEPTO, SI...ENTONCES. Por tanto, permite cálculos y demostraciones como cualquier parte de las matemáticas, además de posibilitar la codificación de la información en el ámbito computacional. (p. 29).

Los documentos y preguntas de este modelo son representados por medio de un conjunto de términos de indización. Este modelo representa la presencia o ausencia de un término en el documento, siendo 1 la presencia del término y 0 la ausencia, de esta manera el resultado de una consulta puede clasificarse en relevante y no relevante.

Al ser un modelo simple ha sido adoptado por muchos de los primeros sistemas bibliográficos comerciales existentes, debido a su fácil diseño e implementación. (García Broncano, s. f.).

ICT - Udlap (s. f.) refiere que el modelo booleano tiene las siguientes ventajas y desventajas en la recuperación de información:

Ventajas:

- La relevancia del documento es binaria: el documento es relevante o no.
- Consultas de una palabra: el documento es relevante solo si en él existe la palabra.
- Se permite el uso de operadores lógicos como AND o el operador OR.

Desventajas:

- No permite diferenciar entre documentos más o menos relevantes
- No importa si el documento tiene las palabras de consulta una o mil veces
- No es posible la ordenación de los resultados

La característica principal de este modelo es considerar la relevancia de los documentos y las variables de peso de los términos de forma binaria. Además de utilizar los ficheros inversos o archivos fantasma. (García Broncano, s. f.).

Antonio & Comeche (s. f.) refiere lo siguiente, En este modelo el fichero inverso resultante en el procesamiento de los documentos almacena los siguientes datos: palabras

que aparecen en los documentos de la colección y valor binario de aparición del término en el documento.

De esta manera una tabla con datos básicos de un fichero inverso de este modelo puede estar representado en la TABLA 1.2.

TABLA 1.2
Tabla básica de fichero inverso del Modelo Booleano

	D1	D2	...	Dn
T1	1	0	...	1
T2	0	1	...	0
...
Tt	0	1	...	1

Fuente: (Antonio & Comeche, s. f.)

De la tabla anterior se puede observar la siguiente información:

- T1, T2, ..., Tn: son los términos de indización.
- D1, D2, ..., Dn: son los documentos que componen la colección
- Los valores "1" y "0": indican la aparición o no aparición respectivamente en los documentos.

• **Modelo Probabilístico**

El modelo probabilístico fue introducido por Robertson y Sparck Jones en la década de los setenta, el cual mejora el rendimiento de los SRI mediante el uso de información procedente de la distribución estadística de los términos. (Antonio & Comeche, s. f.).

Este modelo se basa en la teoría de la probabilidad para la construcción de la función de búsqueda con la que se estima la probabilidad de que un documento es relevante a una pregunta dada por el usuario. Antonio & Comeche (s. f.) refieren que el cálculo de esta función se realiza a partir de:

- La probabilidad de aparición de los términos de cada documento en los documentos relevantes.
- Se recuperan los documentos que tienen mayor probabilidad de relevancia.
- Se ordena los documentos en orden descendentes a la probabilidad de relevancia con la pregunta del usuario.

Este modelo cuenta con un gran problema ya que es difícil determinar las estimaciones de probabilidad. Estas probabilidades no se conocen inicialmente, por lo que este modelo se ve en la necesidad de realizar una hipótesis inicial sobre sus valores. (Antonio & Comeche, s. f.).

- **Modelo Vectorial**

Para la presente investigación se hará mayor énfasis en este modelo, ya que es el más utilizado actualmente por los SRI, principalmente en la web. Como se ha podido apreciar, los modelos anteriormente mencionados realizan procesos diferentes y a la vez similares, ya que por ejemplo el modelo booleano efectúa **equiparación**⁶ exacta en cambio el modelo probabilístico efectúa equiparación parcial. Sin embargo, ninguno de los dos modelos toma en cuenta la frecuencia en la que aparecen los términos de indización en los documentos. (Antonio & Comeche, s. f.).

Para entender la importancia de la frecuencia de este modelo se propone imaginar que, si en un documento aparece una palabra **X** una vez y en otro documento aparece la misma palabra 10 veces, se considera que en el segundo documento la importancia de la palabra **X** es mayor a la del primer documento. Esta es la razón por la cual aparece el modelo Vectorial o también llamado modelo Espacio Vectorial.

Este modelo como su nombre lo indica está basado en el uso de vectores en donde, “Partiendo de que se pueden representar los documentos como vectores de términos, los documentos podrán situarse en un espacio vectorial de n dimensiones, es decir, con tantas dimensiones como elementos tenga el vector” (Rodríguez, 2005, p. 31).

Los términos que formarán la matriz o espacio vectorial serán términos no vacíos, es decir, para este modelo es de vital importancia el procesamiento de los documentos. Además, en este modelo los pesos de los términos en los documentos no son binarios como en el modelo booleano con la finalidad de calcular el grado de similitud entre los documentos en el índice y las consultas. (Carrazana, 2014).

Algunas de las ventajas que Carrazana (2014) menciona son:

- Modelo simple basado en el álgebra lineal
- Peso de términos no binario.
- Aciertos parciales, sin importar que no contenga todos los términos de la consulta
- Ordenación de resultados en base a frecuencia de los términos.
- Implementación eficiente para grandes colecciones de documentos.
- No necesita operadores.

⁶ Establece lo que tiene en común entre la búsqueda del usuario y las representaciones de los documentos.

De igual manera Carrazana (2014) menciona algunas desventajas:

- Se pierde información sintáctica y semántica del documento ya que se basa en la independencia de los términos dentro de este.
- El orden en el cual los términos aparecen en el documento se pierde en la representación espacio vectorial.
- Las palabras de búsqueda deben coincidir con las palabras del documento, partes de una palabra pueden dar en falsos positivos.

Ejemplo

Para una fácil comprensión de este modelo Martínez-Mendez (2004) nos proporciona el siguiente ejemplo básico:

Si nuestro SRI contiene los siguientes cuatro documentos:

- D1: el río Danubio pasa por Viena, su color es azul
- D2: el caudal de un río asciende en invierno
- D3: el río Rhin y el río Danubio tienen mucho caudal
- D4: si un río es navegable, es porque tiene mucho caudal

Mediante estos documentos la matriz correspondiente del modelo Espacio Vectorial se muestra en la TABLA 1.3

TABLA 1.3
Ejemplo de Matriz de términos y documentos en el Espacio Vectorial.

	Río	Danubio	Viena	color	azul	caudal	invierno	Rhin	navegable
D1	1	1	1	1	1	0	0	0	0
D2	1	0	0	0	0	1	1	0	0
D3	2	1	0	0	0	1	0	1	0
D4	1	0	0	0	0	1	0	0	1

Fuente: (Martínez-Mendez, 2004, p. 10)

Para facilitar el presente ejemplo se muestra los términos originales, es decir, no se ha realizado el proceso de stemming, y de igual manera suponiendo que el SRI ha eliminado los determinantes, verbos y preposiciones (el, pasa, por, etc.) que aparecen en los documentos.

El SRI realiza varias operaciones para entregar una respuesta al usuario. Lo primero que hace es traducir la consulta del usuario a un vector. Por ejemplo, si la consulta fuera: “¿cuál es el caudal del río Danubio?”, el vector perteneciente a la consulta sería: $Q = (1,1,0,0,0,1,0,0,0)$. El siguiente paso que el SRI realiza es calcular la similitud que existe entre el vector de la consulta y los vectores de los documentos para finalmente ordenar la respuesta en base a dicha similitud.

Para el cálculo de la similitud lo primero que el SRI debe hacer es construir una matriz de términos y documentos en el Espacio Vectorial con los pesos calculados. Para ello es necesario considerar las siguientes definiciones que Martínez-Mendez (2004) propone:

- **n**: número de términos distintos en la colección de documentos
- **tfij**: número de ocurrencias de término t_j en el documento D_i [frecuencia del término o **tf**]
- **dfj**: número de documentos que contienen el término t_j
- **idfj**: frecuencia inversa del documento (usado para medir el valor de discriminación entre un término a otro). Para calcular este valor usamos la fórmula $\log(d/dfj)$, donde d es el número total de documentos

El peso de cada término es el producto entre la frecuencia del término y su frecuencia inversa ($peso_{td} = frecuencia_{td} * idf_t$). (Martínez-Mendez, 2004). A continuación, se muestra el cálculo de las frecuencias inversas.

- $Idf(\text{río}) = \text{Log}(4/4) = \log(1) = 0$
- $Idf(\text{Danubio}) = \text{Log}(4/2) = \log 2 = 0.301$
- $Idf(\text{Viena}) = \text{Log}(4/1) = \log 4 = 0.602$
- $Idf(\text{color}) = \text{Log}(4/1) = \log 4 = 0.602$
- $Idf(\text{azul}) = \text{Log}(4/1) = \log 4 = 0.602$
- $Idf(\text{caudal}) = \text{Log}(4/3) = \log 1.33 = 0.124$
- $Idf(\text{invierno}) = \text{Log}(4/1) = \log 4 = 0.602$
- $Idf(\text{Rhin}) = \text{Log}(4/1) = \log 4 = 0.602$
- $Idf(\text{navegable}) = \text{Log}(4/1) = \log 4 = 0.602$

Ahora se tiene los valores necesarios para el cálculo de los pesos, los cuales se muestran en la TABLA 1.4.

TABLA 1.4
Matriz de términos y documentos en el Espacio Vectorial con pesos

	Río	Danubio	Viena	color	azul	caudal	invierno	Rhin	navegable
D1	0	0.301	0.602	0.602	0.602	0	0	0	0
D2	0	0	0	0	0	0.124	0.602	0	0
D3	0	0.301	0	0	0	0.124	0	0.602	0
D4	0	0	0	0	0	0.124	0	0	0.602
Q	0	0.301	0	0	0	0.124	0	0	0

Fuente: (Martínez-Mendez, 2004, p. 12)

Ahora corresponde calcular la similitud existente entre los documentos y la consulta del usuario, para ello es necesario realizar el producto escalar de vectores, el cual consiste en multiplicar componente a componente de los vectores y sumar los resultados. Se

realizará el cálculo de la similitud del primer documento y se mostrará los resultados del resto para facilidad del ejemplo

- **Sim (D1, Q)** = $0*0 + 0.301*0.301 + 0.602*0 + 0.602*0 + 0.602*0 + 0*0.124 + 0*0 + 0*0 + 0*0$ = 0.09
- **Sim (D2, Q)** = 0.01
- **Sim (D3, Q)** = 0.10
- **Sim (D4, Q)** = 0.01

El siguiente proceso que realiza el SRI es ordenar los resultados (D3, D1, D2, D4) de manera que la mayor similitud será la más relevante, es decir, será la que más satisface a la consulta del usuario. En el ejemplo propuesto se puede observar que la similitud del documento 1 (D1) es el mayor, por lo tanto, será el más relevante para la consulta.

Se puede observar que en este ejercicio existe un caso de acierto y un caso de fallo de este modelo, ya que el documento con mayor similitud (D3) sí responde a la pregunta y al mismo tiempo los demás documentos no responden adecuadamente (realmente la similitud es muy baja). Estos casos justifican el hecho de que los SRI también presentan documentos no relevantes en sus respuestas.

Aplicaciones del Modelo Espacio Vectorial

Carrazana (2014) menciona algunas de las aplicaciones que utiliza el modelo espacio vectorial para la recuperación de información:

- SMART: primer sistema de recuperación de información basado en el Modelo Espacio Vectorial.
- Apache Lucene: API (Open Source) para RI, que permite realizar búsquedas de texto de alto rendimiento, eficientes y rápidas.
- Weka: aplicación muy usada para realizar minería de datos.
- Sistemas anti-plagio. Utilizado para comparar la similitud entre documentos. Asociando un vector a cada documento.

1.2.4. Comparativa de los modelos

Para resumir sobre los modelos clásicos de RI, en la TABLA 1.5 se presenta una comparativa de los modelos estudiados.

TABLA 1.5
Comparativa entre Modelos clásicos de Recuperación de Información

	Modelo Booleano	Modelo Probabilístico	Modelo Vectorial
Coincidencia o Equiparación	Exacto	Parcial	Parcial
Frecuencia	No se toma en cuenta	No se toma en cuenta	Si se toma en cuenta
Pesos	Binarios	Binarios	No binarios
Base	Álgebra Booleana y Teoría de conjuntos	Teoría de la probabilidad	Álgebra lineal (Uso de vectores)

Autor: Propia
Fuente: (Antonio & Comeche, s. f.)

1.3. APLICACIONES JAVA ENTERPRISE EDITION (JEE)

1.3.1. Introducción

La internet ha tenido una gran evolución durante los últimos años, pasando del uso de páginas web estáticas que permitían el manejo de información en texto plano a aplicaciones web que son capaces de realizar cálculos y procesar información de una manera que en años pasados se creía difícil de lograr.

Las aplicaciones web actualmente se han convertido en una herramienta indispensable para el procesamiento de información, por lo que el avance de las tecnologías de desarrollo web han ido evolucionando de la mano de estas, y está en nuestras manos como desarrolladores investigar las mejores tecnologías para satisfacer las necesidades de los usuarios.

La tecnología de las aplicaciones Java Enterprise Edition constituyen una solución que la empresa Sun propuso con la finalidad de dar soluciones en el desarrollo de **aplicaciones distribuidas**⁷. “La base de esta solución se sustenta en el lenguaje Java, también creado por Sun. Este lenguaje básico también es conocido con el término JSE (Java Standard Edition)” (Groussard, 2010, p. 15).

Otro punto fundamental que Groussard (2010) refiere es que, JEE no es una mejora de JSE, ya que JSE es suficientemente poderosa por sí misma. Dicho de otra manera, JEE

⁷ Aplicaciones compuestas por diferentes componentes que se ejecutan en entornos separados, las que generalmente se encuentran en plataformas diferentes conectadas en red.

se puede considerar como una normativa que describe los elementos que conforma una aplicación distribuida y algunos de estos elementos pueden ser:

- La manera en la que los componentes de una aplicación deben ser desarrollados, entre estos podemos mencionar: servlets, páginas JSP.
- La manera en la que estos componentes deben comunicarse entre sí y con otras aplicaciones, estos pueden ser: JDBC, JavaMail, JNDI.
- La manera en la que se organizan estos componentes para asegurar el correcto funcionamiento de las aplicaciones, en este apartado se puede mencionar el descriptor de despliegue.
- Las diferentes restricciones que los servidores que se encargarán del alojamiento de nuestras aplicaciones deben respetar.

1.3.2. Servidor Web y Servidor de Aplicaciones

Estos dos términos generan gran confusión en el campo del desarrollo de aplicaciones Web. Por un lado, un servidor web es un servidor de archivos web, el que se encarga de gestionar peticiones puramente HTTP. Por otro lado, un servidor de aplicaciones realiza un trabajo más complejo. El servidor de aplicaciones se caracteriza por encargarse de las peticiones dinámicas del cliente, es decir, que contienen código ejecutable alojado en el servidor. En este caso el servidor de aplicaciones inicia sus funciones mediante una petición HTTP por parte del cliente, la cual es analizada para determinar el recurso solicitado, y al contrario de un servidor web, este ejecuta el código y devuelve al cliente el resultado al código previamente ejecutado.

De hecho, la confusión se produce frecuentemente entre estos dos elementos porque generalmente un servidor de aplicaciones toma también las funciones de servidor Web. Cuando el servidor recibe una petición HTTP proveniente del exterior, es la parte del servidor Web la que recibe esta petición y la analiza. Si concierne a un recurso estático, el servidor Web realiza su función yendo a buscar el recurso y reenviándolo al cliente en una respuesta HTTP. Si la petición concierne a un recurso dinámico (código), el servidor Web no sabe tratar esta petición, por lo que la transfiere a la parte correspondiente al servidor de aplicaciones del servidor. Éste realiza su función ejecutando el código correspondiente y generando una respuesta HTTP. Si así lo requiera, el servidor de aplicaciones puede contactar con otro servidor o una base de datos para poder construir la respuesta. Esta respuesta HTTP se transmite al servidor Web que a su vez se encarga de reenviarla al cliente. (Groussard, 2010, p. 16).

1.3.3. Estructura de las aplicaciones JEE

Groussard (2010) refiere que una aplicación Java EE esta generalmente constituida por los siguientes elementos:

- Elementos estáticos como páginas HTML, imágenes, audio, hojas de estilo, etc.
- Elementos dinámicos como servlets, páginas JSP, Java Beans.
- Librerías de clases
- Descriptor de despliegue que es el encargado de definir parámetros importantes para un correcto funcionamiento de los sistemas, estos parámetros pueden ser páginas por defecto, páginas de errores, entre otros.

1.3.4. Java frente a otros lenguajes de programación

En el top 10 de los lenguajes de programación más populares según el Índice TIOBE (The Importance of Being Earnest) realizado por la empresa TIOBE Software BV (Holanda) en enero de 2018, tenemos el siguiente orden: Java, C, C++, Python, C#, JavaScript, Visual Basic, .NET, R, PHP y Perl. Es necesario mencionar que esta lista puede variar dependiendo de las necesidades y objetividad del estudio, por ende, no es una lista que desvalore ni mucho menos desprestigie a los lenguajes de programación.

Además, se puede tomar como referencia resultados ofrecidos por Google Trends que muestra la popularidad de búsquedas de diferentes términos, que en el caso del presente documento serán los 5 lenguajes de programación que lideran la lista mencionada anteriormente.

En las Fig. 6 y 7 se puede observar que en los últimos meses Java ha sido más popular que otros lenguajes de programación, tanto en Ecuador como a nivel mundial.

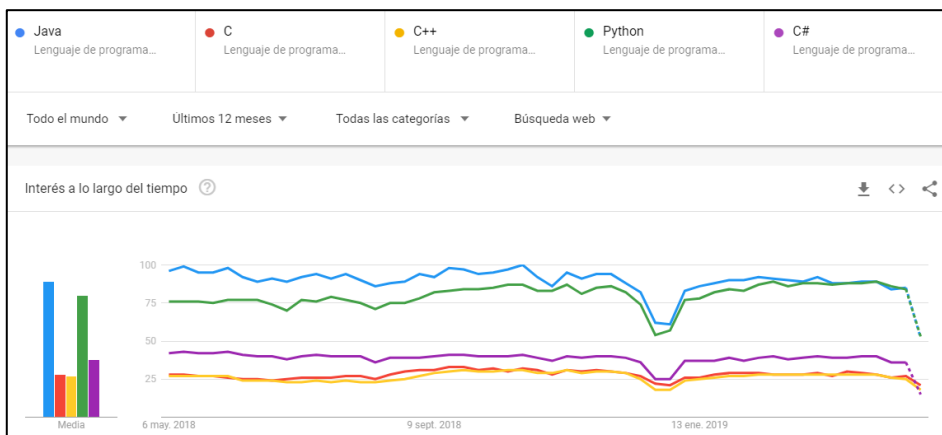


Fig. 6. Comparativa de popularidad 2018 - 2019 de lenguajes de programación a nivel mundial
Fuente: Google Trends



Fig. 7. Comparativa de popularidad 2018 de lenguajes de programación en Ecuador
Fuente: Google Trends

En el presente estudio se ha tomado a Java como mejor opción para nuestra necesidad por su robustez, además de ser la opción perfecta para adaptarse con la API Lucene para búsquedas de alto desempeño desarrollada puramente en este lenguaje.

1.4. EVOLUCIÓN DE LOS SISTEMAS WEB CON RESPECTO A LA BÚSQUEDA DE INFORMACIÓN

En la Fig. 8 se presenta la figura que resume la evolución de los sistemas web con respecto a la recuperación de información

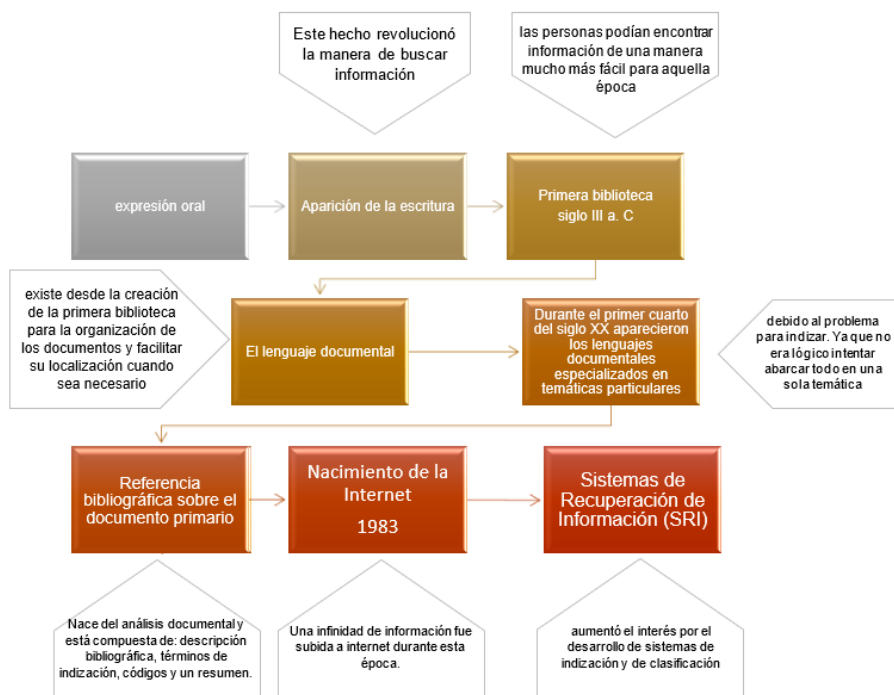


Fig. 8. Evolución de los sistemas web con respecto a la RI
Autor: Propia
Fuente: (Rodríguez Perojo & Ronda León, 2006)

1.5. API LUCENE

1.5.1. Introducción

Lucene es una API o librería de código abierto (Open Source) bajo la licencia de Apache para RI, que permite realizar búsquedas de texto de alto rendimiento, eficientes y rápidas. (Madrid Gorelov et al., 2007).

Esta herramienta integra funciones escritas completamente en el lenguaje de programación JAVA, capaz de proporcionar las características de un motor de búsqueda como indexación y búsquedas de información textual a cualquier aplicación que la necesite, siendo de gran ayuda para los desarrolladores. Esta API tiene versiones para diferentes lenguajes de programación como Perl, C++, C#, Ruby, entre otros. (Madrid Gorelov et al., 2007).

Lucene va más allá que una simple búsqueda en una Base de Datos, ya que esta API utiliza índices que nos permite buscar todo tipo de información que pueda ser representada de una forma textual. MARQUÉS (2015) afirma: “El concepto a nivel de arquitectura de Lucene es simple, básicamente los documentos (document) son divididos en campos (fields) y se genera un índice sobre estos campos de texto”.

Lucene permite indexar y buscar información que se encuentre almacenada en documentos como páginas web e incluso documentos de Microsoft Word, PDF o cualquier tipo de archivo al cual se le pueda extraer información textual. (Hernández Moreno, Mayorga Cervantes, & Torres Téllez, 2015).

En la Fig. 9, se muestra la forma en la que una aplicación hace uso de Lucene

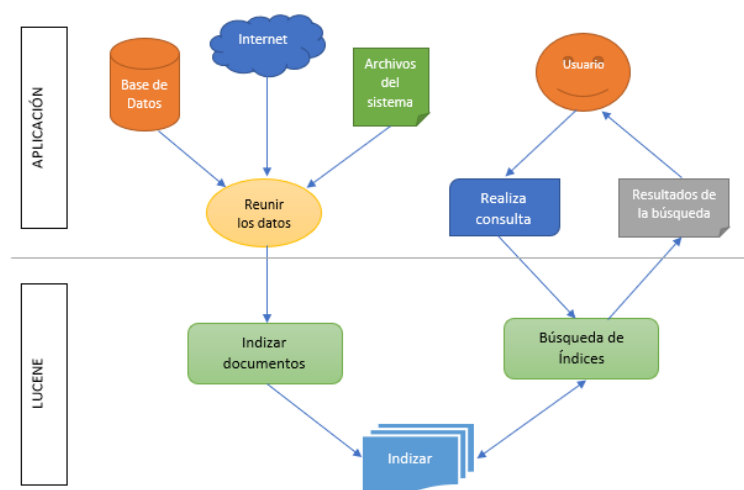


Fig. 9. Uso de Lucene en una aplicación

Autor: Propia

Fuente: (Pablo Ramos Hernández & Alor-Hernández, 2008)

1.5.2. Características Principales

Lucene proporciona a los desarrolladores, distintas características que permiten un fácil uso de esta potente herramienta. “Ésta herramienta tiene utilidades para sistemas de recuperación de información, en específico, utilidades para la implementación un motor de búsqueda y el indexado de documentos” (Hernández Moreno et al., 2015, p. 21).

The Apache Software Foundation (s. f.) en su sitio web oficial presenta las características de Lucene, a continuación, mencionaremos las más importantes:

- Pequeños requisitos de memoria RAM
- Tamaño del índice aproximadamente 20-30% del tamaño del texto indexado
- Búsqueda clasificada - los mejores resultados primero
- Búsqueda de campo (por ejemplo, título, autor, contenido).
- Permite actualización y búsqueda simultáneas.
- Disponible como software de código abierto bajo la licencia Apache, que le permite utilizar Lucene en programas comerciales y de código abierto.

Madrid Gorelov et al. (2007) refiere que, Lucene al ser una librería orientada a objetos se compone de clases que a su vez tiene métodos y propiedades. Las clases que componen esta librería se dividen en clases de indización y clases para la búsqueda, las cuales detallamos a continuación.

1.5.3. Fases Principales de Lucene

Indización

La forma habitual cuando se habla de RI en entornos documentales es **indización**, sin embargo, en entornos informáticos es más frecuente llamarle a este proceso como **indexación**. En realidad, estos dos términos son sinónimos por lo cual se hará mención uno u otro indistintamente. (Pablo Ramos Hernández & Alor-Hernández, 2008).

El proceso de indización es el responsable del análisis y la extracción de la información verdaderamente relevante, de entre toda la información disponible. Posteriormente, con la información que se ha extraído se crea un índice mediante el cual se realizarán las búsquedas de información. El índice es una estructura de datos que agiliza el acceso a la información, este índice permite que la búsqueda de la información requerida sea rápida y eficiente, ya que esta estructura de datos es bastante parecida a los índices de los libros que permiten que los contenidos sean encontrados rápidamente.

Búsqueda

Este proceso permite realizar una consulta en el índice, con la finalidad de obtener los documentos que contienen las palabras o expresiones de las consultas que el usuario requiera.

1.5.4. Clases principales para la indización

- **IndexWriter:** componente central para la indización, su función es crear índices y agregar documentos a uno ya existente. Esta clase permite tener acceso al índice mas no leer ni buscar en él. (Pablo Ramos Hernández & Alor-Hernández, 2008).
- **Directory:** esta clase representa la ubicación del índice de Lucene. “Esta a su vez utiliza subclases FSDirectory para guardar los índices en el sistema de archivos. Esta es la clase que más se usa para el almacenamiento de índices” (Pablo Ramos Hernández & Alor-Hernández, 2008).
- **Analyzer:** esta clase entra en acción antes de realizar la indización, y su función es realizar el procesamiento de documentos, como eliminar las palabras que no aportan relevancia al documento.
- **Document:** representa una colección de campos que serán indexados y almacenados de forma separada. Estos campos representarían los atributos de un documento como título, autor, etc.
- **Field:** representa fragmentos de información de cada **document** que estará almacenado en el índice.

En la Fig. 10 se muestra las clases que intervienen en la indización de documentos con Lucene.

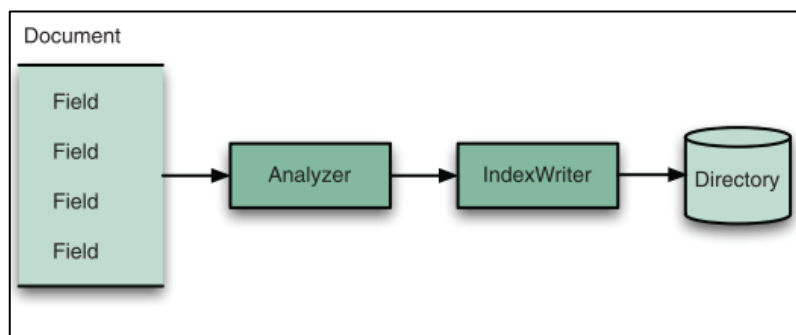


Fig. 10. Clases usadas para la indización de documentos con Lucene.
Fuente: (McCandless, Hatcher, & Gospodnetic, 2010, p. 26)

1.5.5. Clases principales para la búsqueda

- **IndexSearcher:** es la clase central para la búsqueda en los índices de Lucene, esta clase permite abrir el índice y analizar la búsqueda para ofrecer al usuario los resultados relevantes.
- **Term:** similar al objeto Field, “consiste en una dupla de cadena de caracteres: el nombre del campo y el valor del campo” (Madrid Gorelov et al., 2007).
- **Query:** permite definir la consulta sobre la cual se desea realizar la búsqueda.
- **TopDocs:** esta clase es un contenedor que almacena simples punteros a los mejores resultados de búsqueda, es decir, a documentos que cumplen con una consulta determinada. Para cada documento de los mejores resultados, la clase TopDocs almacena el ID de documento (usado para recuperar el documento) y la puntuación flotante de relevancia. (McCandless et al., 2010).
- **QueryParser:** esta clase realiza un parsing de las consultas realizadas por los usuarios. Madrid Gorelov et al. (2007) afirma que: “Esta clase es una de las más convenientes y rápidas en su implementación dentro de Lucene. Partiendo de la consulta del usuario, la analiza y crea internamente los objetos del tipo de la clase Query para satisfacer la búsqueda”.

1.5.6. Proceso de indización y búsqueda con Lucene

En la Fig. 11 se presenta el proceso que realiza la API Lucene para la indización y la búsqueda de documentos.

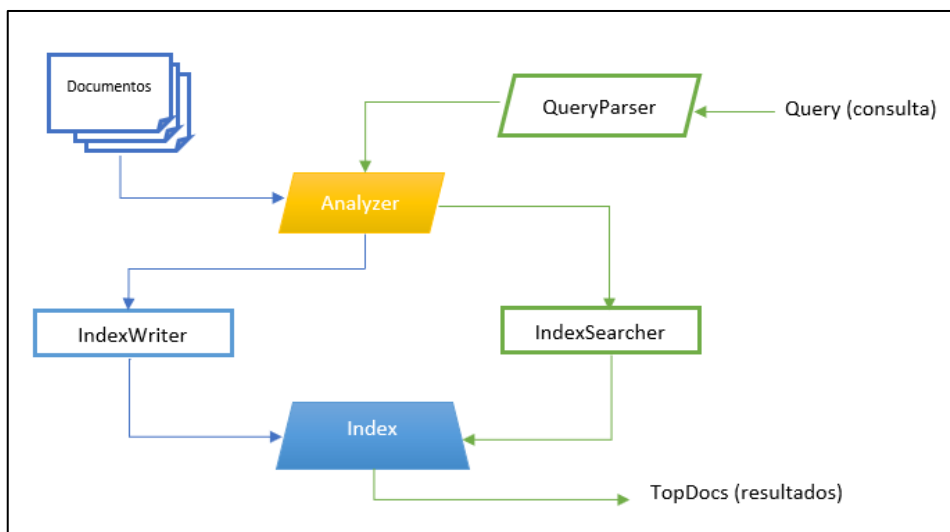


Fig. 11. Proceso de indización y búsqueda con Lucene.
Autor: Propia
Fuente: (Pablo Ramos Hernández & Alor-Hernández, 2008).

1.5.7. Popularidad de Lucene frente a otras APIs para RI

Existen varias APIs que permiten recuperar información. Estas APIs son desarrolladas en diferentes lenguajes de programación con la finalidad de dar funciones de un motor de búsquedas.

Para la presente comparativa se ha tomado algunas de las APIs de código abierto más populares escritas en JAVA. Es necesario hacer mención que las APIs mencionadas a continuación trabajan en el mismo ámbito, ya que muchas veces se confunde el término de API de RI como lo es Lucene, con servidor de búsquedas como lo es Solr, un servidor de búsquedas basado en el API Lucene. Algunas de las APIs tomadas en cuenta son: Nutch, Lucene y YaCy. («Open Source Search Engines in Java», s. f.)

Para facilidad del estudio se usa Google Trends. En la Fig.12 se muestra la popularidad entre las distintas APIs de RI.

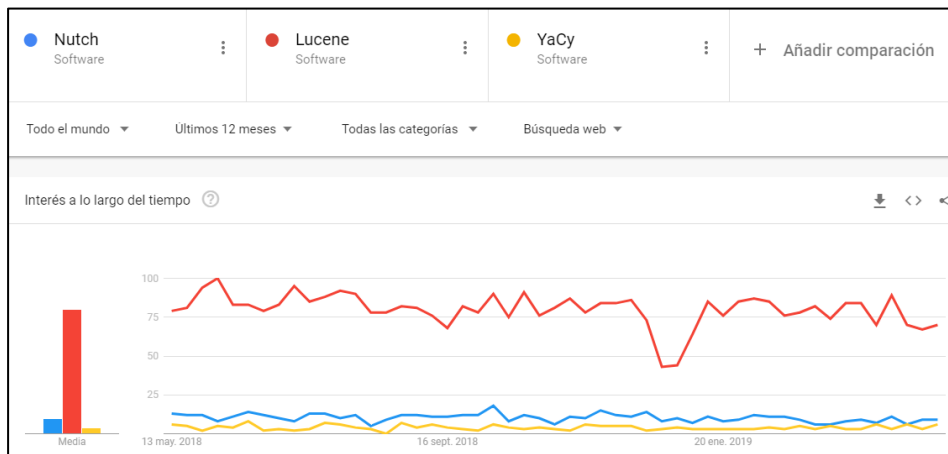


Fig. 12. Comparativa de popularidad 2018 - 2019 de Lucene frente a otras APIs de RI a nivel mundial
Fuente: Google Trends

La popularidad de Lucene frente a otras APIs de RI es muy notable, debido a que proporciona una fácil implementación frente a grandes beneficios, por lo que ha sido usada por miles de desarrolladores en todo el mundo.

1.6. METODOLOGIAS PARA EL DESARROLLO DE SOFTWARE

1.6.1. El Manifiesto Ágil

En marzo del 2001, se reunieron 17 críticos de modelos de producción de software para discutir temas referentes al desarrollo de software. En dicha reunión se presentó el término “Métodos Ágiles” como alternativa a las metodologías formales que según estos críticos eran excesivamente pesadas. (Palacio, 2007).

Palacio (2007) afirma que: “Los integrantes de la reunión resumieron en cuatro postulados lo que ha quedado denominado como “Manifiesto Ágil”, que son los principios sobre los que se basan estos métodos”. Dichos valores son los siguientes:

- Mayor valoración por los individuos y su interacción que por los procesos y las herramientas.
- Mayor valoración por software funcionando que por documentación intensiva.
- Mayor valoración por la colaboración entre desarrolladores y cliente que por la negociación contractual.
- Mayor valoración por la respuesta al cambio que por el seguimiento de un plan

1.6.2. Principios de Manifiesto Ágil

Palacio (2007) refiere que, El manifiesto ágil a través de sus valores en los que está fundamentado establece 12 principios:

1. **Satisfacer al cliente** mediante la entrega oportuna y continua de software funcional.
2. **Requisitos cambiantes en el proceso de desarrollo de software**, incluso luego del proceso de levantamiento de requisitos.
3. **Entrega de software funcional en periodos cortos de tiempo**, los cuales pueden ser en un par de semanas o hasta meses.
4. **Trabajo en colaboración entre desarrolladores y personas del negocio** durante todas las etapas del proyecto.
5. **Motivación a los integrantes del equipo**, dándoles la oportunidad de generar confianza en la realización de sus tareas.
6. **Conversaciones cara a cara**, a fin de tener una comunicación más eficiente y efectiva sobre información que debe conocerse dentro del equipo de desarrollo.
7. **Software funcional** como medida de progreso
8. **Ritmo constante de manera indefinida por parte de todos los miembros del equipo**, como desarrolladores, usuarios y patrocinadores.
9. **Búsqueda de la excelencia** dándole atención continua para enaltecer la agilidad.
10. **Simplicidad** como arte encargada de la maximización de trabajo realizado.
11. **Autoorganización dentro del equipo**, dando como resultado mejores requisitos, arquitecturas y diseños.
12. **Revisiones constantes** que permitan la reflexión del equipo sobre mejoramiento de eficiencia y ajustes de conducta en caso de ser necesario.

1.6.3. Agilismo en desarrollo de software

El agilismo dentro del desarrollo de software hace referencia a un marco de trabajo diferente a las metodologías tradicionales, las mismas que buscan imponer una disciplina rigurosa sobre el proceso de desarrollo de software a fin de que el producto satisfaga los requisitos de usuario y reúna estándares de calidad aceptables. (Labrin, 2004).

Esto no quiere decir que el desarrollo ágil no sea disciplinado ni planificado, al contrario, se requiere de una disciplina comprometida por el desarrollo de software, mediante planificaciones no rigurosas como en el desarrollo tradicional sino planificaciones abiertas a cambios dependiendo de las necesidades que puedan producirse durante el proceso de desarrollo.

Labrin (2004) afirma:

Una metodología tradicional potencia la planificación detallada y de largo alcance de prácticamente todo el desarrollo de software (ejemplo Modelo Cascada). En contraste, las metodologías ágiles proponen procesos que se adaptan y progresan con el cambio, llegando incluso hasta el punto de cambiar ellos mismos.

1.6.4. Comparativa entre metodologías tradicionales y metodologías ágiles

Las principales diferencias entre las metodologías de desarrollo tradicionales con las metodologías ágiles están enfocadas tanto al producto de software como al trabajo en relación del equipo con el cliente. En la TABLA 1.6 se destacan las principales diferencias.

TABLA 1.6
Comparativa entre metodologías tradicionales y metodologías ágiles

Metodologías tradicionales	Metodologías ágiles
Predictivos	Adaptativos
Orientados a procesos	Orientados a personas
Proceso rígido	Proceso flexible
Se concibe como un proyecto	Un proyecto es subdividido en varios proyectos más pequeños
Poca comunicación con el cliente	Comunicación constante con el cliente
Entrega de software al finalizar el desarrollo	Entregas constantes de software
Documentación extensa	Poca documentación

Fuente: (Cadavid, Martínez, & Vélez, 2013, p. 31)

1.6.5. SCRUM como marco de trabajo ágil

Scrum es un marco de trabajo ágil que tiene como finalidad la colaboración en equipo para el cumplimiento en el desarrollo de proyectos de software complejos. Este marco de trabajo se ha caracterizado por su organización en el equipo, en el que se define diferentes roles a los individuos que lo conforman, de igual manera por el uso de eventos y artefactos que ayudan a una mejor relación entre estos.

Roles de Scrum

Levy, Romero Dapozo, & Pasini (2016) refieren que, El equipo de Scrum (Scrum Team) está conformado por:

- **Dueño del producto (Product Owner):** es el representante de los accionistas o clientes que usarán el producto final.
- **El equipo de desarrollo (Development Team):** es el equipo de trabajo que tiene los conocimientos necesarios para cumplir con las metas propuestas en cada uno de los **sprint**⁸.
- **Scrum Master:** persona que lidera al equipo de trabajo al cumplimiento de las reglas o las actividades que se debe desarrollar en el tiempo especificado.

Eventos de Scrum

Sprint: Es el periodo de tiempo en el que el equipo Scrum realiza un incremento en el producto de software. Según Levy et al. (2016) refieren, un sprint a su vez está conformado por otros eventos como son:

- **Reunión de planificación del Sprint:** en la que los miembros del equipo planifican el trabajo a realizar.
- **Scrums diarios (daily Scrums):** reunión que tiene una duración de 15 minutos en la que los miembros del equipo sincronizan su trabajo y discuten sobre sus avances del día anterior.
- **Revisión del Sprint:** en este evento el equipo revisa el incremento realizado y realizan una adaptación en la lista del producto en caso de ser necesario.

⁸ Iteración de tiempo en la que el equipo trabaja para convertir las historias de usuario en una versión del producto totalmente operativo.

- **Retrospectiva de Sprint:** este evento le permite al equipo una autoevaluación con la finalidad de realizar sus respectivas mejoras.

Artefactos de Scrum

Levy et al. (2016) refieren lo siguiente, Los artefactos de Scrum son las herramientas que el equipo usa como base para el cumplimiento de sus tareas, y éstas son:

- **Lista de producto (Product Backlog):** es una lista ordenada de todo lo necesario en el producto.
- **Lista de tareas del Sprint (Sprint Backlog):** es el conjunto de tareas de la lista del producto que se va a realizar durante el sprint a cursar.
- **Incremento:** este artefacto es una versión del producto final, resultado del sprint finalizado.

1.7. NORMAS Y ESTÁNDARES APLICADOS EN LA INGENIERÍA DE SOFTWARE

1.7.1. Introducción

El término norma o estándar se puede definir como una especificación, la cual permite la reglamentación de procesos o productos, con la finalidad de garantizar calidad.

El Instituto de Hidrología Meteorología y Estudios Ambientales (2014) define a los estándares como: “acuerdos documentados que contienen especificaciones técnicas o criterios precisos que son utilizados consistentemente, como reglas, guías o definiciones de características para asegurar que los materiales, productos, procesos y servicios cumplen con su propósito”.

Las normas o estándares son generalmente presentados en documentos técnicos como el resultado de una experiencia o desarrollo tecnológico. Estas son aprobadas por un organismo, el cual puede ser nacional, regional o incluso internacional.

Actualmente existe infinidad de entes que se encargan de elaborar normas o estándares para distintos fines o distintas áreas, para la presente investigación se ha hecho énfasis en entes autoras de normas o estándares que más se usan dentro del área de ingeniería de software, las que se profundizarán en los siguientes apartados.

1.7.2. Asociación de Maquinaria Computacional (ACM)

La Asociación de Maquinaria Computacional (ACM por sus siglas del inglés *Association for Computing Machinery*) es una organización que reúne a los profesionales, estudiantes y toda persona que esté interesada en el área de la computación.

La ACM tiene como finalidad hacer crecer la ciencia de la computación a nivel mundial y para ello esta asociación facilita un sin número de recursos para su comunidad tales como artículos científicos, libros online y una de las librerías digitales más grandes del mundo. Además, la ACM ha desarrollado un estilo en el que cuenta con normas para citas aplicadas a las áreas de la computación, programación y afines.

1.7.3. Instituto de Ingeniería de Software (SEI)

El Instituto de Ingeniería de Software (SEI por sus siglas en inglés *Software Engineering Institute*) es un instituto estadounidense que se encarga de desarrollar modelos de evaluación y mejora en desarrollo de software, este instituto es administrado por la Universidad Carnegie Mellon.

El SEI desarrolló el modelo **SW-CMM**⁹ por lo que se ha considerado como un referente dentro de la Ingeniería de Software. Es importante mencionar que el SW-CMM ha sido el punto de arranque para todos quienes se fundamentan en modelos sobre conceptos de capacidad y madurez. (Israel G., 2018)

1.7.4. Organización Internacional de Normalización (ISO)

La Organización Internacional de Normalización (ISO por sus siglas del inglés *International Organization for Standardization*) es una organización no gubernamental encargada de la elaboración de normas para distintos fines como gestión de calidad, medio ambiente, seguridad, etc., con las cuales busca la estandarización y el mejoramiento de la calidad en productos o servicios en empresas u organizaciones a nivel internacional.

⁹ El Modelo de Madurez de Capacidades (CMM) es un modelo de evaluación a los procesos que realizados internamente por una organización.

La ISO ha elaborado una infinidad de normas o estándares que son aplicadas a nivel internacional. Las normas ISO más usadas en las áreas de ingeniería como en otras áreas son:

- **ISO 9001** - Sistemas de Gestión de Calidad.
- **ISO 14001** - Sistemas de Gestión de Medio Ambiente.
- **ISO 16949 (ISO/TS 16949)** - Calidad en el sector automotriz.
- **ISO 27001** - Sistemas de Gestión de la Seguridad de la Información.
- **ISO 25000** - Requisitos y evaluación de calidad del sistema y del software.

1.7.5. Comisión Electrotécnica Internacional (IEC)

La Comisión Electrotécnica Internacional (IEC por sus siglas del inglés *International Electrotechnical Commission*) es la principal organización mundial que se encarga de la preparación y publicación de normas internacionales para las áreas de tecnología eléctrica, electrónica y afines.

La IEC junto con la ISO forman parte del sistema internacional de normalización abarcando de esta manera diferentes áreas. Mientras la IEC se encarga de las normas que están relacionadas a las áreas de eléctrica y electrónica, la ISO se encarga de normas que estén relacionadas a otros temas, por consecuencia en las normas que no estén relacionadas de manera clara entre estas áreas se realiza un mutuo acuerdo por las dos organizaciones. Un claro ejemplo es la norma ISO / IEC 17025 en la que se establecen los requisitos a cumplir por los laboratorios de ensayo y calibración. («Sobre la Comisión Electrotécnica Internacional (IEC) – Servicio de Acreditación Ecuatoriano», 2017)

1.7.6. Instituto de Ingenieros Eléctricos y Electrónicos (IEEE)

El Instituto de Ingenieros Eléctricos y Electrónicos, conocido como IEEE es una asociación mundial técnico-profesional que se dedica al avance tecnológico al servicio de la humanidad mediante varias actividades, entre ellas se destaca la estandarización.

La Rama Estudiantil de IEEE en UTN.BA (2018) afirma que:

Los objetivos del IEEE son de carácter científico, tecnológico y educativo. Básicamente consisten en contribuir al avance de la teoría y práctica de la Electricidad, la Electrónica, las comunicaciones y el Procesamiento de la Información; a la difusión de los conocimientos teóricos y prácticos, lo cual incluye la formación profesional, así como la inserción al medio.

Algunas de las normas desarrollados por el IEEE son:

- **IEEE 802.** Este proyecto tiene como misión desarrollar estándares de **redes de área local (LAN)**¹⁰ y **redes de área metropolitana (MAN)**.¹¹
- **IEEE 830.** También conocida como Especificación de Requerimientos del Software, Especificación Funcional o Especificación del Sistema y es un documento que tiene como finalidad “reunir los requisitos completos del cliente tal de desarrollar un software de acuerdo a las exigencias del mismo” (Melendez R, 2012).

1.8. DEFINICIÓN DE LA NORMA ISO/IEC 25010

1.8.1. Introducción

“El modelo de calidad representa la piedra angular en torno a la cual se establece el sistema para la evaluación de la calidad del producto” (iso25000.com, 2014). La calidad del producto de software determina el valor del producto, el cuál puede ser determinado en base al cumplimiento de los requerimientos de usuario, es decir, si un producto de software cumple con las características de calidad de la norma ISO/IEC 25010 y cumple con los requerimientos de sus usuarios, entonces, es considerado un producto de calidad.

La norma ISO/IEC 25010 nos presenta dos modelos con sus respectivas características y subcaracterísticas. Fundamentalmente los modelos tienen gran similitud en cuanto a sus características, pero su diferencia está en el ámbito de aplicación, mientras uno se encarga de la calidad en el uso el otro modelo se encarga de la calidad del producto. Para fines del presente proyecto nos centraremos en la definición del modelo de calidad del producto.

1.8.2. Modelo de la Calidad del Producto

“La norma ISO 25010 provee un marco de referencia para medir la calidad del producto software y describe 8 características y 38 subcaracterísticas de calidad de producto software” (Pardo, Pino, García, & Piattini, 2009).

¹⁰ Una red LAN es una red de computadoras que tiene un área o alcance reducido, como una casa o un edificio y que es usada para compartir recursos como una impresora o incluso internet.

¹¹ Una red MAN es una red de alta velocidad que puede dar cobertura a un área geográfica extensa permitiendo la integración de servicios mediante la transmisión de voz y video.

La Fig. 13 detalla las características y subcaracterísticas del modelo de la calidad del producto de la norma ISO/IEC 25010.

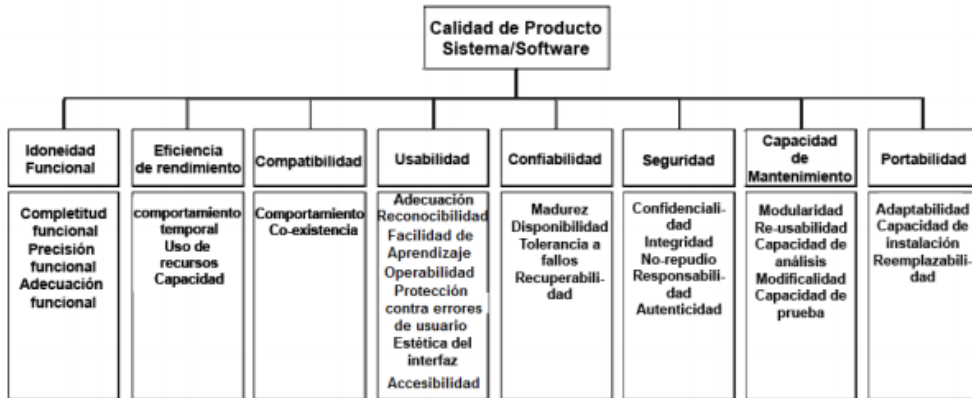


Fig. 13. Modelo De Calidad Del Producto ISO/IEC 25010
 Fuente: (Subcomité 7, 2016)

CAPÍTULO 2

Desarrollo

2.1. Definición del Proceso de Búsqueda de normas

Para el presente proyecto se desarrolló el levantamiento del proceso de Búsqueda de Normas y estándares para Ingenierías, el cual está detallado en el **Anexo A**. A continuación, se muestra el proceso en modo de resumen.

Al proceso de normas se lo ha dividido en dos categorías, las cuales son:

- Con finalidad administrativa.
- Con finalidad de búsqueda.

En la Fig. 14 se muestra el proceso de búsqueda de normas con finalidad administrativa.

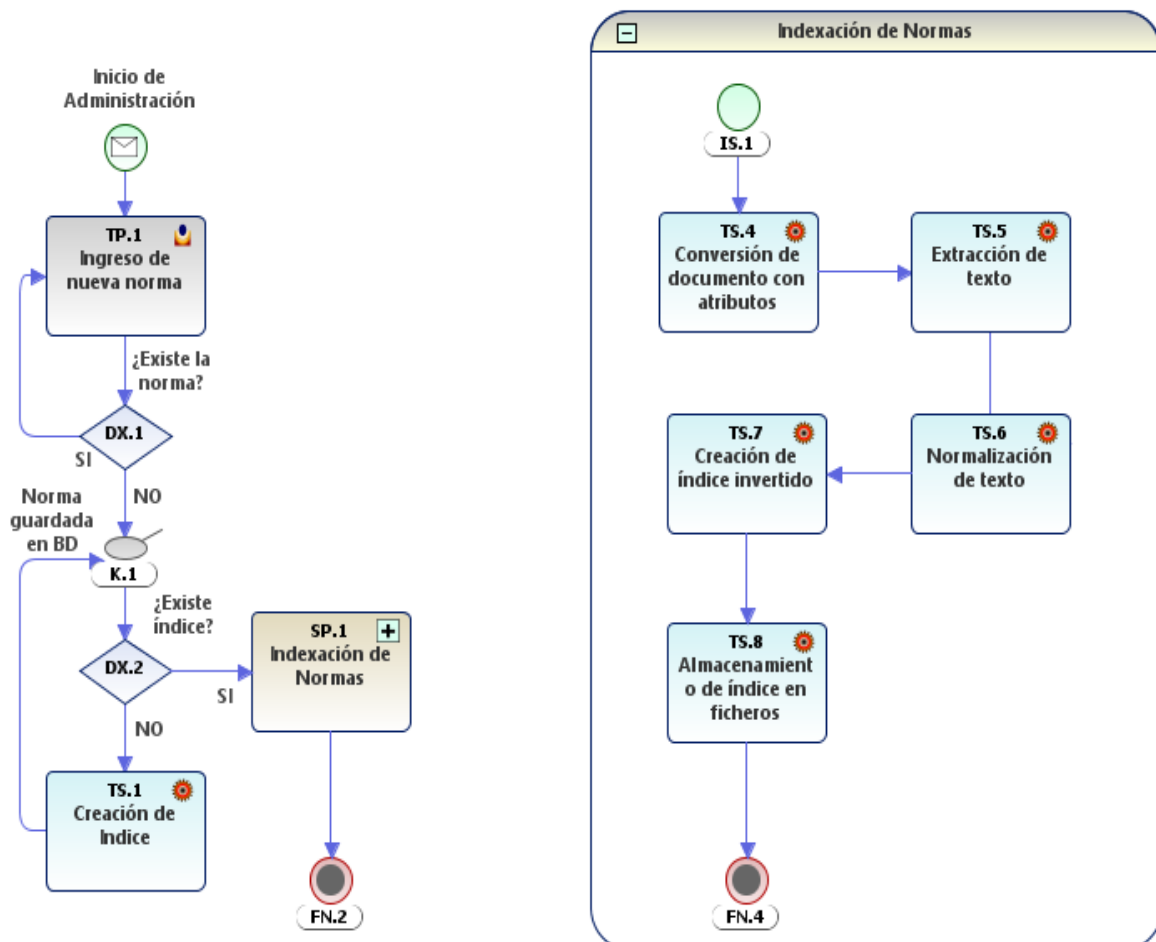


Fig. 14. Diagrama BPMN del Proceso de Búsqueda de normas (con finalidad administrativa)

Fuente: Propia

En la Fig. 15 se muestra el proceso de Búsqueda de normas con finalidad de búsquedas

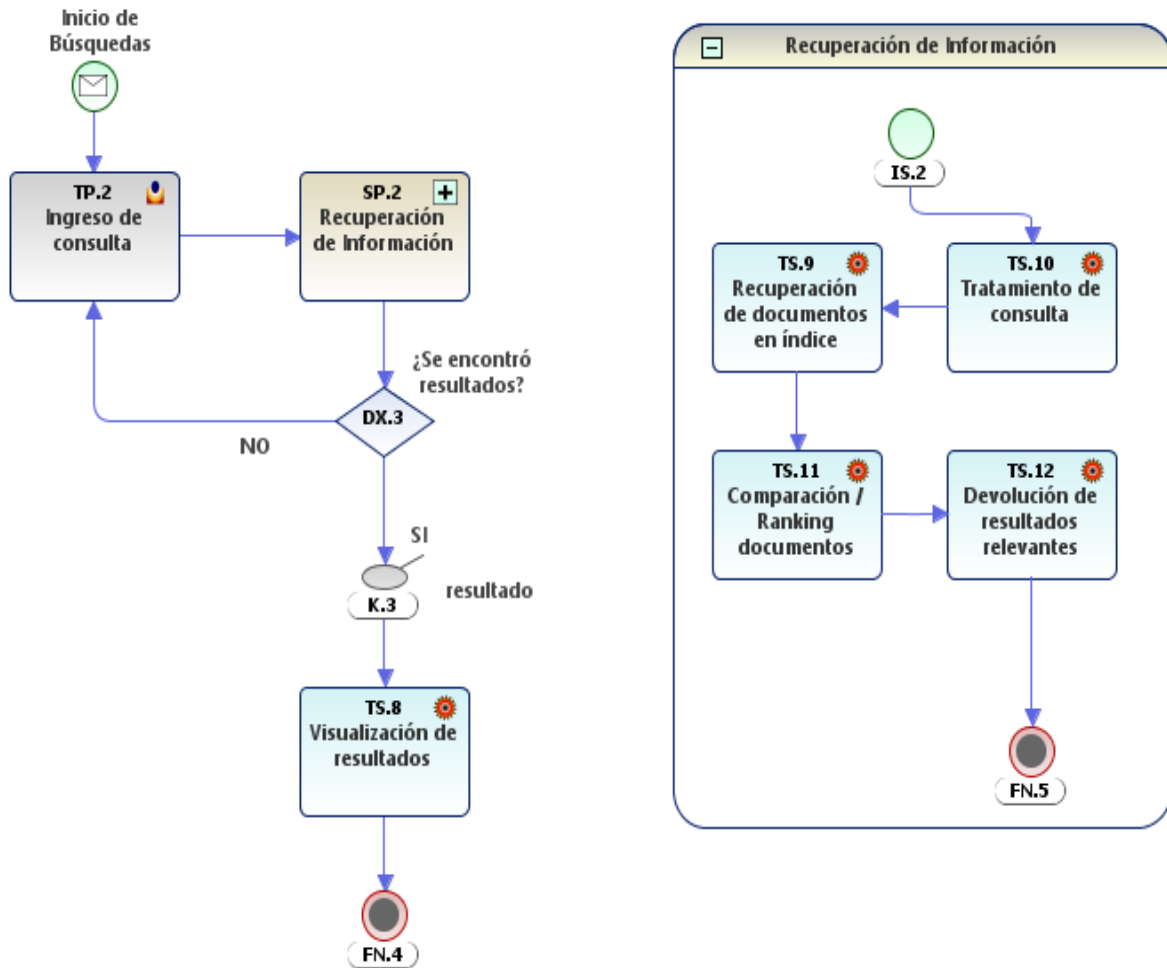


Fig. 15. Diagrama BPMN del Proceso de Búsqueda de normas (con finalidad de búsqueda)

Fuente: Propia

2.2. Definición de indicadores de calidad considerando la subcaracterística de Comportamiento Temporal dentro de la característica de Eficiencia de Desempeño de la norma ISO 25010

En la Fig. 16 se muestra de manera general la manera en la que está distribuida la norma ISO/IEC 25010:2011, de la cuál para el presente proyecto se tomará en cuenta la subcaracterística de Comportamiento Temporal (2.1 en la figura)

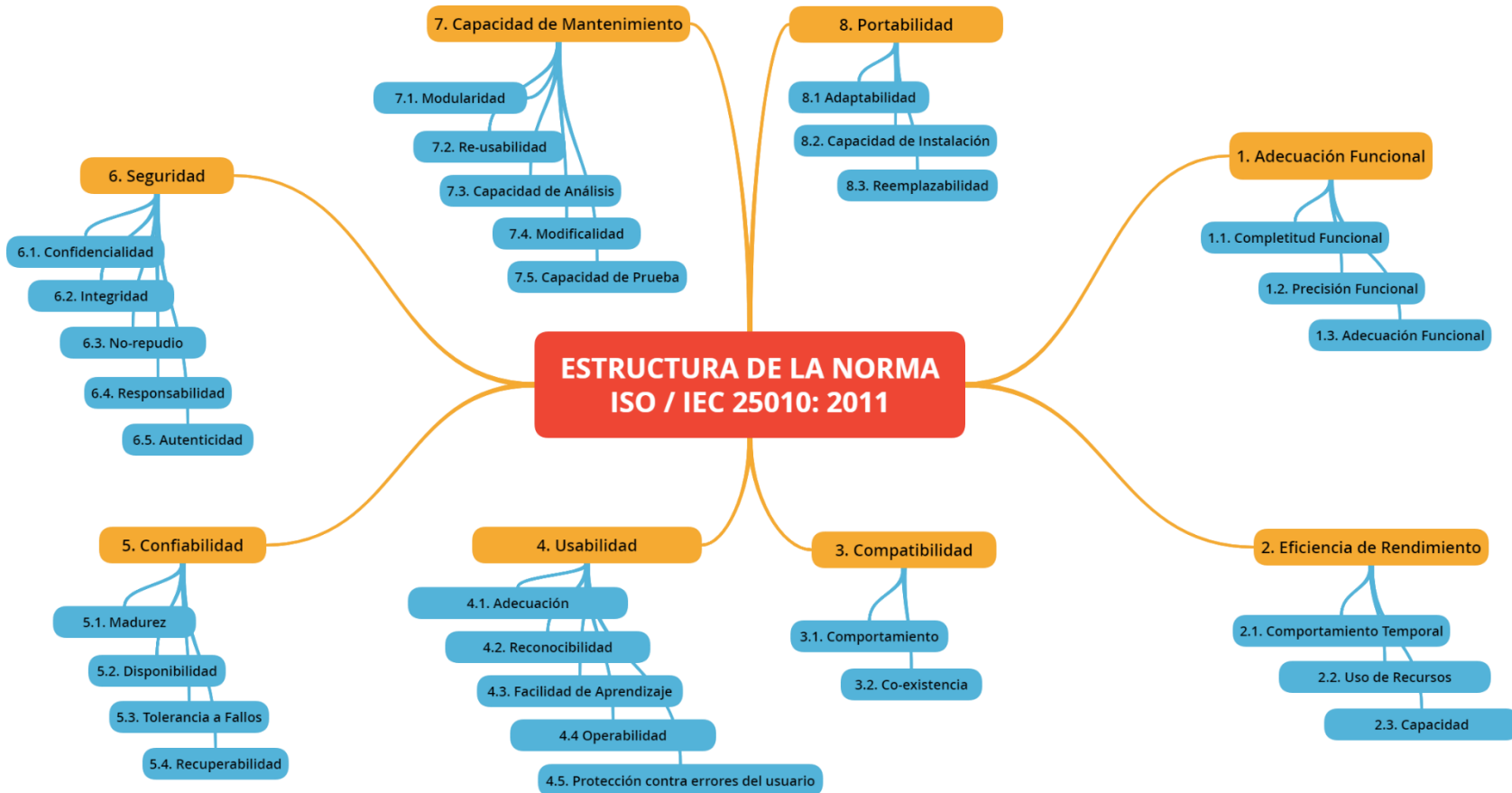


Fig. 16. Estructura General de la norma ISO/IEC 25010:2011
Fuente: Propia

Para cumplir con la subcaracterística de Comportamiento Temporal de la norma ISO/IEC 25010 y realizar la evaluación de calidad del presente proyecto se ha tomado los siguientes indicadores:

- Tiempo de respuesta
 - Con búsquedas de una palabra
 - Con búsquedas de dos palabras
 - Con búsquedas de tres palabras
- Cantidad de Resultados obtenidos
 - Con búsquedas de una palabra
 - Con búsquedas de dos palabras
 - Con búsquedas de tres palabras
- Evaluación de los mismos indicadores realizando búsquedas por palabras claves, las mismas que el sistema de normas incorpora en cada registro de norma.

Se realizará el proceso de medición de estos indicadores por el número de repeticiones que el usuario encuentre pertinente a cada palabra o grupo de palabras, datos que serán de importancia para la validación de resultados.

2.3. Metodología de desarrollo

La metodología de desarrollo usada en el presente proyecto es SCRUM por ser una metodología de desarrollo ágil y que permite un desarrollo incremental en lugar del desarrollo completo del producto.

Es importante mencionar que cada Sprint de desarrollo consta de dos semanas con un total de 28 horas productivas por sprint.

2.4. Roles SCRUM

Product Owner: Ing. Mauricio Xavier Rea

Scrum Master: Ing. Mauricio Rea

Team: Richard Tarupí

2.5. Artefactos SCRUM

2.5.1. Matriz de planificación

PLANIFICACIÓN DE TRABAJOS DE DESARROLLO

Sprint 1
Total horas 28
Fecha Inicio SP1: 21/11/2018
Fecha Final SP1: 5/12/2018

Historia de usuario	Desarrollador	Fase Desarrollo	Tarea	Tipo	Tiempo Estimado (Horas)	Tiempo Real (Horas)	Estado
Matriz de planificación	Richard Tarupí	Planificación	Formalización de la matriz de planificación	Nueva	2:00	1:00	HECHO
			Organización y análisis de los documentos para los Sprint 1 - 2 - 3		1:00	1:00	HECHO
Acta de constitución	Richard Tarupí	Desarrollo	Recolección de información del proyecto (Datos, Patrocinadores)	Nueva	0:30	0:10	HECHO
			Identificación de Propósito y Justificación del proyecto		0:30	0:20	HECHO
			Descripción del Proyecto y Entregable		0:30	0:30	HECHO
			Descripción de Requerimientos de alto nivel (Requisitos de producto y proyecto)		1:00	1:00	HECHO
			Cronograma de hitos principales		0:30	0:20	HECHO
			Definición de Presupuesto estimado		0:30	0:20	HECHO
			Lista de Interesados (stakeholders)		0:20	0:10	HECHO
			Identificación de Requisitos de aprobación del proyecto		0:30	0:30	HECHO
			Asignación del Gerente de Proyecto y nivel de autoridad		0:10	0:10	HECHO
			Asignación de Personal y recursos preasignados		0:30	0:20	HECHO
Especificación de requisitos	Richard Tarupí	Desarrollo	Desarrollo de la parte introductoria (Propósito - Alcance - Personal Involucrado - Definiciones, Abreviaturas - Referencias - Resumen)	Nueva	1:00	0:30	HECHO
			Desarrollo de la descripción general (Perspectiva del Producto - Funcionalidad del Producto -		1:00	0:40	HECHO

			Características de los Usuarios - Restricciones - Suposiciones y dependencias - Evolución del Sistema)				
			Análisis de los requisitos específicos y requisitos de Interfaz (De Usuario - De Hardware - De Software - De Comunicación)	Nueva	1:00	1:25	HECHO
			Análisis de los requisitos funcionales		1:00	2:00	HECHO
			Análisis de los requisitos no funcionales (De Rendimiento - Seguridad - Fiabilidad - Disponibilidad - Mantenibilidad - Portabilidad - Otros)	Nueva	1:00	0:35	HECHO
Cartillas de historias de usuario	Richard Tarupí	Desarrollo	Creación de las historias de usuario: Generales	Nueva	1:00	0:30	HECHO
			Creación de las historias de usuario: Modulo Administrativo		1:00	1:00	HECHO
			Creación de las historias de usuario: Modulo Operativo		1:00	0:50	HECHO
			Creación de las historias de usuario: Modulo de Búsquedas		2:00	1:30	HECHO
Backlog de historias de usuario	Richard Tarupí	Desarrollo	Llenado de la matriz de H.U: Generales	Nueva	1:00	1:00	HECHO
			Llenado de la matriz de H.U: Modulo Administrativo		1:00	0:50	HECHO
			Llenado de la matriz de H.U: Modulo Operativo		1:00	1:30	HECHO
			Llenado de la matriz de H.U: Modulo de Búsquedas		2:00	1:30	HECHO
Casos de uso	Richard Tarupí	Desarrollo	Desarrollo de los diagramas de casos de uso: Generales	Nueva	1:00	1:00	HECHO
			Desarrollo de los diagramas de casos de uso: Modulo Administrativo		1:00	1:00	HECHO
			Desarrollo de los diagramas de casos de uso: Modulo Operativo		1:00	1:00	HECHO
			Desarrollo de los diagramas de casos de uso: Modulo de Búsquedas		2:00	1:35	HECHO
TOTAL SPRINT					28:00	24:15	

Sprint 2
Total horas 28
Fecha Inicio SP2: 7/12/2018
Fecha Final SP2: 21/12/2018

Historia de usuario	Desarrollador	Fase Desarrollo	Tarea	Tipo	Tiempo Estimado (Horas)	Tiempo Real (Horas)	Estado
Diagrama conceptual	Richard Tarupí	Desarrollo	Diagramas conceptuales: Generales	Nueva	0:30	0:30	HECHO
			Diagramas conceptuales: Modulo Administrativo		0:30	0:45	HECHO
			Diagramas conceptuales: Modulo Operativo		1:00	1:15	HECHO
			Diagramas conceptuales: Modulo Búsquedas		1:00	1:50	HECHO
Arquitectura de software	Richard Tarupí	Desarrollo	Desarrollo de la parte introductoria y descripción de la parte arquitectónica	Nueva	1:00	0:30	HECHO
			Análisis de arquitectura y definición de las herramientas y tecnologías a utilizarse		1:00	0:50	HECHO
Wireframe	Richard Tarupí	Desarrollo	Realización del prototipado: General	Nueva	1:00	1:00	HECHO
			Realización del prototipado: Módulo Administrativo		1:00	1:10	HECHO
			Realización del prototipado: Modulo Operativo		2:00	3:00	HECHO
			Realización del prototipado: Modulo Búsquedas		2:00	1:50	HECHO
Instalación y configuración de herramientas a usar	Richard Tarupí	Desarrollo	Instalación de PostgreSQL, Servidor de Aplicaciones WildFly, IDE de desarrollo Eclipse, entre otros	Nueva	2:00	2:00	HECHO
			Configuración de WildFly y Jboos Tools en eclipse		1:00	0:30	HECHO
			Configuración de WildFly 14 en eclipse IDE		1:00	0:30	HECHO
Creación y configuración del proyecto	Richard Tarupí	Desarrollo	Creación del proyecto en el WorkSpace	Nueva	1:00	0:30	HECHO
			Configuración de JPA y JSF		1:00	0:30	HECHO
			Estructuración de carpetas para el proyecto		1:00	0:45	HECHO
			importación de librerías a utilizar		1:00	0:15	HECHO
			mapeo de modelo de Base de Datos al proyecto		1:00	1:00	HECHO
Diseño de plantillas Facelets	Richard Tarupí	Diseño	Investigación de plantillas usando Facelets	Nueva	2:00	5:00	HECHO
			Creación de plantillas Facelets para cada módulo del sistema		4:00	3:00	HECHO
			Aplicación de estilos Bootstrap a las plantillas		2:00	3:00	HECHO
TOTAL SPRINT					28:00	29:40	

Sprint 3
Total horas 28
Fecha Inicio SP3: 27/12/2018
Fecha Final SP3: 10/1/2019

Historia de usuario	Desarrollador	Fase Desarrollo	Tarea	Tipo	Tiempo Estimado (Horas)	Tiempo Real (Horas)	Estado
Diseño y codificación general	Richard Tarupí	Diseño	Diseño y Codificación de la pantalla principal del sistema	Nueva	3:00	4:00	HECHO
			Diseño del login (acceso al sistema)		2:00	1:30	HECHO
		Desarrollo	Codificación del Manager y Controller de Ingreso		1:00	1:30	HECHO
			Vinculación de la vista de inicio y login con el Controller de Ingreso		1:00	1:00	HECHO
			Diseño y codificación de pantalla de selección de rol de usuario luego de acceder al sistema		4:00	3:00	HECHO
CRUD de entidades principales	Richard Tarupí	Diseño	Diseño de pantallas para CRUD del Módulo Administrativo	Nueva	3:00	2:15	HECHO
			Diseño de pantallas para CRUD del Módulo Operativo		3:00	2:45	HECHO
		Desarrollo	Creación y codificación de métodos para CRUD en Manager y Controller de usuarios		4:00	3:00	HECHO
			Creación y codificación de métodos para CRUD en Manager y Controller de normas		5:00	5:00	HECHO
			Vinculación de vistas de CRUD del Módulo Administrativo con el Controller de Usuarios		1:00	1:00	HECHO
			Vinculación de vistas de CRUD del Módulo Administrativo con el Controller de Normas		1:00	1:15	HECHO
TOTAL SPRINT					28:00	26:15	

Sprint 4
Total horas 28
Fecha Inicio SP4: 11/1/2019
Fecha Final SP4: 25/1/2019

Historia de usuario	Desarrollador	Fase Desarrollo	Tarea	Tipo	Tiempo Estimado (Horas)	Tiempo Real (Horas)	Estado
Matriz de planificacion	Richard Tarupí	Planificación	Planificación de actividades para los Sprint 4 - 5 - 6	Nueva	2:00	2:00	HECHO
Incorporación de tags (palabras claves) para una clasificación de normas	Richard Tarupí	Investigación	Investigación de componentes Primefaces para el manejo de Tags (palabras clave)	Nueva	1:00	0:30	HECHO
		Desarrollo	Desarrollo de métodos para el manejo de tags		3:00	5:00	HECHO
		Diseño	Diseño de Apartado en las normas para mostrar una nube de tags		1:00	2:00	HECHO
		Desarrollo	Vinculación de métodos con parte visual de tags		1:00	1:00	HECHO
		Pruebas	Pruebas de funcionamiento de tags		1:00	0:30	HECHO
Implementación de funcionalidades de motor de búsquedas de la API Lucene	Richard Tarupí	Investigación	Investigación de adaptabilidad de búsqueda Lucene con JPA	Nueva	2:00	2:00	HECHO
		Desarrollo	Desarrollo de métodos para indexación de Lucene		5:00	7:00	HECHO
			Desarrollo de métodos para búsquedas en el índice de Lucene		5:00	2:30	HECHO
		Diseño	Diseño de vistas para los resultados de las búsquedas de Lucene		3:00	4:00	HECHO
		Desarrollo	Vinculación de métodos con parte visual de búsquedas		2:00	1:30	HECHO
			Incorporación de Tags en los resultados de búsquedas del sistema		1:00	2:00	HECHO
Pruebas	Pruebas de funcionamiento de las búsquedas Lucene	1:00	1:30	HECHO			
TOTAL SPRINT					28:00	31:30	

Sprint 5
Total horas 28
Fecha Inicio SP5: 26/1/2019
Fecha Final SP5: 8/2/2019

Historia de usuario	Desarrollador	Fase Desarrollo	Tarea	Tipo	Tiempo Estimado (Horas)	Tiempo Real (Horas)	Estado
Desarrollo de sección de normas recientes e incorporación de estructuras gráficas para visualización de familias de normas	Richard Tarupí	Desarrollo	Codificación de la sección de normas más recientes	Nueva	0:30	0:30	HECHO
			Desarrollo de pantallas para el uso de familias de normas de las ingenierías		1:00	1:15	HECHO
			Incorporación de estructura gráfica para visualización de las familias de normas		1:00	1:45	HECHO
			Codificación de estructuras gráficas de familias de normas		5:00	9:30	HECHO
Desarrollo de funcionalidad para administración de archivos	Richard Tarupí	Investigación	Investigación de componentes Primefaces para archivos	Nueva	2:00	1:00	HECHO
		Diseño	Diseño de pantallas para la administración de archivos		2:00	1:00	HECHO
		Desarrollo	Codificación de métodos para almacenamiento de archivos		8:00	7:00	HECHO
			Vinculación de métodos con parte visual de archivos		2:00	1:30	HECHO
		Pruebas	Pruebas de funcionamiento		2:00	2:00	HECHO
Plan de pruebas	Richard Tarupí	Pruebas	Llenado de las matrices de pruebas	Nueva	1:00	1:00	HECHO
			Identificación de la herramienta de reportes y control de incidencias		1:00	0:30	HECHO
Informe del plan de pruebas	Richard Tarupí	Pruebas	Llenado de la matriz de pruebas del sistema, interfaz de usuario y seguridad	Nueva	1:00	1:00	HECHO
			Llenado de la matriz de pruebas del sistema e interfaz de usuario		0:30	0:30	HECHO
			Llenado de la lista de chequeo		1:00	1:00	HECHO
TOTAL SPRINT					28:00	29:30	

2.5.1.1. Cartillas de Historias de Usuario

Historias de Usuario Generales

Historia de Usuario Nro. 1 | Diseño del Página Principal

Historia de Usuario		
Número:	1	Usuario: Público
Nombre historia: Diseño de Página Principal		
Prioridad en negocio: Alta		Riesgo en desarrollo: Media
Puntos estimados: 5		Sprint asignada: 4
Programador responsable: Richard Tarupí		
Descripción: Creación del diseño de la página principal del sistema, la cuál será de acceso público, en esta pantalla se mostrará un menú, una sección con las familias de normas, una sección con las normas más recientes y un buscador de alto rendimiento.		
Observaciones: <i>Esta pantalla es de acceso público por lo que no es necesario iniciar sesión al sistema</i>		

Historia de Usuario Nro. 2 | Acceso al sistema

Historia de Usuario		
Número:	2	Usuario: Administrador del sistema
Nombre historia: Acceso al sistema por login		
Prioridad en negocio: Alta		Riesgo en desarrollo: Media
Puntos estimados: 5		Sprint asignada: 4
Programador responsable: Richard Tarupí		
Descripción: El acceso al sistema mediante login será para los usuarios de tipo Administrador y Operador, el acceso estará controlado por dos credenciales: el número de cédula del usuario y una contraseña.		
Observaciones: <i>Un usuario puede tener más de un rol de acceso por lo que al iniciar sesión se le presentará una pantalla para seleccionar el rol que desea administrar.</i>		

Historias de Usuario Módulo Administrativo

Historia de Usuario Nro. 3 | Diseño Módulo Administrativo

Historia de Usuario	
Número:	3 Usuario: Administrador
Nombre historia: Diseño Módulo Administrativo	
Prioridad en negocio: Media	Riesgo en desarrollo: Media
Puntos estimados: 3	Sprint asignada: 4
Programador responsable: Richard Tarupí	
Descripción: Creación del diseño visual y la conexión del menú del módulo administrativo, en donde debe existir una pantalla de inicio del módulo y la navegación fácil e intuitiva para el usuario por medio de una barra de menú, en donde se redirija a las pantallas visuales de la gestión de usuarios.	
Observaciones: <i>La página de inicio de este módulo corresponde a la primera pantalla visible después de la selección del rol administrador.</i>	

Historia de Usuario Nro. 4 | Gestión de usuarios

Historia de Usuario	
Número:	4 Usuario: Administrador
Nombre historia: Gestión de usuarios	
Prioridad en negocio: Media	Riesgo en desarrollo: Alta
Puntos estimados: 5	Sprint asignada: 4
Programador responsable: Richard Tarupí	
Descripción: La gestión de usuarios, debe mostrar el listado de todos los usuarios, registro de nuevos usuarios y modificación de datos de usuarios ya registrados. Debe contener los atributos: cédula, apellidos, nombres, sexo, dirección, teléfono, e-mail, rol de usuario, clave y un estado.	
Observaciones: <i>No existe la eliminación de un usuario, ya que a un usuario se lo puede inactivar.</i>	

Historia de Usuario Nro. 5 | Asignación de roles de acceso

Historia de Usuario	
Número:	5 Usuario: Administrador
Nombre historia: Asignación de roles de acceso	
Prioridad en negocio: Media	Riesgo en desarrollo: Alta
Puntos estimados: 5	Sprint asignada: 4
Programador responsable: Richard Tarupí	
Descripción: El administrador en sesión podrá asignar o quitar roles de acceso a los demás usuarios.	
Observaciones: <i>Un usuario puede tener más de un rol de acceso.</i>	

Historias de Usuario Módulo Operativo

Historia de Usuario Nro. 6 | Diseño Módulo Operativo

Historia de Usuario	
Número:	6 Usuario: Operador
Nombre historia: Diseño Módulo Operativo	
Prioridad en negocio: Media	Riesgo en desarrollo: Media
Puntos estimados: 3	Sprint asignada: 4
Programador responsable: Richard Tarupí	
Descripción: Creación del diseño visual y la conexión del menú del módulo operativo, en donde se debe existir una pantalla de inicio del módulo y la navegación fácil e intuitiva para el usuario por medio de una barra de menú, en donde se redirija a las pantallas visuales de la gestión de normas.	
Observaciones: <i>La página de inicio de este módulo corresponde a la primera pantalla visible después de la selección del rol operador.</i>	

Historia de Usuario Nro. 7 | Gestión de normas

Historia de Usuario	
Número:	7 Usuario: Operador
Nombre historia: Gestión de normas	
Prioridad en negocio: Media	Riesgo en desarrollo: Alta
Puntos estimados: 5	Sprint asignada: 4
Programador responsable: Richard Tarupí	
Descripción: La gestión de normas debe contener un listado con las normas registradas en el sistema, en donde el operador puede agregar nuevos registros, editar información de las normas agregadas por el usuario en sesión e inactivar las normas que el operador considere que ya no son relevantes para el sistema. Debe contener los siguientes atributos: código, nombre, título, versión, fecha de publicación, información extra, descripción corta, descripción larga y un estado	
Observaciones: <i>No existe la eliminación de normas, ya que a una norma se la puede inactivar.</i>	

Historia de Usuario Nro. 8 | Incorporación de tags o palabras clave

Historia de Usuario	
Número:	8 Usuario: Operador
Nombre historia: Incorporación de tags o palabras clave	
Prioridad en negocio: Media	Riesgo en desarrollo: Alta
Puntos estimados: 5	Sprint asignada: 5
Programador responsable: Richard Tarupí	
Descripción: Al momento de agregar una nueva norma o bien editar su información, se le permite al operador ingresar tags o palabras claves, las que facilitaran la clasificación de resultados de búsquedas.	
Observaciones: <i>Los tags o palabras claves son opcionales</i>	

Historia de Usuario Nro. 9 | Agregar archivos adjuntos de normas

Historia de Usuario	
Número:	9 Usuario: Operador
Nombre historia: Agregar archivos adjuntos de normas	
Prioridad en negocio: Baja	Riesgo en desarrollo: Alta
Puntos estimados: 5	Sprint asignada: 6
Programador responsable: Richard Tarupí	
Descripción: Al momento de agregar una nueva norma o bien editar su información, se le permite al operador ingresar tags o palabras claves, las que facilitaran la clasificación de resultados de búsquedas.	
Observaciones: <i>Los archivos adjuntos son opcionales</i>	

Historias de Usuario Módulo de Búsquedas

Historia de Usuario Nro. 10 | Buscador de alto rendimiento

Historia de Usuario	
Número:	10 Usuario: Público
Nombre historia: Buscador de alto rendimiento	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alta
Puntos estimados: 8	Sprint asignada: 5
Programador responsable: Richard Tarupí	
Descripción: El buscador estará en la sección principal del sistema (bajo el menú) el cual permitirá que el usuario pueda encontrar las normas que necesite. Para esta búsqueda no es necesario que se escriba el nombre o código de la norma, ya que este buscador estará hecho con Lucene (API de Recuperación de Información) la que permite una búsqueda eficiente usando palabras relacionadas a lo que se busca	
Observaciones: <i>Sin observaciones</i>	

Historia de Usuario Nro. 11 | Presentación de resultados de búsqueda

Historia de Usuario	
Número:	11 Usuario: Público
Nombre historia: Presentación de resultados de búsqueda	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alta
Puntos estimados: 8	Sprint asignada: 5
Programador responsable: Richard Tarupí	
Descripción: El resultado de una búsqueda se presentará en una lista. Cada resultado mostrará el título de la norma y una descripción corta. Los resultados mostrados se deben mostrar en orden de más relevante a menos relevante	
Observaciones: <i>Sin observaciones</i>	

Historia de Usuario Nro. 12 | Presentación de una norma en particular

Historia de Usuario	
Número:	12 Usuario: Público
Nombre historia: Presentación de una norma en particular	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alta
Puntos estimados: 8	Sprint asignada: 5
Programador responsable: Richard Tarupí	
Descripción: Las normas serán presentadas en una página que permita visualizar información detallada de esta. La información que se presentará será: nombre, título, versión, fecha de publicación, descripción, organización a la que pertenece la norma (ISO, IEEE, etc.), familia a la que pertenece en caso de pertenecer a una familia de normas y sus tags o palabras clave	
Observaciones: <i>Sin observaciones</i>	

Historia de Usuario	
Número:	13 Usuario: Público
Nombre historia: Búsquedas recientes	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alta
Puntos estimados: 8	Sprint asignada: 6
Programador responsable: Richard Tarupí	
Descripción: En la página principal habrá una sección en la que se muestre las normas más actuales o recientes, con la finalidad de que el usuario tenga cerca estas normas	
Observaciones: <i>Sin observaciones</i>	

Historia de Usuario	
Número:	14 Usuario: Público
Nombre historia: Presentación de familias de normas	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alta
Puntos estimados: 8	Sprint asignada: 6
Programador responsable: Richard Tarupí	
Descripción: Las familias de normas se deben presentar en una estructura jerárquica en la que se pueda visualizar y navegar entre la familia de normas de una manera más amigable para el usuario.	
Observaciones: <i>Sin observaciones</i>	

2.5.2. Casos de Uso

La Fig. 17 muestra el caso de uso para realizar el acceso al sistema por parte de los diferentes roles de usuario.

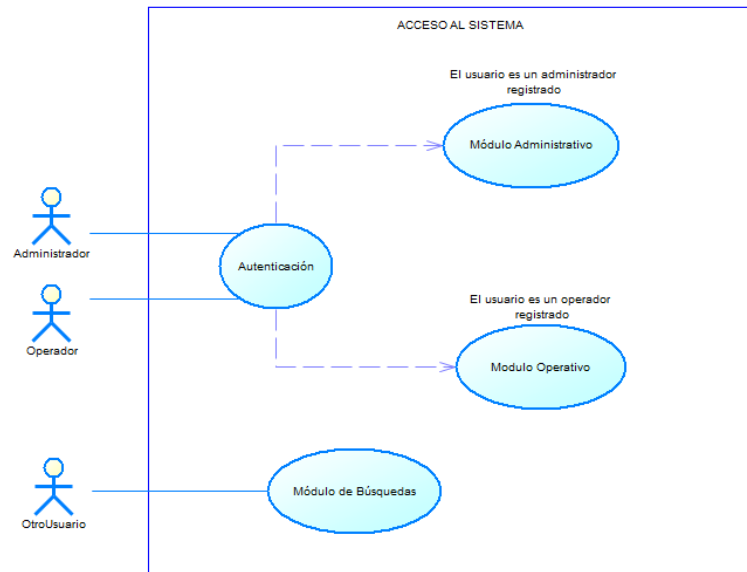


Fig. 17. Caso de Uso 1 – Acceso al Sistema
Fuente: Propia

En la Fig. 18 se presenta el caso de uso para el proceso de gestión de usuarios por parte de los administradores del sistema.

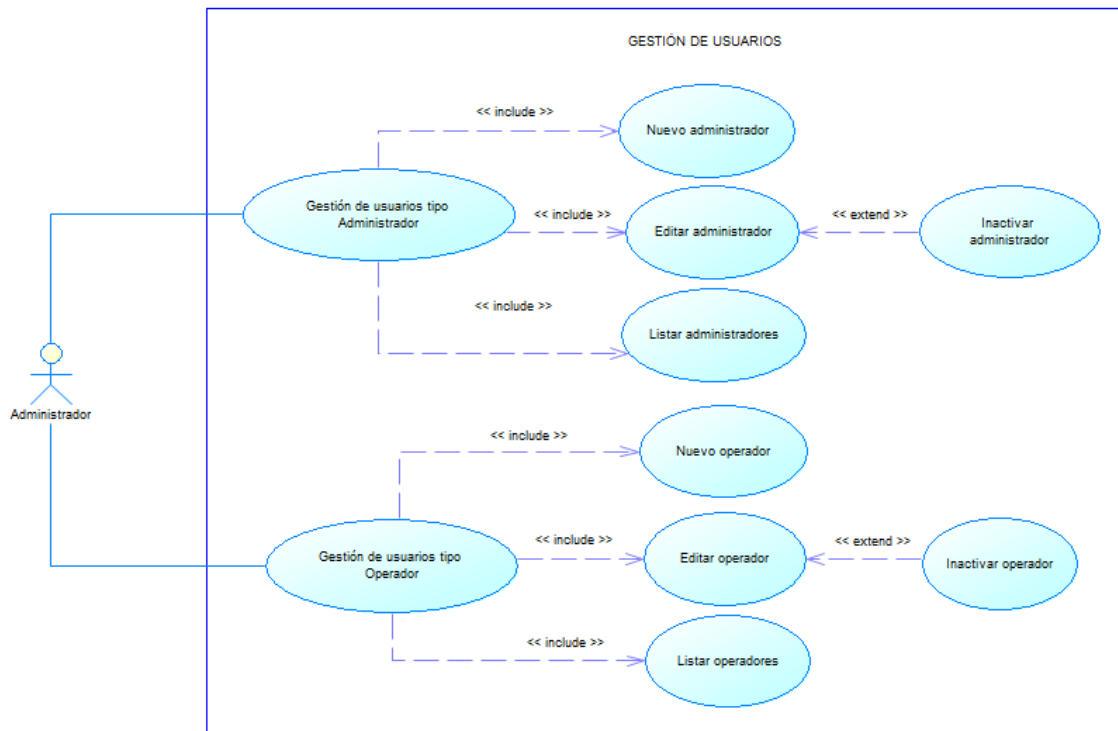


Fig. 18. Caso de Uso 2 – Gestión de usuarios
Fuente: Propia

En la Fig. 19 se presenta el caso de uso correspondiente para el proceso de Gestión de Normas por parte de los Operadores del sistema.

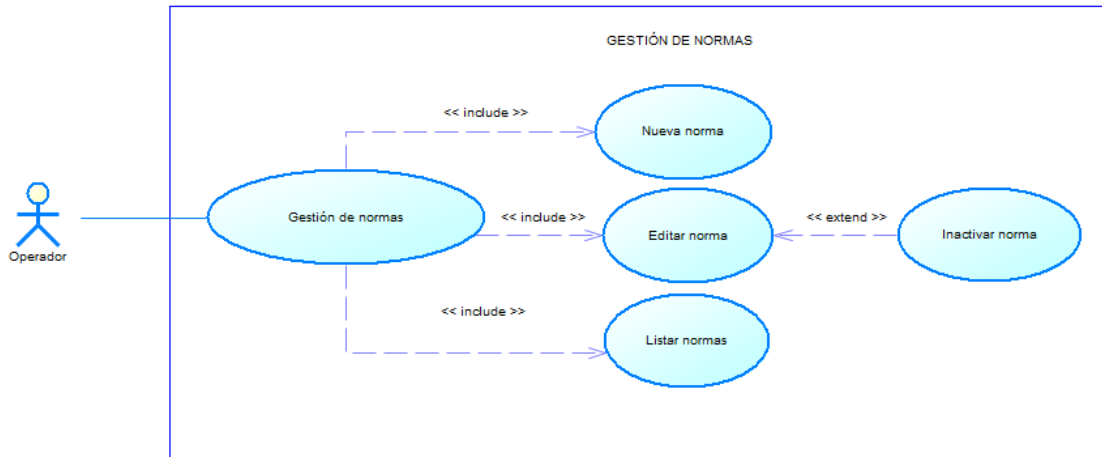


Fig. 19. Caso de Uso 3 – Gestión de Normas
Fuente: Propia

En la Fig. 20 se presenta el caso de uso correspondiente para el proceso de búsqueda de normas realizado por cualquier usuario sin necesidad de tener cuenta de usuario definida.

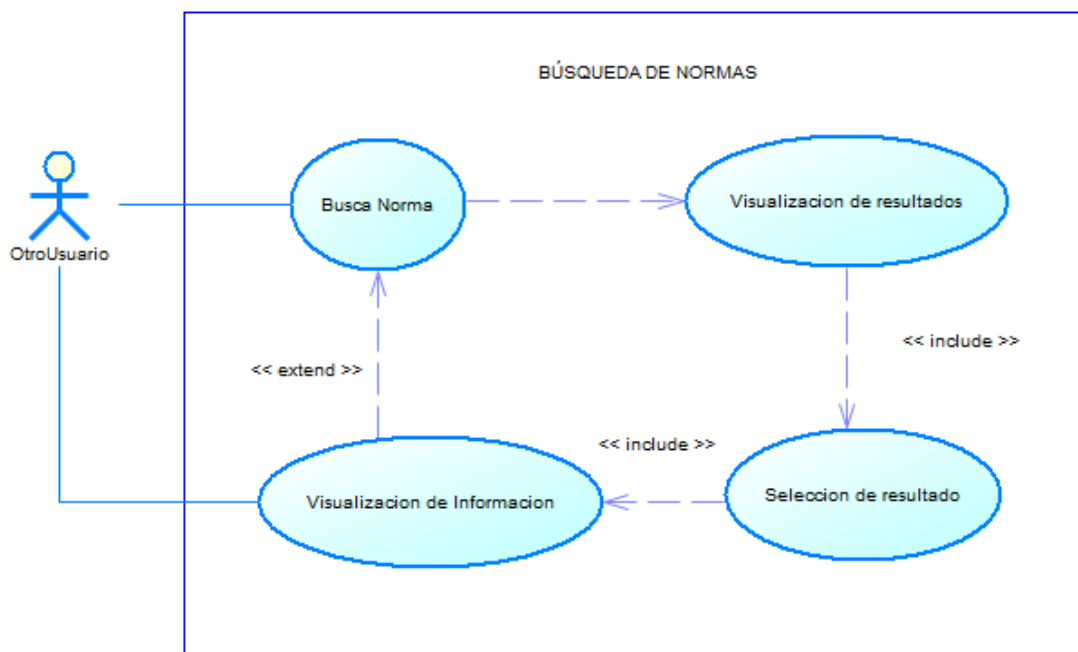


Fig. 20. Caso de Uso 4 – Búsqueda de normas
Fuente: Propia

2.5.3. Diagrama Conceptual

En la Fig. 21 se presenta el diagrama conceptual que se ha diseñado con la finalidad de tener una base robusta para el presente proyecto.

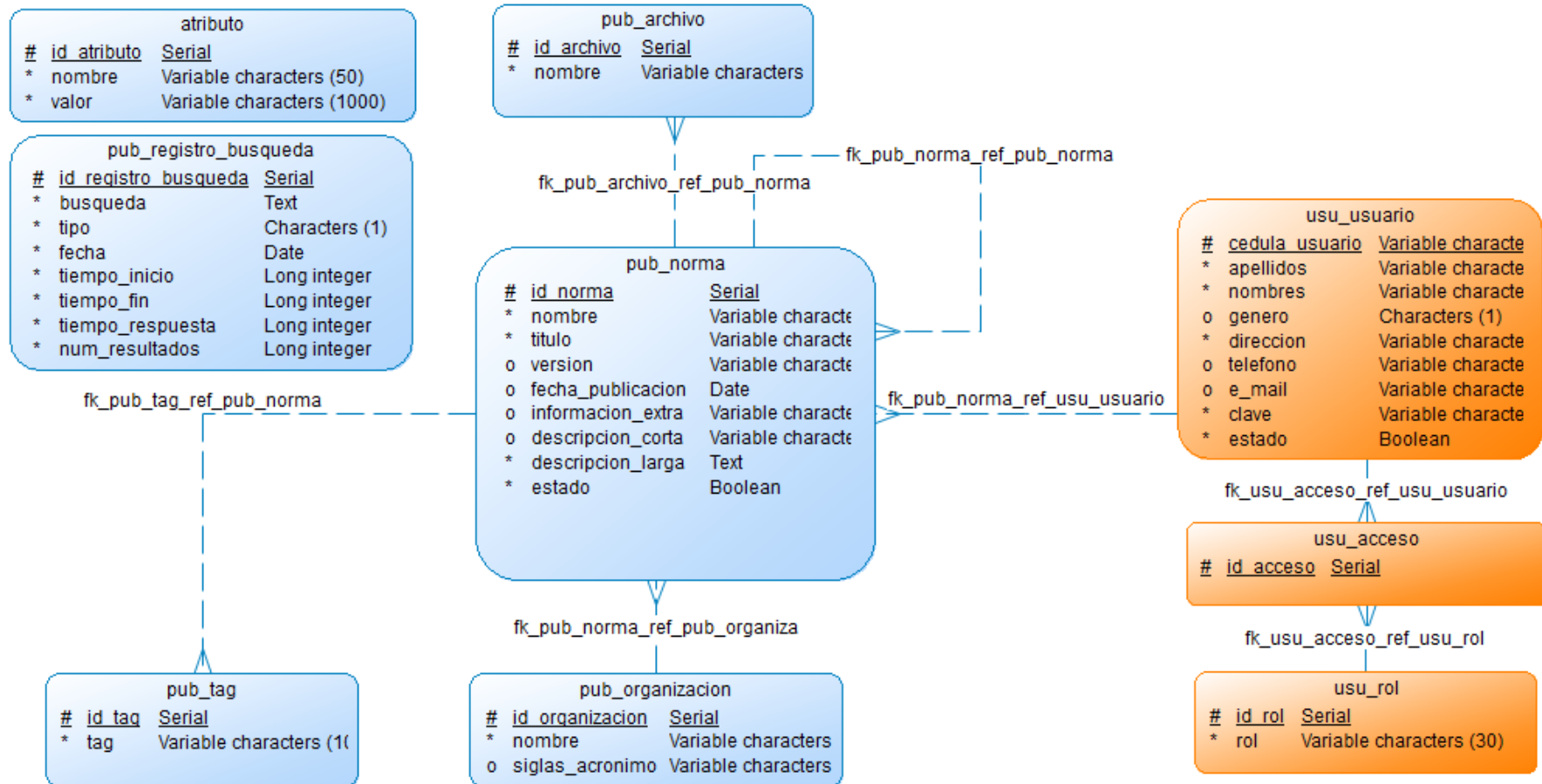


Fig. 21. Diagrama conceptual del sistema
Fuente: Propia

2.5.4. Arquitectura del Software

Lenguajes de Descripción Arquitectónica

DIAGRAMAS UML

- Diagrama de Casos de Uso
- Diagrama Conceptual

Arquitectura de Software

La arquitectura base dentro del proceso es la Arquitectura Monolítica Multicapa Modular, Extensible y Escalable. La arquitectura general del proyecto será la Arquitectura Modelo Vista Controlador (Model – View – Controller).

Arquitectura de Software para el Proyecto:

Herramientas y Tecnologías

- **Entorno de desarrollo:** IDE Eclipse JEE 2018-09
- **Entorno de producción:** Servidor Web WildFly 14.0.1 Final
- **Base de Datos:** PostgreSQL 9.6
- **Sistema Operativo:** Windows 7, 8, 8.1, 10
- **Herramienta Case para manejo de la Base de Datos:** pgAdmin 4
- **Metodología de Desarrollo:** Scrum

Diagrama de la Arquitectura de software

La Fig. 22 muestra el diagrama de la arquitectura del presente proyecto, en la que se puede apreciar la interacción de la aplicación con la API Lucene

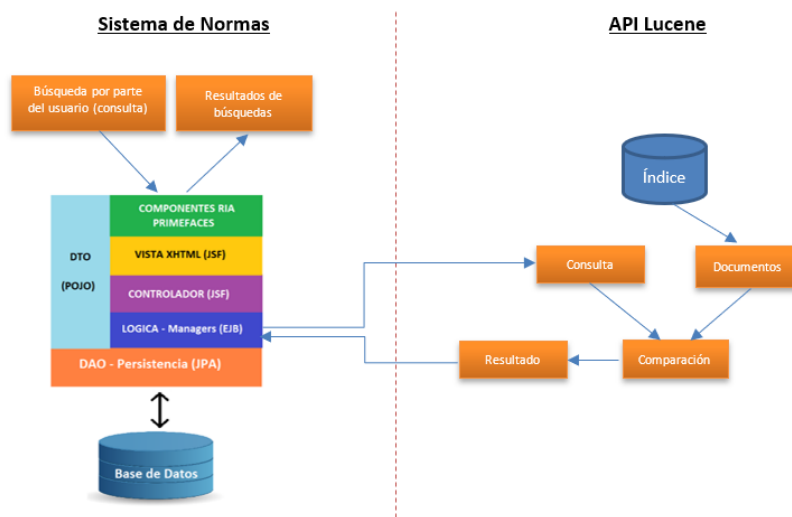


Fig. 22. Diagrama de la arquitectura de software
Fuente: Propia

2.5.5. Wireframe

Previo al desarrollo del sistema se ha realizado un prototipo (wireframe) con la ayuda del software Axure RP 8. En el presente apartado se muestra las interfaces principales del wireframe, para una revisión más detallada revisar el **Anexo B**.

La Fig. 23 muestra el árbol web o estructura del Sistema de Búsqueda de Normas para Ingenierías.

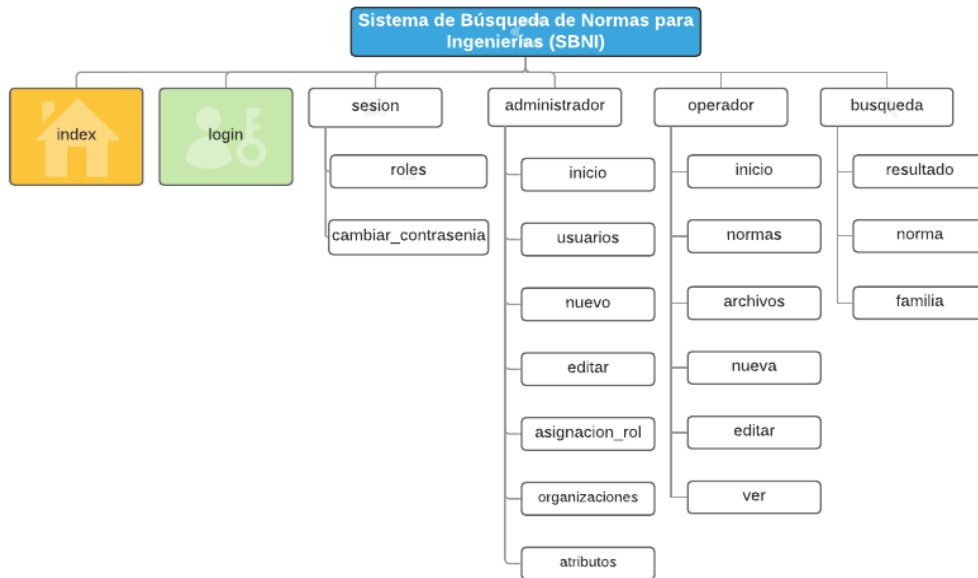


Fig. 23. Árbol de páginas (árbol web) del sistema
Fuente: Propia

Principales Interfaces de usuario

- **index**

En la Fig. 24 se muestra el wireframe de la página principal del sistema.



Fig. 24. Wireframe página index
Fuente: Propia

- **ver (Módulo Operativo)**

En la Fig. 25 se muestra el wireframe de la página ver en la que se muestra la información detallada de una norma en específico

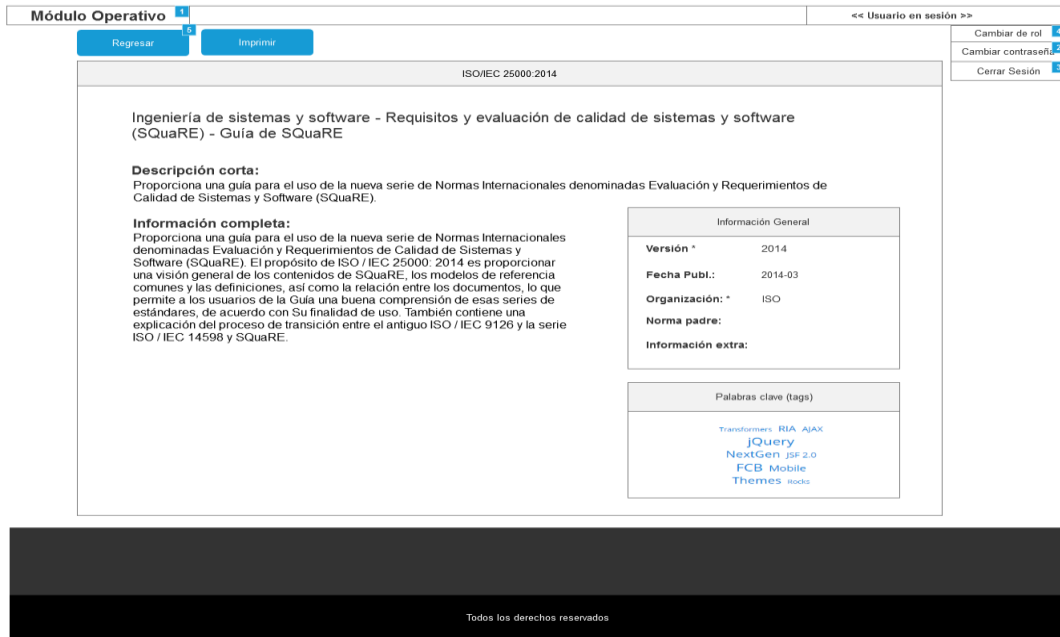


Fig. 25. Wireframe página que muestra información detallada de una norma específica
Fuente: Propia

- **resultado (Módulo de búsquedas)**

En la Fig. 26 se muestra el wireframe de la página de resultados de búsqueda

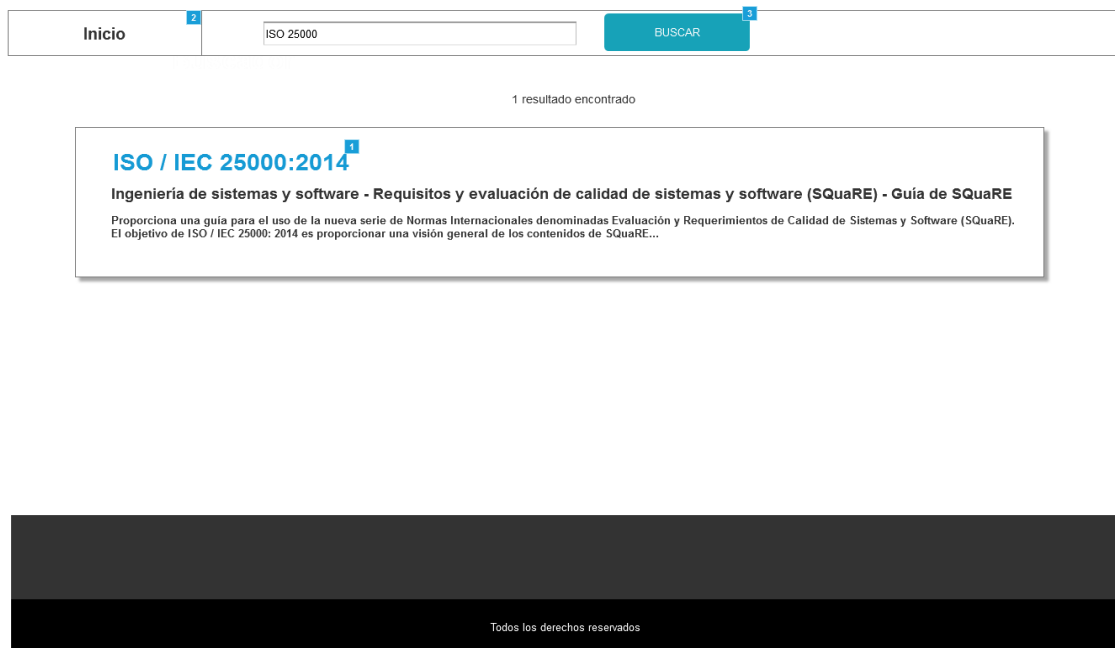


Fig. 26. Wireframe de la página de resultados de búsqueda
Fuente: Propia

2.6. Pruebas de funcionamiento

2.6.1. Plan de Pruebas

El presente plan de pruebas propone como objetivos, identificar los ítems a probar, listar los entregables de las pruebas del proyecto, identificar los tipos de pruebas a utilizar y definir cada una de las pruebas por cada interacción a probar.

Tipos de Pruebas:

- **Pruebas Funcionales**

En las pruebas funcionales se validan los Casos de Uso aprobados, y a partir de ellos se diseñó y ejecutó las pruebas correspondientes.

Módulo Administrativo

En la TABLA 2.1 y la TABLA 2.2 se detalla los objetivos primordiales de las pruebas funcionales del Módulo Administrativo

TABLA 2.1
Objetivo 1 de pruebas funcionales del módulo administrativo

Objetivo de la Prueba	Crear el diseño visual y el menú de navegación del módulo administrativo
Estrategia:	Mediante la implementación de técnicas de frontend, se creó el diseño gráfico del módulo administrativo el cual tiene integrado un menú de navegación y el diseño de la estructura gráfica de las ventanas de este módulo.
Herramientas Requeridas:	Se requiere del servidor WildFly para que el programa se ejecute y mediante un navegador se puede visualizar.
Observaciones:	El diseño visual de este módulo fue una adaptación de un estándar de diseño general de todos los módulos del sistema.

Fuente: Propia

TABLA 2.2
Objetivo 2 de pruebas funcionales del módulo administrativo

Objetivo de la Prueba	Crear la administración de los usuarios
Estrategia:	Mediante la codificación de los métodos para la administración de los usuarios (administradores y operadores) tales como registro, edición, visualización y cambio de estado (activo o inactivo) de los usuarios. Se realizó la codificación según la arquitectura implementada teniendo en cuenta las capas de esta; tales como las entidades, los managers y el controlador que será de intermediario con la vista que será reflejada en una interfaz entendida por el usuario.
Herramientas Requeridas:	Se requiere del servidor WildFly para que el programa se ejecute, el motor de base de datos PostgreSQL y el entorno de desarrollo eclipse.
Observaciones:	La arquitectura del sistema es multicapa

Fuente: Propia

Módulo Operativo

De la TABLA 2.3 a la TABLA 2.5 se detalla los objetivos de las pruebas funcionales del Módulo Operativo

TABLA 2.3
Objetivo 1 de pruebas funcionales del módulo operativo

Objetivo de la Prueba	Crear el diseño visual y la navegación lineal del módulo operativo
Estrategia:	Mediante la implementación de técnicas de frontend, se creó el diseño gráfico del módulo operativo el cual tiene integrado una navegación lineal de recorrido por las diferentes pantallas del formulario.
Herramientas Requeridas:	Se requiere del servidor WildFly para que el programa se ejecute y mediante un navegador se puede visualizar.

Fuente: Propia

TABLA 2.4
Objetivo 2 de pruebas funcionales del módulo operativo

Objetivo de la Prueba	Crear la administración de normas
Estrategia:	Mediante la codificación de los métodos para la administración de normas tales como registro, edición, visualización y cambio de estado (activo o inactivo) de las normas. Se realizó la codificación según la arquitectura implementada teniendo en cuenta las capas de la misma; tales como las entities, los managers y el controlador que será de intermediario con la vista que será reflejada en una interfaz entendida por el usuario.
Herramientas Requeridas:	Se requiere del servidor WildFly para que el programa se ejecute y mediante un navegador se puede visualizar, además del uso de PostgreSQL para la base de datos.

Fuente: Propia

TABLA 2.5
Objetivo 3 de pruebas funcionales del módulo operativo

Objetivo de la Prueba	Implementación de tags o palabras claves en las normas
Estrategia:	En la creación o edición de una norma el operador puede añadir palabras claves las cuales servirán para una clasificación al momento de buscar una norma u obtener resultados filtrados por un tag en específico.
Herramientas Requeridas:	Se requiere del servidor WildFly para que el programa se ejecute y mediante un navegador se puede visualizar, además del uso de PostgreSQL para la base de datos.
Observaciones:	Las palabras claves serán mostradas tanto en el módulo operativo como en el módulo de búsquedas

Fuente: Propia

Módulo de Búsquedas

De la TABLA 2.6 a la TABLA 2.8 se detalla los objetivos de las pruebas funcionales del Módulo de Búsquedas

TABLA 2.6
Objetivo 1 de pruebas funcionales del módulo de búsquedas

Objetivo de la Prueba	Crear el diseño visual y la navegación lineal del módulo de Búsquedas
Estrategia:	Mediante la implementación de técnicas de prototipado, se creó el diseño gráfico del módulo de búsquedas el cual tiene integrado una navegación lineal de recorrido con las diferentes secciones que el módulo tiene
Herramientas Requeridas:	Se requiere del servidor WildFly para que el programa se ejecute y mediante un navegador se puede visualizar.
Observaciones:	El módulo de búsquedas es público

Fuente: Propia

TABLA 2.7
Objetivo 2 de pruebas funcionales del módulo de búsquedas

Objetivo de la Prueba	Crear sección de búsquedas de alto rendimiento
Estrategia:	Para satisfacer el proceso de búsquedas se utiliza la API Lucene que permite búsquedas de alto rendimiento. La codificación de este proceso se acopla a la arquitectura del sistema, obteniendo los métodos de indexación y búsqueda.
Herramientas Requeridas:	Se requiere del servidor WildFly para que el programa se ejecute y mediante un navegador se puede visualizar, además del uso de PostgreSQL para la base de datos.
Observaciones:	El usuario puede realizar búsquedas desde la página principal

Fuente: Propia

TABLA 2.8
Objetivo 3 de pruebas funcionales del módulo de búsquedas

Objetivo de la Prueba	Implementación de estructura gráfica para mostrar familias de normas
Estrategia:	El sistema muestra una sección donde se pueda ver las familias de normas más importantes en una estructura gráfica como árbol, para mejorar la presentación visual y una mayor organización
Herramientas Requeridas:	Se requiere del servidor WildFly para que el programa se ejecute y mediante un navegador se puede visualizar, además del uso de PostgreSQL para la base de datos.
Observaciones:	Es posible navegar por las familias de normas

Fuente: Propia

- **Pruebas de Seguridad**

De la TABLA 2.9 a la TABLA 2.11 se detalla los objetivos primordiales de las pruebas de seguridad.

TABLA 2.9
Objetivo 1 de pruebas de seguridad

Objetivo de la Prueba	Iniciar sesión a usuarios previamente registrados por un administrador del sistema
Estrategia:	Realización de un control de inicio de sesión que permita verificar en la base de datos que un usuario y una clave existan y coincidan con los criterios de ingreso, por medio de técnicas de validación las cuales se conectan en las diferentes capas del sistema (persistencia, manager, Controller, vista).
Herramientas Requeridas:	Entorno de desarrollo Eclipse Oxygen, base de datos PostgreSQL, navegador
Observaciones:	Para iniciar sesión, el usuario debe estar activo

Fuente: Propia

TABLA 2.10
Objetivo 2 de pruebas de seguridad

Objetivo de la Prueba	Acceso de usuarios a funciones del sistema específicas según el rol de usuario
Estrategia:	El sistema controla que, según el rol asignado, se usen funciones específicas, cada usuario tiene la capacidad de usar un módulo y cada módulo tiene diferente funcionalidad; Módulo administrativo – usuario administrador Módulo operativo – usuario operador Módulo de búsquedas – público
Herramientas Requeridas:	Entorno de desarrollo Eclipse 2018-09, base de datos PostgreSQL, navegador
Observaciones:	Ninguna

Fuente: Propia

TABLA 2.11
Objetivo 2 de pruebas de seguridad

Objetivo de la Prueba	Controlar acceso al sistema
Estrategia:	Realización de un control acceso al sistema, donde se controla que la navegación en el sistema sea solo por usuarios con inicio de sesión válida, esto permite que no se ingrese al sistema por medio de direcciones (links) específicos.
Herramientas Requeridas:	Entorno de desarrollo Eclipse 2018-09, base de datos Postgres, navegador
Observaciones:	Por medio de la obtención de links, terceros podrían acceder al sistema sin iniciar sesión, ante esto el sistema controla en cada página que el inicio de sesión sea válido

Fuente: Propia

Recurso Humano del Plan de Pruebas

En la TABLA 2.12 se muestra el recurso humano y el perfil que desempeña cada uno para el plan de pruebas.

TABLA 2.12
Recurso humano del plan de pruebas

NOMBRES	PERFIL DEL RECURSO HUMANO
Richard Fernando Tarupí	Desarrollador – Estudiante UTN
Ing. Mauricio Rea	Tester - Docente UTN
Ing. Alexander Guevara	Tester - Docente UTN

Fuente: Propia

2.6.2. Informe de Plan de Pruebas

- **Pruebas del Sistema**

En la TABLA 2.13 se muestra el resultado de las pruebas del sistema.

TABLA 2.13
Resultado de las pruebas del sistema

Caso de Uso	<Identificador del Caso de uso>	<Número total de casos de prueba ejecutados de acuerdo al escenario>
Acceso al Sistema	Caso de Uso – Gestión 1	3
Administración de usuarios	Caso de Uso – Gestión 2	5
Administración de Normas	Caso de Uso – Gestión 3	5
Búsqueda de Normas	Caso de Uso – Gestión 4	5

Fuente: Propia

- **Pruebas de interfaz de Usuario**

En la TABLA 2.14 se muestra el resultado de las pruebas de interfaz de usuario el que se verá reflejado en el siguiente informe o lista de chequeo.

TABLA 2.14
Resultado de las pruebas de interfaz de usuario

Elemento a Revisar	SI	NO	No Aplica	Observaciones
Ubicación estratégica de los menús de navegación	X			S/O
Estructura organizativa de las funciones principales del cada módulo	X			S/O
Estructura organizativa de las funciones secundarias de cada módulo	X			Defectos corregidos
Adaptabilidad visual y contraste de colores	X			S/O
Botones identificativos según la funcionalidad que ejecutan	X			S/O

Fuente: Propia

- **Pruebas de Seguridad**

En la TABLA 2.15 se muestra el resultado de las pruebas de seguridad el que se verá reflejado en el siguiente informe o lista de chequeo:

TABLA 2.15
Resultado de las pruebas de seguridad

Elemento a Revisar	SI	NO	No Aplica	Observaciones
Iniciar sesión a usuarios previamente registrados por un administrador del sistema	X			S/O
Acceso de usuarios a funciones del sistema específicas según el rol de usuario	X			S/O
Controlar acceso al sistema	X			Con defectos ya corregidos

Fuente: Propia

2.7. Implementación de la solución

El Sistema de Búsqueda de Normas para Ingenierías (SBNI de aquí en adelante) ha sido implementado en un servidor virtual de la Carrera de Ingeniería en Sistemas Computacionales, el cuál es de acceso público para todo docente y estudiante tanto de la carrera como de la Facultad de Ingenierías en Ciencias Aplicadas, e incluso permite el acceso desde cualquier facultad de la Universidad Técnica del Norte.

El SBNI posteriormente a su implementación ha sido presentado y probado por un grupo de estudiantes y docentes de la Carrera de Ingeniería de Software ya que la herramienta será utilizada inicialmente por ellos y por el resto de facultades de la universidad en un futuro.

CAPÍTULO 3

Validación

3.1. Análisis e Interpretación de resultados

Para el análisis e interpretación de resultados del SBNI se pidió la colaboración de tres cursos de estudiantes de la Carrera de Ingeniería en Sistemas Computacionales e Ingeniería de Software, a fin de realizar búsquedas reales que permitan tomar tiempos de respuesta precisos al estar el sistema en un ámbito de producción. Los tiempos de respuesta obtenidos se analizaron en base a la subcaracterística de Comportamiento Temporal de la norma ISO 25010, ya que esta señala que el tiempo de respuesta es de gran importancia para el aseguramiento de la calidad en un producto de software y ante ello se ha decidido realizar una comparativa entre los dos métodos de búsqueda, mediante los índices de Lucene y mediante una búsqueda por TAGS usando la tecnología de **JPQL**¹² propia de Java Enterprise.

Para los presentes resultados se ingresaron 150 normas referentes a tópicos de ingenierías y tecnologías de la información. Además, se preparó un banco de palabras sugeridas para que los estudiantes puedan realizar búsquedas más centralizadas.

Es importante hacer mención que la unidad de tiempo en la que se presenta los tiempos de consulta es en segundos y fue tomada usando el reloj del sistema a una precisión de milisegundos.

Además, se utiliza técnicas de estadística descriptiva con la finalidad de calcular la media, conocida como el valor promedio de las muestras y la desviación estándar que representa que tan dispersos están los datos con respecto a la media, a fin de analizar la eficiencia que los dos métodos de búsqueda proporcionan al usuario.

¹² Java Persistence Query Language (JPQL) es un lenguaje de consultas orientado a objetos en la cual se realiza consultas sobre las entidades que se encuentran en una base de datos relacional.

3.1.1. Resultados obtenidos con búsquedas en los índices de Lucene

- **Búsquedas con una sola palabra**

En la TABLA 3.1 se muestra los resultados del tiempo de respuesta y cantidad de resultados obtenidos, de las búsquedas de una sola palabra usando el índice de Lucene.

TABLA 3.1
Tiempos de respuesta y resultados en búsquedas de una sola palabra con Lucene

Palabra(s)	Repeticiones	Tiempo de respuesta (segundos)	N° resultados
calidad	155	0,026	28
información	54	0,015	52
seguridad	90	0,004	14
sistemas	57	0,023	54
software	74	0,019	25
tecnología	27	0,010	40
MEDIA		0,016	
DESV.		0,008	
ESTANDAR			

Fuente: Propia

- **Búsquedas con dos palabras**

En la TABLA 3.2 se muestra los resultados del tiempo de respuesta y cantidad de resultados obtenidos de las búsquedas con dos palabras usando el índice de Lucene.

TABLA 3.2
Tiempos de respuesta y resultados en búsquedas de dos palabras con Lucene

Palabra(s)	Repeticiones	Tiempo de respuesta (segundos)	N° resultados
calidad	35	0,021	37
software			
modelos	63	0,021	38
calidad			
producto	56	0,024	59
calidad			
seguridad	31	0,014	24
informática			
sistemas	48	0,022	55
informáticos			
MEDIA		0,020	
DESV.		0,004	
ESTANDAR			

Fuente: Propia

- **Búsquedas con tres palabras**

En la TABLA 3.3 se muestra los resultados del tiempo de respuesta y cantidad de resultados obtenidos de las búsquedas con tres palabras usando el índice de Lucene.

TABLA 3.3
Tiempos de respuesta y resultados en búsquedas de tres palabras con Lucene

Palabra(s)	Repeticiones	Tiempo de respuesta (segundos)	N° resultados
base de datos	61	0,012	30
calidad de software	37	0,020	37
calidad software sistemas	8	0,025	69
modelos calidad software	19	0,023	46
sistemas gestión seguridad	20	0,023	60
técnicas de seguridad informática	50	0,015	56
MEDIA		0,020	
DESV. ESTANDAR		0,005	

Fuente: Propia

En la TABLA 3.4 se muestra los resultados generales del tiempo de respuesta con sus respectivos cálculos de la media y su desviación estándar con el buscador de Lucene.

TABLA 3.4
Media de tiempos de respuesta con desviación estándar en búsquedas con Lucene

N° palabras	Repeticiones	MEDIA Tiempo de respuesta (segundos)	DESV. ESTANDAR
UNA PALABRA	76	0,0161	0,008
DOS PALABRAS	47	0,0202	0,004
TRES PALABRAS	33	0,0196	0,005

Fuente: Propia

En la Fig. 27 se presenta el gráfico de los resultados de las búsquedas generales utilizando el buscador de Lucene, junto a la desviación estándar. En este gráfico se puede apreciar que la desviación estándar o dicho de mejor manera, que la dispersión de los datos con respecto a su media va estabilizándose con búsquedas de más palabras, es decir, que el tiempo de respuesta con Lucene no va a variar significativamente con búsquedas de varias palabras, siendo uno de los objetivos del presente proyecto que busca recuperar información de manera eficiente. También se puede apreciar que la línea de tendencia de los tiempos de respuesta es logarítmica, es decir, que su crecimiento será leve.

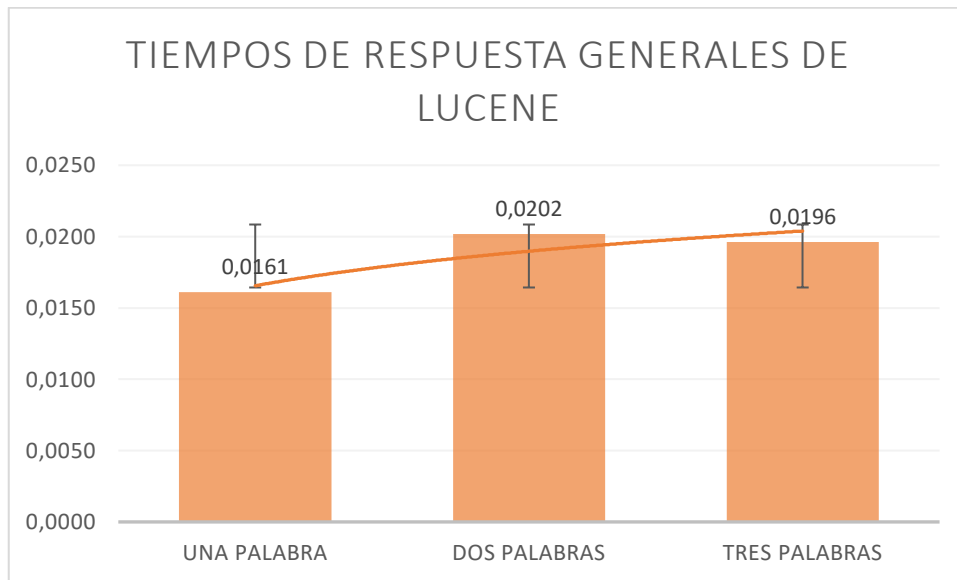


Fig. 27. Resultados generales de búsquedas con Lucene
Fuente: Propia

En la TABLA 3.5 se presenta la cantidad de resultados obtenidos con búsquedas de una, dos y tres palabras respectivamente, usando el buscador de LUCENE.

TABLA 3.5
Cantidad de resultados en búsquedas con Lucene

Nº palabras	Nº de búsquedas	Resultados obtenidos
UNA PALABRA	457	213
DOS PALABRAS	233	213
TRES PALABRAS	195	298

Fuente: Propia

3.1.2. Resultados obtenidos con búsquedas en tags con consulta JPQL

- **Búsquedas con una sola palabra**

En la TABLA 3.6 se muestra los resultados del tiempo de respuesta y cantidad de resultados obtenidos de las búsquedas de una sola palabra usando consultas JPQL en los tags del sistema.

TABLA 3.6
Tiempos de respuesta y resultados en búsquedas de una sola palabra con JPQL en tags del sistema

Palabra(s)	Repeticiones	Tiempo de respuesta (segundos)	N° resultados
calidad	24	0,012	13
información	6	0,010	25
seguridad	20	0,006	9
sistemas	7	0,011	27
software	10	0,014	23
tecnología	15	0,013	26
MEDIA		0,011	
DESV.		0,003	
ESTANDAR			

Fuente: Propia

- **Búsquedas con dos palabras**

En la TABLA 3.7 se muestra los resultados del tiempo de respuesta y cantidad de resultados obtenidos de las búsquedas de dos palabras usando consultas JPQL en los tags del sistema.

TABLA 3.7
Tiempos de respuesta y resultados en búsquedas de dos palabras con JPQL en tags del sistema

Palabra(s)	Repeticiones	Tiempo de respuesta (segundos)	N° resultados
calidad software	5	0,020	26
modelos calidad	9	0,010	13
producto calidad	3	0,019	34
tecnología informática	4	0,014	26
sistemas informáticos	3	0,009	27
MEDIA		0,014	
DESV.		0,005	
ESTANDAR			

Fuente: Propia

- **Búsquedas con tres palabras**

En la TABLA 3.8 se muestra los resultados del tiempo de respuesta y cantidad de resultados obtenidos de las búsquedas de dos palabras usando consultas JPQL en los tags del sistema.

TABLA 3.8
Tiempos de respuesta y resultados en búsquedas de tres palabras con JPQL en tags del sistema

Palabra(s)	Repeticiones	Tiempo de respuesta (segundos)	N° resultados
base de datos	4	0,021	70
calidad de software	6	0,024	69
calidad software sistemas	3	0,017	38
modelos calidad software	4	0,019	26
sistemas gestión seguridad	6	0,023	30
técnicas seguridad informática	2	0,009	13
MEDIA		0,019	
DESV. ESTANDAR		0,006	

Fuente: Propia

En la TABLA 3.9 se muestra los resultados generales del tiempo de respuesta con sus respectivos cálculos de la media y su desviación estándar con el buscador de JPQL en los tags del sistema.

TABLA 3.9
Media de tiempos de respuesta con desviación estándar en búsquedas con Lucene

N° palabras	Repeticiones	MEDIA Tiempo de respuesta (segundos)	DESV. ESTANDAR
UNA PALABRA	14	0,011	0,003
DOS PALABRAS	5	0,014	0,005
TRES PALABRAS	4	0,019	0,006

Fuente: Propia

En la Fig. 28 se presenta el gráfico de los resultados de las búsquedas generales utilizando el buscador de Lucene, junto a la desviación estándar. En este gráfico se puede apreciar que la desviación estándar o dispersión de los datos con respecto a la media es irregular, y esto se debe al rápido crecimiento que los tiempos de respuesta tienen entre más palabras se añaden a las búsquedas. También se puede apreciar que la línea de tendencia de los tiempos de respuesta es exponencial, es decir, que su crecimiento dependerá en gran parte de las palabras que se ingrese en la búsqueda, dicho de otra manera, entre más palabras tenga una búsqueda mayor será el tiempo de respuesta.

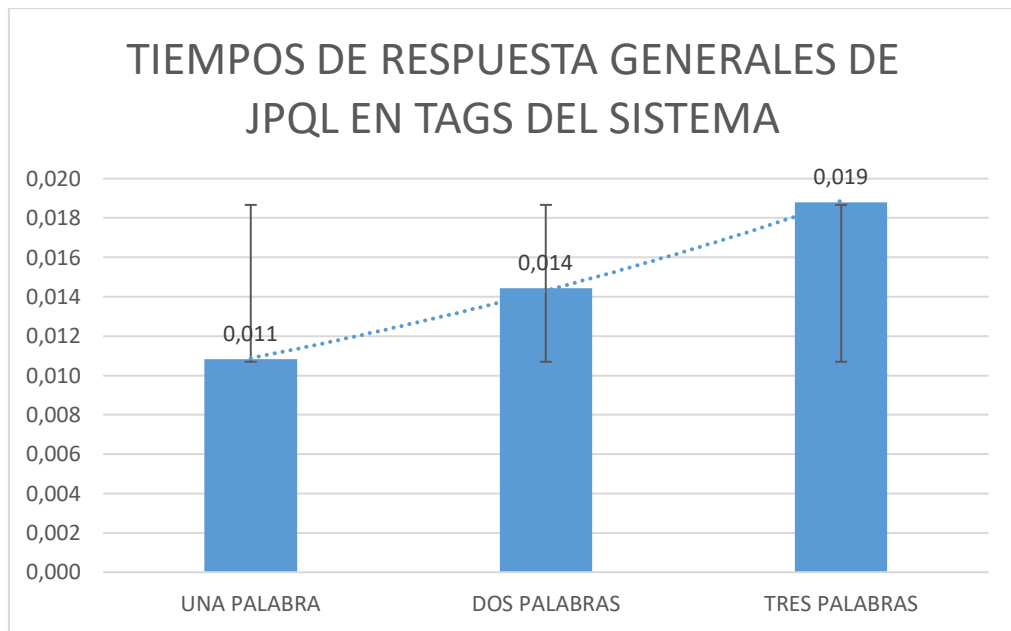


Fig. 28. Resultados generales de búsquedas con JPQL en tags del sistema
Fuente: Propia

En la TABLA 3.10 se presenta la cantidad de resultados obtenidos con búsquedas de una, dos y tres palabras respectivamente, usando el buscador de JPQL en los tags del sistema.

TABLA 3.10
Cantidad de resultados en búsquedas con JPQL en tags del sistema

Nº palabras	Nº de búsquedas	Resultados obtenidos
UNA PALABRA	82	123
DOS PALABRAS	24	126
TRES PALABRAS	25	246

Fuente: Propia

3.1.3. Comparativa de resultados entre Lucene y consulta JPQL en tags del sistema.

En la Fig. 29 se presenta el gráfico comparativo de los tiempos de respuesta con una, dos y tres palabras utilizando los diferentes buscadores, en este gráfico como anteriormente se ha descrito se aprecia el crecimiento en los tiempos de respuesta que presenta cada método de búsqueda, siendo Lucene el más eficiente por no presentar crecimiento significativo al ingresar más palabras a una búsqueda. Por el contrario, a pesar de que los tiempos de respuesta del buscador de JPQL sobre los tags del sistema son más bajos que los obtenidos con Lucene, el incremento de estos es exponencial siendo no eficiente al momento de realizar búsquedas de varias palabras sin mencionar que el buscador de JPQL en los tags no es capaz de realizar todo el procesamiento de documentos que Lucene nos proporciona.

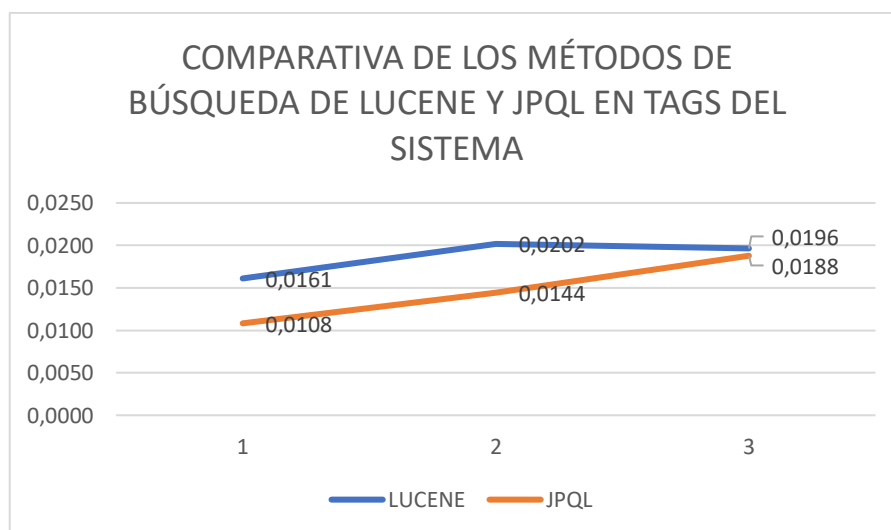


Fig. 29. Comparativa de los buscadores con Lucene y TAGS
Fuente: Propia

A los resultados anteriormente presentados se puede agregar que Lucene muestra los resultados por orden de relevancia y que realmente cumplen con los criterios de búsqueda, al contrario de una consulta JPQL sobre los tags de las normas que no puede mostrar los resultados por orden de relevancia y muestra datos que pueden no ser relevantes para el usuario, debido a que varias normas pueden tener tags que no correspondan a su área de aplicación, lo que hace que la búsqueda no sea la deseada.

Al culminar esta etapa de pruebas se notó en los estudiantes un mayor interés por el buscador de LUCENE, ya que con este los estudiantes realizaron un total de 885 búsquedas, mientras que con el buscador mediante TAGS solamente se realizaron 131 búsquedas.

3.2. Análisis de Impactos

El presente sistema presentó buenos resultados en cuanto a eficiencia en el tiempo de respuesta, ya que muestra resultados de manera rápida (tiempo de respuesta medio de 0.019 segundos en 885 búsquedas), sin embargo, el buscador mediante TAGS presentó un tiempo menor (tiempo de respuesta medio de 0,015 en 131 búsquedas). Sin embargo, el crecimiento de la línea de tendencia (logarítmica) del buscador de Lucene es más eficiente que la de JPQL (exponencial), lo cual es de gran importancia en búsquedas que necesiten un criterio de búsqueda mas completo.

Los factores que respaldan la diferencia en los tiempos de respuesta de los dos métodos de búsqueda están relacionados a los beneficios que el buscador de LUCENE proporciona al usuario y que sin embargo el buscador de TAGS no es capaz de realizar, estos beneficios están relacionados al procesamiento de la información, tales como, la limpieza de palabras vacías, palabras muy frecuentes, palabras muy poco frecuentes, palabras que no son relevantes para el usuario, entre otras. Además, el buscador con LUCENE permite ordenar los resultados de manera que será presentado primero lo más relevante para un criterio de búsqueda, mientras que el buscador mediante TAGS que utiliza consultas de JPQL simplemente muestra los datos que cumplen exactamente con el criterio de búsqueda sin realizar ningún tipo de procesamiento ni cálculos de relevancia, omitiendo de esta manera información que podría ser de gran beneficio para los usuarios.

Para más información sobre la manera en la que Lucene realiza el procesamiento de información y recuperación de información sírvase leer el ejemplo presentado en el Modelo Vectorial de la sección 1.2.3 (Modelos de Recuperación de Información) y la Fig. 5 (Procesamiento de documentos para RI).

De igual manera es importante reiterar que el buscador de Lucene presentó mayor número de resultados (media de 241 resultados en 885 búsquedas realizadas) en comparación al buscador mediante TAGS que presentó menor cantidad de resultados (media de 165 resultados en 131 búsquedas), lo cual es importante al momento de necesitar alternativas a un criterio de búsquedas. Para mayor detalle sírvase revisar el punto 3.1 del presente capítulo.

CONCLUSIONES

- La Recuperación de Información (RI) juntamente con sus modelos de RI permite a los sistemas en entornos web integrar características de un motor de búsquedas, como la indexación de términos y el procesamiento de información permitiendo la optimización del proceso de búsquedas para los usuarios, presentando resultados más relevantes con respecto a las consultas realizadas.
- El presente sistema web desarrollado con tecnologías Java EE ha permitido comprobar la fácil adaptación que tienen estas tecnologías con la API Lucene la que se encarga de la indexación y Recuperación de Información dentro del Sistema de Búsquedas de normas y estándares más utilizados en las áreas de ingeniería.
- La API de Recuperación de Información Lucene ha permitido comprender la importancia que tiene el orden de relevancia de resultados que deben ser presentados al usuario, ya que una búsqueda tradicional no es capaz de ordenar resultados por relevancia sino simplemente se encarga de mostrar resultados a dichas búsquedas.
- Existe una gran variedad de normas y estándares que son utilizados con frecuencia en las áreas de ingeniería, por lo que el presente proyecto se acopló perfectamente a las necesidades de los usuarios, permitiendo obtener resultados que no se basen en la precisión sino a la presentación de resultados más relevantes a sus consultas.
- Mediante la validación de resultados usando la subcaracterística de Comportamiento Temporal de la Norma ISO/IEC 25010, sobre el presente proyecto se ha llegado a comprobar que el tiempo de respuesta de las búsquedas usando la API Lucene es mayor a una búsqueda tradicional debido a que Lucene proporciona grandes beneficios de procesamiento de consultas y presentación de resultados, beneficios que una consulta tradicional no es capaz de realizar y es un factor de gran importancia dentro de los sistemas de recuperación de información, como es el caso del presente.

RECOMENDACIONES

- El presente proyecto ha permitido apreciar los beneficios que la Recuperación de Información puede proporcionar a nuestros sistemas, por consiguiente, es recomendable seguir investigando sobre RI, ya que es un área que abarca muchos más de los beneficios estudiados con el presente.
- Es recomendable continuar con el estudio de la API Lucene, ya que incorpora funciones que pueden ser de gran utilidad en sistemas que manejan grandes cantidades de información, y en el presente proyecto no se ha logrado realizar un estudio más minucioso de esta potente herramienta.
- El presente estudio ha permitido generar más interés en estas API de RI por lo que es de importancia recomendar el estudio de otras API que permitan búsquedas de alto desempeño, para de esta manera tener un amplio conocimiento sobre estas y poder realizar comparativas que ayuden ampliar la perspectiva de los beneficios que pueden generar sobre nuestros sistemas de búsquedas futuros.
- Con respecto a la eficiencia de las búsquedas es recomendable continuar ingresando información de más normas, para de esta manera continuar estudiando la eficiencia y los tiempos de respuesta que esta herramienta puede generar sobre una base de datos con gran cantidad de datos.
- Como recomendación final es importante mencionar que para un futuro estudio se podría desarrollar un sistema similar al presente, pero con el uso de bases de datos NoSQL, y así analizar el comportamiento que se obtiene, ya que en el estudio realizado se ha encontrado información referente a la efectividad que la API Lucene podría tener sobre bases de datos de este tipo.

GLOSARIO DE TÉRMINOS

- **Recuperación de información:** Recuperación de Información o RI hace referencia al área que consiste en aplicar un conjunto de métodos y técnicas para buscar y recuperar información de manera eficiente y que sea relevante para las necesidades de información.
- **Indexación:** Indexar o Indizar se le conoce al proceso de ordenamiento de una serie de datos o información de acuerdo con un criterio en común a todos los datos, con la finalidad de facilitar la consulta y análisis de estos.
- **CEPAL:** Una de las cinco comisiones regionales de las Naciones Unidas.
- **Lematización:** La lematización es un proceso lingüístico que tiene como finalidad hallar el lema correspondiente dada una forma flexionada. Llámesele lema a la forma que se acepta como representante de un conjunto de formas flexionadas de una misma palabra.
- **Fichero inverso:** Proviene de la acción de indizar en donde se almacenan las relaciones existentes entre los términos y los documentos de los que se realiza la indización.
- **Equiparación:** Establece lo que tiene en común entre la búsqueda del usuario y las representaciones de los documentos.
- **Aplicaciones distribuidas:** Aplicaciones compuestas por diferentes componentes que se ejecutan en entornos separados, las que generalmente se encuentran en plataformas diferentes conectadas en red.
- **Fichero inverso:** Proviene de la acción de indizar en donde se almacenan las relaciones existentes entre los términos y los documentos de los que se realiza la indización.
- **Equiparación:** Establece lo que tiene en común entre la búsqueda del usuario y las representaciones de los documentos.
- **Sprint:** Iteración de tiempo en la que el equipo trabaja para convertir las historias de usuario en una versión del producto totalmente operativo.
- **SW-CMM:** El Modelo de Madurez de Capacidades (CMM) es un modelo de evaluación a los procesos que realizados internamente por una organización.
- **Redes de área local (LAN):** Una red LAN es una red de computadoras que tiene un área o alcance reducido, como una casa o un edificio y que es usada para compartir recursos como una impresora o incluso internet.

- **Redes de área metropolitana:** Una red MAN es una red de alta velocidad que puede dar cobertura a un área geográfica extensa permitiendo la integración de servicios mediante la transmisión de voz y video.
- **JPQL:** Java Persistence Query Language (JPQL) es un lenguaje de consultas orientado a objetos en la cual se realiza consultas sobre las entidades que se encuentran en una base de datos relacional.

BIBLIOGRAFIA

- Alonso, M. A., & Vilares, J. (s. f.). *Recuperación de Información en Internet. Tema 3: Principios de Recuperación de Información*. Recuperado de http://www.grupocole.org/docencia/rrii/RII_3.pdf
- Antonio, J., & Comeche, M. (s. f.). *Los modelos clásicos de Recuperación de información y su vigencia*. Recuperado de <http://www.internetworldstats.com/emarketing.htm>.
- Bordignon, F., & Tolosa Chacón, G. (2007). Recuperación de información: un área de investigación en crecimiento. *Ciencias de la Información*, 38(1-2), 13-24. Recuperado de <http://www.redalyc.org/articulo.oa?id=181414865002>
- Cadavid, A. N., Martínez, J. D. F., & Vélez, J. M. (2013). Revisión de metodologías ágiles para el desarrollo de software. *Prospectiva*, 11(2), 30-39.
- Carrazana, E. (2014). *Modelo Espacio Vectorial*. Recuperado de https://www.researchgate.net/publication/274074252_Modelo_Espacio_Vectorial
- Cordina, L. (1995). Teoría de recuperación de información: modelos fundamentales y aplicaciones a la gestión documental. Recuperado 29 de abril de 2019, de http://www.elprofesionaldelainformacion.com/contenidos/1995/octubre/teora_de_recuperacin_de_informacin_modelos_fundamentales_y_aplicaciones_a_la_gestin_documental.html
- García Broncano, R. (s. f.). *RECUPERACIÓN Y ORGANIZACIÓN DE LA INFORMACIÓN*. Recuperado de <http://modelosrecuperacion.tripod.com/modelosrecuperacion.pdf>
- Groussard, T. (2010). *Java Enterprise Edition: Desarrollo de aplicaciones web con JEE 6*. ENI Editions. Recuperado de <https://books.google.es/books?hl=es&lr=&id=MdoMHI4QiJ0C&oi=fnd&pg=PP21&dq=aplicaciones+java+enterprise&ots=wOzLdND6CA&sig=MWELdZ8i4MKliZ0hBayVm1ur6QE#v=onepage&q=aplicaciones java enterprise&f=false>
- Hernández Moreno, J., Mayorga Cervantes, A., & Torres Téllez, M. J. (2015). *Atenea: sistema de búsqueda de documentos de trabajos terminales de la Escuela Superior de Cómputo*. INSTITUTO POLITÉCNICO NACIONAL. Recuperado de [https://tesis.ipn.mx/bitstream/handle/123456789/20155/Reporte Técnico TT 2014-A010.pdf?sequence=1&isAllowed=y](https://tesis.ipn.mx/bitstream/handle/123456789/20155/Reporte_Técnico_TT_2014-A010.pdf?sequence=1&isAllowed=y)

ICT - Udlap. (s. f.). 8. Datos no estructurados. Recuperado 23 de octubre de 2018, de <http://ict.udlap.mx/people/carlos/is346/admon08.html>

Instituto de Hidrología Meteorología y Estudios Ambientales. (2014). NORMAS Y ESTÁNDARES - IDEAM. Recuperado 30 de enero de 2019, de <http://www.ideam.gov.co/web/ecosistemas/normas-estandares>

iso25000.com. (2014). ISO 25010. Recuperado 15 de octubre de 2018, de <https://iso25000.com/index.php/normas-iso-25000/iso-25010>

Israel G. (2018). CMMI, El Modelo. Recuperado 22 de junio de 2019, de <https://ipmoguide.com/cmmi-el-modelo/>

Jaimes, L. G., & Riveros, F. V. (2005). Modelos clásicos de recuperación de la información. *Revista Integración, temas de matemáticas*, 23(1), 17-26.

Labrin, B. C. (2004). Gestión del conocimiento en enfoques de desarrollo de software tradicional y agilista. En *VI Workshop de Investigadores en Ciencias de la Computación*. Recuperado de http://sedici.unlp.edu.ar/bitstream/handle/10915/21214/Documento_completo.pdf?sequence=1&isAllowed=y

Levy, S. J., Romero Dapozo, J. R., & Pasini, A. C. (2016). Implementación práctica del agilismo en proyecto de Ingeniería de Software. En *XIX Concurso de Trabajos Estudiantiles (EST 2016)-JAIIO 45 (Tres de Febrero, 2016)*. Recuperado de http://sedici.unlp.edu.ar/bitstream/handle/10915/58198/Documento_completo.pdf-PDFA.pdf?sequence=1&isAllowed=y

Madrid Gorelov, V. P., Zazo, Á. F., Figuerola, C. G., & Berrocal, J. L. (2007). *Librerías Lucene y dotLucene para Recuperación de Información. Estudio y desarrollo de casos prácticos*. Recuperado de <http://reina.usal.es>

MARQUÉS, M. P. (2015). *BIG DATA - Técnicas, herramientas y aplicaciones*. (Alfaomega Grupo Editor, Ed.) (Primera Ed). Recuperado de <https://books.google.com.ec/books?id=R6ywDAAAQBAJ>

Martinez-Mendez, F.-J. (2004). *Recuperación de Información: Modelos, Sistemas y Evaluación*.

McCandless, M., Hatcher, E., & Gospodnetic, O. (2010). *Lucene in Action, Second Edition:*

Covers Apache Lucene 3.0. Manning Publications Co. Greenwich, CT, USA ©2010.
Recuperado de http://www.imamu.edu.sa/dcontent/IT_Topics/java/luceneinaction.pdf

Melendez R, P. G. (2012). IEEE 830. Recuperado 31 de enero de 2019, de
<https://sites.google.com/site/adsicae/definicion-de-requerimientos/ieee830>

Open Source Search Engines in Java. (s. f.). Recuperado 6 de mayo de 2019, de
<http://www.java-source.net/open-source/search-engines>

Pablo Ramos Hernández, J., & Alor-Hernández, G. (2008). Indización y Búsqueda a través de Lucene. *Veracruz, Sinaloa*.

Palacio, J. (2007). *Flexibilidad con Scrum*. Recuperado de
https://www.scrummanager.net/files/flexibilidad_con_scrum.pdf

Pardo, C., Pino, F. J., García, F., & Piattini, M. (2009). Analizando el apoyo de marcos SPI a las características de calidad del producto ISO 25010. *REICIS. Revista Española de Innovación, Calidad e Ingeniería del Software*, 5(2), 6-16. Recuperado de
<http://www.redalyc.org/articulo.oa?id=92217153003>

Rama Estudiantil de IEEE en UTN.BA. (2018). Objetivos del IEEE - SB IEEE | UTN.BA. Recuperado 31 de enero de 2019, de https://ieee.frba.utn.edu.ar/?page_id=40

Rodríguez Leyva, P., Viltres Sala, H., & Pons Flores, L. A. (2016). Componentes y funcionalidades de un sistema de recuperación de la información. *Revista Cubana de Ciencias Informáticas*, 10, 150-161. Recuperado de
<http://www.redalyc.org/articulo.oa?id=378346333012>

Rodriguez, M. L. (2005). *Modelos de Recuperación de la Información basados en Información Lingüística Difusa y Algoritmos Evolutivos. Mejorando la Representación de las Necesidades de Información. Extraído el 27 de abril de 2010. Departamento de Ciencias de la Computacion e Inteligencia Artificial. UNIVERSIDAD DE GRANADA*.

Rodríguez Perojo, K., & Ronda León, R. (2006). El web como sistema de información. Recuperado 2 de febrero de 2019, de
http://bvs.sld.cu/revistas/aci/vol14_1_06/aci08106.htm

Sobre la Comisión Electrotécnica Internacional (IEC) – Servicio de Acreditación Ecuatoriano. (2017). Recuperado 22 de junio de 2019, de
<https://www.acreditacion.gob.ec/sobre-iec/>

Subcomité 7. (2016). NC ISO/IEC 25010:2016 SQuaRE – Modelos de la Calidad de Software y Sistemas – Subcomité 7. Recuperado 15 de octubre de 2018, de <https://subcomite7.cubava.cu/2016/06/17/nc-isoiec-250102016-square-modelos-de-la-calidad-de-software-y-sistemas/#.W8USb2hKjIU>

The Apache Software Foundation. (s. f.). Apache Lucene - Apache Lucene Core. Recuperado 14 de octubre de 2018, de <https://lucene.apache.org/core/>

Tolosa, G. H., & Bordignon, F. (s. f.). Introducción a la Recuperación de Información. Conceptos, modelos y algoritmos básicos. Recuperado de <http://eprints.rclis.org/12243/1/Introduccion-RI-v9f.pdf>

ANEXOS

Anexo A: Levantamiento de Proceso de Búsqueda de Normas

Nombre del proceso	Búsqueda de Normas y Estándares
Descripción	El proceso de búsqueda de normas busca la implementación de una herramienta tecnológica de tal manera que se haga el seguimiento del proceso.
Propósito	Mejorar el proceso de Búsqueda de Normas, de modo que tanto docentes como estudiantes de la Universidad Técnica del Norte dispongan de la herramienta necesaria para elección de normas o estándares apropiadas para sus proyectos de investigación.

OBJETIVO DEL DOCUMENTO

Establecer una guía de referencia para la implementación del proceso de Búsqueda de Normas y Estándares, en el marco de la gestión por procesos, enfocada hacia su mejoramiento mediante la plataforma AURAPORTAL BPMS.

DEFINICIONES

- **TP:** Tarea Personal, corresponde a las tareas que el usuario va a realizar.
- **TS:** Tarea de Sistema, se ejecuta de manera automática por el sistema.
- **UTN:** Universidad Técnica del Norte
- **SBNI:** Sistema de Búsqueda de Normas para Ingenierías

OBJETIVO Y ALCANCE DEL PROCESO

El proceso de Búsqueda de Normas y Estándares contempla los siguientes subprocesos

- Indexación de normas (Con finalidad administrativa)
- Recuperación de Información (Con finalidad de búsqueda)

ACTORES DEL PROCESO (ROLES)

ROL	DESCRIPCION
Administradores	Responsable de la administración del SBNI
Docente	Persona que inicia el proceso de búsquedas
Estudiante	Persona que inicia el proceso de búsquedas

DIAGRAMA BPM

El siguiente diagrama esquematiza la secuencia de actividades que seguirá el proceso de búsqueda de normas.

El proceso de búsqueda de normas inicia cuando el estudiante o docente inician el uso del SBNI ya sea para el uso administrativo o de búsquedas. En el caso de ser de manera administrativa se realiza el subproceso de indexación de normas y en el caso de búsquedas se realiza el subproceso de recuperación de información.

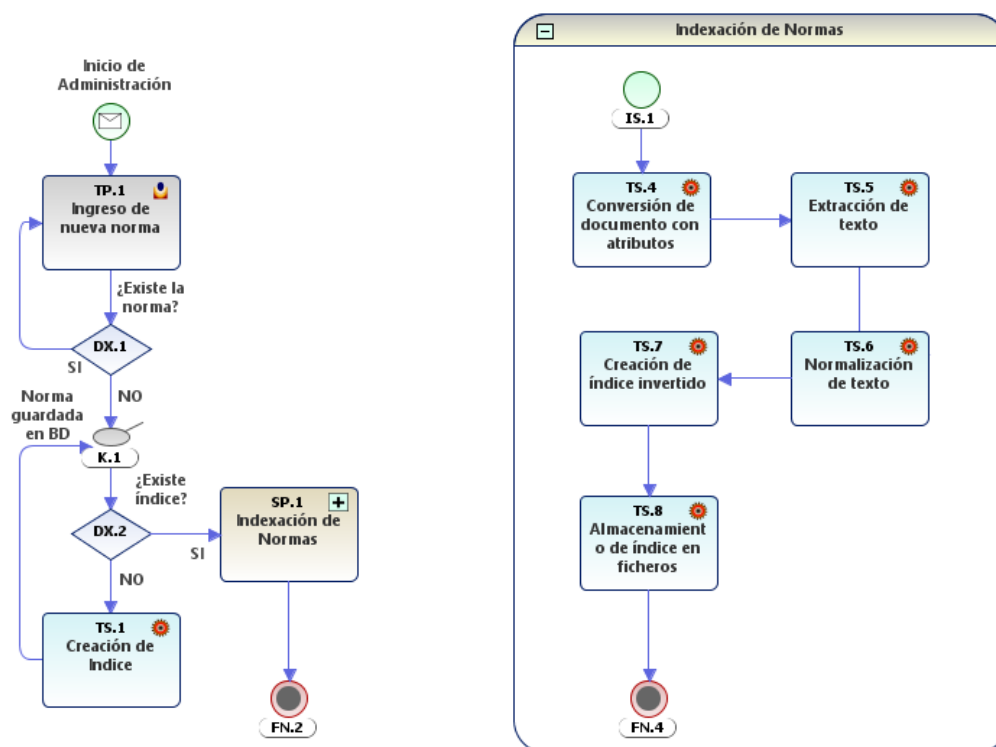


Diagrama BPMN del Proceso de Búsqueda de normas (con finalidad administrativa)

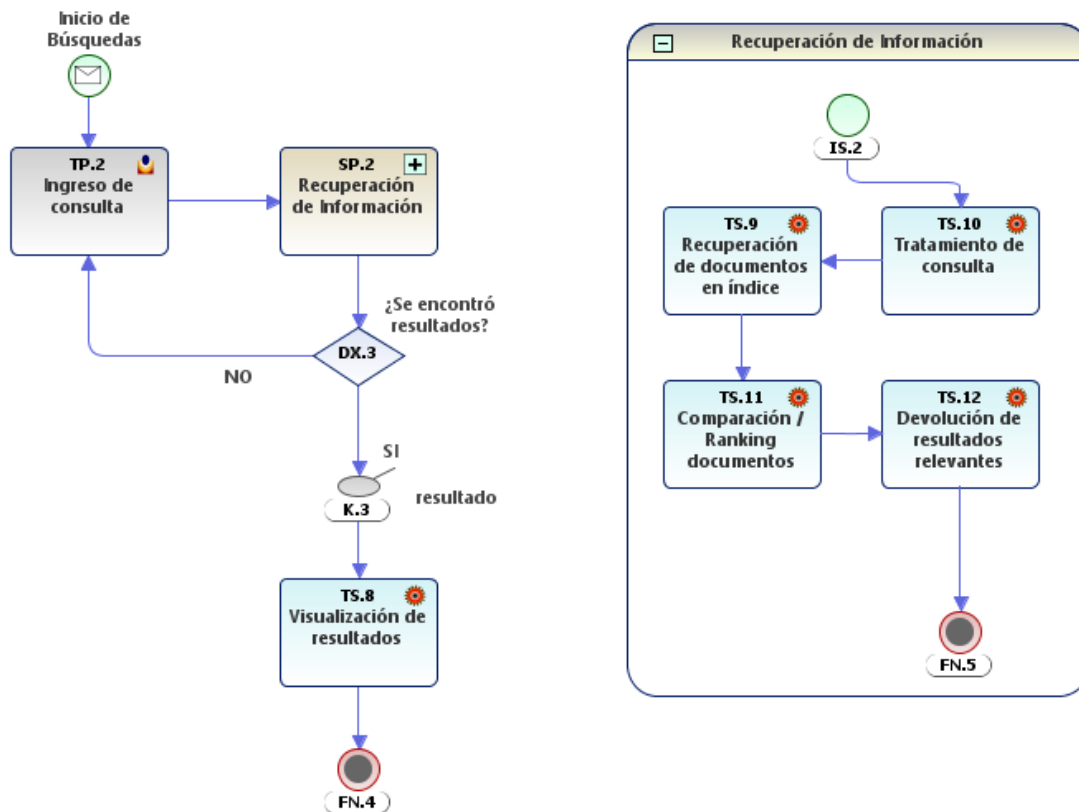


Diagrama BPMN del Proceso de Búsqueda de normas (con finalidad de búsqueda)

DESCRIPCION DETALLADA DEL PROCESO

En esta sección se describe en forma detallada los eventos, tareas y compuertas del proceso y su funcionamiento.

IDENTIFICACION	IM Inicio de Administración
TIPO	Inicio de Mensaje
DESCRIPCION DE FUNCIONALIDAD	<ul style="list-style-type: none"> El proceso de búsqueda de normas y estándares con finalidad administrativa iniciará el momento que se el administrador encargado del ingreso de normas del SBNI ingrese una nueva norma. Una vez creado el proceso el sistema realiza los subprocesos correspondientes
RESPONSABLE	Sistema
FORMULARIO	No aplica

IDENTIFICACION	TP.1 Ingreso de nueva norma
TIPO	Tarea Personal
DESCRIPCION DE FUNCIONALIDAD	<ul style="list-style-type: none"> El administrador encargado de gestión de normas deberá registrar los respectivos datos de la nueva norma.
RESPONSABLE	Administrador encargado de la gestión de normas
FORMULARIO	No Aplica

IDENTIFICACION	K.1 Norma guardada en BD
TIPO	Punto de Control
DESCRIPCION DE FUNCIONALIDAD	<ul style="list-style-type: none"> • Si la norma registrada en TP.1 no existe significa que el registro es nuevo entonces se guarda en la Base de Datos, caso contrario se debe ingresar una norma nueva.
FORMULARIO	No Aplica

IDENTIFICACION	TS.1 Creación de Índice
TIPO	Tarea de sistema
DESCRIPCION DE FUNCIONALIDAD	<ul style="list-style-type: none"> • Una vez guardada la norma en la Base de Datos se procede a guardar en el índice. Esta Tarea de sistema se ejecuta en caso de no existir un índice y procederá automáticamente a crear uno.
RESPONSABLE	Sistema
FORMULARIO	No Aplica

IDENTIFICACION	SP.1 Indexación de Normas
TIPO	Subproceso
DESCRIPCION DE FUNCIONALIDAD	<ul style="list-style-type: none"> • Una vez que se comprueba la existencia de índice en DX.2 se inicia este subproceso que dispara la TS. 4.
RESPONSABLE	Sistema
FORMULARIO	No Aplica

IDENTIFICACION	TS.4 Conversión de documento con atributos
TIPO	Tarea de sistema
DESCRIPCION DE FUNCIONALIDAD	<ul style="list-style-type: none"> • Esta conversión consiste en crear un documento que tenga los atributos (Fields) correspondientes a la entidad de la Base de Datos. El sistema se encarga de seleccionar los atributos que serán relevantes para las posteriores búsquedas
RESPONSABLE	Sistema
FORMULARIO	No Aplica

IDENTIFICACION	TS.5 Extracción de texto
TIPO	Tarea de sistema
DESCRIPCION DE FUNCIONALIDAD	<ul style="list-style-type: none"> • Esta tarea se ejecuta una vez creado el documento, y consiste en extraer el texto de la norma e irla separando en palabras
RESPONSABLE	Sistema
FORMULARIO	No Aplica

IDENTIFICACION	TS.6 Normalización de texto
TIPO	Tarea de sistema
DESCRIPCION DE FUNCIONALIDAD	<ul style="list-style-type: none"> • Esta tarea tiene como finalidad eliminar palabras que no ayudan en la recuperación de información y realizar un tratamiento a cada palabra existente
RESPONSABLE	Sistema
FORMULARIO	No Aplica

IDENTIFICACION	TS.7 Creación de índice invertido
TIPO	Tarea de sistema
DESCRIPCION DE FUNCIONALIDAD	<ul style="list-style-type: none"> En Esta tarea consiste en contar las palabras repetidas y crear un índice que contenga las palabras con el número de veces que se repite.
RESPONSABLE	Sistema
FORMULARIO	No Aplica

IDENTIFICACION	TS.8 Almacenamiento de índice en ficheros
TIPO	Tarea de sistema
DESCRIPCION DE FUNCIONALIDAD	<ul style="list-style-type: none"> En esta tarea el sistema procede almacenar el índice en ficheros dentro de un directorio del sistema operativo.
RESPONSABLE	Sistema
FORMULARIO	No Aplica

IDENTIFICACION	IM Inicio de Búsquedas
TIPO	Inicio de Mensaje
DESCRIPCION DE FUNCIONALIDAD	<ul style="list-style-type: none"> El proceso de búsqueda de normas y estándares iniciará el momento que se el docente o estudiante realiza una búsqueda en el SBNI. Una vez creado el proceso el sistema realiza los subprocesos correspondientes
RESPONSABLE	Sistema
FORMULARIO	No aplica

IDENTIFICACION	TP.2 Ingreso de consulta
TIPO	Tarea Personal
DESCRIPCION DE FUNCIONALIDAD	<ul style="list-style-type: none"> La tarea consiste en recibir una consulta por parte del docente o estudiante, para una posterior búsqueda
RESPONSABLE	Usuario del sistema (Docente o Estudiante)
FORMULARIO	No Aplica

IDENTIFICACION	SP.2 Recuperación de Información
TIPO	Subproceso
DESCRIPCION DE FUNCIONALIDAD	<ul style="list-style-type: none"> La consulta del usuario será tratada en este subproceso, el que disparará la TS.10
RESPONSABLE	Sistema
FORMULARIO	No Aplica

IDENTIFICACION	TS.10 Tratamiento de consulta
TIPO	Tarea de Sistema
DESCRIPCION DE FUNCIONALIDAD	<ul style="list-style-type: none"> Esta tarea tiene como finalidad convertir la consulta en palabras para su posterior comparación con el índice
RESPONSABLE	Sistema
FORMULARIO	No Aplica

IDENTIFICACION	TS.9 Recuperación de documentos en índice
TIPO	Tarea de Sistema
DESCRIPCION DE FUNCIONALIDAD	<ul style="list-style-type: none"> Esta tarea tiene como finalidad abrir los archivos de índice y tenerlos listos para una comparación con la consulta
RESPONSABLE	Sistema
FORMULARIO	No Aplica

IDENTIFICACION	TS.11 Comparación / Ranking documentos
TIPO	Tarea de Sistema
DESCRIPCION DE FUNCIONALIDAD	<ul style="list-style-type: none"> En esta tarea se realiza una comparación entre la consulta y los documentos almacenados en el índice. Esta tarea realiza además un ranking de los documentos más relevantes a la consulta de usuario
RESPONSABLE	Sistema
FORMULARIO	No Aplica

IDENTIFICACION	TS.12 Devolución de resultados relevantes
TIPO	Tarea de Sistema
DESCRIPCION DE FUNCIONALIDAD	<ul style="list-style-type: none"> Esta tarea tiene como finalidad devolver al usuario los documentos o resultados más relevantes a la consulta realizada por el usuario.
RESPONSABLE	Sistema
FORMULARIO	No Aplica

IDENTIFICACION	TS.8 Visualización de resultados
TIPO	Tarea de Sistema
DESCRIPCION DE FUNCIONALIDAD	<ul style="list-style-type: none"> Esta tarea se ejecuta luego de obtener resultados del SP.2, en caso de no obtenerse ningún resultado se debe ingresar una nueva consulta por parte del usuario.
RESPONSABLE	Sistema
FORMULARIO	No Aplica

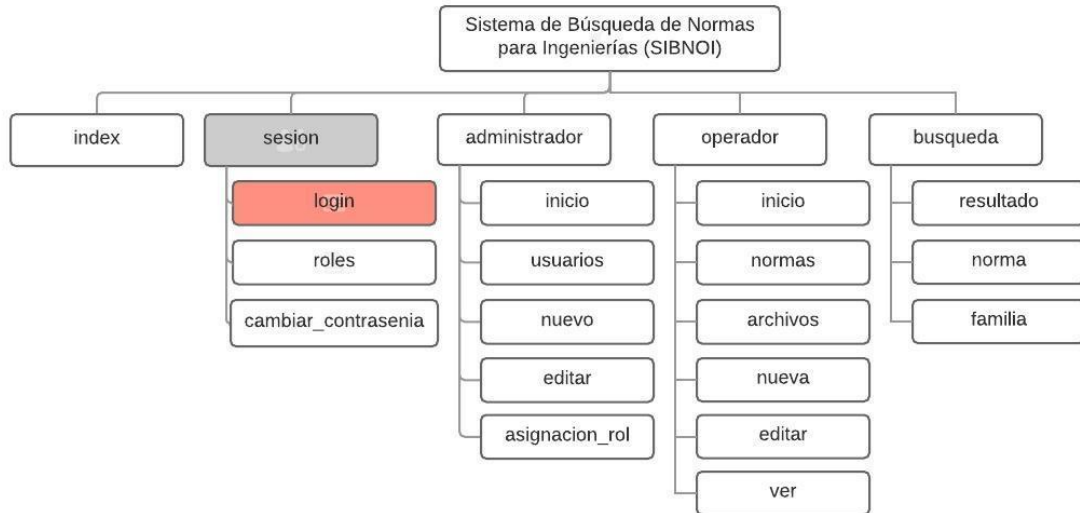
PUNTOS DE CONTROL

Se ha incluido los siguientes puntos de control que permitirán visualizar el estado de los hitos de búsqueda de normas y estándares:

- **NORMA GUARDADA EN BD:** Normas que se han guardado exitosamente en la base de datos, independiente de si se guardará en el índice
- **RESULTADO:** Las normas más relevantes que se ha recuperado de los índices con respecto a una consulta proporcionada por el usuario del SBNI.

Anexo B: Wireframes SBNI

Árbol web



Wireframes

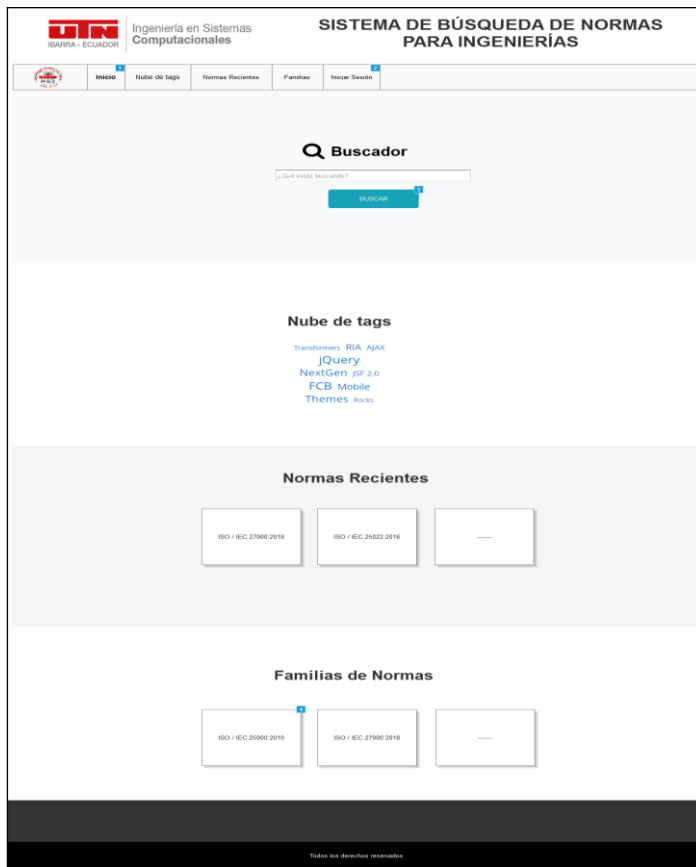
Los Wireframes de las distintas interfaces de usuario han sido diseñadas con la herramienta Axure RP 8, misma que ha facilitado el prototipado del Sistema de Búsqueda de Normas para Ingenierías (SBNI) y ha proporcionado la funcionalidad de generar un reporte con todas las interfaces de usuario y las distintas interacciones de estas.

Mencionadas interfaces e interacciones son detalladas a continuación.

Wireframes Generales

- index

Interfaz de usuario



Interacciones de página

N° nota	Interacción	Texto
1	OnClick: Case 1: Open index in Current Window	Inicio
2	OnClick: Case 1: Open login in Current Window	Iniciar Sesión
3	OnClick: Case 1 (If text on txtBuscar equals "ISO 25000"): Open resultado in Current Window	BUSCAR
4	OnClick: Case 1: Open familia in Current Window	ISO / IEC 25000:2010

- roles

Interfaz de usuario



Interacciones de página

N° nota	Interacción	Texto
1	OnClick: Case 1: Open login in Current Window	Cerrar Sesión
2	OnClick: Case 1: Open inicio in Current Window	SINOI
3	OnClick: Case 1: Open inicio in Current Window	Ingresar
4	OnClick: Case 1: Open inicio in Current Window	Ingresar
5	OnClick: Case 1: Open cambiar_contraseña in Current Window	Cambiar contraseña

- login

Interfaz de usuario

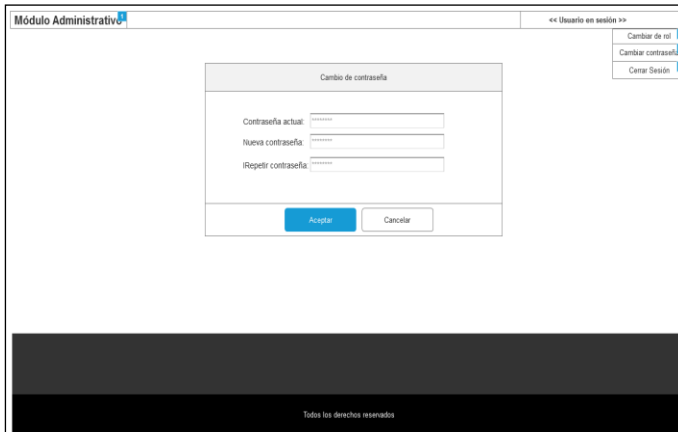


Interacciones de página

N° nota	Interacción	Texto
1	OnClick: Case 1: Open roles in Current Window	Iniciar Sesión

- cambiar_contraseña

Interfaz de usuario



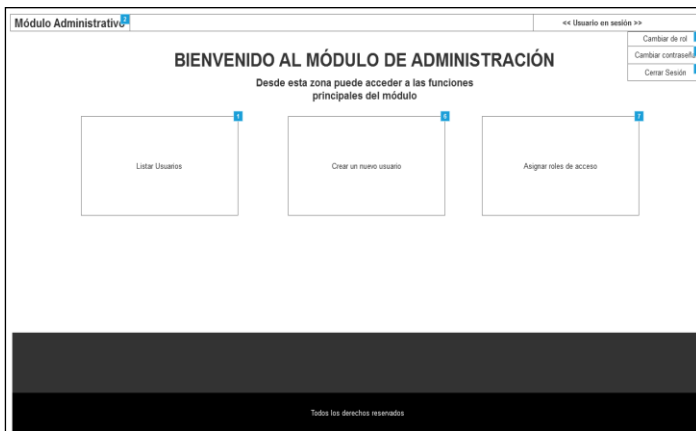
Interacciones de página

N° nota	Interacción	Texto
1	OnClick: Case 1: Open inicio in Current Window	Módulo Administrativo
2	OnClick: Case 1: Open cambiar_contraseña in Current Window	Cambiar contraseña
3	OnClick: Case 1: Open login in Current Window	Cerrar Sesión
4	OnClick: Case 1: Open roles in Current Window	Cambiar de rol

Wireframes del Módulo de Administración

- inicio

Interfaz de usuario

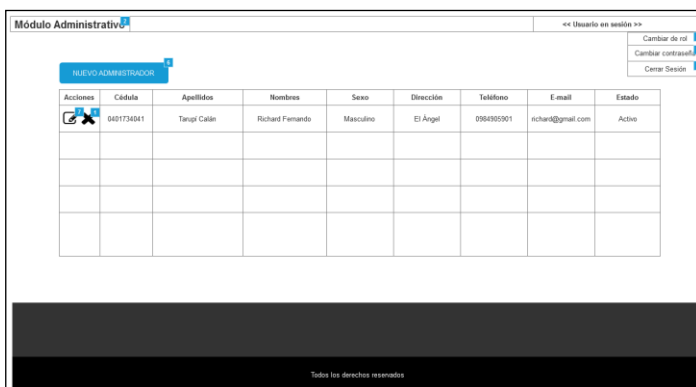


Interacciones de página

N° nota	Interacción	Texto
1	OnClick: Case 1: Open usuarios in Current Window	Listar Usuarios
2	OnClick: Case 1: Open inicio in Current Window	Módulo Administrativo
3	OnClick: Case 1: Open cambiar_contraseña in Current Window	Cambiar contraseña
5	OnClick: Case 1: Open roles in Current Window	Cambiar de rol
6	OnClick: Case 1: Open nuevo in Current Window	Crear un nuevo usuario
7	OnClick: Case 1: Open asignacion_rol in Current Window	Asignar roles de acceso

- usuarios

Interfaz de usuario



Interacciones de página

N° nota	Interacción	Texto
1	OnClick: Case 1 (If text on celEstado1 equals "Activo"): Set This to Activar Set text on celEstado1 equal to "Inactivo" Case 2 (Else If text on celEstado1 equals "Inactivo"): Set This to Inactivar Set text on celEstado1 equal to "Activo"	
2	OnClick: Case 1: Open inicio in Current Window	Módulo Administrativo
6	OnClick: Case 1: Open nuevo in Current Window	NUEVO ADMINISTRADOR

- nuevo

Interfaz de usuario

The screenshot shows a web interface for creating a new user. The page title is 'Módulo Administrativo' and the user is logged in as 'Usuario en sesión'. The main form is titled 'Nuevo Usuario' and contains the following fields: 'N° de cédula' (ID number), 'Apellidos' (Last name), 'Nombres' (First name), 'Sexo' (Gender), 'Dirección' (Address), 'Teléfono' (Phone), and 'Correo electrónico' (Email). A note states 'Los campos marcados con * son obligatorios' (Fields marked with * are mandatory). The form includes 'Guardar' (Save) and 'Cancelar' (Cancel) buttons. A sidebar on the right contains links for 'Cambiar de rol', 'Cambiar contraseña', and 'Cerrar Sesión'. The footer contains the text 'Todos los derechos reservados'.

Interacciones de página

N° nota	Interacción	Texto
1	OnClick: Case 1: Open inicio in Current Window	Módulo Administrativo
2	OnClick: Case 1: Open cambiar_contraseña in Current Window	Cambiar contraseña
3	OnClick: Case 1: Open login in Current Window	Cerrar Sesión
4	OnClick: Case 1: Open roles in Current Window	Cambiar de rol
5	OnClick: Case 1: Open usuarios in Current Window	Guardar
6	OnClick: Case 1: Open usuarios in Current Window	Cancelar

- editar

Interfaz de usuario

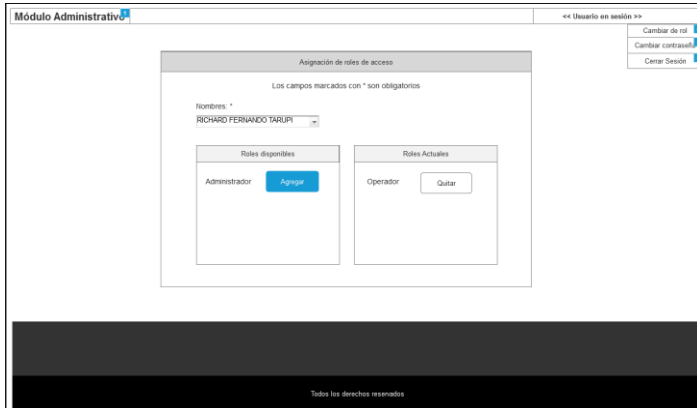
The screenshot shows a web interface for editing an existing user. The page title is 'Módulo Administrativo' and the user is logged in as 'Usuario en sesión'. The main form is titled 'Editar Usuario' and contains the following fields: 'N° de cédula' (ID number), 'Apellidos' (Last name), 'Nombres' (First name), 'Sexo' (Gender), 'Dirección' (Address), 'Teléfono' (Phone), and 'Correo electrónico' (Email). A note states 'Los campos marcados con * son obligatorios' (Fields marked with * are mandatory). The form includes 'Guardar' (Save) and 'Cancelar' (Cancel) buttons. A sidebar on the right contains links for 'Cambiar de rol', 'Cambiar contraseña', and 'Cerrar Sesión'. The footer contains the text 'Todos los derechos reservados'.

Interacciones de página

N° nota	Interacción	Texto
1	OnClick: Case 1: Open inicio in Current Window	Módulo Administrativo
2	OnClick: Case 1: Open cambiar_contraseña in Current Window	Cambiar contraseña
3	OnClick: Case 1: Open login in Current Window	Cerrar Sesión
4	OnClick: Case 1: Open roles in Current Window	Cambiar de rol
5	OnClick: Case 1: Open usuarios in Current Window	Guardar
6	OnClick: Case 1: Open usuarios in Current Window	Cancelar

- asignacion_rol

Interfaz de usuario



Interacciones de página

Nº nota	Interacción	Texto
1	OnClick: Case 1: Open inicio in Current Window	Módulo Administrativo
2	OnClick: Case 1: Open cambiar_contraseña in Current Window	Cambiar contraseña
3	OnClick: Case 1: Open login in Current Window	Cerrar Sesión
4	OnClick: Case 1: Open roles in Current Window	Cambiar de rol

Wireframes del Módulo Operativo

- inicio

Interfaz de usuario

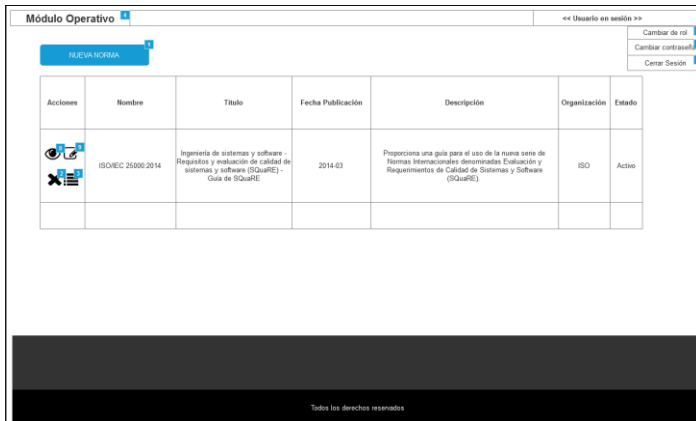


Interacciones de página

Nº nota	Interacción	Texto
1	OnClick: Case 1: Open inicio in Current Window	Módulo Operativo
2	OnClick: Case 1: Open cambiar_contraseña in Current Window	Cambiar contraseña
3	OnClick: Case 1: Open normas in Current Window	Listar sus Normas
4	OnClick: Case 1: Open login in Current Window	Cerrar Sesión
5	OnClick: Case 1: Open nueva in Current Window	Crear Norma
6	OnClick: Case 1: Open roles in Current Window	Cambiar de rol

- normas

Interfaz de usuario

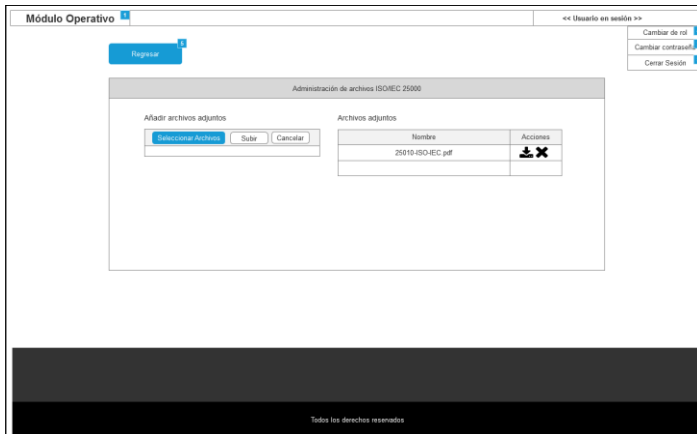


Interacciones de página

N° nota	Interacción	Texto
1	OnClick: Case 2: Open nueva in Current Window	NUEVA NORMA
2	OnClick: Case 1 (If text on celEstado1 equals "Activo"): Set This to Activar Set text on celEstado1 equal to "Inactivo" Case 2 (Else If text on celEstado1 equals "Inactivo"): Set This to Inactivar Set text on celEstado1 equal to "Activo"	
3	OnClick: Case 1: Open archivos in Current Window	
4	OnClick: Case 1: Open inicio in Current Window	Módulo Operativo
5	OnClick: Case 1: Open cambiar_contraseña in Current Window	Cambiar contraseña
6	OnClick: Case 1: Open login in Current Window	Cerrar Sesión
7	OnClick: Case 1: Open roles in Current Window	Cambiar de rol
8	OnClick: Case 1: Open ver in Current Window	
9	OnClick: Case 1: Open editar in Current Window	

- archivos

Interfaz de usuario

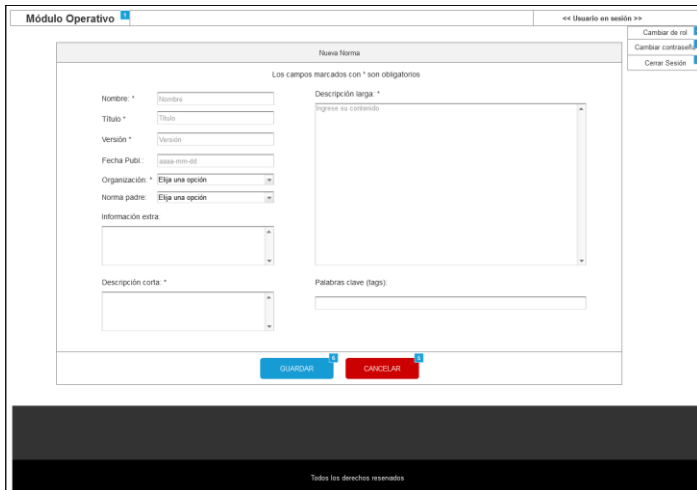


Interacciones de página

N° nota	Interacción	Texto
1	OnClick: Case 1: Open inicio in Current Window	Módulo Operativo
2	OnClick: Case 1: Open cambiar_contraseña in Current Window	Cambiar contraseña
3	OnClick: Case 1: Open login in Current Window	Cerrar Sesión
4	OnClick: Case 1: Open roles in Current Window	Cambiar de rol
5	OnClick: Case 1: Open normas in Current Window	Regresar

- nueva

Interfaz de usuario



Interacciones de página

N° nota	Interacción	Texto
1	OnClick: Case 1: Open inicio in Current Window	Módulo Operativo
2	OnClick: Case 1: Open cambiar_contraseña in Current Window	Cambiar contraseña
3	OnClick: Case 1: Open login in Current Window	Cerrar Sesión
4	OnClick: Case 1: Open roles in Current Window	Cambiar de rol
5	OnClick: Case 1: Open normas in Current Window	CANCELAR
6	OnClick: Case 1: Open normas in Current Window	GUARDAR

- editar

Interfaz de usuario

The screenshot shows the 'Editar Norma' form. It includes fields for 'Nombre' (ISO/IEC 25000 2014), 'Título' (Ingeniería de sistemas y software), 'Versión' (2014), 'Fecha Publ.' (2014-03), 'Organización' (dropdown), and 'Norma padre' (dropdown). There are also sections for 'Descripción larga', 'Descripción corta', and 'Palabras clave (tags)'. At the bottom, there are 'GUARDAR' and 'CANCELAR' buttons.

Interacciones de página

N° nota	Interacción	Texto
1	OnClick: Case 1: Open inicio in Current Window	Módulo Operativo
2	OnClick: Case 1: Open cambiar_contraseña in Current Window	Cambiar contraseña
3	OnClick: Case 1: Open login in Current Window	Cerrar Sesión
4	OnClick: Case 1: Open roles in Current Window	Cambiar de rol
5	OnClick: Case 1: Open normas in Current Window	CANCELAR
6	OnClick: Case 1: Open normas in Current Window	GUARDAR

- ver

Interfaz de usuario

The screenshot shows the 'Ver' page for the 'ISO/IEC 25000 2014' standard. It displays the title 'Ingeniería de sistemas y software - Requisitos y evaluación de calidad de sistemas y software (SQuaRE) - Guía de SQuaRE'. It includes a 'Descripción corta' and 'Información completa' section. On the right, there is a summary table for 'Información General' with fields for 'Versión', 'Fecha Publ.', 'Organización', and 'Norma padre'. Below this is a 'Palabras clave (tags)' section listing 'jQuery', 'NextGen', 'FCB Mobile', and 'Themes'. At the top, there are 'Regresar' and 'Imprimir' buttons.

Interacciones de página

N° nota	Interacción	Texto
1	OnClick: Case 1: Open inicio in Current Window	Módulo Operativo
2	OnClick: Case 1: Open cambiar_contraseña in Current Window	Cambiar contraseña
3	OnClick: Case 1: Open login in Current Window	Cerrar Sesión
4	OnClick: Case 1: Open roles in Current Window	Cambiar de rol
5	OnClick: Case 1: Open normas in Current Window	Regresar

Wireframes del Módulo Público

- resultado

Interfaz de usuario

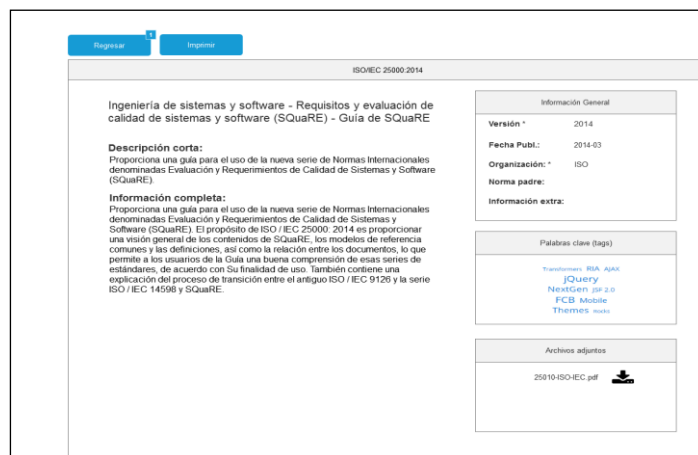


Interacciones de página

Nº nota	Interacción	Texto
1	OnClick: Case 1: Open norma in Current Window	ISO / IEC 25000:2014
2	OnClick: Case 1: Open index in Current Window	Inicio
3	OnClick: Case 1 (If text on Unidentified equals "ISO 25000"): Open resultado in Current Window	BUSCAR

- norma

Interfaz de usuario



Interacciones de página

Nº nota	Interacción	Texto
1	OnClick: Case 1: Open normas in Current Window	Regresar

- familia

Interfaz de usuario



Interacciones de página

Nº nota	Interacción	Texto
1	OnClick: Case 1: Open index in Current Window	Inicio

