

**UNIVERSIDAD POLITÉCNICA SALESIANA
SEDE QUITO**

**CARRERA:
INGENIERÍA DE SISTEMAS**

**Trabajo de titulación previo a la obtención del título de:
INGENIERA E INGENIERO DE SISTEMAS**

**TEMA:
DESARROLLO DE UNA APLICACIÓN WEB PARA EL REGISTRO Y
SEGUIMIENTO DE OPCIONES DE TITULACIÓN PARA LA CARRERA DE
INGENIERÍA DE SISTEMAS DE LA UNIVERSIDAD POLITÉCNICA
SALESIANA SEDE QUITO.**

**AUTORES:
MARÍA ELIZABETH FLORES SÁNCHEZ
DIEGO ARMANDO QUISUPANGUI LEMA**

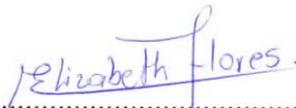
**TUTOR:
FRANKLIN EDMUNDO HURTADO LARREA**

Quito, agosto del 2016

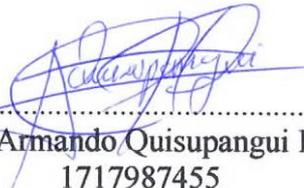
CESIÓN DE DERECHOS DE AUTOR

Nosotros María Elizabeth Flores Sánchez y Diego Armando Quisupangui Lema, con documento de identificación N° 1719355552 y 1717987455 respectivamente, manifestamos con voluntad y cedemos a la Universidad Politécnica Salesiana la titularidad sobre los derechos patrimoniales en virtud de que somos autores del trabajo de titulación intitulado: DESARROLLO DE UNA APLICACIÓN WEB PARA EL REGISTRO Y SEGUIMIENTO DE OPCIONES DE TITULACIÓN PARA LA CARRERA DE INGENIERÍA DE SISTEMAS DE LA UNIVERSIDAD POLITÉCNICA SALESIANA SEDE QUITO, mismo que ha sido desarrollado para optar por el título de: Ingeniero e Ingeniero de Sistemas, en la Universidad Politécnica Salesiana, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

En aplicación a lo determinado en la Ley de Propiedad Intelectual, en nuestra condición de autores nos reservamos los derechos morales de la obra antes citada. En concordancia, suscribo este documento en el momento que hago entrega del trabajo final en formato impreso y digital a la Biblioteca de la Universidad Politécnica Salesiana.



.....
María Elizabeth Flores Sánchez
1719355552



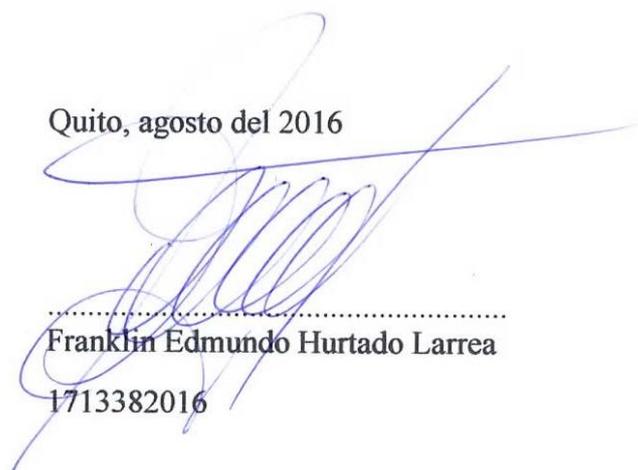
.....
Diego Armando Quisupangui Lema
1717987455

Quito, agosto de 2016

DECLARATORIA DE COAUTORÍA DEL DOCENTE TUTOR

Yo, declaro que bajo mi dirección y asesoría fue desarrollado el trabajo de titulación **DESARROLLO DE UNA APLICACIÓN WEB PARA EL REGISTRO Y SEGUIMIENTO DE OPCIONES DE TITULACIÓN PARA LA CARRERA DE INGENIERÍA DE SISTEMAS DE LA UNIVERSIDAD POLITÉCNICA SALESIANA SEDE QUITO** realizado por María Elizabeth Flores Sánchez y Diego Armando Quisupangui Lema, obteniendo un producto que cumple con todos los requisitos estipulados por la Universidad Politécnica Salesiana para ser considerados como trabajo final de titulación.

Quito, agosto del 2016



.....
Franklin Edmundo Hurtado Larrea

1713382016

Agradecimiento

Agradecemos a la Universidad Politécnica Salesiana por habernos brindado la formación académica para llegar a ser personas profesionales en la carrera de Ingeniería en Sistemas, además de ofrecer todas las herramientas necesarias para el desarrollo del proyecto de titulación.

A nuestro tutor Franklin Hurtado, por los conocimientos y experiencia transmitidos durante las tutorías ofrecidas en el desarrollo del proyecto, al tiempo que nos brindó al guiarnos y encaminarnos por las mejores alternativas a seguir para llegar a la culminación del propósito planteado.

Índice

Introducción	1
Justificación.....	3
Objetivos	5
Objetivo general.	5
Objetivos específicos.....	5
Alcance.....	6
Marco Metodológico.....	8
Capítulo 1	10
1.1 Marco referencial o institucional.....	10
1.2 Marco Legal	10
1.3 Marco Teórico	12
1.3.1 Metodologías Ágiles	12
1.3.2 Scrum	12
1.3.3 Técnicas de recolección de información	15
1.3.4 BPMN	17
1.3.5 Product Breakdown Structure (PBS)	17
1.3.6 Diagrama de Casos de uso	17
1.3.7 Diagrama de secuencia.....	18
1.3.8 Diagrama de clases.....	18
1.3.9 Herramientas de desarrollo	18
1.4 Factibilidad del proyecto.	20
1.4.1 Factibilidad técnica.	21
1.4.2 Factibilidad económica.	21
1.5 Pruebas	21
Capítulo 2.....	23
Análisis y Diseño	23
2.1 Diagrama de Procesos	23
2.2 Requerimientos del sistema.....	24
2.2.1 Product Backlog.....	27
2.3 Diagramas de casos de uso	31
2.3.1 Registrar usuario	32
2.3.2 Publicación de temas.....	32

2.3.3	Inscripción.....	34
2.3.4	Aprobación de proyecto	35
2.3.5	Seguimiento de proyecto.....	36
2.4	Diagramas de Secuencia.....	37
2.4.1	Gestión de docente	37
2.4.2	Inscripción de estudiante.....	38
2.4.3	Seguimiento estudiante	38
2.4.4	Gestión de CUTS	39
2.5	Modelo lógico de base de datos.....	40
Capítulo 3	42
3.1	Estándares de programación.....	43
3.1.1	Notación	43
3.2	Construcción de la aplicación web	44
3.2.1	Diagrama físico	44
3.2.2	Hibernate	46
3.2.3	Diagrama de clases.....	52
3.2.4	Código relevante	55
3.2.5	Diccionario de datos.....	59
3.3	Pruebas	61
3.3.1	Plan de Pruebas	61
3.3.2	Ambiente de pruebas.....	62
3.3.3	Duración de pruebas.....	63
3.3.4	Pruebas Funcionales.....	63
3.3.5	Pruebas no funcionales.....	66
Conclusiones	69
Recomendaciones	71
Referencias	72
Anexos	74

Índice de Figuras

Figura 1. PBS inicial.....	6
Figura 2. Proceso global de titulación.....	24
Figura 3. Caso de uso. Registro de usuario.....	32
Figura 4. Caso de uso. Publicación de temas.....	33
Figura 5. Caso de uso. Inscripción a unidad de titulación.....	34
Figura 6. Caso de uso. Aprobación del tema con resolución.....	35
Figura 7. Caso de uso. Seguimiento de proyecto.....	36
Figura 8. Diagrama de secuencia. Gestión de tema.....	37
Figura 9. Diagrama de secuencia. Gestión de inscripción.....	38
Figura 10. Diagrama de secuencia. Seguimiento de estudiante.....	39
Figura 11. Diagrama de secuencia. Gestión de CUTS.....	40
Figura 12. Diagrama lógico de base de datos en su versión final.....	41
Figura 13. Arquitectura MVC del sistema informático.....	42
Figura 14. Diagrama físico de base de datos.....	45
Figura 15. Archivo de configuración conexión a base de datos mediante <i>Hibernate</i>	46
Figura 16. Archivo de configuración con tablas de la base de datos.....	47
Figura 17. Parámetros mapeados desde tabla tema.....	48
Figura 18. Etiquetas JPA de la entidad tema.....	48
Figura 19. Mapeo <i>Hibernate</i> de la tabla tema.....	49
Figura 20. Archivo generado por <i>Hibernate</i> para manejo de sesiones.....	49
Figura 21. Interfaz Java con patrón Dao para tema.....	50
Figura 22. Implementación de interfaz <i>Dao</i> de tema.....	51
Figura 23. Ejemplo de uso de interfaz en método de la capa lógica.....	52
Figura 24. Estructura de diagrama de clases organizada por paquetes.....	52
Figura 25. Diagrama de clase paquete <i>bean</i>	53
Figura 26. Diagrama de clase del paquete modelo que contiene entidades.....	54
Figura 27. Diagrama de clase ampliación de entidad tema.....	54
Figura 28. Consulta de id para la última convocatoria habilitada.....	56
Figura 29. Consulta para obtener entidad convocatoria por su id.....	56
Figura 30. Actualización de estado de tema.....	57
Figura 31. Notificación <i>push</i>	58

Figura 32. Generar diálogo externo	59
Figura 33. Porcentaje de usabilidad de la aplicación.	66

Índice de Tablas

Tabla 1. Product backlog, versión 1.....	9
Tabla 2. Metodologías.....	12
Tabla 3. Factibilidad técnica	21
Tabla 4. Recursos del sistema	21
Tabla 5. Especificación de requerimientos	25
Tabla 6. Product backlog.....	28
Tabla 7. Sprint backlog	29
Tabla 8. Descripción de los actores del sistema.....	31
Tabla 9. Registrar usuario	32
Tabla 10. Publicación de temas.....	33
Tabla 11. Inscripción a opción de titulación	34
Tabla 12. Aprobación de proyecto	35
Tabla 13. Seguimiento de proyecto.....	36
Tabla 14. Prefijos	43
Tabla 15. Notación CamelCase.....	44
Tabla 16. Esquema de diccionario de datos	60
Tabla 17. Plan de pruebas	61
Tabla 18. Características de máquina.....	62
Tabla 19. Duración de pruebas.....	63
Tabla 20. Resultado de pruebas funcionales	65
Tabla 21. Carga de usuarios	67
Tabla 22. Resultado de pruebas	67
Tabla 23. Carga de usuarios	68
Tabla 24. Resultado de pruebas	68

Índice de Anexos

Anexos	74
Anexo 1. Diagramas de proceso	74
Anexo 2. Diagramas de clases.....	78
Anexo 3. Diccionario de base de datos	83
Anexo 4. Pruebas.....	95
Anexo 5. Pruebas con Jmeter	106

Resumen

Este proyecto se enfoca en el desarrollo de una aplicación web para el seguimiento de los proyectos de titulación de la Carrera de Ingeniería de Sistemas de Universidad Politécnica Salesiana Sede Quito, con el fin de solventar los inconvenientes que esta entidad experimenta actualmente, por ejemplo: la información desactualizada referente al desarrollo de los proyectos, posible sobrecarga de proyectos asignados a tutores, etc.

Para ello se analizó el nuevo proceso de titulación y se realizó el desarrollo de la aplicación de modo que los usuarios puedan disponer de la información clave del nuevo procedimiento de manera oportuna, precisa y segura.

Las herramientas utilizadas en el desarrollo de la aplicación son : Hibernate 3.6 como framework en la capa de persistencia, gestor de base de datos Postgres, capa de negocios con JSF 2.0 y su capa de presentación Primefaces 5.3.

El capítulo uno abarca información de las herramientas y la metodología que se utiliza para el desarrollo del sistema informático.

En el capítulo dos se realiza un análisis del problema y se presenta el diseño del prototipo de cada módulo del sistema.

El capítulo tres detalla la construcción de la aplicación, considerando aspectos como arquitectura, código relevante y la información resultante de la aplicación de los diferentes tipos de pruebas.

Abstract

This project focuses on the development of a web application for the track of the titulation projects in the Systems Engineering career at the Politécnica Salesiana University in Quito, for the purpose of solving the inconveniences that this entity experiences at present, for example: outdated information referring to the development of the projects, possible overload of projects assigned to tutors, etc.

For this it was analyzed the new titulation process and the application was developed so that users can have key information of the new procedure in an opportune, precise and safe way.

The tools used for the application development are: Hibernate 3.6 as framework for the persistence layer, PostgreSQL as the database management system, business layer with Java Server Faces 2.0 and its presentation layer Primefaces 5.3.

The first chapter covers information about the methodology and tools that were used during the development of the informatics system.

In chapter two it is made an analysis of the problem and it presents the design prototype of each system module.

In chapter three it is detailed the construction of the application, considering several aspects such as software architecture, relevant code and the resulting information from the various kinds of tests applied.

Introducción

En el Ecuador, hasta el año 2015 existía una sola opción de titulación que era la realización de una tesis de grado, luego de esta fecha y por disposición del Consejo de Educación Superior (CES) se establece la creación de la Unidad de Titulación, que permite: “ a) el desarrollo de un trabajo de titulación basado en procesos de investigación e intervención o, b) la preparación y aprobación de un examen de grado de carácter complejo.” (Consejo Superior U. , 2014)

La Universidad Politécnica Salesiana se acoge a la disposición del CES por lo cual crea la Unidad de Titulación Especial (UTE), para gestionar los procesos de esta unidad dispone la implementación de una coordinación de titulación especial en cada carrera, por consiguiente la dirección de carrera de Ingeniería de Sistemas decide ofertar las siguientes opciones de titulación: proyecto técnico, artículo científico y examen complejo.

Actualmente en la Carrera de Ingeniería de Sistemas de la Universidad Politécnica Salesiana se maneja el proceso de titulación por medio de hojas de cálculo, que almacena información relevante a los proyectos, no existe un sistema para la gestión de los trabajos de titulación, lo que causa una posible sobrecarga de horas asignadas a tutores, redundancia de datos respecto a las horas asignadas de cada tutor, tiempo de duración, porcentaje de avance de cada proyecto, no se tiene actualiza periódicamente los datos del área de titulación; es por esto que se decidió realizar un software que facilite la gestión en la Coordinación de la Unidad de Titulación Especial de Ingeniería de Sistemas (CUTS).

Con esta aplicación la CUTS podrá realizar el seguimiento de las distintas opciones de titulación, dispondrá de información referente al docente en el desarrollo de los proyectos de titulación e información del estado actual en el avance del proyecto.

El sistema informático se construyó utilizando herramientas como: Netbeans 7.4, para la codificación, ya que ofrece la compatibilidad para todas las especificaciones de Java EE 7, JSF 2.0, para la arquitectura de desarrollo web, PostgreSQL, como motor de base de datos relacional, Primefaces, para facilitar el desarrollo de la capa presentación.

Justificación

“El Consejo de Educación Superior CES creado de acuerdo al artículo 353 de la Constitución de la República del Ecuador, es el organismo independiente que se encarga de planificar, regular y coordinar internamente el Sistema de Educación Superior, y la relación entre sus distintos actores con la Función Ejecutiva y la sociedad ecuatoriana” (Consejo Superior d. E., 2013), su principal propósito es garantizar la calidad de la educación superior, generando para ello, reglamentos que buscan regular los procesos de las instituciones universitarias, tal es el caso de la normativa N° RPC-SE-13-N° 051-2013, precisamente enfocado en la orientación del funcionamiento de las Unidades de Titulación Especial; este instrumento es analizado con mayor detalle en el marco referencial de este documento.

La carrera de Ingeniería de Sistemas acorde a los reglamentos establecidos por la Universidad Politécnica Salesiana maneja su proceso de titulación a través de documentos de hojas de cálculo, ya que no existe un sistema que gestione la información adecuadamente, esto causa una posible sobrecarga de horas asignadas a tutores, exceso de archivos, dispersión de la información lo que lleva a no tener información vital para la toma de decisiones operativas, malgasto de recursos.

Con el avance de internet las aplicaciones web han tomado relevancia debido a características como: la portabilidad que la aplicación brinda a diferentes sistemas operativos, bajo coste en actualización, no requiere el cliente de un software especializado para acceder a la aplicación, información centralizada, acceso a la última versión, manipulación de información en tiempo real y, acceso desde cualquier lugar, (Ferrer Martínez, 2014) por lo que se decidió la realización de una aplicación web con conexión a base de datos basado en un diagrama entidad relación para almacenar información de los proyectos de titulación, por tanto permite

gestionar temas, evolución de trabajo, hitos, sesión de usuarios, inscripciones al proceso de titulación, entre otros, con lo que se puede determinar el tiempo de desarrollo, porcentajes de avance de cada proyecto, con esto se busca reemplazar la gestión actual en el proceso de titulación.

Objetivos

Objetivo general.

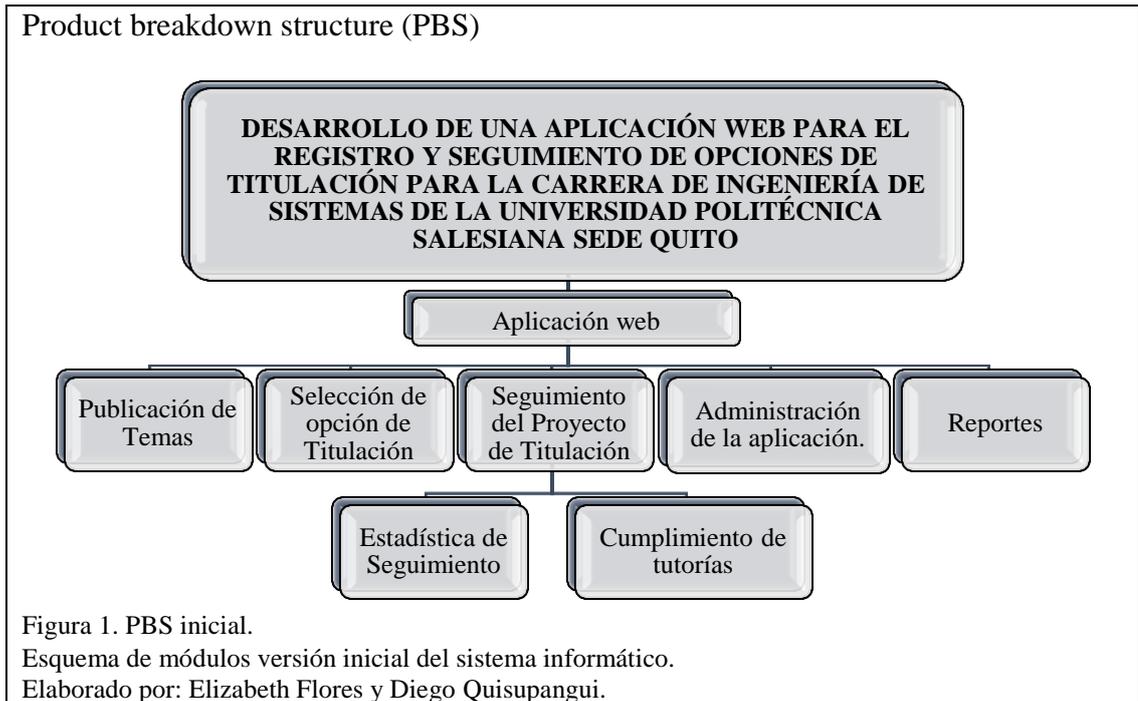
- Desarrollar una aplicación web que permitirá registrar y dar seguimiento a las opciones de titulación para la carrera de Ingeniería de Sistemas de la Universidad Politécnica Salesiana Sede Quito campus Sur.

Objetivos específicos.

- Estudiar el proceso de titulación tal como se está manejando actualmente en la Carrera de Ingeniería de Sistemas.
- Especificar los requerimientos de la Coordinación de la Unidad de Titulación de la Carrera de Ingeniería de Sistemas de la Universidad Politécnica Salesiana para que el software permita el registro y seguimiento del proceso de proyectos de titulación.
- Construir una aplicación web basada en Java Server Faces(JSF) para el uso de estudiantes, docentes, personal administrativo de la Coordinación de la Unidad de Titulación Especial de Sistemas(CUTS), en el proceso de registro de seguimiento de a proyectos de titulación.
- Realizar las pruebas a la aplicación web de manera que respondan favorablemente a los requerimientos y procesos establecidos.

Alcance

La aplicación permitirá satisfacer las necesidades de la gestión de los proyectos de titulación que tendrá como resultado un sistema robusto, en el siguiente diagrama de Estructura de desglose del producto o Product breakdown structure por sus siglas en inglés PBS se detallan los módulos que tendrá la aplicación web.



A continuación se describe de manera breve cada una de las partes detalladas en la Figura 1.

Publicación de temas: Se registran y se enlistan los temas propuestos por los docentes para los proyectos técnicos y los artículos académicos, los cuales serán seleccionados por los estudiantes para su desarrollo.

Selección de opción de titulación: el estudiante tendrá la posibilidad de seleccionar una de las tres opciones para el desarrollo en el proceso de titulación como son: Examen Complexivo, Proyecto técnico, Artículo académico

Seguimiento de Proyecto de titulación: se realiza el registro de: avance de los proyectos o artículos según los hitos entregados al tutor, seguimiento de los talleres

realizados en cada reunión, validación de asistencia del estudiante; con esta información se genera un reporte de asistencia, que podrá visualizar el tutor y la Coordinación de la Unidad de Titulación Especial.

Administración de la aplicación: el producto final contará con un módulo de administración para la gestión de usuarios y parametrización de opciones.

Reportes: se visualizará los siguientes reportes: asistencia de estudiantes a tutorías, estudiantes inscritos al proceso de titulación, avance de proyecto por docentes, información de temas desarrollados por docente.

Marco Metodológico

Análisis y Justificación de la Metodología

Las metodologías de desarrollo de software son conjuntos integrados de técnicas y métodos que permiten visualizar las actividades del ciclo de vida de un proyecto (INTECO, 2009). Dentro de las metodologías existe un grupo conocido como desarrollo ágil que se basa en la adaptabilidad de los procesos, un ejemplo de metodologías ágiles es Extreme Programming (XP) y Scrum, la metodología XP se basa en tener un software de calidad enfocado en la codificación del código, para que se ejecute correctamente y no tener riesgos se debe seguir las prácticas básicas del XP sin realizar cambio alguno (González Campos & Fernández Martínez, 2006), Scrum es una metodología que principalmente se enfoca en las funcionalidades principales las cuales se ejecutan en un periodo corto de tiempo realizando incrementos llamados Sprint.

El proyecto requiere que sea una aplicación web, adaptando el nuevo proceso de titulación permitiendo gestionar la información relevante a los proyectos de titulación; para solventar el desarrollo del proyecto se decide implementar la metodología Scrum ya que permite tener cambios en el avance del proyecto, debido a que hubo requisitos pocos claros por lo que es un nuevo proceso de titulación, de la metodología se usaron los artefactos Product backlog, Sprint backlog e Incremento, ya que como se menciona en el marco teórico solo se debe usar los artefactos que se necesite para el desarrollo del mismo.

El Product backlog generó una lista de entregables para cada Sprint, esta lista contiene los distintos módulos que tiene la aplicación (Tabla 1); el Sprint backlog, generó una tabla de tareas que el grupo de trabajo propuso al inicio del proyecto.

Las listas del Product backlog y el Sprint backlog cambian al transcurrir el proyecto en cada incremento, ya que cada vez se fueron refinando los requerimientos al tener

más conocimiento del proceso de titulación, las últimas versiones se encuentran adjuntas en los Anexos.

Para obtener los requerimientos para la aplicación se utilizó técnicas de recolección de información; la entrevista, para esto se realizó grabaciones en cada reunión realizada con el usuario.

En la primera reunión se realizó un análisis del seguimiento de los nuevos procesos de titulación para la aplicación, en los siguientes Sprint, se entregaba un incremento de cada módulo de la aplicación.

Tabla 1. Product backlog, versión 1.

Pila de producto: Aplicación web para el registro y seguimiento de proyectos de titulación			
Scrum Team: Franklin Hurtado, Elizabeth Flores, Diego Quisupangui			
Product Owner: Franklin Hurtado			
Development Team: Elizabeth Flores, Diego Quisupangui			
ID	Requerimientos	Prioridad	Estimación del valor
A1	Análisis del desarrollo de la página web	2	8
A2	Módulo de ingreso de temas por docente	3	9
A3	Módulo de selección de opción de titulación	3	8
A4	Módulo de seguimiento de proyectos de titulación	3	6
A5	Módulo de seguimiento de talleres	4	8
A6	Módulo de administración de sistema	4	6

Nota: Módulos iniciales del sistema informático.

Elaborado por: Elizabeth Flores y Diego Quisupangui.

Capítulo 1

1.1 Marco referencial o institucional

En el año 2014 la Universidad Politécnica Salesiana se acoge al nuevo reglamento del Consejo de Educación, con la creación de la Unidad de Titulación Especial (UTE) con la Resolución de Consejo Superior N° 173-09-2014-10-15.

La UTE se acoge al artículo 21 de la Resolución del Consejo Superior N° RPC-SE-13-N° 051-2013, en el cual indica que cada carrera debe tener por lo menos dos opciones de titulación. De acuerdo a esto, la UTE se acoge a las siguientes opciones: Proyecto Técnico, Artículo Científico y Examen Complexivo.

1.2 Marco Legal

La Resolución del Consejo de Educación Superior N° RPC-SE-13-N° 051-2013, es el documento en el cual se define el Reglamento de Régimen Académico y aplica a la educación superior pública y privada. En este caso se detalla que la Unidad de titulación es la que decide los desafíos que debe tomar el estudiante para poder finalizar sus estudios los cuales son un desarrollo de trabajo de titulación o la aprobación de un examen complexivo para que se validen los conocimientos adquiridos durante el tiempo de estudio. (Consejo Superior d. E., 2013)

La resolución N° 173-09-2014-10-15 de la Universidad Politécnica Salesiana, es el documento en el cual se aprueba la creación de la Unidad de Titulación Especial de la Universidad Politécnica Salesiana, por Disposición Transitoria Quinta del Reglamento de Régimen Académico del Consejo de Educación Superior, la misma se encarga de los nuevos procesos de titulación que deben ser basados en la investigación (trabajo de titulación) o en la aprobación de un examen complexivo, además en el Artículo 3 define “para la gestión y desarrollo de los procesos de la

Unidad de Titulación Especial, cada carrera de grado y programa académico de posgrado implementará una Coordinación de Titulación Especial” (Salesiana C. S., 2014).

La Coordinación de la Unidad de Titulación Especial de la carrera de Ingeniería de Sistemas (CUTS) dispone de los siguientes esquemas de investigación: proyecto técnico, artículo académico además de examen complejo que contribuyen al desarrollo de los profesionales, los saberes, la tecnología, y las ciencias. (Consejo Superior U. , 2014)

Descripción breve de la Universidad Politécnica Salesiana

La Universidad Politécnica Salesiana nace en el año 1994 la misma cuenta con su matriz en Cuenca y sedes en Quito y Guayaquil cada una con diferentes campus, es una institución autónoma de educación superior particular, católica, cofinanciada por el Estado, “la UPS se crea como una institución con derecho privado sin fines de lucro, con personería jurídica propia y autonomía responsable: académica, administrativa, financiera y orgánica” (Salesiana U. P., 2012).

La carrera de Ingeniería de Sistemas sede Quito campus sur con 15 años de creación no posee autonomía respecto al manejo de sus procesos de gestión, antes de realizar algún cambio dentro de la carrera estos se deben dirigirse directamente a las autoridades de la institución.

1.3 Marco Teórico

1.3.1 Metodologías Ágiles

El objetivo de las Metodologías Ágiles es enfocado a equipos de Software que se dedican al desarrollo de los mismos puedan ajustarse a los cambios que se den a través del desarrollo y así tener distintas alternativas mediante una documentación que se da en cada actividad.

En la actualidad existen varias metodologías para el desarrollo de software como: Scrum, Crystal Methodologies, Dynamic Systems Development Method (DSDM), Adaptive Software Development (ASD), Feature-Driven Development (FDD), Lean Development (LD), eXtreme Programming (XP).

En la Tabla 2 se mencionan las diferencias entre metodologías ágiles y tradicionales.

Tabla 2. Metodologías

Metodologías Ágiles	Metodologías Tradicionales
Se encuentra preparada para cambios durante el proyecto	No está preparada para los cambios
Se rige en equipo	Se rige externamente
Proceso menos controlado	Es un proceso controlado con normas y políticas
No existe contrato	Existe contrato
Cliente parte del equipo	Cliente interactúa con el equipo de desarrollo en reuniones
Pocos artefactos	Más artefactos
Pocos Roles	Más roles
Menos énfasis en la arquitectura del software	Arquitectura del Software es esencial y se expresa mediante modelos.

Nota: Comparativo Metodologías

Fuente: (Letelier Torres & Sánchez López, 2003).

1.3.2 Scrum

Scrum es un proceso para desarrollar software incrementalmente en entornos complejos que tienen requisitos poco claros o cambian con mucha frecuencia.

(Laínez Fuentes, 2015)

“Scrum es una metodología fácil de aprender sin embargo es todo un reto aplicarlo, este no ofrece un conjunto preciso de instrucciones a seguir de manera que el grupo tendrá que suministrar mucha de las predicciones y estimaciones, permite anticipar posibles tropiezos en sus planes evitando tiempo valioso y dinero a la vez” (Dimes, 2015)

Además según menciona José Rubén Laínez Fuentes en su libro de Desarrollo de Software Ágil “las metodologías ágiles reconocen la naturaleza empírica del software y están preparadas para acoger los cambios frecuentes”. (Laínez Fuentes, 2015).

Artefactos.

“Los Artefactos de Scrum son subproductos de las actividades del marco de trabajo que le brinda dirección y transferencia al equipo”. (Navarro Cadavid, Fernández Martínez, & Morales Vélez, 2013)

Peter Hundermark, menciona que los artefactos que deben ser usados durante el desarrollo de un proyecto mediante Scrum son aquellos que pueden ser realmente valiosos para el desarrollo del proyecto con el propósito de eliminar documentación no relevante.

Entre los artefactos de Scrum se pueden listar:

- Product Backlog, lista de requisitos de usuario, que puede ser cambiado en cualquier momento del proyecto.
- Sprint Backlog, lista de tareas que el equipo debe cumplir para el siguiente incremento del producto.

- **Incremento.** Es el final de un Sprint y el mismo debe estar terminado para comenzar uno nuevo, para que pueda ser utilizado por el Dueño del Producto y ser validado por el mismo y así decidir si se utilizará o se realizará cambios.

Iteraciones de Scrum

Sprint: “una iteración o ciclo repetitivo de trabajo semejante, que produce un incremento de producto o sistema. No mayor de un mes, y usualmente mayor de una semana. La duración es fija a lo largo de todo el trabajo, y todos los equipos que trabajan en el mismo sistema o producto usan la misma duración de ciclo” (Deemer, Benefield, Larman, & Vodde, 2012).

Reunión de Sprint, se planifica en dos partes, la primera se selecciona las funcionalidades de la lista del producto que serán realizadas en el nuevo Sprint por parte de todo el equipo del Scrum; en la segunda parte el equipo de desarrollo se organiza para definir la lista de trabajo del Sprint (Sprint Backlog) y lograr un mayor entendimiento de los funcionalidades requeridas por el Product owner.

Revisión de Sprint, reuniones en las que se presenta el incremento al usuario.

Pilares del Scrum

Tres pilares soportan toda la implementación del control de procesos empírico, esto se define como el conocimiento que viene de la experiencia y la toma de decisiones basadas en lo que es conocido.

Transparencia, los procesos deben ser claros y concisos con un estándar de comunicación para el entendimiento entre los responsables.

Inspección, se debe revisar con frecuencia los artefactos de Scrum para verificar el progreso y detectar variaciones.

Adaptación, se investiga que los procesos no se desvíen del límite ya que si da el caso el producto será inaceptable y esto causa que deba ser ajustado y corregido.

(Sutherland & Schwaber, 2013)

Equipo Scrum

Está conformado por tres actores principales: equipo de desarrollo, propietario de producto y el Scrum master.

El equipo de desarrollo (*Development Team*) es autoorganizado y multifuncional, se encarga de entregar productos de forma iterativa e incremental, además de la presentación de versiones funcionales en cada Sprint al propietario.

Propietario del producto (*Product Owner*) es el responsable de maximizar el valor del producto y del trabajo del Equipo de desarrollo, es responsable de gestionar la Lista del producto (Product Backlog),

Scrum Master es el responsable de asegurar el aprendizaje y la aplicación de Scrum al negocio. (Sutherland & Schwaber, 2013)

1.3.3 Técnicas de recolección de información

“Es un procedimiento amplio que se especifican en ciertas técnicas de alcance general. Estas prescriben unos procedimientos relativos a varias acciones que debe realizar el investigador.” (Yuni & Urbano, 2006).

Los instrumentos para la obtención de información son mecanismos o dispositivos que utiliza el investigador para generar datos. Estos instrumentos pueden ser de carácter mecánico, cuestionarios, cámara de video, guía de observación, etc.

Entrevista

La entrevista es un método que comprende la reunión de datos a través de una interacción oral directa entre individuos, es importante para la recopilación de requerimientos durante el desarrollo de un sistema porque la información obtenida es relevante ya que proviene de las peticiones o necesidades del usuario.

Las entrevistas revelan la información sobre:

- Opiniones del entrevistado.
- Sensaciones del entrevistado.
- Sobre el estado actual del sistema.
- Metas de organización y personales
- Procedimientos informales

Fases de la entrevista

- Producción: comprende los requisitos previos para la ejecución de la entrevista, estos requisitos son: Elegir al entrevistador, Recoger información sobre el entrevistado o tema a tratar, elaborar el cuestionario.
- Realización: Incluye formas de cortesía, uso de preguntas breves, correctas y expresadas de forma clara.
- Registro: se puede utilizar video, audio digital, anotaciones, todo esto bajo el consentimiento del encuestado.
- Informe: se da a conocer el contenido de la entrevista.

(Soria Solís & Condor Tinoco, 2015)

“En algunos casos, como mecanismo de recogida de información se utilizan medios electrónicos, así tenemos el caso de las grabaciones de audio recogidas en la entrevista o tomas de video realizadas.” (Gil Pascual, 2011)

1.3.4 BPMN

El Modelo y Notación de Procesos de Negocio o por sus siglas en inglés Business Process Model and Notation (BPMN) “es una notación gráfica que describe la lógica de los pasos de un proceso de negocio” (Suite, 2016).

“El objetivo de la BPMN es desarrollar una notación gráfica que permitiera automatizar en forma más rápida los procesos, para entender la BPMN se debe conocer el concepto de BPN, es un enfoque sistemático para identificar levantar documentar, diseñar, ejecutar, medir y controlar tanto los procesos manuales como automatizados, con la finalidad de lograr a través de sus resultados en forma consistente los objetivos del negocio que se encuentran alineados con la estrategia de la organización” (Suite, 2016).

La notación que contiene BPMN tiene una serie de símbolos y a través de reglas específica como se deben relacionar estos objetos.

1.3.5 Product Breakdown Structure (PBS)

“La Product Breakdown Structure (PBS), es una representación del producto o sistema a desarrollar en el proyecto en forma de árbol que describe al producto en diferentes niveles desde el más alto que representa a todo el producto y descendiendo progresivamente hacia subsistemas, elementos de configuración y componentes (niveles más bajos)” (Escribano Romero, 2011)

1.3.6 Diagrama de Casos de uso

“Los Casos de uso proveen una forma simple y comprensiva de identificar y capturar las historias narradas por los stakeholders o las personas que tienen el interés en el resultado del proyecto, permitiendo que los requerimientos del sistema sean

fácilmente capturados, compartidos y comprendidos, además de ser ejecutables y testeables” (Girbal , Gardella , & Peñalva , 2013).

“El enfoque de Casos de uso permite establecer una correspondencia directa con los llamados métodos ágiles para el desarrollo de software en la cual el sujeto que aprende verá cómo los contenidos previos y las actividades de los Casos de Uso se integran particularmente con una metodología ágil como Scrum, un marco de trabajo para la gestión y desarrollo de software basado en un proceso iterativo e incremental” (Girbal , Gardella , & Peñalva , 2013).

1.3.7 Diagrama de secuencia

M. Jesús Marco Galindo afirma que el diagrama de secuencia permite modelar el comportamiento del sistema, conociendo las funcionalidades del sistema, mostrando un conjunto de mensajes (interacciones) mostrando fácilmente la ordenación temporal de los mensajes.

1.3.8 Diagrama de clases

El diagrama de clases, además de ser de uso extendido, también está sujeto a la más amplia gama de concepto de modelado, describe los tipos de objetos que hay en el sistema y diversas clases de relaciones estáticas que existen entre ellos, también muestra los atributos y operaciones de una clase. (Martin Fowler, Kendall Scott, 1999)

1.3.9 Herramientas de desarrollo

Aplicación Web

Una aplicación web es un conjunto de herramientas orientadas al usuario con el fin de que este pueda acceder a un servidor mediante el uso del navegador que se conecta a internet (Cardador Cabello, 2014), tiene una arquitectura de tres niveles

cliente (navegador web), servidor de aplicaciones (servidor web), servidor de datos (base de datos).

Antonio Luis Cardador Cabello menciona las ventajas de una aplicación web:

- No es necesario instalar nada de parte del cliente.
- No es necesario que el cliente actualice nada.
- No hay problema de actualizaciones de versiones. Todas usan la misma versión.
- Centralización de información.
- No se requiere un sistema operativo determinado, ni software ni hardware determinado.
- Se puede acceder desde un ordenador que tenga conexión a internet o desde red interna.

Para el desarrollo de la aplicación web se consideró lo siguiente: la herramienta de desarrollo es Netbeans, el motor de base de datos en Postgresql, el servidor de aplicaciones es Glassfish y el framework es JSF con componentes en Primefaces.

JavaServer Faces JSF

JSF es una especificación de Java Enterprise Edition, framework para el desarrollo web que simplifica el desarrollo web basado en Java Servlet y JavaServer Pages JSP, además de soportar el Modelo Vista Controlador o por sus términos en inglés Model View Controller (MVC), y puede implementar lenguaje basado en XML para interfaz de usuario.

(Rios, 2015) En su libro indica que herramientas incluye dentro de JSF:

“Un conjunto de APIs para representar componentes de una interfaz de usuario y administrar su estado, manejar eventos, validar entrada, definir un esquema de navegación de las páginas y dar soporte para internacionalización y accesibilidad,

conjunto por defecto de componentes para la interfaz de usuario, dos bibliotecas de etiquetas personalizadas para Java Server Pages que permiten expresar una interfaz de Java Server Faces dentro de una página JSP, un modelo de eventos en lado del Servidor, administración de Estados, Beans administrados” (Rios, 2015).

Hibernate

Hibernate es un framework bajo los términos de la licencia GNU LGPL, permite la interacción de aplicaciones Java con las bases de datos, mediante el uso de Mapeo Relacional de Objetos, ORM por sus siglas en inglés (Object Relational Mapping) que se encarga de mapear objetos Java con una base de datos relacional esto para la programación orientada a objetos, simplifica la creación, manipulación y acceso de datos.

Primefaces

Es una librería de código abierto de componentes para JSF, que facilitan la creación de aplicaciones web, entre sus principales características están: soporte Ajax, soporte para crear aplicación web para móviles, compatible con otras librerías, uso de javascript, entre otros.

Glassfish

Es un servidor de aplicaciones de software libre que permite ejecutar aplicaciones Java EE, bajo la licencia dual CDDL y GNU GPL.

Es de arquitectura modular, es decir, que al momento de descargar y realizar la instalación se puede instalar las aplicaciones seleccionadas que buscan optimizar memoria y espacio en disco.

1.4 Factibilidad del proyecto.

Para la factibilidad del desarrollo del proyecto se realiza un análisis técnico, económico y operativo que identifica las necesidades para la realización del mismo.

1.4.1 Factibilidad técnica.

Para la factibilidad técnica se realizó un estudio de herramientas necesarias para el desarrollo del proyecto con un enfoque de hardware y software. En la tabla 3. Muestra los recursos técnicos que se usaron para el desarrollo del proyecto.

Tabla 3. Factibilidad técnica

No.	Recurso	Características
1	Acceso a Internet.	Internet de alta velocidad
2	Portátil (2)	- HP, Procesador Core I5, 6GB de RAM, DD 500 GB. - Acer, Procesador Core I3, 3GB de RAM, DD 300 GB.

Nota: Características de las computadoras que se usó en el desarrollo del proyecto.
Elaborado por: Elizabeth Flores y Diego Quisupangui.

Para la implantación del sistema se requiere un servidor con las características siguiente:

Tabla 4. Recursos del sistema

Recurso	Características
Sistema operativo	Linux, distribución Centos 6.5
Base de datos	PostgreSQL
Navegador	Mozilla Firefox, Google Chrome.

Nota: Recursos para la instalación del sistema.
Elaborado por: Elizabeth Flores y Diego Quisupangui.

1.4.2 Factibilidad económica.

En la factibilidad económica se revisa los recursos económicos para el desarrollo del sistema en este caso se tiene costos generales que fueron cuadernos, hojas, lápices, etc.; costos personales fueron pasajes de transporte y horas de desarrollo en las cuales se utilizó 8 horas diarias.

1.5 Pruebas

Las pruebas a realizar en la aplicación son:

Funcionales: consisten en asegurar el funcionamiento del sistema desarrollado acorde a los requisitos del cliente, para esto se utilizó las pruebas de caja blanca donde se realiza un desarrollo de pasos del estado de la aplicación en varios puntos.

Se genera una tabla con los datos: proceso, actor, prerequisite, escenario, pasos, resultado esperado, resultado alcanzado.

No Funcionales: válida a la aplicación si es confiable, usable y libre de errores (Castellanos Rodríguez & Beovides Luis, 2013), para estas pruebas se utilizó la herramienta JMeter, y el nuevo sistema de usabilidad SIRIUS.

Sirius: “proporcionan a desarrolladores y evaluadores un conjunto de criterios o pautas para considerar en el proceso de desarrollo y evaluación de un sitio web, los aspectos y criterios a evaluar para detectar los errores de usabilidad son: Aspectos generales, Identidad e Información, Estructura y Navegación, Rotulado, Layout de la página, Entendibilidad y facilidad en la interacción, Control y retroalimentación, Elementos multimedia, Búsqueda, Ayuda” (Suárez Torrente , 2011).

En cada punto a evaluar se tendrá distintos aspectos a los cuales se les asignará un valor durante el proceso de evaluación.

Capítulo 2

Análisis y Diseño

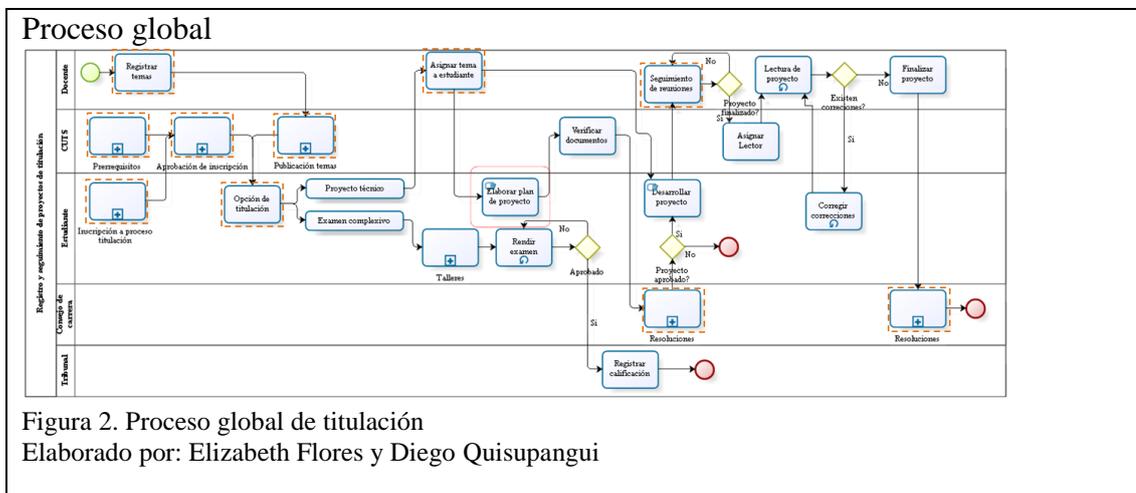
Debido a que Scrum fue la metodología seleccionada para el desarrollo de la aplicación informática, se usaron iteraciones en el proceso para crear componentes de software, entonces el análisis y diseño, debido a que son disciplinas clave en el proceso de desarrollo, también se ajustaron a la perspectiva de desarrollo incremental e iterativa, lo cual significó la generación de artefactos que fueron refinados a lo largo del proyecto y en la medida que el incremento iba madurando. A pesar de que Scrum, tal como se mencionó en la descripción de la metodología no contempla artefactos específicos para el análisis y diseño, tampoco restringe el uso de estos, por ello en el presente proyecto se utilizaron artefactos como: diagrama de proceso, casos de uso, diagrama de actividades y diagrama de clases.

2.1 Diagrama de Procesos

En primera instancia se creó el esquema PBS que brinda una perspectiva funcional y de alto nivel del software, a partir de este diagrama se alimentó al Product Backlog, mismo que organiza el trabajo requerido para el desarrollo de la aplicación, estos componentes se los puede visualizar con más detalle en la introducción de este documento; con el entendimiento de los artefactos anteriormente mencionados se procedió a realizar el levantamiento de información del proceso con el cual se creó el diagrama global de proceso de negocio con la notación BPMN (Business Process Modeling Notation), que en el contexto del presente proyecto fue utilizada para describir los procesos debido a que esta prácticas, tal como lo mencionó en el marco teórico son especialmente útiles para identificar las funcionalidades que van a formar parte del software.

En la Figura 2 se puede observar el proceso global de proyectos de titulación, mismo que sirvió para identificar las primeras funcionalidades que se manejaban de forma manual, entre las que fueron automatizadas se puede mencionar: Publicación de tema, Prerrequisitos, Registrar temas, Aprobación de inscripción, Inscripción a proceso de titulación, Asignar tema a estudiante, Seguimiento de reuniones para titulación, Asignar lector, Resoluciones; así mismo, el mencionado diagrama, fue útil para identificar los actores involucrados: Docente, CUTS, Estudiante, Consejo de Carrera; cabe recalcar que se realizó levantamiento de información y modelamiento ya que no existía un diagrama que refleje este proceso, en la avance del proyecto los artefactos sufrirán cambios para mejorar el entendimiento, en el Anexo 1. Diagramas se puede visualizar los subprocesos con mayor detalle.

En la Figura 2 se puede observar los subprocesos que están resaltados por cuadros de color rojo, estos serán automatizados a través del sistema informático.



2.2 Requerimientos del sistema

La metodología Scrum sugiere artefactos que contienen las tareas que deben ser desarrolladas para generar los incrementos de software, en este proyecto la especificación de requerimientos se la realizó a través de las reuniones con el *Product owner* que son propias de la metodología Scrum conjuntamente con la

técnica de la entrevista, mismas que retroalimentaron para el avance del proyecto, por lo que no se registró en un documento específico, se documentó y almacenó en medios magnéticos en este caso grabaciones, aclarando que se obtuvieron con el respectivo permiso del *Product owner*; la identificación de funcionalidades que proveen los diagramas de proceso del negocio anteriormente detallados puede ayudar y facilitar la especificación de requerimientos de cada proceso, cabe mencionar que cada uno de los requerimientos del software fueron afinados y refinados en el desarrollo de las respectivas iteraciones.

En la Tabla 5 se pueden observar los requerimientos más relevantes de cada subproceso del negocio, que fueron cambiando en el transcurso de cada Sprint, a continuación las últimas versiones de los requerimientos del sistema informático desarrollado.

La Tabla 5 contiene la siguiente estructura: **Código:** que enlista el número de requerimientos específicos agrupado por proceso, esto fue conveniente para identificar cada uno y determinar a qué proceso pertenecen; **Proceso:** campo que contiene el nombre del proceso que fue automatizado; **Requerimiento:** campo que contiene la funcionalidad; **RN:** Requerimiento nuevo, es el requerimiento que fue encontrando en el desarrollo de los Sprint; **RF:** Requerimiento refinado, es el requerimiento que cambio a través de un análisis más detallado; y **Descripción:** campo que sirve para ampliar el entendimiento de la funcionalidad.

Tabla 5. Especificación de requerimientos

Código	Proceso	Requerimiento	RN	RF	Descripción
ER01.1	Prerrequisito	Registro de periodo, convocatoria de temas, convocatoria a inscripción, requisitos	X	X	El periodo es esencial para la creación de convocatorias y requisitos

ER01.2		Asignación de docentes a periodo	X		Los docentes son los que participarán en el periodo
ER01.3		Registro de horas asignadas, vigentes y de lectura,	X	X	
ER02.1	Publicación de tema	Registro, envío de tema	X	X	Para el envío es necesario que exista una convocatoria habilitada
ER02.2		Aprobación, publicación de tema		X	
ER02.3		Registro de informe		X	Se crea un informe de cambios del tema
ER03.1	Inscripción a proyecto de titulación	Al momento de la inscripción, seleccionar opción de titulación, registrar requisitos	X	X	Los requisitos serán preguntas referentes al estado académico del estudiante
ER03.2		Validación, aprobación de inscripción		X	Se realizará de acuerdo a ciertos requisitos
ER04.1	Aprobación de proyecto	Asignación de tema a estudiante	X	X	Solo a estudiantes aprobados o en lista de espera
ER04.2		Asignación de lector de plan	X		
ER04.3		Pre revisión de documentos de plan	X		
ER04.5		Registro de resolución	X		
ER05.1	Seguimiento de proyecto	Registro, validación de cita		X	Serán las citas para las reuniones en el transcurso del avance del proyecto
ER05.2		Registro, validación de hitos		X	Serán fechas y entregables del software

ER05.3		Asignación de lector proyecto	X		
ER05.4		Finalizar proyecto	X		Docente tutor evalúa la terminación del proyecto
ER06.1	Resolución de proyecto	Cambio de proceso en proyecto	X	X	
ER06.2		Registro de resolución por tipo	X	X	
ER07.1	Configuración	Parametrización	X	X	El software capaz de cambiar parámetros de configuración
ER07.2		Manejo de perfiles, sesiones de usuario	X		
ER07.3		Notificaciones por correo electrónico	X		
ER07.4		Manejo de estados en tema, inscripción, asignación de lector, asignación de tema a estudiante	X		

Nota: Especificación de requerimientos más relevantes agrupados por proceso.
Elaborado por: Elizabeth Flores y Diego Quisupangui.

2.2.1 Product Backlog

En la Tabla 6 se puede observar los cambios que se dieron en el transcurso del avance del proyecto en cuanto a la organización del trabajo por módulos.

La Tabla 6 contiene la siguiente estructura: **ID**: es el campo formado por un código que sirvió para diferenciar los Sprint que asociados al Sprint backlog; **Requerimientos**: es el campo que contiene el nombre de los módulos del sistema informático que se realizaron; **Prioridad**: es el campo que establece la importancia de cada requerimiento de la pila del producto tomando en cuenta desde 1 al 5 (5 prioridad más alta); **Estimación del valor**: es el esfuerzo dado al comenzar el

proyecto de cada requerimiento, este valor va cambiando al transcurriendo el desarrollo.

Tabla 6. Product backlog

Pila de producto:			
Aplicación web para el registro y seguimiento de proyectos de titulación			
ID	Requerimientos	Prioridad	Estimación del valor
A1	Análisis de proceso	3	8
A2	Módulo de registro, publicación de temas	3	9
A3	Módulo de inscripción a proceso de titulación	3	8
A4	Módulo de pre-requisito	3	8
A5	Módulo de aprobación de tema	3	8
A6	Módulo de seguimiento de proyecto	3	6
A7	Módulo de resoluciones	4	8
A8	Módulo de administración de sistema	4	6

Nota: Módulos realizados en su versión final del sistema informático.

Elaborado por: Elizabeth Flores y Diego Quisupangui.

En la Tabla 7 se puede visualizar las tareas que se dieron en el desarrollo de software, destacando que es la última versión de este artefacto ya que en el avance del proyecto en cada iteración que se realizaba se dieron cambios relevantes lo que hizo que se aumentaran, eliminaran y cambiarán tareas.

La Tabla 7 consta de los siguientes campos: **RQ**: es el código de la pila del producto al que pertenece; **Tarea de Sprint**: contiene las tareas realizadas; **IM**: Importancia: es un valor de 1 al 10 entre más importante sea la tarea va aumentando su valor; **ET**: Estimación de esfuerzo inicial (días): es un valor en días del tiempo estimado en que se terminara la tarea por parte de los desarrolladores este tiempo puede cambiar según avanza el proyecto.

Tabla 7. Sprint backlog

Pila de entrega:			
Aplicación web para el registro y seguimiento de opciones de titulación para la Carrera de Ingeniería de Sistemas de la Universidad Politécnica Salesiana Sede Quito.			
RQ	Tarea del Sprint	IM	ET
A1	Definir ambiente de desarrollo	5	3
	Diseño de base de datos	9	3
	Diagrama de Procesos	10	3
Total de días:			9
A2	Análisis del proceso de publicación de temas	6	2
	Investigación manejo de temas	5	2
	Prototipo funcional del registro de los temas	7	2
	Registro de tema, descripción	1	1
	Registro de número de estudiantes para proyecto de titulación	3	2
	Límite de caracteres en tema, descripción	2	3
	Análisis de proceso de envío a publicación	3	2
	Registro de convocatoria al enviar el tema a publicación	2	1
	Listado de temas por docente	4	2
	Aprobación de tema	3	1
	Publicación de tema	3	2
	Manejo de notificaciones de temas	7	5
	Manejo de estados para temas	7	5
	Habilitar edición por estado de tema	5	2
Registro de informe de cambios para tema	5	2	
Total de días:			34
A3	Registro de convocatoria para inscripción	5	3
	Registro de usuario mediante interfaz	7	3
	Validación de cédula de estudiante	6	1
	Validación de email	4	1
	Visualización de temas publicados	4	1
	Registro de inscripción	3	2
	Selección de opción de titulación	5	1
	Registro de requisitos	6	2
	Aprobación de inscripción	7	1
	Notificación de estado de inscripción	7	3
	Manejo de estados para estudiantes aprobados	3	3
Total de días:			21
A4	Registro de periodo	8	2
	Registro de convocatorias a tema, inscripciones	7	5

	Asignar docentes al periodo	6	3
	Registrar requisitos	5	2
	Validación de campos numéricos	7	4
	Visualización de la opción escogida	8	3
Total de días:			19
A5	Asignar estudiante aprobado a tema	10	3
	Listar estudiantes aprobados	7	3
	Asignación de lector de plan a tema	7	3
	Notificación de acciones realizadas en tema	5	2
	Pre revisión de documentos	7	3
	Manejo de estados para tema	7	2
	Registro de tipo de resoluciones	8	1
	Registro de resolución por tema	10	2
	Validación de campos para resolución	10	3
Total de días:			22
A6	Registro de hitos	9	3
	Validación de hitos	5	1
	Registro de cita	7	3
	Validación de campos de hitos	5	3
	Validación de campos de cita	8	3
	Notificación de citas	7	2
	Asignación de lector de proyecto	5	2
	Notificación de lector	6	1
	Liberación de lector de plan	7	1
	Manejo de estados en cita	5	2
	Manejo de estados en hitos	9	6
	Total de días:		
A7	Registro de resolución	8	5
	Acción de tipo de resolución	10	10
	Cambio de proceso por tipo de resolución	10	5
	Notificación de resolución	7	3
	Validación de campos por tipo de resolución	5	10
Total de días:			33
A8	Control de usuarios	6	7
	Parametrización de opciones	5	10
	Configuración de notificaciones	7	7
	Configuración de mail	5	7
	Crud de usuarios	7	7
	Editar datos de cuenta personal	5	5
	Roles y perfiles de seguridad	5	10
Total de días:			53

Nota: Tareas de Sprint

Elaborado por: Elizabeth Flores y Diego Quisupangui.

2.3 Diagramas de casos de uso

Los diagramas de casos de uso permitieron tener un conocimiento del funcionamiento del producto; ayudaron a lograr una mayor comprensión de las historias (funcionalidades) de los actores y además sirvieron como modelo para la creación de los componentes funcionales del sistema informático; para un mayor entendimiento en cuanto al software se detallan los diagramas de casos de uso del sistema informático en las últimas versiones los cuales fueron cambiando en cada Sprint.

La Tabla 8 está compuesta por los siguientes campos: **Rol**: es el campo que enlista los actores identificados para el sistema informático; **Tarea**: describe las tareas que cada rol va a realizar en el sistema.

Tabla 8. Descripción de los actores del sistema.

Rol	Tarea
Administrador	Usuario que administra el sistema, es el que tiene acceso a las configuraciones necesarias para que el sistema tenga una funcionalidad correcta.
Estudiante	Usuario que ingresa al sistema, realiza la inscripción con la selección de una opción de titulación, visualiza el listado de temas y artículos publicados, realiza el registro de hitos y valida las tareas planteadas por el docente tutor.
CUTS	Usuario que ingresa al sistema, registra periodo, convocatorias para inscripción, publicación de temas, asignar docentes a periodo, revisar inscripciones, asignar revisores, lectores de plan y proyectos, revisar temas de los Proyectos técnicos y Artículos académicos creados y enviados por el docente, publicar temas, registrar resoluciones.
Docente	Usuario que ingresa al sistema, registra los temas para los Proyectos técnicos y Artículos académicos, envía temas para revisión de aprobación a publicación, asigna tema a estudiante(s) previamente seleccionado(s) por la CUTS, registro de citas, tareas para la Evolución de trabajo, valida los Hitos ingresados por el estudiante, revisa temas asignados para la publicación.

Nota: Funciones que realiza cada rol en la aplicación.

Elaborado por: Elizabeth Flores y Diego Quisupangui.

A continuación se describen los casos de uso del sistema informático, cada uno formado por un diagrama y una tabla en la cual contiene la siguiente estructura: **Caso**

de uso: nombre del caso de uso actual; **Actores:** involucrados en la funcionalidad;

Curso normal: secuencia de pasos (proceso) que sigue el actor por el sistema;

Alternativa: opciones a seguir dentro del sistema;

2.3.1 Registrar usuario

Proceso que el usuario (estudiante) realiza para registrarse en el sistema, ingresando datos personales, datos de usuario, finalizando con la confirmación del registro.

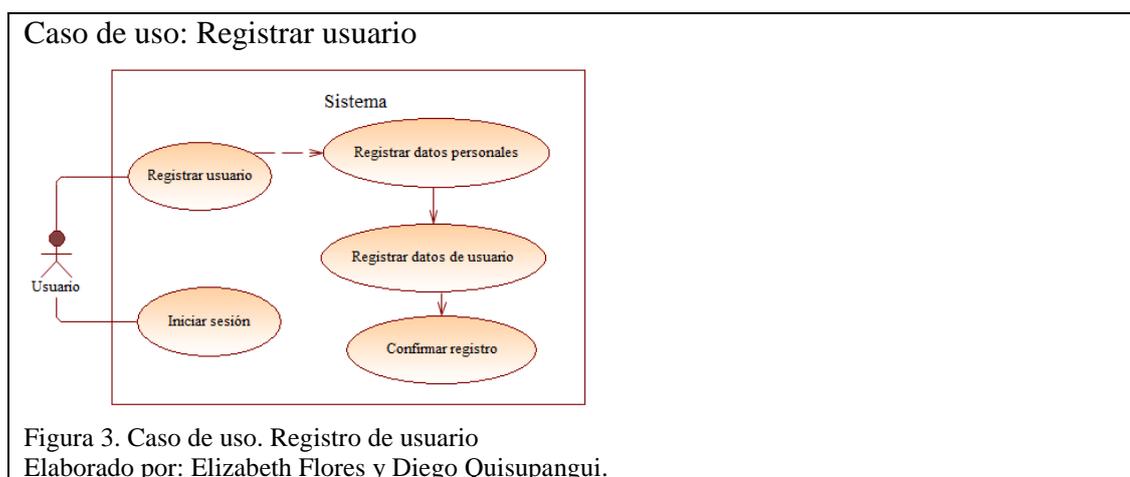


Tabla 9. Registrar usuario

Caso de uso: Registrar usuario	
Actores: Estudiante, Docente, CUTS, Administrador	
Curso Normal	Alternativa
1. Registrar usuario	1.1. Registra datos personales 1.2. Registra datos usuario 1.3. Confirma registro
2. Inicia sesión	

Nota: descripción del caso de uso registro de usuario.
Elaborado por: Elizabeth Flores y Diego Quisupangui.

2.3.2 Publicación de temas

Es el proceso que se realiza para la publicación de un nuevo tema ya sea de un Proyecto técnico o de un Artículo académico, inicia con la convocatoria por parte de la CUTS, una vez habilitada la convocatoria el docente registra y envía el tema para la revisión, la CUTS asigna a un docente para revisar el nuevo tema, el docente

revisa el tema y emite un informe con el resultado de la revisión dependiendo de este resultado se publica o se realizan cambios al tema, el estudiante enlista los temas publicados.

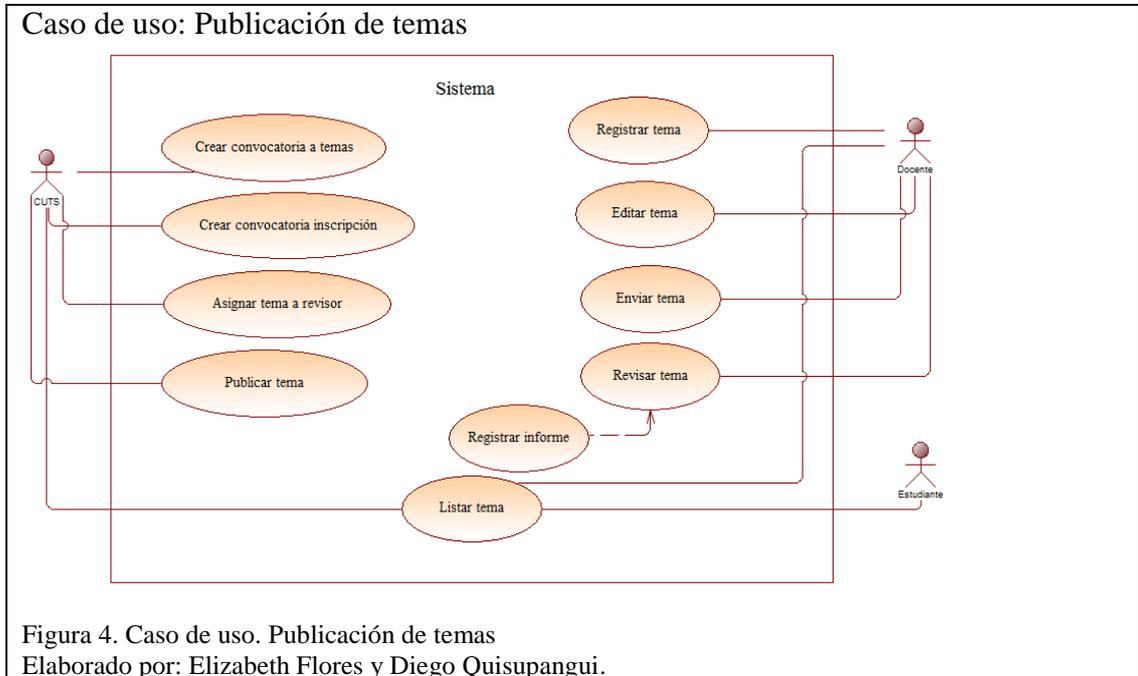


Tabla 10. Publicación de temas

Caso de uso: Publicación de temas	
Actores: CUTS, Docente, Estudiante.	
Curso Normal	Alternativa
1. Crea convocatoria a temas (CUTS).	
2. Crea convocatoria inscripción (CUTS).	
3. Crea tema (Docente)	
4. Envía tema (Docente).	
5. Asigna tema a docente (CUTS)	
6. Revisa tema (Docente)	
7. Registra informe (Docente)	7.1 Emite informe 7.2 Aprueba tema 7.3 Emite cambio de tema
8. CUTS pública tema	8.1 Publicación de tema
9. Visualizan temas los actores.	

Nota: Descripción del caso de uso publicación de temas.
Elaborado por: Elizabeth Flores y Diego Quisupangui.

2.3.3 Inscripción

Proceso que realiza el estudiante para la inscripción a la Unidad de titulación que requiere que se elija una opción de titulación, el registro de requisitos e información relevante para la aprobación de inscripción.

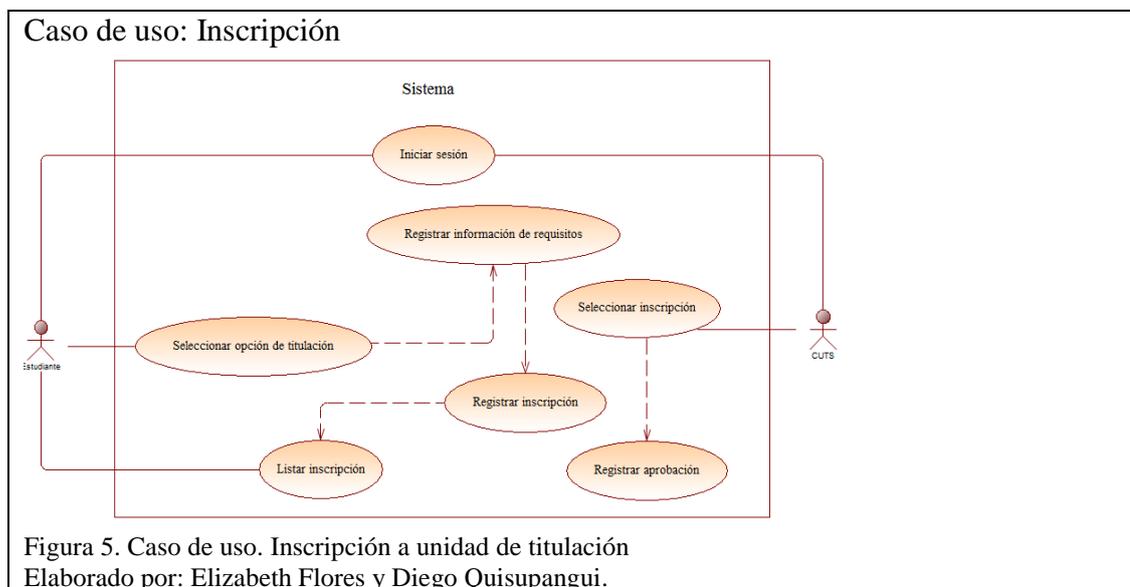


Tabla 11. Inscripción a opción de titulación

Caso de uso: Inscripción a opción de titulación	
Actores: Estudiante, CUTS	
Curso Normal	Alternativas
1. Inicia sesión (Estudiante, CUTS)	
2. El estudiante ingresa al Sistema	
3. Selecciona opción de titulación	3.1 Proyecto técnico 3.2 Artículo académico 3.3 Examen complejo
4. Ingresar información de requisitos	
5. Registra inscripción	
6. Selecciona inscripción (CUTS).	6.1 Visualiza información relevante del estudiante
7. Registra aprobación de inscripción (CUTS)	7.1 Registro de estados de la inscripción
8. El estudiante y la CUTS visualiza el estado de la inscripción.	

Nota: Descripción del caso de uso inscripción a la opción de titulación.
Elaborado por: Elizabeth Flores y Diego Quisupangui.

2.3.4 Aprobación de proyecto

Proceso que realiza en conjunto con el docente, estudiante y la CUTS, el docente visualiza los temas publicados, asigna un tema al estudiante previamente aprobados por la CUTS; La CUTS visualiza temas asignados, asigna lector de plan, valida documentos necesarios para la aprobación, registra datos de resolución emitida, el estudiante visualiza la resolución, si el tema es aprobado se inicia el desarrollo como proyecto de titulación.

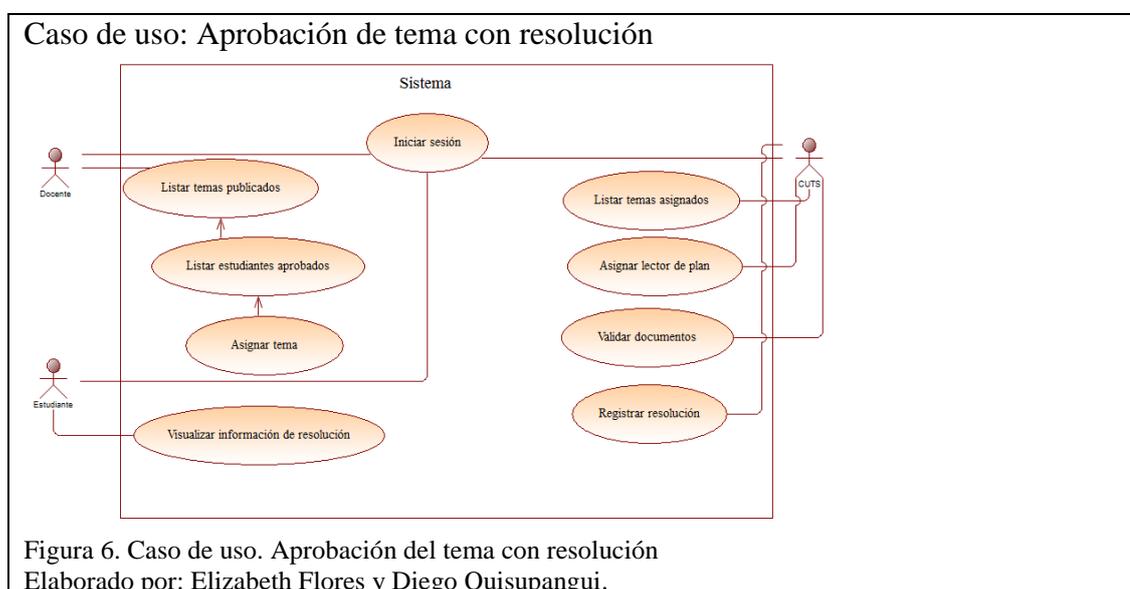


Tabla 12. Aprobación de proyecto

Caso de uso: Aprobación de proyecto	
Actores: Estudiante, CUTS, Docente	
Curso Normal	Alternativas
1. Docente visualiza temas publicados	
2. Docente visualiza estudiantes aprobados	
3. Docente asigna tema a estudiante	
4. CUTS Visualiza temas asignados	
5. CUTS asigna lector de plan	
6. CUTS valida documentos pre-aprobación del tema a proyecto	
7. CUTS registra resolución	
8. Los actores visualizan resoluciones de temas aprobados a proyecto	

Nota: descripción del caso de uso aprobación del tema con resolución.
Elaborado por: Elizabeth Flores y Diego Quisupangui.

2.3.5 Seguimiento de proyecto

En este proceso se detalla el seguimiento de los proyectos de titulación con el que se realiza el registro de citas y tareas para la Evolución de trabajo que contiene el avance del proyecto, además de registro de hitos para la validación del cumplimiento del cronograma.

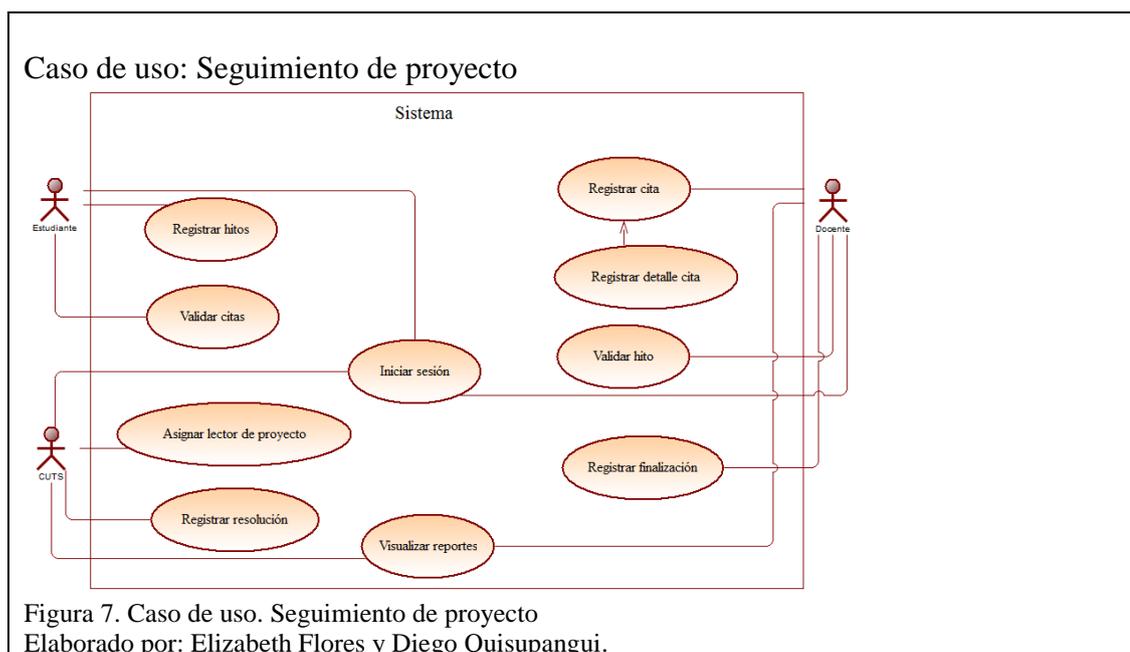


Tabla 13. Seguimiento de proyecto

Caso de uso: Seguimiento de proyecto	
Actores: Estudiante, CUTS, Docente	
Curso Normal	Alternativas
1. Estudiante registra datos del tema aprobado con resolución.	
2. Estudiante registra Hitos.	
3. Docente registra cita	
3. Estudiante válida tareas de cada sesión de trabajo	
5. Docente valida Hitos	
6. Docente visualizar reportes de sus proyectos	
7. Docente registra finalización de cada uno de sus proyectos	
8. CUTS registra resoluciones	
9. Al finalizar el proyecto la CUTS asigna lector al proyecto	

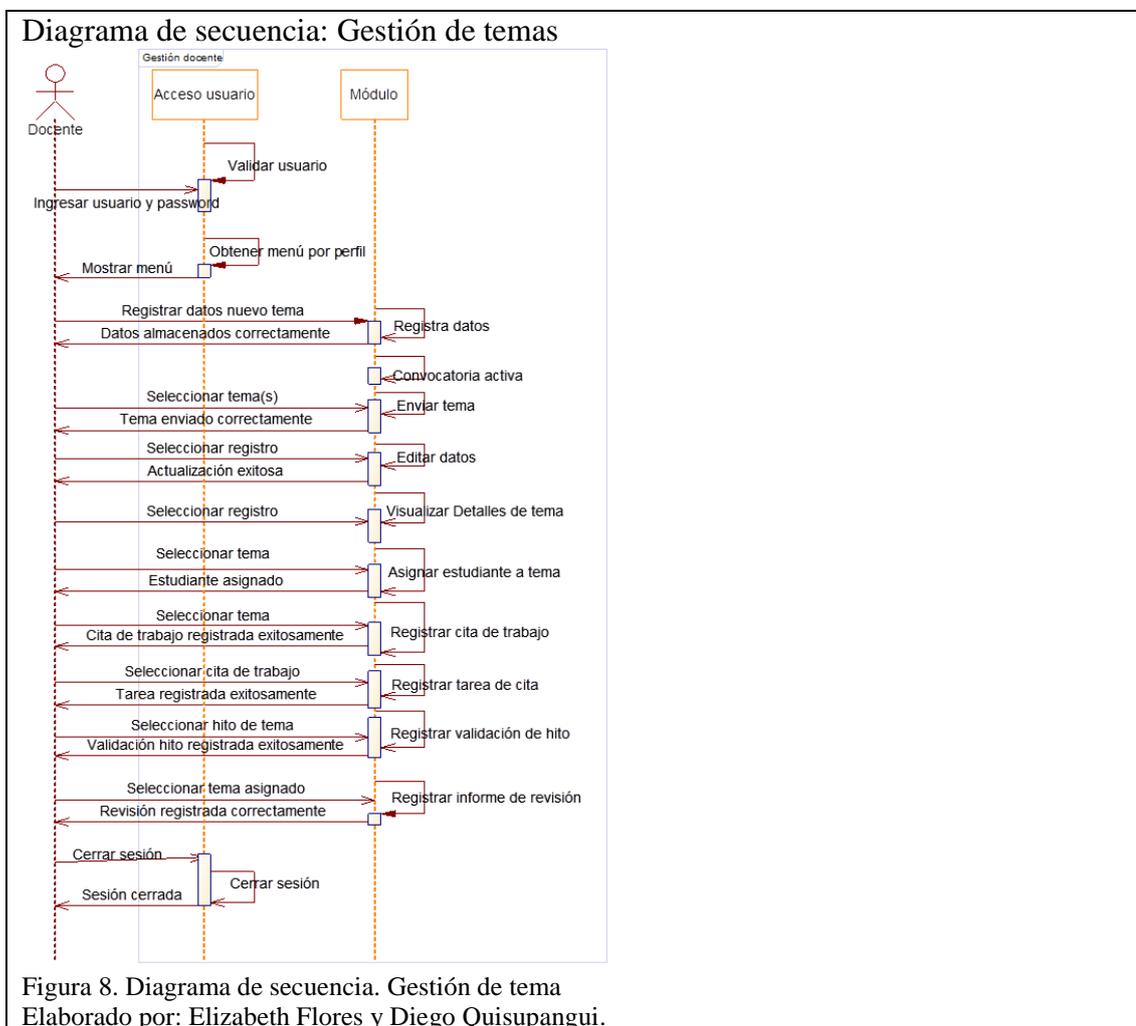
Nota: descripción del caso de uso seguimiento de proyecto.
Elaborado por: Elizabeth Flores y Diego Quisupangui.

2.4 Diagramas de Secuencia

Los diagramas de secuencia ayudaron a conocer el comportamiento del sistema en sus diferentes módulos, en cada Sprint el comportamiento fue evolucionando de acuerdo a los requerimientos que el Product Owner requería hasta la versión final.

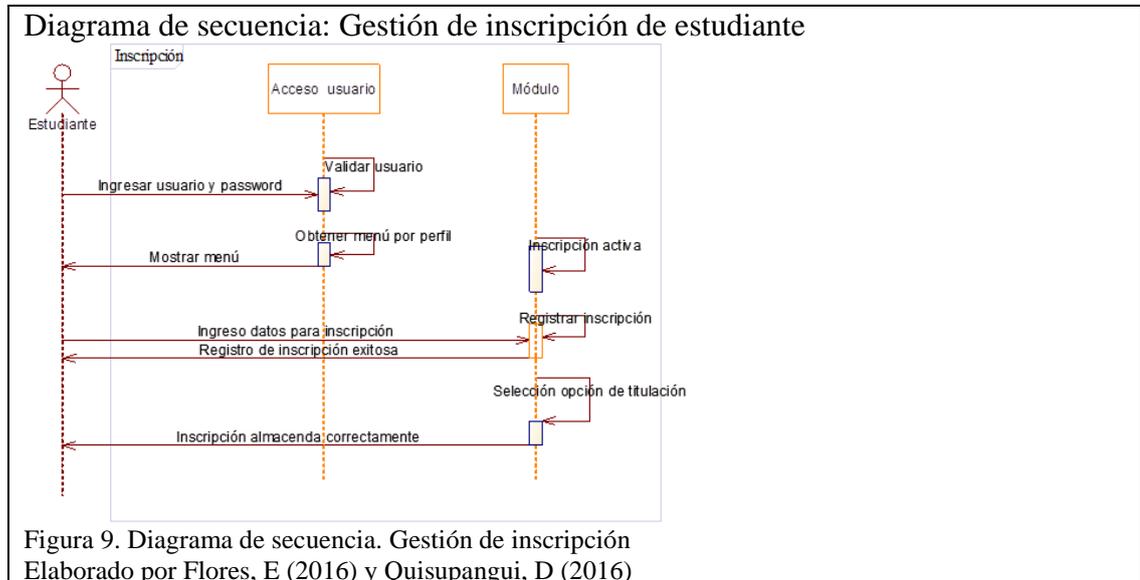
2.4.1 Gestión de docente

La Figura 8 describe el proceso que el actor y el docente realizan con el sistema informático permitiendo entre las principales: el ingreso de un nuevo tema, la asignación de un tema a estudiante, seguimiento del proyecto.



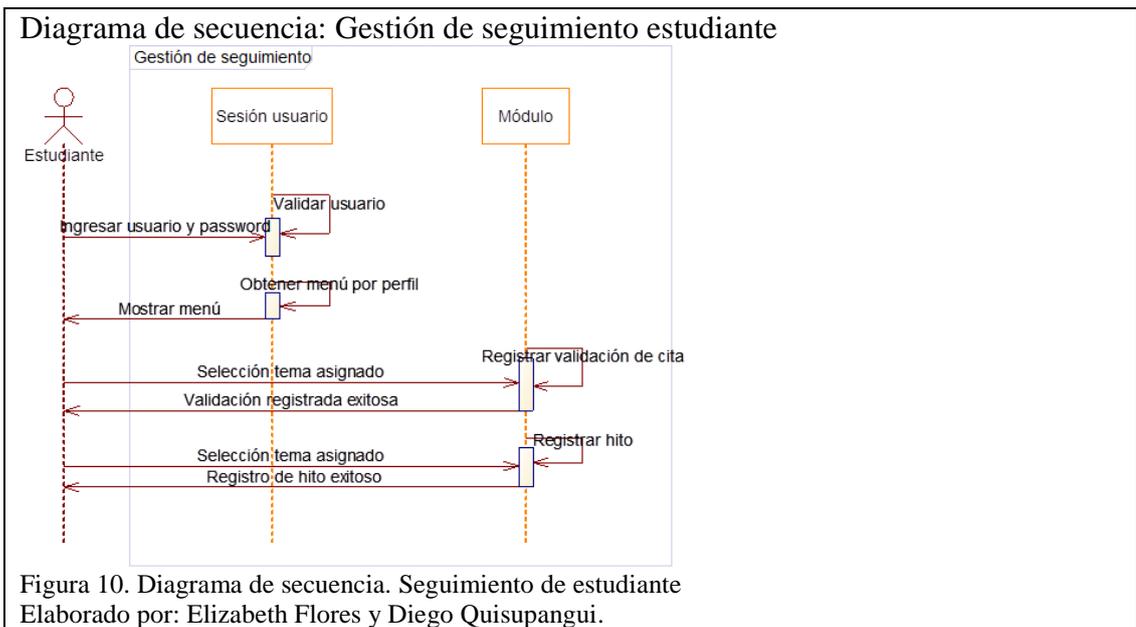
2.4.2 Inscripción de estudiante

El diagrama de secuencia de inscripción del estudiante se detalla el proceso que el estudiante debe realizar para solicitar un tema de proyecto de titulación según la opción seleccionada al momento de su inscripción.



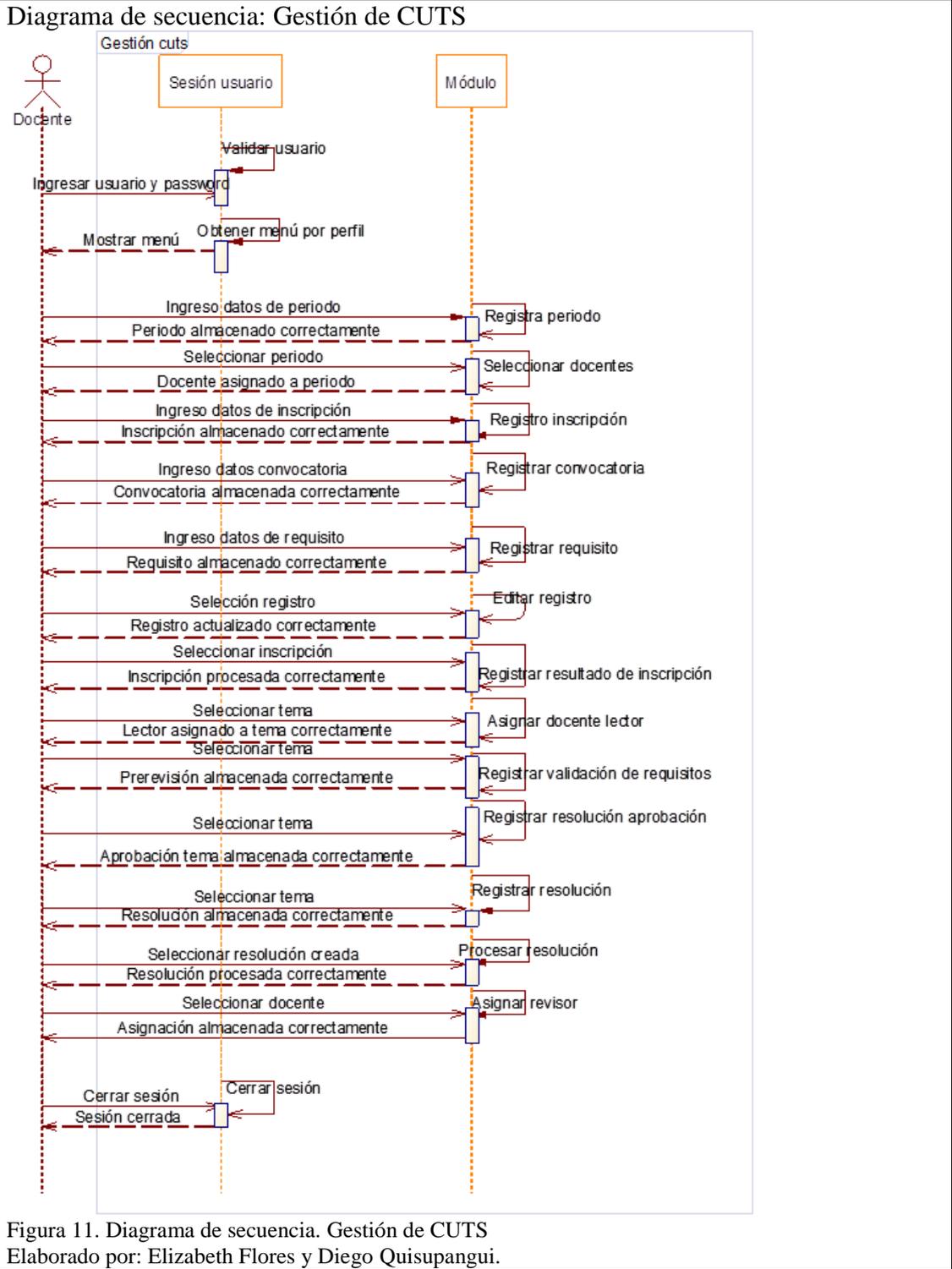
2.4.3 Seguimiento estudiante

En el diagrama se detalla el proceso que realiza el estudiante cuando tiene asignado un tema previamente aprobado como proyecto e inicia el desarrollo del mismo, para esto debe registrar los hitos de acuerdo al cronograma establecido en el plan y validar las citas que el docente tutor establece para el avance en cuanto a la evolución de trabajo.



2.4.4 Gestión de CUTS

El diagrama de gestión de CUTS detalla los procesos que realiza la Coordinación de Unidad de Titulación Especial como son el registro de periodo, docentes, convocatoria, entre otros.



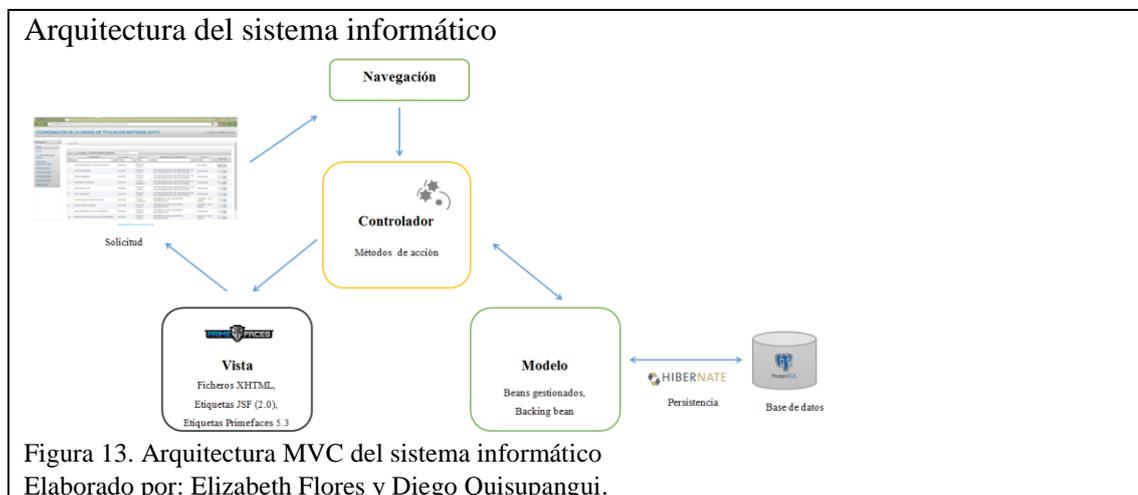
2.5 Modelo lógico de base de datos

En la Figura 12 se puede observar el diagrama lógico de la base de datos que es un modelo independiente del DBMS procedentes del inglés, Data Base Management

Capítulo 3

Construcción y pruebas

Se llevó a cabo la construcción de componentes además de pruebas tal como lo establece la metodología Scrum en cada una de las interacciones. Para la construcción de componentes de software se utilizó JSF 2.0 como framework de desarrollo, que entre otras características tiene la posibilidad de manejar el patrón Modelo Vista Controlador (MVC) el cual provee la posibilidad de reutilización de código, separación de conceptos, organiza el código en función de responsabilidades, la posibilidad de trabajar simultáneamente entre varios miembros de un equipo; entonces en el contexto de este patrón arquitectónico, se usó Primefaces 5.3 en la capa presentación por tener variedad de componentes necesarios para este sistema informático, PostgreSQL como motor de base de datos e *Hibernate* que se define como una herramienta de Mapeo de Objeto Relacional la cual facilita el mapeo de atributos entre una base de datos relacional y el modelo de objetos de una aplicación que agiliza la relación entre la base de datos y la aplicación, en la capa de persistencia de datos; en la Figura 8, se puede observar el diagrama que contiene la arquitectura de acuerdo al patrón MVC conjuntamente con las herramientas utilizadas.



El sistema informático se puede organizar en su arquitectura de la siguiente manera:

Presentación: Muestra la interfaz para el usuario, en esta capa tenemos formularios que son desarrollados en archivos con extensión *xhtml*, con etiquetas propias de JSF y *Primefaces*.

Lógica de negocio: son los componentes desarrollados en lenguaje Java.

Acceso a base de datos: Provee el acceso a la información existente almacenada, en esta capa se tiene *Hibernate* que realiza la persistencia a la base de datos.

3.1 Estándares de programación

Los estándares de programación que se usaron en la creación de componentes comprenden aspectos de la generación de código, ya que al tener un estándar establecido puede facilitar la creación de nuevos componentes y al mantenimiento, al aplicar estos estándares se puede obtener un código legible que facilita la revisión y la documentación.

Para un mejor entendimiento dentro del código se usaron prefijos, algunos de estos se encuentra en la Tabla 14. Prefijos

Tabla 14. Prefijos

Tipo de control	Prefijo	Ejemplo
Botón	btn	btnAbrir
Calendar	Cal	Calcita
Tabla	Tb	tbTema
Botón comando	cmb	cmbGuardar
Cuadro de texto	txt	txtApellido

Nota: Prefijos usados en dentro del código.

Elaborado por: Elizabeth Flores y Diego Quisupangui.

3.1.1 Notación

Para el desarrollo del sistema informático se ha utilizado la notación CamelCase la cual hace uso de mayúsculas y minúsculas para la definición de variables, clases, métodos, etc. utilizados en la codificación

En la Tabla 15 se pueden observar algunos ejemplos de cómo se usó en la codificación del sistema.

Tabla 15. Notación CamelCase

No.	Tipo de identificador	Tipo	Ejemplo
1	Clases	Upper CamelCase	ListaBean
2	Métodos	lower CamelCase	updateTemaRevisorEstado
3	Variables	lower CamelCase	listDocente
4	Constantes	Upper	NOMBRE_RECURSO

Nota: Ejemplos de notación CamelCase del sistema informático.

Elaborado por: Elizabeth Flores y Diego Quisupangui.

3.2 Construcción de la aplicación web

En el desarrollo de los componentes de software que son parte de la aplicación web se realizaron iteraciones en las cuales tanto el código, esquema de base de datos y demás componentes del software, tenían un cambio constante adaptando la lógica para cumplir con los requerimientos del Product owner; a continuación se describe cada uno de los procesos realizados:

3.2.1 Diagrama físico

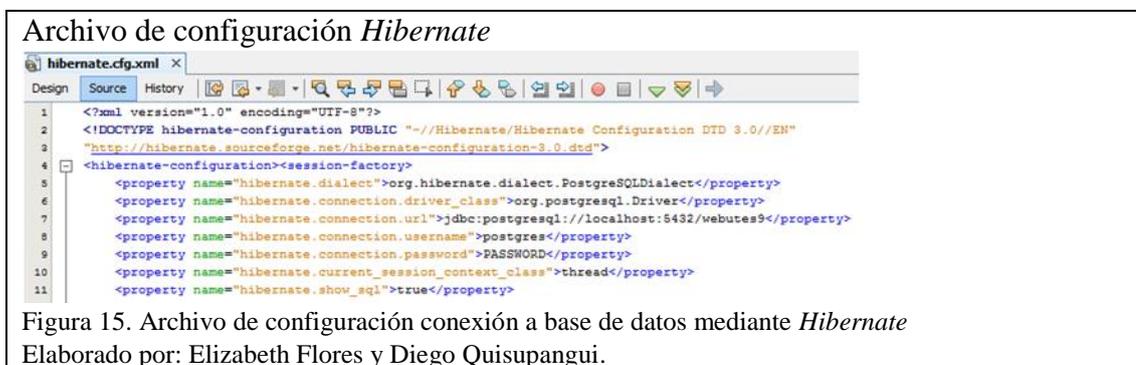
Este diagrama se generó a partir del diagrama lógico presentado en el capítulo anterior de este documento, este se generó para el gestor de base de datos PostgreSQL que contienen los tipos de datos soportados por esta herramienta.

3.2.2 Hibernate

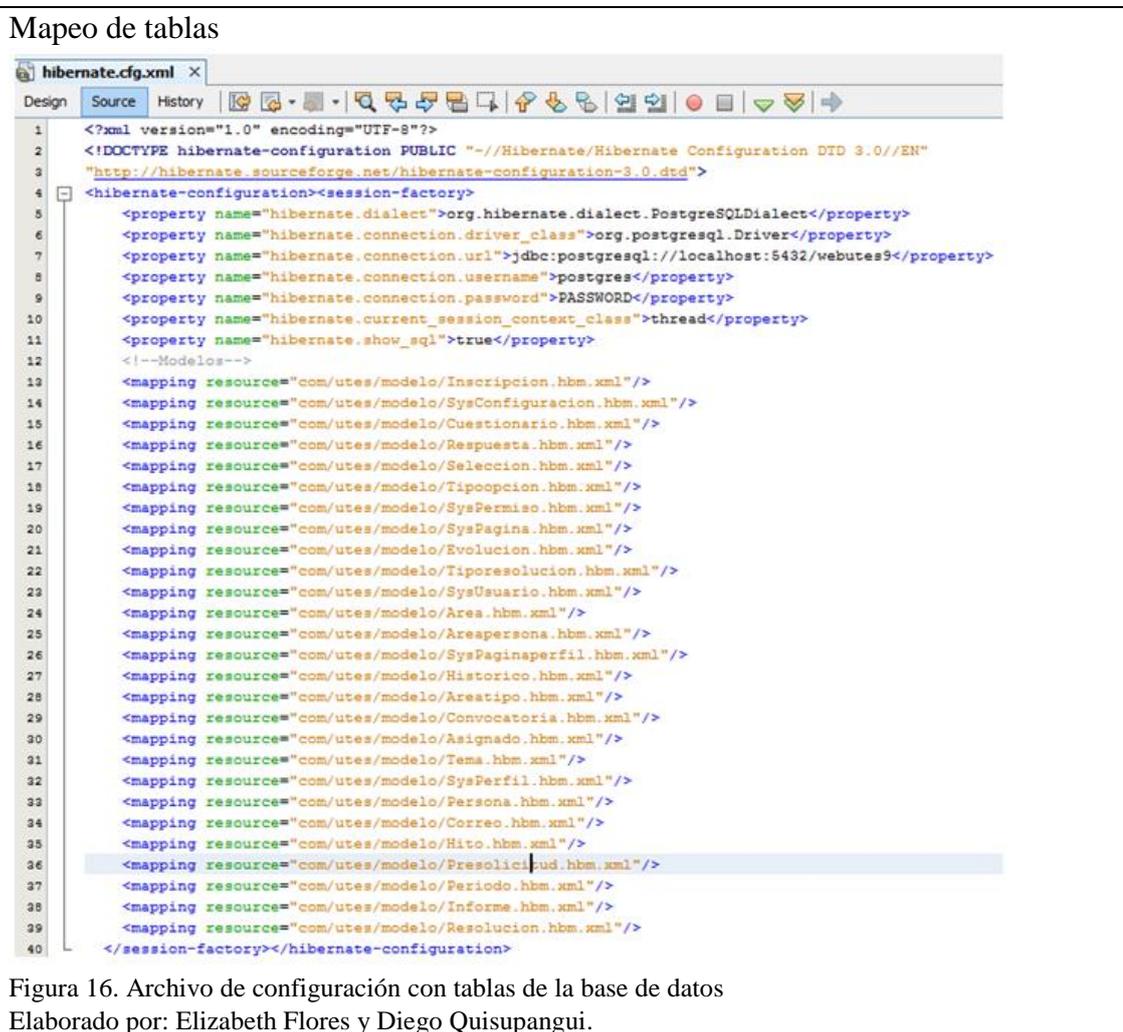
Posterior a la creación de la base de datos en Postgres se utilizó Hibernate para el manejo de persistencia, a través de esta herramienta se realizó la conexión y la esquematización de las tablas en la base de datos a clases Java mediante la opción de ingeniería inversa por lo que se generó dos tipos de archivos: 1. POJO (Plain Old Java Object) son las clases Java que representan objetos que tienen correspondencia con las tablas de la base de datos relacional mediante la definición de variables privadas y sus métodos de acceso *getters* y *setters*, al momento de generar los archivos POJO mediante el uso de la herramienta Hibernate se puede añadir anotaciones compatibles con JPA (Java Persistence API), estas son conocidas también como anotaciones EJB 3.0, las cuales proporcionan un modelo de persistencia de objetos, con estas anotaciones añadidas a los archivos POJO se define la clase entidad de JPA; 2. Ficheros de mapeo (*.hbm.xml*), que indican el mapeo entre los atributos de una clase y los campos de la tabla mediante etiquetas propias de Hibernate.

A continuación se detallan los archivos de configuración de Hibernate y los archivos generados por la ingeniería inversa.

En la Figura 15 se puede observar el archivo que contiene los parámetros de configuración para la conexión hacia la base de datos



En la Figura 16 se puede observar el mapeo de los archivos *.hbm.xml* que hacen referencia a las tablas de la base de datos.



Se generaron los archivos POJO (Plain Old Java Object) que sirvieron para tener una conexión directa con los campos de cada tabla mediante sus *getters* y *setters*.

En la Figura 17 se muestra la entidad Tema creada a partir del archivo POJO, donde se puede visualizar las anotaciones JPA, variables con los tipos de dato específicos que corresponden a los campos de la tabla tema.

Entidad tema

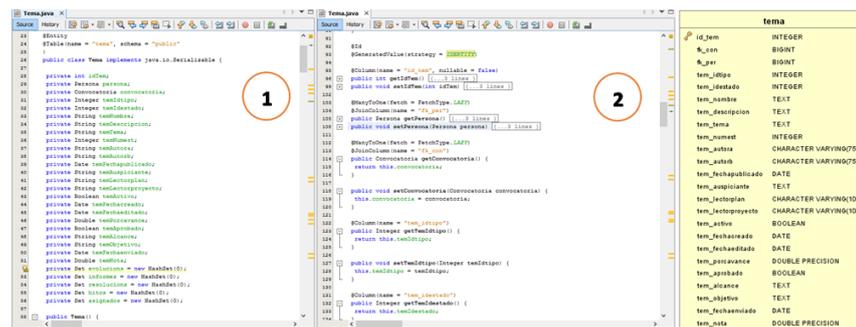


Figura 17. Parámetros mapeados desde tabla tema.
Elaborado por: Elizabeth Flores y Diego Quisupangui.

En la Figura 18 se puede observar las siguientes anotaciones JPA: `@Id` es el campo donde se define la clave primaria, `IDENTITY` contiene la estrategia para la generación de identificadores; `@ManyToOne` es la relación de muchos a uno que conjuntamente con el campo `@JoinColumn` especifica la columna a la que hace referencia la relación, también puede ser del tipo `@OneToMany` o también `@ManyToMany` dependiendo de la relación existente en la base de datos; `@Column` es la anotación que hace referencia al nombre del campo de la tabla en la base de datos, se puede añadir más características para definir al campo.

Anotaciones JPA

```

→ @Id
@GeneratedValue(strategy = IDENTITY)

→ @Column(name = "id_tem", unique = true, nullable = false)
public int getIdTem() {...3 lines}
public void setIdTem(int idTem) {...3 lines}

→ @ManyToOne(fetch = FetchType.LAZY)
@JoinColumn(name = "fk_per")
public Persona getPersona() {...3 lines}
public void setPersona(Persona persona) {...3 lines}

→ @ManyToOne(fetch = FetchType.LAZY)
@JoinColumn(name = "fk_con")
public Convocatoria getConvocatoria() {...3 lines}
public void setConvocatoria(Convocatoria convocatoria) {...3 lines}

→ @Column(name = "tem_idtipo")
public Integer getTemIdtipo() {...3 lines}
public void setTemIdtipo(Integer temIdtipo) {...3 lines}

@Column(name = "tem_idestado")
public Integer getTemIdestado() {...3 lines}
public void setTemIdestado(Integer temIdestado) {...3 lines}

@Column(name = "tem_nombre")
public String getTemNombre() {
    return this.temNombre;
}

public void setTemNombre(String temNombre) {
    this.temNombre = temNombre;
}
    
```

Figura 18. Etiquetas JPA de la entidad tema.
Elaborado por: Elizabeth Flores y Diego Quisupangui.

En la Figura 19 se puede observar el archivo *xml* que sirve para la definición de parámetros de la tabla tema, las relaciones con otras entidades, los campos y tipos de datos.

Archivo mapeo *Hibernate* a tabla tema

```
<hibernate-mapping>
  <class name="com.utes.modelo.Tema" table="tema" schema="public">
    <id name="idTem" type="int">
      <column name="id_tem" />
      <generator class="identity"/>
    </id>
    <many-to-one name="persona" class="com.utes.modelo.Persona" fetch="select">
      <column name="fk_per" />
    </many-to-one>
    <many-to-one name="convocatoria" class="com.utes.modelo.Convocatoria" fetch="select">
      <column name="fk_con" />
    </many-to-one>
    <property name="temIdtipo" type="java.lang.Integer">
      <column name="tem_idtipo" />
    </property>
    <property name="temIdestado" type="java.lang.Integer">
      <column name="tem_idestado" />
    </property>
    <property name="temNombre" type="string">
      <column name="tem_nombre" />
    </property>
    <property name="temDescripcion" type="string">
      <column name="tem_descripcion" />
    </property>
    <property name="temTema" type="string">
      <column name="tem_tema" />
    </property>
    <property name="temNumest" type="java.lang.Integer">
      <column name="tem_numest" />
    </property>
    <property name="temAutora" type="string">
      <column name="tem_autora" length="75" />
    </property>
  </class>
</hibernate-mapping>
```

Figura 19. Mapeo *Hibernate* de la tabla tema.
Elaborado por: Elizabeth Flores y Diego Quisupangui.

En la Figura 20 se puede observar el archivo para manejo de sesiones que hace referencia al archivo de configuración anteriormente mencionado en la Figura 15 de este documento, esta clase sesión es la que se utilizó para el acceso hacia las entidades.

Archivo para manejo de sesión

```
import org.hibernate.cfg.AnnotationConfiguration;
import org.hibernate.SessionFactory;

/** Hibernate Utility class with a convenient method to get Session Factory ...5 lines */
public class HibernateUtil {

    private static final SessionFactory sessionFactory;

    static {
        try {
            // Create the SessionFactory from standard (hibernate.cfg.xml)
            // config file.
            sessionFactory = new AnnotationConfiguration().configure().buildSessionFactory();
        } catch (Throwable ex) {
            // Log the exception.
            System.err.println("Initial SessionFactory creation failed." + ex);
            throw new ExceptionInInitializerError(ex);
        }
    }

    public static SessionFactory getSessionFactory() {
        return sessionFactory;
    }
}
```

Figura 20. Archivo generado por *Hibernate* para manejo de sesiones
Elaborado por: Elizabeth Flores y Diego Quisupangui.

Luego de la creación de entidades JPA, se realizó la conexión con la capa lógica, esto se realizó con el patrón DAO (Data Access Object) que suministra una interfaz común entre la capa lógica y la capa de base de datos.

En la Figura 21 Se puede observar la interfaz *Dao* de tema con los métodos que tienen que ser implementados, esta interfaz ayudo en la organización del código.

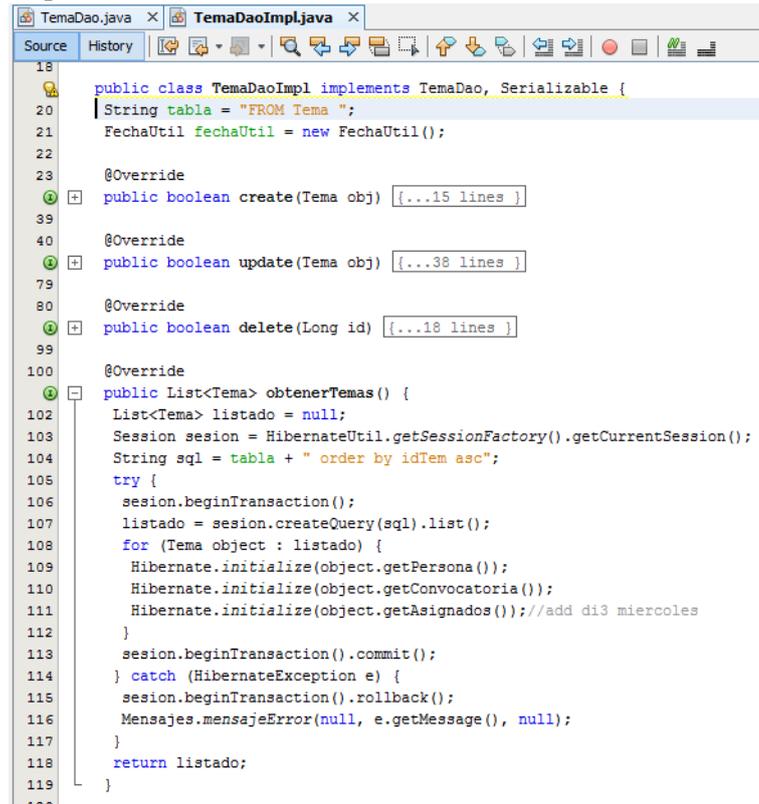
Ejemplo de interfaz *DAO* de entidad tema

```
1  ...5 lines
6  package com.utes.dao;
7
8  import ...3 lines
11
12  /**...4 lines */
17  public interface TemaDao {
18  /** <b> Creación de registro ...9 lines */
19  public boolean create(Tema obj);
28
29  /** <b> Actualización del registro ...9 lines */
30  public boolean update(Tema obj);
39
40  /** <b> Eliminar registro ...9 lines */
41  public boolean delete(Long id);
50
51  /** <b> Obtener todos los temas ...8 lines */
52  public List<Tema> obtenerTemas();
60
61  /** <b> Obtener todos los temas ...9 lines */
62  public List<Tema> obtenerTemasExceptoEstado(String codigos);
71
72  /** <b> Obtener todos los temas por ids estados </b> ...9 lines */
73  public List<Tema> obtenerTemasEstados(String codigos);
74
```

Figura 21. Interfaz Java con patrón Dao para tema
Elaborado por: Elizabeth Flores y Diego Quisupangui.

En la Figura 22 se puede observar la clase implementación de la interfaz *Dao* de tema en la cual contiene la codificación necesaria para crear, editar, consultar datos dependiendo las necesidades.

Implementación de interfaz DAO



```
18 public class TemaDaoImpl implements TemaDao, Serializable {
19     String tabla = "FROM Tema ";
20     FechaUtil fechaUtil = new FechaUtil();
21
22
23     @Override
24     public boolean create(Tema obj) {...15 lines }
25
26
27     @Override
28     public boolean update(Tema obj) {...38 lines }
29
30
31     @Override
32     public boolean delete(Long id) {...18 lines }
33
34
35     @Override
36     public List<Tema> obtenerTemas() {
37         List<Tema> listado = null;
38         Session session = HibernateUtil.getSessionFactory().getCurrentSession();
39         String sql = tabla + " order by idTem asc";
40         try {
41             session.beginTransaction();
42             listado = session.createQuery(sql).list();
43             for (Tema object : listado) {
44                 Hibernate.initialize(object.getPersona());
45                 Hibernate.initialize(object.getConvocatoria());
46                 Hibernate.initialize(object.getAsignados()); //add di3 miercoles
47             }
48             session.beginTransaction().commit();
49         } catch (HibernateException e) {
50             session.beginTransaction().rollback();
51             Mensajes.mensajeError(null, e.getMessage(), null);
52         }
53         return listado;
54     }
55 }
```

Figura 22. Implementación de interfaz Dao de tema
Elaborado por: Elizabeth Flores y Diego Quisupangui.

Al tener archivos *Dao* que interactúan con la base de datos se crearon los *Bean* que son componentes de software que entre sus características permiten la reutilización de código, encapsulan información, estos componentes son similares los POJO ya que se tienen métodos *getters* y *setters*, también brindan la posibilidad de añadir anotaciones para que pueda ser gestionado por JSF por lo que pueden ser accedidos desde una página JSF (*xhtml*), lo cual ayudó en la comunicación directa con la capa presentación.

En la Figura 23 se puede observar el archivo *Bean* de tema que interactúa directamente con la capa presentación, se puede apreciar la creación de la variable *daotema* que hace referencia a la interfaz *TemaDao*, está se implementa a través de la llamada a la clase *TemaDaoImpl()* la cual contiene los métodos que interactúan con la base de datos; además contiene el método *updateTemaInicio* que sirvió para la

actualización de datos al momento de editar un tema en la capa presentación por medio de los componentes JSF.

Ejemplo de conexión a interfaces *Dao* desde capa lógica

```

@ManagedBean
@Named(value = "temasBean")
@SessionScoped
public class TemasBean implements Serializable, Estaticos {

    TemaDao daotema;

    public TemasBean() {
        daotema = new TemaDaoImpl();
    }

    public void updateTemaInicio(ActionEvent actionEvent) {
        RequestContext context = RequestContext.getCurrentInstance();
        boolean validacion = false;
        try {
            if (daotema.obtenerTemaxNombre(this.enTemaDetalleSeleccion.getTemNombre()) == null) {
                if (daotema.update(this.enTemaDetalleSeleccion)) {
                    //notificacionGlobal("Notificación tema", "Se actualizo un tema", "/actualizatema");
                    reloadLists(1);
                    Mensajes.mensajeInfo(null, MENSAJE_OK_ACTUALIZA, null);
                    enTemaDetalleSeleccion = null;
                    validacion = true;
                } else {
                    Mensajes.mensajeError(null, MENSAJE_ERROR_ACTUALIZA, null);
                } else {
                    Mensajes.mensajeError(null, MENSAJE_ERROR_EXISTE, null);
                }
            } catch (Exception e) {
                throw e;
            }
            context.addCallbackParam("docenteeditatema", validacion);
        }
    }
}

```

Figura 23. Ejemplo de uso de interfaz en método de la capa lógica.
Elaborado por: Elizabeth Flores y Diego Quisupangui.

3.2.3 Diagrama de clases

Este diagrama de clases se desarrolló a partir de la generación de modelos a través de la ingeniería reversa realizada con la herramienta *Hibernate*.

En la Figura 24 se puede observar el esquema de paquetes que contienen el diagrama de clases del sistema informático realizado. (Gracia Burgués, 2016)

Diagrama de clases global

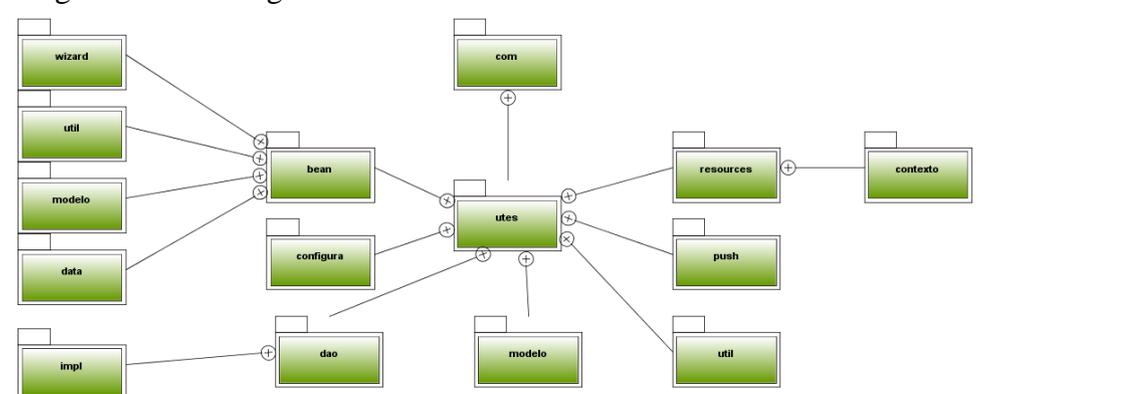
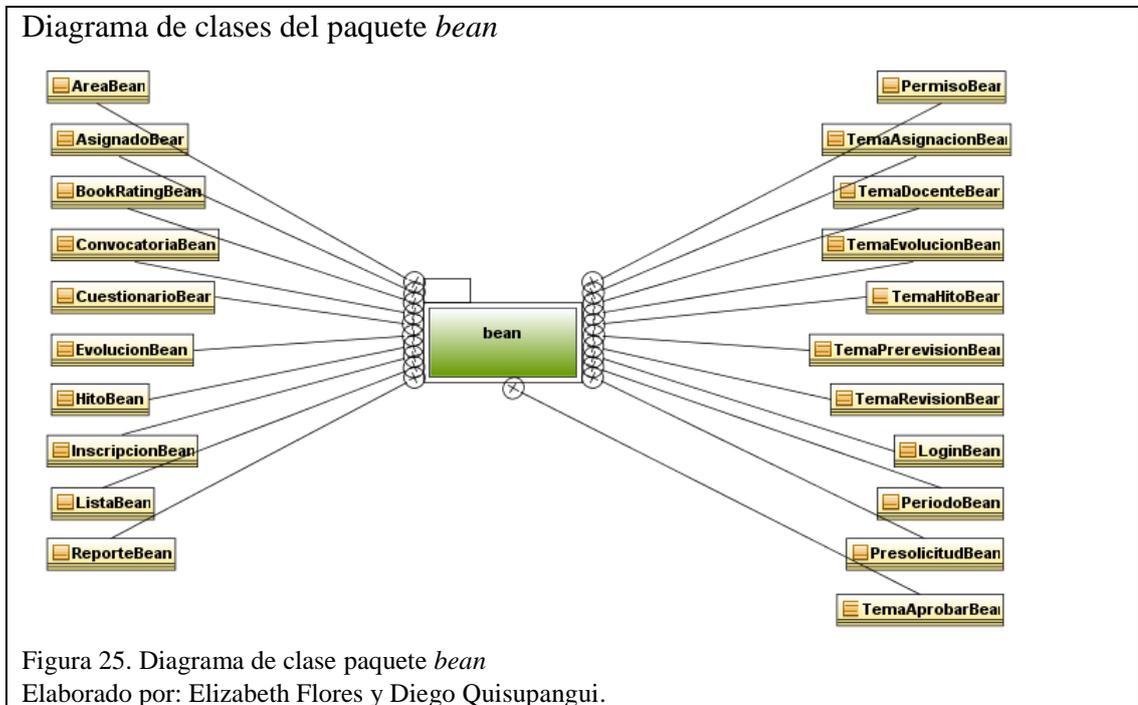


Figura 24. Estructura de diagrama de clases organizada por paquetes
Elaborado por: Elizabeth Flores y Diego Quisupangui.

En la Figura 25 se puede observar el diagrama de clases que hace referencia al paquete *bean*, este es parte del paquete principal *com.utes* como se puede apreciar en la Figura 24; el paquete *bean* contiene todas las clases *Bean* que fueron utilizadas en el desarrollo de los componentes JSF en la capa presentación.



En la Figura 26 se puede observar el diagrama de clases del paquete *com.utes.modelo* que contiene las entidades creadas a partir de la base de datos que fueron generadas por medio de *Hibernate*, se puede apreciar que cada una de las entidades tiene varias relaciones hacia otras, esto es similar a las relaciones entre tablas en la base de datos.

Diagrama de clases para el paquete modelo

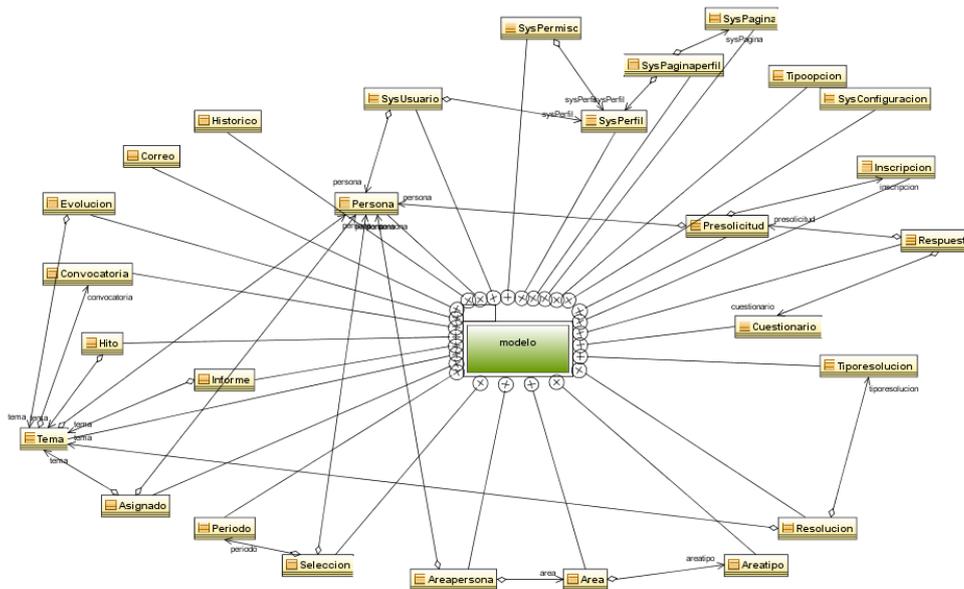


Figura 26. Diagrama de clase del paquete modelo que contiene entidades
Elaborado por: Elizabeth Flores y Diego Quisupangui.

En la Figura 27 se puede observar cómo está conformada la entidad Tema con sus respectivos atributos que fueron generados por la herramienta de Hibernate.

Diagrama de clases ampliado la clase entidad tema

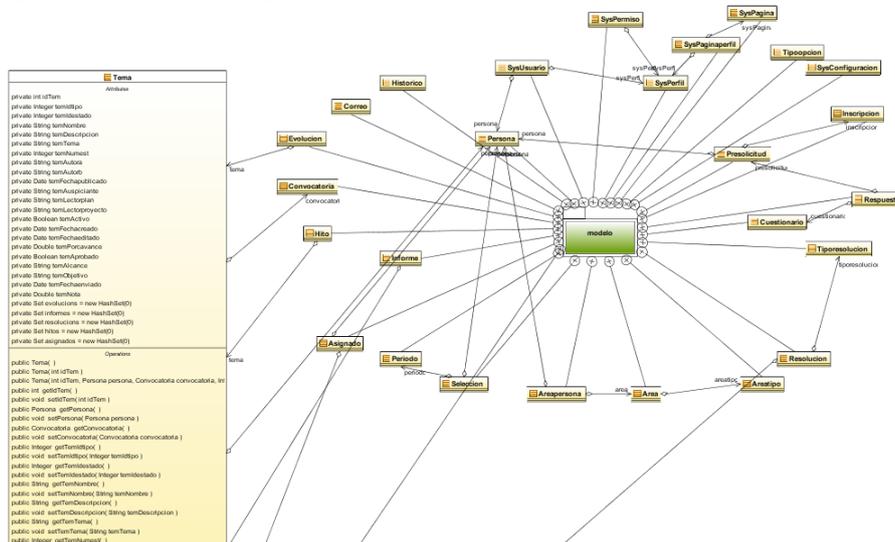


Figura 27. Diagrama de clase ampliación de entidad tema
Elaborado por: Elizabeth Flores y Diego Quisupangui.

Para mayor detalle de los diagramas de clases se lo puede revisar el Anexo 2. Diagramas de clases.

3.2.4 Código relevante

En el desarrollo del sistema informático a través de cada iteración que se realizaba se encontraron funcionalidades importantes que debían ser resueltas, para ello se realizó el respectivo análisis para entender la lógica que se necesitaba codificar; a continuación se detalla el código más relevante realizado en cada uno de los procesos y subprocesos del negocio, refiriéndose a negocio como la organización que realiza actividades en este caso la CUTS, teniendo en cuenta que el código descrito está en su última versión.

3.2.4.1 Código en proceso Prerrequisitos

En el desarrollo de los componentes en algunos requerimientos se hacía referencia a que era necesario obtener información del último registro habilitado de convocatoria, inscripción y periodo, debido a que al realizar un nuevo registro o acción que tengan relación a lo antes mencionado no sea posible seleccionar en los respectivos formularios y que se añada automáticamente al nuevo registro.

En los siguientes ítems se puede apreciar porciones de código que ayudaron a resolver la funcionalidad requerida.

- Consulta de último registro de convocatoria con sentencia SQL, que retorna el valor del secuencial.

Código de consulta para la obtener convocatoria

```
@Override
public String obtenerUltimoRegistroConvocatoria() {
    /*
     * CONSULTA SQL EN POSTGRES
     */
    String valor = "0";
    Session sesion = HibernateUtil.getSessionFactory().getCurrentSession();
    Query query;
    List result;
    try {
        sesion.beginTransaction();
        query = sesion.createQuery("SELECT last_value FROM convocatoria_id_con_seq");
        result = query.list();
        if (result.get(0) != null) {
            valor = "" + result.get(0);
        } else {
            valor = "-1";
        }
        sesion.beginTransaction().commit();
    } catch (HibernateException e) {
        sesion.beginTransaction().rollback();
        Mensajes.mensajeError(null, e.getMessage(), null);
        throw e;
    }
    return valor;
}
```

Figura 28. Consulta de id para la última convocatoria habilitada

Elaborado por: Elizabeth Flores y Diego Quisupangui.

b) Con el id de secuencia obtenido de la consulta a la base de datos para obtener la última convocatoria se realiza la consulta para obtener una entidad convocatoria que tendrá todos los campos de la tabla y que serán usados cuando sea necesario.

Código consulta de entidad convocatoria por id de convocatoria

```
//Obtiene el ID de la SECUENCIA de la tabla convocatoria
String idlastconv = daoconvocatoria.obtenerUltimoRegistroConvocatoria();
//Obtiene la entidad convocatoria si se tiene un ID y si esta activa esa convocatoria ultima almacenada
enconvocatoriaseleccion = daoconvocatoria.obtenerConvocatoriaPorIdActivo(idlastconv, true);
```

Figura 29. Consulta para obtener entidad convocatoria por su id.

Elaborado por: Elizabeth Flores y Diego Quisupangui.

3.2.4.2 Código en proceso Publicación de tema

En el proceso de Publicación de temas donde intervienen varios actores se identificó que el tema pasa por diferentes estados que definen el comportamiento, por lo que fue necesario actualizar este campo constantemente, esto se resolvió con la tabla estado y tipo de estado que contienen los diferentes estados que fueron requeridos no solamente para este proceso sino que también para los siguientes procesos detallados más adelante en este documento.

Código de actualización de estado

```
public void updateTemaRevisorEstado(ActionEvent actionEvent) {
    Map<String, String> params;
    String idtem = "";
    try {
        params = FacesContext.getCurrentInstance().getExternalContext().getRequestParameterMap();
        idtem = params.get("idtem");
        if (idtem.isEmpty() != true) {
            Tema objtema = daotema.obtenerTemaxId(Integer.parseInt("" + idtem));
            if (objtema != null) {
                if (objtema.getTemIdestado() == ESTADO_TEMA_PRE_ASIGNAREVISOR) {
                    objtema.setTemIdestado(Integer.parseInt("" + ESTADO_TEMA_PRE_REVISADO));
                    if (daotema.update(objtema)) {
                        Asignado auxenasignado = daoasigna.obtenerEntidadAsignadoxIdPersona(
                            enpersonasesion.getIdPer(), TIPO_ID_ASIGNACION_REVISOR + "", objtema.getIdTem() + ""
                        );
                        auxenasignado.setAsgIdestadotema(ESTADO_ID_TEMA_REVISA_TERMINADO);
                        if (daoasigna.update(auxenasignado)) {
                            reloadLists(1);
                            Mensajes.mensajeInfo(null, MENSAJE_OK_ACTUALIZA, null);
                            reloadLists(3);
                            reloadLists(1);
                        } else {
                            Mensajes.mensajeError(null, MENSAJE_ERROR_ACTUALIZA, null);
                        } else {
                            Mensajes.mensajeError(null, MENSAJE_ERROR_ACTUALIZA, null);
                        } else {
                            Mensajes.mensajeError(null, MENSAJE_ERROR_ACTUALIZA, null);
                        }
                    }
                }
            }
        } catch (NumberFormatException e) {
            throw e;
        }
    }
}
```

Figura 30. Actualización de estado de tema

Elaborado por: Elizabeth Flores y Diego Quisupangui.

El método *updateTemaRevisionEstado* contiene la estructura que se utilizó para los siguientes procesos: Proceso Inscripción y Aprobación de proyecto, debido a que se tenían varios requerimientos con la misma lógica en donde se manejaban estados que cambiaban constantemente, algunos de los requerimientos que hicieron uso de la estructura del método antes mencionado son: revisar inscripción, aprobación de tema como proyecto, asignación de revisor a tema, asignación de tema a estudiante.

Para el uso de notificaciones a través del sistema informático se hizo uso de uno de los componentes que provee *Primefaces* como es la notificación *Push* que se detalla en la Figura 31.

En la Figura 31 se puede observar los siguientes ítems: 1. es el código para la llamada a método *notificacionGlobal* que contiene los parámetros necesarios para la notificación como son: resumen es el texto que sirve como título de la notificación, detalle es el texto con la acción que se realizó, canal es donde se coloca el nombre del canal a donde se envía la información; 2. se crea un bus de conexión hacia la

clase que contiene la creación del canal; 3. la clase *Nt_ntema* que contiene anotaciones con el nombre del canal que son propias de Primefaces, para el método *onMessage* que sirvió para el retorno de los datos enviados y que posteriormente serán visualizados en la página *xhtml*; 4. porción de código que permite mostrar la notificación leída en página *xhtml*, el retorno de información se lo debe mostrar en el componente de Primefaces *growl*.

Notificaciones de sistema informático

```

1  notificacionGlobal("Notificaciones", "Se registro un nuevo tema", "/temaenviado");

2  public void notificacionGlobal(String resumen, String detalle, String canal) {
    EventBus eventBus = EventBusFactory.getDefault().eventBus();
    eventBus.publish(canal, new FacesMessage(StringEscapeUtils.escapeHtml3(resumen),
        StringEscapeUtils.escapeHtml3(detalle)));
    }

    import javax.faces.application.FacesMessage;
    import org.primefaces.push.annotation.OnMessage;
    import org.primefaces.push.annotation.PushEndpoint;
    import org.primefaces.push.impl.JSONEncoder;

3  @PushEndpoint("/nuevotema")
    public class Nt_ntema {

        @OnMessage(encoders = {JSONEncoder.class})
        public FacesMessage onMessage(FacesMessage message) {
            return message;
        }
    }

4  <p:growl id="growlNotifica" widgetVar="wNotifica" showDetail="true" life="5000"/>
    <p:socket onMessage="handleMessage" channel="/temaenviado" >
    <p:ajax event="message" update=":tabview:formDataRevisor:datatableRevisor" />
    </p:socket>

```

Figura 31. Notificación *push*
 Elaborado por: Elizabeth Flores y Diego Quisupangui.

3.2.4.3 Proceso Seguimiento de proyecto

En este proceso se puede destacar que los actores docente, estudiante y CUTS están involucrados en el transcurso de avance del proyecto por lo que se maneja un registro y validación de citas e hitos, esto se lo realiza de forma asincrónica ya que una vez cambiado el estado de la cita se pueden realizar acciones sobre ésta, también se puede realizar la asignación de lector que también requiere de una manejo por estados y dependiendo del tipo de resolución se pueden efectuar acciones que involucran cambios representativos en el tema.

En la Figura 32 se describe el método denominado *verDialogoExterno* que construye el componente de Primefaces *Dialog* que es un cuadro de diálogo, sirvió para que al momento de desplegar este componente contenga una página con el formulario para realizar la acción que se requiera de acuerdo al tipo de resolución; se puede apreciar inicialmente las propiedades del *Dialog* que entre las más importantes están el alto, ancho, si es *responsive* esto es si cambia la forma de presentación de los componentes al cambiar el tamaño de la página; en el código de selección se puede apreciar el código que abre el *Dialog* con los parámetros de página y propiedades del diálogo.

Método para abrir diálogo externo

```

Map<String, List<String>> parametrosDialog = new HashMap<>();

public void verDialogoExterno(ActionEvent actionEvent, String idtipo, String idres) {
    if (idres.isEmpty() != true && idres.equals("null") != true) {
        this.enResolucionSeleccion = daoresolucion.obtenerResolucionId(Integer.parseInt(idres));
    }
    Map<String, Object> options = new HashMap<>();
    options.put("modal", false);
    options.put("draggable", false);
    options.put("resizable", true);
    options.put("width", 800);
    options.put("height", 450);
    options.put("contentWidth", 800);
    options.put("contentHeight", 450);
    options.put("includeViewParams", true);
    options.put("responsive", true);
    List<String> values = new ArrayList<>();
    values.add("" + enResolucionSeleccion.getIdRel());
    parametrosDialog.put("idresolucion", values);

    if (idtipo.isEmpty() != true && idtipo.equals("null") != true) {
        Integer idtipopres = Integer.parseInt(idtipo);

        switch (idtipopres) {
            case 1:
                //proceso de aprobación de tema a proyecto
                break;
            case 2:
                RequestContext.getCurrentInstance().openDialog("/paginas/privadas/utes/dialog/cambiotutor", options, parametrosDialog);
                break;
            case 3:
                RequestContext.getCurrentInstance().openDialog("/paginas/privadas/utes/dialog/cambiotema", options, parametrosDialog);
                break;
            case 4:

```

Figura 32. Generar diálogo externo
Elaborado por: Elizabeth Flores y Diego Quisupangui.

3.2.5 Diccionario de datos

El diccionario muestra cómo está compuesta la base de datos con los diferentes tipos de datos para el gestor *PostgreSQL 9.0*; Para la estructura del diccionario de datos se describen los siguientes campos: **Nombre tabla:** describe el nombre de la tabla; **Versión:** contiene la versión de la tabla (versión última producto de iteraciones en el desarrollo del sistema informático); **Descripción:** contiene una breve descripción de

la tabla o de un campo; **No:** contiene el número de campo; **Tipo:** contiene el tipo de dato específico para el gestor de base de datos; **Relaciones:** que campo tiene o sirve para relación hacia otra tabla; **Campos clave:** son los campos que son necesarios para un nuevo registro.

En la Tabla 16 se representa la estructura de la tabla con los campos antes descritos y contiene el diccionario de datos de la tabla tema.

Tabla 16. Esquema de diccionario de datos

Nombre tabla:		Tema	Versión:	4.1
Descripción:		Tabla de datos que contiene temas		
No.	Campo	Tipo	Descripción	
1	id_tem	Integer	Identificador de tema	
2	fk_con	Bigint	Identificador de tabla convocatoria	
3	fk_per	Bigint	Identificador de tabla persona	
4	tem_idtipo	integer	Identificador de opción de titulación	
5	tem_idestado	integer	Identificador de estado	
6	tem_nombre	Text	Tema	
7	tem_descripcion	Text	Descripción de tema	
8	tem_tema	Text	Texto de tema	
9	tem_numest	integer	Número de estudiantes para el tema	
10	tem_autora	character varying	Nombres de autor 1	
11	tem_autorb	character varying	Nombres de autor 2	
12	tem_fechapublicado	Date	Fecha de publicación	
13	tem_auspiciante	Text	Nombre de auspiciante	
14	tem_lectorplan	character varying	Nombre de lector de plan	
15	tem_lectorproyecto	character varying	Nombre de lector de proyecto	
16	tem_activo	boolean	Habilita o deshabilita registro	
17	tem_fecha creado	date	Fecha de registro	
18	tem_fechaeditado	date	Fecha de modificación	
19	tem_poravance	double precision	Porcentaje de avance	
20	tem_aprobado	boolean	Valor de aprobado	
21	tem_alcance	Text	Alcance de tema	
22	tem_objetivo	Text	Objetivos de tema	
23	tem_fechaenviado	Date	Fecha de envió a revisión	
Relaciones:			Campos clave:	
fk_con, fk_per			tem_idtipo, tem_idestado, tem_nombre	

Nota: Ejemplo de estructura de diccionario de datos, tabla tema.

Elaborado por: Elizabeth Flores y Diego Quisupangui.

Para ampliación del diccionario de datos, con la información de todas las tablas, estas pueden ser visualizadas en el Anexo 3. Diccionario de base de datos de este documento.

3.3 Pruebas

Las pruebas realizadas en el sistema informático a través del plan de pruebas descrito a continuación sirvieron para identificar los elementos a ser probados y los recursos necesarios realizando una documentación para la entrega final.

3.3.1 Plan de Pruebas

Este plan describe las pruebas realizadas al sistema informático para validar su adecuado funcionamiento antes de su entrega, además de los métodos y herramientas que se usaron para este proceso.

Para ejecutar este plan se han considerado las siguientes referencias: casos de uso, tiempos de ejecución del sistema informático y las funcionalidades del producto final.

Tabla 17. Plan de pruebas

Tipo de Prueba	Descripción de la prueba	Herramientas utilizadas
Pruebas funcionales	<p>Las pruebas funcionales ayudan a conocer las inconsistencias que hay dentro de la aplicación.</p> <p>Las pruebas que se realizaron son de caja negra:</p> <ul style="list-style-type: none"> • Se valida errores de interfaz • Funcionalidad de la aplicación. • Errores de comportamiento. <p>Las técnicas que se ejecutaron fueron :</p> <ul style="list-style-type: none"> • Generar casos de prueba • Validación de los casos de prueba realizando la ejecución de cada 	<p>Al realizar las pruebas de caja negra no se utiliza ninguna herramienta en especial.</p>

	<p>uno de ellos.</p> <ul style="list-style-type: none"> Llevar un registro de los defectos encontrados en el sistema, para luego realizar el cambio. <p>Este tipo de pruebas se realizan a partir de los casos de uso, probando que las funcionalidades asociados a estos no presenten errores.</p>	
Pruebas no funcionales	<p>Las pruebas de rendimiento Se realizan para ver el comportamiento del sistema informático con una carga de usuarios esperada</p> <p>Las pruebas de estrés Se realizan para poner a prueba al sistema informático y observar el punto de ruptura o colapso con una carga de usuarios que excedan a los usuarios esperados.</p> <p>Las pruebas de usabilidad Se realiza para medir la facilidad de uso de la aplicación.</p>	<p>JMETER, para realizar las pruebas de rendimiento.</p> <p>JMETER, para realizar las pruebas de estrés.</p> <p>El método Sirius, ayudará a detectar deficiencias en la interfaz de la aplicación.</p>

Nota: Plan de pruebas.

Elaborado por: Elizabeth Flores y Diego Quisupangui.

3.3.2 Ambiente de pruebas

Respecto a las pruebas funcionales y no funcionales se buscó el cumplimiento de lo descrito en los casos de uso.

Las pruebas se realizaron en una máquina con características mínimas, que sirvió como servidor, en la cual se instaló *Glassfish 4* como servidor de aplicaciones y *Postgres* como gestor de base de datos.

Tabla 18. Características de máquina

Marca	Procesador	Memoria	Sistema operativo	Pantalla
PC Biostar	Intel Pentium Dual	2 GB	Windows 7 64 bits	Monitor 15 pulgadas

Nota: Características de hardware.

Elaborado por: Elizabeth Flores y Diego Quisupangui.

3.3.3 Duración de pruebas

En la Tabla 19 se puede observar la duración de las pruebas realizadas.

Tabla 19. Duración de pruebas

Tipo de pruebas	Tiempo de duración en horas
Funcionales	20
Rendimiento	14
Estrés	16

Nota: Horas planteadas para pruebas.

Elaborado por: Elizabeth Flores y Diego Quisupangui.

3.3.4 Pruebas Funcionales

Las pruebas funcionales se realizaron con pruebas de caja negra las cuales validaron los errores de interfaz y funciones incorrectas, esto se realizó con ayuda de los casos de uso, y así llegar a cumplir los requerimientos funcionales antes mencionados.

Los casos de prueba de registraron en una tabla con los siguientes campos: **Proceso referente al caso de uso:** son los casos de uso a los que se hace referencia; **Actor:** son los actores que intervienen en la aplicación; **Prerrequisito:** es lo que el usuario necesita para ingresar a la aplicación; **Escenario:** es la acción que se realiza dentro la aplicación; **Pasos:** es la secuencia que se ejecuta para cumplir cada escenario; **Resultado esperado:** es el correcto funcionamiento de la aplicación de cada escenario; **Resultado alcanzado:** es lo que se obtuvo del sistema informático luego de la ejecución.

En el Anexo 4. Pruebas se podrán visualizar los resultados de las pruebas de caja negra realizadas.

En las pruebas se obtuvieron algunos errores iniciales de funcionalidades que se pueden apreciar en la columna Resultado alcanzado, estos fueron corregidos tomando como referencia la funcionalidad esperada.

Entre algunos de los errores que se obtuvieron de la ejecución de las pruebas de caja negra, se describen los siguientes:

- Al momento de guardar una nueva inscripción se almacenaba con datos erróneos, esto se dio debido a problemas de configuración del componente utilizado, para la corrección se cambió el tipo de componente de flujo de trabajo.
- Un error recurrente en varias funcionalidades del sistema informático fue el en el botón de acción que al momento de cambiar su estado se debería deshabilitar para que se active en un estado inicial, este error se corrigió mediante un método que habilita y deshabilita el botón dependiendo su estado o estados.

3.3.4.1 Resultado de Pruebas funcionales

Después de realizar las pruebas con ayuda de los casos de uso y con los cambios realizados al encontrar defectos se obtuvieron varias tablas con las respectivas pruebas y defectos corregidos.

La Tabla 20 contiene los siguientes campos: **Proceso**, caso de uso al cual se hace referencia; **Actor**, usuario que realiza la acción; **Resultado**, comportamiento del sistema; **Estado**, situación en la que se encuentra; **Observación**, comentario.

En la Tabla 20 se pueden observar el extracto de una de las matrices que contienen los resultados finales de las pruebas funcionales realizadas al sistema informático, para mayor entendimiento de todas las pruebas en el Anexo 4. Pruebas;**Error! No se encuentra el origen de la referencia..**

Tabla 20. Resultado de pruebas funcionales

Proceso	Actor	Pre Requisito	Escenario	Pasos	Resultado Esperado	Resultado alcanzado
Publicación de Tema	Docente autor	Usuario y contraseña proporcionado por el administrador del sistema	Crear tema	1. Selecciona tipo	Guardar y se visualizar nuevo tema de proyecto o artículo de titulación	OK
				2. Ingresar título		
				3. Ingresar Descripción		
				4. Ingresar número de estudiantes		
	Enviar Tema		1. Seleccionar tema creado	Visualización del cambio de estado del tema y añadir convocatoria actual al tema; notificación vía email	OK	
			2. Enviar temas			
	CUTS		Asignación de revisor de tema	1. Seleccionar tema	Nuevo estado cambia a Asignado	OK
				2. Acciones: escoger revisor de tema		
	Docente revisor		Revisión de Temas	1. Ingresa informe del tema revisado	Visualiza:	OK
				2. Selecciona estado	Op1 Revisado Op2 En revisión	
CUTS	Opción 1. Publicar tema	1. Seleccionar tema	Visualiza en la tabla temas publicados	OK		
		2. Publicar tema				
Docente autor	Opción 2. Publicar tema	1. Selecciona el tema con estado revisión	Visualiza el cambio de estado “Enviado” en los perfiles coordinación y docente autor	OK		

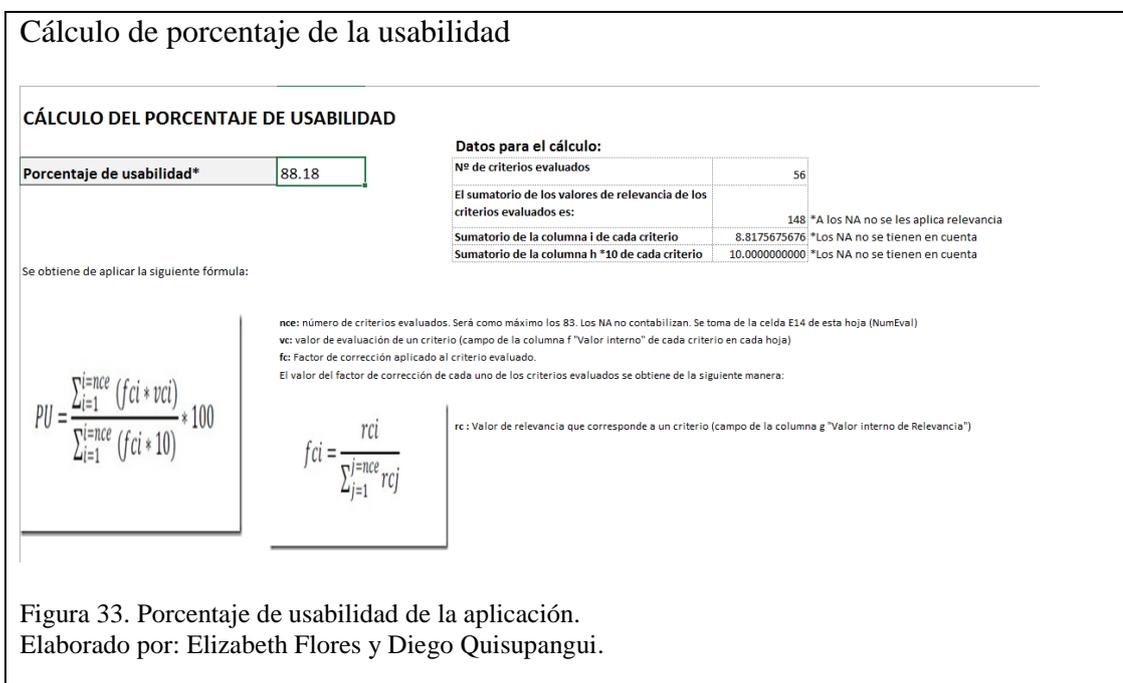
Nota: Resultados de pruebas funcionales finales.
Elaborado por: Elizabeth Flores y Diego Quisupangui.

3.3.5 Pruebas no funcionales

3.3.5.1 Pruebas de usabilidad

Las pruebas de usabilidad ayudarán a validar el uso de la aplicación, para esto se utilizó la herramienta Sirius (Sistema de evaluación de la usabilidad web orientado al usuario) que facilita la detección de las deficiencias de uso dentro de la aplicación según el Modelo de evaluación de Hassan y Fernández.

Con los datos proporcionados por la herramienta de Sirius se determinó la usabilidad de la aplicación que se muestra en la Figura 33 el cual da un porcentaje de 88.18%.



En el Anexo 4. Pruebas se podrá visualizar los pasos que se siguieron para obtener el porcentaje de usabilidad de la aplicación.

3.3.5.2 Pruebas de rendimiento

Las pruebas de rendimiento se ejecutaron con la herramienta JMETER, mediante la simulación de actividad en el sistema informático por los grupos de actores que se identificaron, esto quiere decir que las pruebas fueron realizadas con base en los perfiles Estudiante, Docente y CUTS, para estas pruebas se consideraron los

criterios: Carga inicial: es el número de usuarios que van a utilizar el sistema informático, Carga máxima: es el número de usuarios sobrepasando hasta 4 veces el valor de carga inicial.

Tabla 21. Carga de usuarios

Grupo de funcionalidad	# Carga inicial	#Carga máxima
Estudiante	100 usuarios	200 usuarios
Docente	40 usuarios	120 usuarios
CUTS	10 usuarios	40 usuarios

Nota: Número de usuarios para pruebas de rendimiento.

Elaborado por: Elizabeth Flores y Diego Quisupangui.

Los resultados obtenidos de las pruebas se muestran en la Tabla 22 la cual contiene los siguientes campos: **Media**: tiempo promedio en milisegundos para un conjunto de resultados; **Muestra**: número de peticiones; **Rendimiento**: peticiones por segundo/minuto/hora; **CI**: Carga inicial; **CM**: Carga máxima.

Se puede apreciar que de acuerdo al grupo de funcionalidad a las cuales se realizaron las pruebas con una carga inicial y una carga máxima como se detalla en la Tabla 21 se puede apreciar que los tiempos no excedan más de 10 segundos en cuanto a la Media: es el tiempo promedio, además de que el rendimiento, y finalmente en las peticiones de Estudiante el rendimiento es alto de acuerdo a el número de usuarios en la prueba.

Tabla 22. Resultado de pruebas

	Estudiante		Docente		CUTS	
	CI	CM	CI	CM	CI	CM
# Usuarios	100	200	40	120	10	40
Muestra	1500	3000	1400	4168	345	1395
Media	4128 ms	5138 ms	1831 ms	4567 ms	820 ms	1227 ms
Rendimiento	14,5/s	26,3/s	7,1/s	12,3/s	2.8/s	4.3/s
milisegundo=ms						
segundos=s						

Nota: Resultados de pruebas realizadas.

Elaborado por: Elizabeth Flores y Diego Quisupangui.

3.3.5.3 Pruebas de estrés

Estas pruebas se realizaron para identificar el punto de quiebre del sistema informático por grupo de funcionalidad, en este caso como se muestra en la Tabla 23 la carga de usuarios con la que se busca generar el punto de quiebre, cabe mencionar que esas pruebas se las realizaron por grupo de funcionalidad relacionando al actor del sistema informático.

Tabla 23. Carga de usuarios

Grupo de funcionalidad	# carga inicial	Rendimiento	Estrés
Estudiante	100 usuarios	200 usuarios	600 usuarios
Docente	40 usuarios	120 usuarios	200 usuarios
CUTS	10 usuarios	40 usuarios	120 usuarios

Nota: Carga de usuarios por perfil.

Elaborado por: Elizabeth Flores y Diego Quisupangui.

Los resultados de las pruebas de estrés se definen en la Tabla 24 donde se puede apreciar un considerable incremento el tiempo promedio para las pruebas más altas como es el caso de estudiante con 600 peticiones, en estas pruebas el rendimiento también sufrió un cambio considerable al comparar con las pruebas de rendimiento, finalmente se pudo observar que las pruebas por grupo de funcionalidad en la carga máxima, el sistema informático sigue operando correctamente.

Tabla 24. Resultado de pruebas

	Estudiante		Docente		CUTS	
	CI	CM	CI	CM	CI	CM
Usuarios	100	600	40	200	10	120
Muestra	1500	8994	1400	6944	345	4200
Media	4128 ms	13473 ms	1831 ms	3252 ms	820 ms	2245 ms
Rendimiento	14,5/s	38,9/s	7,1/s	23,3/s	2.8/s	15.3/s
milisegundo=ms						
segundos=s						

Nota: Resultados de pruebas realizados.

Elaborado por: Elizabeth Flores y Diego Quisupangui.

Conclusiones

- La utilización del Modelo y Notación de Procesos de Negocio (BPMN) permitió el entendimiento de los procesos que se ejecutan en el área titulación de la Carrera de Ingeniería de Sistemas, mediante diagramas que sirvieron como base para la creación de los artefactos propios del desarrollo del proyecto.
- Luego de la implementación, el personal de la Coordinación de la Unidad de Titulación Especial de Ingeniería de Sistemas de la Universidad Politécnica Salesiana tendrá la posibilidad de manejar el proceso de titulación de una mejor forma debido a que el sistema informático automatiza buena parte del proceso.
- Para el desarrollo del sistema informático de registro y seguimiento de proyectos de titulación, se usaron herramientas y frameworks con licencia Open Source que están disponibles en Internet, por lo que no se paga licencia, lo que permitió una reducción significativa de costos
- La codificación del sistema informático fue realizada en Java con el framework de desarrollo JSF que al ser orientado al entorno web permite la compatibilidad en los diferentes sistemas operativos.
- Scrum, la metodología ágil utilizada en el desarrollo del software para la unidad de titulación permitió, mediante la generación y utilización de algunos de sus artefactos importantes como: la pila del producto y pila de tareas, se creen incrementos de software funcionales en los que respondieron de forma acertada a los requerimientos del cliente.
- Las diferentes pruebas realizadas al sistema informático para el registro y seguimiento de proyectos de titulación evidenciaron ciertos errores en

codificación, componentes que fueron corregidos de forma inmediata para el óptimo funcionamiento de la aplicación.

- La utilización del software JMeter como herramienta de pruebas ayudó a validar el rendimiento del sistema informático en relación a la carga de usuarios; se ejecutó con una carga máxima de 600 usuarios, en la cual el tiempo promedio fue 13473 milisegundos y con un rendimiento de 38,9/segundo, por lo que se obtuvo una respuesta favorable del sistema informático ante las peticiones realizadas.

Recomendaciones

- Para el correcto funcionamiento del sistema informático, se recomienda que el servidor de aplicaciones mantenga un enlace a Internet con una velocidad de transmisión mínima de 256 kbps debido a que esto es necesario para las notificaciones de correo referente al proceso de titulación sean generadas y enviadas adecuadamente.
- Promover el uso de la aplicación dentro de la Carrera de Ingeniería de Sistemas para demostrar la importancia que ésta tendría en cuanto a la gestión de opciones de titulación y sobre todo por la optimización que se daría al proceso.
- Se recomienda la generación de respaldos automáticos de la base de datos del sistema informático para no tener inconvenientes por pérdida del servicio debido a agentes externos.
- Se recomienda el uso de software con licencia Open Source debido a que cumple con las mismas funcionalidades que los de pago, no se paga por licencia, mantenimiento de software y de personal encargado, lo que permite disminuir gastos, además de la disponibilidad del código fuente que permite personalización.
- Los componentes de software orientados a un tipo de framework son útiles para el desarrollo y presentación de un sistema informático por lo que se recomienda la utilización de los mismos ya que al ser manejados por comunidades, resultan ser robustos y actualizados.

Referencias

(s.f.).

Cardador Cabello, A. (2014). *Desarrollo de aplicaciones web distribuidas : desarrollo de aplicaciones con tecnologías web*. IC Editorial.

Castellanos Rodríguez, D., & Beovides Luis, S. (2013). *Procedimiento de Pruebas No Funcionales para software educativo*. La Habana.

Consejo Superior , U. (2014). *Resolución N° 204-11-2014-12-17*.

Consejo Superior, d. E. (2013). *RPC-SE-13-No.051-2013*.

Deemer, P., Benefield, G., Larman, C., & Vodde, B. (2012). *Scrum Primer*.

Dimes, T. (2015). *Conceptos Básicos De Scrum: Desarrollo De Software Agile Y Manejo De Proyectos Agile*.

Escribano Romero, P. (2011). *Creación y uso de patrones de producto en el marco del proceso de gestión de Proyectos de Software*. Universidad Carlos III de Madrid.

Ferrer Martínez, J. (2014). *Implementación de Aplicaciones Web*. Rama-Editorial.

Gil Pascual, J. A. (2011). *Técnicas e instrumentos para la recogida de información*. Editorial UNED.

Girbal , E., Gardella , C., & Peñalva , M. (2013). *Integrando Casos de uso 2.0 y el Desarrollo Ágil de Software con procesos metacognitivos*. Universidad Tecnológica Nacional, Facultad Regional La Plata.

González Campos, M., & Fernández Martínez, M. (s.f.). *Programación Extrema: Prácticas, Aceptación y Controversia*. Universidad Autónoma de Cd. Juárez.

González Campos, S., & Fernández Martínez, L. (2006). *Programación Extrema: Prácticas, Aceptación y Controversia*. Universidad Autónoma de Cd. Juárez.

Gracia Burgués, E. (2016). *Aprende a Modelar Aplicaciones con UML: 2ª Edición*. IT Campus Academy.

INTECO, L. N. (2009). *Ingeniería del Software: Metodologías y Ciclos de Vida*.

Laínez Fuentes, J. (2015). *Desarrollo de Software ÁGIL: Extreme Programming y Scrum*.

Letelier Torres, P., & Sánchez López, E. A. (2003). *Metodologías Ágiles en el Desarrollo de Software*.

Martin Fowler, Kendall Scott. (1999). *UML gota a gota*. Pearson Educación.

Navarro Cadavid, A., Fernández Martínez, J., & Morales Vélez, J. (2013). *Revisión de metodologías ágiles para el desarrollo de software*.

Rios, S. (2015). *JSF 2 + Hibernate 4 + Spring 4: PrimeFaces 5 with JAX-WS y EJB'S*.

- Salesiana, C. S. (2014). *Resoluciones - Acta N° 173-09-2014-10-15*. Universidad Politécnica Salesiana.
- Salesiana, U. P. (2012). *Estatuto de la Universidad Politécnica Salesiana 0002-01-2012-01-11*. Universidad Politécnica Salesiana.
- Soria Solís, I., & Condor Tinoco, E. (2015). *Compendio de Ingeniería De Software*.
- Suárez Torrente , M. (2011). *SIRIUS: Sistema de Evaluación de la Usabilidad Web Orientado al Usuario y basado en la Determinación de Tareas Críticas*. Universidad de Oviedo.
- Suite, B. (10 de Enero de 2016). *BPMN 2.0*. Obtenido de Bizagi:
<http://www.bizagi.com/>
- Sutherland, J., & Schwaber, K. (2013). *La Guía de Scrum*.
- Yuni, J., & Urbano, C. (2006). *Técnicas para investigar 2*. Editorial Brujas.

Anexos

Anexo 1. Diagramas de proceso

A continuación se describen los subprocesos relacionados con el proceso global de la unidad de titulación

Diagrama de proceso Prerrequisitos

En la Figura anexo 1 se puede observar el diagrama que describe el proceso no automatizado de prerrequisitos que fue útil para determinar funcionalidades necesarias para el inicio del proceso de titulación, funcionalidades relevantes como: en cada periodo es necesario asignar docentes, registrar convocatorias para temas, inscripciones y requisitos.

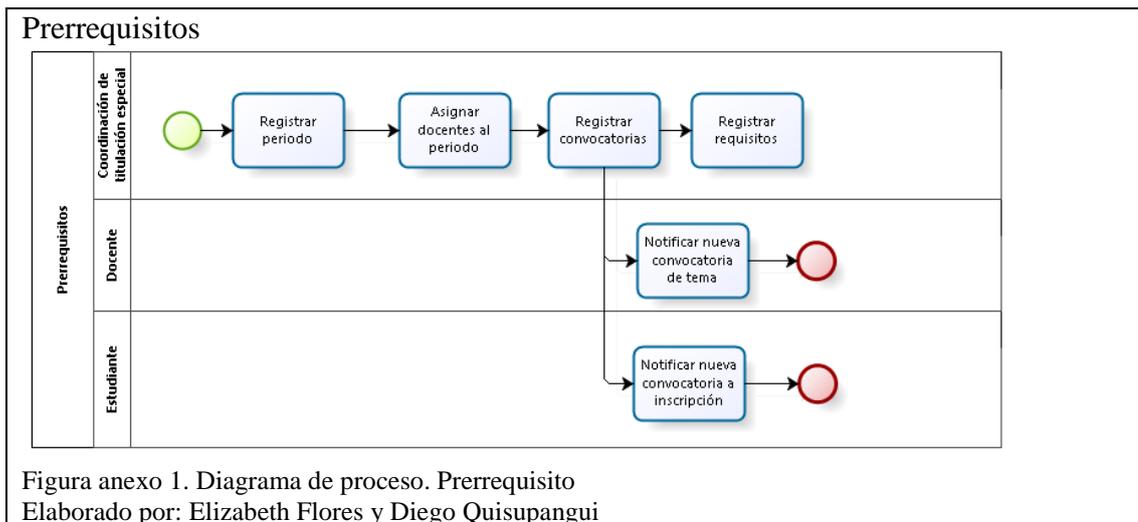


Diagrama de proceso Publicación de tema

En la Figura anexo 2 se observa el diagrama que describe el proceso para la gestión de publicación de temas, entonces este diagrama fue útil para identificar los actores implicados como: CUTS, Docente y Docente revisor, además de las funcionalidades que habrán de ser automatizadas.

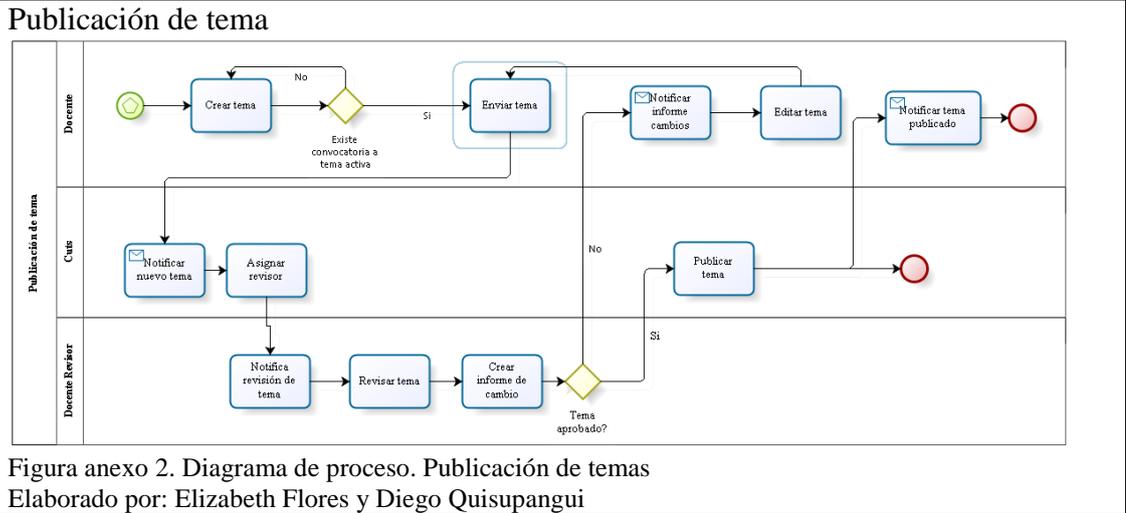


Diagrama de proceso Inscripción a unidad de titulación

En la Figura anexo 3 se observa el diagrama que describe el proceso de inscripción a unidad de titulación en el cual se identificaron los actores: Estudiante, CUTS, además de funcionalidades como: el proceso de inscripción se realiza una sola vez por cada periodo, el encargado de aprobar es la CUTS.

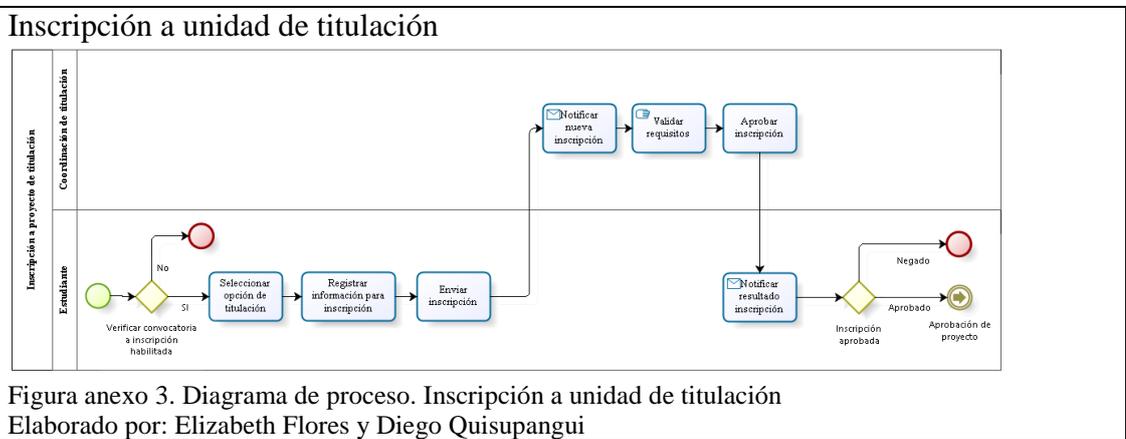


Diagrama de proceso Aprobación de proyecto

En la Figura anexo 4 se puede observar el diagrama que describe el proceso para la gestión de aprobación de proyecto, en el cual brindó la utilidad de identificar los actores: Docente, CUTS, Estudiante y diferenciar Docente tutor de Docente lector, además de funcionalidades que habrán de ser automatizadas y otras que tienen que

ser ejecutadas de forma manual como es el caso de la creación, desarrollo del plan, lectura de plan de proyecto y Pre revisión de documentos ya que este proceso es la validación física de los documentos del estudiante.

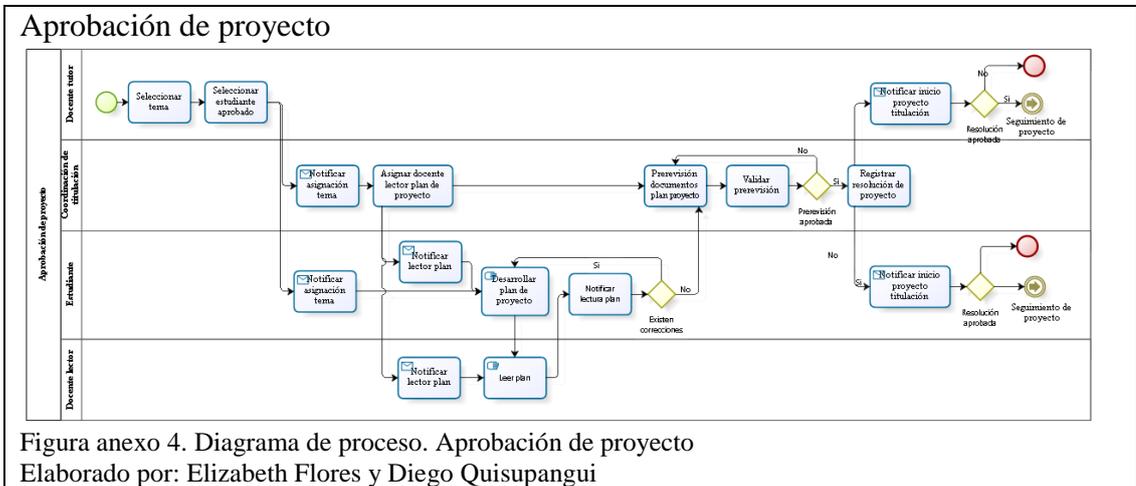


Diagrama de proceso Seguimiento de proyecto

En la Figura anexo 5 se puede observar el diagrama que describe el proceso de gestión para el seguimiento de proyectos, fue útil para identificar las siguientes funcionalidades: realizadas por el Docente tutor creación de cita, validación de hito; por el Estudiante creación de hito, validación de cita; por la CUTS asignar lector al proyecto, registro de nota final; y finalmente funcionalidades manuales como: Lectura del proyecto, corrección del documento y presentación del documento.

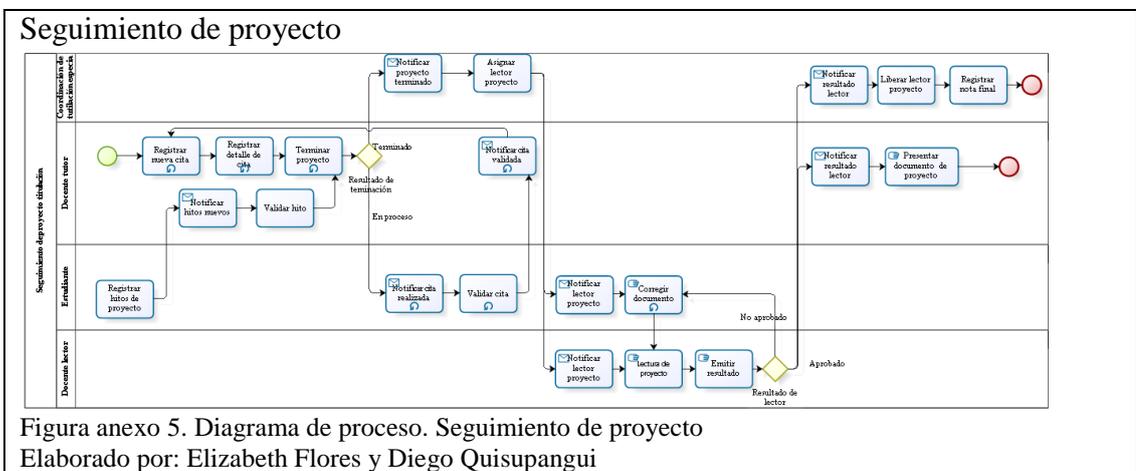
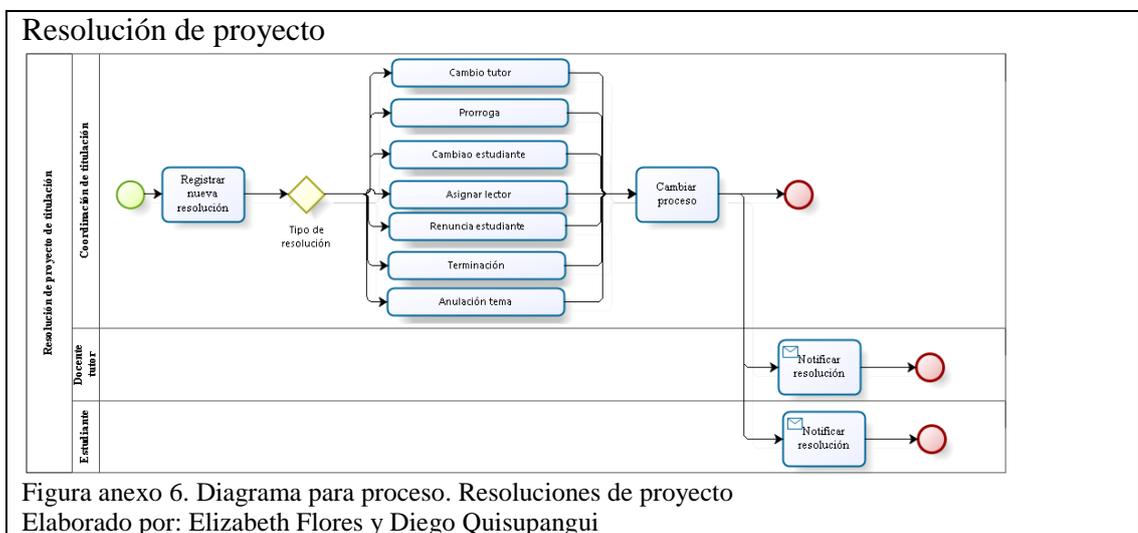


Diagrama de proceso de Resolución de proyecto

En la Figura anexo 6 se puede observar el diagrama para la gestión de resoluciones del proyecto de titulación, fue útil para identificar funcionalidades como: el listado de las resoluciones que afectan directamente al proyecto donde cada una afecta de diferente manera al desarrollo del mismo, además se identificaron los actores como son: Coordinación de titulación especial, docente, estudiante.



Anexo 2. Diagramas de clases

Diagrama global de paquetes

En la Figura anexo 7 se puede apreciar el esquema de paquetes que forman parte de la estructura de clases del sistema informático.

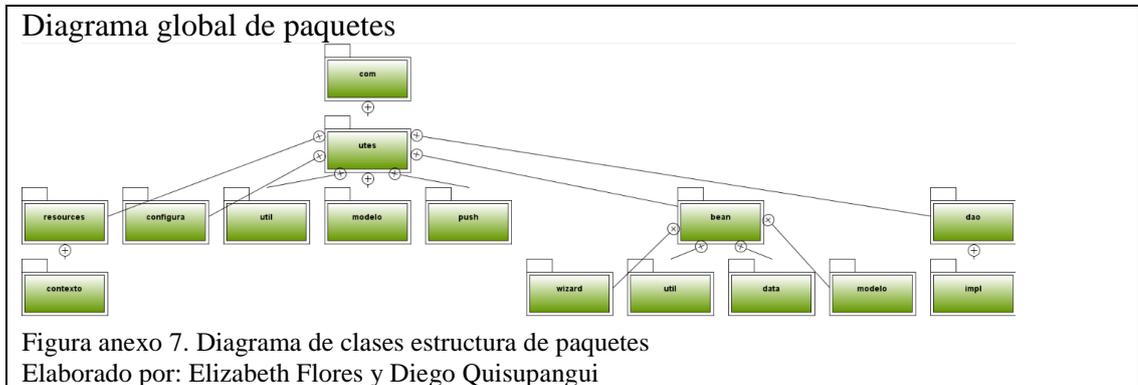
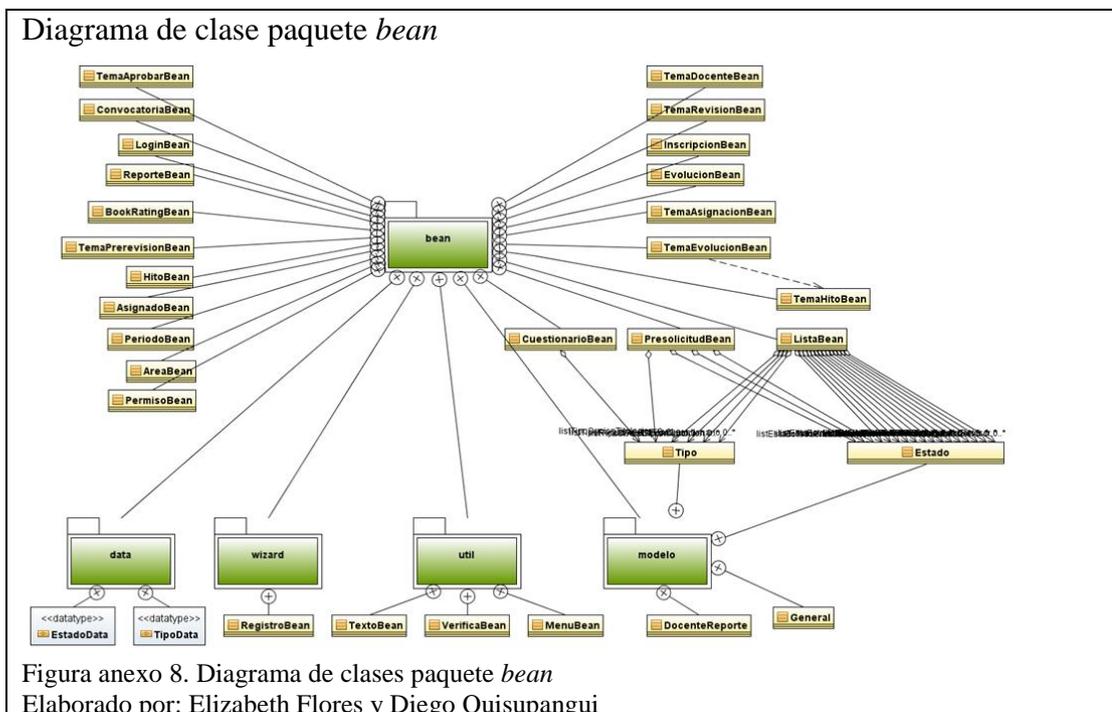


Diagrama de clase paquete *bean*

En la Figura anexo 8 se puede observar las clases *BEAN* que son parte del paquete denominado *bean*, las relaciones que puede tener con otras clases para la reutilización de código.



Ampliación de clase entidad *SysUsuario*

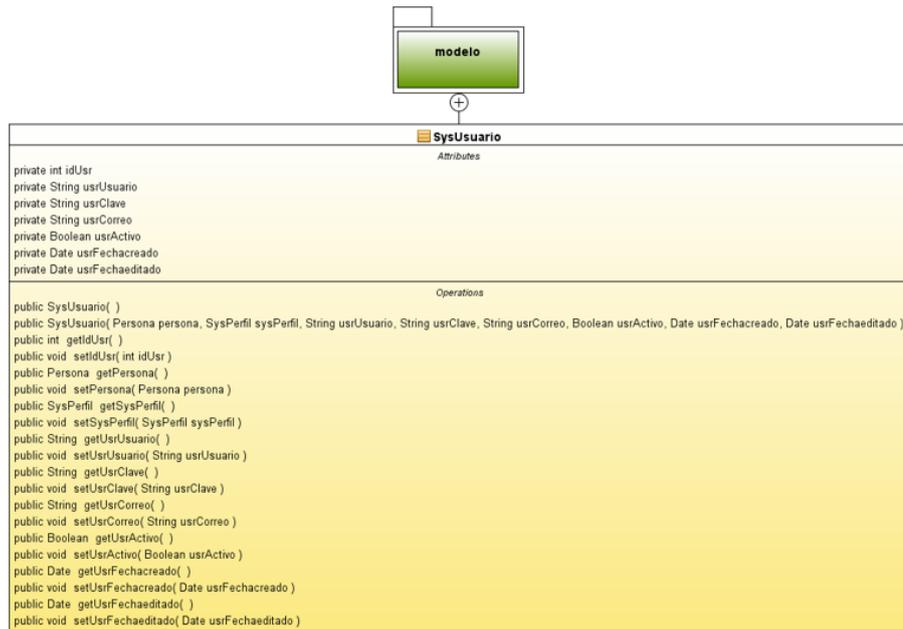


Figura anexo 11. Diagrama de clases *SysUsuario*
Elaborado por: Elizabeth Flores y Diego Quisupangui

Diagrama de clases paquete *dao*

En la Figura anexo 12 se puede apreciar el diagrama de clases para los paquetes denominados *dao* e *impl* respectivamente, el paquete *dao* tiene las interfaces que contienen los métodos que fueron implementados en el paquete *impl*.

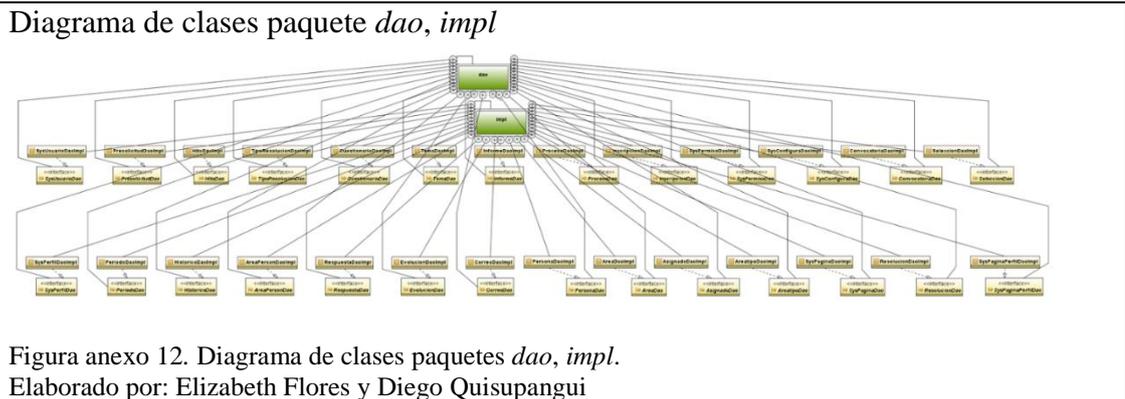


Figura anexo 12. Diagrama de clases paquetes *dao*, *impl*.
Elaborado por: Elizabeth Flores y Diego Quisupangui

En la Figura anexo 13 se tiene la ampliación de la interfaz *SysUsuarioDao* y la implementación *SysUsuarioDaoImpl* de los métodos usados.

Ampliación de interfaz *dao* y su respectiva implementación

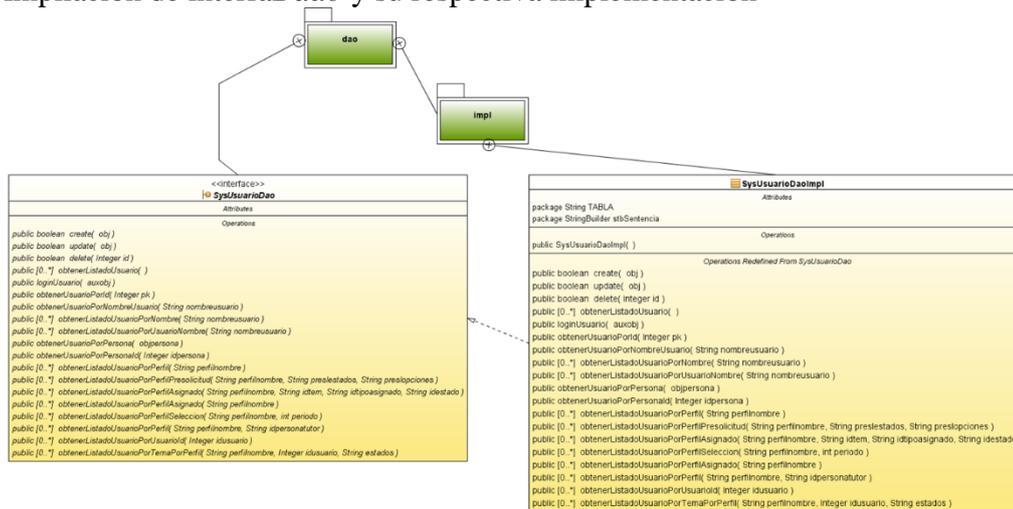


Figura anexo 13. Ampliación de interfaz y clase implementación
Elaborado por: Elizabeth Flores y Diego Quisupangui

Diagrama de clases paquete *util*

En la Figura anexo 14 se puede apreciar el diagrama de clases para el paquete denominado *util* que contiene las clases de ayuda en ciertos procesos como manejo de fechas, correo, validación de documento y cadenas.

Diagrama de clase paquete *util*

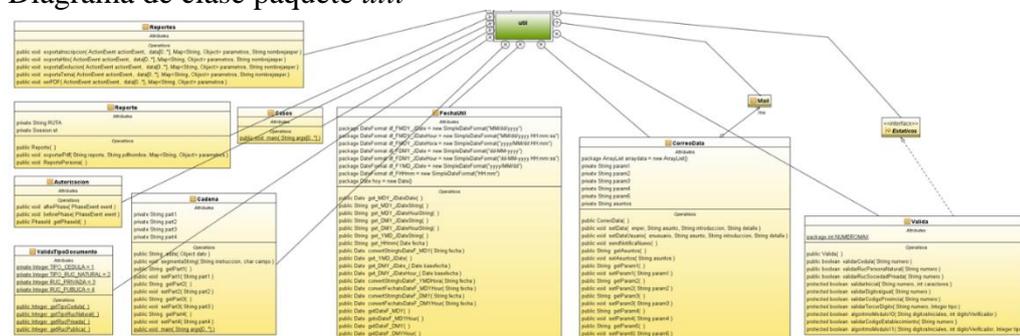


Figura anexo 14. Diagrama de clases paquete *util*
Elaborado por: Elizabeth Flores y Diego Quisupangui

Diagrama de clases paquete *resources*

En la Figura anexo 15 se puede apreciar el paquete denominado *resources* que contiene la clase que permite el manejo de archivo con extensión *.properties*, este es usado para almacenar varias de texto que no cambian en la ejecución del sistema informático.

Diagrama de clases paquete *resources*

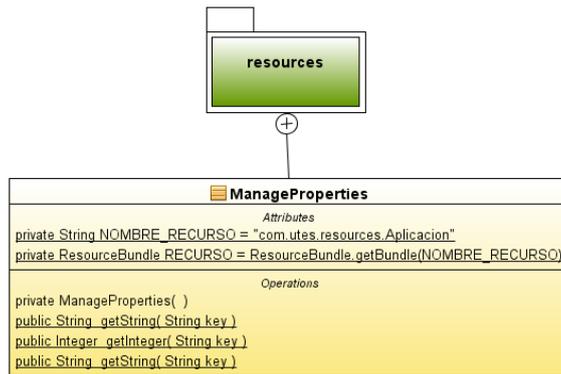


Figura anexo 15. Diagrama de clases paquete *resources*
Elaborado por: Elizabeth Flores y Diego Quisupangui

Anexo 3. Diccionario de base de datos

Tabla anexo 1. Diccionario de área

Nombre tabla:	area	Versión:	4.1
Descripción:	Tabla de datos para el manejo de áreas de especialidad		
No.	Campo	Tipo	Descripción
1	id_are	integer	Identificador de tabla área, secuencial, clave primaria
2	fk_aret	bigint	Identificador de la tabla área tipo
3	are_nombre	text	Nombre de área de especialidad
4	are_activo	boolean	Campo habilitar o deshabilitar registro
Relaciones:		Campos clave:	
fk_aret		id_are, fk_aret	

Nota: Diccionario de base de datos de tabla area.
Elaborado por: Elizabeth Flores y Diego Quisupangui

Tabla anexo 2. Diccionario de areapersona

Nombre tabla:	areapersona	Versión:	4.1
Descripción:	Tabla de datos para la asignación de área a persona		
No.	Campo	Tipo	Descripción
1	id_arper	integer	Identificador de tabla areapersona, secuencial, clave primaria
2	fk_per	bigint	Identificador de tabla persona
3	fk_are	bigint	Identificador de tabla área
4	arper_activo	boolean	Campo habilitar o deshabilitar registro
Relaciones:		Campos clave:	
fk_are, fk_per		id_arper, fk_are, fk_per	

Nota: Diccionario de base de datos de tabla areapersona.
Elaborado por: Elizabeth Flores y Diego Quisupangui

Tabla anexo 3. Diccionario de areatipo

Nombre tabla:	areatipo	Versión:	4.1
Descripción:	Tabla de datos que contiene los tipos de áreas		
No.	Campo	Tipo	Descripción
1	id_aret	integer	Identificador de tabla área tipo
2	aret_nombre	character varying	Nombre de tipo de área
3	aret_activo	boolean	Campo habilita o deshabilita registro
Relaciones:		Campos clave:	
		id_aret	

Nota: Diccionario de base de datos de tabla areatipo.
Elaborado por: Elizabeth Flores y Diego Quisupangui

Tabla anexo 4. Diccionario de asignación

Nombre tabla:		asignado	Versión:	4.1
Descripción:		Tabla de datos que contiene las personas asignadas a un tema		
No.	Campo	Tipo	Descripción	
1	id_asg	integer	Identificador de tabla asignación, secuencial, clave primaria	
2	fk_tem	bigint	Identificador de tabla tema	
3	fk_per	bigint	Identificador de tabla persona	
4	asg_idtipo	integer	Identificador de tipo de asignación	
5	asg_fecharegistro	date	Fecha de registro	
6	asg_activo	boolean	Campo habilita o deshabilita registro	
7	asg_idestadotema	integer	Identificador de estado de tema	
Relaciones:			Campos clave:	
fk_tem, fk_per			id_asg, fk_tem, fk_per, asg_idtipo, asg_idestadotema	

Nota: Diccionario de base de datos de tabla asignación.

Elaborado por: Elizabeth Flores y Diego Quisupangui

Tabla anexo 5. Diccionario de convocatoria

Nombre tabla:		convocatoria	Versión:	4.1
Descripción:		Tabla de datos que contiene las convocatorias		
No.	Campo	Tipo	Descripción	
1	id_con	integer	Identificador de tabla convocatoria, secuencial, clave primaria	
2	con_nombre	character varying	Nombre de convocatoria	
3	con_numerotema	integer	Número de temas para la convocatoria	
4	con_periodo	integer	Número de periodo en el cual se crea la convocatoria	
5	con_fechainicio	date	Fecha de inicio convocatoria	
6	con_fechafin	date	Fecha de fin de convocatoria	
7	con_activo	boolean	Campo habilita o deshabilita registro	
8	con_descripcion	character varying	Describe la convocatoria	
9	con_secuencia	integer	Número de secuencia para código de convocatoria	
Relaciones:			Campos clave:	
			con_nombre, con_numerotema	

Nota: Diccionario de base de datos de tabla convocatoria.

Elaborado por: Elizabeth Flores y Diego Quisupangui

Tabla anexo 6. Diccionario de cuestionario

Nombre tabla:		cuestionario	Versión: 4.1
Descripción:		Tabla de datos que contiene las preguntas	
No.	Campo	Tipo	Descripción
1	id_cue	integer	Identificador de cuestionario, clave primaria
2	cue_idtipo	integer	Número de tipo de cuestionario
3	cue_idinscripcion	integer	Número de inscripción
4	cue_pregunta	text	Pregunta de cuestionario
5	cue_activo	boolean	Habilita registro
Relaciones:		Campos clave:	
		cue_idtipo, cue_idinscripcion	

Nota: Diccionario de base de datos de tabla cuestionario.

Elaborado por: Elizabeth Flores y Diego Quisupangui

Tabla anexo 7. Diccionario de evolución.

Nombre tabla:		evolucion	Versión: 4.1
Descripción:		Tabla de datos que contiene evolución de trabajo de tema	
No.	Campo	Tipo	Descripción
1	id_evl	integer	Identificador de evolución, clave primaria
2	fk_tem	bigint	Identificador de tabla tema
3	evl_idestado	integer	Identificador de estado de evolución
4	evl_codigo	character varying	Código generado en caracteres
5	evl_fechacita	date	Fecha de cita
6	evl_fecharegistro	date	Fecha de registro desde el sistema
7	evl_tiempoduracion	integer	Tiempo de duración de la cita
8	evl_descripcion	text	Descripción de temas tratados en la cita
9	evl_tareas	text	Tareas para la cita
10	evl_valida	boolean	Validación de la cita
11	evl_fechavalida	date	Fecha en que se realiza la validación
12	evl_comentario	text	Comentario de la validación
13	evl_activo	boolean	Habilita o deshabilita registro
14	evl_secuencia	integer	Contador de registros por tema
15	evl_porcentaje	double precision	Porcentaje de avance del proyecto
16	evl_horacita	character varying	Hora de la cita
Relaciones:		Campos clave:	
fk_tem		id_evl, fk_tem, evl_fechacita, evl_horacita, evl_codigo	

Nota: Diccionario de base de datos de tabla evolución.

Elaborado por: Elizabeth Flores y Diego Quisupangui

Tabla anexo 8. Diccionario histórico

Nombre tabla:		historico	Versión: 4.1
Descripción:		Tabla de datos que contiene los registros históricos de cambios	
No.	Campo	Tipo	Descripción
1	pk_his	integer	Identificador de histórico
2	his_tema	integer	Identificador del tema que se realiza algún cambio
3	his_persona	integer	Identificador de la persona relacionado con el tema
4	his_fecha cambio	date	Fecha de cambio
5	his_campo	character varying	Detalle de campos cambiados
6	his_valor	text	Detalle de valores de campos cambiados
Relaciones:		Campos clave:	
		pk_his, his_tema, his_persona, his_campo	

Nota: Diccionario de base de datos de tabla histórico.
Elaborado por: Elizabeth Flores y Diego Quisupangui

Tabla anexo 9. Diccionario hito

Nombre tabla:		hito	Versión: 4.1
Descripción:		Tabla de datos que contiene hitos relacionado al tema	
No.	Campo	Tipo	Descripción
1	id_hit	integer	Identificador de hito, secuencial, clave primaria
2	fk_tem	bigint	Identificador de tabla tema
3	hit_idestado	integer	Identificador de estado de hito
4	hit_codigo (100)	character varying	Código generado en caracteres
5	hit_descripcion	text	Descripción de hito
6	hit_fecha entrega	date	Fecha de presentación de hito
7	hit_comentario	text	Comentario de validación de hito
8	hit_fecha valida	date	Fecha de revisión de hito
9	hit_secuencia	integer	Contador de registros por tema
10	hit_valida	boolean	Validación de hito
Relaciones:		Campos clave:	
fk_tem		id_hit, fk_tem, hit_idestado, hit_fecha entrega	

Nota: Diccionario de base de datos de tabla hito.
Elaborado por: Elizabeth Flores y Diego Quisupangui

Tabla anexo 10. Diccionario de informe

Nombre tabla:		informe	Versión:	4.1
Descripción:		Tabla de datos que contiene informes de cambio de tema		
No.	Campo	Tipo	Descripción	
1	id_inf	integer	Identificador de informe	
2	fk_tem	bigint	Identificador de tabla tema	
3	inf_idestado	integer	Identificador de estado de informe	
4	inf_fecha	date	Fecha registro de informe	
5	inf_informe	text	Detalle de informe	
6	inf_tema	text	Nombre de tema	
7	inf_observacion	text	Observación de informe	
8	inf_activo	boolean	Habilita o deshabilita registro	
9	fk_per	bigint	Identificador de tabla persona	
Relaciones:			Campos clave:	
fk_tem, fk_per			inf_informe, inf_idestado, fk_tem	

Nota: Diccionario de base de datos de tabla informe.

Elaborado por: Elizabeth Flores y Diego Quisupangui

Tabla anexo 11. Diccionario de inscripción

Nombre tabla:		inscripcion	Versión:	4.1
Descripción:		Tabla de datos que contiene inscripción creadas		
No.	Campo	Tipo	Descripción	
1	id_ins	integer	Identificador de inscripción	
2	ins_nombre	character varying	Código de inscripción	
3	ins_periodo	integer	Número de periodo	
4	ins_fechainicio	date	Fecha de inicio para registro de periodo	
5	ins_fechafin	date	Fecha de fin de registro de periodo	
6	ins_activo	boolean	Habilita o deshabilita registro	
7	ins_secuencia	integer	Contador de registros	
Relaciones:			Campos clave:	
			id_ins, ins_periodo, ins_fechainicio, ins_fechafin	

Nota: Diccionario de base de datos de tabla inscripción.

Elaborado por: Elizabeth Flores y Diego Quisupangui

Tabla anexo 12. Diccionario de periodo

Nombre tabla:	periodo	Versión:	4.1
Descripción:	Tabla de datos que contiene periodos creados		
No.	Campo	Tipo	Descripción
1	id_prd	integer	Identificador de periodo
2	prd_numero	integer	Número de periodo
3	prd_fechainicio	date	Fecha inicio periodo
4	prd_fechafin	date	Fecha fin periodo
5	prd_activo	boolean	Habilita o deshabilita registro
6	prd_anioinicio	integer	Año de inicio de periodo
7	prd_aniofin	integer	Año fin de periodo
Relaciones:		Campos clave:	
		id_prd, prd_numero, prd_fechainicio, prd_fecha fin	

Nota: Diccionario de base de datos de tabla periodo.

Elaborado por: Elizabeth Flores y Diego Quisupangui

Tabla anexo 13. Diccionario de persona

Nombre tabla:	persona	Versión:	4.1
Descripción:	Tabla de datos que contiene datos de persona		
No.	Campo	Tipo	Descripción
1	id_per	integer	Identificador de persona, secuencial, clave primaria
2	per_cedula (13)	character varying	Cédula de persona
3	per_nombre (75)	character varying	Nombres de persona
4	per_apellido (75)	character varying	Apellidos de persona
5	per_direccion (150)	character varying	Dirección de persona
6	per_telefono (15)	character varying	Teléfono de persona
7	per_celular (15)	character varying	Celular de persona
8	persexo (1)	character	Sexo
9	per_fecha creado	date	Fecha de registro
10	per_fecha editado	date	Fecha de edición
Relaciones:		Campos clave:	
		id_per, per_cedula, per_nombre, per_apellido	

Nota: Diccionario de base de datos de tabla persona.

Elaborado por: Elizabeth Flores y Diego Quisupangui

Tabla anexo 14. Diccionario de pre-solicitud

Nombre tabla:		presolicitud	Versión:	4.1
Descripción:		Tabla de datos que contiene las inscripciones		
No.	Campo	Tipo	Descripción	
1	id_psl	integer	Identificador de presolicitud	
2	fk_per	bigint	Identificador de tabla persona	
3	fk_ins	bigint	Identificador de tabla de inscripción	
4	psl_idestado	integer	Identificador de estado de presolicitud	
5	psl_idopcion	integer	Identificador de opción titulación	
6	psl_fecha	date	Fecha de registro de presolicitud	
7	psl_mensaje	text	Mensaje de criterios faltantes	
8	psl_fecharevision	date	Fecha revisión de la presolicitud	
9	psl_observacion	text	Observación de presolicitud	
10	psl_prerevision	text	Observación de presolicitud en prerevisión	
11	psl_fecha prerevision	date	Fecha de prerevisión	
12	psl_idestadoanterior	integer	Identificador de estado de la presolicitud	
13	psl_activo	boolean	Habilita o deshabilita registro	
Relaciones:			Campos clave:	
fk_per, fk_ins			psl_idestado, psl_idopcion	

Nota: Diccionario de base de datos de tabla pre-solicitud.

Elaborado por: Elizabeth Flores y Diego Quisupangui

Tabla anexo 15. Diccionario de resolución

Nombre tabla:		resolucion	Versión:	4.1
Descripción:		Tabla de datos que contiene las resoluciones del proyecto		
No.	Campo	Tipo	Descripción	
1	id_rsl	integer	Identificador de resolución	
2	fk_tem	bigint	Identificador de tabla tema	
3	fk_trsl	bigint	Identificador de tabla tipo resolución	
4	rsl_numero	text	Número de resolución	
5	rsl_fecharesolucion	date	Fecha de resolución	
6	rsl_fecha inicio	date	Fecha inicio de resolución aprobación	
7	rsl_fecha entrega	date	Fecha culminación de resolución aprobación	
8	rsl_activo	boolean	Habilita o deshabilita registro	
9	rsl_observacion	text	Observación de resolución	
10	rsl_idestado	integer	Identificador de estado	
11	fk_per	bigint	Identificador de tabla persona	
Relaciones:			Campos clave:	
fk_tem, fk_trsl, fk_per			id_rsl, rsl_numero, rsl_fecharesolución, rsl_idestado	

Nota: Diccionario de base de datos de tabla resolución.

Elaborado por: Elizabeth Flores y Diego Quisupangui

Tabla anexo 16. Diccionario de respuesta

Nombre tabla:		respuesta	Versión:	4.1
Descripción:		Tabla de datos que contiene respuestas de presolicitud		
No.	Campo	Tipo	Descripción	
1	id_res	integer	Identificador de respuesta, clave primaria	
2	fk_psl	bigint	Identificador de tabla presolicitud	
3	fk_cue	bigint	Identificador de tabla cuestionario	
4	res_valor	boolean	Valor de respuesta	
5	res_validacion	boolean	Valor de respuesta de validación	
6	res_prerevision	boolean	Valor de respuesta de prevalidación	
Relaciones:			Campos clave:	
fk_psl, fk_cue			res_valor, res_validacion, res_prevalidacion	

Nota: Diccionario de base de datos de tabla respuesta.

Elaborado por: Elizabeth Flores y Diego Quisupangui

Tabla anexo 17. Diccionario de selección.

Nombre tabla:		seleccion	Versión:	4.1
Descripción:		Tabla de datos que contiene los docentes en un periodo		
No.	Campo	Tipo	Descripción	
1	id_sel	integer	Identificador de selección, clave primaria	
2	fk_per	bigint	Identificador de tabla persona	
3	fk_prd	bigint	Identificador de tabla periodo	
4	sel_horaasignada	integer	Horas asignadas de proyectos	
5	sel_horavigente	integer	Horas de proyectos vigentes	
6	sel_horalectura	integer	Horas de lectura de proyectos	
Relaciones:			Campos clave:	
fk_per, fk_prd			id_sel	

Nota: Diccionario de base de datos de tabla selección.

Elaborado por: Elizabeth Flores y Diego Quisupangui

Tabla anexo 18. Diccionario de configuración

Nombre tabla:		sys_configuracion	Versión:	4.1
Descripción:		Tabla de datos que contiene configuración de sistema		
No.	Campo	Tipo	Descripción	
1	pk_conf	Integer	Identificador de configuración	
2	conf_campo (100)	character varying	Nombre del campo	
3	conf_estado	Boolean	Estado del campo	
4	conf_valor (250)	character varying	Valor del campo	
5	conf_tipo (50)	character varying	Tipo de campo	
6	conf_activo	Boolean	Habilita o deshabilita registro	
Relaciones:			Campos clave:	
			conf_campo, conf_estado, conf_valor	

Nota: Diccionario de base de datos de tabla configuración.

Elaborado por: Elizabeth Flores y Diego Quisupangui

Tabla anexo 19. Diccionario de página

Nombre tabla:		sys_pagina	Versión:	4.1
Descripción:		Tabla de datos que contiene páginas del sistema		
No.	Campo	Tipo	Descripción	
1	id_pag	integer	Identificador de página, secuencial, clave primaria	
2	pag_principal	integer	Identificador de página principal	
3	pag_orden	integer	Número de orden de pagina	
4	pag_etiqueta (50)	character varying	Nombre de la pagina	
5	pag_tipo (1)	character	Tipo de pagina	
6	pag_activo	boolean	Habilita o deshabilita registro	
7	pag_url (100)	character varying	Dirección de la pagina	
8	pag_titulo (50)	character varying	Título de la pagina	
Relaciones:			Campos clave:	
			pag_tipo, pag_url	

Nota: Diccionario de base de datos de tabla página.

Elaborado por: Elizabeth Flores y Diego Quisupangui

Tabla anexo 20. Diccionario de detalle página perfil

Nombre tabla:		sys_paginaperfil	Versión: 4.1
Descripción:		Tabla de datos que contiene paginas por perfil	
No.	Campo	Tipo	Descripción
1	id_pgprf	integer	Identificador, secuencial, clave primaria
2	fk_prf	bigint	Identificador de tabla perfil
3	fk_pag	bigint	Identificador de tabla pagina
4	pgprf_activo	boolean	Habilita o deshabilita registro
Relaciones:		Campos clave:	
fk_prf, fk_pag			

Nota: Diccionario de base de datos de tabla página de perfil.

Elaborado por: Elizabeth Flores y Diego Quisupangui

Tabla anexo 21. Diccionario de perfil

Nombre tabla:		sys_perfil	Versión: 4.1
Descripción:		Tabla de datos que contiene perfiles	
No.	Campo	Tipo	Descripción
1	id_prf	integer	Identificador de perfil, clave primaria
2	prf_nombre (30)	character varying	Nombre de perfil
3	prf_activo	boolean	Habilita o deshabilita registro
Relaciones:		Campos clave:	
		id_prf, prf_nombre	

Nota: Diccionario de base de datos de tabla perfil.

Elaborado por: Elizabeth Flores y Diego Quisupangui

Tabla anexo 22. Diccionario de permiso de perfil

Nombre tabla:		permiso	Versión: 4.1
Descripción:		Tabla de datos que contiene permiso de perfil	
No.	Campo	Tipo	Descripción
1	id_perm	integer	Identificador de permiso
2	fk_prf	bigint	Identificador de permiso
3	perm_leer	boolean	Habilita o deshabilita lectura
4	perm_editar	boolean	Habilita o deshabilita edición
5	perm_eliminar	boolean	Habilita o deshabilita eliminación
6	perm_crear	boolean	Habilita o deshabilita creación
7	perm_exportar	boolean	Habilita o deshabilita exportación
8	perm_activo	boolean	Habilita o deshabilita registro
Relaciones:		Campos clave:	

Nota: Diccionario de base de datos de tabla permiso de perfil.

Elaborado por: Elizabeth Flores y Diego Quisupangui

Tabla anexo 23. Diccionario de usuario

Nombre tabla:		sys_usuario	Versión: 4.1
Descripción:		Tabla de datos que contiene usuarios del sistema	
No	Campo	Tipo	Descripción
1	id_usr	integer	Identificador de usuario
2	fk_prf	bigint	Identificador de tabla perfil
3	fk_per	bigint	Identificador de tabla persona
4	usr_usuario	text	Nombre de usuario
5	usr_clave	text	Clave de usuario
6	usr_correo (75)	character varying	Correo de usuario
7	usr_activo	boolean	Habilita o deshabilita registro
8	usr_fecha creado	date	Fecha de registro de usuario
9	usr_fecha editado	date	Fecha de modificación
Relaciones:		Campos clave:	
fk_prf, fk_per		usr_usuario, usr_clave, usr_correo	

Nota: Diccionario de base de datos de tabla usuario.

Elaborado por: Elizabeth Flores y Diego Quisupangui

Tabla anexo 24. Diccionario de tema

Nombre tabla:		tema	Versión: 4.1
Descripción:		Tabla de datos que contiene temas	
No.	Campo	Tipo	Descripción
1	id_tem	integer	Identificador de tema
2	fk_con	bigint	Identificador de tabla convocatoria
3	fk_per	bigint	Identificador de tabla persona
4	tem_idtipo	integer	Identificador de opción de titulación
5	tem_idestado	integer	Identificador de estado
6	tem_nombre	text	Tema
7	tem_descripcion	text	Descripción de tema
8	tem_tema	text	Texto de tema
9	tem_numest	integer	Número de estudiantes para el tema
10	tem_autora	character varying	Nombres de autor 1
11	tem_autorb	character varying	Nombres de autor 2
12	tem_fecha publicado	date	Fecha de publicación
13	tem_auspiciante	text	Nombre de auspiciante
14	tem_lectorplan	character varying	Nombre de lector de plan
15	tem_lectorproyecto	character varying	Nombre de lector de proyecto
16	tem_activo	boolean	Habilita o deshabilita registro
17	tem_fecha creado	date	Fecha de registro

18	tem_fechaeditado	date	Fecha de modificación
19	tem_poravance	double precision	Porcentaje de avance
20	tem_aprobado	boolean	Valor de aprobado
21	tem_alcance	text	Alcance de tema
22	tem_objetivo	text	Objetivos de tema
23	tem_fechaenviado	date	Fecha de envío a revisión
24	tem_nota	double precision	Valor de nota final
25	tem_observacion	text	Observación de tema
26	tem_fechainicio	date	Fecha inicio proyecto
27	tem_fechaentrega	date	Fecha entrega proyecto
28	tem_idperiodo	integer	Id de periodo
30	tem_fecharesolucion	date	Fecha de resolución aprobación
Relaciones:			Campos clave:
fk_con, fk_per			tem_idtipo, tem_idestado, tem_nombre

Nota: Diccionario de base de datos de tabla tema.

Elaborado por: Elizabeth Flores y Diego Quisupangui

Tabla anexo 25. Diccionario de tipo de resolución

Nombre tabla:		tipo resolución	Versión:	4.1
Descripción:		Tabla de datos que contiene tipos de resoluciones		
No.	Campo	Tipo	Descripción	
1	id_trsl	integer	Identificador de tipo de resolución	
2	trsl_titulo (100)	character varying	Nombre de tipo de resolución	
3	trsl_descripcion	text	Descripción	
4	trsl_active	boolean	Habilita o deshabilita registro	
Relaciones:			Campos clave:	
			trsl_titulo	

Nota: Diccionario de base de datos de tabla tipo resolución.

Elaborado por: Elizabeth Flores y Diego Quisupangui

Anexo 4. Pruebas

Tabla anexo 26. Pruebas Funcionales

Proceso	Actor	Pre Requisito	Escenario	Pasos	Resultado Esperado	Resultado alcanzado
Inscripción	Estudiante	Cédula, nombre, apellido.	Registro al sistema (estudiante)	1. Ingresa datos de cédula, nombre, apellido, sexo(femenino, masculino) 2. Ingresar usuario, clave y repetir la clave 3. Verifica datos	Confirmación de datos, guarda correctamente	OK
		Usuario y contraseña	Ingreso al sistema (Estudiante)	1. Ingreso Usuario 2. Ingrese contraseña	Ingreso correcto al sistema	OK
		Opción de titulación	Inscripción a un nuevo proyecto o examen complejo	1. Escoger opción de titulación 2. Escoger requisitos 3. Guardar datos	Guarda y se visualiza inscripción de la opción de titulación (proyecto, artículo, examen)	Al momento de guardar la inscripción, guarda datos erróneos, ya que el componente no funciona correctamente
	CUTS		Validar inscripción CUTS	1. Selecciona estado de inscripción 2. Validación de requisitos para la inscripción	Visualización del estado respecto a la inscripción para el proyecto (Aprobado, aprobado en lista de espera, anulado)	OK
	Estudiante	Usuario y contraseña	Estado de Inscripción	1. Ingresar al menú de inscripción	Visualización del estado de la inscripción del estudiante	Se debe deshabilitar opciones de menú: hito, evolución

Nota: Pruebas funcionales realizadas al sistema de la inscripción.

Elaborado por: Elizabeth Flores y Diego Quisupangui

Proceso	Actor	Pre Requisito	Escenario	Pasos	Resultado Esperado	Resultado alcanzado
Publicación de Tema	Docente autor	Usuario y contraseña proporcionado por el administrador del sistema	Crear tema	1. Selecciona tipo	Guarda y se visualiza nuevo tema de proyecto o artículo de titulación	OK
				2. Ingresar título		
				3. Ingresar Descripción		
				4. Ingresar número de estudiantes		
	Enviar Tema		1. Seleccionar tema creado	Visualización del cambio de estado del tema y añadir convocatoria actual al tema; notificación vía email	OK	
			2. Enviar temas			
	Asignación de revisor de tema		1. Seleccionar tema	Nuevo estado cambia a Asignado	No refrescar la tabla asignación de tema y debe pasar a temas publicados.	
			2. Acciones: escoger revisor de tema			
	Revisión de Temas		1. Ingresar informe del tema revisado	Visualiza :	Se debe bloquear botón para no realizar más cambios después de ya haber guardado el estado.	
			2. Selecciona estado	Opción 1. Revisado		
	Opción 1. Publicar tema		1. Seleccionar tema	Visualiza en la tabla temas publicados		
			2. Publicar tema			
Opción 2. Publicar tema	1. Selecciona el tema con estado revisión	Visualiza el cambio de estado “Enviado” en los perfiles coordinación y docente autor	OK			
	2. Lectura del informe del tema					
	3. Editar temas					
	4. Enviar temas					

Nota: Pruebas funcionales realizadas al sistema de la publicación del tema.

Elaborado por: Elizabeth Flores y Diego Quisupangui

Proceso	Actor	Pre Requisito	Escenario	Pasos	Resultado Esperado	Resultado alcanzado
Aprobación de proyecto	Docente tutor	Usuario y contraseña proporcionado por el administrador del sistema	Asigna Estudiantes al proyecto	1. Visualización del Tema	Visualización de los estudiantes asignados al proyecto	OK
				2. Visualización del número de estudiantes necesarios para el proyecto		
				3. Escoger Estudiantes		
	CUTS		Asignar lector al plan	1. Acciones: visualización de datos del Proyecto	Visualización del lector del Plan	OK
				2. Escoger Lector del Plan		
	CUTS		Pre Revisión	1. Acciones: visualización de datos del tema	Visualización del cambio de estado en la tabla Pre Revisión-Temas	OK
				2. Acción: visualización de la validación de inscripción		
				3. Validación de requisitos para la aprobación del tema		
				4. Ingresar observación		
	CUTS		Aprobación	1. Acciones: visualización de datos del tema	Visualización del nuevo estado del tema (aprobado), se envía a aprobado.	Bloqueo nuevo, listado.
				2. Resolución: visualización de los datos para el registro de la resolución		
				3. Escoge tipo de resolución.		
4. Ingresar código de resolución						
5. Ingresar fecha de resolución						
6. Ingresar fecha de inicio del proyecto						
7. Ingresar fecha fin del fin del proyecto						
8. Ingresar una observación						
						Aparecen varias opciones y solo se debe visualizar la opción de aprobación

Nota: Pruebas funcionales realizadas al sistema de la aprobación del proyecto.

Elaborado por: Elizabeth Flores y Diego Quisupangui

Proceso	Actor	Pre Requisito	Escenario	Pasos	Resultado Esperado	Resultado alcanzado
Seguimiento de proyecto	Estudiante	Usuario y contraseña proporcionado por el administrador del sistema.	Hito	1. Acciones: visualización de los hitos del proyecto	Visualización de un nuevo hito en la ventana hitos del proyecto	OK
				2. Hito: visualización para un nuevo registro de hito		
				3. Ingresa fecha de entrega		
				4. Ingresa descripción del hito		
	Docente autor		Hito	1. Acciones: visualización de los hitos del proyecto	Visualización de la fecha de validación, cambio de estado en el perfil del docente y estudiante	OK
				2. Acciones: visualización del hito		
				3. Escoger estado		
				4. Ingresar comentario		
	Evolución de trabajo		1. Seleccionar tema	1. Visualización de la nueva cita en la tabla de evolución del tema.	Al Re-agendar cita se debe limpiar campos.	
			2. Acciones: visualización de la evolución del tema			
3. Cita: visualización para un nuevo registro de una cita						
4. Ingresar fecha de la cita						
5. Ingresar tarea a presentar						
6. Selección de la cita creada		2. Cambio de estado de la evolución del tema				
7. Registro de desarrollo de cita.						

Nota: Pruebas funcionales realizadas al sistema del seguimiento del proyecto.

Elaborado por: Elizabeth Flores y Diego Quisupangui

Pruebas de caja blanca última versión

Tabla anexo 27. Pruebas funcionales inscripción

Proceso	Actor	Pre Requisito	Escenario	Pasos	Resultado Esperado	Resultado alcanzado
Inscripción	Estudiante	Cédula, nombre, apellido.	Registro al sistema (estudiante)	1. Ingresar datos de cédula, nombre, apellido, sexo (femenino, masculino)	Confirmación de datos, guarda correctamente	OK
				2. Ingresar usuario, clave y repetir la clave		
				3. Verifica datos		
		Usuario y contraseña	Ingreso al sistema (Estudiante)	1. Ingreso Usuario	Ingreso correcto al sistema	OK
	2. Ingrese contraseña					
	Opción de titulación	Inscripción a un nuevo proyecto o examen complejo	1. Escoger opción de titulación	Guarda y se visualiza inscripción de la opción de titulación (proyecto, artículo, examen)	OK	
			2. Escoger requisitos			
			3. Guardar datos			
	Usuario y contraseña	Estado de Inscripción	1. Ingresar al menú de inscripción	Visualización del estado de la inscripción del estudiante	OK	
	CUTS	Validar inscripción CUTS	1. Selecciona estado de inscripción	Visualización del estado respecto a la inscripción para el proyecto (Aprobado, aprobado en lista de espera, anulado)	OK	
2. Validación de requisitos para la inscripción						

Nota: Pruebas funcionales realizadas después del cambio al sistema de la inscripción.

Elaborado por: Elizabeth Flores y Diego Quisupangui

Tabla anexo 28. Pruebas funcionales Publicación del tema

Proceso	Actor	Pre Requisito	Escenario	Pasos	Resultado Esperado	Resultado alcanzado
Publicación de Tema	Docente autor	Usuario y contraseña proporcionado por el administrador del sistema	Crear tema	1. Selecciona tipo	Guarda y se visualiza nuevo tema de proyecto o artículo de titulación	OK
				2. Ingresar título		
				3. Ingresar Descripción		
				4. Ingresar número de estudiantes		
	Enviar Tema		1. Seleccionar tema creado	Visualización del cambio de estado del tema y añadir convocatoria actual al tema; notificación vía email	OK	
			2. Enviar temas			
	CUTS		Asignación de revisor de tema	1. Seleccionar tema	Nuevo estado cambia a Asignado	OK
				2. Acciones: escoger revisor de tema		
	Docente revisor		Revisión de Temas	1. Ingresa informe del tema revisado	Visualiza :	OK
				2. Selecciona estado	Opción 1. Revisado Opción 2. En revisión	
	CUTS		Opción 1. Publicar tema	1. Seleccionar tema	Visualiza en la tabla temas publicados	OK
				2. Publicar tema		
Docente autor	Opción 2. Publicar tema	1. Selecciona el tema con estado revisión	Visualiza el cambio de estado a enviado en los perfiles coordinación y docente autor	OK		
		2. Lectura del informe del tema				
		3. Editar temas				
		4. Enviar temas				

Nota: Pruebas funcionales realizadas después del cambio de la publicación del tema.

Elaborado por: Elizabeth Flores y Diego Quisupangui

Tabla anexo 29. Pruebas funcionales aprobación de proyecto

Proceso	Actor	Pre Requisito	Escenario	Pasos	Resultado Esperado	Resultado alcanzado						
Aprobación de proyecto	Docente tutor	Usuario y contraseña proporcionado por el administrador del sistema	Asigna Estudiantes al proyecto	1. Visualización del Tema	Visualización de los estudiantes asignados al proyecto	OK						
				2. Visualización del número de estudiantes necesarios para el proyecto								
				3. Escoger Estudiantes								
	CUTS		Asignar lector al plan	1. Acciones: visualización de datos del Proyecto	Visualización del lector del Plan	OK						
				2. Escoger Lector del Plan								
	Pre Revisión		1. Acciones: visualización de datos del tema	2. Acción: visualización de la validación de inscripción	3. Validación de requisitos para la aprobación del tema	4. Ingresar observación	Visualización del cambio de estado en la tabla Pre Revisión-Temas	OK				
									Aprobación	1. Acciones: visualización de datos del tema	Visualización del nuevo estado del tema (aprobado), se envía a aprobado.	OK
										2. Resolución: visualización de los datos para el registro de la resolución		
										3. Selección tipo de resolución.		
	Aprobación		4. Ingresar código de resolución	Visualización del nuevo estado del tema (aprobado), se envía a aprobado.	OK							
			5. Ingresar fecha de resolución									
			6. Ingresar fecha de inicio del proyecto									
			7. Ingresar fecha fin del fin del proyecto									
			8. Ingresar una observación									

Nota: Pruebas funcionales realizadas después del cambio al sistema de la aprobación del proyecto.

Elaborado por: Elizabeth Flores y Diego Quisupangui

Tabla anexo 30. Pruebas funcionales seguimiento del proyecto

Proceso	Actor	Pre Requisito	Escenario	Pasos	Resultado Esperado	Resultado alcanzado
Seguimiento de proyecto	Estudiante	Usuario y contraseña proporcionado por el administrador del sistema	Hito	1. Acciones: visualización de los hitos del proyecto	Visualización de un nuevo hito en la ventana hitos del proyecto	OK
				2. Hito: visualización para un nuevo registro de hito		
				3. Ingresar fecha de entrega		
				4. Ingresar descripción del hito		
	Docente autor		Hito	1. Acciones: visualización de los hitos del proyecto	Visualización de la fecha de validación, cambio de estado en el perfil del docente y estudiante	OK
				2. Acciones: visualización del hito		
				3. Escoger estado		
				4. Ingresar comentario		
	Evolución de trabajo		Evolución de trabajo	1. Seleccionar tema	1. Visualización de la nueva cita en la tabla de evolución del tema.	OK
				2. Acciones: visualización de la evolución del tema		
				3. Cita: visualización para un nuevo registro de una cita		
				4. Ingresar fecha de la cita	2. Cambio de estado de la evolución del tema	
				5. Ingresar tarea a presentar		
				6. Selección de la cita creada		
7. Registro de desarrollo de cita.						

Nota: Pruebas funcionales realizadas después del cambio del seguimiento del proyecto.

Elaborado por: Elizabeth Flores y Diego Quisupangui

Resultados de pruebas funcionales

Tabla anexo 31. Pruebas funcionales Inscripción

Actores	Proceso	Resultado	Estado	Observación
Estudiante	Inscripción	Se realiza correctamente la inscripción de estudiantes.	Cerrado	Se realizó cambios en los defectos encontrados:
				No se guarda datos erróneos el componente funciona correctamente.
Coordinación				Se deshabilito las opciones del menú (hito, evolución).

Nota: Pruebas funcionales, observación del cambio después de las pruebas.

Elaborado por: Elizabeth Flores y Diego Quisupangui

Tabla anexo 32. Pruebas funcionales publicación tema

Actores	Proceso	Resultado	Estado	Observación
Coordinación	Publicación del tema	Se realiza correctamente la publicación del tema	Cerrado	Se realizó cambios en los defectos encontrados
Docente revisor				Se refresca la tabla de asignación de tema, y pasa a temas publicados.
Docente Autor				Se bloquea botón para no realizar cambios después de haber ya guardado el estado.

Nota: Pruebas funcionales, observación del cambio después de las pruebas.

Elaborado por: Elizabeth Flores y Diego Quisupangui

Tabla anexo 33. Pruebas funcionales aprobación del proyecto

Actores	Proceso	Resultado	Estado	Observación
Coordinación	Aprobación del proyecto	Se realiza correctamente la aprobación del proyecto	Cerrado	Se realiza el cambio para que solo se pueda visualizar la aprobación del tema y no otras opciones
Docente				

Nota: Pruebas funcionales, observación del cambio después de las pruebas

Elaborado por: Elizabeth Flores y Diego Quisupangui

Pruebas no funcionales

Los datos a evaluar por Sirius fueron explicados en el marco teórico.

En las figuras desde la Figura anexo 16 a la Figura anexo 25 muestran los datos obtenidos en la evaluación realizada a la aplicación, con los resultados y la fórmula de Sirius se obtiene el porcentaje de usabilidad de la aplicación.

Aspectos Generales

CRITERIOS DEL "HEURÍSTICO ASPECTOS GENERALES": Elementos relacionados con los objetivos del sitio, el look & feel, coherencia y nivel de actualización de contenidos.

Código	Criterio	Valor	Relevancia	Comentarios	Valor interno	Valor interno de relevancia	Factor de corrección	Valor parcial	Valor parcial 2
AG1	Objetivo del sitio web concreto y bien definido	5	MA		5	4	0.0270270270	0.2424242424	0.270270270270270
AG2	Contenidos y servicios ofrecidos precisos y completos	6	CR		6	8	0.0540405404	0.432424324	0.540540540405404
AG3	Estructura general del sitio web orientada al usuario	7	MA		7	4	0.0270270270	0.1898989898	0.270270270270270
AG4	Look & feel general que corresponde con los objetivos, características, contenidos y servicios del sitio	6	ME		6	2	0.0315151515	0.1630303030	0.1515151515151515
AG5	Diseño general del sitio web reconocible	7	ME		7	2	0.0315151515	0.0945454545	0.1515151515151515
AG6	Diseño general del sitio web coherente	5	MA		5	4	0.0270270270	0.2424242424	0.270270270270270
AG7	Se utiliza el idioma del usuario	5	MA		5	4	0.0270270270	0.2702702702	0.270270270270270
AG8	Se da soporte a otros idiomas	NA			0				
AG9	Traducción del sitio completa y correcta	NA			0				
AG10	Sitio web actualizado periódicamente	NA			0				
									28

Instrucciones
 1. Evalúe cada criterio y rellene la columna "Valor" según la leyenda.
 2. Añada sus comentarios o la referencia a los mismos en otro documento anexo en la columna "Comentarios".

Leyenda Valor
 0 No se cumple en absoluto
 10 Se cumple totalmente
 NA Criterio no aplicable en el sitio
 NTS No se cumple en todo el sitio

NEP No se cumple en los enlaces principales
 NFP No se cumple en la página principal
 NPI No se cumple en alguna página interior
 S Se cumple el criterio

Figura anexo 16. Aspectos Generales de la aplicación
 Elaborado por: Elizabeth Flores y Diego Quisupangui.

Identidad e información

CRITERIOS DEL "IDENTIDAD E INFORMACIÓN": Elementos relacionados con la identidad del sitio, la información proporcionada sobre el proveedor y la autoría de los contenidos.

Código	Criterio	Valor	Relevancia	Comentarios	Valor interno	Valor interno de relevancia	Factor de corrección	Valor parcial	Valor parcial 2
I1	Identidad o logotipo significativo, identificable y suficientemente visible	4	MO		4	1	0.0067567568	0.0270270270	0.067567567568
I2	Identidad del sitio en todas las páginas	NA			0				
I3	Estlogan o tagline adecuado al objetivo del sitio	NA			0				
I4	Se ofrece información sobre el sitio web, empresa	NA			0				
I5	Existen mecanismos de contacto	NA			0				
I6	Se ofrece información sobre la protección de datos de carácter personal o los derechos de autor de los contenidos del sitio web	S	ME		10	2	0.0135135135	0.1351351351	0.135135135135135
I7	Se ofrece información sobre el autor, fuentes y fechas de creación y revisión en artículos, noticias, informes	NA			0				
									3

Instrucciones
 1. Evalúe cada criterio y rellene la columna "Valor" según la leyenda.
 2. Añada sus comentarios o la referencia a los mismos en otro documento anexo en la columna "Comentarios".

Leyenda Valor
 0 No se cumple en absoluto
 10 Se cumple totalmente
 NA Criterio no aplicable en el sitio
 NTS No se cumple en todo el sitio

NEP No se cumple en los enlaces principales
 NFP No se cumple en la página principal
 NPI No se cumple en alguna página interior
 S Se cumple el criterio

Figura anexo 17. Identidad e información de la aplicación
 Elaborado por: Elizabeth Flores y Diego Quisupangui.

Estructura y navegación

CRITERIOS DEL "ESTRUCTURA Y NAVEGACIÓN": Elementos relacionados con la idoneidad de la arquitectura de la información y la navegación del sitio.

Código	Criterio	Valor	Relevancia	Comentarios	Valor interno	Valor interno de relevancia	Factor de corrección	Valor parcial	Valor parcial 2
EN1	Se ha evitado pantalla de bienvenida	S	MO		10	1	0.0067567568	0.067567567568	0.067567567568
EN2	Estructura de organización y navegación adecuada	10	MA		10	4	0.0270270270	0.2702702702	0.270270270270270
EN3	Organización de elementos consistente con las convenciones	9	ME		9	2	0.0135135135	0.1824242424	0.135135135135135
EN4	Control del número de elementos y de términos por elemento en los menús de navegación	10	MA		10	4	0.0270270270	0.2702702702	0.270270270270270
EN5	Equilibrio entre profundidad y anchura en el caso de estructura jerárquica	S	MA		10	4	0.0270270270	0.2702702702	0.270270270270270
EN6	Enlaces fácilmente reconocibles como tales	NA			0				
EN7	La caracterización de los enlaces indica su estado (visitados, activos)	S	MA		10	4	0.0270270270	0.2702702702	0.270270270270270
EN8	No hay redundancia de enlaces	S	MA		10	4	0.0270270270	0.2702702702	0.270270270270270
EN9	No hay enlaces rotos	S	MA		10	4	0.0270270270	0.2702702702	0.270270270270270
EN10	No hay enlaces que lleven a la misma página que se está visualizando	S	MO		10	1	0.0067567568	0.067567567568	0.067567567568
EN11	En las imágenes de enlace se indica el contenido al que se va a acceder	NA			0				
EN12	Entre un enlace para volver al inicio en cada página	NA			0				
EN13	Existen elementos de navegación que orienten al usuario acerca de dónde está y cómo deshacer su navegación (ej. migas)	NA			0				
EN14	Existe mapa del sitio para acceder directamente a los contenidos sin navegar	NA			0				
									28

Instrucciones
 1. Evalúe cada criterio y rellene la columna "Valor" según la leyenda.
 2. Añada sus comentarios o la referencia a los mismos en otro documento anexo en la columna "Comentarios".

Leyenda Valor
 0 No se cumple en absoluto
 10 Se cumple totalmente
 NA Criterio no aplicable en el sitio
 NTS No se cumple en todo el sitio

NEP No se cumple en los enlaces principales
 NFP No se cumple en la página principal
 NPI No se cumple en alguna página interior
 S Se cumple el criterio

Figura anexo 18. Estructura y navegación de la aplicación.
 Elaborado por: Elizabeth Flores y Diego Quisupangui.

Rotulado

CRITERIOS DEL "ROTULADO": Elementos relacionados con la significación, corrección y familiaridad del rotulado de los contenidos.

Código	Criterio	Valor	Relevancia	Comentarios	Valor interno	Valor interno de relevancia	Factor de corrección	Valor parcial	Valor parcial 2
RO1	Rotulos significativos	NA			0				
RO2	Sistema de rotulado controlado y preciso	S	ME		10	2	0.0135135135	0.1351351351	0.135135135135135
RO3	Título de las páginas, correcto y clarificado	S	ME		10	2	0.0135135135	0.1351351351	0.135135135135135
RO4	URL principal correcta, clara y fácil de recordar	7	MO		7	1	0.0067567568	0.0472372373	0.067567567568
RO5	URLs de páginas internas claras	NFP	ME		5	2	0.0135135135	0.0675675676	0.135135135135135
RO6	URLs de páginas internas permanentes	S	ME		10	2	0.0135135135	0.1351351351	0.135135135135135
									9

Instrucciones
 1. Evalúe cada criterio y rellene la columna "Valor" según la leyenda.
 2. Añada sus comentarios o la referencia a los mismos en otro documento anexo en la columna "Comentarios".

Leyenda Valor
 0 No se cumple en absoluto
 10 Se cumple totalmente
 NA Criterio no aplicable en el sitio
 NTS No se cumple en todo el sitio

NEP No se cumple en los enlaces principales
 NFP No se cumple en la página principal
 NPI No se cumple en alguna página interior
 S Se cumple el criterio

Figura anexo 19. Rotulado de la aplicación
 Elaborado por: Elizabeth Flores y Diego Quisupangui.

Layout de la página

CRITERIOS DEL "LAYOUT DE LA PÁGINA": Elementos relacionados con la distribución y el aspecto de los elementos de navegación e información en la interfaz									
Código	Criterio	Valor	Relevancia	Comentarios	Valor interno	Valor interno de relevancia	Factor de corrección	Valor parcial	Valor parcial
LA.1	Se aprovechan las zonas de alta jerarquía informativa de la página para contenidos de mayor relevancia	S	ME		10	2	0.0135135135	0.1351351351	0.1351351351
LA.2	Se ha evitado la sobrecarga informativa	S	ME		10	2	0.0135135135	0.1351351351	0.1351351351
LA.3	Es una interfaz limpia, sin ruido visual	S	ME		10	2	0.0135135135	0.1351351351	0.1351351351
LA.4	Existen zonas en blanco entre los objetos informativos de la página para poder desahogar la vista	S	ME		10	2	0.0135135135	0.1351351351	0.1351351351
LA.5	Uso correcto del espacio visual de la página	3	ME		3	2	0.0135135135	0.1216216216	0.1351351351
LA.6	Se utiliza correctamente la jerarquía visual para expresar las relaciones del tipo "parte de" entre los elementos de la página	3	ME		3	2	0.0135135135	0.1216216216	0.1351351351
LA.7	Se ha controlado la longitud de página	S	ME		10	2	0.0135135135	0.1351351351	0.1351351351
LA.8	La versión impresa de la página es correcta	S	MA		10	4	0.0270270270	0.2702702703	0.2702702703
LA.9	El texto de la página se lee sin dificultad	S	MA		10	4	0.0270270270	0.2702702703	0.2702702703
LA.10	Se ha evitado el texto parpadante y deslizante	S	ME		10	2	0.0135135135	0.1351351351	0.1351351351
Instrucciones					Legenda Valor				
1. Evalúe cada criterio y rellene la columna "Valor" según la leyenda					0	No se cumple en absoluto		NEP	No se cumple en los enlaces principales
2. Añada sus comentarios o la referencia a los mismos en otro documento anexo en la columna "Comentarios"					10	Se cumple totalmente		NPP	No se cumple en la página principal
					NA	Criterio no aplicable en el sitio		NPI	No se cumple en alguna página interior
					NTS	No se cumple en todo el sitio		S	Se cumple el criterio

Figura anexo 20. Layout de la página

Elaborado por: Elizabeth Flores y Diego Quisupangui.

Entendibilidad y factibilidad

CRITERIOS DEL "ENTENDIBILIDAD Y FACILIDAD EN LA INTERACCIÓN": Elementos relacionados con la adecuación y calidad de los contenidos textuales, iconos y controles de la interfaz									
Código	Criterio	Valor	Relevancia	Comentarios	Valor interno	Valor interno de relevancia	Factor de corrección	Valor parcial	Valor parcial 2
EF.1	Se emplea un lenguaje claro y conciso	S	MA		10	4	0.0270270270	0.2702702703	0.2702702703
EF.2	Lenguaje amigable, familiar y cercano	S	ME		10	2	0.0135135135	0.1351351351	0.1351351351
EF.3	Cada párrafo expresa una idea	S	ME		10	2	0.0135135135	0.1351351351	0.1351351351
EF.4	Uso consistente de los controles de la interfaz	S	ME		10	2	0.0135135135	0.1351351351	0.1351351351
EF.5	Mejoras visuales reconocibles y comprensibles por cualquier usuario (ej: iconos)	S	ME		10	2	0.0135135135	0.1351351351	0.1351351351
EF.6	Si se usan menús desplegable, orden coherente o alfabético	S	ME		10	2	0.0135135135	0.1351351351	0.1351351351
EF.7	Si el usuario tiene que rellenar un campo, las opciones disponibles se pueden seleccionar en vez de tener que escribir	NTS	ME		0	2	0.0135135135	0.0000000000	0.1351351351
Instrucciones					Legenda Valor				
1. Evalúe cada criterio y rellene la columna "Valor" según la leyenda					0	No se cumple en absoluto		NEP	No se cumple en los enlaces principales
2. Añada sus comentarios o la referencia a los mismos en otro documento anexo en la columna "Comentarios"					10	Se cumple totalmente		NPP	No se cumple en la página principal
					NA	Criterio no aplicable en el sitio		NPI	No se cumple en alguna página interior
					NTS	No se cumple en todo el sitio		S	Se cumple el criterio

Figura anexo 21. Entendibilidad y factibilidad de la aplicación.

Elaborado por: Elizabeth Flores y Diego Quisupangui.

Control y retroalimentación

CRITERIOS DEL "CONTROL Y RETORALIMENTACIÓN": Elementos relacionados con libertad del usuario en la navegación y la información proporcionada al mismo en el proceso de interacción con el sitio.									
Código	Criterio	Valor	Relevancia	Comentarios	Valor interno	Valor interno de relevancia	Factor de corrección	Valor parcial	Valor parcial 2
CR.1	El usuario tiene todo el control sobre la interfaz	S	ME		10	2	0.0135135135	0.1351351351	0.1351351351
CR.2	Se informa al usuario acerca de lo que está pasando	S	MD		10	1	0.0067567568	0.0675675676	0.0675675676
CR.3	Se informa al usuario de lo que ha pasado	S	ME		10	2	0.0135135135	0.1351351351	0.1351351351
CR.4	evitar errores	S	ME		10	2	0.0135135135	0.1351351351	0.1351351351
CR.5	Cuando se produce un error, se informa de forma clara y no alarmista al usuario de lo ocurrido y de cómo solucionar el problema	S	MD		10	1	0.0067567568	0.0675675676	0.0675675676
CR.6	Se ha controlado el tiempo de respuesta	S	MD		10	1	0.0067567568	0.0675675676	0.0675675676
CR.7	Se ha evitado que las ventanas del sitio anulen o se superpongan a la del navegador	S	ME		10	2	0.0135135135	0.1351351351	0.1351351351
CR.8	Se ha evitado la proliferación de ventanas en la pantalla del usuario	S	ME		10	2	0.0135135135	0.1351351351	0.1351351351
CR.9	Se ha evitado la desatención por parte del usuario de páginas adicionales	NA			0				
CR.10	Si existen tareas de varios pasos, se indica al usuario en cual está y cuantos faltan para completar la tarea	NTS	MD		0	1	0.0067567568	0.0000000000	0.0675675676
Instrucciones					Legenda Valor				
1. Evalúe cada criterio y rellene la columna "Valor" según la leyenda					0	No se cumple en absoluto		NEP	No se cumple en los enlaces principales
2. Añada sus comentarios o la referencia a los mismos en otro documento anexo en la columna "Comentarios"					10	Se cumple totalmente		NPP	No se cumple en la página principal
					NA	Criterio no aplicable en el sitio		NPI	No se cumple en alguna página interior
					NTS	No se cumple en todo el sitio		S	Se cumple el criterio

Figura anexo 22. Control y retroalimentación de la página.

Elaborado por: Elizabeth Flores y Diego Quisupangui.

Elementos multimedia

CRITERIOS DEL "ELEMENTOS MULTIMEDIA": Elementos relacionados con el grado de adecuación de los contenidos multimedia al sitio web.									
Código	Criterio	Valor	Relevancia	Comentarios	Valor interno	Valor interno de relevancia	Factor de corrección	Valor parcial	Valor
EM.1	Fotografías bien recortadas	NA			0				
EM.2	Fotografías comprensibles	NA			0				
EM.3	Fotografías con correcta resolución	NA			0				
EM.4	El uso de imágenes o animaciones proporciona algún tipo de valor añadido	NA			0				
EM.5	Se ha evitado el uso de animaciones ciclicas	NA			0				
EM.6	El uso de sonido proporciona algún tipo de valor añadido	NA			0				
Instrucciones					Legenda Valor				
1. Evalúe cada criterio y rellene la columna "Valor" según la leyenda					0	No se cumple en absoluto		NEP	No se cumple en los enlaces principales
2. Añada sus comentarios o la referencia a los mismos en otro documento anexo en la columna "Comentarios"					10	Se cumple totalmente		NPP	No se cumple en la página principal
					NA	Criterio no aplicable en el sitio		NPI	No se cumple en alguna página interior
					NTS	No se cumple en todo el sitio		S	Se cumple el criterio

Figura anexo 23. Elementos multimedia de la página.

Elaborado por: Elizabeth Flores y Diego Quisupangui.

Búsqueda

CRITERIOS DEL "BÚSQUEDA": Elementos relacionados con el buscador implementado en el sitio web.

Código	Criterio	Valor	Relevancia	Comentarios	Valor interno	Valor interno de relevancia	Factor de corrección	Valor parcial	Valor parcial
BU.1	La búsqueda, si es necesaria, se encuentra accesible desde todas las páginas del sitio	NA			0				
BU.2	Es fácilmente reconocible como tal	5	ME		5	2	0.0135135135	0.0675675676	0.1351351351
BU.3	Se encuentra fácilmente accesible	9	ME		9	2	0.0135135135	0.1216216216	0.1351351351
BU.4	La caja de texto es lo suficientemente ancha	9	MA		9	4	0.0270270270	0.2432432432	0.2702702702
BU.5	Sistema de búsqueda simple y claro	9	CR		9	8	0.0540540541	0.4864864865	0.5405405405
BU.6	Permite la búsqueda avanzada	3	ME		3	2	0.0135135135	0.0405405405	0.1351351351
BU.7	Muestra los resultados de la búsqueda de forma comprensible para el usuario	9	MA		9	4	0.0270270270	0.2432432432	0.2702702702
BU.8	Asiste al usuario en caso de no poder ofrecer resultados para una consultada dada	2	MA		2	4	0.0270270270	0.0540540541	0.2702702702
					26				

Instrucciones

1. Evalúe cada criterio y rellene la columna "Valor" según la leyenda

2. Añada sus comentarios o la referencia a los mismos en otro documento anexo en la columna "Comentarios"

Leyenda Valor

0 No se cumple en absoluto

10 Se cumple totalmente

NA Criterio no aplicable en el sitio

NTS No se cumple en todo el sitio

NEP

NPP

NPI

S

No se cumple en los enlaces principales

No se cumple en la página principal

No se cumple en alguna página interior

Se cumple el criterio

Figura anexo 24. Búsqueda de la página.

Elaborado por: Elizabeth Flores y Diego Quisupangui.

Ayuda

CRITERIOS DEL "AYUDA": Elementos relacionados con la ayuda ofrecida al usuario durante la navegación por el sitio.

Código	Criterio	Valor	Relevancia	Comentarios	Valor interno	Valor interno de relevancia	Factor de corrección	Valor parcial
AY.1	El enlace a la sección de Ayuda está colocado en una zona visible y estándar	NA			0			
AY.2	Fácil acceso y retorno al del sistema de ayuda	NA			0			
AY.3	Se ofrece ayuda contextual en tareas complejas	NA			0			
AY.4	FAQs (si las hay) correcta la elección como la redacción de las preguntas	NA			0			
AY.5	FAQs (si las hay) correcta la redacción de las respuestas	NA			0			

Instrucciones

1. Evalúe cada criterio y rellene la columna "Valor" según la leyenda

2. Añada sus comentarios o la referencia a los mismos en otro documento anexo en la columna "Comentarios"

Leyenda Valor

0 No se cumple en absoluto

10 Se cumple totalmente

NA Criterio no aplicable en el sitio

NTS No se cumple en todo el sitio

NEP

NPP

NPI

S

No se cumple en los enlaces principales

No se cumple en la página principal

No se cumple en alguna página interior

Se cumple el criterio

Figura anexo 25. Ayuda de la página.

Elaborado por: Elizabeth Flores y Diego Quisupangui.

Anexo 5. Pruebas con Jmeter

Para la ejecución de las pruebas realizaron en dos partes: 1. Pruebas de rendimiento;

2. Pruebas de estrés.

Tabla anexo 34. Parámetros de carga

Grupo de funcionalidad	# Carga inicial	Rendimiento	Estrés
Estudiante	100 usuarios	200 usuarios	600 usuarios
Docente	40 usuarios	120 usuarios	200 usuarios
CUTS	10 usuarios	40 usuarios	120 usuarios

Nota: Parámetros de carga pruebas Jmeter.

Elaborado por: Elizabeth Flores y Diego Quisupangui

Para un entendimiento de las variables de las gráficas generadas por las pruebas realizadas se tiene los siguientes campos: # **Usuario**: cantidad de usuarios de carga para la prueba; # **Muestra**: cantidad de hilos ejecutados para la página probada;

Media: tiempo promedio en milisegundos para un conjunto de resultados; **Mediana**:

valor en tiempo del percentil 50.; **Línea de 90%**: máximo tiempo utilizado por el 90%

de la muestra; **Línea de 95%**: máximo tiempo utilizado por el 95% de la muestra; **Línea de 99%**:

máximo tiempo utilizado por el 99% de la muestra; **Min**: el

tiempo más corto para las muestras; **Máx**: tiempo más largo para las muestras; **Error %**: porcentaje de soluciones con errores; **Rendimiento**: esta se puede medir en peticiones por segundo/minuto/hora; Kb/segundos: el rendimiento medido en kilobytes por segundo.

Pruebas de rendimiento

Las pruebas de rendimiento se realizaron a los perfiles Estudiante, Docente, CUTS, a continuación se presentan las configuraciones y resultados.

En la Figura anexo 26 se puede observar el ambiente de pruebas para el perfil Estudiante, contiene las páginas del sistema informático en donde va a realizar alguna acción.

Prueba de perfil Estudiante

GH-Estudiante

- Valores por Defecto para Petición HTTP
- Visualizador Spline
- Ver Resultados en Árbol
- Ver Árbol de Resultados
- Informe Agregado
- Gráfico de Resultados
- Gráfico
- Ver Resultados en Árbol
- Home
- Logeo
- Navegar
 - 36 /webutes9/faces/paginas/privadas/estudiante/index.xhtml
 - 38 /webutes9/faces/paginas/privadas/estudiante/index.xhtml
 - 39 /webutes9/faces/paginas/privadas/estudiante/presolicitud.xhtml
 - 53 /webutes9/primepush/convocatoriema
 - 55 /webutes9/faces/paginas/privadas/estudiante/presolicitud.xhtml
 - 56 /webutes9/faces/paginas/privadas/estudiante/fito.xhtml
 - 54 /webutes9/primepush/convocatoriema
 - 57 /webutes9/primepush/convocatoriema
 - 70 /webutes9/faces/paginas/privadas/estudiante/evolucion.xhtml
 - 82 /webutes9/faces/paginas/privadas/estudiante/index.xhtml
- Salir

Figura anexo 26. Configuración de prueba a perfil Estudiante
Elaborado por: Elizabeth Flores y Diego Quisupangui

Gráficos de pruebas 100 usuarios

Ver Resultados en Árbol

Nombre: GH-Estudiante en árbol

Comentarios:

Exportar todos los datos a Excel

Mostrar en árbol

Muestra #	Tiempo de conexión	Tiempo de Respuesta	Estado	Rendimiento	Latencia	Conexión (Promedio)
1	11.09.01.470	Estudiante 1.2	OK	24210	845	24210
2	11.09.01.470	Estudiante 1.2	OK	24210	845	24210
3	11.09.01.470	Estudiante 1.2	OK	24210	845	24210
4	11.09.01.470	Estudiante 1.2	OK	24210	845	24210
5	11.09.01.470	Estudiante 1.2	OK	24210	845	24210
6	11.09.01.470	Estudiante 1.2	OK	24210	845	24210
7	11.09.01.470	Estudiante 1.2	OK	24210	845	24210
8	11.09.01.470	Estudiante 1.2	OK	24210	845	24210
9	11.09.01.470	Estudiante 1.2	OK	24210	845	24210
10	11.09.01.470	Estudiante 1.2	OK	24210	845	24210
11	11.09.01.470	Estudiante 1.2	OK	24210	845	24210
12	11.09.01.470	Estudiante 1.2	OK	24210	845	24210
13	11.09.01.470	Estudiante 1.2	OK	24210	845	24210
14	11.09.01.470	Estudiante 1.2	OK	24210	845	24210
15	11.09.01.470	Estudiante 1.2	OK	24210	845	24210
16	11.09.01.470	Estudiante 1.2	OK	24210	845	24210
17	11.09.01.470	Estudiante 1.2	OK	24210	845	24210
18	11.09.01.470	Estudiante 1.2	OK	24210	845	24210
19	11.09.01.470	Estudiante 1.2	OK	24210	845	24210
20	11.09.01.470	Estudiante 1.2	OK	24210	845	24210
21	11.09.01.470	Estudiante 1.2	OK	24210	845	24210
22	11.09.01.470	Estudiante 1.2	OK	24210	845	24210
23	11.09.01.470	Estudiante 1.2	OK	24210	845	24210
24	11.09.01.470	Estudiante 1.2	OK	24210	845	24210
25	11.09.01.470	Estudiante 1.2	OK	24210	845	24210
26	11.09.01.470	Estudiante 1.2	OK	24210	845	24210
27	11.09.01.470	Estudiante 1.2	OK	24210	845	24210

Gráfico

Nombre: GH-Estudiante en árbol

Comentarios:

Exportar todos los datos a Excel

Muestra #	Rendimiento	Latencia	Conexión (Promedio)
1	24210	845	24210
2	24210	845	24210
3	24210	845	24210
4	24210	845	24210
5	24210	845	24210
6	24210	845	24210
7	24210	845	24210
8	24210	845	24210
9	24210	845	24210
10	24210	845	24210
11	24210	845	24210
12	24210	845	24210
13	24210	845	24210
14	24210	845	24210
15	24210	845	24210
16	24210	845	24210
17	24210	845	24210
18	24210	845	24210
19	24210	845	24210
20	24210	845	24210
21	24210	845	24210
22	24210	845	24210
23	24210	845	24210
24	24210	845	24210
25	24210	845	24210
26	24210	845	24210
27	24210	845	24210

Figura anexo 27. Resultado de prueba perfil Estudiante con 100 usuarios
Elaborado por: Elizabeth Flores y Diego Quisupangui

Gráficos de pruebas 200 usuarios



Figura anexo 28. Resultado de prueba perfil Estudiante con 200 usuarios

Elaborado por: Elizabeth Flores y Diego Quisupangui

En la Figura anexo 29 se puede observar el ambiente de pruebas para el perfil

Docente con las páginas asignadas.

Prueba de perfil Docente

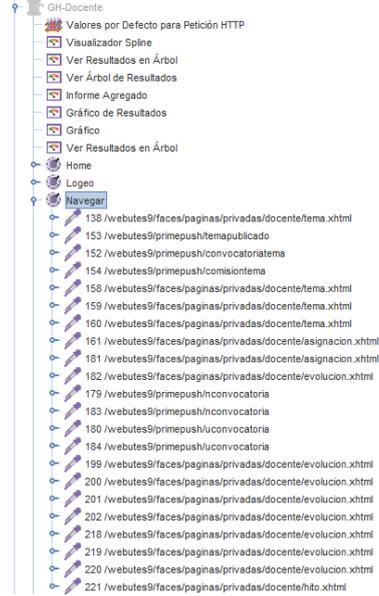
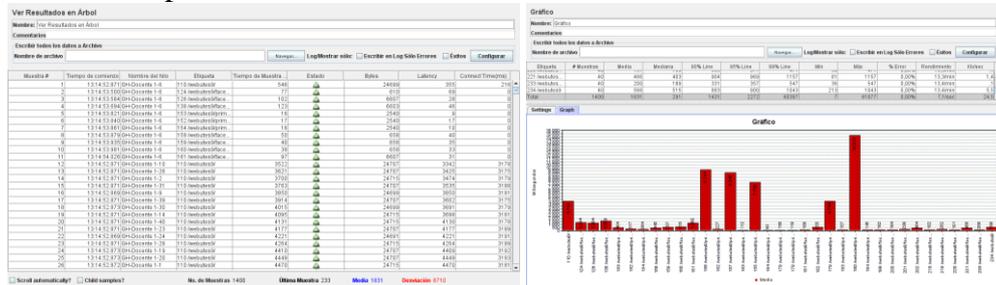


Figura anexo 29. Configuración de prueba perfil Docente

Elaborado por: Elizabeth Flores y Diego Quisupangui

Gráficos de pruebas 40 usuarios



Gráficos de pruebas 40 usuarios

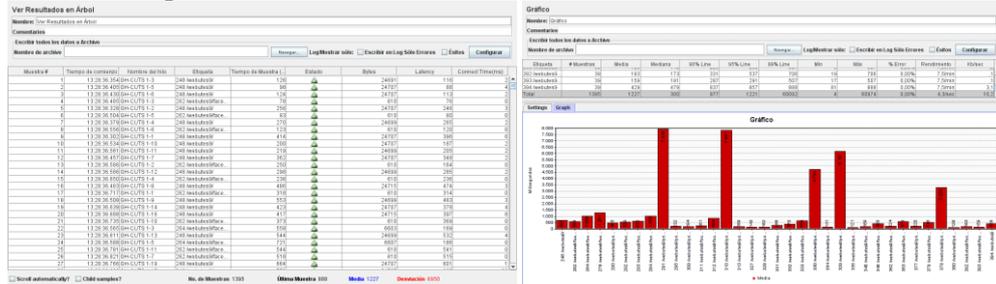
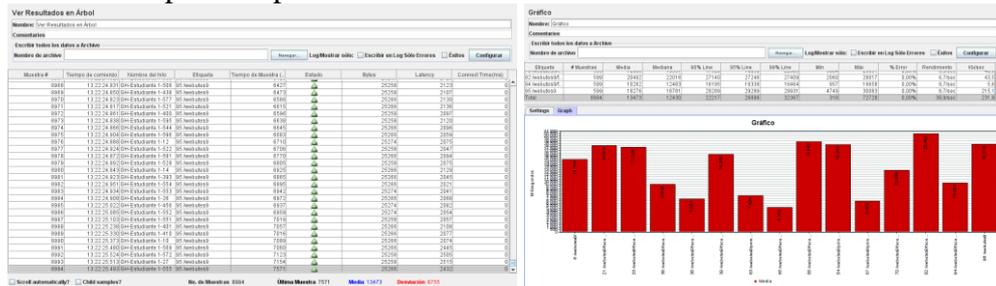


Figura anexo 34. Resultado de prueba perfil CUTS con 40 usuarios
Elaborado por: Elizabeth Flores y Diego Quisupangui

Pruebas de estrés

Para las pruebas de estrés se buscó que las pruebas sean hasta 4 veces más que el valor máximo de las pruebas de rendimiento, estas pruebas se realizaron de igual manera a los perfiles Estudiante, Docente y CUTS.

Gráficos de pruebas perfil Estudiante con 600 usuarios



Gráficos de pruebas perfil CUTS con 120 usuarios

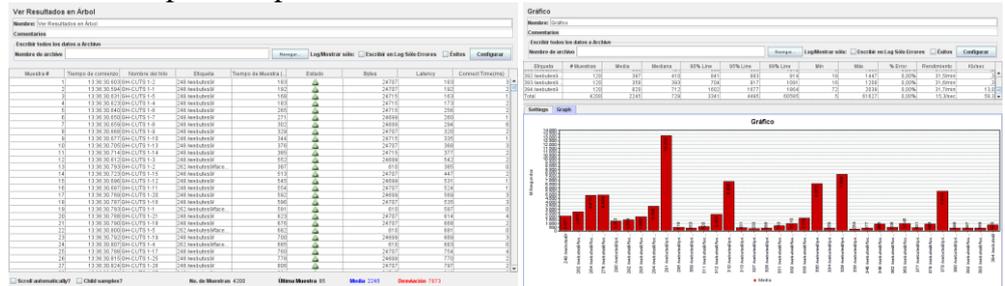


Figura anexo 37. Resultado de prueba perfil CUTS con 120 usuarios

Elaborado por: Elizabeth Flores y Diego Quisupangui