



UNIVERSIDAD TÉCNICA DE AMBATO

**FACULTAD DE INGENIERÍA EN SISTEMAS, ELECTRÓNICA E
INDUSTRIAL**

**CARRERA DE INGENIERÍA EN SISTEMAS COMPUTACIONALES E
INFORMÁTICOS**

Tema:

**SISTEMA WEB INTEGRADO PARA LA GESTIÓN DE TRÁMITES
ACADÉMICO ADMINISTRATIVOS EN LA FACULTAD DE INGENIERÍA
EN SISTEMAS, ELECTRÓNICA E INDUSTRIAL.**

Trabajo de Titulación Modalidad: Proyecto de Investigación, presentado previo a la obtención del título de Ingeniero en Sistemas Computacionales e Informáticos.

ÁREA: Desarrollo de software

LÍNEA DE INVESTIGACIÓN: Aplicación web

AUTOR: Guillermo Alexander Barcia Gallardo

TUTOR: Ing. Mg Clay Aldás Flores

Ambato – Ecuador

marzo - 2023

APROBACIÓN DEL TUTOR

En calidad de tutor del Trabajo de Titulación con el tema: SISTEMA WEB INTEGRADO PARA LA GESTIÓN DE TRÁMITES ACADÉMICO ADMINISTRATIVOS EN LA FACULTAD DE INGENIERÍA EN SISTEMAS, ELECTRÓNICA E INDUSTRIAL, desarrollado bajo la modalidad Proyecto de Investigación, por el señor Guillermo Alexander Barcia Gallardo, estudiante de la Carrera de Ingeniería en Sistemas Computacionales e Informáticos, de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, de la Universidad Técnica de Ambato, me permito indicar que el estudiante ha sido tutorado durante todo el desarrollo del trabajo hasta su conclusión, de acuerdo a lo dispuesto en el Artículo 15 del Reglamento para obtener el Título de Tercer Nivel, de Grado de la Universidad Técnica de Ambato, y el numeral 7.4 del respectivo instructivo.

Ambato, marzo 2023

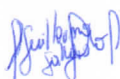
Ing. Mg Clay Fernando Aldás Flores

TUTOR

AUTORÍA

El presente Proyecto de Investigación titulado: SISTEMA WEB INTEGRADO PARA LA GESTIÓN DE TRÁMITES ACADÉMICO ADMINISTRATIVOS EN LA FACULTAD DE INGENIERÍA EN SISTEMAS, ELECTRÓNICA E INDUSTRIAL, es absolutamente original, auténtico y personal. En tal virtud, el contenido, efectos legales y académicos que se desprenden del mismo son de exclusiva responsabilidad del autor.

Ambato, marzo 2023



Guillermo Alexander Barcia Gallardo

C.C. 0804808335

AUTOR

DERECHOS DE AUTOR

Autorizo a la Universidad Técnica de Ambato, para que haga uso de este Trabajo de Titulación como un documento disponible para la lectura, consulta y procesos de investigación.

Cedo los derechos de mi Trabajo de Titulación en favor de la Universidad Técnica de Ambato, con fines de difusión pública. Además, autorizo su reproducción total o parcial dentro de las regulaciones de la Institución.

Ambato, marzo 2023



Guillermo Alexander Barcia Gallardo

C.C. 0804808335

AUTOR

APROBACIÓN DEL TRIBUNAL DE GRADO

En calidad de par calificador del Informe Final del Trabajo de Titulación presentado por el señor Guillermo Alexander Barcia Gallardo, estudiante de la Carrera de Ingeniería en Sistemas Computacionales e Informáticos, de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, bajo la Modalidad Proyecto de Investigación, titulado SISTEMA WEB INTEGRADO PARA LA GESTIÓN DE TRÁMITES ACADÉMICO ADMINISTRATIVOS EN LA FACULTAD DE INGENIERÍA EN SISTEMAS, ELECTRÓNICA E INDUSTRIAL, nos permitimos informar que el trabajo ha sido revisado y calificado de acuerdo al Artículo 17 del Reglamento para obtener el Título de Tercer Nivel, de Grado de la Universidad Técnica de Ambato, y al numeral 7.6 del respectivo instructivo. Para cuya constancia suscribimos, conjuntamente con la señora Presidente del Tribunal.

Ambato, marzo 2023

Ing. Pilar Urrutia, Mg.

PRESIDENTE DEL TRIBUNAL

Ing. Julio Balarezo, PhD.

PROFESOR CALIFICADOR

Ing. Rubén Nogales, Mg.

PROFESOR CALIFICADOR

DEDICATORIA

El presente trabajo de investigación lo dedico de manera especial a mis padres por ser mi pilar fundamental durante toda mi vida y mi educación, mi mamá Gloria quien siempre ha estado presente con palabras de aliento, mi papá Guillermo quien ha sido siempre mi apoyo incondicional, sin ellos nada de esto sería posible.

A mis tíos Carlos y Viviana por su apoyo durante toda mi vida, por siempre tenderme la mano cuando lo he necesitado.

A mi novia por su apoyo y motivación brindado durante todo el proceso de este trabajo.

Quiero dedicar también mi tesis a todos mis amigos de la universidad, de la vida y del trabajo, gracias por sus palabras de apoyo y aliento.

Guillermo Alexander Barcia Gallardo

AGRADECIMIENTO

A Dios por guiarme en cada paso de mi vida, darme su bendición y su protección para enfrentarme y luchar ante los obstáculos.

A mi familia por ser ese apoyo incondicional, por enseñarme a luchar por mis sueños, por siempre tener un abrazo o una palabra de aliento para motivarme incluso en los días más difíciles.

A la Universidad Técnica de Ambato y a la Facultad de Ingeniería en Sistemas, Electrónica e Industrial en la cual tuve la oportunidad de formarme en un profesional.

A los docentes de la Carrera de Ingeniería en Sistemas Computacionales e Informáticos por su aporte profesional durante mi formación académica.

A mi tutor Mg. Ing. Clay Fernando Aldas Flores quien ha sido una guía y asesor para culminar con éxito mi trabajo de titulación.

Guillermo Alexander Barcia Gallardo

ÍNDICE GENERAL DE CONTENIDOS

APROBACIÓN DEL TUTOR	ii
AUTORÍA	iii
DERECHOS DE AUTOR	iv
APROBACIÓN TRIBUNAL DE GRADO	v
DEDICATORIA	vi
AGRADECIMIENTO	vii
ÍNDICE GENERAL DE CONTENIDOS	viii
ÍNDICE DE FIGURAS	x
ÍNDICE DE TABLAS	xi
ÍNDICE DE ANEXOS	xvi
RESUMEN EJECUTIVO	xvii
ABSTRACT	xviii
INTRODUCCIÓN	xix
CAPÍTULO I.- MARCO TEÓRICO	1
1.1 Tema de investigación	1
1.2 Antecedentes investigativos	1
1.2.1 Contextualización del problema	2
1.2.2 Fundamentación teórica	4
1.3 Objetivos	7
1.3.1 General	7
1.3.2 Específicos	7
CAPÍTULO II.- METODOLOGÍA	8
2.1 Materiales	8
2.2 Población y muestra	10
2.2.1 Recolección de información	10
CAPÍTULO III.- RESULTADOS Y DISCUSIÓN	16
3.1 Desarrollo de la propuesta	16
3.1.1 Análisis de los procesos administrativos que se realizan en la FISEI para su automatización.	16

3.2 Metodología de desarrollo	17
3.2.1 Comparación de metodologías para el desarrollo del sistema	17
3.2.2 Metodología seleccionada.....	26
3.2.3 Análisis de las herramientas de desarrollo	26
3.2.4 Implantación del sistema web integrado para la gestión de trámites académico- administrativos en la Facultad de Ingeniería en Sistemas, Electrónica e Industrial	40
CAPÍTULO IV.- CONCLUSIONES Y RECOMENDACIONES.....	112
4.1 Conclusiones	112
4.2 Recomendaciones	113
BIBLIOGRAFÍA.....	114
ANEXOS	119

ÍNDICE DE FIGURAS

Figura 1.1.- Capaz Framework.....	5
Figura 1.2.- Logo Laravel.	5
Figura 1.3.- Logo reactJs.....	6
Figura 3.4.- Estructura organizacional de la FISEI de la UTA.	16
Figura 3.5.- Proceso de Consejo Directivo de Facultad (Áreas Internas/ Externas)..	18
Figura 3.6.- Proceso de Consejo Académico de Facultad (Estudiantes).....	19
Figura 3.7.- Proceso de Consejo Académico de Facultad.....	20
Figura 3.8.- Proceso de Unidad de Titulación e Integración Curricular	21
Figura 3.9.- Clasificación de proyectos por colores.....	24
Figura 3.10.- Web más utilizados entre los desarrolladores a nivel mundial.	28
Figura 3.11.- Estructura del proyecto web.	39
Figura 3.12.- Estructura del proyecto web.....	40
Figura 3.13.- Arquitectura LAMP + ExpressJs.....	76
Figura 3.14.- Diseño de interfaz.....	77
Figura 3.15.- Formulario de ingreso al sistema.....	78
Figura 3.16.- Agregar estudiantes.	78
Figura 3.17.- Agregar múltiples estudiantes.	79
Figura 3.18.- Crear docente.....	79
Figura 3.19.- Cargos.....	80
Figura 3.20.- Crear cargos.....	80
Figura 3.21.- Actualizar cargo.	80
Figura 3.22.- Añadir y listar estudiantes.	81
Figura 3.23.- Añadir y listar docentes.....	81
Figura 3.24.- Visualizar carreras universitarias.	82
Figura 3.25.- Editar carreras universitarias.	82
Figura 3.26.- Opción de perfil.....	83
Figura 3.27.- Opción para cambiar la contraseña.	83
Figura 3.28.- Crear nuevo usuario.....	83
Figura 3.29.- Usuarios del sistema.....	84
Figura 3.30.- Actualizar usuario.....	84
Figura 3.31.- Visualizar procesos académicos.....	85
Figura 3.32.- Añadir procesos académicos.	85
Figura 3.33.- Editar procesos.	86
Figura 3.34.- Visualizar plantilla.	86
Figura 3.35.- Añadir plantillas.	86
Figura 3.36.- Actualizar plantillas.....	87
Figura 3.37.- Plantilla de documentos para reemplazar con datos requeridos.....	88
Figura 3.38.- Ejemplo de plantilla utilizando las etiquetas de {{FROM}}.....	89

Figura 3.39.- Ejemplo de plantilla utilizando las etiquetas de {{TO}}.	89
Figura 3.40.- Visualización de consejos.	90
Figura 3.41.- Acciones.	91
Figura 3.42.- Crear consejos.	91
Figura 3.43.- Modificar consejos.	91
Figura 3.44.- Modificar consejos.	92
Figura 3.45.- Eliminar consejos.	92
Figura 3.46.- Asistencia.	92
Figura 3.47.- Reservar numeración.	93
Figura 3.48.- Ver números reservados y encolados.	93
Figura 3.49.- Generar documento.	94
Figura 3.50.- Generación del documento con los datos de las etiquetas.	94
Figura 3.51.- Editar documentos-resolución o acuerdo.	95
Figura 3.52.- Edición del documento en Google Docs.	95
Figura 3.53.- Creación de documentos y notificaciones.	96
Figura 3.54.- Consulta de resoluciones.	96
Figura 3.55.- Plantilla separador por actas.	97
Figura 3.56.- Plantilla acta.	97
Figura 3.57.- Generación de acta.	98
Figura 3.58.- Cerrar consejos.	98

ÍNDICE DE TABLAS

Tabla 2.1. Matriz de preguntas para entrevista No 1 (Parte 1).	8
Tabla 2.2. Matriz de preguntas para entrevista No 1 (Parte 2)	9
Tabla 2.3. Muestra.	10
Tabla 2.4. Recolección de información.	11
Tabla 2.5. Recolección información-entrevista #1 (Parte 1).	11
Tabla 2.6.- Recolección información-entrevista #2 (Parte 1).	12
Tabla 2.7.-Recolección información-entrevista #3.	14
Tabla 2.8.-Procesamiento de datos de los resultados obtenidos de las entrevistas. ...	15
Tabla 3.9.-Comparación de metodologías.	26
Tabla 3.10.- Cuadro comparativo entre ReactJs, VueJs y Blazor.	31
Tabla 3.11.- Cuadro comparativo entre Laravel, ASP.NET Core y Node.js.	36
Tabla 3.12.-Modelo historia de usuario.	40
Tabla 3.13.-Descripción de la historia de usuario.	41

Tabla 3.14.-Historia de usuario 1.....	41
Tabla 3.15.-Historia de usuario 2.....	42
Tabla 3.16.-Historia de usuario 3.....	42
Tabla 3.17.-Historia de usuario 4.....	42
Tabla 3.18.-Historia de usuario 5.....	43
Tabla 3.19.-Historia de usuario 6.....	43
Tabla 3.20.-Historia de usuario 7.....	43
Tabla 3.21.-Historia de usuario 8.....	44
Tabla 3.22.-Historia de usuario 9.....	44
Tabla 3.23.-Historia de usuario 10.....	44
Tabla 3.24.-Historia de usuario 11.....	45
Tabla 3.25.-Historia de usuario 12.....	45
Tabla 3.26.-Historia de usuario 13.....	46
Tabla 3.27.-Historia de usuario 14.....	46
Tabla 3.28.- Historia de usuario 15.....	46
Tabla 3.29.- Historia de usuario 16.....	47
Tabla 3.30.-Historia de usuario 17.....	47
Tabla 3.31.- Historia de usuario 18.....	47
Tabla 3.32.-Historia de usuario 19.....	48
Tabla 3.33.-Historia de usuario 20.....	48
Tabla 3.34.-Historia de usuario 21.....	49
Tabla 3.35.- Historia de usuario 22.....	49
Tabla 3.36.- Historia de usuario 23.....	49
Tabla 3.37.-Historia de usuario 24.....	50
Tabla 3.38.-Historia de usuario 25.....	50
Tabla 3.39.- Historia de usuario 26.....	51
Tabla 3.40.-Historia de usuario 27.....	51
Tabla 3.41.-Historia de usuario 28.....	52
Tabla 3.42.-Historia de usuario 29.....	52
Tabla 3.43.-Historia de usuario 30.....	53
Tabla 3.44.- Estimación de historias de usuario.	54
Tabla 3.45.-Plan de entregas.	55
Tabla 3.46.-Equipo y roles.	55

Tabla 3.47.- Tarea-determinar la estructura del proyecto.....	56
Tabla 3.48.- Tarea- definición y modelado de la base de datos.....	56
Tabla 3.49.-Tarea- definir plantilla para el diseño de las interfaces del sistema.	56
Tabla 3.50.-Tarea- diseño de menú principal.....	57
Tabla 3.51.-Tarea-diseño de formulario de ingreso.....	57
Tabla 3.52.- Tarea- codificación para validar las credenciales de ingreso.....	57
Tabla 3.53.-Tarea- codificación agregar estudiantes por archivo Excel.....	58
Tabla 3.54.-Tarea- codificación agregar estudiantes manualmente.....	58
Tabla 3.55.-Tarea- buscador por información del estudiante.....	58
Tabla 3.56.-Tarea- codificación agregar docentes manualmente.....	59
Tabla 3.57.- Tarea- buscador por información del docente.....	59
Tabla 3.58.- Tarea- presentación de información detallada de los estudiantes.....	59
Tabla 3.59.- Tarea- presentación de información detallada de los docentes.....	60
Tabla 3.60.- Tarea- diseño de interfaz de visualización de carreras universitarias....	60
Tabla 3.61.- Tarea- buscador por nombre de la carrera universitaria.....	60
Tabla 3.62.- Tarea-diseño de interfaz para crear, editar e inhabilitar carrera universitaria.....	61
Tabla 3.63.- Tarea- diseño de interfaz para poder ver los datos del usuario y cambiar la contraseña.....	61
Tabla 3.64.- Tarea- diseño de interfaz de perfil.....	61
Tabla 3.65.- Tarea- implementación del paquete spatie/laravel-permission.....	62
Tabla 3.66.- Tarea- formulario funcional de usuarios.....	62
Tabla 3.67.- Tarea- diseño de interfaz de visualización de procesos académicos. ...	62
Tabla 3.68.- Tarea- codificación para crear los procesos académicos.....	63
Tabla 3.69.- Tarea- Buscador por nombre del proceso académico.....	63
Tabla 3.70.- Tarea- codificación para editar los procesos académicos.....	63
Tabla 3.71.- Tarea- diseño de interfaz de visualización de plantillas.....	64
Tabla 3.72.- Tarea- codificación para crear las plantillas.....	64
Tabla 3.73.- Tarea- buscador por nombre de la plantilla.....	64
Tabla 3.74.- Tarea- codificación para editar las plantillas.....	65
Tabla 3.75.- Tarea- realizar un formulario en donde se refleje el archivo Word.....	65
Tabla 3.76.-Tarea- diseño de interfaz de visualización de consejo directivo de facultad.....	66

Tabla 3.77.-Tarea- diseño de interfaz de visualización de consejo académico.	66
Tabla 3.78.- Tarea- diseño de interfaz de visualización de consejo titulación.	66
Tabla 3.79.- Tarea-diseño de interfaz de visualización de consejo integración curricular.	67
Tabla 3.80.-Tarea- diseño de interfaz de asistencia.	67
Tabla 3.81.- Tarea-diseño de interfaz de agregar miembros.....	67
Tabla 3.82.- Tarea- codificación para crear los consejos directivos de facultad.	68
Tabla 3.83.- Tarea- codificación para crear los consejos académicos.	68
Tabla 3.84.- Tarea- codificación para crear los consejos de titulación.	68
Tabla 3.85.- Tarea- codificación para crear los consejos de integración curricular.	69
Tabla 3.86.- Tarea- codificación para modificar los consejos directivos de facultad	69
Tabla 3.87.- Tarea- codificación para modificar los consejos académicos.....	69
Tabla 3.88.- Tarea- codificación para crear los consejos de titulación.....	70
Tabla 3.89.- Tarea- codificación para modificar los consejos de integración curricular.	70
Tabla 3.90.- Tarea- desarrollo de funcionalidad para eliminar consejo directivo de facultad.....	70
Tabla 3.91.- Tarea- desarrollo de funcionalidad para eliminar consejo académico... ..	71
Tabla 3.92.- Tarea- desarrollo de funcionalidad para eliminar consejo de titulación.....	71
Tabla 3.93.- Tarea- desarrollo de funcionalidad para eliminar consejo de integración curricular.	71
Tabla 3.94.- Tarea- desarrollo de funcionalidad y de interfaz para designar los miembros.....	72
Tabla 3.95.- Tarea- diseño de la interfaz de números reservados.....	72
Tabla 3.96.- Tarea- diseño de interfaz para formulario documentos.	72
Tabla 3.97.- Tarea- desarrollo de funcionalidad para ver números reservados y encolados.....	73
Tabla 3.98.- Tarea- desarrollo de funcionalidad para guardar documentos.....	73
Tabla 3.99.- Tarea-desarrollo de funcionalidad para editar documentos.....	73
Tabla 3.100.- Tarea- notificar los trámites académicos de los estudiantes.....	74
Tabla 3.101.- Tarea- desarrollo de funcionalidad para procesar documentos.	74
Tabla 3.102.- Tarea- desarrollo de funcionalidad para generar plantilla acta.....	75
Tabla 3.103.- Tarea- desarrollo de funcionalidad para cerrar consejos.	75

Tabla 3.104.- Resumen de historias de usuario de la iteración 1.....	75
Tabla 3.105.- Resumen de historias de usuario de la iteración 2.....	77
Tabla 3.106.- Resumen de historias de usuario de la iteración 3.....	85
Tabla 3.107.- Resumen de historias de usuario de la iteración 4.....	90
Tabla 3.108.- Prueba de aceptación - formulario de ingreso al sistema web integrado.	99
Tabla 3.109.- Prueba de aceptación - formulario estudiantes.....	100
Tabla 3.114.- Prueba de aceptación - formulario docentes.....	100
Tabla 3.115.- Prueba de aceptación – información estudiantes.....	101
Tabla 3.116.- Prueba de aceptación - información docentes.	101
Tabla 3.117.- Prueba de aceptación - visualizar carreras universitarias.	102
Tabla 3.118.- Prueba de aceptación - formulario carreras universitarias.....	102
Tabla 3.119.- Prueba de aceptación - gestión de perfil de usuario.	103
Tabla 3.120.- Prueba de aceptación - gestión de usuarios y roles.....	103
Tabla 3.121.- Prueba de aceptación - visualizar procesos académicos.....	104
Tabla 3.122.- Prueba de aceptación – formulario de procesos académicos.....	104
Tabla 3.123.- Prueba de aceptación - editar procesos académicos.	104
Tabla 3.124.- Prueba de aceptación - visualización de plantillas.	105
Tabla 3.125.- Prueba de aceptación - crear plantillas.	105
Tabla 3.126.- Prueba de aceptación - crear plantillas.	106
Tabla 3.127.- Prueba de aceptación - formulario de plantilla de Word.	106
Tabla 3.128.- Prueba de aceptación - visualización de consejos.	107
Tabla 3.129.- Prueba de aceptación - crear consejos.	107
Tabla 3.130.- Prueba de aceptación - modificar consejos.....	107
Tabla 3.131.- Prueba de aceptación - eliminar consejos.....	108
Tabla 3.132.- Prueba de aceptación - formulario de miembros o representantes. ...	108
Tabla 3.133.- Prueba de aceptación - formulario de números reservados de documentación.	109
Tabla 3.134.- Prueba de aceptación - crear documentos.....	109
Tabla 3.135.- Prueba de aceptación - editar documentos.....	110
Tabla 3.136.- Prueba de aceptación – notificación de trámites académicos.....	110
Tabla 3.137.- Prueba de aceptación – generación de actas de Consejo.	111
Tabla 3.138.- Prueba de aceptación – cerrar consejos.	111

ÍNDICE DE ANEXOS

Anexo 1. Base de datos.	119
Anexo 2. Configuración de API de Google.	120
Anexo 3. Codificación.	125

RESUMEN EJECUTIVO

El presente proyecto de investigación surge por la necesidad de mejorar la gestión de la documentación cumpliendo con las necesidades y expectativas de los usuarios, por lo tanto, se propone implementar un sistema web integrado para la gestión de trámites académico-administrativos en la Facultad de Ingeniería en Sistemas, Electrónica e Industrial (FISEI).

La metodología utilizada para el desarrollo del proyecto de investigación es la metodología Extreme Programming (XP), la cual consta de cuatro fases. La fase de exploración identifica las necesidades del cliente, la fase de planificación es la que estima el esfuerzo, la fase de iteraciones se encarga de crear la solución y la fase de pruebas entrega el producto final al cliente. Se utilizó el patrón de arquitectura en capas, en el cual el cliente está realizado con la tecnología React Js y el servidor con Laravel. También se trabaja con la base de datos MySQL y la API de Google Drive.

El resultado obtenido es un sistema web que integra las áreas de Unidad de Titulación e Integración Curricular, Consejo Académico y Secretaría General de Facultad, con el propósito de generar trámites de una forma eficiente y segura.

La implementación del sistema es de gran ayuda para el manejo de los documentos que se realizan en las áreas anteriormente indicadas, puesto que el sistema permite visualizar el estado en que se encuentran los documentos, promoviendo una mejora en la organización individual y colectiva de las personas que trabajan en dichas áreas, además de optimizar el tiempo y los recursos de los estudiantes o docentes en los trámites requeridos.

Palabras clave: Sistema web integrado, Back End, Laravel, Front End, ReactJS.

ABSTRACT

This research project arises from the need to improve the management of documentation meeting the needs and expectations of customers; therefore, it is proposed to implement an integrated web system for the management of academic-administrative procedures in the Faculty of Systems Engineering, Electronics and Industrial (FISEI).

The methodology used for the development of the research project is the Extreme Programming (XP) methodology, which consists of four phases. The exploration phase identifies the customer's needs, the planning phase estimates the effort, the iterations phase is responsible for creating the solution, and the testing phase delivers the final product to the customer. The layered architecture pattern was used, in which the client is made with React Js technology and the server with Laravel. It also works with the MySQL database and the Google Drive API.

The result obtained is a web system that integrates the areas of Degree and Curricular Integration Unit, Academic Council of the Faculty and General Secretariat, with the purpose of generating procedures in an efficient and safe way.

The implementation of the system is of great help for the management of the documents that are carried out in the Faculty, since the system allows to visualize the state in which the documents are, promoting an improvement in the individual and collective organization of the people who work in these areas, in addition to optimizing the time and resources of the students or teachers in the required procedures.

Keywords: Integrated web system, Back End, Laravel, Front End, ReactJS.

INTRODUCCIÓN

Hasta el 2022 se aprecia que la innovación a nivel mundial está en constante movimiento y evolución. Dicha apreciación está dada porque se evidencia que cada día existen avances que buscan optimizar los recursos. Estos recursos se reflejan en diferentes ámbitos de la tecnología, según las necesidades y de los objetivos que se deseen cumplir [1].

Al implementar un sistema basado en tecnologías web, se obtiene una aplicación donde se puede acceder a través del protocolo HTTP (Hypertext Transfer Protocol), el cual funciona en combinación con una variedad de funciones de desarrollo web para brindar a los usuarios y clientes un entorno confiable y seguro. La función principal del protocolo HTTP es controlar y gestionar el flujo de datos [2].

Dentro de cualquier institución educativa la gestión de trámites académicos es fundamental, debido a que es necesario llevar una planificación, organización, dirección y control, los cuales se fundamentan en una sucesión de actividades con el fin de garantizar el cumplimiento de las metas dentro de las universidades o instituciones educativas [3].

Las Universidades en el Ecuador aún tienen varios obstáculos para llevar a cabo sus actividades académicas por lo que el uso de la tecnología es parte fundamental para la optimización del tiempo y de los recursos, donde el uso de tecnologías web ayudarían en la gestión de trámites, logrando eficiencia y efectividad en el procesamiento de datos [4].

La FISEI de la Universidad Técnica de Ambato (UTA) tiene como prioridad cumplir con las necesidades y expectativas de los estudiantes con el fin de alcanzar una cultura de calidad. Por lo tanto, el presente trabajo de investigación se basará en la creación de un sistema web integrado para mejorar la gestión de trámites académicos.

En este documento se presentará la siguiente estructura: Capítulo I Marco Teórico, el cual consta de la parte fundamental de la investigación, identificando las fuentes primarias y secundarias en donde se sustenta la investigación, Capítulo II

Metodología, es donde se aplica diferentes procedimientos para alcanzar los objetivos que rige la investigación, Capítulo III Resultados y Discusión, se describe los principales hallazgos e interpretación de la investigación y por último en el Capítulo IV Conclusiones y Recomendaciones, es la parte final de la investigación y es donde se evidencia el cumplimiento de los objetivos.

CAPÍTULO I.- MARCO TEÓRICO

1.1 Tema de investigación

SISTEMA WEB INTEGRADO PARA LA GESTIÓN DE TRÁMITES ACADÉMICO ADMINISTRATIVOS EN LA FACULTAD DE INGENIERÍA EN SISTEMAS, ELECTRÓNICA E INDUSTRIAL.

1.2 Antecedentes investigativos

Revisando la información existente en la biblioteca de la FISEI de la UTA, se ha encontrado trabajos que servirán de apoyo en el trabajo de investigación:

Se ha creado una aplicación web para la automatización en la gestión de trámites en la secretaría de la FISEI para agilizar los procesos utilizando arquitectura LAMP (Linux-Apache-MySQL-PHP), el cual incluye el sistema operativo Linux, servidor web Apache, base de datos MySQL (Structured Query Language), además del lenguaje de programación PHP (Hypertext Preprocessor). La autora del proyecto indica que el sistema le permite al usuario generar y almacenar los trámites académicos, generar actas de Consejo Directivo, además, la consulta de trámites realizados por los estudiantes incluso permite la creación, edición y eliminación de los procesos para los trámites posteriores [5]. Por lo tanto, se puede concluir que mediante la automatización de procesos administrativos se puede mejorar los tiempos de respuesta del usuario, logrando gestionar de manera organizada los trámites académicos. Así mismo la arquitectura LAMP permitirá desarrollar el sistema web integrado con un enfoque abierto para el desarrollo de back-end.

La automatización de procesos académicos en el ámbito universitario toma un papel central en un contexto organizado, siendo necesario implementar sistemas informáticos para un mejor desempeño y crecimiento, para lo cual se aplicó la metodología AUP (Agile Unified Process), identificando que es efectiva para el desarrollo ágil de aplicativos Web, ya que cuenta de forma directa la interacción del equipo de desarrollo con el cliente en cada una de las fases, logrando obtener un producto final que cumpla con las especificaciones requeridas [6]. Por lo tanto, se concluye que la metodología AUP permitirá describir de una manera simple y fácil

de entender la forma de desarrollar aplicaciones de software de negocio, aplicando las técnicas ágiles de un modelado ágil.

Al implementar un sistema Workflow se obtienen varios beneficios como:

- Contar con indicadores que permitan medir el cumplimiento de objetivos propuestos.
- Identificar mejoras en la empresa al tener un entendimiento común de la información en todas las áreas [7].
- Gracias a la aplicación de un enfoque por procesos se identificó de manera ágil y sencilla los problemas y la rápida solución dentro del proceso en estudio, logrando mejorar la gestión en un menor tiempo.

A través de este trabajo se concluye que, mediante un sistema que gestione información se puede llevar un control de cumplimiento de objetivos, además de tener más eficiencia y eficacia en los procesos que se deben llevar a cabo.

Gracias a la creación de un sistema informático, se observa que existe un mejor control y dominio de la información, logrando llevar a cabo una centralización de la información en la cual un sitio web engloba de manera general la información de la carrera, procesos de trabajo de graduación, creación de perfiles docentes, entre otras; por la descripción antes mencionada, este trabajo es una muestra de innovación que cubre las necesidades y principales para solventar la mayoría de inquietudes a personas que deseen estudiar la carrera de Ingeniería de Sistemas Informáticos o que necesitan información siendo alumnos” [8]. Con este trabajo se concluye que, mediante los sistemas informáticos se pueden generar soluciones efectivas para un mejor control y manejo de información. Sin embargo, le hace falta muchas mejoras las cuales el sistema que se va a desarrollar va a incluir nuevas funcionalidades tales como: subida de archivos Excel de los estudiantes para hacer el respectivo guardado en la base de datos.

1.2.1 Contextualización del problema

Las tendencias tecnológicas aportan significativamente al rendimiento de los recursos informáticos, el desarrollo de software con el pasar del tiempo ha tomado importancia dentro de las instituciones promoviendo la utilización de sistemas

informáticos que mejoran la calidad de sus procesos. Tanto instituciones públicas como privadas realizan procesos que involucran información y documentación, por lo tanto, es necesario llevar una adecuada gestión administrativa y financiera de los procesos que se realizan todos los días [9].

La tecnología mejora los procesos y facilita el trabajo de las personas, actualmente la secretaría general de la FISEI, cuenta con un sistema web para la gestión de trámites académicos, no obstante, este sistema tiene aspectos cuya funcionalidad y diseño de interfaces deben ser mejorados debido a que presentan algunas falencias.

Los problemas en el sistema en cuanto a funcionalidad se presentan al momento de realizar el proceso de guardar la ficha de los estudiantes, este proceso se lo realiza cargando un archivo Excel que tiene los datos principales de cada uno de los estudiantes que se matriculan en un periodo académico; al momento de subir el archivo en Excel si este tiene campos vacíos o espacios en blanco el sistema no reconoce los datos y no permite realizar una búsqueda o filtrar por datos, entonces el usuario debe antes corregir manualmente, esto provoca pérdida de tiempo y le quita funcionalidad al sistema.

El proceso de generación de los números de las actas no se encuentra automatizado, este se lo realiza manualmente quitándole así eficacia al sistema, además los tiempos de respuesta al generar las actas son demasiado largos, esto debido a la presencia de latencia puesto que el código en el cual se desarrolló el sistema de la secretaría general de la FISEI le hacen falta algunas mejoras.

Por otra parte, las interfaces de usuario no son amigables tampoco son intuitivas, tal es el caso, de la ventana principal en donde se reflejan todos los consejos que han sido creados en diferentes fechas, en este menú existe mucho desglose de información sin una paginación correcta, lo que provoca un cansancio en el usuario.

Además, no existen buscadores óptimos que permitan realizar filtros de diferentes campos, tales como: buscar por número de solicitud, apellidos del estudiante, cédula, entre otros, que permitan encontrar alguna solicitud o plantilla de algún formato de un estudiante, de manera más rápida, eficiente y eficaz.

Hay que destacar que existe mucha información que no se encuentra segmentada, por lo que, se requiere reestructurar la base de datos, en vista de que existe inconsistencias en la información que se guarda por la duplicidad de los datos.

Por estas razones se hace necesario mejorar el sistema inicial corrigiendo los errores e inconvenientes que presenta. Además, complementarlo con funcionalidades adicionales que abarquen otros aspectos de la gestión administrativa y la fortalezcan, simplificándose su tramitación, evitando duplicidades y unificando procesos, entre otros beneficios. Por otra parte, se tiene la integración del sistema con las áreas de titulación y consejo académico de Facultad.

1.2.2 Fundamentación teórica

Administración de procesos: Las empresas modernas basan sus operaciones en procesos, estos definen las actividades que deben cumplirse para obtener servicios o productos de calidad que les permita competir con efectividad en los diferentes mercados. Además, permite a las gerencias de las empresas medir la capacidad del negocio y poder plantearse objetivos. La administración por procesos pertenece a la política de mejora continua de la empresa para que pueda crecer organizacionalmente [10].

Sistemas de gestión: Es una estructura probada para la gestión y mejora continua de las políticas, los procedimientos y procesos de la organización. Estos sistemas ayudan a las organizaciones a alcanzar sus objetivos a través de estrategias tales como la optimización de procesos, enfoques centrados en la gestión y el pensamiento disciplinado. [11]

Aplicaciones web: Una aplicación web es un Sistema informático donde grandes cantidades de datos estructurados son consultados, procesados y analizados mediante navegadores, los cuales incluyen un alto grado de interacción con los usuarios. Además, las aplicaciones web, se deben presentar como funcionales, actualizables, escalables y seguras, el diseño de la interfaz debe ser claro, simple y estructurado [11].

Framework

El Framework es una estructura software compuesta de componentes personalizables e intercambiables para el desarrollo de una aplicación en otras palabras se considera

como una aplicación genérica incompleta y configurable con la que se puede construir una aplicación concreta [5].

Por otra parte, permite agregar funcionalidad extendida a un lenguaje de programación, porque automatiza muchos de los patrones de programación para orientarlos a un determinado propósito, mejorando la estructura del código y haciéndolo más entendible y de cierta forma sostenible que permite separar en capas la aplicación. En general, divide la aplicación en tres capas:

- La lógica de presentación que administra las interacciones entre el usuario y el software.
- La lógica de datos que permite el acceso a un agente de almacenamiento persistente u otros.
- La lógica de dominio o de negocio, que manipula los modelos de datos de acuerdo con los comandos recibidos desde la presentación [12].

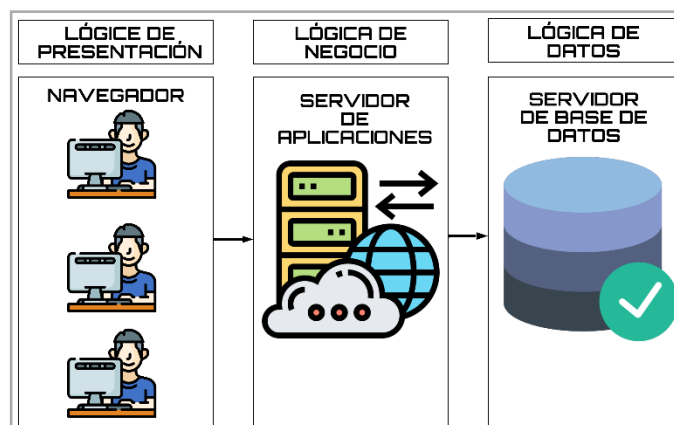


Figura 1.1.- Capaz Framework.
Elaborado por: El investigador.

Laravel



Figura 1.2.- Logo Laravel.

Fuente: Universidad Técnica del Norte [13].

El lenguaje de programación más utilizado en el diseño web es PHP (Hypertext Pre-Processor) debido a su confiabilidad, costo efectivo y fácil navegación que puede

crear sitios web altamente interactivos [8, 9]. Uno de los mejores y superiores marcos para PHP en comparación con otros marcos es Laravel [9]. Laravel está diseñado para mejorar la calidad del software, simplificar la autenticación, facilitar el enrutamiento, facilitar el acceso y aumentar el poder dentro del marco del sitio web. Laravel es una aplicación de marco con una sintaxis elegante y tiene amplias funciones como seguridad, almacenamiento de contraseñas, recordatorios y restablecimiento de contraseñas, cifrado y validación [10]. Según una encuesta realizada en marzo de 2015 sobre la popularidad de PHP framework, Laravel fue el framework más popular, seguido consecutivamente por Symfony2, Nette, CodeIgniter, Yii2 y otros [11].

React js

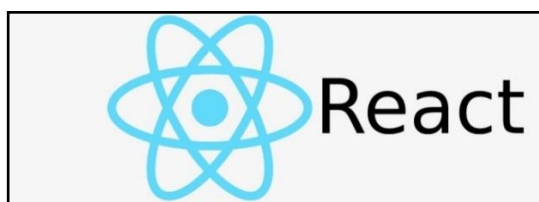


Figura 1.3.- Logo reactJs.

Fuente: Universidad Técnica del Norte [13].

React.js o ReactJS es una biblioteca de JavaScript de código abierto, que fue desarrollado por la empresa de Facebook, sirve especialmente para crear interfaces de usuario orientado a componente para el desarrollo de aplicaciones de una sola página o, en otros términos, conocido como simple page applications. Una de las ventajas de React, es que ayuda a los desarrolladores a construir aplicaciones que usan datos que combinan todo el tiempo, dado que, su objetivo es ser sencillo, declarativo y fácil de combinar [13]. Mediante esta biblioteca, se permite estructurar de mejor manera las interfaces de la página web o sistema web, ya que, al utilizar aplicaciones de una sola página, me permite tener mayor fluidez y mejorar de la experiencia de usuario, aparte de que es muy ventajoso para los servidores, porque tienen que soportar menos la carga al ejecutarse mayoría de la lógica en el lado del cliente, por lo que, es una tecnología muy usada en la actualidad.

1.3 Objetivos

1.3.1 General

Implementar un sistema web integrado para la gestión de trámites académico administrativos en la Facultad de Ingeniería en Sistemas, Electrónica e Industrial.

1.3.2 Específicos

- Analizar los procesos administrativos que se realizan en la Facultad de Ingeniería en Sistemas, Electrónica e Industrial para su debida automatización.
- Seleccionar las herramientas adecuadas para el desarrollo de la aplicación propuesta.
- Implantar el sistema web integrado para la gestión de trámites académico-administrativos en la Facultad de Ingeniería en Sistemas, Electrónica e Industrial.

CAPÍTULO II.- METODOLOGÍA

2.1 Materiales

En el presente proyecto de investigación se aplicó tres entrevistas para la recolección de la información, en base a una serie de preguntas que permite determinar los requerimientos funcionales de la aplicación.

Una de las entrevistas fue realizada a la Ab. Daniela Montenegro, que es la persona encargada de la secretaria general de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, quien da a conocer todos los procesos académicos que administra en su puesto de trabajo. Teniendo en cuenta, que todos los tramites que ingresan en la Facultad, pasan por esta área que ella administra, por lo que es una de las personas principales con las que se va a trabajar para realizar el sistema.

La segunda entrevista fue realizada a las secretarias encargadas de diferentes áreas en la Facultad, que en este caso son dos personas: Ab. Elena Arcos encargada del área de Titulación e Integración Curricular. La Ing. Myriam Viteri encargada de Consejo Académico de Facultad, las cuales se encargan de manejar tramites académicos que posteriormente son entregados a la Ab. Daniela Montenegro, por lo que, se pretende tomar sus requerimientos mediante la entrevista para definir un modelado de procesos de cómo cambian de etapa cada tramite académico.

ENTREVISTAS

Entrevista # 1: Dirigida a la administradora encargada de la secretaría general de la FISEI.

Objetivo:

Determinar los requerimientos, flujo de procesos de cómo se generan las resoluciones de los tramites académicos y funcionalidad del sistema web integrado.

N	Pregunta
1	¿Cuál es el problema que busca resolver a través del desarrollo del Sistema Web?
2	¿Cuáles son los trámites académicos que usted maneja en su labor diaria?
3	¿Los trámites que usted maneja son los mismos que manejan en las otras áreas?

Tabla 2.1. Matriz de preguntas para entrevista No 1 (Parte 1).
Elaborador por: El investigador.

4	¿Con que áreas usted trabaja con respecto a los tramites académicos anteriormente descritos?
5	¿Qué usuarios aparte de usted van a hacer uso del sistema web integrado?
6	En base al sistema que ocupa actualmente. Indique usted que mejoras le gustaría que se implementen en la nueva versión del nuevo sistema
7	¿Quién formula los tramites académicos?
8	Me podría indicar los diferentes tramites académicos que se utilizan en las anteriores áreas
9	¿Cuáles son los tramites académicos que no pasan por algún área en concreto, sino que pasan directo a secretaria de la Facultad?

Tabla 2.1. Matriz de preguntas para entrevista No 1 (Parte 2).

Elaborador por: El investigador.

Entrevista # 2: Dirigida a las secretarias encargadas de las diferentes áreas de la FISEI.

Objetivo:

Determinar los requerimientos, flujo de procesos de cómo se generan las resoluciones de los tramites académicos y funcionalidad del sistema web integrado.

N	Pregunta
1	¿Qué área es la que maneja usted?
2	¿Cuáles son los trámites académicos que usted maneja? Indique que tramites académicos son:
3	¿Qué procesos actualmente se encuentran automatizados en la gestión de los tramites académicos?
4	¿Qué tiempo demanda en su puesto de trabajo la gestión de tramites académicos?
5	¿Piensa Ud. que el desarrollo de una aplicación WEB le ayudaría a gestionar los tramites de una manera más eficiente? Justifique.
6	Ha manejado alguna vez algún sistema de trámites académicos. Indique que módulos y que funcionalidades realiza dicho sistema.

Tabla 2.2. Matriz de preguntas para entrevista No 1 (Parte 2).

Elaborador por: El investigador.

2.2 Métodos

2.2.1 Modalidad de investigación

Las modalidades de investigación son: de Campo, Bibliográfica y Modalidades Especiales.

Investigación de campo. - El investigador. llevará a cabo la recolección de información directamente de los involucrados a partir de una encuesta y la entrevista para formalizar los requerimientos sobre los procesos administrativos realizados de forma manual.

Investigación bibliográfica. - La investigación bibliográfica es porque utiliza fuentes como libros, tesis, artículos, revistas, etc. para la construcción de los antecedentes investigativos y marco teórico para analizar los diferentes Sistemas Web de tramites académicos administrativos.

Investigación modalidades especiales. - Porque se trata de implementar un sistema web en el que permita mejorar los procesos de resoluciones para los trámites de los usuarios como alternativa a una solución eficiente.

2.2 Población y muestra

Población: Para este caso se ha tomado a las principales áreas en las cuales trabajan las secretarías que son: Secretaría General, Consejo Académico de Facultad y la Unidad de Titulación e Integración Curricular.

Muestra: La muestra que se ha tomado con el fin de obtener el nivel de aceptación del sistema web integrado se toma de manera representativa como muestra lo siguiente:

Nº	Población	Número	Porcentaje (%)
1	Secretaria General y Consejo Directivo	1	33.33%
2	Consejo Académico de la Facultad	1	33.33%
3	Unidad de Titulación de Grado y Unidad de Integración Curricular	1	33.33%
Total		3	100%

Tabla 2.3. Muestra.

Elaborador por: El investigador.

El muestreo no se lo realizó puesto que la población total no excede los 100 individuos, por lo que, se trabajará con el total de la población sin que se requiera sacar una muestra representativa.

2.2.1 Recolección de información

Para recolectar la información se utilizará el siguiente cuestionario.

Preguntas Básicas	Explicación
¿Para qué?	Para alcanzar los objetivos de la investigación
¿De qué personas u objetos?	Departamentos, Secretaría de Facultad de Ingeniería en Sistemas, Electrónica e Industrial
¿Sobre qué aspectos?	Gestión de Trámites Académicos
¿Quién, Quiénes?	Investigador: Guillermo Barcia Gallardo
¿Cuándo?	Período académico octubre 2021 – febrero 2022
¿Dónde?	Universidad Técnica de Ambato en la Facultad de Ingeniería en Sistemas, Electrónica e Industrial
¿Cuántas veces?	Tres
¿Qué técnicas de recolección?	Entrevista
¿Con qué?	Cuestionario de entrevista

Tabla 2.4. Recolección de información.

Elaborador por: El investigador.

La primera entrevista se la realizó a la Ab. Daniela Montenegro, en donde se obtuvieron los siguientes resultados.

Pregunta	Resultado
¿Cuál es el problema que busca resolver a través del desarrollo del Sistema Web?	Optimizar tiempo, agilizar procesos, digitalizar trámites, mejorar niveles de eficiencia y efectividad.
¿Cuáles son los trámites académicos que usted maneja en su labor diaria?	<ul style="list-style-type: none"> - Elaboración de resoluciones. - Elaboración de informes académicos jurídicos. - Certificaciones. - Actas. - Memorándums.
¿Los trámites que usted maneja son los mismos que manejan en las otras áreas?	No son los mismos, pero guardan relación porque algunos trámites dependen de los otros.
¿Con que áreas usted trabaja con respecto a los tramites académicos anteriormente descritos?	Consejo Académico de Facultad, Unidad de Titulación, Unidad de Integración Curricular, secretarías de Carrera, Coordinaciones de carrera.
¿Qué usuarios aparte de usted van a hacer uso del sistema web integrado?	<ul style="list-style-type: none"> - Secretaría de Unidad de Titulación e Integración. - Secretaría de Consejo Académico. - Secretaría de Carrera y de Información. - Secretaría de Decanato.

Tabla 2.5. Recolección información-entrevista #1 (Parte 1).

Elaborador por: El investigador.

En base al sistema que ocupa actualmente. Indique usted que mejoras le gustaría que se implementen en la nueva versión del nuevo sistema	Búsqueda de datos que permita filtrar por diferentes campos, seguridad, es decir que tenga roles de usuario, que sea más flexible el manejo y que sea interactivo entre sus interfaces.
¿Quién formula los tramites académicos?	Cada responsable de área se encarga de realizar sus respectivos trámites académicos.
Me podría indicar los diferentes tramites académicos que se utilizan en las anteriores áreas	Acuerdos, Resoluciones, Actas y Certificaciones.
¿Cuáles son los tramites académicos que no pasan por algún área en concreto, sino que pasan directo a Secretaría de la Facultad?	Oficios, Memorándums, solicitudes o en otros términos se les denomina trámites varios.

Tabla 2.5 - Recolección información-entrevista #1 (Parte 2).

Elaborador por: El investigador.

Conclusión: Es necesario realizar un sistema web integrado, que permita optimizar los tiempos de respuesta hacia los estudiantes quienes necesitan recibir una notificación inmediata de la resolución de sus trámites académicos. Considerando que son diferentes las áreas que se encargan del manejo de trámites académicos las cuales son: Unidad de Titulación y Consejo Académico de Facultad.

La segunda entrevista se la realizó a la Ab. Elena Arcos y se obtuvieron las siguientes respuestas.

Pregunta	Respuesta
¿Qué área es la que maneja usted?	Unidad de titulación.
¿Cuáles son los trámites académicos que usted maneja? Indique que tramites académicos son:	Modalidad, aprobación de proyecto de investigación, aprobación de artículo académico, aprobación de proyecto de integración curricular y aprobación de proyecto de articulo académico en integración curricular. Examen complejo, prórroga gratuita y pagadas, informes de primera y segunda oportunidad de examen complejo.

Tabla 2.6. Recolección información-entrevista #2 (Parte 1).

Elaborador por: El investigador.

¿Qué procesos actualmente se encuentran automatizados en la gestión de los tramites académicos?	Ninguno.
¿Qué tiempo demanda en su puesto de trabajo la gestión de tramites académicos?	Las ocho horas diarias.
¿Cómo es la numeración que ustedes ocupan en sus trámites académicos?	Cada carrera tiene una numeración, en la empieza desde uno, pero cambia la nomenclatura dependiendo de la carrera. UTS (Carrera de Sistemas) UTE (Carrera de Electrónica) UICT (Carrera de Telecomunicaciones) UICS (Carrera de Ingeniería en Software) UICTI (Carrera de Tecnologías de la Información) UICI (Carrera de Ingeniería Industrial)
¿Piensa Ud. que el desarrollo de una aplicación WEB le ayudaría a gestionar los tramites de una manera más eficiente? Justifique.	Si, para evitar errores, no se equivocarían en números de los acuerdos o resoluciones, en ingresar los nombres de las carreras y se disminuye el tiempo de espera del estudiante de su trámite académico.
Ha manejado alguna vez algún sistema de trámites académicos. Indique que módulos y que funcionalidades realiza dicho sistema.	Nunca.

Tabla 2.6 - Recolección información-entrevista #2 (Parte 2).

Elaborador por: El investigador.

Conclusión: En esta entrevista se concluye que, la persona encargada del área de Unidad de Titulación, no ha utilizado alguna herramienta que le permita gestionar de manera óptima y rápida los trámites académicos, por lo tanto se hace necesario realizar el sistema web integrado, sin embargo, la numeración de estos trámites es diferente a como se maneja en Secretaria de Facultad, ya que, en este caso, depende de la carrera para ir numerando los Acuerdos o Resoluciones, que posteriormente son enviados a la abogada Daniela Montenegro.

La tercera entrevista que se realizó a Ing. Myrian Viteri y se obtuvieron las siguientes respuestas.

Pregunta	Respuesta
¿Qué área es la que maneja usted?	Consejo Académico de Facultad (Subdecanato).
¿Cuáles son los trámites académicos que usted maneja? Indique que tramites académicos son:	Distributivos, horarios de docentes y autoridades, informes regulares (informes de culminación de periodo académico, tutorías académicas). Reingresos, terceras matriculas, retiros de asignaturas, cambios de universidad, cambios de carrera.
¿Qué procesos actualmente se encuentran automatizados en la gestión de los tramites académicos?	Por el Quipux, un sistema que ocupan para el ingreso de documentos y diferentes trámites. Dichos trámites se los recibe y responde a nivel de universidad.
¿Qué tiempo demanda en su puesto de trabajo la gestión de tramites académicos?	De las ocho horas al día, son siete horas trabajadas en el cual se realiza la gestión de trámites académicos.
¿Cómo es la numeración que ustedes ocupan en sus trámites académicos?	La numeración empieza desde 1 y se reinicia cada año, es decir, que cada año, empieza desde 1. Un ejemplo: Acuerdo 001-CADFISEI
¿Piensa Ud. que el desarrollo de una aplicación WEB le ayudaría a gestionar los tramites de una manera más eficiente? Justifique.	Si, por que ayudaría a reducir errores que surgen en los trámites, debido a que, en ocasiones los nombres del estudiante no se encuentran bien escritos. Al haber una gran cantidad de tramites, es posible que existan errores tipográficos al pasar las resoluciones a Consejo Directivo, por lo que, mediante un sistema, se reduce el tiempo de respuesta y las equivocaciones humanas.
Ha manejado alguna vez algún sistema de trámites académicos. Indique que módulos y que funcionalidades realiza dicho sistema.	Si, fue un programa piloto, en el cual contaba con módulos de estudiantes y docentes, que permitía filtrar por los datos principales y que ya le aparecía los formatos de los Acuerdos que utilizan.

Tabla 2.7.-Recolección información-entrevista #3.

Elaborador por: El investigador.

Conclusión: En esta última entrevista, se concluye que, la numeración de los trámites académicos es similar a la que utilizan en la Secretaría de Facultad, pero con la diferencia, de que en esta área de Consejo Académico de Facultad se realizan Acuerdos. Se debe considerar, que la persona encargada del área, anteriormente si ha utilizado un programa piloto, entonces es necesario realizar un sistema web integrado, esto ayudaría a minimizar errores tipográficos, ya sea en los nombres de los estudiantes y en las carreras.

2.2.4. Procesamiento y análisis de datos

En base a la aplicación de la entrevista realizada a las personas encargadas de las diferentes áreas, se realiza una tabla resumen donde se evidencia el nombre del entrevistado, los requerimientos y la conclusión general.

N°	Entrevistado	Requerimientos	Conclusión general
1	Ab. Daniela Montenegro	Lo que se necesita es optimizar tiempo, agilizar procesos, digitalizar trámites, mejorar niveles de eficiencia y efectividad.	La importancia de realizar un sistema web integrado, es necesario, ya que permitirá automatizar algunos procesos, que se realizan en los trámites académicos. Teniendo en cuenta que los datos de los estudiantes ya se van a tener almacenados, se va a tener una mejor gestión de la información, permitiendo así que no exista equivocaciones tipográficas en los documentos de los trámites, por no tener ingresado correctamente los datos del alumno.
2	Ab. Elena Arcos	Evitar equivocaciones en los números de los acuerdos o resoluciones, disminuyendo el tiempo de espera del estudiante por su trámite académico.	Debido a la gran demanda de documentos que manejan, se toma en ocasiones más de ocho horas diarias de trabajo, por lo que, con un sistema web integrado, se pretende mejorar estos tiempos.
3	Ing. Myrian Viteri	Optimizar el tiempo de respuesta y sobre todo reducir las equivocaciones humanas, debido a que, en ocasiones los nombres del estudiante no se encuentran bien escritos y al haber una gran cantidad de tramites, pueden existir equivocaciones tipográficas al pasar las resoluciones a Consejo Directivo.	Existe la necesidad de implementar el sistema web integrado, ya que, actualmente 2 de las 3 áreas no tienen una aplicación para gestionar los documentos de los trámites académicos, por lo que, mediante esta herramienta tecnológica le servirá para mejorar los tiempos de respuesta hacia los estudiantes y ser más eficientes en su puesto de trabajo. Los resultados obtenidos en las entrevistas sirven para determinar los requerimientos y funcionalidad del sistema.

Tabla 2.8.-Procesamiento de datos de los resultados obtenidos de las entrevistas.

Elaborador por: El investigador.

CAPÍTULO III.- RESULTADOS Y DISCUSIÓN

3.1 Desarrollo de la propuesta

Se realiza el desarrollo de la propuesta según los objetivos específicos planteados al inicio del proyecto de investigación, los cuales se detallan a continuación.

3.1.1 Análisis de los procesos administrativos que se realizan en la FISEI para su automatización.

En la FISEI de la UTA existen tres áreas, una de ellas es el área de titulación la cual se divide en dos Consejos que son: Unidad de Titulación (Malla Antigua: Carrera de Sistemas Computacionales e Informáticos, Electrónica, Telecomunicaciones e Industrial en Procesos de Automatización) y Unidad de Integración Curricular (Nueva Malla: Tecnologías de la Información, Software, Industrial, Electrónica y Comunicaciones), las otras dos áreas constan cada una de un Consejo, esto se puede evidenciar en la figura 3.4.

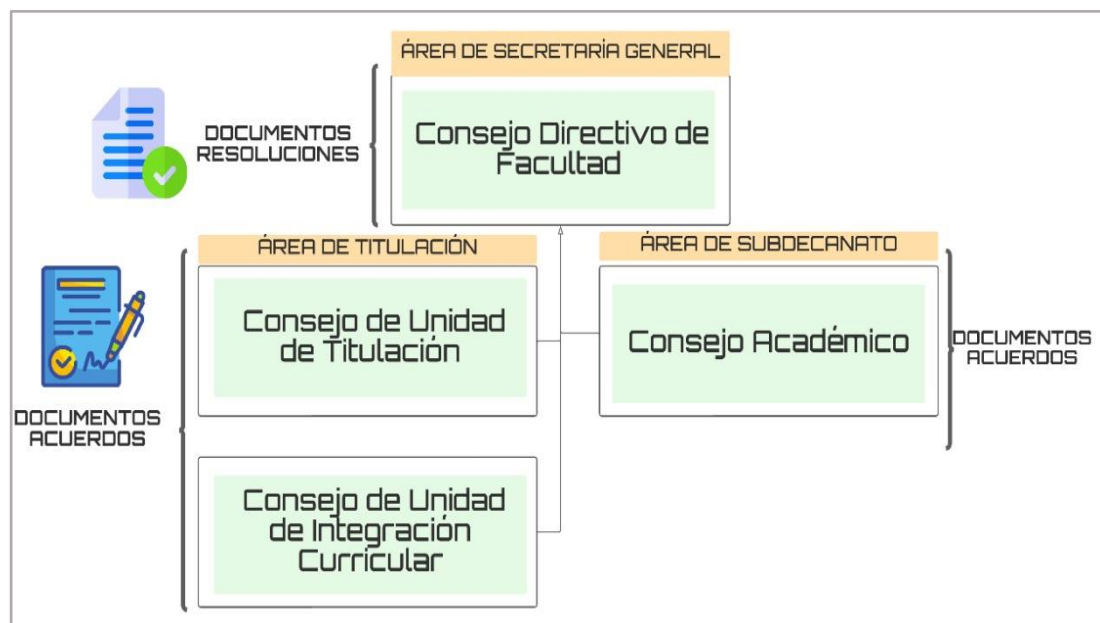


Figura 3.4.- Estructura organizacional de la FISEI de la UTA.

Elaborado por: El investigador.

A continuación, se realiza una representación gráfica de los principales procesos que se llevan a cabo en cada área de la FISEI, las cuales muestran una secuencia e interacción de las actividades realizadas, además de la identificación de las personas que intervienen en cada proceso a través de símbolos gráficos, con el fin de

proporcionar una mejor visualización, en cuanto al funcionamiento del proceso y sobre todo buscar una mejora continua.

El proceso de Consejo Directivo de Facultad se enfoca en dos aspectos, las áreas internas – externas y los estudiantes, en la figura 3.5 se muestra el proceso para las áreas internas – externas y se explica la secuencia correcta del ingreso de la documentación por Quipux, identificando los involucrados en el proceso los cuales son: responsable de información, la secretaría de Consejo Directivo de Facultad y la sesión de Consejo Directivo en donde se realiza la resolución del trámite.

3.2 Metodología de desarrollo

Para la realización del proyecto de investigación es necesario seleccionar una metodología de desarrollo del sistema integrado. A continuación, se realiza una comparación de metodologías ágiles para el proyecto, seleccionando la metodología que cumpla con las necesidades y expectativas del investigador y del cliente.

3.2.1 Comparación de metodologías para el desarrollo del sistema

Para el desarrollo de este sistema, se deberá partir escogiendo la metodología que mejor se adapte al proyecto con la finalidad de conseguir los objetivos que se plantearon. Como punto de partida de deberá realizar el análisis de características entre las metodologías listadas en la tabla 3.9 y posterior se seleccionará la metodología que más se ajusta al sistema que se desea implementar.

Para poder decidir sobre cual metodología es la que se deberá escoger se realizará un estudio comparativo de las principales metodologías ágiles.

Proceso de Consejo Directivo de Facultad (Áreas Internas/ Externas)

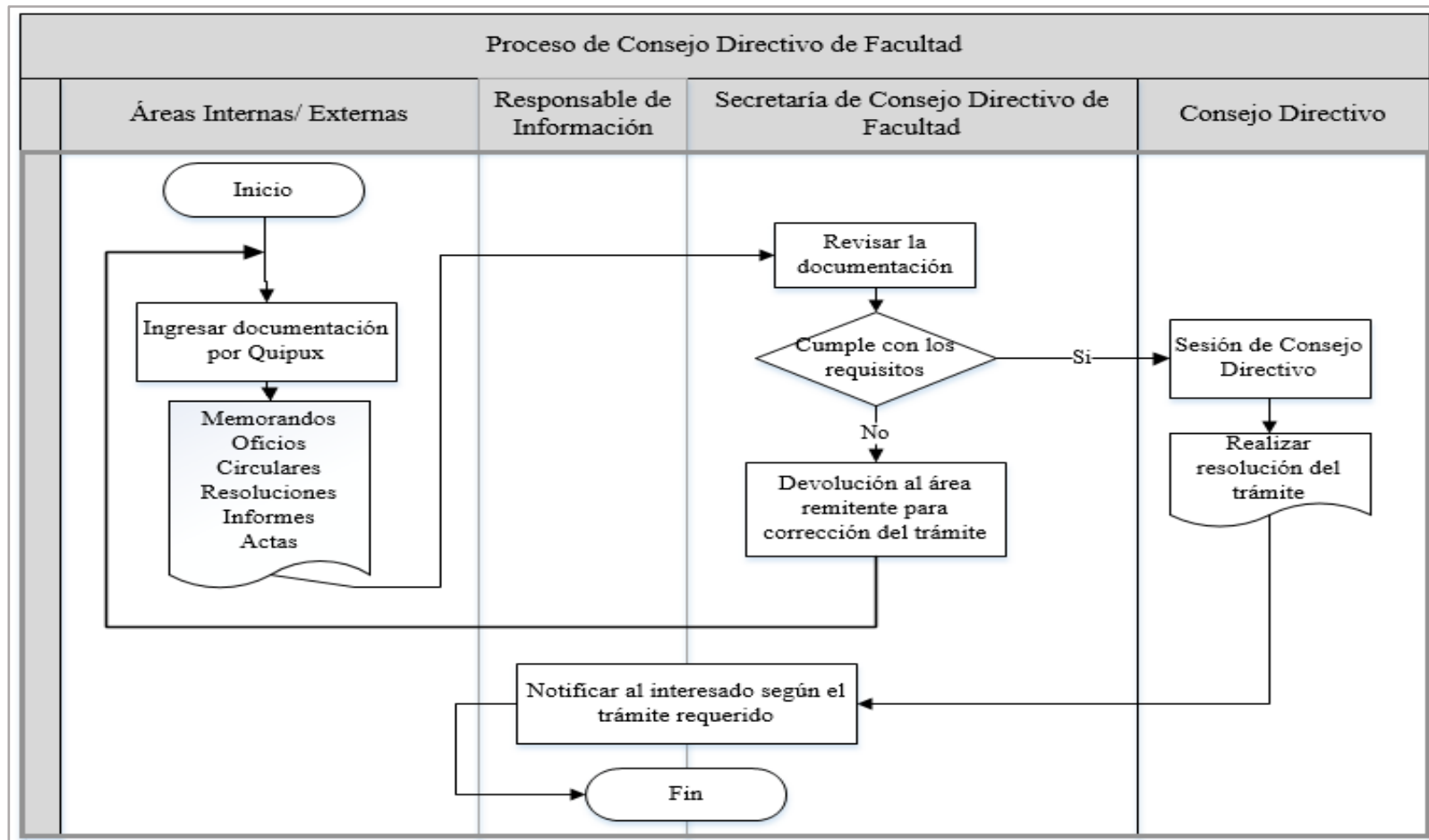


Figura 3.5.- Proceso de Consejo Directivo de Facultad (Áreas Internas/ Externas).
Elaborado por: El investigador.

Proceso de Consejo Directivo de Facultad (Estudiantes)

En la figura 3.6 se muestra el proceso para los estudiantes, donde el ingreso de un trámite se lo hace por solicitud y los tipos de trámites que son: oficios y anexos.

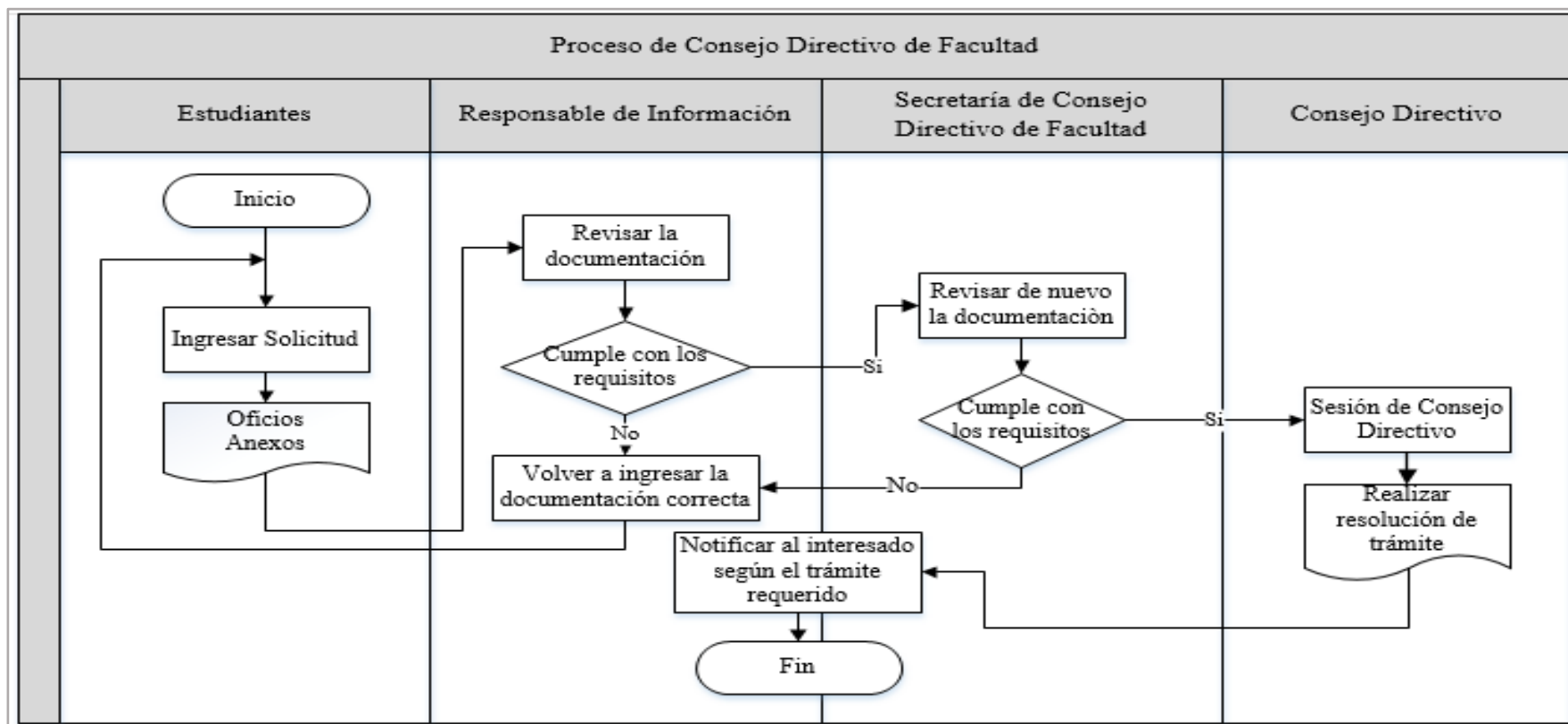


Figura 3.6.- Proceso de Consejo Académico de Facultad (Estudiantes).
Elaborado por: El investigador.

Proceso de Consejo Académico de Facultad

En este proceso intervienen los estudiantes, el responsable de información, la secretaria de Consejo Directivo de Facultad y el Consejo Directivo, en donde inicia con una solicitud del estudiante y termina con la obtención de una resolución del trámite requerido, como se puede observar en la figura 3.7.

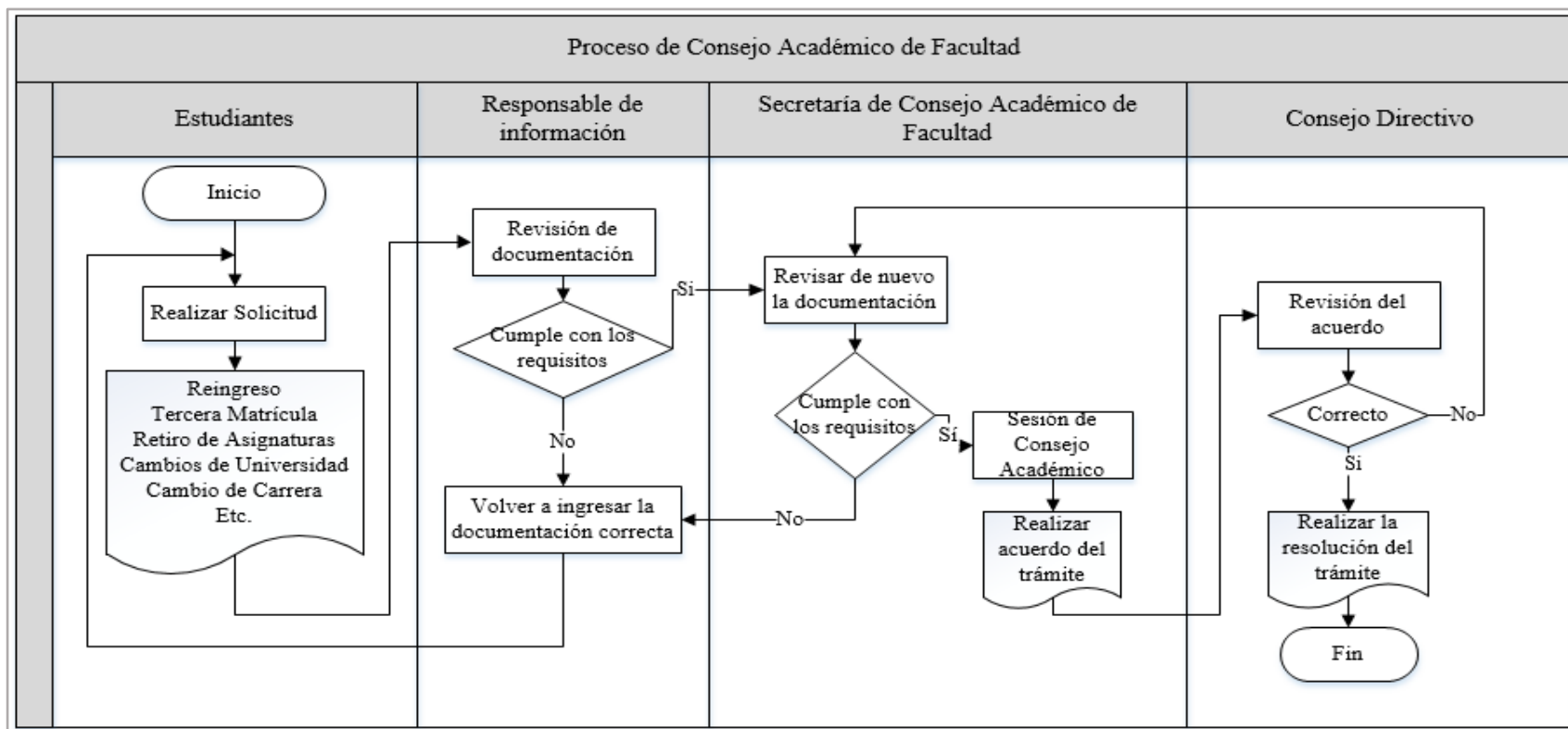


Figura 3.7.- Proceso de Consejo Académico de Facultad.
Elaborado por: El investigador.

Proceso de Unidad de Titulación e Integración Curricular

En el siguiente proceso intervienen los estudiantes, un responsable de información, la secretaría de Titulación e Integración Curricular y el Consejo Directivo. La única variación con respecto al anterior proceso es el tipo de trámite, tal como lo muestra la figura 3.8.

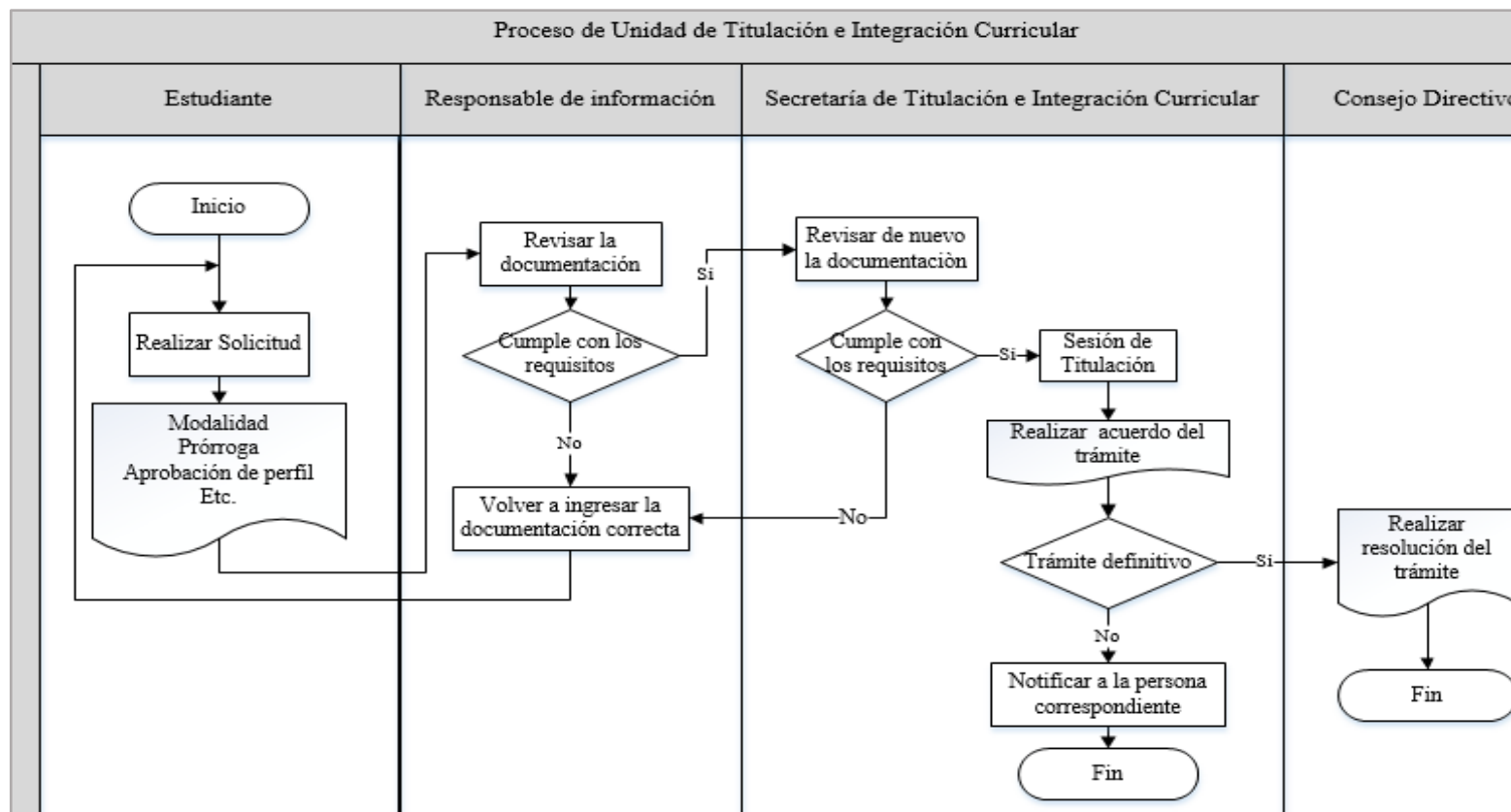


Figura 3.8.- Proceso de Unidad de Titulación e Integración Curricular

Elaborado por: El investigador.

Metodología XP (Extreme Programming)

Es una metodología ágil conocida. Fue desarrollada por Kent Beck buscando guiar equipos de desarrollo de software pequeños, en ambientes de requerimientos imprecisos o cambiantes. La cual ha evolucionado a partir de los problemas causados por los largos ciclos de desarrollo de los modelos de desarrollo tradicionales. El proceso XP puede caracterizarse por ciclos de desarrollo cortos, planificación incremental, retroalimentación continua, confianza en la comunicación y diseño evolutivo [14]. Además, es una metodología centrada en potenciar las relaciones interpersonales como clave para el éxito en desarrollo de software, fomentando el trabajo en equipo de este modo propiciando un buen clima de trabajo.

Por consiguiente, se tiene en consideración las características que son esenciales de XP, organizadas en los tres apartados siguientes: historias de usuario, roles, proceso y prácticas [15].

Características [16]

- La interacción entre el cliente y el equipo de desarrollo es constante.
- La planificación es flexible y abierta.
- Cuenta con una rápida respuesta a cambios.
- Requisitos del cliente y trabajo del equipo del proyecto son factores prioritarios para el éxito del mismo [17].
- Cuenta con cinco fases de desarrollo; planificación, diseño, codificación, pruebas y lanzamiento.
- Hay una lista predeterminada de funciones para cada miembro del equipo [15].

Ventajas [18]

- Posee una programación organizada.
- Es un acelerador de las tareas, optimizando el ahorro de tiempo y de dinero.
- El cliente tiene el control ya que el proceso se maneja en función de sus prioridades.
- El producto obtenido pasa por varias pruebas continuas verificando la funcionalidad.
- Es una de las mejores metodologías aplicables.

- Existe una mejora continua de todos los procesos, además del equipo de desarrollo.
- Se lo puede aplicar en cualquier lenguaje de programación.

Desventajas [16]

- Dificultad en llevar un registro de lo que se ha realizado.
- Se recomienda aplicar la metodología solo a proyectos con objetivos a corto plazo, por el hecho de que, si son proyectos extremos, la programación puede tener alguna falla.
- En el caso de que el proceso falle, el gasto representado es considerable.

Metodología Scrum

Es un proceso iterativo e incremental para desarrollar cualquier producto o administrar cualquier trabajo, basándose primero en construir funcionalidad de mayor valor para el cliente y en los principios de inspección continua, adaptación, autogestión e innovación.

Características [19]

- Organización autónoma de los equipos, es decir el equipo de desarrollo encuentra su propia forma de operar.
- Cuenta con fases de desarrollo simultáneas.
- Tiene un entorno que potencia el aprendizaje individual y grupal ya que es un sistema multinivel.
- Incluye puntos de inversión que ayudarán a reducir la ambigüedad y las tensiones, promoviendo la estabilidad.
- Cuenta con tres fases; preparación, juego, postjuego.
- Transmisión organizacional del aprendizaje.

Ventajas [20]

- Crea productos en un tiempo menor, listos para salir al mercado.
- Ofrece el marco necesario para cambiar de rumbo o de enfoque cuando el proyecto lo necesita.
- Ayuda a comunicarse de manera interna, ayudando a retroalimentar el proceso y encontrar los errores.

- Evalúa el rendimiento individual y del equipo, propiciando su mejora.
- Las necesidades del cliente pueden ser gestionadas de manera rápida.

Desventajas [21]

- Cuando los objetivos no están definidos correctamente, existirá ambigüedad en todo el proceso.
- Si el cliente tiene demandas nuevas en partes avanzadas del proceso, debe establecer fechas límite.
- Es indispensable que los miembros tengan experiencia o capacidad; de lo contrario, el proyecto puede fallar.

Metodología Crystal

Se trata de un conjunto de metodologías para el desarrollo de software, se caracteriza por estar centradas en las personas que conforman el equipo y la reducción al máximo del número de artefactos producidos. [15] Está compuesta por: Crystal Clear, Crystal Yellow, Crystal Orange y Crystal Red, el nivel de opacidad del color en el nombre indica un mayor número de personas implicadas en el desarrollo, un mayor tamaño del proyecto y, por lo tanto, la necesidad de mayor control en el proceso [20]. En la metodología Crystal existen dos reglas. La primera indica que los ciclos donde se crean los incrementos no deben exceder cuatro meses; la segunda, que es necesario realizar un taller de reflexión después de cada entrega para afinar la metodología [20].



Figura 3.9.- Clasificación de proyectos por colores.

Fuente: Universidad Politécnica de Valencia [15].

Características [22]

- Cuentan con la mejora continua en las iteraciones.
- El equipo está en una misma ubicación física, para lograr la comunicación cara a cara.
- Fácil acceso a usuarios expertos, es decir no exigen que los usuarios estén continuamente junto al equipo de proyecto.

- Consta de un entorno técnico con pruebas automatizadas, gestión de la configuración e integración continua.
- Es ultraligero es debido a que Crystal no recomienda demasiada documentación, administración general e informes.
- Utiliza un enfoque en las personas y la comunicación como su principio organizador.
- Tiene procesos flexibles y configurables.
- Participación activa del usuario.

Ventajas [23]

- Se centra en la Criticidad y el tamaño del equipo, lo que ayuda en la definición y control del sistema.
- El equipo puede reducir artefactos intermedios a medida que se produzca código funcional y se usen canales de comunicación óptimos.
- El factor principal para Crystal es el equipo de trabajo y la comunicación.
- Son apropiadas para entornos ligeros.
- Al estar diseñada para el cambio experimenta reducción de costo.
- Presenta una planificación más transparente para los clientes.
- Permite tener una muy útil realimentación de los usuarios.

Desventajas [24]

- Delimita el alcance del proyecto con el cliente.
- Como no es una metodología rígida, dejando abierta la posibilidad de agregar y suprimir fases, productos, estrategias, técnicas que pueden afectar en el desarrollo del sistema.

Comparativa de metodologías ágiles

Las metodologías indicadas anteriormente sirven para procesar los requerimientos que los usuarios necesiten con la finalidad de obtener un sistema acorde a las necesidades. Además, se han elegido las siguientes características porque se necesita realizar una comparación entre las metodologías que mejor se adaptarían al proyecto de investigación.

Metodología	Extreme Programming	Scrum	Crystal
Tamaño de proyectos	Pequeños y medianos	Pequeños y medianos	Pequeños, medianos y grandes proyectos
Tamaño de equipo	Menor que 10	Múltiples equipos menores que 10	Desde 3 integrantes hasta 100 integrantes dependiendo del tipo de metodología escogida.
Objetivos	Basada en dar prioridad a trabajos con resultado directo: -Satisfacción del cliente. - Trabajo en grupo. - Actuar sobre variables: Costo, Tiempo, Calidad y Alcance.	Indicado para proyectos en entornos complejos: -Obtener resultados pronto. -Requisitos cambiantes. -Innovación y competitividad fundamentales.	Enfocado en ajustar las técnicas utilizadas en un proyecto: -Fortalecer proceso de comunicación en equipo. -Integración continua. -Procesos flexibles y configurables.
Estilo de código	Limpio y sencillo.	No especificado.	No especificado.
Cultura de negocio	Colaborativo y cooperativo.	No especificado.	Colaborativo y cooperativo.
Estilo de desarrollo	Iterativo y rápido.	No especificado.	Iterativo incremental.
Tipo de desarrollo	-Planificación del proyecto -Diseño. -Codificación. -Pruebas.	-Desarrollo simple que requiere trabajo duro. -Control de forma empírica y adaptable a la evolución del proyecto.	-Iteración integrada en el sistema general. -Refinamiento. -Finalización software en general.

Tabla 3.9.-Comparación de metodologías.
Elaborado por: El investigador.

3.2.2 Metodología seleccionada

Para la elección de la metodología apropiada se debe tomar en consideración la naturaleza del proyecto a desarrollar, es por ello por lo que se realizó un cuadro comparativo en la tabla 3.9, y se optó por la metodología XP debido a la viabilidad que ofrece en la realización de pruebas unitarias, ayudando a la prevención y corrección de errores que se expongan al desarrollar el proyecto. Además, permitirá obtener productos tangibles en corto plazo y también, las funcionalidades descritas en cada historia de usuario.

3.2.3 Análisis de las herramientas de desarrollo

3.2.3.1 Estudio comparativo entre tecnologías de front-end

Se evidencia en el año 2022 la existencia de diferentes tecnologías para el desarrollo de Front End, entre los cuales los más destacados son: .Net Framework, VueJS,

ReactJS, Gatsby, Angular, entre otros. Todas estas tecnologías sirven para el desarrollo de interfaces de usuarios en páginas web. Para el presente proyecto de investigación, se realiza una comparación de características entre estas tecnologías ReactJS, .Net Framework y VueJS, con la finalidad de determinar la mejor tecnología que se adapte a los requerimientos del proyecto.

VueJS

Es un marco progresivo para crear interfaces de usuario, basado en JavaScript y creado por Evan You. Inicialmente, se diseñó como un marco para crear aplicaciones de una sola página conocidos como SPA (Single Page Application), pero con el paso del tiempo, VueJS se ha ampliado significativamente [29].

Características [30]

- Es versátil ya que el núcleo es bastante pequeño y se escala mediante plugins, donde Vue es una librería muy parecida a React.
- Es escalable por ser versatilidad.
- Es reactivo.
- El core ocupa 74KB, como ves es bastante liviano.
- Crece a un ritmo importante con más 66500 estrellas en GitHub y 130 personas contribuyendo al core cada día.
- Licencia MIT.
- Curva de aprendizaje baja,
- Framework MVVM, es decir tiene facilidad para construir un código bien estructurado.

Ventajas [31]

- Fácil de aprender y utilizar, si se tiene algo de experiencia en desarrollo Front End, en un par de días ya puede dominar este marco de desarrollo.
- Ligerero, haciendo referencia a que el marco pesa 18KB, lo que permite descargar en cuestión de segundos, con la ventaja de su característica que es rápida y puede superar constantemente a los marcos voluminosos.
- Reutilización de componentes, que son básicamente fragmentos de código que se puede reutilizar como plantillas para elementos de sistemas similares.

- Documentación concisa y soporte comunitario, ya que se encuentra respaldado por lo que es de código abierto.

Desventajas [32]

- Limitaciones en el uso de la comunidad, ya que existe la barrera del idioma, que es principal obstáculo para usar este marco de desarrollo, porque sus creadores son chino-americanos y prefieren el idioma chino a cualquier otro, que no se utiliza en la mayoría de los países.
- Falta de apoyo para proyectos a gran escala, por ende, no tiene la característica de ser una tecnología escalable si se trata de grandes proyectos.
- Flexibilidad excesiva, por lo que, genera mucha más irregularidades y errores en el código.

ReactJS

Todos los desarrolladores web aprecian y reconocen a React.js como una de las principales bibliotecas de JavaScript de front-end. Muchas empresas multinacionales populares como Facebook, Netflix, Salesforce, Dropbox y otros servicios utilizan React para potenciar sus aplicaciones web. Hay más de 10,499,785 sitios web en vivo creados con React.js según Builtwith. Según un estudio publicado en Stantista (con tamaño de muestra 67.593 encuestados), React.js fue un claro ganador en la lista de los marcos web más utilizados (40,14%) entre los desarrolladores de todo el mundo en 2022 [25].

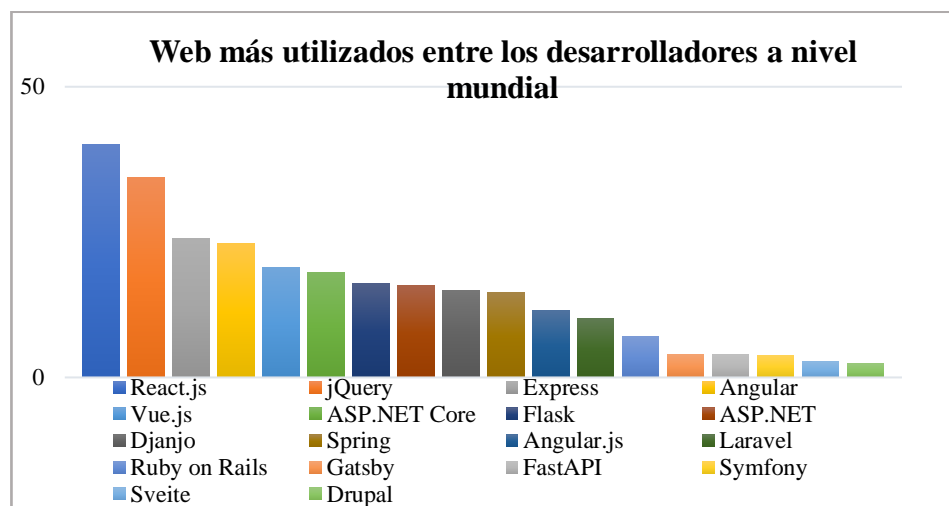


Figura 3.10.- Web más utilizados entre los desarrolladores a nivel mundial.
Fuente: GLOWID IT SOLUTIONS PVT.LTD [25].

Características [26]

- Flujo de datos unidireccional.
- Es de código abierto.
- Ayuda a crear impresionantes aplicaciones web.
- Requieren un mínimo esfuerzo y codificación.
- Las interfaces de usuario son muy veloces.
- La codificación es reutilizable.
- Ofrece una biblioteca de JavaScript muy amplia, que brinda, flexibilidad para las aplicaciones web.

Ventajas [27]

- Desarrollo rentable.
- Aprovecha JavaScript.
- Fácil de aprender y usar, si sabes JavaScript, puede comprender fácilmente y crear aplicaciones web usando React en pocos días.
- Reducción de código gracias a la utilización de componentes, por lo que, proporciona menos codificación y más funcionalidad.
- Excelente rendimiento de la aplicación.
- Diseño modular porque se trabaja por bloques libres llamados módulos, que son reutilizables para API (Interfaz de Programación de Aplicaciones) Web y Móvil.

Desventajas [28]

- Para los desarrolladores novatos en esta nueva tecnología se le resulta difícil de acoplar JSX (JavaScript XML), ya que es una extensión de sintaxis que permite mezclar HTML con JavaScript, sin embargo, para los nuevos programadores, se les resulta complicado en la curva de aprendizaje.
- Ausencia de documentación oficial.
- La alta velocidad de desarrollo de ReactJS solo deja lugar a una documentación apropiada, por lo tanto, diferentes programadores contribuyen sin un enfoque común.
- No existe un estándar de desarrollo por lo que se tiene demasiadas elecciones a tomar.

- Enfoque no vigilante en la interfaz de usuario.

Blazor

Blazor es un marco web relativamente nuevo de Microsoft ASP.NET Core que permite a los desarrolladores escribir código para navegadores en C#, el cual se basa en tecnologías web existentes como HTML (Lenguaje de Marcado de Hipertexto) y CSS (Hojas de Estilo en Cascada), pero luego permite que el desarrollador use la sintaxis de C# y Razor en lugar de JavaScript. Blazor es una sintaxis de marcado de plantilla popular para .NET (Red), proporcionando una plataforma para permitir el desarrollo de Aplicaciones de una Sola Página (SPA) dinámicas y contemporáneas mientras usa .NET de un extremo a otro [33].

Características [34]

- Blazor (Browser + Razor) permite a los desarrolladores crear una interfaz de usuario web interactiva y reutilizable para aplicaciones del lado del cliente escritas en .NET e implementadas en WebAssembly Utiliza código c#, en lugar de JavaScript.
- Con el código del cliente y del servidor escrito en C#, le permite compartir código y biblioteca.
- Es compatible con los componentes integrados de formulario y entrada para implementar un excelente nivel de abstracción.

Ventajas [35]

- Se ejecuta en un entorno de espacio aislado seguro para la memoria y es rápido en lo que respecta a la ejecución. Funciona de manera similar a las aplicaciones nativas.
- Es compatible con todas las características de SPA necesarias, que incluyen componentes, enrutamiento e inserción de dependencias para una experiencia de desarrollo enriquecida.
- Las aplicaciones creadas por Blazor se pueden implementar y ejecutar como archivos estáticos, donde las máquinas no tienen .NET.
- Es compatible con todos los navegadores populares, incluidas las versiones anteriores.
- Puede funcionar perfectamente sin complementos de navegador.

Desventajas [36]

- Al ser el navegador el que ejecuta toda la aplicación, se restringe sus capacidades.
- Al ser descargada toda la información al navegador, no sería conveniente almacenar información sensible como contraseñas en este tipo de aplicaciones.
- No se tiene un soporte a conexiones fuera de línea, es decir, siempre tendremos que tener una conexión activa a internet.
- Se necesita sí o sí, un servidor ASP.NET Core para ejecutar la aplicación.
- Y también, al necesitar de una conexión constante al servidor, la aplicación será menos rápida que Blazor Webassembly. Esto no quiero decir que la aplicación se ejecutará lentamente, puede que ni siquiera notemos la diferencia de velocidad, pero siempre una aplicación Blazor WebAssembly se ejecutará más rápido que una aplicación tipo Blazor Server.

Características	Reactjs	Vuejs	Blazor
Creador	Facebook	Evan You	Microsoft
Tipo	Librería	Framework	Framework
Ideal para	Desarrollo web moderno, para grandes proyectos.	Desarrollo web y aplicaciones simples.	Desarrollo web y aplicaciones de todo tipo, desde aplicaciones de gestión, hasta un emulador.
Lenguaje	JavaScript	JavaScript	C# y Razor pages
Modelo DOM	Virtual	Virtual	Virtual
Curva de aprendizaje	Media	Fácil	Fácil
Arquitectura	Arquitectura Flux	MVC	MVC
Robustez	Grande	Mediana	Grande

Tabla 3.10.- Cuadro comparativo entre ReactJs, VueJs y Blazor.
Elaborado por: El investigador.

3.2.3.2 Tecnología de Desarrollo Front End Seleccionada

Presentadas las características más destacadas, con las ventajas, desventajas y cuadro comparativo entre los diferentes marcos de desarrollo para el Front End, se pudo

determinar que la tecnología a utilizar es la Librería de ReactJS, debido a que, se encuentra desarrollada en una de las bibliotecas más extensas en el mercado como lo es JavaScript, por lo tanto, ofrece diferentes ventajas dentro de las aplicaciones web, por la flexibilidad y reutilización del código gracias a la generación de múltiples componentes.

Además, se encuentra respaldada por diferentes empresas que han utilizado esta tecnología para crear interfaces web modernas, tales como netflix, amazon, spotify y por la escalabilidad que tiene al poder trabajar con proyectos grandes. Tomando en cuenta que en el 2022 muchas personas se están sumando en utilizar esta librería por las características anteriormente mencionadas y muchas mejoras actuales, que se dan en este marco de desarrollo web para Front End [36].

3.2.3.3 Estudio comparativo entre tecnologías de back-end

Laravel

Es un marco de desarrollo de aplicaciones web, basado en el lenguaje de programación PHP de código abierto, en donde, su patrón arquitectónico es MVC (Modelo-Vista-Controlador). Además, de reconocer que este es el framework PHP más utilizado y conocido en el mundo. El uso de Laravel ayuda a evitar perder cientos de horas de tiempo y escribir proyectos seguros [37].

Al ser un marco que se basa en PHP que se encuentra en el lado del servidor, hace que esta tecnología trabaje en el entorno de Back End, sin embargo, con laravel, se puede desarrollar aplicaciones web de pila completa, es decir, que tiene las capacidades de admitir el trabajo Front End, gracias a su arquitectura MVC [37].

Ventajas

- Gran comunidad de usuarios, por lo que cuando se tiene algún problema, se pueda solventar de la manera más rápida posible. Por otro lado, la documentación de Laravel es muy útil, se puede encontrar explicaciones buenas y detalladas de estilo de codificación, métodos y muchas cosas más.
- Hay que destacar, que una de las ventajas más poderosas de Laravel, es el hecho de tener su propio gestor de comandos, tal como, es el caso de Artisan, que es una herramienta de línea de comandos que ayuda a crear el código esqueleto y ejecuta el sistema de base de datos.

- Por otra parte, el marco de Laravel, tiene la facilidad de conectarse a la base de datos, utilizando el poderoso ORM (Modelo relacional de objetos) llamado Eloquent, que permite utilizar la sintaxis PHP para crear el modelo de base de datos, sin necesidad de escribir o saber SQL (Lenguaje de consulta estructurada) para actualizar o modificar la base de datos [37].

Desventajas

- Se debe tener en cuenta que, el marco de Laravel se basa en las reglas más estándar de orientación a objetos, patrones de diseño y la arquitectura MVC, por lo cual, se debe aprender cómo usarlo y familiarizarse con los estándares, antes de ponerse a desarrollar un proyecto.
- Se debe tener paciencia y constancia entender PHP, llevara un tiempo ya que, la documentación de Laravel no es tan agradable [37].

ASP.NET Core

ASP.NET Core es un nuevo marco multiplataforma y de código abierto para crear aplicaciones modernas conectadas a Internet basadas en la nube, como aplicaciones web, las cuales pueden ejecutarse en .NET Core o en .NET Framework completo. Fue diseñado para proporcionar un marco de desarrollo optimizado para aplicaciones que se implementan en la nube o se ejecutan en las instalaciones. Consta de componentes modulares con una sobrecarga mínima, por lo que conserva la flexibilidad mientras construye sus soluciones. Puede desarrollar y ejecutar sus aplicaciones en Windows, Mac y Linux., además es de código abierto [38].

Ventajas

- El paquete NuGet de la plataforma .net core ofrece una funcionalidad pequeña y discreta en comparación con el marco monolítico de .NET. Las aplicaciones son más ligeras y esbeltas.
- El núcleo de ASP.NET brinda soporte directo y libera a los desarrolladores de las dependencias de terceros, lo que mejora el rendimiento general.
- Debido a su tamaño más pequeño, ofrece sorprendentes beneficios de rendimiento.

- El marco ofrece Visual Studio, que está integrado con el entorno de desarrollo. Sirve como un marco de construcción vital y facilita a los desarrolladores la creación de aplicaciones de manera eficiente.
- Ofrece despliegue modular [39].

Desventajas

- Ciertos códigos no funcionan.
- Realizar cambios en la aplicación puede ser un dolor de cabeza porque es posible que los cambios no funcionen en la próxima versión de la aplicación.
- En comparación con las alternativas de código abierto, resulta un poco caro debido a los cargos adicionales, como las licencias de Visual Studio, las licencias de Windows Server y las licencias de SQL Server.
- Necesitará utilidades de terceros, como Internet Service Manager, para acceder a los ajustes de configuración en una aplicación web ASP porque está almacenada en una metabase de IIS (Internet Information Services) en un formato propietario. Además, la extracción y manipulación de la configuración es una tarea bastante ardua debido al soporte limitado.
- Es posible que también tenga que lidiar con documentación inconveniente al usar el marco de trabajo para compilar las aplicaciones MVC en el alojamiento de MVC de red asp [40].

Node JS

Es una plataforma construida sobre el tiempo de ejecución de JavaScript de Chrome para crear fácilmente aplicaciones de red rápidas y escalables, el cual utiliza un modelo de E/S sin bloqueo y basado en eventos que lo hace liviano y eficiente, perfecto para aplicaciones en tiempo real con uso intensivo de datos que se ejecutan en dispositivos distribuidos, consta de un entorno de tiempo de ejecución multiplataforma de código abierto para desarrollar aplicaciones de red y del lado del servidor [41].

Ventajas

- Tiene un entorno de tiempo de ejecución de código abierto, por lo que es de uso gratuito bajo la licencia MIT. Node JS es la multiplataforma que se

ejecuta en Windows, Mac, Linux. También hay muchos módulos de código abierto compatibles con Node JS.

- El lenguaje Javascript se usa para codificar en Node JS tanto para el frontend como para el backend. Como desarrollador, no es necesario que aprenda un nuevo lenguaje de programación del lado del servidor para Node JS. Incluso un desarrollador de Java novato puede encontrar fácil trabajar en la plataforma Node JS.
- Todas las capas, desde el front-end hasta el back-end, admiten el mismo formato de datos, es decir, JSON (notación de objetos de script de Java).
- Node JS utiliza el motor V8 desarrollado por Google para Chrome. V8 compila JavaScript en código de máquina nativo directamente, lo que conduce a velocidades más rápidas y una ejecución eficiente del código.
- Node JS utiliza programación asíncrona. Cada operación de E/S no bloquea, puede realizar varias operaciones al mismo tiempo. Los usuarios pueden enviar fácilmente múltiples correos, operaciones de base de datos y leer archivos simultáneamente.
- Node JS Community es una red inclusiva de desarrolladores que se reúnen para discutir, aprender y contribuir a los diversos proyectos. Esta comunidad también está activa en GitHub.
- La plataforma Node JS admite el desarrollo de la API de servicios web RESTful con mucha facilidad.
- Los módulos de Node JS se almacenan en caché una vez que se cargan en la aplicación por primera vez. Es posible que varias llamadas para cargar un módulo no hagan que el código del módulo se ejecute varias veces.
- En la aplicación Node JS, necesitamos escribir código en Java Script. Hoy en día tenemos muchos marcos y herramientas de pruebas unitarias de Java Script como Jasmin Framework [42].

Desventajas

- No es compatible con la programación de subprocessos múltiples, esta es la razón por la que solo se prefiere para las aplicaciones livianas (juegos en línea, chat). Si lo está utilizando para aplicaciones pesadas, su elección es incorrecta.

- Tiene algunos problemas de coherencia. La mayoría de las veces, la nueva API viene con muchos cambios anteriores, luego los programadores deben realizar cambios en el código para que sea compatible.
- El bucle de eventos es la característica principal de Node.js, y es posible que se estén ejecutando demasiadas devoluciones de llamada en su código. Por lo tanto, Node JS tampoco es adecuado para tareas que requieren un uso intensivo de la CPU, si está haciendo eso, podría ralentizar su aplicación.
- La mayoría de las veces, las bases de datos relacionales se comportan de manera extraña al trabajar con el Node JS [42].

Características	Laravel	ASP.NET Core	Node JS
Creador	Taylor Otwell	Microsoft	Ryan Dahl
Lenguaje	PHP	C#	JavaScript
Integración directa con servicios mail	Si	No	Si
Promotor de plantillas	Si	No	No
Rendimiento	Si	Si	Si
Seguridad	Si	No	Si
Comunidad	Alta	Alta	Alta
Soporte y ayuda	Si	Si	Si

Tabla 3.11.- Cuadro comparativo entre Laravel, ASP.NET Core y Node JS.
Elaborado por: El investigador.

3.2.3.4 Tecnología de Desarrollo Back End Seleccionada

En base al estudio realizado entre las tecnologías Back-End y el cuadro comparativo mencionado anteriormente, el framework que mejor se adapta para el objetivo de proyecto es Laravel, por las ventajas que ofrece, tales como las notificaciones que se van a utilizar para emitir mensajes para los trámites de los estudiantes. Además, se puede deducir mediante la tabla anterior, que PHP tiene más ventajas sobre .NET y Node JS, por lo cual, se opta por utilizar la tecnología Laravel [38].

API (Interfaz de Programación de Aplicaciones)

Es el conjunto de protocolos, rutinas, funciones y/o comandos que los programadores utilizan para facilitar la interacción entre distintos servicios de software, los cuales funcionan con cualquier lenguaje de programación común, el enfoque más popular para entregar API web es REST (Transferencia de Estado Representacional), además permite que un servicio de software acceda a los datos de otro servicio de software

sin que el desarrollador necesite saber cómo funciona el otro servicio. Una API puede verse compuesta por dos elementos fundamentales: una especificación técnica que establece cómo se puede intercambiar información entre programas (que a su vez se compone de protocolos de solicitud de procesamiento y entrega de datos) y una interfaz de software que de alguna manera pública esa especificación [43].

API REST

Es una interfaz de programación de aplicaciones (API o API web), en donde el cliente envía una solicitud y el servidor devuelve una respuesta al cliente mediante el uso de recursos, dicha información se entrega por medio de HTTP (Hypertext Transfer Protocol) en uno de estos formatos: JSON (Notación de objeto de JavaScript), HTML (HyperText Markup Language), entre otros. En otras palabras, REST es un estilo arquitectónico para desarrollar servicios web [45]. Rest usa http para definir la acción deseada. Para permitir las comunicaciones, hay métodos de recursos o tipos de solicitudes disponibles, los cuales son:

- GET: Leer datos, recuperar información.
- POST: Insertar datos o un nuevo recurso.
- PUT: Actualiza un recurso existente.
- DELETE: Elimina un recurso existente [46].

API Google Drive

Google Drive es un servicio de alojamiento de archivos en la nube, que fue desarrollado y creado por Google. Permite subir información de todo tipo de archivos, además de que tiene la opción de poder compartir con diferentes usuarios.

Al utilizar la API de Google Drive le permite crear aplicaciones que aprovechen las características de tener un almacenamiento en la nube y que se integren con todas las herramientas que este servicio ofrece. Por lo que, en el caso del sistema web integrado, sirve para almacenar todos los documentos que se vayan generando a través del tiempo y a su vez, utilizar las herramientas de documentos de Google, para la edición de plantillas y creación de nuevos documentos [44].

Además, permite gestionar de manera ordenada la información de los diferentes documentos, ya que se pueden crear carpetas y clasificar los diferentes trámites

académicos. Teniendo en cuenta también, la opción de poder compartir documentos a diferentes usuarios.

Gracias a la integración de la interfaz de usuario de Google Drive, se puede interactuar con los archivos de Drive, lo que permite, almacenar y editar los diferentes documentos que se vayan ingresando en el sistema [44].

Base de datos MySQL

Dado que MySQL es uno de los sistemas de gestión de base de datos más populares para aplicaciones web basadas en web y por su gran compatibilidad para trabajar con PHP, además, de ser gratuito y que está constantemente actualizando nuevas características y medidas seguridad [47].

Según las características mencionadas anteriormente, se decidió trabajar con este gestor de base de datos MySQL para el desarrollo de la presente investigación. Además, porque tiene compatibilidad para trabajar con Laravel, ya que este puede crear la base de datos mediante código y al correr migraciones se tiene la estructura de la base de datos.

Arquitectura en Capas

La arquitectura en capas, hoy en día, es la más utilizada por los desarrolladores, donde una capa representa una unidad propia con sus objetivos funcionales, sus dependencias y sus colaboraciones hacia el sistema. Se pueden encontrar arquitecturas en capas de dos capas, tres capas y n capas [48].

Arquitectura tres capas

Las aplicaciones web se modelan mediante capas, donde una capa representa un elemento que procesa o trata información [49].

Modelo de tres Capas

Es un modelo que supera las limitaciones de las arquitecturas ajustadas al modelo de dos capas, la cual introduce una capa intermedia llamada capa de proceso entre la presentación y los datos, esta capa intermedia centraliza la lógica de negocio, haciendo la administración más sencilla, los datos se pueden integrar de múltiples fuentes, las aplicaciones web actuales se ajustan a este modelo [50]. Las capas de estos modelos son:

- Capa de presentación (inicia en el cliente y en el servidor).
- Capa de proceso (servidor web).
- Capa de datos (servidor de datos).

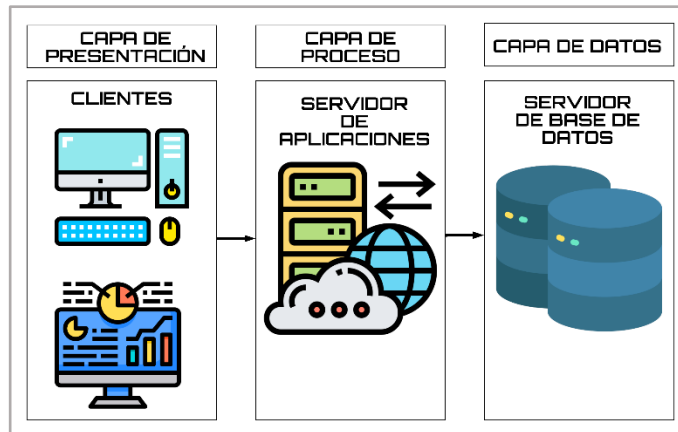


Figura 3.11.- Estructura del proyecto web.
Elaborado por: El investigador.

Protocolos de comunicación en la arquitectura cliente servidor

Para que el cliente establezca una comunicación exitosa con el servidor, se utilizan diferentes protocolos de red, según la naturaleza de los datos que se han de intercambiar [51].

Hypertext Transfer Protocol (HTTP)

Es el protocolo principal utilizado por la World Wide Web (WWW) para la comunicación, el cual se basa en el modelo cliente servidor y se encarga de definir el formato y la manera de transmisión de los mensajes desde que el cliente efectúa la petición y se devuelve la respuesta por parte del servidor, donde un navegador funciona como cliente HTTP, cuando se usa para enviar solicitudes al servidor y es el servidor quien envía la respuesta de la información [51].

Estructura del proyecto

En la figura 3.12 se muestra la estructura del proyecto, donde se identifica que el cliente está conformado por la tecnología React JS esta ocupa la biblioteca Material UI para las interfaces, en el cual el cliente envía la solicitud y mediante las API REST se consumen en el servidor que va a estar compuesto por Laravel, utilizando la base de datos MySQL y la API de Google Drive.

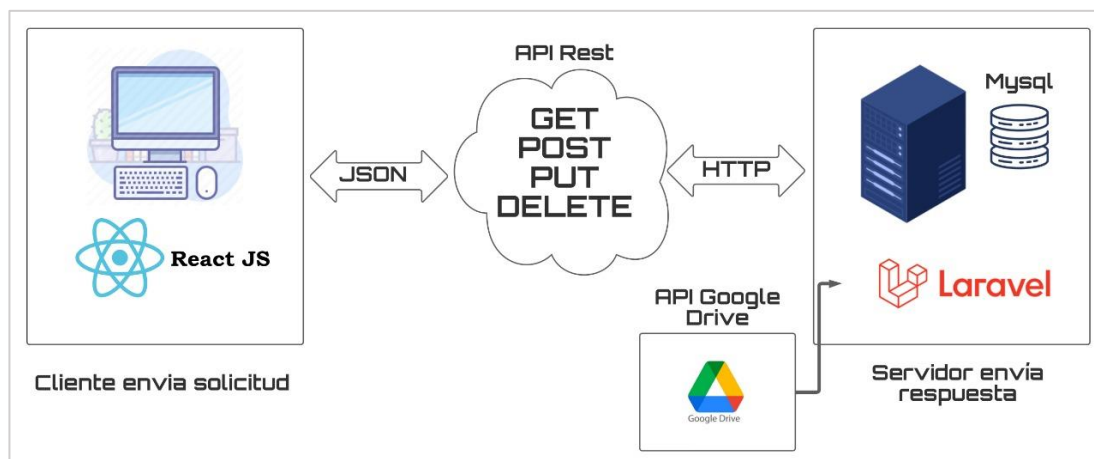


Figura 3.12.- Estructura del proyecto web.
Elaborado por: El investigador.

3.2.4 Implantación del sistema web integrado para la gestión de trámites académico-administrativos en la Facultad de Ingeniería en Sistemas, Electrónica e Industrial

La implantación del sistema web integrado en la gestión de trámites académico-administrativos se basa en cuatro fases, fase de exploración, fase de planificación del proyecto, la fase de iteración y la fase de pruebas, las cuales se detallan a continuación.

Fase 1: Exploración:

Historias de usuario

En la metodología de desarrollo ágil, las historias de usuario permiten definir las funcionalidades del sistema acorde a los requerimientos que fueron planteados, las mismas que se dividirán en actividades que se detallan en cada Historia de usuario que se genera.

Historia de usuario	
Número:	Usuario:
Nombre historia:	
Prioridad en negocio:	Riesgo en desarrollo:
Puntos estimados:	Iteración Asignada:
Programador responsable:	
Descripción:	
Observación:	

Tabla 3.12.-Modelo historia de usuario.
Elaborado por: El investigador.

Descripción de la Historia de usuario

En la tabla 3.13 se describe los elementos de la plantilla de Historia de usuario.

Historia de usuario	
Número: Identificador de cada historia de usuario.	Usuario: Persona(s) que se le(s) asigna la historia de usuario.
Nombre historia: Título asignado a la historia de usuario.	
Prioridad en negocio: Cuantificador asignado, según la necesidad del negocio y se clasifica: Alto, Media y Baja.	Riesgo en desarrollo: Cuantificador asignado, según el impacto de desarrollo en caso de falla y se clasifica: Alto, Media y Baja.
Puntos estimados: Cantidad de días estimados para el desarrollo de la historia de usuario.	Iteración Asignada: Identificación en el que se designa la historia de usuario.
Programador responsable: Persona encargada de la historia de usuario.	
Descripción: Información en lenguaje natural para mejorar en entendimiento la historia de usuario.	
Observación: Campo opcional, en el cual se aclara información en caso de ser requerido.	

Tabla 3.13.-Descripción de la historia de usuario.

Elaborado por: El investigador.

Historia de usuario	
Número: 1	Usuario: Desarrollador
Nombre historia: Determinar la estructura del proyecto.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alto
Puntos estimados: 3	Iteración Asignada: 1
Programador responsable: Guillermo Barcia	
Descripción: Estructurar la arquitectura del proyecto, tanto en Back End y Front End del desarrollo de sistema web.	
Observación: Determinar la arquitectura del proyecto y tecnologías necesarias para el desarrollo.	

Tabla 3.14.-Historia de usuario 1.

Elaborado por: El investigador.

Historia de usuario	
Número: 2	Usuario: Desarrollador
Nombre historia: Análisis y diseño de la base de datos.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alto
Puntos estimados: 4	Iteración Asignada: 1
Programador responsable: Guillermo Barcia	

Descripción: Analizar cómo se encuentra la base de datos actualmente y realizar cambios pertinentes en el diagrama entidad relación, para el desarrollo del sistema web integrado.
Observación: Se requiere hacer mejoras y agregar nuevas tablas para las funcionalidades del aplicativo web.

Tabla 3.15.-Historia de usuario 2.
Elaborado por: El investigador.

Historia de usuario	
Número: 3	Usuario: Usuario Final
Nombre historia: Diseño de interfaces.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alto
Puntos estimados: 5	Iteración Asignada: 1
Programador responsable: Guillermo Barcia	
Descripción: Diseñar nuevas interfaces, con las nuevas tecnologías que se vayan a utilizar en el proyecto, con características responsivas, que permitan manejar el sistema web desde un ordenador.	
Observación: Las interfaces deben ser amigables e intuitivas para el usuario.	

Tabla 3.16.-Historia de usuario 3.
Elaborado por: El investigador.

Historia de usuario	
Número: 4	Usuario: Usuario Final
Nombre historia: Formulario de ingreso al sistema web integrado.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alto
Puntos estimados: 3	Iteración Asignada: 2
Programador responsable: Guillermo Barcia	
Descripción: Diseñar la interfaz del formulario de ingreso, en donde el usuario pueda ingresar al sistema, mediante un correo electrónico y su respectiva contraseña.	
Observación: Los usuarios se autenticarán con un correo institucional.	

Tabla 3.17.-Historia de usuario 4.
Elaborado por: El investigador.

Historia de usuario	
Número: 5	Usuario: Usuario Final
Nombre historia: Formulario estudiantes.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alto
Puntos estimados: 2	Iteración Asignada: 2
Programador responsable: Guillermo Barcia	
Descripción: El usuario podrá cargar los estudiantes, mediante un archivo Excel, pero también puede hacer una carga manual, ingresando los respectivos datos del estudiante.	
Observación: Contiene un formato de ayuda para subir el archivo requerido.	

Tabla 3.18.-Historia de usuario 5.
Elaborado por: El investigador.

Historia de usuario	
Número: 6	Usuario: Usuario Final
Nombre historia: Formulario docentes.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alto
Puntos estimados: 2	Iteración Asignada: 2
Programador responsable: Guillermo Barcia	
Descripción: El usuario podrá cargar los datos, de forma manual ingresando los respectivos datos del docente.	
Observación: Al momento de crear un docente se debe crear con un formato de Ing. Juan Pérez.	

Tabla 3.19.-Historia de usuario 6.
Elaborado por: El investigador.

Historia de usuario	
Número: 7	Usuario: Usuario Final
Nombre historia: Información Estudiantes.	
Prioridad en negocio: Media	Riesgo en desarrollo: Alto
Puntos estimados: 1	Iteración Asignada: 2
Programador responsable: Guillermo Barcia	
Descripción: El usuario podrá visualizar los datos de un estudiante, los cuales son: cédula, nombres, correos, celular, matrícula, folio. Además, podrá hacer una búsqueda por los campos cédula, nombres, matrícula y folio.	
Observación: Dentro de la lista donde se puede ver los estudiantes, se puede hacer una edición de los datos.	

Tabla 3.20.-Historia de usuario 7.
Elaborado por: El investigador.

Historia de usuario	
Número: 8	Usuario: Usuario Final
Nombre historia: Información Docentes.	
Prioridad en negocio: Media	Riesgo en desarrollo: Alto
Puntos estimados: 1	Iteración Asignada: 2
Programador responsable: Guillermo Barcia	
Descripción: El usuario podrá visualizar los datos de un docente, los cuales son: cédula, nombres, correo y celular. Además, podrá hacer una búsqueda por los campos cédula, nombres, correo y celular.	
Observación: Dentro de la lista donde se puede ver los docentes, se puede hacer una edición de los datos.	

Tabla 3.21.-Historia de usuario 8.

Elaborado por: El investigador.

Historia de usuario	
Número: 9	Usuario: Usuario Final
Nombre historia: Visualizar carreras universitarias.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Bajo
Puntos estimados: 2	Iteración Asignada: 2
Programador responsable: Guillermo Barcia	
Descripción: El usuario podrá visualizar todas las carreras universitarias.	
Observación: Se reflejará las carreras universitarias en una tabla.	

Tabla 3.22.-Historia de usuario 9.

Elaborado por: El investigador.

Historia de usuario	
Número: 10	Usuario: Usuario Final
Nombre historia: Formulario carreras universitarias.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alto
Puntos estimados: 3	Iteración Asignada: 2
Programador responsable: Guillermo Barcia	
Descripción: El usuario podrá editar carreras universitarias e inhabilitar.	
Observación: Sin observación.	

Tabla 3.23.-Historia de usuario 10.

Elaborado por: El investigador.

Historia de usuario	
Número: 11	Usuario: Usuario Final
Nombre historia: Gestión de perfil de usuario.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alto
Puntos estimados: 3	Iteración Asignada: 2
Programador responsable: Guillermo Barcia	
Descripción: El usuario podrá modificar su contraseña, ingresando su contraseña actual y la nueva.	
Observación: Para cada actualización e ingreso de un registro debe existir la validación de datos.	

Tabla 3.24.-Historia de usuario 11.

Elaborado por: El investigador.

Historia de usuario	
Número: 12	Usuario: Usuario Final
Nombre historia: Gestión de usuarios y roles.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alto
Puntos estimados: 3	Iteración Asignada: 2
Programador responsable: Guillermo Barcia	
Descripción: Se podrá visualizar los usuarios registrados, en los cuales puede actualizar y dar de baja a un usuario. Hay que tener en cuenta que, el sistema debe contar con 4 roles:	
Rol Administrador: Permite realizar todas las acciones del sistema.	
Rol Administrador plantillas: Permite realizar todas las acciones del sistema, excepto crear usuarios.	
Rol Escritor: Solo permite crear documentos, editar y ver los tramites académicos.	
Rol Lectura: Solo puede visualizar los documentos generados.	
También se tiene la opción de que el usuario pueda agregar a qué módulos tiene acceso, ya que al ser un sistema web integrado esta subclasificado por los módulos de: Secretaría de Facultad, Secretaría de Consejo Académico de Facultad y Secretaría de Titulación.	
Observación: Para cada actualización e ingreso de un registro debe existir la validación de datos.	

Tabla 3.25.-Historia de usuario 12.

Elaborado por: El investigador.

Historia de usuario	
Número: 13	Usuario: Usuario Final
Nombre historia: Visualizar procesos académicos.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Bajo
Puntos estimados: 2	Iteración Asignada: 3
Programador responsable: Guillermo Barcia	

Descripción: El usuario podrá visualizar todos los procesos académicos que existen, para la clasificación de cada uno de los trámites académicos que existan.
Observación: Se puede crear, editar y dar de baja los procesos académicos.

Tabla 3.26.-Historia de usuario 13.

Elaborado por: El investigador.

Historia de usuario	
Número: 14	Usuario: Usuario Final
Nombre historia: Formulario de crear procesos académicos.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alto
Puntos estimados: 3	Iteración Asignada: 3
Programador responsable: Guillermo Barcia	
Descripción: El usuario podrá crear nuevos procesos académicos en los cuales, dentro de esos procesos académicos, se pueden crear los documentos respectivos de los diferentes tramites académicos que existan. En este caso, se denominan plantillas.	
Observación: Los procesos académicos, hace referencia a que sección pertenece cada plantilla, estos procesos pueden ser: titulación, vinculación, etc.	

Tabla 3.27.-Historia de usuario 14.

Elaborado por: El investigador.

Historia de usuario	
Número: 15	Usuario: Usuario Final
Nombre historia: Formulario de editar procesos académicos.	
Prioridad en negocio: Media	Riesgo en desarrollo: Alto
Puntos estimados: 1	Iteración Asignada: 3
Programador responsable: Guillermo Barcia	
Descripción: El usuario podrá editar los procesos académicos existentes, en el cual, tiene la opción de inhabilitar dicho proceso y de modificar el nombre.	
Observación: Se deben llenar los campos requeridos.	

Tabla 3.28.- Historia de usuario 15.

Elaborado por: El investigador.

Historia de usuario	
Número: 16	Usuario: Usuario Final
Nombre historia: Visualización de plantillas.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Bajo
Puntos estimados: 3	Iteración Asignada: 3

Programador responsable: Guillermo Barcia
Descripción: El usuario podrá visualizar todas las plantillas, que se encuentren creado dentro de ese proceso académico.
Observación: Las plantillas serán almacenadas en Google Drive. Además, de que se podrá crear, editar y dar de baja las plantillas.

Tabla 3.29.- Historia de usuario 16.
Elaborado por: El investigador.

Historia de usuario	
Número: 17	Usuario: Usuario Final
Nombre historia: Formulario de crear plantillas.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alto
Puntos estimados: 3	Iteración Asignada: 3
Programador responsable: Guillermo Barcia	
Descripción: El usuario podrá crear plantillas con su respectivo nombre, en los cuales pueden modificar y a su vez inactivar la plantilla, para poder pasar a otro proceso en el que pertenezca. Además, de que las plantillas pueden cambiarse de un proceso a otro.	
Observación: Una vez finalizada la creación de la plantilla, se abre el editor de texto de Google Drive el cual, contendrá la plantilla complementada con los datos enviados en el formulario. En esta sección se puede editar el texto de los documentos.	

Tabla 3.30.-Historia de usuario 17.
Elaborado por: El investigador.

Historia de usuario	
Número: 18	Usuario: Usuario Final
Nombre historia: Formulario de editar plantillas.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alto
Puntos estimados: 2	Iteración Asignada: 3
Programador responsable: Guillermo Barcia	
Descripción: El usuario podrá modificar y a su vez inactivar la plantilla, para poder pasar a otro proceso académico en el que pertenezca. Además, el usuario puede modificar el texto de la plantilla, teniendo en cuenta que el guardado es automático.	
Observación: Las plantillas pueden cambiarse de un proceso académico a otro.	

Tabla 3.31.- Historia de usuario 18.
Elaborado por: El investigador.

Historia de usuario	
Número: 19	Usuario: Usuario Final
Nombre historia: Formulario de plantilla de Word.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alto
Puntos estimados: 2	Iteración Asignada: 3
Programador responsable: Guillermo Barcia	
Descripción: El usuario podrá abrir la plantilla en donde, se abre un Word para poder editar y estructurar la plantilla de resolución o acuerdo.	
Observación: Las plantillas se guardan automáticamente ya que se encuentran almacenadas en la nube.	

Tabla 3.32.-Historia de usuario 19.
Elaborado por: El investigador.

Historia de usuario	
Número: 20	Usuario: Usuario Final
Nombre historia: Visualización de Consejos.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Bajo
Puntos estimados: 2	Iteración Asignada: 4
Programador responsable: Guillermo Barcia	
Descripción: Estos Consejos están divididos en 4 secciones: Consejo Directivo de Facultad, Consejo Académico, Consejo de Titulación y Consejo de Integración Curricular. El usuario podrá visualizar los Consejos que se encuentran creados, cerrados y activos, dependiendo en que sección del sistema se encuentre. Además, podrá crear, editar, borrar el Consejo directivo, generar asistencia del responsable o presidente y de los docentes, generar acta, ver resoluciones y cerrar el Consejo.	
Observación: Dentro de un Consejo, se generan los documentos de todos los tramites académicos que se generaron en dicho Consejo, para luego poder generar el acta.	

Tabla 3.33.-Historia de usuario 20.
Elaborado por: El investigador.

Historia de usuario	
Número: 21	Usuario: Usuario Final
Nombre historia: Crear Consejos.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alto
Puntos estimados: 2	Iteración Asignada: 4
Programador responsable: Guillermo Barcia	
Descripción: El usuario podrá crear el Consejo dependiendo en que sección del sistema se	

<p>encuentre (Facultad, Académico, Titulación o Integración Curricular), para ello debe llenar los siguientes campos del formulario:</p> <ul style="list-style-type: none"> • Fecha de Consejo (Facultad, Académico, Titulación o Integración Curricular). • Tipo de sesión: Ordinaria o Extraordinaria. • Nombre Consejo (Facultad, Académico, Titulación o Integración Curricular).
<p>Observación: Todos los campos son obligatorios.</p>

Tabla 3.34.-Historia de usuario 21.
Elaborado por: El investigador.

Historia de usuario	
Número: 22	Usuario: Usuario Final
Nombre historia: Modificar Consejos.	
Prioridad en negocio: Media	Riesgo en desarrollo: Alto
Puntos estimados: 1	Iteración Asignada: 4
Programador responsable: Guillermo Barcia	
Descripción: El usuario podrá modificar Consejo dependiendo en que sección del sistema se encuentre (Facultad, Académico, Titulación o Integración Curricular), los campos editables son nombre, fecha y tipo de sesión.	
Observación: Todos los campos son obligatorios.	

Tabla 3.35.- Historia de usuario 22.
Elaborado por: El investigador.

Historia de usuario	
Número: 23	Usuario: Usuario Final
Nombre historia: Eliminar Consejos.	
Prioridad en negocio: Media	Riesgo en desarrollo: Alto
Puntos estimados: 1	Iteración Asignada: 4
Programador responsable: Guillermo Barcia	
Descripción: El usuario podrá eliminar un Consejo dependiendo en que sección del sistema se encuentre (Facultad, Académico, Titulación o Integración Curricular), siempre y cuando no existan documentos (resoluciones o acuerdos) creados de por medio.	
Observación: Mensaje de alerta para confirmar la acción.	

Tabla 3.36.- Historia de usuario 23.
Elaborado por: El investigador.

Historia de usuario	
Número: 24	Usuario: Usuario Final
Nombre historia: Formulario de miembros o representantes de los Consejos.	
Prioridad en negocio: Media	Riesgo en desarrollo: Alto
Puntos estimados: 3	Iteración Asignada: 4
Programador responsable: Guillermo Barcia	
Descripción: El usuario podrá designar los miembros y representantes de dicho Consejo creado, dependiendo en que sección del sistema se encuentre (Facultad, Académico, Titulación o Integración Curricular), para verificar la asistencia y notificación de dichos miembros al correo electrónico.	
Observación: Debe ingresar en la opción de asistencia - miembros para poder agregar.	

Tabla 3.37.-Historia de usuario 24.

Elaborado por: El investigador.

Historia de usuario	
Número: 25	Usuario: Usuario Final
Nombre historia: Formulario de números reservados de documentación.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alto
Puntos estimados: 3	Iteración Asignada: 4
Programador responsable: Guillermo Barcia	
Descripción: El usuario podrá reservar diferentes números de resoluciones, posterior al último asignado al documento por crear. De manera tal, que quedan números reservados para cierto Consejo, dependiendo en que sección del sistema se encuentre (Facultad, Académico, Titulación o Integración Curricular).	
Observación: Los números de documentos reservados, se pueden ver a que Consejo directivo pertenece al momento de que se vaya a crear un documento. Además, al momento de crear un documento, con un número reservado que no pertenezca a ese Consejo, se emitirá un mensaje de alerta.	

Tabla 3.38.-Historia de usuario 25.

Elaborado por: El investigador.

Historia de usuario	
Número: 26	Usuario: Usuario Final
Nombre historia: Crear Documentos – Resolución o Acuerdo.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alto
Puntos estimados: 4	Iteración Asignada: 4
Programador responsable: Guillermo Barcia	
<p>Descripción: El usuario deberá llenar los campos del formulario, el cual, permite seleccionar el proceso y la plantilla a la que pertenece en dicho proceso.</p> <p>Al momento de crear un documento, retorna un campo autonumérico que es el último valor del documento disponible. Además, puede revisar si se dicho número se encuentra reservado.</p> <p>Contiene un campo, para seleccionar el Consejo que pertenece dicho documento, dependiendo en que sección del sistema se encuentre (Facultad, Académico, Titulación o Integración Curricular).</p> <p>Contiene un campo, que filtra por número de cédula, nombres, folio o matrícula del estudiante para pueda escoger el usuario de manera más rápida y sencilla, además de agregar uno o varios docentes. También con un campo, denominado comentario para cualquier observación.</p>	
<p>Observación: Es necesario llenar todos los campos requeridos a excepción del campo estudiante, docente o comentario. Además, tiene la opción de revisar los números reservados o encolados que tenga en el sistema. El número celular contiene un enlace que permite dirigirse al API de WhatsApp.</p>	

Tabla 3.39.- Historia de usuario 26.

Elaborado por: El investigador.

Historia de usuario	
Número: 27	Usuario: Usuario Final
Nombre historia: Editar Documentos – Resolución o Acuerdo.	
Prioridad en negocio: Media	Riesgo en desarrollo: Alto
Puntos estimados: 4	Iteración Asignada: 4
Programador responsable: Guillermo Barcia	
<p>Descripción: El usuario podrá editar el texto del documento creado.</p>	
<p>Observación: La actualización es automática por lo que trabaja con Google Drive.</p>	

Tabla 3.40.- Historia de usuario 27.

Elaborado por: El investigador.

Historia de usuario	
Número: 28	Usuario: Usuario Final
Nombre historia: Notificación de trámites académicos (Resoluciones o Acuerdos).	
Prioridad en negocio: Media	Riesgo en desarrollo: Alto
Puntos estimados: 4	Iteración Asignada: 4
Programador responsable: Guillermo Barcia	
Descripción: El usuario podrá notificar al estudiante que el documento ya ha sido generado. Dicha notificación, le llega al estudiante mediante el correo electrónico o al WhatsApp del número celular que tenga ingresado.	
Observación: El correo donde se envía la notificación va a ser el institucional del estudiante.	

Tabla 3.41.-Historia de usuario 28.
Elaborado por: El investigador.

Historia de usuario	
Número: 29	Usuario: Usuario Final
Nombre historia: Generar Acta de Consejos.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alto
Puntos estimados: 4	Iteración Asignada: 4
Programador responsable: Guillermo Barcia	
Descripción: El usuario podrá generar el acta del Consejo que fue creado en esa fecha, dependiendo en que sección del sistema se encuentre (Facultad, Académico, Titulación o Integración Curricular). Se divide en dos partes, Procesar Documentos: donde que une todas las resoluciones que hayan sido creado en dicho Consejo, la segunda parte hace referencia a generar plantilla acta, en donde se reemplaza los valores del acta y donde luego se copian todos los documentos que han sido procesados al acta.	
Observación: Dicha acta, es un resumen de todos los tramites académicos que se realizaron en el Consejo Directivo de Facultad. El texto de cada uno de los documentos generados, deben estar entre los comandos {{FROM}} y {{TO}} , para la unificación de todos los documentos y la copia de todos los documentos de texto dinámico, es decir que, se copian tablas y el formato del texto de la plantilla. Dicho documento tiene la opción de descargar, para luego copiar al Documento de Google, donde se encuentra el Acta.	

Tabla 3.42.-Historia de usuario 29.
Elaborado por: El investigador.

Historia de usuario	
Número: 30	Usuario: Usuario Final
Nombre historia: Cerrar Consejos.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alto
Puntos estimados: 4	Iteración Asignada: 5
Programador responsable: Guillermo Barcia	
Descripción: El usuario podrá cerrar el Consejo.	
Observación: Dicho Consejo ya sea de Facultad, Académico, Titulación o Integración Curricular, va a ser cerrado siempre y cuando se haya generado un acta, teniendo en cuenta que, una vez cerrada el Consejo, ya no se poder realizar ningún cambio, ni volver a generar actas, solo se podrá, visualizar la Acta creada.	

Tabla 3.43.-Historia de usuario 30.
Elaborado por: El investigador.

Fase 2: Planificación del proyecto

El proyecto se enfoca en la creación de un sistema web integrado, que cumpla con los objetivos establecidos, por lo que se decide una jornada de trabajo diario de 4 horas dedicadas, entre lunes a viernes y 8 horas dedicadas los sábados en el desarrollo de las historias de usuario. De forma que, se plantea la estimación de esfuerzo para cada historia expuesta a continuación:

N°	Historia de usuario	Tiempo Estimado	
		Días	Horas
1	Determinar la estructura del proyecto	2	8
2	Análisis y diseño de la base de datos	5	20
3	Diseño de interfaces	3	12
4	Formulario de ingreso al sistema web integrado	3	12
5	Formularios estudiantes	2	8
6	Formularios docentes	2	8
7	Información Estudiantes	1	4
8	Información Docentes	1	4
9	Visualizar carreras universitarias	1	4
10	Formulario carreras universitarias	3	12
11	Gestión de perfil de usuario	1	4
12	Gestión de usuarios y roles	5	20
13	Visualizar procesos académicos	1	4
14	Formulario de crear procesos académicos	2	8
15	Formulario de editar procesos académicos	1	4
16	Visualización de plantillas	1	4
17	Formulario de crear plantillas	3	12
18	Formulario de editar plantillas	1	4

19	Formulario de plantilla de Word	2	8
20	Visualización de Consejos	2	8
21	Crear Consejos	2	8
22	Modificar Consejos	1	4
23	Eliminar Consejos	1	4
24	Formulario de miembros o representantes de los Consejos	2	8
25	Formulario de números reservados de documentación	2	8
26	Crear documentos – Resolución o Acuerdo.	5	20
27	Editar documentos– Resolución o Acuerdo.	2	8
28	Notificación de tramites académicos (Resoluciones o Acuerdos).	2	8
29	Generar Acta de los Consejos	5	20
30	Cerrar Consejos	2	8
Tiempo estimado total		66	174

Tabla 3.44.- Estimación de historias de usuario.

Elaborado por: El investigador.

Una vez realizado la estimación de historias de usuario, se procede a plantear la realización del plan de entregas tomando en cuenta las iteraciones que se propone, que se detalla a continuación:

N° Iteración	Historia de usuario	Tiempo Estimado	
		Días	Horas
1	H01	2	8
	H02	5	20
	H03	4	16
2	H04	2	8
	H05	3	12
	H06	2	8
	H07	1	4
	H08	1	4
	H09	1	4
	H10	2	8
	H11	1	4
	H12	5	20
3	H13	1	4
	H14	2	8
	H15	1	4
	H16	1	4
	H17	3	12
	H18	1	4
	H19	2	8
4	H20	6	24

	H21	4	16
	H22	4	16
	H23	4	16
	H24	1	4
	H25	1	4
	H26	3	12
	H27	1	4
	H28	1	4
	H29	2	8
	H30	1	4
Tiempo estimado total		68	272

Tabla 3.45.-Plan de entregas.
Elaborado por: El investigador.

Definición de equipos y roles

En esta sección se especifica los roles en el desarrollo de la metodología en esencial, ya que, permite tener una mejor gestión con las personas que están involucradas en el proyecto y con la finalidad de cumplir la planificación del proyecto.

Metodología	Miembro	Rol
XP (Extreme Programming)	Guillermo Barcia Gallardo	Desarrollador
	Entrevistados	Clientes
	Entrevistados, Guillermo Barcia	Encargado de pruebas (Tester)
	Ing. Clay Aldas	Entrenador Coach
	Guillermo Barcia, Ing. Clay Aldas. Entrevistado	Encargado de seguimiento
	Guillermo Barcia	Consultor

Tabla 3.46.-Equipo y roles.
Elaborado por: El investigador.

Tareas

Para el desarrollo de las historias de usuario, se hace el uso de tareas del método XP, que representa un conjunto de actividades con una fecha de inicio y fin, mismas que son asignadas a un director que es el responsable de completar la tarea dentro del tiempo estimado.

Tareas: Determinar la estructura del proyecto - H01

Tarea	Responsable: Guillermo Barcia
N° Tarea: T1	Código de Historia: H01
Tipo de Tarea: Desarrollo	Puntos Estimados: 3
Nombre de Tarea: Determinar la estructura del proyecto.	
Fecha inicio: 03/03/2022	Fecha fin: 04/03/2022
Descripción: Se analizará las diferentes tecnologías que existen actualmente, con la finalidad de poder escoger la mejor herramienta para trabajar y de igual manera la arquitectura con la que se va a realizar el sistema web integrado.	

Tabla 3.47.- Tarea-determinar la estructura del proyecto.

Elaborado por: El investigador.

Tareas: Análisis de la base de datos - H02

Tarea	Responsable: Guillermo Barcia
N° Tarea: T2	Código de Historia: H02
Tipo de Tarea: Desarrollo	Puntos Estimados: 4
Nombre de Tarea: Definición y modelado de la base de datos.	
Fecha inicio: 07/03/2022	Fecha fin: 11/03/2022
Descripción: Se analizará detenidamente las funcionalidades del sistema y los requerimientos del usuario, previo al modelado de la base de datos donde se establecerá las entidades necesarias, para poder generar el modelo entidad relación.	

Tabla 3.48.- Tarea- definición y modelado de la base de datos.

Elaborado por: El investigador.

Tareas: Diseño de interfaces - H03

Tarea	Responsable: Guillermo Barcia
N° Tarea: T3	Código de Historia: H03
Tipo de Tarea: Desarrollo	Puntos Estimados: 5
Nombre de Tarea: Definir plantilla para el diseño de las interfaces del sistema.	
Fecha inicio: 14/03/2022	Fecha fin: 15/03/2022
Descripción: Se escogió trabajar con la librería de React denominada Mui, que proporciona una biblioteca robusta, personalizable y accesible de componentes básicos y avanzados, que permite construir un diseño más rápido	

Tabla 3.49.-Tarea- definir plantilla para el diseño de las interfaces del sistema.

Elaborado por: El investigador.

Tarea	Responsable: Guillermo Barcia
N° Tarea: T4	Código de Historia: H03
Tipo de Tarea: Desarrollo	Puntos Estimados: 5
Nombre de Tarea: Diseño de menú principal.	
Fecha inicio: 16/03/2022	Fecha fin: 17/03/2022
Descripción: Presentar un diseño intuitivo, en el que el usuario pueda acceder a los diferentes módulos tales como: inicio de sesión, formulario carreras, formulario estudiantes y docentes, procesos académicos, roles de usuario, perfil, Consejos directivos y documentos.	

Tabla 3.50.-Tarea- diseño de menú principal.

Elaborado por: El investigador.

Tareas: Formulario de ingreso al sistema web integrado - H04

Tarea	Responsable: Guillermo Barcia
N° Tarea: T5	Código de Historia: H04
Tipo de Tarea: Desarrollo	Puntos Estimados: 3
Nombre de Tarea: Diseño de formulario de ingreso.	
Fecha inicio: 18/03/2022	Fecha fin: 18/03/2022
Descripción: Mostrar la interfaz, en donde el usuario procede a ingresar al sistema, mediante un correo electrónico y su respectiva contraseña.	

Tabla 3.51.-Tarea-diseño de formulario de ingreso.

Elaborado por: El investigador.

Tarea	Responsable: Guillermo Barcia
N° Tarea: T6	Código de Historia: H04
Tipo de Tarea: Desarrollo	Puntos Estimados: 3
Nombre de Tarea: Codificación para validar las credenciales de ingreso.	
Fecha inicio: 21/03/2022	Fecha fin: 21/03/2022
Descripción: Si se ingresó las credenciales correctamente, se procederá a ingresar al menú principal, caso contrario, se mostrará un mensaje de alerta que el usuario o contraseña han sido incorrectos.	

Tabla 3.52.- Tarea-codificación para validar las credenciales de ingreso.

Elaborado por: El investigador.

Tareas: Formulario estudiantes - H05

Tarea	Responsable: Guillermo Barcia
N° Tarea: T7	Código de Historia: H05
Tipo de Tarea: Desarrollo	Puntos Estimados: 3
Nombre de Tarea: Codificación agregar estudiantes por archivo Excel.	
Fecha inicio: 22/03/2022	Fecha fin: 22/03/2022
Descripción: Desarrollar un método que permita subir los datos del estudiante mediante un Excel, en el cual, debe tener una estructura constituida por: cédula, apellidos, nombres, teléfono, celular, correo, correo institucional, matrícula y folio.	

Tabla 3.53.-Tarea-codificación agregar estudiantes por archivo Excel.**Elaborado por:** El investigador.

Tarea	Responsable: Guillermo Barcia
N° Tarea: T8	Código de Historia: H05
Tipo de Tarea: Desarrollo	Puntos Estimados: 2
Nombre de Tarea: Codificación agregar estudiantes manualmente.	
Fecha inicio: 23/03/2022	Fecha fin: 23/03/2022
Descripción: Desarrollar un método para subir la información de un estudiante manualmente, teniendo en cuenta que datos son obligatorios para almacenar en la base de datos.	

Tabla 3.54.-Tarea-codificación agregar estudiantes manualmente.**Elaborado por:** El investigador.

Tarea	Responsable: Guillermo Barcia
N° Tarea: T9	Código de Historia: H05
Tipo de Tarea: Desarrollo	Puntos Estimados: 2
Nombre de Tarea: Buscador por información del estudiante.	
Fecha inicio: 24/03/2022	Fecha fin: 24/03/2022
Descripción: Desarrollar un buscador el cual permita filtrar por cédula, nombre, apellido, matrícula y folio.	

Tabla 3.55.-Tarea-buscador por información del estudiante.**Elaborado por:** El investigador.

Tareas: Formulario docentes - H06

Tarea	Responsable: Guillermo Barcia
N° Tarea: T10	Código de Historia: H06
Tipo de Tarea: Desarrollo	Puntos Estimados: 2
Nombre de Tarea: Codificación agregar docentes manualmente.	
Fecha inicio: 25/03/2022	Fecha fin: 25/03/2022
Descripción: Desarrollar un método para subir la información de un estudiante manualmente, teniendo en cuenta que datos son obligatorios para almacenar en la base de datos.	

Tabla 3.56.-Tarea- codificación agregar docentes manualmente.

Elaborado por: El investigador.

Tarea	Responsable: Guillermo Barcia
N° Tarea: T11	Código de Historia: H06
Tipo de Tarea: Desarrollo	Puntos Estimados: 2
Nombre de Tarea: Buscador por información del docente.	
Fecha inicio: 28/03/2022	Fecha fin: 28/03/2022
Descripción: Desarrollar un buscador, el cual permita filtrar por cédula, nombre, apellido, matrícula y folio.	

Tabla 3.57.- Tarea-buscador por información del docente.

Elaborado por: El investigador.

Tareas: Información Estudiantes - H07

Tarea	Responsable: Guillermo Barcia
N° Tarea: T12	Código de Historia: H07
Tipo de Tarea: Desarrollo	Puntos Estimados: 1
Nombre de Tarea: Presentación de información detallada de los estudiantes.	
Fecha inicio: 29/03/2022	Fecha fin: 29/03/2022
Descripción: Presentar todos los datos del estudiante en una tabla, en el cual, podrá realizar la acción de editar los datos y de inhabilitar las columnas. Adicional a esto, se tiene la tabla paginada por un filtro de 25, 50 o 100 filas, dependiendo de cuanto escoja el usuario.	

Tabla 3.58.- Tarea-presentación de información detallada de los estudiantes.

Elaborado por: El investigador.

Tareas: Información Docentes - H08

Tarea	Responsable: Guillermo Barcia
N° Tarea: T13	Código de Historia: H08
Tipo de Tarea: Desarrollo	Puntos Estimados: 1
Nombre de Tarea: Presentación de información detallada de los docentes.	
Fecha inicio: 30/03/2022	Fecha fin: 30/03/2022
Descripción: Presentar todos los datos del docente en una tabla, en el cual, podrá realizar la acción de editar los datos y de inhabilitar las columnas. Adicional a esto, se tiene la tabla paginada por un filtro de 25, 50 o 100 filas, dependiendo de cuanto escoja el usuario.	

Tabla 3.59.- Tarea- presentación de información detallada de los docentes.**Elaborado por:** El investigador.**Tareas: Visualizar carreras universitarias – H09**

Tarea	Responsable: Guillermo Barcia
N° Tarea: T14	Código de Historia: H09
Tipo de Tarea: Desarrollo	Puntos Estimados: 2
Nombre de Tarea: Diseño de interfaz de visualización de carreras universitarias.	
Fecha inicio: 31/03/2022	Fecha fin: 31/03/2022
Descripción: Presentar al usuario una interfaz en el cual, pueda ver todas las carreras universitarias. La información será mostrada en una tabla, que se encuentra paginada por secciones de 25, 50 o 100 filas.	

Tabla 3.60.- Tarea- diseño de interfaz de visualización de carreras universitarias.**Elaborado por:** El investigador.**Tareas: Formulario carreras universitarias – H10**

Tarea	Responsable: Guillermo Barcia
N° Tarea: T15	Código de Historia: H10
Tipo de Tarea: Desarrollo	Puntos Estimados: 3
Nombre de Tarea: Buscador por nombre de la carrera universitaria.	
Fecha inicio: 1/04/2022	Fecha fin: 1/04/2022
Descripción: Desarrollar un buscador, que permita buscar por nombre de la carrera universitaria, en el cual, a medida que el usuario ingrese letras se va a ir filtrando las coincidencias.	

Tabla 3.61.- Tarea- buscador por nombre de la carrera universitaria.**Elaborado por:** El investigador.

Tarea	Responsable: Guillermo Barcia
N° Tarea: T16	Código de Historia: H10
Tipo de Tarea: Desarrollo	Puntos Estimados: 3
Nombre de Tarea: Diseño de interfaz para crear, editar e inhabilitar carrera universitaria.	
Fecha inicio: 04/04/2022	Fecha fin: 04/04/2022
Descripción: Realizar la interfaz para poder crear, editar e inhabilitar carreras universitarias, además de desarrollar la respectiva funcionalidad para poder almacenar los datos.	

Tabla 3.62.- Tarea-diseño de interfaz para crear, editar e inhabilitar carrera universitaria.
Elaborado por: El investigador.

Tareas: Gestión de perfil de usuario– H11

Tarea	Responsable: Guillermo Barcia
N° Tarea: T17	Código de Historia: H11
Tipo de Tarea: Desarrollo	Puntos Estimados: 2
Nombre de Tarea: Diseño de interfaz para poder ver los datos del usuario y cambiar la contraseña.	
Fecha inicio: 05/04/2022	Fecha fin: 05/04/2022
Descripción: Realizar la interfaz, para ver los datos del usuario y desarrollar la lógica para poder cambiar la contraseña, ingresando la contraseña anterior y la nueva.	

Tabla 3.63.- Tarea-diseño de interfaz para poder ver los datos del usuario y cambiar la contraseña.
Elaborado por: El investigador.

Tareas: Gestión de usuarios y roles – H12

Tarea	Responsable: Guillermo Barcia
N° Tarea: T18	Código de Historia: H12
Tipo de Tarea: Desarrollo	Puntos Estimados: 2
Nombre de Tarea: Diseño de interfaz de perfil.	
Fecha inicio: 06/04/2022	Fecha fin: 06/04/2022
Descripción: Presentar al usuario una interfaz en el cual, pueda modificar su nombre de usuario y su contraseña. Teniendo en cuenta, que para cambiar la contraseña se requiere ingresar la contraseña anterior.	

Tabla 3.64.- Tarea-diseño de interfaz de perfil.
Elaborado por: El investigador.

Tarea	Responsable: Guillermo Barcia
N° Tarea: T19	Código de Historia: H12
Tipo de Tarea: Desarrollo	Puntos Estimados: 4
Nombre de Tarea: Implementación del paquete Spatie/Laravel-permission.	
Fecha inicio: 07/03/2022	Fecha fin: 08/04/2022
Descripción: Implementar el paquete Spatie, con la finalidad de poder manejar permisos y roles de manera flexible, a la hora de tomar decisiones sobre a qué grupo de usuarios, se le muestra algún tipo de contenido y a quienes se les debe ocultar.	

Tabla 3.65.- Tarea- implementación del paquete spatie/laravel-permission.
Elaborado por El investigador.

Tarea	Responsable: Guillermo Barcia
N° Tarea: T20	Código de Historia: H12
Tipo de Tarea: Desarrollo	Puntos Estimados: 3
Nombre de Tarea: Formulario funcional de usuarios.	
Fecha inicio: 11/04/2022	Fecha fin: 12/04/2022
Descripción: Presentar al usuario una interfaz en el cual, agregar nuevos usuarios, seleccionando el rol que le corresponde, su nombre, correo principal (institucional) y un correo secundario (Gmail). Se debe tener en consideración, que la única persona que tiene acceso a este módulo es la persona con rol de Administrador.	

Tabla 3.66.- Tarea- formulario funcional de usuarios.
Elaborado por: El investigador.

Tareas: Visualizar procesos académicos – H13

Tarea	Responsable: Guillermo Barcia
N° Tarea: T21	Código de Historia: H13
Tipo de Tarea: Desarrollo	Puntos Estimados: 3
Nombre de Tarea: Diseño de interfaz de visualización de procesos académicos.	
Fecha inicio: 13/04/2022	Fecha fin: 13/04/2022
Descripción: Presentar al usuario una interfaz en el cual, pueda ver todos los procesos académicos que existen y que tenga la opción de poder añadir, editar y ver las plantillas que tienen en ese proceso académico. La información será mostrada en una tabla, que se encuentra paginada por secciones de 25, 50 o 100 filas.	

Tabla 3.67.- Tarea- diseño de interfaz de visualización de procesos académicos.
Elaborado por: El investigador.

Tareas: Formulario de crear procesos académicos – H14

Tarea	Responsable: Guillermo Barcia
N° Tarea: T22	Código de Historia: H14
Tipo de Tarea: Desarrollo	Puntos Estimados: 3
Nombre de Tarea: Codificación para crear los procesos académicos.	
Fecha inicio: 14/04/2022	Fecha fin: 14/04/2022
Descripción: Desarrollar el respectivo modulo, para crear los procesos académicos, en el cual se debe llenar los campos de nombre y que el proceso está activo.	

Tabla 3.68.- Tarea-codificación para crear los procesos académicos.

Elaborado por: El investigador.

Tarea	Responsable: Guillermo Barcia
N° Tarea: T23	Código de Historia: H14
Tipo de Tarea: Desarrollo	Puntos Estimados: 3
Nombre de Tarea: Buscador por nombre del proceso académico.	
Fecha inicio: 15/04/2022	Fecha fin: 15/04/2022
Descripción: Desarrollar un buscador, que permita buscar por nombre del proceso académico, en el cual, a medida que el usuario ingrese letras se va a ir filtrando las coincidencias.	

Tabla 3.69.- Tarea-buscador por nombre del proceso académico.

Elaborado por: El investigador.

Tareas: Editar procesos académicos – H15

Tarea	Responsable: Guillermo Barcia
N° Tarea: T24	Código de Historia: H15
Tipo de Tarea: Desarrollo	Puntos Estimados: 3
Nombre de Tarea: Codificación para editar los procesos académicos.	
Fecha inicio: 18/04/2022	Fecha fin: 18/04/2022
Descripción: Desarrollar el respectivo modulo, para editar los procesos académicos, en el cual se debe llenar los campos respectivos. Además, el usuario puede escoger si desea inactivar dicho proceso académico.	

Tabla 3.70.- Tarea-codificación para editar los procesos académicos.

Elaborado por: El investigador.

Tareas: Visualización de plantillas – H16

Tarea	Responsable: Guillermo Barcia
N° Tarea: T25	Código de Historia: H16
Tipo de Tarea: Desarrollo	Puntos Estimados: 3
Nombre de Tarea: Diseño de interfaz de visualización de plantillas.	
Fecha inicio: 19/04/2022	Fecha fin: 19/04/2022
Descripción: Presentar al usuario una interfaz en el cual, pueda ver todas las plantillas que existan dentro de dicho proceso académico en el que pertenece. La información, será mostrada en una tabla que, se encuentra paginada por secciones de 25, 50 o 100 filas.	

Tabla 3.71.- Tarea- diseño de interfaz de visualización de plantillas.

Elaborado por: El investigador.

Tareas: Crear plantillas – H17

Tarea	Responsable: Guillermo Barcia
N° Tarea: T26	Código de Historia: H17
Tipo de Tarea: Desarrollo	Puntos Estimados: 3
Nombre de Tarea: Codificación para crear las plantillas.	
Fecha inicio: 20/04/2022	Fecha fin: 20/04/2022
Descripción: Desarrollar el respectivo modulo, para crear las plantillas, en el cual se debe llenar los campos de nombre y que la plantilla se encuentre activa.	

Tabla 3.72.- Tarea- codificación para crear las plantillas.

Elaborado por: El investigador.

Tarea	Responsable: Guillermo Barcia
N° Tarea: T27	Código de Historia: H17
Tipo de Tarea: Desarrollo	Puntos Estimados: 3
Nombre de Tarea: Buscador por nombre de la plantilla.	
Fecha inicio: 21/04/2022	Fecha fin: 22/04/2022
Descripción: Desarrollar un buscador, que permita buscar por nombre de la plantilla dentro del proceso académico, en el cual, a medida que el usuario ingrese letras se filtra las coincidencias.	

Tabla 3.73.- Tarea- buscador por nombre de la plantilla.

Elaborado por: El investigador.

Tareas: Formulario de editar plantillas – H18

Tarea	Responsable: Guillermo Barcia
N° Tarea: T28	Código de Historia: H18
Tipo de Tarea: Desarrollo	Puntos Estimados: 3
Nombre de Tarea: Codificación para editar las plantillas.	
Fecha inicio: 25/04/2022	Fecha fin: 25/04/2022
Descripción: Desarrollar el respectivo modulo, para editar las plantillas del proceso académico en el que se encuentra almacenada. El usuario puede escoger si desea inactivar dicha plantilla y también puede cambiar de proceso académico.	

Tabla 3.74.- Tarea- codificación para editar las plantillas.**Elaborado por:** El investigador.**Tareas:** Formulario de plantilla de Word – H19

Tarea	Responsable: Guillermo Barcia
N° Tarea: T29	Código de Historia: H19
Tipo de Tarea: Desarrollo	Puntos Estimados: 3
Nombre de Tarea: Realizar un formulario en donde se refleje el archivo Word.	
Fecha inicio: 26/04/2022	Fecha fin: 27/04/2022
Descripción: Desarrollar el respectivo modulo, para ver las plantillas en un nuevo formulario y editar la estructura de ellas, que servirá para formar el acta de cada Consejo, ya sea Directivo de Facultad, Académico, de Titulación o Integración Curricular. Se utiliza el API de Google Drive, para estructurar las plantillas de los procesos académicos. En este se agregan las plantillas con las respectivas etiquetas.	

Tabla 3.75.- Tarea- realizar un formulario en donde se refleje el archivo Word.**Elaborado por:** El investigador.**Tareas:** Visualización de Consejos – H20

Tarea	Responsable: Guillermo Barcia
N° Tarea: T30	Código de Historia: H20
Tipo de Tarea: Desarrollo	Puntos Estimados: 3
Nombre de Tarea: Diseño de interfaz de visualización de Consejo Directivo de Facultad.	
Fecha inicio: 28/04/2022	Fecha fin: 28/04/2022
Descripción: Presentar al usuario una interfaz, donde refleje todos los Consejos Directivos de Facultad creados, pueda visualizar el Estado y realizar las respectivas acciones como editar, visualizar miembros, asistencia, ver documentos (resoluciones), cerrar y borrar Consejo Directivo. Además, consta de un buscador, que permite filtrar por nombre del Consejo, en el cual, a medida	

que el usuario ingrese letras se va filtrando las coincidencias.

Tabla 3.76.-Tarea- diseño de interfaz de visualización de consejo directivo de facultad.
Elaborado por: El investigador.

Tarea	Responsable: Guillermo Barcia
N° Tarea: T31	Código de Historia: H20
Tipo de Tarea: Desarrollo	Puntos Estimados: 3
Nombre de Tarea: Diseño de interfaz de visualización de Consejo Académico.	
Fecha inicio: 29/04/2022	Fecha fin: 29/04/2022
Descripción: Presentar al usuario una interfaz, donde refleje todos los Consejos Académicos creados, pueda visualizar el Estado y realizar las respectivas acciones como editar, visualizar miembros, asistencia, ver documentos (acuerdos), cerrar y borrar Consejo Académico. Además, consta de un buscador, que permite filtrar por nombre del Consejo, en el cual, a medida que el usuario ingrese letras se va filtrando las coincidencias.	

Tabla 3.77.-Tarea- diseño de interfaz de visualización de consejo académico.
Elaborado por: El investigador.

Tarea	Responsable: Guillermo Barcia
N° Tarea: T32	Código de Historia: H20
Tipo de Tarea: Desarrollo	Puntos Estimados: 3
Nombre de Tarea: Diseño de interfaz de visualización de Consejo Titulación.	
Fecha inicio: 02/05/2022	Fecha fin: 02/05/2022
Descripción: Presentar al usuario una interfaz, donde refleje todos los Consejos Titulación creados, pueda visualizar el Estado y realizar las respectivas acciones como editar, visualizar miembros, asistencia, ver documentos (acuerdos), cerrar y borrar Consejo Titulación. Además, consta de un buscador, que permite filtrar por nombre del Consejo, en el cual, a medida que el usuario ingrese letras se va filtrando las coincidencias.	

Tabla 3.78.- Tarea- diseño de interfaz de visualización de consejo titulación.
Elaborado por: El investigador.

Tarea	Responsable: Guillermo Barcia
N° Tarea: T33	Código de Historia: H20
Tipo de Tarea: Desarrollo	Puntos Estimados: 3
Nombre de Tarea: Diseño de interfaz de visualización de Consejo de Integración Curricular.	
Fecha inicio: 03/05/2022	Fecha fin: 03/05/2022
Descripción: Presentar al usuario una interfaz, donde refleje todos los Consejos Integración	

Curricular creados, pueda visualizar el Estado y realizar las respectivas acciones como editar, visualizar miembros, asistencia, ver documentos (acuerdos), cerrar y borrar Consejo de Integración Curricular. Además, consta de un buscador, que permite filtrar por nombre del Consejo, en el cual, a medida que el usuario ingrese letras se va filtrando las coincidencias.

Tabla 3.79.- Tarea-diseño de interfaz de visualización de consejo integración curricular.
Elaborado por: El investigador.

Tarea	Responsable: Guillermo Barcia
N° Tarea: T34	Código de Historia: H20
Tipo de Tarea: Desarrollo	Puntos Estimados: 3
Nombre de Tarea: Diseño de interfaz de asistencia.	
Fecha inicio: 04/05/2022	Fecha fin: 04/05/2022
Descripción: Presentar al usuario una interfaz, donde se refleje el botón agregar miembros, notificar y marcar asistencia. En el cual, se agregan las personas que van a asistir al Consejo ya sea Directivo de Facultad, Académico, Titulación o Integración Curricular.	

Tabla 3.80.-Tarea- diseño de interfaz de asistencia.
Elaborado por: El investigador.

Tarea	Responsable: Guillermo Barcia
N° Tarea: T35	Código de Historia: H20
Tipo de Tarea: Desarrollo	Puntos Estimados: 3
Nombre de Tarea: Diseño de interfaz de agregar miembros.	
Fecha inicio: 05/05/2022	Fecha fin: 05/05/2022
Descripción: Presentar al usuario una interfaz, donde se abra una ventana emergente, donde pueda agregar los miembros que en este caso son los docentes. El usuario podrá seleccionar si es un responsable o no.	

Tabla 3.81.- Tarea-diseño de interfaz de agregar miembros.
Elaborado por: El investigador.

Tareas: Crear Consejos – H21

Tarea	Responsable: Guillermo Barcia
N° Tarea: T36	Código de Historia: H21
Tipo de Tarea: Desarrollo	Puntos Estimados: 3

Nombre de Tarea: Codificación para crear los Consejos Directivos de Facultad.	
Fecha inicio: 06/05/2022	Fecha fin: 06/05/2022
Descripción: Desarrollar el respectivo modulo, para crear los Consejos Directivos de Facultad, en el cual se debe seleccionar el tipo de sesión: Extraordinaria u Ordinaria, el nombre y la fecha-hora que se crea el Consejo.	

Tabla 3.82.- Tarea-codificación para crear los consejos directivos de facultad.
Elaborado por: El investigador.

Tarea	Responsable: Guillermo Barcia
N° Tarea: T37	Código de Historia: H21
Tipo de Tarea: Desarrollo	Puntos Estimados: 3
Nombre de Tarea: Codificación para crear los Consejos Académicos.	
Fecha inicio: 09/05/2022	Fecha fin: 09/05/2022
Descripción: Desarrollar el respectivo modulo, para crear los Consejos Académicos, en el cual se debe seleccionar el tipo de sesión: Extraordinaria u Ordinaria, el nombre y la fecha-hora que se crea el Consejo.	

Tabla 3.83.- Tarea-codificación para crear los consejos académicos.
Elaborado por: El investigador.

Tarea	Responsable: Guillermo Barcia
N° Tarea: T38	Código de Historia: H21
Tipo de Tarea: Desarrollo	Puntos Estimados: 3
Nombre de Tarea: Codificación para crear los Consejos de Titulación.	
Fecha inicio: 10/05/2022	Fecha fin: 10/05/2022
Descripción: Desarrollar el respectivo modulo, para crear los Consejos de Titulación, en el cual se debe seleccionar el tipo de sesión: Extraordinaria u Ordinaria, el nombre y la fecha-hora que se crea el Consejo.	

Tabla 3.84.- Tarea-codificación para crear los consejos de titulación.
Elaborado por: El investigador.

Tarea	Responsable: Guillermo Barcia
N° Tarea: T39	Código de Historia: H21

Tipo de Tarea: Desarrollo	Puntos Estimados: 3
Nombre de Tarea: Codificación para crear los Consejos de Integración Curricular.	
Fecha inicio: 11/05/2022	Fecha fin: 11/05/2022
Descripción: Desarrollar el respectivo modulo, para crear los Consejos de Integración Curricular, en el cual se debe seleccionar el tipo de sesión: Extraordinaria u Ordinaria, el nombre y la fecha-hora que se crea el Consejo.	

Tabla 3.85.- Tarea-codificación para crear los consejos de integración curricular.
Elaborado por: El investigador.

Tareas: Modificar Consejos– H22

Tarea	Responsable: Guillermo Barcia
N° Tarea: T40	Código de Historia: H22
Tipo de Tarea: Desarrollo	Puntos Estimados: 3
Nombre de Tarea: Codificación para modificar los Consejos Directivos de Facultad.	
Fecha inicio: 12/05/2022	Fecha fin: 12/05/2022
Descripción: Desarrollar el respectivo modulo, para modificar los Consejos Directivos de Facultad, en el cual se debe llenar los campos respectivos. Los campos para modificar son el tipo de sesión, nombre y fecha.	

Tabla 3.86.- Tarea-codificación para modificar los consejos directivos de facultad
Elaborado por: El investigador.

Tarea	Responsable: Guillermo Barcia
N° Tarea: T41	Código de Historia: H22
Tipo de Tarea: Desarrollo	Puntos Estimados: 3
Nombre de Tarea: Codificación para modificar los Consejos Académicos.	
Fecha inicio: 13/05/2022	Fecha fin: 13/05/2022
Descripción: Desarrollar el respectivo modulo, para modificar los Consejos Académicos, en el cual se debe llenar los campos respectivos. Los campos para modificar son el tipo de sesión, nombre y fecha.	

Tabla 3.87.- Tarea-codificación para modificar los consejos académicos.
Elaborado por: El investigador.

Tarea	Responsable: Guillermo Barcia
N° Tarea: T42	Código de Historia: H22
Tipo de Tarea: Desarrollo	Puntos Estimados: 3
Nombre de Tarea: Codificación para crear los Consejos de Titulación.	
Fecha inicio: 16/05/2022	Fecha fin: 16/05/2022
Descripción: Desarrollar el respectivo modulo, para crear los Consejos de Titulación, en el cual se debe seleccionar el tipo de sesión: Extraordinaria u Ordinaria, el nombre y la fecha-hora que se crea el consejo.	

Tabla 3.88.- Tarea- codificación para crear los consejos de titulación.

Elaborado por: El investigador.

Tarea	Responsable: Guillermo Barcia
N° Tarea: T43	Código de Historia: H22
Tipo de Tarea: Desarrollo	Puntos Estimados: 3
Nombre de Tarea: Codificación para modificar los Consejos de Integración Curricular.	
Fecha inicio: 17/05/2022	Fecha fin: 17/05/2022
Descripción: Desarrollar el respectivo modulo, para modificar los Consejos de Integración Curricular, en el cual se debe llenar los campos respectivos. Los campos para modificar son el tipo de sesión, nombre y fecha.	

Tabla 3.89.- Tarea- codificación para modificar los consejos de integración curricular.

Elaborado por: El investigador.

Tareas: Eliminar Consejo Directivo – H23

Tarea	Responsable: Guillermo Barcia
N° Tarea: T44	Código de Historia: H23
Tipo de Tarea: Desarrollo	Puntos Estimados: 3
Nombre de Tarea: Desarrollo de funcionalidad para eliminar Consejo Directivo de Facultad.	
Fecha inicio: 18/05/2022	Fecha fin: 18/05/2022
Descripción: Desarrollar el respectivo modulo, para eliminar Consejo Directivo de Facultad. Se debe tomar en cuenta que, el usuario puede eliminar un Consejo Directivo, siempre y cuando este no se encuentre en estado cerrado.	

Tabla 3.90.- Tarea- desarrollo de funcionalidad para eliminar consejo directivo de facultad.

Elaborado por: El investigador.

Tarea	Responsable: Guillermo Barcia
N° Tarea: T45	Código de Historia: H23
Tipo de Tarea: Desarrollo	Puntos Estimados: 3
Nombre de Tarea: Desarrollo de funcionalidad para eliminar Consejo Académico.	
Fecha inicio: 19/05/2022	Fecha fin: 19/05/2022
Descripción: Desarrollar el respectivo modulo, para eliminar Consejo Académico. Se debe tomar en cuenta que, el usuario puede eliminar un Consejo Académico, siempre y cuando este no se encuentre en estado cerrado.	

Tabla 3.91.- Tarea- desarrollo de funcionalidad para eliminar consejo académico.
Elaborado por: El investigador.

Tarea	Responsable: Guillermo Barcia
N° Tarea: T46	Código de Historia: H23
Tipo de Tarea: Desarrollo	Puntos Estimados: 3
Nombre de Tarea: Desarrollo de funcionalidad para eliminar Consejo de Titulación.	
Fecha inicio: 20/05/2022	Fecha fin: 20/05/2022
Descripción: Desarrollar el respectivo modulo, para eliminar Consejo de Titulación. Se debe tomar en cuenta que, el usuario puede eliminar un Consejo de Titulación, siempre y cuando este no se encuentre en estado cerrado.	

Tabla 3.92.- Tarea- desarrollo de funcionalidad para eliminar consejo de titulación.
Elaborado por: El investigador.

Tarea	Responsable: Guillermo Barcia
N° Tarea: T47	Código de Historia: H23
Tipo de Tarea: Desarrollo	Puntos Estimados: 3
Nombre de Tarea: Desarrollo de funcionalidad para eliminar Consejo de Integración Curricular.	
Fecha inicio: 23/05/2022	Fecha fin: 23/05/2022
Descripción: Desarrollar el respectivo modulo, para eliminar Consejo de Integración Curricular. Se debe tomar en cuenta que, el usuario puede eliminar un Consejo de Integración Curricular, siempre y cuando este no se encuentre en estado cerrado.	

Tabla 3.93.- Tarea- desarrollo de funcionalidad para eliminar consejo de integración curricular.
Elaborado por: El investigador.

Tareas: Formulario de miembros o representantes de los Consejos – H24

Tarea	Responsable: Guillermo Barcia
N° Tarea: T48	Código de Historia: H24
Tipo de Tarea: Desarrollo	Puntos Estimados: 3
Nombre de Tarea: Desarrollo de funcionalidad y de interfaz para designar los miembros.	
Fecha inicio: 24/05/2022	Fecha fin: 24/05/2022
Descripción: Desarrollar el respectivo módulo, para asignar los miembros, si es responsable o no y definir la asistencia de dicho miembro.	

Tabla 3.94.- Tarea- desarrollo de funcionalidad y de interfaz para designar los miembros.

Elaborado por: El investigador.

Tareas: Formulario de números reservados de documentación – H25

Tarea	Responsable: Guillermo Barcia
N° Tarea: T49	Código de Historia: H25
Tipo de Tarea: Desarrollo	Puntos Estimados: 3
Nombre de Tarea: Diseño de la interfaz de números reservados.	
Fecha inicio: 25/05/2022	Fecha fin: 25/05/2022
Descripción: Diseñar la interfaz, en donde el usuario pueda seleccionar el Consejo (Directivo, Académico, Titulación o Integración Curricular), al cual, va a reservar los números reservados.	

Tabla 3.95.- Tarea- diseño de la interfaz de números reservados.

Elaborado por: El investigador.

Tareas: Crear documentos – H26

Tarea	Responsable: Guillermo Barcia
N° Tarea: T50	Código de Historia: H26
Tipo de Tarea: Desarrollo	Puntos Estimados: 4
Nombre de Tarea: Diseño de interfaz para formulario documentos.	
Fecha inicio: 26/05/2022	Fecha fin: 26/05/2022
Descripción: Diseñar la interfaz, en donde el usuario pueda ingresar y ver los números de las resoluciones, escoger el Consejo, dependiendo de qué opción del sistema escoja (Directivo, Académico, Titulación o Integración Curricular), procesos académicos, plantillas, buscar estudiante, buscar docentes e ingresar descripción.	

Tabla 3.96.- Tarea- diseño de interfaz para formulario documentos.

Elaborado por: El investigador.

Tarea	Responsable: Guillermo Barcia
N° Tarea: T51	Código de Historia: H26
Tipo de Tarea: Desarrollo	Puntos Estimados: 4
Nombre de Tarea: Desarrollo de funcionalidad para ver números reservados y encolados.	
Fecha inicio: 27/05/2022	Fecha fin: 27/05/2022
Descripción: Desarrollar la funcionalidad, para que el usuario puede ver los números reservados que están ocupados y a que Consejo pertenecen (Directivo, Académico, Titulación o Integración Curricular). Además, de ver los números encolados que se refiere a los documentos que hayan sido borrados.	

Tabla 3.97.- Tarea- desarrollo de funcionalidad para ver números reservados y encolados.
Elaborado por: El investigador.

Tarea	Responsable: Guillermo Barcia
N° Tarea: T52	Código de Historia: H26
Tipo de Tarea: Desarrollo	Puntos Estimados: 4
Nombre de Tarea: Desarrollo de funcionalidad para guardar documentos.	
Fecha inicio: 30/05/2022	Fecha fin: 30/05/2022
Descripción: Se agrega la carga de datos para que se refleje los Consejos ya sea Directivo, Académico, Titulación o Integración Curricular, procesos académicos y plantillas. Además, de que el usuario tiene una opción de búsqueda por estudiante y docente. En el caso de ser un alumno se busca por cédula, nombres, apellidos, folio y matrícula. En el caso de ser profesor se busca por nombres. El usuario podrá guardar todos estos datos.	

Tabla 3.98.- Tarea-desarrollo de funcionalidad para guardar documentos.
Elaborado por: El investigador.

Tareas: Editar documentos – H27

Tarea	Responsable: Guillermo Barcia
N° Tarea: T53	Código de Historia: H27
Tipo de Tarea: Desarrollo	Puntos Estimados: 3
Nombre de Tarea: Desarrollo de funcionalidad para editar documentos.	
Fecha inicio: 31/05/2022	Fecha fin: 31/05/2022
Descripción: Desarrollar funcionalidad, para que el usuario pueda visualizar todos los documentos creados, en dicho Consejo (Directivo, Académico, Titulación o Integración Curricular) y pueda editar dicho documento mediante el editor de texto de Google Drive.	

Tabla 3.99.- Tarea-desarrollo de funcionalidad para editar documentos.
Elaborado por: El investigador.

Tareas: Notificación de tramites académicos – H28

Tarea	Responsable: Guillermo Barcia
N° Tarea: T54	Código de Historia: H28
Tipo de Tarea: Desarrollo	Puntos Estimados: 3
Nombre de Tarea: Notificar los trámites académicos de los estudiantes.	
Fecha inicio: 01/06/2022	Fecha fin: 01/06/2022
Descripción: Desarrollar la funcionalidad y diseñar los iconos respectivos, para notificar los tramites, mediante un mensaje de texto que será enviado por WhatsApp o por el correo institucional.	

Tabla 3.100.- Tarea- notificar los trámites académicos de los estudiantes.**Elaborado por:** El investigador.**Tareas:** Generar Acta – H29

Tarea	Responsable: Guillermo Barcia
N° Tarea: T55	Código de Historia: H29
Tipo de Tarea: Desarrollo	Puntos Estimados: 5
Nombre de Tarea: Desarrollo de funcionalidad para procesar documentos.	
Fecha inicio: 02/06/2022	Fecha fin: 02/06/2022
Descripción: Desarrollar funcionalidad para que el usuario pueda procesar los documentos de dicho Consejo (Directivo, Académico, Titulación o Integración Curricular). Lo que realiza esta opción, es recopilar la información de todos los documentos ya sean Resoluciones o Acuerdos, unifica todos los documentos y los separa con una plantilla, para que luego se pueda este documento, con todas resoluciones que han sido creadas en dicho Consejo. Toda esta funcionalidad se le debe atribuir las etiquetas {{FROM}} y {{TO}} para copiar la información del texto de los documentos, incluyendo negritas, cursivas, tablas, imágenes, etc.	

Tabla 3.101.- Tarea- desarrollo de funcionalidad para procesar documentos.**Elaborado por:** El investigador.

Tarea	Responsable: Guillermo Barcia
N° Tarea: T56	Código de Historia: H29
Tipo de Tarea: Desarrollo	Puntos Estimados: 5
Nombre de Tarea: Desarrollo de funcionalidad para generar plantilla acta.	
Fecha inicio: 03/06/2022	Fecha fin: 03/06/2022

Descripción: Desarrollar funcionalidad para que el usuario pueda generar la plantilla de dicho Consejo (Directivo, Académico, Titulación o Integración Curricular). En esta opción, el usuario puede reemplazar el acta que pertenece, el año, la fecha en letras, las personas que asistieron y el responsable del Consejo. En esta Acta, se adjunta los documentos que han sido procesados de las resoluciones o acuerdos.

Tabla 3.102.- Tarea-desarrollo de funcionalidad para generar plantilla acta.
Elaborado por: El investigador.

Tareas: Cerrar Consejos – H30

Tarea	Responsable: Guillermo Barcia
N° Tarea: T57	Código de Historia: H30
Tipo de Tarea: Desarrollo	Puntos Estimados: 4
Nombre de Tarea: Desarrollo de funcionalidad para cerrar Consejos.	
Fecha inicio: 06/06/2022	Fecha fin: 06/06/2022
Descripción: Desarrollar funcionalidad para que el usuario pueda cerrar el Consejo que haya seleccionado (Directivo, Académico, Titulación o Integración Curricular). En esta opción, al usuario le aparece un mensaje de alerta, si está seguro de realizar esta acción, ya que, si el caso es afirmativo, ya no se podrán generar actas.	

Tabla 3.103.- Tarea-desarrollo de funcionalidad para cerrar consejos.
Elaborado por: El investigador.

Fase 3: Iteraciones

Con la finalidad de cumplir el desarrollo del sistema, se procede a trabajar mediante iteraciones que en este caso son 4, tal como fue planteado en el proceso de elaboración y valoraciones de las historias de usuarios, tal y como indica la metodología XP. A continuación, se tiene la iteración 1:

Iteración	N° Historia de usuario	Historia de usuario	Tiempo Estimado	
			Días	Horas
Iteración 1	H01	Determinar la estructura del proyecto	2	8
	H02	Análisis y diseño de base datos	5	20
	H03	Diseño de interfaces	3	12

Tabla 3.104.- Resumen de historias de usuario de la iteración 1.
Elaborado por: El investigador.

H01: Determinar la estructura del proyecto

En la figura 3.13 se muestra la arquitectura del proyecto, en el que básicamente se ilustra como es el funcionamiento de esta arquitectura en el que según sus siglas

LAMP (Linux- Apache-MYSQL-PHP) + Express Js, son los que forman la arquitectura en el servidor en que se va a implementar el sistema web integrado.

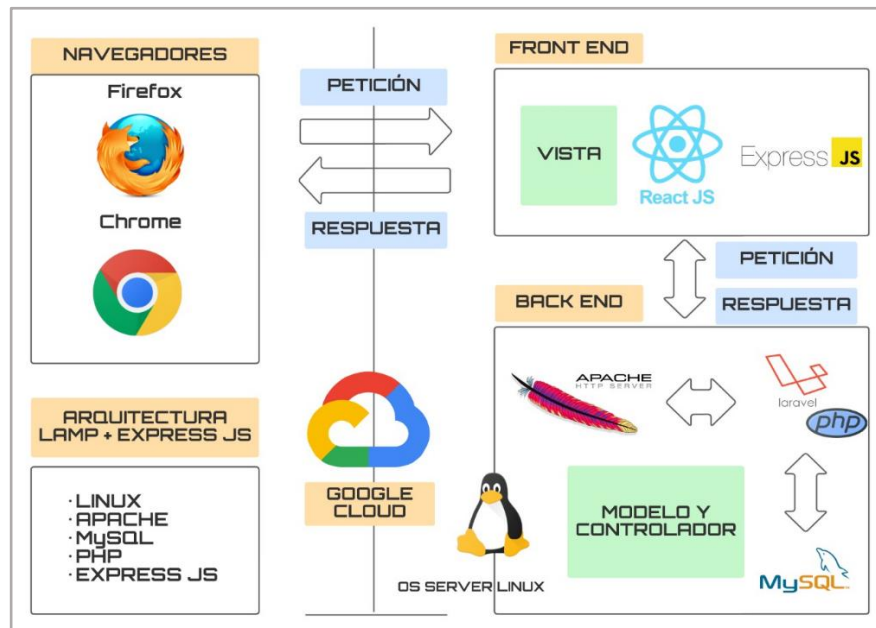


Figura 3.13.- Arquitectura LAMP + ExpressJs.
Elaborado por: El investigador.

H02: Análisis y diseño de la base de datos

En el anexo 1, se muestra el modelo del diseño relacional de la base de datos, en donde se guarda la respectiva información de los trámites académicos en la tabla documentos y se tiene todo el proceso que contienen para la creación de estos, que son: docentes, estudiantes, consejos, actas entre otros.

H03: Diseño de interfaces

Material UI

Conocido con sus acrónimos MUI (Material para Construir Interfaces) se caracteriza por ser una biblioteca robusta, personalizable y accesible a componentes básicos y avanzados, en el cual ofrece la capacidad de construir un sistema con un diseño elegante, con el marco front-end de React.js.

Con MUI se tiene la facilidad de crear diseños profesionales modernos, sin necesidad la iniciar un CSS (Cascading Style Sheets) completo desde cero, por lo que, se convierte en una herramienta muy eficaz para los desarrolladores porque facilita la vida al momento de crear sus interfaces de usuario. Ver figura 3.14



Figura 3.14.- Diseño de interfaz.
Elaborado por: El investigador.


Iteración 2

Iteración	Nº Historia de usuario	Historia de usuario	Tiempo Estimado	
			Días	Horas
Iteración 2	H04	Formulario de ingreso al sistema web integrado	2	8
	H05	Formularios estudiantes	3	12
	H06	Formularios docentes	2	8
	H07	Información estudiantes	1	4
	H08	Información docentes	1	4
	H09	Visualizar carreras universitarias	1	4
	H10	Formulario carreras universitarias	2	8
	H11	Gestión de perfil de usuario	1	4
	H12	Gestión de usuarios y roles	5	20

Tabla 3.105.- Resumen de historias de usuario de la iteración 2.
Elaborado por: El investigador.

H04: Formulario de ingreso al sistema web integrado

Se evidencia en la figura 3.15 el formulario de ingreso al sistema, El cual contiene dos campos: email y contraseña, que se debe ingresar correctamente para poder utilizar las funcionalidades del sistema.


FACULTAD DE INGENIERÍA EN SISTEMAS, ELECTRÓNICA E INDUSTRIAL

Email *

Contaseña *

INGRESAR

Figura 3.15.- Formulario de ingreso al sistema.
Elaborado por: El investigador.

H05: Formulario estudiantes


En la figura 3.16, se evidencia la interfaz para agregar estudiantes ingresando la información respectiva.

Agregar estudiante

Carrera
 Seleccionar...

Cédula *

Nombres * Apellidos *

Género
 Seleccionar... Fecha de nacimiento * 

Celular * Teléfono

Correo UTA * Correo

Matrícula Folio

LIMPIAR **GUARDAR**

Figura 3.16.- Agregar estudiantes.
Elaborado por: El investigador.

En la figura 3.17, se tiene la opción para subir múltiples alumnos mediante un archivo de Excel seleccionando la carrera a la que pertenece.

#	Cédula	Apellidos	Nombres	Teléfono	Celular
2	1804413373	ACUÑA AGUAGUIÑA	CARLOS JOSE		
3	1804775912	AGUAYSA CARRILLO	KAREN LISSETTE		
9	1804363883	ALTAMIRANO LOPEZ	ANTHONY MIGUEL	2489131	0984
10	0502856966	ALVAREZ MARTINEZ	ROGER LENIN	2489132	0984

Figura 3.17.- Agregar múltiples estudiantes.
Elaborado por: El investigador.

H06: Formulario docentes

A continuación, en la figura 3.18 se tiene la interfaz para agregar un docente ingresando la información requerida.

Cédula *	Nombres *
Celular *	Teléfono
Correo UTA *	Correo

Figura 3.18.- Crear docente.
Elaborado por: El investigador.

En la figura 3.19 se observa la opción de añadir cargos, donde se debe ingresar el nombre la variable y el docente.

Nombre	Variable	Docente	Acciones
Coordinador de sistemas	{{COOR_SISTEMAS}}	Federico Sigrid Stamm	
Coordinador de Electrónica	{{COOR_ELECTRONICA}}	Ewell Theresia Franecki Jerde	
Coordinador de Industrial	{{COOR_INDUSTRIAL}}	Bailee Kemmer	

Figura 3.19.- Cargos.
Elaborado por: El investigador.

Se puede agregar un cargo ingresando la siguiente información como lo muestra la figura 3.20.

Crear cargo

Nombre *

Variable *

Docente *

En el campo docente se tiene un buscador que filtra por cédula, nombres o apellidos

LIMPIAR GUARDAR

Figura 3.20.- Crear cargos.
Elaborado por: El investigador.

En la figura 3.21 se puede actualizar los datos de un docente para un cargo a excepción del campo variable, ya que este no es editable.

Actualizar cargo

Nombre *

Coordinador de sistemas

Variable

{{COOR_SISTEMAS}}

Docente *

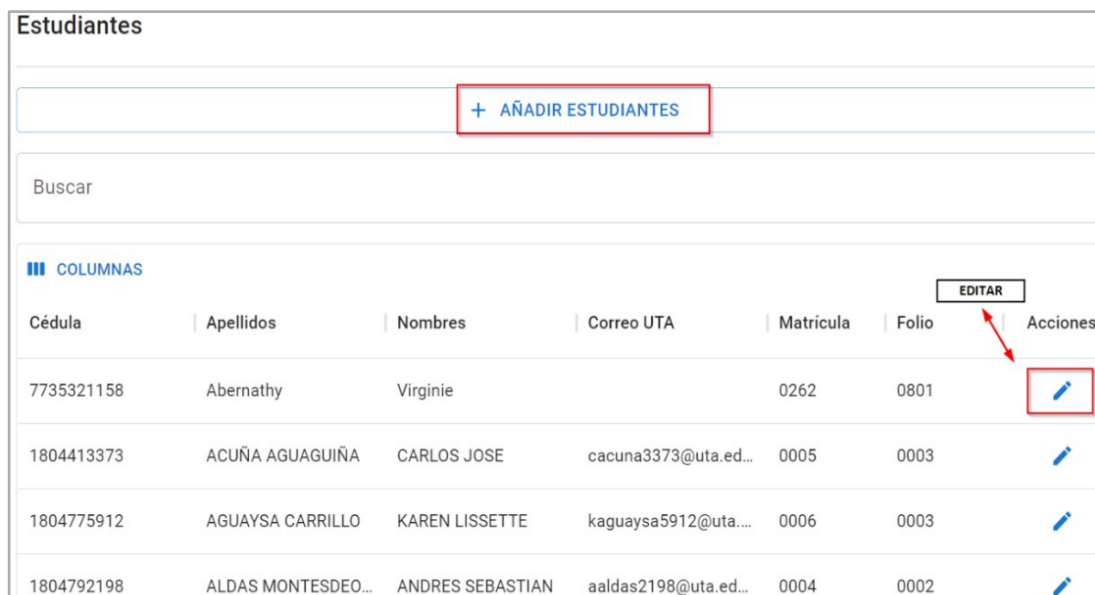
Ing. Clay Aldás

LIMPIAR GUARDAR

Figura 3.21.- Actualizar cargo.
Elaborado por: El investigador.

H07: Información estudiantes

En la figura 3.22 se evidencia la interfaz para añadir estudiantes, el cual contiene, la carrera, cédula, nombres completos, celular, correo, matricula, folio, además existe la opción editar datos.



The screenshot shows a web interface for managing students. At the top, there is a header 'Estudiantes' and a button '+ AÑADIR ESTUDIANTES'. Below this is a search bar labeled 'Buscar'. A table with columns 'Cédula', 'Apellidos', 'Nombres', 'Correo UTA', 'Matricula', 'Folio', and 'Acciones' is displayed. The 'Acciones' column contains an 'EDITAR' button and a pencil icon. The first row of the table is highlighted with a red box around the pencil icon.





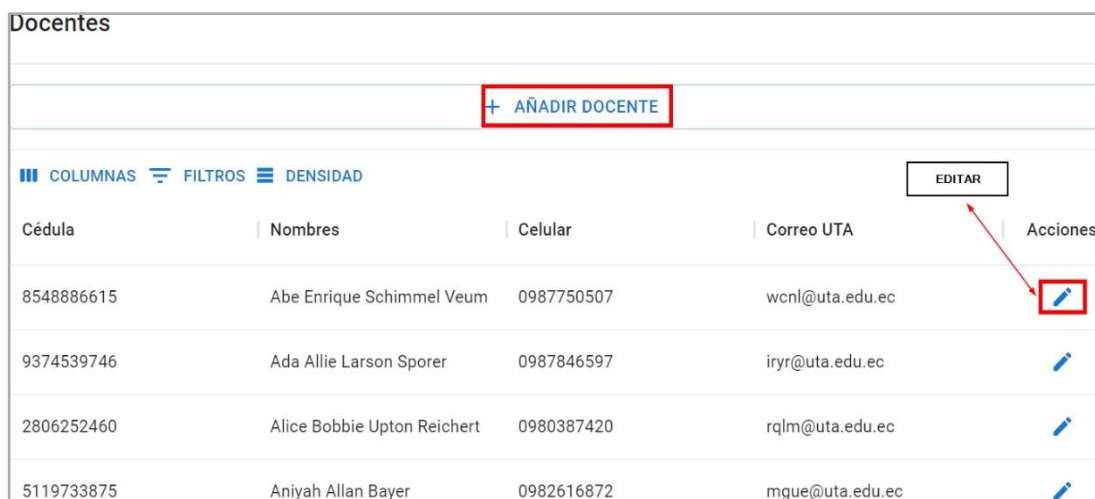
Cédula	Apellidos	Nombres	Correo UTA	Matricula	Folio	Acciones
7735321158	Abernathy	Virginie		0262	0801	
1804413373	ACUÑA AGUAGUIÑA	CARLOS JOSE	cacuna3373@uta.ed...	0005	0003	
1804775912	AGUAYSA CARRILLO	KAREN LISSETTE	kaguaysa5912@uta....	0006	0003	
1804792198	ALDAS MONTESDEO...	ANDRES SEBASTIAN	aaldas2198@uta.ed...	0004	0002	

Figura 3.22.- Añadir y listar estudiantes.

Elaborado por: El investigador.

H08: Información docentes

En la figura 3.23, se evidencia la interfaz para visualizar los docentes, en el cual consta de dos botones para añadir o editar.



The screenshot shows a web interface for managing teachers. At the top, there is a header 'Docentes' and a button '+ AÑADIR DOCENTE'. Below this are controls for 'COLUMNAS', 'FILTROS', and 'DENSIDAD'. A table with columns 'Cédula', 'Nombres', 'Celular', 'Correo UTA', and 'Acciones' is displayed. The 'Acciones' column contains an 'EDITAR' button and a pencil icon. The first row of the table is highlighted with a red box around the pencil icon.





Cédula	Nombres	Celular	Correo UTA	Acciones
8548886615	Abe Enrique Schimmel Veum	0987750507	wcnl@uta.edu.ec	
9374539746	Ada Allie Larson Sporer	0987846597	iryrr@uta.edu.ec	
2806252460	Alice Bobbie Upton Reichert	0980387420	rqlm@uta.edu.ec	
5119733875	Aniyah Allan Bayer	0982616872	mgue@uta.edu.ec	

Figura 3.23.- Añadir y listar docentes.

Elaborado por: El investigador.

H09: Visualizar carreras universitarias

En la figura 3.24 se muestran en un listado las carreras existentes en la FISEI.








Nombre ↑	Estado	Acciones
Ingeniería en Electrónica y Comunicaciones	Activado	
Ingeniería en Sistemas Computacionales e Informáticos	Activado	
Ingeniería Industrial	Activado	
Ingeniería Industrial en Procesos de Automatización	Activado	
Software	Activado	
Tecnologías de la Información	Activado	
Telecomunicaciones	Activado	

Figura 3.24.- Visualizar carreras universitarias.
Elaborado por: El investigador.

H10: Formulario carreras universitarias

En la figura 3.25 se muestra la opción para actualizar o editar la carrera universitaria para lo cual se debe ingresar el nombre de la carrera y los créditos.

Actualizar carrera	
Nombre *	Créditos *
Ingeniería en Electrónica y Telecomunicaciones	246
Título (Masculino) *	Título (Femenino) *
Ingeniero en Electrónica y Comunicaciones	Ingeniera en Electrónica y Comunicaciones
Estado	
Activado <input checked="" type="checkbox"/>	
LIMPIAR	
GUARDAR	

Figura 3.25.- Editar carreras universitarias.
Elaborado por: El investigador

H11: Gestión de perfil de usuario

En la figura 3.26 se muestra el perfil de usuario, donde se puede cambiar el nombre y solo visualizar el rol, los módulos asignados y los correos (institucional y Gmail).

Figura 3.26.- Opción de perfil.
Elaborado por: El investigador.

Además, en la figura 3.27 se muestra la opción para cambiar la contraseña.

Figura 3.27.- Opción para cambiar la contraseña.
Elaborado por: El investigador.

H12: Gestión de usuarios y roles

En la siguiente interfaz se muestra la creación de un usuario seleccionando el rol y los módulos que vaya a tener acceso al sistema como lo indica la figura 3.28.

Figura 3.28.- Crear nuevo usuario.
Elaborado por: El investigador.

En la figura 3.29 se observa la interfaz para ver los usuarios creados, en donde la opción para añadir, editar y restablecer contraseña.

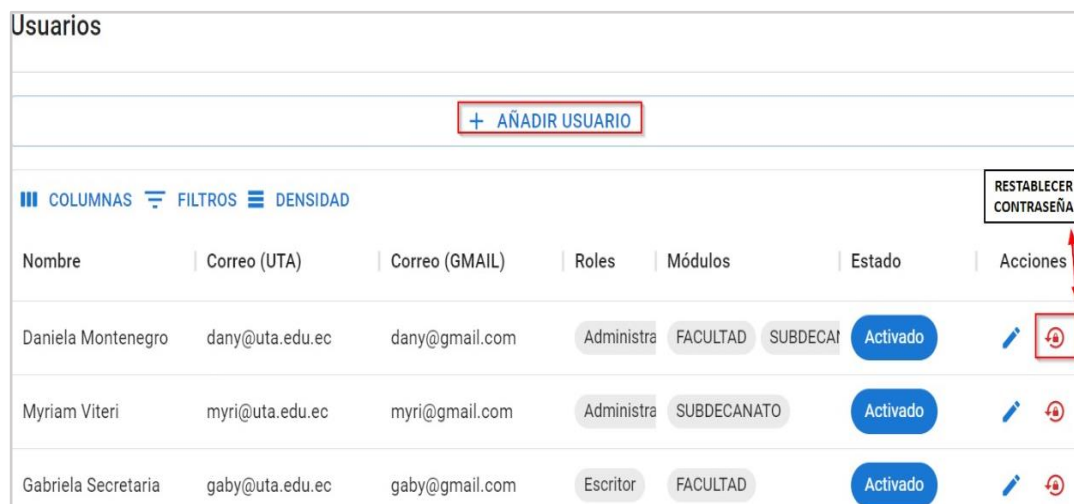


Figura 3.29.- Usuarios del sistema.
Elaborado por: El investigador.

En la figura 3.30 se muestra la interfaz para actualizar un usuario, donde tiene la opción para inactivarlo y cambiar los campos según sea necesario.

Figura 3.30.- Actualizar usuario.
Elaborado por: El investigador.

Iteración 3

Iteración	N° Historia de usuario	Historia de usuario	Tiempo Estimado	
			Días	Horas
Iteración 3	H13	Visualizar procesos académicos	1	4
	H14	Formulario de crear procesos académicos	2	8
	H15	Formulario de editar procesos académicos	1	4
	H16	Visualización de plantillas	1	4

	H17	Formulario de crear plantillas	3	12
	H18	Formulario de editar plantillas	1	4
	H19	Formulario de plantilla de Word	2	8

Tabla 3.106.- Resumen de historias de usuario de la iteración 3.
Elaborado por: El investigador.

H13: Visualizar procesos académicos

La visualización de procesos académicos se evidencia en la figura 3.31, en donde se puede crear procesos, ver plantillas y editarlas.

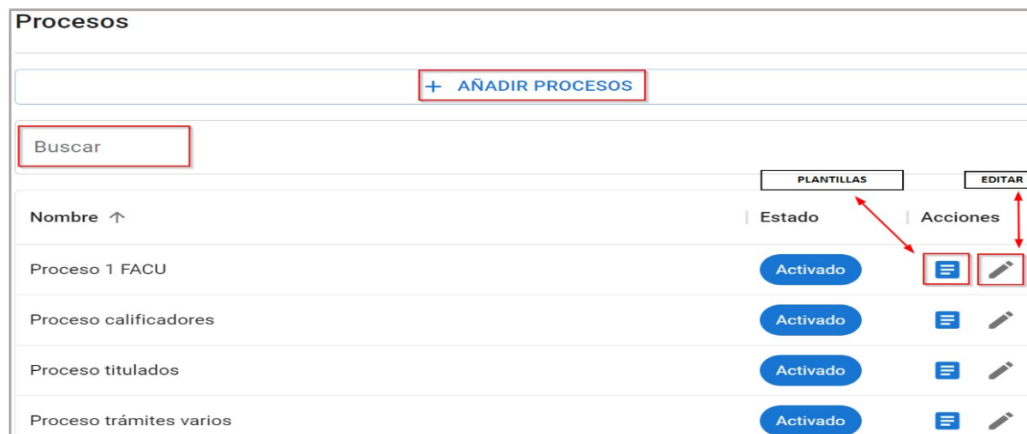


Figura 3.31.- Visualizar procesos académicos.
Elaborado por: El investigador

H14: Formulario de crear procesos académicos

El formulario de crear procesos académicos se muestra en la figura 3.32, en donde se debe ingresar el nombre del proceso académico.

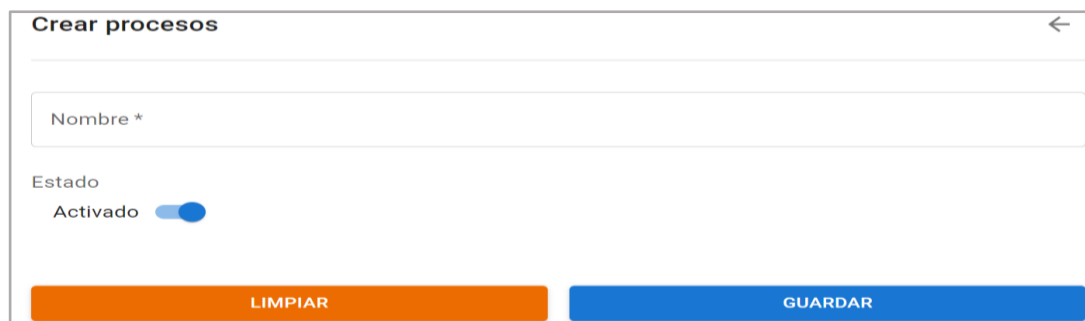


Figura 3.32.- Añadir procesos académicos.
Elaborado por: El investigador.

H15: Editar procesos académicos

La interfaz para editar procesos académicos se evidencia en la figura 3.33, en donde se debe colocar el nombre del proceso y se puede activar o desactivar el proceso.

Figura 3.33.- Editar procesos.
Elaborado por: El investigador.

H16: Visualización de plantillas

En la figura 3.34, se muestra la interfaz para visualizar plantillas, además se muestra que la opción para verificar los documentos almacenados en Google Drive.

Figura 3.34.- Visualizar plantilla.
Elaborado por: El investigador.

H17: Formulario crear plantillas

En la figura 3.35, se muestra el formulario para crear plantillas, en el cual se debe ingresar el nombre de la plantilla.

Figura 3.35.- Añadir plantillas.
Elaborado por: El investigador.

H18: Formulario editar plantillas

En la figura 3.36, se muestra el formulario para editar plantillas, en el cual se debe seleccionar el proceso y el nombre.

Actualizar plantilla

Procesos

Proceso 1 FACU

Nombre *

Calificadores

Estado

Activado

LIMPIAR

GUARDAR

Al momento de actualizar plantilla se puede cambiar a otro proceso

Figura 3.36.- Actualizar plantillas.
Elaborado por: El investigador.

H19: Formulario de la plantilla de Word.

A continuación, se evidencia en la figura 3.37, la plantilla de documentos para reemplazar los datos requeridos.

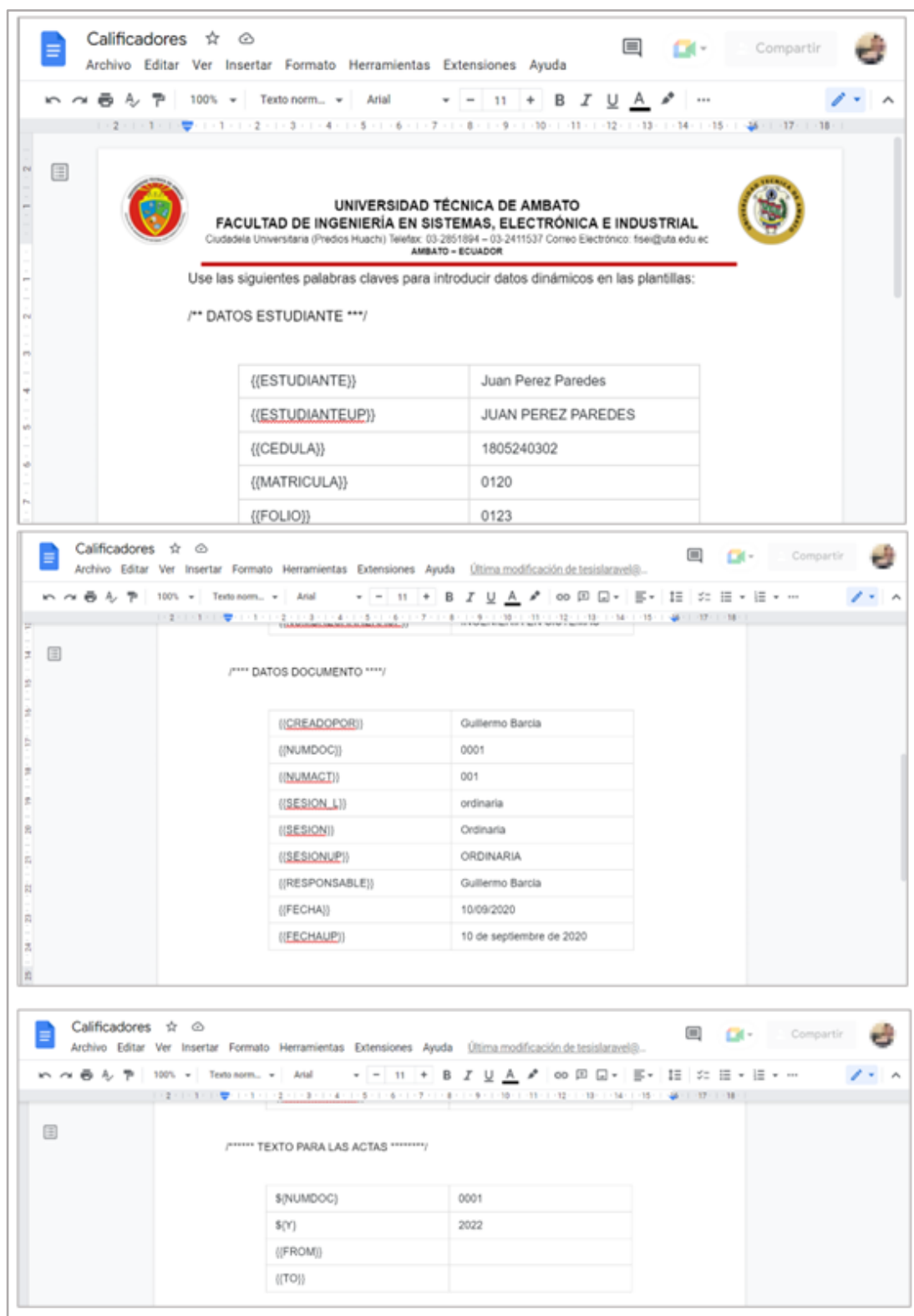


Figura 3.37.- Plantilla de documentos para reemplazar con datos requeridos.
Elaborado por: El investigador.

En la figura 3.38 se observa una plantilla de ejemplo donde se tiene la etiqueta `{{FROM}}` que sirve para el copiado de los documentos o resoluciones que se encuentren creados, al momento de generar un acta en un Consejo Directivo.

UNIVERSIDAD TÉCNICA DE AMBATO
FACULTAD DE INGENIERÍA EN SISTEMAS, ELECTRÓNICA E INDUSTRIAL
 Ciudadela Universitaria (Predios Huachi) Telefax: 03-2851894 – 03-2411537 Correo Electrónico: fisei@uta.edu.ec
 AMBATO – ECUADOR

Ambato, `{{FECHAUP}}`
 Resolución `{{NUMDOC}}`-P-CD-FISEI-UTA-2021

Señor/ita
`{{ESTUDIANTE}}`
ESTUDIANTE DE LA CARRERA DE `{{NOMBRECARRERAUP}}`
 Facultad de Ingeniería en Sistemas, Electrónica e Industrial
 Presente. -

De mi consideración:]

Consejo Directivo de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, en sesión `{{SESION}}` del día `{{FECHAUP}}`, realizada por medios electrónicos utilizando la plataforma tecnológica Zoom de conformidad con lo dispuesto en el artículo 64 del Código Orgánico Administrativo conoce la solicitud s/n de fecha 14 de diciembre de 2021, suscrita por el/la señor/ita `{{ESTUDIANTE}}`, estudiante de la Carrera de `{{NOMBRECARRERA}}`, solicitando se designe Tribunal de Calificación del Trabajo de Titulación, Modalidad **Proyecto de Investigación**. Al respecto; se RESUELVE:

`{{FROM}}`

DESIGNAR A LOS DOCENTES QUE SE DETALLAN A CONTINUACIÓN EN CALIDAD DE PROFESORES CALIFICADORES DEL TRABAJO DE TITULACIÓN: "SISTEMA DE DIAGNÓSTICO DE FALLAS PROFIBUS-DP DE LA MÁQUINA MP5 DE GRUPO FAMILIA MEDIANTE REDES NEURONALES ARTIFICIALES", PRESENTADO POR EL/LA SEÑOR/ITA `{{ESTUDIANTEUP}}`, QUIEN HA CULMINADO LA MALLA CURRICULAR DE LA CARRERA DE `{{NOMBRECARRERAUP}}`, DE CONFORMIDAD A LO DISPUESTO EN EL NUMERAL 9 DE LOS "LINEAMIENTOS GENERALES PARA LA APLICACIÓN DE INSTRUCTIVOS DE LAS MODALIDADES DE TITULACIÓN DE LAS FACULTADES DE LA UNIVERSIDAD TÉCNICA DE AMBATO", NORMATIVA APLICABLE AL CASO.

- `{{DOCENTE_N_0}}` (Principal)
- `{{DOCENTE_N_1}}` (Principal)
- `{{DOCENTE_N_2}}` (Suplente)

Figura 3.38.- Ejemplo de plantilla utilizando las etiquetas de `{{FROM}}`.
Elaborado por: El investigador.

De igual forma en la figura 3.39 la etiqueta `{{TO}}` me indica hasta que parte del documento se copia para generar el acta.

UNIVERSIDAD TÉCNICA DE AMBATO
FACULTAD DE INGENIERÍA EN SISTEMAS, ELECTRÓNICA E INDUSTRIAL
 Ciudadela Universitaria (Predios Huachi) Telefax: 03-2851894 – 03-2411537 Correo Electrónico: fisei@uta.edu.ec
 AMBATO – ECUADOR

COMUNICAR A LOS SEÑORES PROFESORES CALIFICADORES DEL TRABAJO DE TITULACIÓN QUE, EN BASE A LO DISPUESTO EN LOS "LINEAMIENTOS GENERALES PARA LA APLICACIÓN DE INSTRUCTIVOS DE LAS MODALIDADES DE TITULACIÓN DE LAS FACULTADES DE LA UNIVERSIDAD TÉCNICA DE AMBATO", TENDRÁN UN TIEMPO MÁXIMO DE 30 DÍAS DESPUÉS DE RECIBIR LA NOTIFICACIÓN DEL CONSEJO DIRECTIVO, PARA EMITIR LA CORRESPONDIENTE CALIFICACIÓN.

<u>TITULO</u>	<u>DESCRIPCION</u>
CONTENIDO 1	<code>DESCRIPCION 1</code>
CONTENIDO 2	<code>DESCRIPCION 2</code>

INFORMAR A EL/LA SEÑOR/ITA `{{ESTUDIANTEUP}}`, QUE UNA VEZ QUE LOS PROFESORES CALIFICADORES CONSIGNEN LAS CALIFICACIONES Y LAS MISMAS CUMPLAN CON EL PUNTAJE REQUERIDO, PODRÁ SOLICITAR LA SUSTENTACIÓN ORAL DEL TRABAJO DE TITULACIÓN, SIEMPRE Y CUANDO HAYA CUMPLIDO CON TODOS LOS REQUISITOS PREVIOS.

REMITIR LA DOCUMENTACIÓN A LA SECRETARÍA DE LA UNIDAD DE TITULACIÓN DE LA FACULTAD PARA SU CORRESPONDIENTE TRÁMITE Y POSTERIOR ARCHIVO.

`{{TO}}`

Atentamente,

Ing. Pilar Urrutia, Mg.
 PRESIDENTE
 CC:
 Secretaria de Carrera
 Secretaria de la Unidad de Titulación

Figura 3.39.- Ejemplo de plantilla utilizando las etiquetas de `{{TO}}`.
Elaborado por: El investigador.

Iteración 4

Iteración	N° Historia de usuario	Historia de usuario	Tiempo Estimado	
			Días	Horas
Iteración	H20	Visualización de Consejos	6	24
	H21	Crear Consejos	4	16
	H22	Modificar Consejos	4	16
	H23	Eliminar Consejos	4	16
	H24	Formulario de miembros o representantes de los Consejos	1	4
	H25	Formulario de números reservados de documentación	1	4
	H26	Crear documentos – Resolución o Acuerdo.	3	12
	H27	Editar documentos– Resolución o Acuerdo.	1	4
	H28	Notificación de tramites académicos (Resoluciones o Acuerdos).	1	4
	H29	Generar Acta de los Consejos	2	8
	H30	Cerrar Consejos	1	4

Tabla 3.107.- Resumen de historias de usuario de la iteración 4.

Elaborado por: El investigador.

H20: Visualización de consejos

La visualización de consejos se evidencia en la figura 3.40, en donde se puede identificar los consejos creados, en qué estado se encuentra, buscar, añadir, ver las plantillas tanto para el acta y separador.

The screenshot shows a web interface titled 'Consejos'. At the top, there are three buttons: '+ AÑADIR CONSEJO', 'PLANTILLA ACTA', and 'PLANTILLA SEPARADOR'. Below these is a search bar labeled 'Buscar'. The main part of the interface is a table with the following columns: Nombre, Fecha, Tipo, Estado, and Acciones. Two rows of data are visible, both with the state 'Abierto'. The 'Acciones' column for each row contains a set of icons for editing, deleting, and other actions.

Nombre	Fecha	Tipo	Estado	Acciones
Sesión FACU 31/10/2022	31/10/2022 11:47 AM	Extraordinaria	Abierto	[Edit, Check, Add, Lock, Delete]
Sesión FACU 22/10/2022	22/10/2022 05:00 AM	Ordinaria	Abierto	[Edit, Check, Add, Lock, Delete]

Figura 3.40.- Visualización de consejos.

Elaborado por: El investigador.

En la figura 3.41 se muestran las diferentes acciones que se pueden realizar.

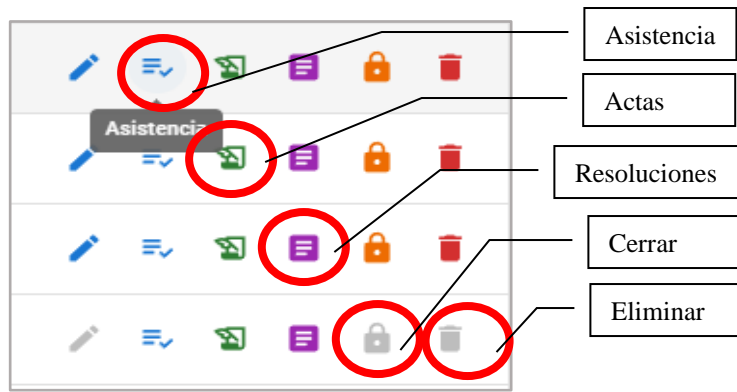


Figura 3.41.- Acciones.
Elaborado por: El investigador.

H21: Crear consejos

La creación de consejos se evidencia en la figura 3.42, en donde se puede ingresar el tipo, el nombre y la fecha.

Figura 3.42.- Crear consejos.
Elaborado por: El investigador.

H22: Modificar consejos

La modificación de consejos se evidencia en la figura 3.43, en donde se puede ingresar el tipo el cual puede ser ordinaria o extraordinaria, además permite el ingreso del nombre y la fecha.

Figura 3.43.- Modificar consejos.
Elaborado por: El investigador.

H23: Eliminar consejos

Para eliminar un consejo se debe dar clic en el basurero de color rojo, como se muestra en la figura 3.44.







Nombre	Fecha	Tipo	Estado	Acciones
Sesión FACU 31/1...	31/10/2022 05:00 AM	Ordinaria	Abierto	     

Figura 3.44.- Modificar consejos.

Elaborado por: El investigador.

Para posteriormente dar clic en eliminar dicho consejo, siempre y cuando no se haya generado el acta.

Eliminar

¿Está seguro que desea eliminar el registro **Sesión FACU 31/10/2022**?

[CANCELAR](#) [ELIMINAR](#)

Figura 3.45.- Eliminar consejos.

Elaborado por: El investigador.




H24: Formulario de miembros o representantes de los consejos

En la figura 3.46 se evidencia el formulario de miembros o representantes de los consejos, donde se puede agregar un miembro, notificar y marcar la asistencia.

Miembros

[+ AGREGAR MIEMBRO](#) [NOTIFICAR](#) [MARCAR ASISTENCIA](#)

COLUMNAS

<input checked="" type="checkbox"/>	Nombres	Notificado	Asistió	Responsable	Acciones
<input checked="" type="checkbox"/>	Ame Gottlieb Raynor	X	✓	✓	
<input checked="" type="checkbox"/>	Ing. Franklin Mayorga	X	✓	X	
<input checked="" type="checkbox"/>	Ing. Mg. Clay Aldás	X	✓	X	

3 filas seleccionadas


[PAGINACION](#)  Filas por página: 100 ▾ 1-3 de 3

Figura 3.46.- Asistencia.

Elaborado por: El investigador.

H25: Formulario de números reservados de documentación

En la figura 3.47 se evidencia el formulario de números reservados de la documentación, donde se puede seleccionar el consejo y reservar los números.

Reservar numeración

Consejo *
Sesión FACU 01/09/2022

Desde * 1 Hasta * 10

Total de números reservados 10 Siguiete número disponible 11

Esté número no está dentro del rango seleccionado a reservar

LIMPIAR GUARDAR

Figura 3.47.- Reservar numeración.
Elaborado por: El investigador.

En la figura 3.48 se muestra los números reservados y encolados que se encuentran disponibles.

Numeración

Números reservados	Consejo
10	Consejo 13 Abril
9	Consejo 13 Abril
8	Consejo 13 Abril
7	Consejo 13 Abril
6	Consejo 13 Abril
5	Consejo 13 Abril
4	Consejo 13 Abril
3	Consejo 13 Abril

Filas por página: 100 1-8 de 8

Números encolados

14
13
12
11

CANCELAR ACEPTAR

Figura 3.48.- Ver números reservados y encolados.
Elaborado por: El investigador.

H26: Crear documentos – Resolución o Acuerdo

La creación de documentos como resolución o acuerdo se muestra en la figura 3.49.

Generar documento

Número *
1

Consejo *
Sesión FACU 01/09/2022

Procesos *
Proceso 1 FACU

Plantillas *
Calificadores

Estudiante
0804808335 - GUILLERMO ALEXANDER BARCIA GALLARDO

Docentes
Ing. Clay Aldás × Eileen Gleichner × Billy Douglas Braun × Cédula | N...

Descripción

Mantener formato

LIMPIAR GUARDAR

Figura 3.49.- Generar documento.
Elaborado por: El investigador.

En la figura 3.50 se observa la generación del documento con los datos de las etiquetas.

Ambato, 01 de septiembre de 2022
Resolución 0001-P-CD-FISEI-UTA-2021

Señor/ita
Guillermo Alexander Barcia Gallardo
ESTUDIANTE DE LA CARRERA DE INGENIERÍA EN SISTEMAS COMPUTACIONALES E INFORMATICOS
Facultad de Ingeniería en Sistemas, Electrónica e Industrial
Presente. -

De mi consideración:

Consejo Directivo de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, en sesión ordinaria del día 01 de septiembre de 2022, realizada por medios electrónicos utilizando la plataforma tecnológica Zoom de conformidad con lo dispuesto en el artículo 64 del Código Orgánico Administrativo conoce la solicitud s/n de fecha 14 de diciembre de 2021, suscrita por el/la señor/ita Guillermo Alexander Barcia Gallardo, estudiante de la Carrera de Ingeniería en Sistemas Computacionales e Informáticos, solicitando se designe Tribunal de Calificación del Trabajo de Titulación, Modalidad **Proyecto de Investigación**. Al respecto; se RESUELVE:

{{FROM}}

DESIGNAR A LOS DOCENTES QUE SE DETALLAN A CONTINUACIÓN EN CALIDAD DE PROFESORES CALIFICADORES DEL TRABAJO DE TITULACIÓN: "SISTEMA DE DIAGNÓSTICO DE FALLAS PROFIBUS-DP DE LA MÁQUINA MP5 DE GRUPO FAMILIA MEDIANTE REDES NEURONALES ARTIFICIALES", PRESENTADO POR EL/LA SEÑOR/ITA GUILLERMO ALEXANDER BARCIA GALLARDO, QUIEN HA CULMINADO LA MALLA CURRICULAR DE LA CARRERA DE INGENIERÍA EN SISTEMAS COMPUTACIONALES E INFORMATICOS. DE CONFORMIDAD A LO DISPUESTO EN EL NUMERAL 9 DE LOS "LINEAMIENTOS GENERALES PARA LA APLICACIÓN DE INSTRUCTIVOS DE LAS MODALIDADES DE TITULACIÓN DE LAS FACULTADES DE LA UNIVERSIDAD TÉCNICA DE AMBATO", NORMATIVA APLICABLE AL CASO.

- Ing. Clay Aldás (Principal)
- Eileen Gleichner (Principal)
- Billy Douglas Braun (Suplente)

Figura 3.50.- Generación del documento con los datos de las etiquetas.
Elaborado por: El investigador.

H27: Editar documentos – Resolución o Acuerdo

Primero se debe seleccionar el consejo del documento que se quiera editar y dar clic en ver documento.

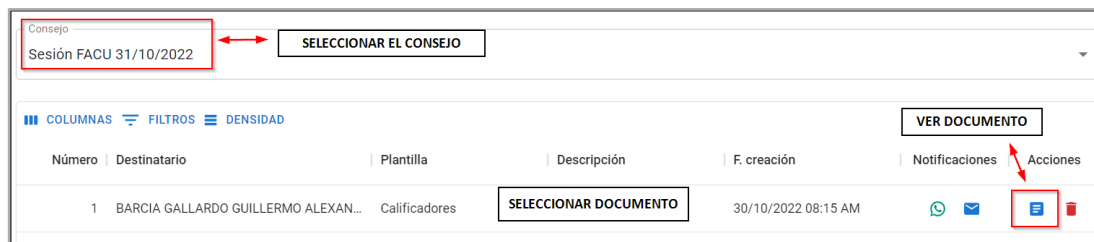


Figura 3.51.- Editar documentos-resolución o acuerdo.
Elaborado por: El investigador.

Posteriormente se procede a editar el documento que se abre en una pestaña de Google Docs.

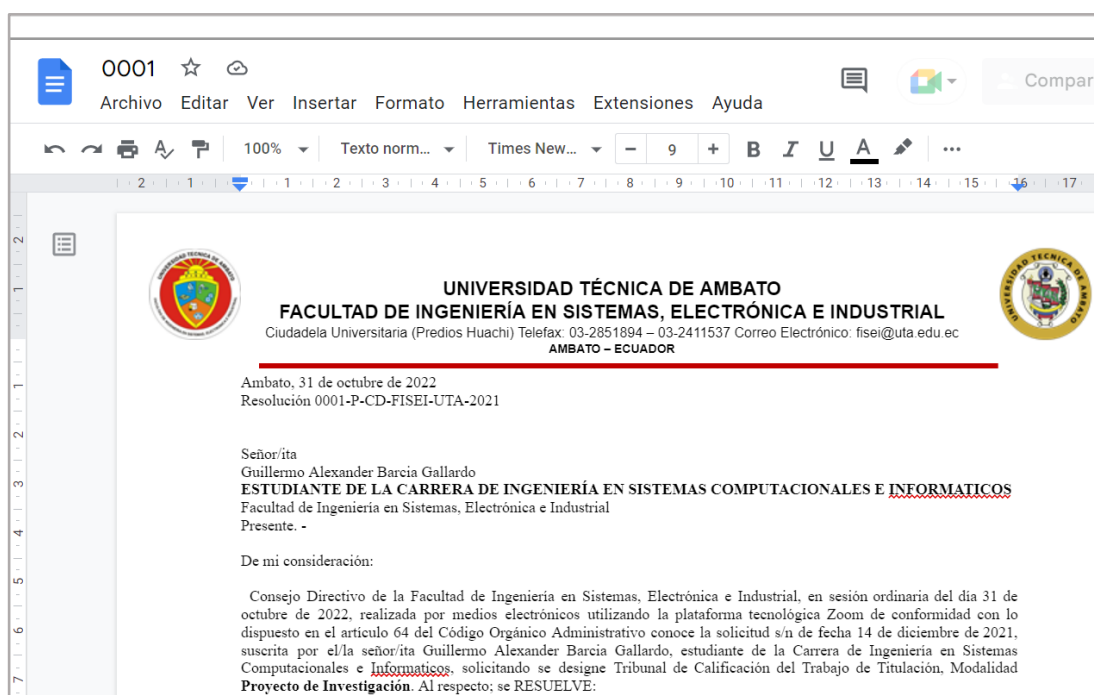


Figura 3.52.- Edición del documento en Google Docs.
Elaborado por: El investigador.

H28: Notificación de trámites académicos (Resoluciones o Acuerdos)

En la figura 3.53, se muestra la creación de documentos y notificaciones. Las notificaciones pueden ser al WhatsApp o al correo electrónico.



Figura 3.53.- Creación de documentos y notificaciones.
Elaborado por: El investigador.

En la figura 3.54 se evidencia la consulta de resoluciones.

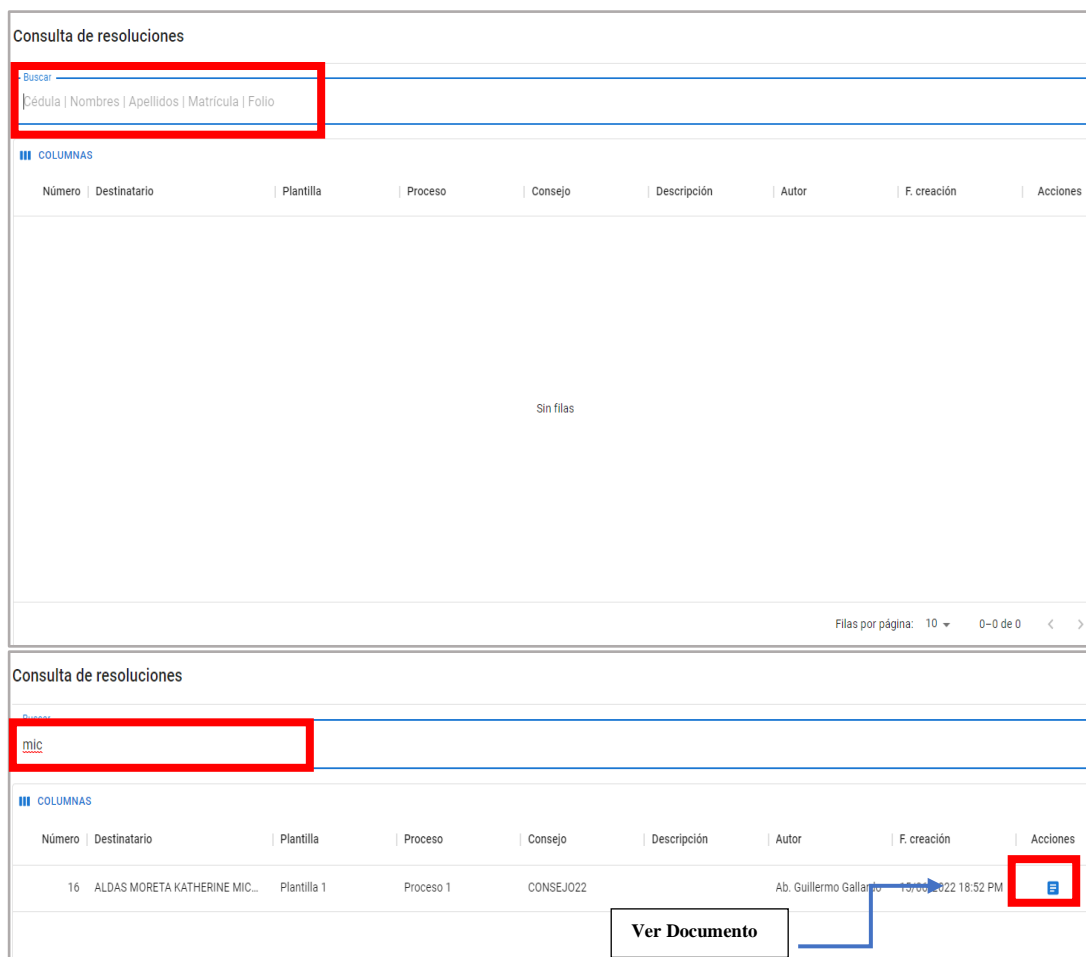


Figura 3.54.- Consulta de resoluciones.
Elaborado por: El investigador.

H29: Generar acta de los consejos

En la figura 3.55 se evidencia la plantilla de separación por actas.

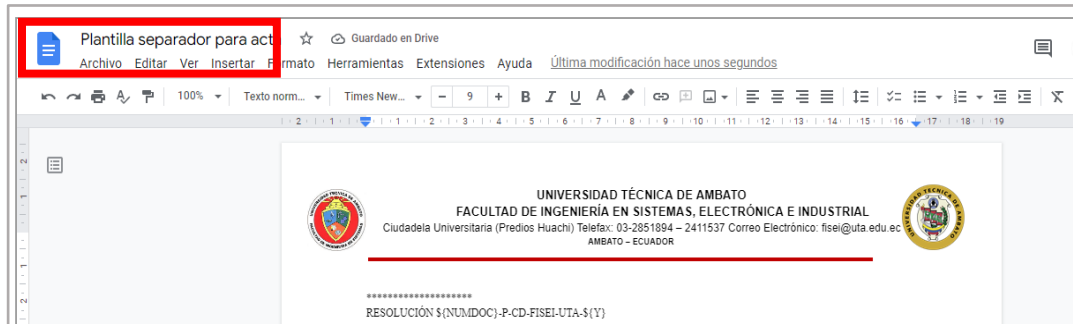


Figura 3.55.- Plantilla separador por actas.
Elaborado por: El investigador.

En la figura 3.56 se muestra el formato de la plantilla acta, que será utilizada al momento de generar el acta.

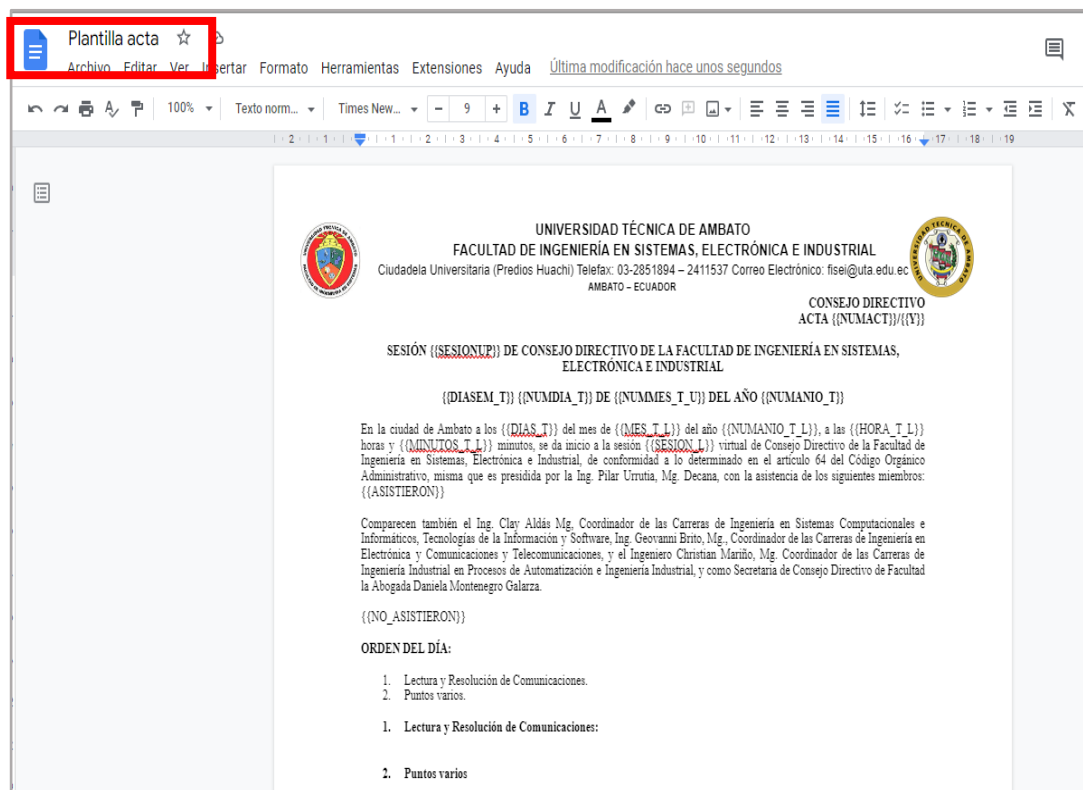


Figura 3.56.- Plantilla acta.
Elaborado por: El investigador.

En la figura 3.57 se evidencia el proceso para poder generar un acta, en el cual consta de 4 botones, el primero a realizar es el “Procesar documentos”, después dar clic en “Generar plantilla acta”, para luego proceder a la descarga de los documentos de todas las resoluciones o acuerdos que se realizan en ese Consejo Directivo y luego unificar en el botón de “Ver Acta”.

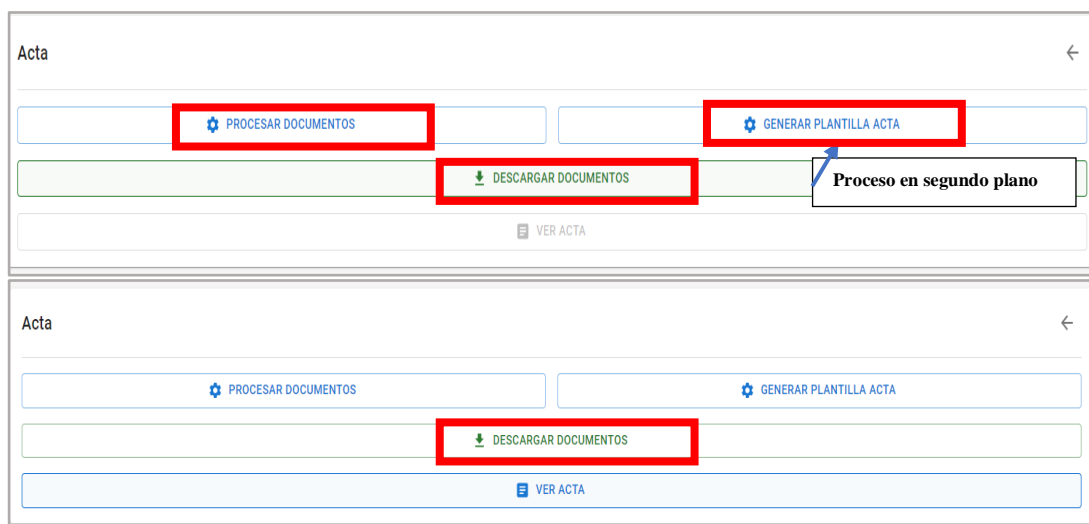


Figura 3.57.- Generación de acta.
Elaborado por: El investigador.

H30: Cerrar consejos

Anteriormente en la figura 3.58 la cual muestra las acciones que se pueden realizar en la interfaz, existe un botón en forma de candado de cerrar, en el cual al momento de dar clic sobre él se muestra la figura 3.56, tomando en cuenta que una vez cerrado ese consejo ya no se permite generar más actas, solo tiene opción a consultar documentos.



Figura 3.58.- Cerrar consejos.
Elaborado por: El investigador.

La configuración de la API de Google se encuentra en el anexo 2, mientras que la codificación del sistema web integrado se evidencia en el anexo 3.

Fase 4: Pruebas

Para determinar la correcta implementación del sistema web integrado, es necesario realizar las pruebas correspondientes y verificar que los requerimientos recolectados en primera instancia se efectúen correctamente. Es por ello, que se opta por las pruebas de caja negra, se las conoce también como black box testing, pueden definirse como una técnica donde se busca la verificación de las funcionalidades del software o aplicación analizada, donde las pruebas se llevan a cabo con

desconocimiento del funcionamiento del sistema interno con la ayuda del usuario final, debido a que se enfoca en la entrada y salida de un software, tomando como base sus especificaciones y requisitos.

Teniendo en cuenta que, para la aplicación de estas pruebas, se basa en los requerimientos de software para determinar la aceptación de las historias de usuario establecidas al principio de la metodología, por lo que continuación, en esta fase se utiliza las pruebas de aceptación. Las pruebas fueron realizadas en conjunto con la Ab. Daniela Montenegro, quien es la que certifica en funcionamiento de cada uno de los módulos que está compuesto el sistema.

Prueba de aceptación	
Código: P01	Historia de usuario: H04
Nombre: Formulario de ingreso al sistema web integrado.	
Descripción: Un usuario tendrá que ingresar su correo y contraseña para poder ingresar al sistema.	
Condiciones de ejecución: Ninguna.	
Entrada/ Proceso de Ejecución: -El usuario realizará el ingreso de su correo institucional y contraseña. -El usuario presionará un clic en el botón Ingresar.	
Resultado esperado: El usuario podrá visualizar la interfaz de inicio, con sus respectivos módulos del sistema web integrado.	
Evaluación de la prueba: La prueba se realizó de manera satisfactoria.	

Tabla 3.108.- Prueba de aceptación - formulario de ingreso al sistema web integrado.

Elaborado por: El investigador.

Prueba de aceptación	
Código: P02	Historia de usuario: H05
Nombre: Formulario estudiantes.	
Descripción: El usuario, puede agregar manualmente cada estudiante o subir en una carga de archivo de tipo Excel.	
Condiciones de ejecución: El formato en Excel debe cumplir un estándar, para poder subir los datos y se requiere que el usuario haya ingresado al sistema.	
Entrada/ Proceso de Ejecución: <ul style="list-style-type: none"> • Opción Agregar estudiante. -El usuario dará clic en Agregar estudiante. -El usuario deberá seleccionar la carrera, llenar campos como cédula, nombres, apellidos, celular,	

teléfono, correo institucional, correo, matrícula y folio. -El usuario dará clic en Guardar. <ul style="list-style-type: none"> • Opción Agregar múltiples estudiantes. -El usuario dará clic en Agregar múltiples estudiantes. -El usuario deberá seleccionar la carrera y subir el archivo en formato Excel. -Posterior se reflejará una tabla con todos los datos de los estudiantes que fueron subidos. -El usuario dará clic en Guardar.
Resultado esperado: Según la opción seleccionada, el usuario podrá guardar estudiantes, ya sea de forma manual o por una carga de archivos.
Evaluación de la prueba: La prueba se realizó de manera satisfactoria.

Tabla 3.109.- Prueba de aceptación - formulario estudiantes.
Elaborado por: El investigador.

Prueba de aceptación	
Código: P03	Historia de usuario: H06
Nombre: Formulario docentes.	
Descripción: El usuario tendrá que llenar los campos requeridos para guardar un docente.	
Condiciones de ejecución: El usuario haya ingresado al sistema.	
Entrada/ Proceso de Ejecución: -El usuario dará clic en Añadir Docentes. -El usuario deberá llenar los campos de: cédula, nombres, celular, teléfono, correo institucional y personal. -El usuario dará clic en Guardar.	
Resultado esperado: El usuario podrá guardar los docentes y se visualizará en una tabla.	
Evaluación de la prueba: La prueba se realizó de manera satisfactoria.	

Tabla 3.110.- Prueba de aceptación - formulario docentes.
Elaborado por: El investigador.

Prueba de aceptación	
Código: P04	Historia de usuario: H07
Nombre: Información estudiantes.	
Descripción: El usuario podrá ver todos los estudiantes que hayan sido agregados hasta la fecha, además de que puede realizar una opción de búsqueda por los filtros de: cédula, nombres, matrícula y folio.	
Condiciones de ejecución: El usuario debe tener ingresado estudiantes.	
Entrada/ Proceso de Ejecución: -El usuario dará clic en el menú Estudiantes.	

-El usuario podrá hacer una búsqueda, dando clic en el campo buscar por cualquiera de los filtros que le indica. -Posterior a ello, a medida que el usuario tipea caracteres se hará una búsqueda recursiva de los datos.
Resultado esperado: El usuario podrá visualizar todos los estudiantes de manera estructurada, mediante filtros de búsqueda y paginación personalizable.
Evaluación de la prueba: La prueba se realizó de manera satisfactoria.

Tabla 3.111.- Prueba de aceptación – información estudiantes.
Elaborado por: El investigador.

Prueba de aceptación	
Código: P05	Historia de usuario: H08
Nombre: Información docentes.	
Descripción: El usuario podrá ver todos los docentes, que hayan sido agregados hasta la fecha y puede filtrar por todos los campos.	
Condiciones de ejecución: El usuario debe tener ingresado docentes.	
Entrada/ Proceso de Ejecución: -El usuario dará clic en el menú Docentes. -El usuario podrá realizar filtros por columnas, filtros por cualquier columna y podrá exportar los datos como archivo CSV o PDF.	
Resultado esperado: El usuario podrá visualizar todos los docentes de manera estructurada, mediante filtros de búsqueda y paginación personalizable.	
Evaluación de la prueba: La prueba se realizó de manera satisfactoria.	

Tabla 3.112.- Prueba de aceptación - información docentes.
Elaborado por: El investigador.

Prueba de aceptación	
Código: P06	Historia de usuario: H09
Nombre: Visualizar carreras universitarias.	
Descripción: El usuario podrá visualizar todos los procesos académicos existentes.	
Condiciones de ejecución: El usuario haya ingresado al sistema.	
Entrada/ Proceso de Ejecución: -El usuario dará clic en el menú Carreras. -El usuario visualizará las carreras existentes, viendo si están en estado: Activado o Desactivado. -El usuario podrá buscar las carreras universitarias.	
Resultado esperado: El usuario puede ver todas las carreras universitarias.	

Descripción: El usuario podrá visualizar todas las carreras universitarias existentes.

Tabla 3.113.- Prueba de aceptación - visualizar carreras universitarias.

Elaborado por: El investigador.

Prueba de aceptación	
Código: P07	Historia de usuario: H10
Nombre: Formulario carreras universitarias.	
Descripción: El usuario tiene la opción de añadir carreras universitarias.	
Condiciones de ejecución: El usuario haya ingresado al sistema.	
Entrada/ Proceso de Ejecución:	
Crear	
-El usuario dará clic en el botón de Crear Carrera.	
-El usuario deberá llenar los campos correspondientes.	
-El usuario dará clic en el botón guardar.	
Modificar	
-El usuario dará clic en el ícono de Editar.	
-El usuario podrá cambiar el nombre de la carrera o inactivar carrera.	
-El usuario dará clic en el botón guardar.	
Resultado esperado: El usuario puede crear, modificar o inactivar una carrera universitaria.	
Evaluación de la prueba: La prueba se realizó de manera satisfactoria.	

Tabla 3.114.- Prueba de aceptación - formulario carreras universitarias.

Elaborado por: El investigador.

Prueba de aceptación	
Código: P08	Historia de usuario: H11
Nombre: Gestión de perfil de usuario.	
Descripción: El usuario podrá visualizar sus datos de perfil y cambiar la contraseña.	
Condiciones de ejecución: El usuario haya ingresado al sistema.	
Entrada/ Proceso de Ejecución:	
-El usuario dará clic en el menú Perfil.	
-El usuario visualizará su nombre de perfil que puede editar, en cambio el rol, el correo institucional y personal solo puede visualizar.	
-El usuario podrá cambiar su contraseña ingresando la anterior clave y la nueva, con su respectiva confirmación.	
-El usuario dará clic en actualizar contraseña.	

Resultado esperado: El usuario podrá cambiar su nombre de perfil y contraseña.
Evaluación de la prueba: La prueba se realizó de manera satisfactoria.

Tabla 3.115.- Prueba de aceptación - gestión de perfil de usuario.

Elaborado por: El investigador.

Prueba de aceptación	
Código: P09	Historia de usuario: H12
Nombre: Gestión de usuarios y roles	
Descripción: El usuario con permisos de administrador puede gestionar usuarios, es decir, añadir, editar e inactivar usuario con su respectivo rol, ya sea: administrador, administrador temporal, lector y escritor.	
Condiciones de ejecución: El usuario debe tener permisos de administrador.	
Entrada/ Proceso de Ejecución: -El usuario dará clic en el menú Usuarios. -El usuario podrá visualizar todos los usuarios que ya han sido creados, en el cual, tiene la opción de editar usuarios. Además, de ver el estado del usuario y añadir usuario. -Si el usuario da clic en añadir, debe seleccionar el rol, nombre, correo institucional y correo secundario. -Si el usuario da clic en editar, puede editar todos los campos y puede inactivar o activar un usuario.	
Resultado esperado: El usuario puede tener una gestión de usuarios, para crear, editar o inactivar cuentas.	
Evaluación de la prueba: La prueba se realizó de manera satisfactoria.	

Tabla 3.116.- Prueba de aceptación - gestión de usuarios y roles.

Elaborado por: El investigador.

Prueba de aceptación	
Código: P10	Historia de usuario: H13
Nombre: Visualizar procesos académicos.	
Descripción: El usuario podrá visualizar todos los procesos académicos existentes.	
Condiciones de ejecución: El usuario haya ingresado al sistema.	
Entrada/ Proceso de Ejecución: -El usuario dará clic en el menú procesos. -El usuario visualizará los procesos académicos existentes, viendo si están en estado: Activado o Desactivado. -El usuario podrá buscar los procesos académicos por nombre.	
Resultado esperado: El usuario puede ver todos los procesos académicos.	
Evaluación de la prueba: La prueba se realizó de manera satisfactoria.	

Tabla 3.117.- Prueba de aceptación - visualizar procesos académicos.
Elaborado por: El investigador.

Prueba de aceptación	
Código: P12	Historia de usuario: H14
Nombre: Formulario crear procesos académicos.	
Descripción: El usuario tiene la opción de añadir procesos académicos.	
Condiciones de ejecución: El usuario haya ingresado al sistema.	
Entrada/ Proceso de Ejecución: -El usuario dará clic en el botón de Añadir Procesos. -El usuario debe llenar los campos correspondientes. -El usuario dará clic en el botón guardar.	
Resultado esperado: El usuario puede añadir nuevos procesos académicos.	
Evaluación de la prueba: La prueba se realizó de manera satisfactoria.	

Tabla 3.118.- Prueba de aceptación – formulario de procesos académicos.
Elaborado por: El investigador.

Prueba de aceptación	
Código: P13	Historia de usuario: H15
Nombre: Editar procesos académicos	
Descripción: El usuario tiene la opción de editar los procesos académicos.	
Condiciones de ejecución: El usuario haya ingresado al sistema.	
Entrada/ Proceso de Ejecución: -El usuario dará clic en el botón Editar. -El usuario puede modificar el nombre y cambiar de estado: Activado o Desactivado. -El usuario dará clic en el botón Guardar.	
Resultado esperado: El usuario puede editar o inactivar los procesos académicos.	
Evaluación de la prueba: La prueba se realizó de manera satisfactoria.	

Tabla 3.119.- Prueba de aceptación - editar procesos académicos.
Elaborado por: El investigador.

Prueba de aceptación	
Código: P14	Historia de usuario: H16
Nombre: Visualización de plantillas.	

Descripción: El usuario visualizará las plantillas existentes, del proceso académico seleccionado.
Condiciones de ejecución: El usuario haya ingresado al sistema.
Entrada/ Proceso de Ejecución: -El usuario dará selecciona el proceso académico y dará clic en Plantillas. -El usuario visualizará las plantillas existentes, si están en estado: Activado o Desactivado y el autor que creó la plantilla. -El usuario podrá buscar las plantillas por nombre.
Resultado esperado: El usuario puede visualizar las plantillas de los procesos académicos.
Evaluación de la prueba: La prueba se realizó de manera satisfactoria.

Tabla 3.120.- Prueba de aceptación - visualización de plantillas.
Elaborado por: El investigador.

Prueba de aceptación	
Código: P15	Historia de usuario: H17
Nombre: Crear plantillas	
Descripción: El usuario tiene la opción de añadir plantillas.	
Condiciones de ejecución: El usuario haya ingresado al sistema.	
Entrada/ Proceso de Ejecución: -El usuario dará clic en el botón de Añadir Procesos. -El usuario debe llenar los campos correspondientes. -El usuario dará clic en el botón guardar.	
Resultado esperado: El usuario puede crear plantillas.	
Evaluación de la prueba: La prueba se realizó de manera satisfactoria.	

Tabla 3.121.- Prueba de aceptación - crear plantillas.
Elaborado por: El investigador.

Prueba de aceptación	
Código: P16	Historia de usuario: H18
Nombre: Editar plantillas	
Descripción: El usuario tiene la opción de editar las plantillas.	
Condiciones de ejecución: El usuario haya ingresado al sistema.	
Entrada/ Proceso de Ejecución: -El usuario dará clic en el botón Editar. -El usuario puede modificar el nombre y cambiar de estado: Activado o Desactivado. Además, tiene la opción de escoger si quiere cambiar a otro proceso académico. -El usuario dará clic en el botón Guardar.	
Resultado esperado: El usuario puede editar y cambiar de proceso académico la plantilla.	

Evaluación de la prueba: La prueba se realizó de manera satisfactoria.

Tabla 3.122.- Prueba de aceptación - crear plantillas.

Elaborado por: El investigador.

Prueba de aceptación	
Código: P17	Historia de usuario: H19
Nombre: Formulario de plantilla de Word	
Descripción: El usuario tiene la opción de modificar en Word la estructura de la plantilla.	
Condiciones de ejecución: El usuario haya ingresado al proceso académico y escoja la plantilla que quiera modificar en el Word.	
Entrada/ Proceso de Ejecución: -El usuario dará clic en el icono de Word Ver Documento. -El usuario puede visualizar en una nueva ventana el Word con el documento, en el cual se actualiza automáticamente. -El usuario puede editar en tiempo real y el guardado es automático. -El usuario visualizará el documento con las etiquetas que en esta se crean por defecto, para adaptar a la plantilla que se requiera.	
Resultado esperado: El usuario puede editar y cambiar de proceso académico la plantilla.	
Evaluación de la prueba: La prueba se realizó de manera satisfactoria.	

Tabla 3.123.- Prueba de aceptación - formulario de plantilla de Word.

Elaborado por: El investigador.

Prueba de aceptación	
Código: P18	Historia de usuario: H20
Nombre: Visualización de Consejos.	
Descripción: El usuario podrá visualizar todos los Consejos académicos existentes, ya sea el tipo de Consejo: Directivo de Facultad, Académico, de Titulación y de Integración Curricular.	
Condiciones de ejecución: El usuario haya ingresado al sistema.	
Entrada/ Proceso de Ejecución: -El usuario dará clic en el menú de Consejos. -El usuario visualizará los Consejos académicos existentes, viendo si están en estado: Activado o Cerrado, la fecha de creación y el tipo de Consejo: Extraordinaria u Ordinaria. -El usuario podrá buscar los Consejos académicos por nombre. -El usuario visualizará las diferentes acciones que puede realizar en este menú, tales como, ingresar asistencia, genera acta, ver resoluciones o acuerdos, cerrar y borrar Consejo.	
Resultado esperado: El usuario podrá visualizar todos los Consejos académicos existentes.	
Evaluación de la prueba: La prueba se realizó de manera satisfactoria.	

Tabla 3.124.- Prueba de aceptación - visualización de consejos.
Elaborado por: El investigador.

Prueba de aceptación	
Código: P19	Historia de usuario: H21
Nombre: Crear Consejos.	
Descripción: El usuario tiene la opción de añadir Consejos (Directivo de Facultad, Académico, de Titulación y de Integración Curricular).	
Condiciones de ejecución: El usuario haya ingresado al sistema.	
Entrada/ Proceso de Ejecución: -El usuario dará clic en el botón de Añadir Consejo. -El usuario seleccionará el tipo de sesión (Extraordinaria u Ordinaria), nombre y la fecha-hora a la que se crea el Consejo. -El usuario dará clic en el botón guardar.	
Resultado esperado: El usuario puede crear nuevos Consejos.	
Evaluación de la prueba: La prueba se realizó de manera satisfactoria.	

Tabla 3.125.- Prueba de aceptación - crear consejos.
Elaborado por: El investigador.

Prueba de aceptación	
Código: P20	Historia de usuario: H22
Nombre: Modificar Consejos.	
Descripción: El usuario tiene la opción de modificar los Consejos (Directivo de Facultad, Académico, de Titulación y de Integración Curricular).	
Condiciones de ejecución: El usuario haya ingresado al sistema.	
Entrada/ Proceso de Ejecución: -El usuario dará clic en el botón Modificar. -El usuario puede modificar el Consejo directivo, siempre y cuando se encuentre en estado Abierto, por lo que, puede modificar el tipo de sesión (Extraordinaria u Ordinaria), nombre y fecha. -El usuario dará clic en el botón Guardar.	
Resultado esperado: El usuario puede modificar el Consejo directivo.	
Evaluación de la prueba: La prueba se realizó de manera satisfactoria.	

Tabla 3.126.- Prueba de aceptación - modificar consejos.
Elaborado por: El investigador.

Prueba de aceptación	
Código: P21	Historia de usuario: H23
Nombre: Eliminar Consejos.	
Descripción: El usuario tiene la opción de eliminar Consejos (Directivo de Facultad, Académico, de Titulación y de Integración Curricular).	
Condiciones de ejecución: El Consejo debe estar en estado Abierto.	
Entrada/ Proceso de Ejecución: -El usuario dará clic en el icono de color rojo de una papelera. -El usuario visualizará una ventana emergente, preguntando si está seguro de eliminar dicho Consejo seleccionado. -El usuario tiene la opción de dar clic en eliminar o cancelar.	
Resultado esperado: El usuario puede eliminar Consejos	
Evaluación de la prueba: La prueba se realizó de manera satisfactoria.	

Tabla 3.127.- Prueba de aceptación - eliminar consejos.

Elaborado por: El investigador.

Prueba de aceptación	
Código: P22	Historia de usuario: H24
Nombre: Formulario de miembros o representantes	
Descripción: El usuario tiene la opción de agregar los miembros o responsable de un Consejo (Directivo de Facultad, Académico, de Titulación y de Integración Curricular).	
Condiciones de ejecución: El Consejo debe estar en estado Abierto y debe estar creado.	
Entrada/ Proceso de Ejecución: -El usuario dará clic en el icono de color azul miembros. -El usuario visualizará todos los miembros que pertenezcan en ese Consejo. -El usuario tiene la opción de dar clic en el botón agregar miembro. -El usuario visualizará una ventana emergente, en donde se elige la responsabilidad del miembro. -El usuario dará clic en agregar. -El usuario notificará por correo electrónico a las personas que asistirán a ese Consejo. -El usuario marcará asistencia a las personas que asistan al Consejo.	
Resultado esperado: El usuario puede agregar miembros de un Consejo.	
Evaluación de la prueba: La prueba se realizó de manera satisfactoria.	

Tabla 3.128.- Prueba de aceptación - formulario de miembros o representantes.

Elaborado por: El investigador.

Prueba de aceptación	
Código: P23	Historia de usuario: H25

Nombre: Formulario de números reservados de documentación.
Descripción: El usuario tiene la opción de seleccionar números reservados para los documentos.
Condiciones de ejecución: El usuario debe tener Consejos abiertos.
Entrada/ Proceso de Ejecución: - El usuario dará clic en el botón Números Reservados. -El usuario debe seleccionar el Consejo, y escoger el inicio y el final para reservar los números. -El usuario dará clic en Guardar.
Resultado esperado: El usuario puede generar números reservados, seleccionando el Consejo.
Evaluación de la prueba: La prueba se realizó de manera satisfactoria

Tabla 3.129.- Prueba de aceptación - formulario de números reservados de documentación.
Elaborado por: El investigador.

Prueba de aceptación	
Código: P24	Historia de usuario: H26
Nombre: Crear documentos.	
Descripción: El usuario tiene la opción de seleccionar en menú de documentos, para poder visualizar los documentos que estén creados en dicho Consejo (Directivo de Facultad, Académico, de Titulación y de Integración Curricular) y poder crear más documentos.	
Condiciones de ejecución: El usuario debe tener Consejos abiertos (Directivo de Facultad, Académico, de Titulación y de Integración Curricular), procesos académicos y plantillas creadas.	
Entrada/ Proceso de Ejecución: -El usuario dará clic en el botón Crear Documentos. -El usuario debe ingresar el número del documento, seleccionar el Consejo, proceso académico, plantilla, buscar estudiante, buscar docente e ingresar una descripción de tipo no obligatorio. -El usuario dará clic en Guardar.	
Resultado esperado: El usuario puede guardar nuevos documentos.	
Evaluación de la prueba: La prueba se realizó de manera satisfactoria.	

Tabla 3.130.- Prueba de aceptación - crear documentos.
Elaborado por: El investigador.

Prueba de aceptación	
Código: P25	Historia de usuario: H27
Nombre: Editar documentos.	
Descripción: El usuario tiene un icono en forma de documento de Word que permite editar el documento en tiempo real, es decir, que su guardado va a ser automático.	
Condiciones de ejecución: El usuario debe tener creado documentos.	
Entrada/ Proceso de Ejecución:	

-El usuario dará clic en el menú Documentos.
-El usuario visualizará los documentos, que estén creados en el Consejo seleccionado.
-El usuario dará clic en el icono azul de Word.
-El usuario editará el documento de Word que está alojado en Google Drive, considerando que, el guardado es automático.
Resultado esperado: El usuario puede guardar editar los documentos en tiempo real.
Evaluación de la prueba: La prueba se realizó de manera satisfactoria.

Tabla 3.131.- Prueba de aceptación - editar documentos.
Elaborado por: El investigador.

Prueba de aceptación	
Código: P26	Historia de usuario: H28
Nombre: Notificación de trámites académicos.	
Descripción: El usuario tiene la opción de notificar por dos medios: ya sea por vía WhatsApp o por correo institucional.	
Condiciones de ejecución: El usuario debe tener creado documentos, para emitir dichos documentos de los trámites académicos.	
Entrada/ Proceso de Ejecución: -El usuario dará clic en el menú Documentos. -El usuario seleccionará el Consejo. -El usuario dará clic en los iconos de notificaciones, ya sea por WhatsApp o por correo. -El usuario visualizará una ventana emergente, para revisar que información va a notificar. -El usuario dará clic en notificar.	
Resultado esperado: El usuario puede notificar los documentos por WhatsApp o por correo.	
Evaluación de la prueba: La prueba se realizó de manera satisfactoria.	

Tabla 3.132.- Prueba de aceptación – notificación de trámites académicos.
Elaborado por: El investigador.

Prueba de aceptación	
Código: P27	Historia de usuario: H29
Nombre: Generación de Actas de Consejo.	
Descripción: El usuario tiene dos botones: Procesar Documentos y Generar Plantilla Acta, en el cual el primer botón, realiza la unificación de las resoluciones o acuerdos y el segundo botón, reemplaza la información de las etiquetas para concretarse el Acta. La funcionalidad de procesar documentos se realiza en segundo plano, es decir, que el usuario puede hacer cualquier otra opción hasta que se generen todos los documentos.	
Condiciones de ejecución: El usuario debe tener creado documentos ya sea resoluciones o	

acuerdos, aparte de tener un responsable y haber ingresado la asistencia a los miembros del Consejo.
<p>Entrada/ Proceso de Ejecución:</p> <ul style="list-style-type: none"> -El usuario dará clic en el menú Consejos. -El usuario dará clic en el ícono verde, donde se genera el Acta. -El usuario dará clic en Procesar Documentos (Se desbloquee el botón Generar Plantilla Acta y se refleja el proceso el segundo plano). -El usuario dará clic en el botón Generar Plantilla Acta. -El usuario dará clic en Descargar Documentos de los documentos que fueron unificados en el Proceso de los documentos. -El usuario dará clic en Ver Acta, para después copiar la información del archivo que se descargó.
<p>Resultado esperado: El usuario tiene su acta generada de todos los documentos (Resoluciones o acuerdos) en dicho Consejo.</p>
<p>Evaluación de la prueba: La prueba se realizó de manera satisfactoria.</p>

Tabla 3.133.- Prueba de aceptación – generación de actas de Consejo.

Elaborado por: El investigador.

Prueba de aceptación	
Código: P28	Historia de usuario: H30
Nombre: Cerrar Consejos.	
Descripción: El usuario tiene la opción de cerrar el Consejo, con una alerta de confirmación.	
Condiciones de ejecución: El usuario debe tener creado documentos ya sea resoluciones o acuerdos, aparte de tener un responsable y haber tomado la asistencia al miembro.	
<p>Entrada/ Proceso de Ejecución:</p> <ul style="list-style-type: none"> -El usuario dará clic en el menú Consejos. -El usuario dará clic en el ícono amarillo, donde se cierra el Acta. -El usuario visualizará el mensaje de alerta si está seguro de cerrar ese Consejo. -El usuario tiene dos opciones, cerrar Consejo o cancelar. -Si el usuario cancela ya no puede generar actas. 	
Resultado esperado: El usuario puede cerrar Consejos.	
Evaluación de la prueba: La prueba se realizó de manera satisfactoria.	

Tabla 3.134.- Prueba de aceptación – cerrar consejos.

Elaborado por: El investigador.

CAPÍTULO IV.- CONCLUSIONES Y RECOMENDACIONES

4.1 Conclusiones

En la FISEI se maneja documentación variada para los trámites académicos administrativos por lo que se planteó el proyecto de crear un Sistema Web Integrado para la gestión de estos, con la finalidad de reducir tiempos de respuesta para las resoluciones de los estudiantes o docentes, notificando por correo institucional o número de WhatsApp, es por ello que se tiene las siguientes conclusiones:

- Una vez realizado el análisis de los procesos administrativos de la FISEI se concluye que, la información de docentes y estudiantes debe estar estructurada en una base de datos ya que esto permite tener datos fiables y de fácil consulta con ayuda de una interfaz web; por lo tanto, para la creación de un documento sea resolución o acuerdo se reduce los errores provocados por el mal ingreso de información, debido a que el sistema genera la documentación utilizando plantillas, que están compuestas por variables que serán reemplazadas a través de los buscadores óptimos.
- La tecnología Back-end que mejor se adaptó a los requerimientos del proyecto es el Framework de Laravel por su fácil integración con la API de Google Drive, la cual es una de las herramientas adecuadas para el desarrollo de la aplicación porque permite guardar los documentos en un alojamiento de archivos en la web.
- La tecnología Front-end que mejor se ajustó a los requerimientos del proyecto es: React js debido a que al ser unas de las bibliotecas más extensas y utilizadas del mercado se convierte en la herramienta apropiada para desarrollar interfaces interactivas e intuitivas del proyecto planteado, con un código flexible y bien estructurado que brindó la posibilidad de reutilizar código gracias a la generación de múltiples componentes, con la finalidad de crear aplicaciones con interfaz amigable para el usuario.
- Otra de las herramientas que fueron de mucha ayuda para el proyecto son las APIs de Google que constan de muchos productos y servicios como la API de Google Docs y Google Drive, que facilitó el manejo de la información documental de las diferentes áreas de la FISEI mediante editores de texto así

como también el almacenamiento de información en un alojamiento en la nube por lo que, su implementación en proyectos que manejen trámites académicos es de gran apoyo para la generación de documentos.

- El sistema web integrado se pudo implementar en la FISEI, en donde para llevar a cabo la generación de un acta de Consejo Directivo, la edición se realizaba de forma manual, en cada trámite académico ya sea resolución o acuerdo se copiaba y pegaba el cuerpo del documento en el acta; mientras que con la implementación del sistema se lo realiza de forma automática sin la necesidad de copiar los documentos uno a uno, de este modo se reduce los tiempos de respuesta y el exceso de trabajo operativo por parte de las secretarías, considerando que en cada consejo directivo se manejan muchos documentos.

4.2 Recomendaciones

- Utilizar tecnologías web nuevas, con una arquitectura escalable utilizando servicios web API RESTFULL, que permiten tener segmentado la parte del servidor y cliente, con una visión a futuro donde se desarrolle una aplicación móvil para que los estudiantes revisen sus trámites académicos, programando el consumo de los servicios en el lenguaje de programación escogido.
- Enlazar la base de datos de la Dirección de Tecnología de Información y Comunicación (DITIC) con el sistema implementado, con la finalidad de corroborar los datos de los estudiantes y obtener la información de forma rápida para el ingreso de estos.
- Se recomienda a futuro implementar el sistema web en todas las Facultades de la Universidad Técnica de Ambato porque sirve para reducir los tiempos de respuesta y mejoras en la atención a los usuarios.

BIBLIOGRAFÍA

- [1] C. Vera, «Desarrollo e implementación de un sistema web para el control de inventarios y alquiler de maquinarias de la empresa Megarent S.A.» Julio 2019. [En línea]. Available: <https://dspace.ups.edu.ec/bitstream/123456789/17525/1/UPS-GT002706.pdf>. [Último acceso: Julio 2022].
- [2] R. Mizanur, «Web System Development Process: The Complete Guide,» junio 2022. [En línea]. Available: <https://riseuplabs.com/web-system-development-process/>. [Último acceso: 2022].
- [3] K. Marín, Atencio y Edith, «Proceso Académico-Administrativo,» Omnia, vol. 14, n° 3, pp. 160-183, 2018.
- [4] F. Syafariani, «Web-Based Academic Information System,» IOP Conference Series: Materials Science and Engineering, pp. 1-6, Noviembre 2019.
- [5] A. Barrionuevo, «Sistema Automatizado para la gestión de trámites académicos en la secretaría de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial,» Universidad Técnica de Ambato, Ambato, 2021.
- [6] C. Gallardo y N. Andrés, «Desarrollo de un sistema infomático aplicando la metodología up, para optimizar la gestión de trámites académicos en la Universidad Nacional Chimborazo,» Universidad Nacional de Chimborazo, Riobamba, 2019.
- [7] W. d. I. C. Delgado, «Implementación de sistema de workflow para gestionar la información en el proceso de otorgamiento de crédito caso Cooperativa Pacífico,» Universidad San Ignacio de Lolaya, Lima, 2016.
- [8] F. Cosme, F. Polanco, J. García y E. Serrano, «Sistema informático para la gestión de procesos académicos y administrativos de la Escuela de Ingeniería de Sistemas Informáticos,» Universidad del Salvador, Salvador, 2019.
- [9] L. d. R. G. Chiluisa, «La gestión de archivos incide en la imagen institucional del Área administrativa de la unidad metropolitana de salud centro de la ciudad de quito,» Universidad Técnica de Ambato, Ambato, 2014.
- [10] J. Fernández, Gestión por procesos, Madrid: ESIC, 2009.
- [11] E. G. F. Sánchez, «Implementación de un sistema workflow en el proceso de adquisición de bienes y servicios en el departamento de Compras Públicas de la Universidad Estatal Amazónica de la ciudad de Puyo,» Universidad Católica del Ecuador, Ambato, 2015.
- [12] G. Martínez, G. Camacho y B. Guitiérrez, «Diseño de framework web para el desarrollo dinámico de aplicaciones,» Scientia Et Technica, vol. XVI, n° 44, pp. 178-183, 2010.

- [13] S. Carranco, «Desarrollo de un sistema web para la gestión y monitoreo de los procesos judiciales y de recursos humanos, con informes estadísticos de créditos, agencias y desempeño social para la Cooperativa de Ahorro y Crédito Mujeres Unidas,» Universidad Técnica del Norte, Ibarra, 2018.
- [14] A. Chandrababu, «Una comparación entre las metodologías de desarrollo de software ágiles y tradicionales,» Academia, p. 8, 2005.
- [15] J. Canós, P. Letelier y C. Penadés, Artists, Metodologías Ágiles en el Desarrollo de Software. [Art]. Universidad Politécnica de Valencia, 2012.
- [16] G. Mancuzo, «Metodología XP: La Mejor Vía para el Desarrollo de Software,» 8 agosto 2020. [En línea]. Available: <https://blog.comparasoftware.com/metodologia-xp/>. [Último acceso: 24 octubre 2022].
- [17] J. Joskowicz, Reglas y Prácticas en eXtreme Programming, España, 2008.
- [18] A. Sergio, «Ventajas y desventajas de Extreme Programming,» Septiembre 2018. [En línea]. Available: <https://iswugaps2extremeprogramming.wordpress.com/2015/09/14/ventajas-y-desventajas/>. [Último acceso: octubre 2021].
- [19] M. A. Awad, A Comparison between Agile and Traditional Software Development Methodologies, The University of Western Australia, 2005.
- [20] A. Navarro, J. Fernández y J. Morales, «Revisión de metodologías ágiles para el desarrollo de software,» Red de Revistas Científicas de América Latina, el Caribe, España y Portuga, vol. 11, nº 2, p. 11, 2018.
- [21] I. Sordo, «Metodología Scrum: qué es, cuáles son sus fases y cómo implementarla,» 23 enero 2021. [En línea]. Available: <https://blog.hubspot.es/marketing/metodologia-scrum>. [Último acceso: 2021].
- [22] J. Garzas, «Las metodologías Crystal. Otras metodologías ágiles que, quizás, te puedan encajar más que Scrum,» 25 septiembre 2020. [En línea]. Available: <https://www.javiergarzas.com/2012/09/metodologias-crystal.html>.
- [23] M. Temoc, «Crystal Methodologies,» 4 marzo 2018. [En línea]. Available: <http://crystalmethodologies.blogspot.com/>.
- [24] M. Montealegre, U. Sanchez, M. Torres y F. Tivar, «Scrum & Crystal,» Ecuador, 2017.
- [25] B. Saurabh, «React Advantages and Disadvantages- why React.js,» 14 Marzo 2022. [En línea]. Available: <https://aglowiditsolutions.com/blog/pros-and-cons-of-reactjs>.
- [26] M. Parada, «React Js,» 21 mayo 2021. [En línea]. Available: <https://openwebinars.net/blog/que-es-react/>.

- [27] GEEK, «React JS,» 26 mayo 2020. [En línea]. Available: <https://ifgeekthen.nttdata.com/es/que-es-reactjs-conceptos-basicos>.
- [28] A. Thompson, «¿Cuáles son las ventajas y desventajas de React JS?,» 10 septiembre 2022. [En línea]. Available: <https://www.affiliatebay.net/es/advantages-and-disadvantages-of-react-js/>.
- [29] Altexsoft, «The Good and the Bad of Vue.js Framework Programming,» 11 Septiembre 2019. [En línea]. Available: <https://www.altexsoft.com/blog/engineering/pros-and-cons-of-vue-js/>. [Último acceso: 2022].
- [30] J. Moncayo, «Características de Vue.JS,» 26 octubre 2017. [En línea]. Available: <https://openwebinars.net/blog/que-es-vuejs/>.
- [31] A. Vega, «¿Por qué Vue.js está ganando tanta popularidad?,» 7 enero 2018. [En línea]. Available: <https://adrianalonso.es/desarrollo-web/frontend/frameworks-javascript/por-que-vue-js-esta-ganando-tanta-popularidad/>.
- [32] D. Grankin, «Pros and Cons of Vue.js Framework Programming,» Diciembre 2020. [En línea]. Available: <https://ddi-dev.com/blog/programming/the-good-and-the-bad-of-vue-js-framework-programming/>. [Último acceso: 2022].
- [33] J. Beres, Blazor vs Angular Comparison in 20, Estados Unidos : Infragistics, 2021.
- [34] V. Guzman, «Introducción a Blazor,» 17 agosto 2020. [En línea]. Available: <https://es.slideshare.net/LucioMSP/introduccion-a-blazor>.
- [35] A. Singh, «Comparison Of Blazor Vs JavaScript Frameworks,» Zenesys, Texas, 2021.
- [36] H. Pérez, «El ecosistema web hoy en día,» El camino DEV, Perú, 2022.
- [37] Erfan, «Advantages and disadvantages of Laravel,» 22 Febrero 2021. [En línea]. Available: <https://ded9.com/advantages-and-disadvantages-of-laravel/>. [Último acceso: 2022].
- [38] A. Troelsen y P. Japikse, «Introducing ASP.NET,» Noviembre 2017. [En línea]. Available: https://www.researchgate.net/publication/321187989_Introducing_ASPNET_MVC. [Último acceso: 2022].
- [39] NavicoSoft, «Advantages and disadvantages of ASP.NET core framework?,» ASP.NET Core, Estados Unidos, 2022.
- [40] D. Roth y R. Anderson, «Introduction to ASP.NET Core,» Microsoft, Estados Unidos, 2020.
- [41] Node.Js, «Node.js - Introduction,» Tutorialspoint, Estados Unidos, 2020.
- [42] S. Sethiya, «Advantages & Disadvantages of Using @Nodejs,» Medium, Estados Unidos,

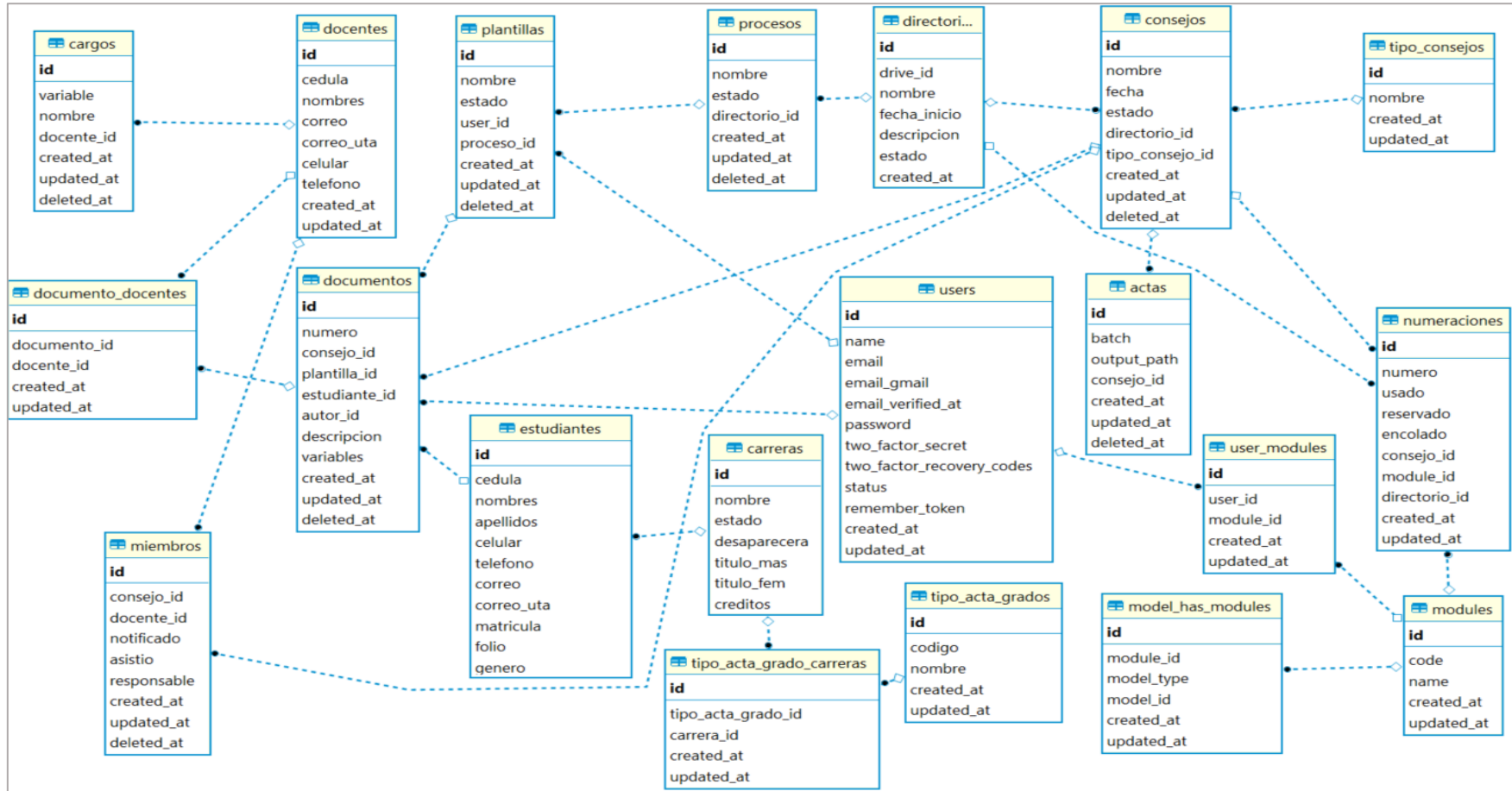
2018.

- [43] Techopedia, «Application Programming Interface (API),» 27 Febrero 2022. [En línea]. Available: <https://www.techopedia.com/definition/24407/application-programming-interface-api>. [Último acceso: 2022].
- [44] Google Drive, «Introduction to Google Drive API,» 7 Abril 2022. [En línea]. Available: <https://developers.google.com/drive/api/guides/about-sdk>. [Último acceso: 2022].
- [45] C. Castillo, K. Cancino, V. García y A. De la Cruz, «Diseño de un Sistema web para el control de Curriculum Vitae Electrónico de personal docente basado en una arquitectura orientada a servicios (API REST),» *Revista de Investigación en Tecnologías de la Información*, vol. 10, n° 20, pp. 28-42, 2022.
- [46] S. Anshu y R. Virender, «API Features Individualizing of Web Services: REST and SOAP,» *International Journal of Innovative Technology and Exploring Engineering*, vol. 8, n° 9, pp. 664-671, 2019.
- [47] J. Lozano, «Creación y gestión de una base de datos con MYSQL y PHPMYADMIN,» 2018. [En línea]. Available: [Creación y gestión de una base de datos con MYSQL y PHPMYADMIN](#). [Último acceso: 2022].
- [48] Vargas y Maltés, «Programación en Capas,» *Universidad de Costa Rica*, n° 506, pp. 1-5, 2018.
- [49] D. Cardacci, «Arquitectura de software académica para la comprensión del desarrollo de software en capas,» *Universidad del Centro de Estudios Macroeconómicos de Argentina*, n° 574, pp. 1-23, 2019.
- [50] Instituto Tecnológico de Matehuala, «Arquitectura de las aplicaciones Web,» 22 Enero 2022. [En línea]. Available: <https://programacionwebisc.wordpress.com/2-1-arquitectura-de-las-aplicaciones-web/>. [Último acceso: 2022].
- [51] J. Angulo, «Evaluación de la Eficiencia de las Tecnologías GraphQL y REST en la implementación de servicios web consumidos por aplicaciones Android,» 2020. [En línea]. Available: <https://repositorio.uss.edu.pe/bitstream/handle/20.500.12802/6729/Angulo%20Vásquez%20Jorge%20Luis%20Mateo.pdf?sequence=1&isAllowed=y..> [Último acceso: 2022].
- [52] R. Camacho, «Sistema Integrado,» 17 Junio 2021. [En línea]. Available: <https://es.parasoft.com/blog/what-is-an-embedded-system/#:~:text=Los%20sistemas%20integrados%20son%20sistemas,tienen%20una%20funci%C3%B3n%20operativa%20dedicada..> [Último acceso: 13 Julio 2022].
- [53] E. Ávila y A. Meneses, «Comparative Evaluation of Delfdroid whit XP and Scrum using the 4-DAT,» *Scielo*, vol. 7, n° 1, pp. 16-23, Marzo 2018.

- [54] Tycoonstory, «The Advantages and Disadvantages Of .NET Framework Programming,» 10 Febrero 2022. [En línea]. Available: <https://ddi-dev.com/blog/programming/the-good-and-the-bad-of-vue-js-framework-programming/>. [Último acceso: 2022].
- [55] A. Melnichuk, «Pros and Cons of .NET Framework,» 24 Julio 2020. [En línea]. Available: <https://ncube.com/blog/pros-and-cons-of-net-framework>. [Último acceso: 2022].
- [56] Fercalan, «Cuadro comparativo entre metodologías SCRUM, XP & RUP,» 17 Noviembre 2018. [En línea]. Available: <https://fercalan1598.blogspot.com/2017/>. [Último acceso: Julio 2022].
- [57] Tech Vidvad, «Python Advantages and Disadvantages – Step in the right direction,» Python Tutorials, Estados Unidos, 2022.
- [58] IntelliPaat, «Advantages and Disadvantages of Python,» Python Estudios, Canadá, 2022.
- [59] J. Tudor, Python para principiantes, Venezuela: Millennium Plubishing, 2021.

ANEXOS

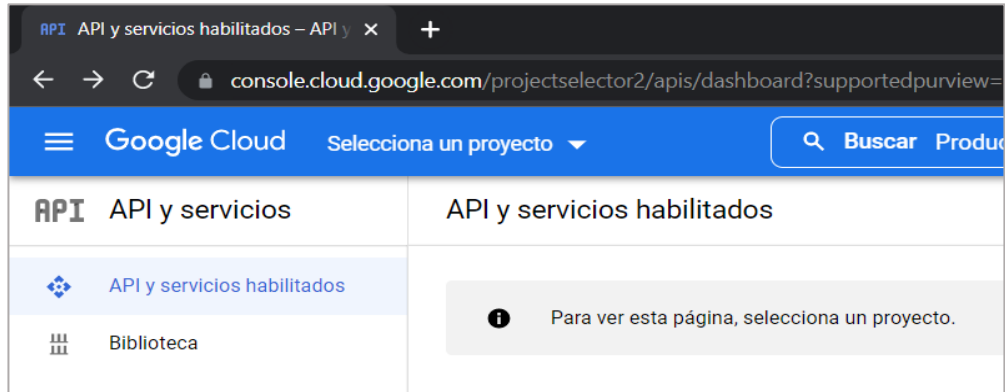
Anexo 1. Base de datos.



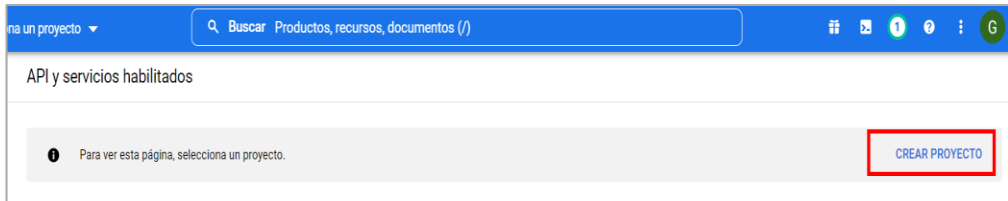
Anexo 2. Configuración de API de Google.

Antes de la configuración de la API de Google se debe realizar los siguientes pasos

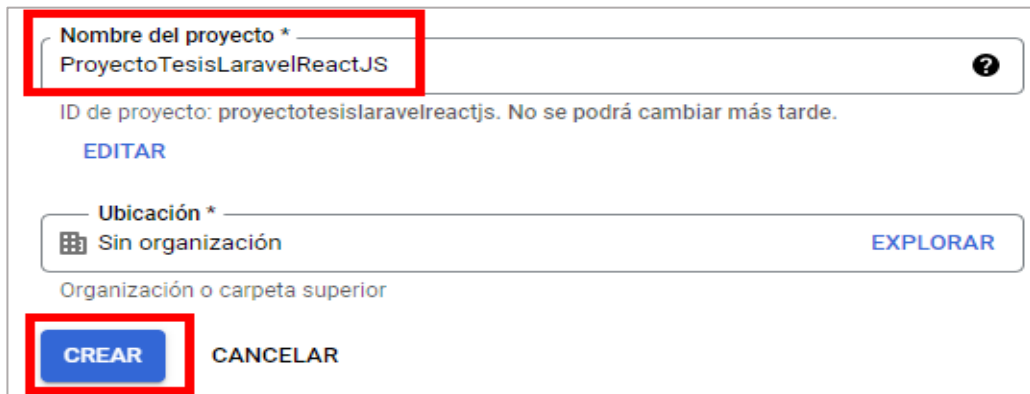
1.- Se debe ingresar en la página de consola de Google Cloud.



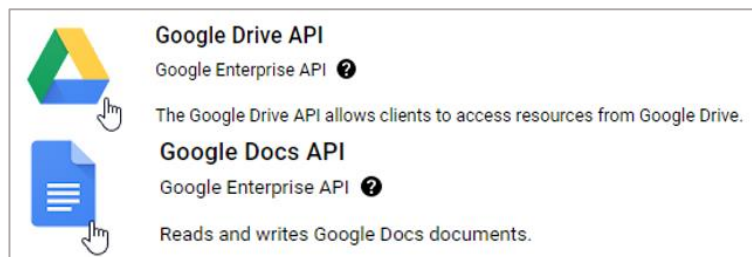
2.- Se empieza creando un proyecto.



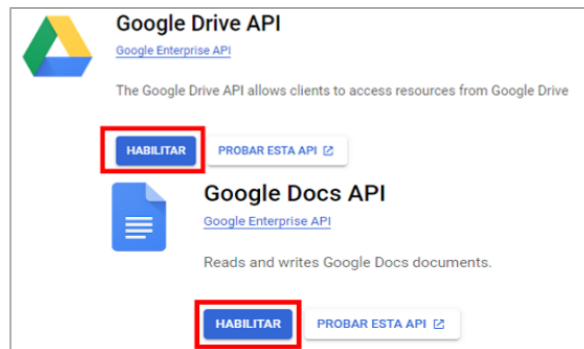
3.- Se agrega el nombre del proyecto y se procede a crear.



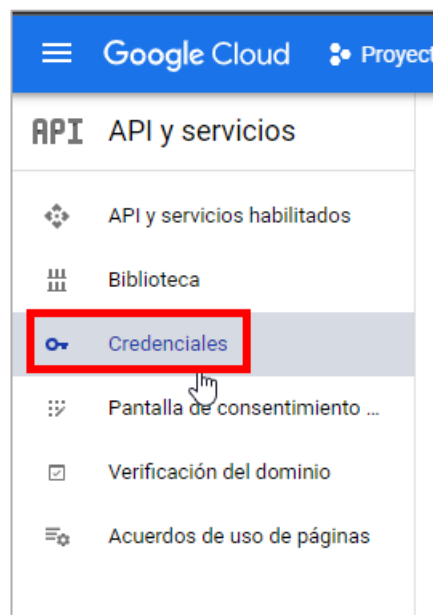
4.- Las API a utilizar son Google Drive Api y Google Docs API.



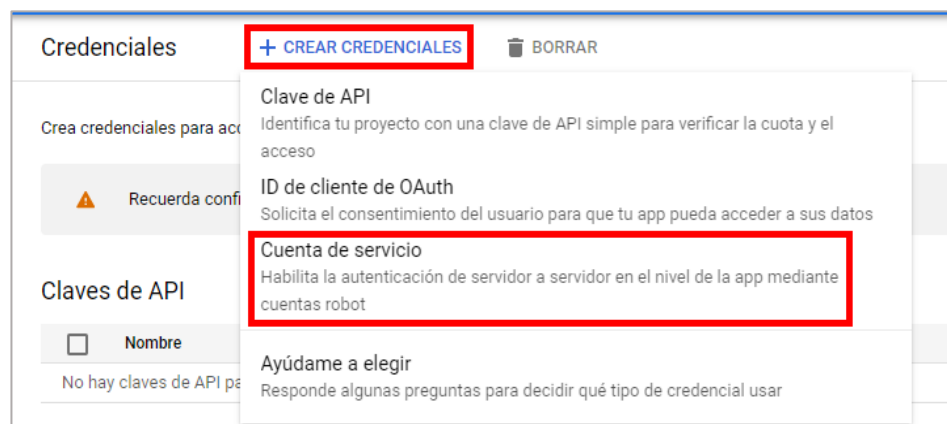
5.-Se procede a habilitar dichas API.



6.-Dentro del panel de API y Servicios, se procede a seleccionar credenciales.



7.- Se da clic en crear credenciales, para luego seleccionar Cuenta de servicio.



8.-Se crea el nombre de la cuenta de servicio, para que cree una cuenta de gmail, en el que se va a dar permisos para almacenar los documentos de google drive.

1 Detalles de la cuenta de servicio

Nombre de la cuenta de servicio
Mostrar nombre de esta cuenta de servicio

ID de la cuenta de servicio * X ↺

Dirección de correo electrónico:
 📧

Descripción de la cuenta de servicio
Describe lo que hará esta cuenta de servicio

CREAR Y CONTINUAR

9.-Una vez creado aparece de la siguiente forma, en donde puede editar o borrar dicha cuenta de servicio.

Cuentas de servicio [Administrar cuentas de servicio](#)

<input type="checkbox"/>	Correo electrónico	Nombre ↑	Acciones
<input type="checkbox"/>	<input type="text" value="projectplantillasfisei@projectotesislaravelreactjs.iam.gserviceaccount.com"/>	ProyectoPlantillasFisei	✎ 🗑️

10.-Dentro de Google Drive, se comparte la carpeta con la cuenta de servicio creada anteriormente.

Compartir "Documentos FISEI" ? ⚙️

Añadir personas y grupos

Personas con acceso

- G [User] Propietario
- P Editor

Acceso general

🔒 Restringido ▼
Solo los usuarios con acceso pueden abrir el enlace

[🔗 Copiar enlace](#) Hecho

11.-Se crea una clave para el proyecto.

The screenshot shows the 'Claves' (Keys) tab in the 'ProyectoPlantillasFisei' project settings. The 'CLAVES' tab is highlighted with a red box. Below the tabs, there is a warning message about service account keys and a dropdown menu labeled 'AGREGAR CLAVE' (Add Key) with a red box around it. The dropdown menu is open, showing two options: 'Crear clave nueva' (Create new key) and 'Subir clave existente' (Upload existing key), both with red boxes around them. The 'Crear clave nueva' option is selected. Below the dropdown, there are columns for 'Fecha de creación de la clave' (Key creation date) and 'Fecha de vencimiento de la clave' (Key expiration date).

12.-Se crea una clave con formato JSON.

The screenshot shows the 'Crear clave privada para "ProyectoPlantillasFisei"' (Create private key for "ProyectoPlantillasFisei") dialog box. The dialog box contains instructions to download a private key file and store it in a secure location. Below the instructions, there is a 'Tipo de clave' (Key type) section with two radio button options: 'JSON' (Recommended) and 'P12' (For backward compatibility with code in P12 format). The 'JSON' option is selected and highlighted with a red box. At the bottom right of the dialog box, there are 'CANCELAR' and 'CREAR' buttons, with the 'CREAR' button highlighted with a red box. Below the dialog box, a message states 'Se guardó la clave privada en tu computadora' (Private key saved to your computer) and provides a warning icon and a link to 'Más información sobre las prácticas recomendadas' (More information about recommended practices). A 'CERRAR' (Close) button is located at the bottom right of the message area.

13.-Estas credenciales se reemplazan en el archivo de configuración. ENV del proyecto Back End.

```
{
  "type": "application",
  "project_id": "my-project-123456789",
  "private_key_id": "my-private-key-id",
  "private_key": "-----BEGIN PRIVATE KEY-----\n",
  "client_email": "my-client-email@my-project-123456789.iam.gcp.io",
  "client_id": "my-client-id",
  "auth_uri": "https://accounts.google.com/o/oauth2/auth",
  "token_uri": "https://accounts.google.com/o/oauth2/token",
  "auth_provider_x509_cert_url": "https://www.googleapis.com/oauth2/v1/certs",
  "client_x509_cert_url": "https://www.googleapis.com/oauth2/v1/certs"
}

GOOGLE_DRIVE_ROOT_DIRECTORY_ID=

GOOGLE_APPLICATION_CREDENTIALS_TYPE=
GOOGLE_APPLICATION_CREDENTIALS_PROJECT_ID=
GOOGLE_APPLICATION_CREDENTIALS_PRIVATE_KEY_ID=
GOOGLE_APPLICATION_CREDENTIALS_PRIVATE_KEY=
GOOGLE_APPLICATION_CREDENTIALS_CLIENT_EMAIL=
GOOGLE_APPLICATION_CREDENTIALS_CLIENT_ID=
GOOGLE_APPLICATION_CREDENTIALS_AUTH_URI=
GOOGLE_APPLICATION_CREDENTIALS_TOKEN_URI=
GOOGLE_APPLICATION_CREDENTIALS_AUTH_PROVIDER_X509_CERT_URL=
GOOGLE_APPLICATION_CREDENTIALS_CLIENT_X509_CERT_URL=

GOOGLE_DRIVE_ROOT_DIRECTORY_ID="my-google-drive-root-directory-id"

GOOGLE_APPLICATION_CREDENTIALS_TYPE="application"
GOOGLE_APPLICATION_CREDENTIALS_PROJECT_ID="my-project-123456789"
GOOGLE_APPLICATION_CREDENTIALS_PRIVATE_KEY_ID="my-private-key-id"
GOOGLE_APPLICATION_CREDENTIALS_PRIVATE_KEY="-----BEGIN PRIVATE KEY-----\n"
GOOGLE_APPLICATION_CREDENTIALS_CLIENT_EMAIL="my-client-email@my-project-123456789.iam.gcp.io"
GOOGLE_APPLICATION_CREDENTIALS_CLIENT_ID="my-client-id"
GOOGLE_APPLICATION_CREDENTIALS_AUTH_URI="https://accounts.google.com/o/oauth2/auth"
GOOGLE_APPLICATION_CREDENTIALS_TOKEN_URI="https://accounts.google.com/o/oauth2/token"
GOOGLE_APPLICATION_CREDENTIALS_AUTH_PROVIDER_X509_CERT_URL="https://www.googleapis.com/oauth2/v1/certs"
GOOGLE_APPLICATION_CREDENTIALS_CLIENT_X509_CERT_URL="https://www.googleapis.com/oauth2/v1/certs"

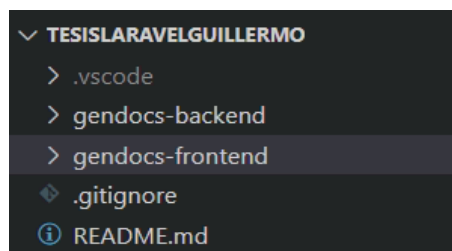
DEFAULT_TEMPLATE="my-default-template"
```


Anexo 3. Codificación.

El proyecto se divide en dos secciones, la primera se denomina gendocs-backend en donde se tiene toda la lógica de negocio, la conexión de base de datos y configuración de la API de Google. Mientras que la segunda sección corresponde a gendocs-frontend es donde se encuentran todas las interfaces de usuario que se visualizan en un navegador web.

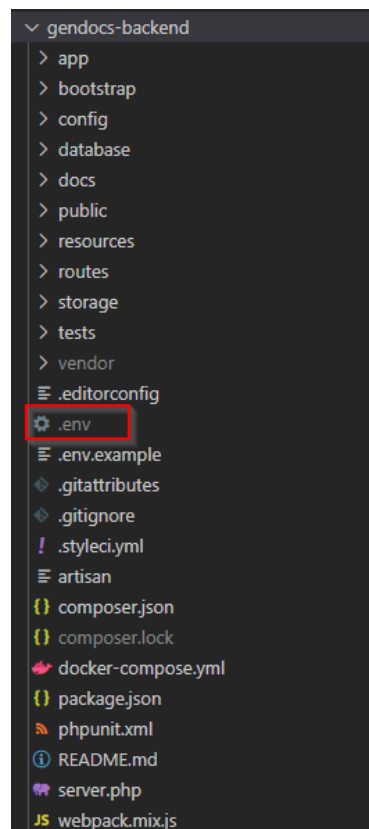
a) Código del proyecto

Se compone por dos proyectos: gendocs-backend que es la lógica de negocio y gendocs-frontend que son donde se realizan las interfaces de usuario.



En un proyecto realizado en Laravel, lo primero a configurar es el archivo .env.

b) Configuración archivo. Env



El archivo .env es la configuración que tiene Laravel, aquí se configuran las API a consumir, base de datos, puertos, servidor de Mail, entre otros.

c) Configuración de las API

Se reemplazan los valores para la puesta a producción del sistema web.

```
APP_NAME=Laravel
APP_ENV=local
APP_KEY=base64:3IzV7QGAXTu8cYeWsq7Srh2RoeXWnFOimwTMS0+vPyA=
APP_DEBUG=true
APP_URL=http://localhost:3001

LOG_CHANNEL=stack
LOG_DEPRECATIONS_CHANNEL=null
LOG_LEVEL=debug

DB_CONNECTION=mysql
DB_HOST=mysql
DB_PORT=3306
DB_DATABASE=gendocs_backend
DB_USERNAME=root
DB_PASSWORD=

BROADCAST_DRIVER=log
CACHE_DRIVER=file
FILESYSTEM_DRIVER=local
QUEUE_CONNECTION=database
SESSION_DRIVER=file
SESSION_LIFETIME=120

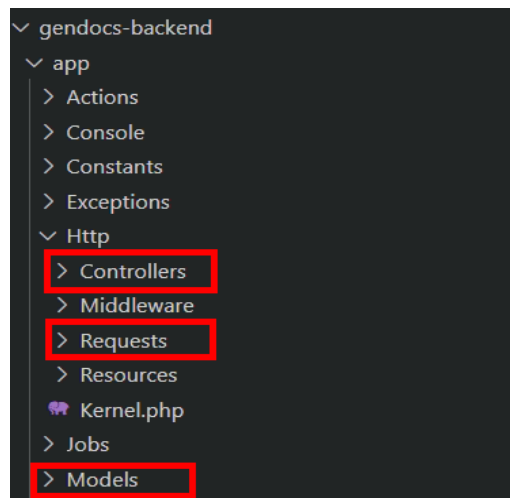
MEMCACHED_HOST=127.0.0.1

REDIS_HOST=127.0.0.1
REDIS_PASSWORD=null
REDIS_PORT=6379

MAIL_MAILER=smtp
MAIL_HOST=smtp.mailtrap.io
MAIL_PORT=2525
MAIL_USERNAME=7e2b19129014a0
MAIL_PASSWORD=ba4bd84624f915
MAIL_ENCRYPTION=tls
MAIL_FROM_ADDRESS=null
MAIL_FROM_NAME="${APP_NAME}"
```

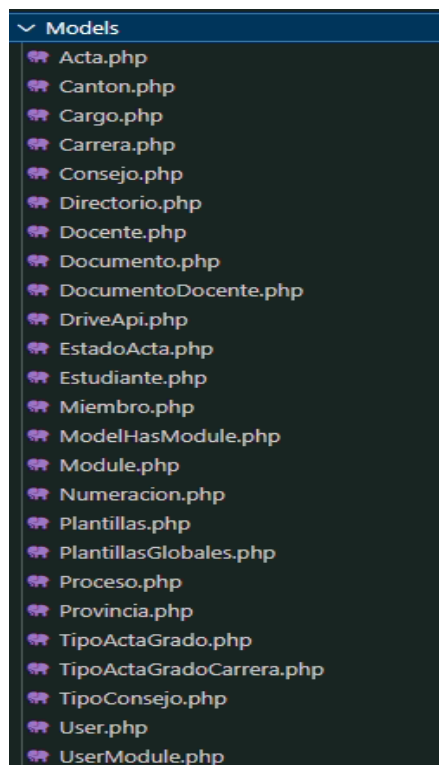
Carpeta de los modelos, controladores y solicitudes (Models- Controllers - Request)

En estas carpetas se encuentra la lógica y reglas de negocio, que permiten realizar los servicios web para su posterior consumo.



Modelos

En la siguiente figura se muestran los siguientes modelos que son caracterizados por ser clases que hacen consultas en la base de datos.



Ejemplo de un modelo de la clase docente, en donde constan los campos que representan a esta tabla.

```
class Docente extends Model
{
    use HasFactory, Filterable;

    protected $fillable = [
        'cedula', 'nombres', 'correo',
        'correo_uta', 'celular', 'telefono',
    ];

    public const FILTERS = ['search'];

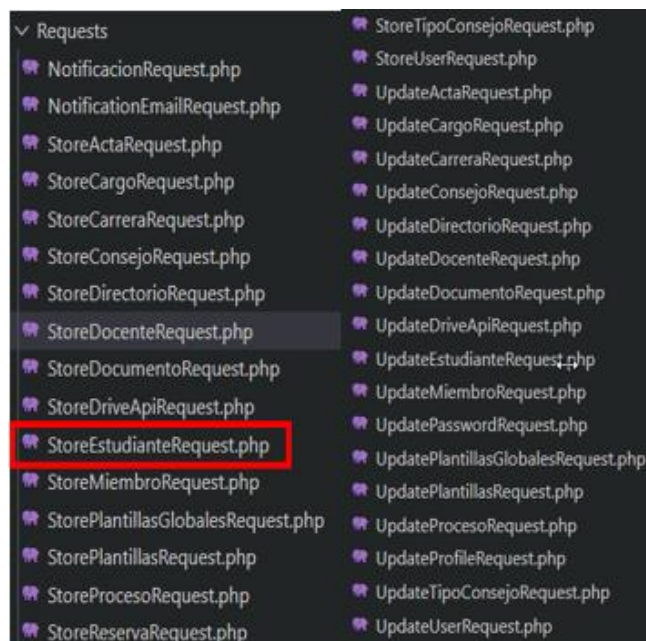
    public function fields()
    {
        return [
            'id' => $this->id,
            "cedula" => $this->cedula,
            "nombres" => $this->nombres,
            "celular" => $this->celular,
            "telefono" => $this->telefono,
            "correo" => $this->correo,
            "correo_uta" => $this->correo_uta,
        ];
    }

    public function scopeSearch(Builder $query, $filter)
    {
        $filter = preg_replace('/\s+/', '%', $filter);

        return $query
            ->where('cedula', 'like', "%$filter%")
            ->orWhere("nombres", 'like', "%$filter%");
    }
}
```

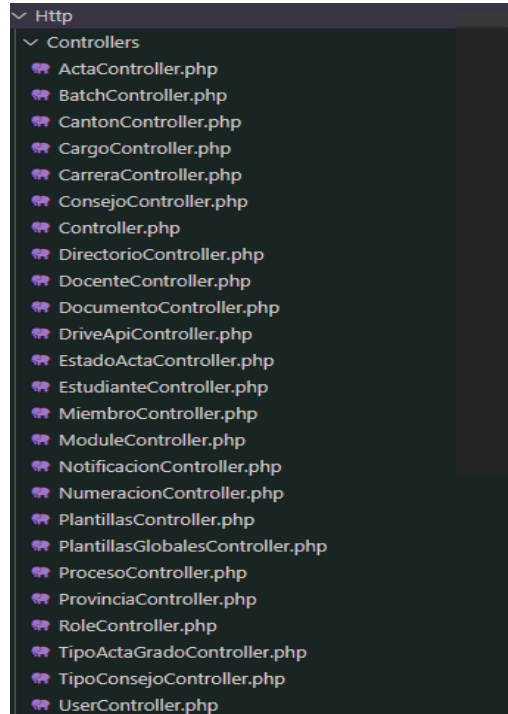
Requests.

Los requests permiten obtener la información sobre el cliente que realiza la solicitud y los datos que pueda estar enviando.



Controladores

En los controladores se encuentra toda la lógica del negocio, en donde se crean los métodos API REST (GET, POST, PUT, DELETE) de cada uno de los módulos que están compuestos en el sistema tal como lo indica la siguiente figura.



Controlador de usuarios

En esta sección se tiene un constructor para autorizar al usuario a ingresar en la API de Google, además de un servicio web para obtener su propio usuario y mostrar la colección de usuarios.

```
class UserController extends Controller
{
    protected GoogleDriveService $googleDriveService;

    public function __construct(GoogleDriveService $googleDriveService)
    {
        $this->authorizeResource(User::class);
        $this->googleDriveService = $googleDriveService;
    }

    public function me(Request $request)
    {
        return ResourceObject::make($request->user());
    }

    public function index()
    {
        return ResourceCollection::make(User::all());
    }
}
```

Controlador de usuarios

A continuación, se observa la lógica para crear un usuario.

```
public function store(StoreUserRequest $request)
{
    $validated = $request->validated();

    try {
        DB::beginTransaction();

        $role = Role::find($validated['rol']);

        $tempPassword = Str::random(8);

        $userCreated = User::create([
            'name' => $validated['nombre'],
            'email' => $validated['correo_principal'],
            'email_gmail' => $validated['correo_secundario'],
            'password' => Hash::make($tempPassword),
        ]);

        $userCreated->assignRole($role);
        $userCreated->modules()->attach($validated['modulos']);

        $userCreated->notify(new UserCreated($request->user(), $tempPassword));

        $permission = $this->googleDriveService->shareFolder(
            $userCreated->email_gmail,
            Directorio::query()->activeDirectory()->drive_id,
            $role->name_role_drive,
        );

        $userCreated->permission()->create([
            'google_drive_id' => $permission->id,
        ]);

        DB::commit();

        return ResourceObject::make($userCreated);
    } catch (\Exception $e) {
        DB::rollBack();
        return response()->noContent(ResponseAlias::HTTP_UNPROCESSABLE_ENTITY);
    }
}
```

Actualización de usuarios

Se evidencia la lógica para actualizar usuarios, teniendo en cuenta los permisos que requiere dar el api de Google drive.

```
public function update(UpdateUserRequest $request, User $user)
{
    $validated = $request->validated();

    try {
        DB::beginTransaction();

        $role = Role::find($validated['rol']);

        $user->fill([
            'name' => $validated['nombre'],
            'status' => $validated['status'],
            'email' => $validated['correo_principal'],
            'email_gmail' => $validated['correo_secundario'],
        ]);

        $user->modules()->sync($validated['modulos']);

        $needsNewAccess = false;

        if ($user->isDirty('status')) {
            if ($user->status) {
                $needsNewAccess = true;
            } else {
                // Remove old email
                $permission = $user->permission;
                if ($permission) {
                    $this->googleDriveService->deletePermission(
                        Directorio::query()->activeDirectory()->drive_id,
                        $user->permission->google_drive_id
                    );

                    $permission->delete();
                }
            }
        }

        if (
            !$user->hasRole($role) ||
            $user->isDirty('email_gmail') ||
            $needsNewAccess
        ) {
```

```
            // Remove old email
            $permission = $user->permission;
            if ($permission) {
                $this->googleDriveService->deletePermission(
                    Directorio::query()->activeDirectory()->drive_id,
                    $user->permission->google_drive_id
                );

                $permission->delete();
            }

            // Add new permissions to new email
            $permission = $this->googleDriveService->shareFolder(
                $user->email_gmail,
                Directorio::query()->activeDirectory()->drive_id,
                $role->name_role_drive,
            );

            // Update relationship table
            $user->permission()->create([
                'google_drive_id' => $permission->id,
            ]);

            // Add new roles
            $user->roles()->detach();
            $user->assignRole($role);
        }

        $user->save();

        DB::commit();

        return ResourceObject::make($user);
    } catch (\Exception $e) {
        DB::rollBack();
        return response()->json([
            'errors' => $e->getMessage(),
        ], ResponseAlias::HTTP_UNPROCESSABLE_ENTITY);
    }
}
```

Actualización de la contraseña

Se muestra la lógica para actualizar la contraseña validando el request, ingresando la respectiva contraseña.

```
public function updatePassword(UpdatePasswordRequest $request)
{
    $validated = $request->validated();
    $user = \request()->user();

    if (!isset($validated['current_password']) ||
        !Hash::check($validated['current_password'], $user->password)) {
        return response()->json([
            'errors' => trans('validation.custom.user.update.password')
        ], ResponseAlias::HTTP_UNPROCESSABLE_ENTITY);
    }

    $user->forceFill([
        'password' => Hash::make($validated['password']),
    ]->save();

    return response()->noContent(ResponseAlias::HTTP_OK);
}
```

Restablecer contraseña

La lógica para restablecer contraseña se evidencia a continuación, donde se busca el email y se actualiza la contraseña de email.

```
public function resetPassword(Request $request)
{
    $email = $request->email;

    $user = User::whereEmail($email)->first();

    if (!$user) {
        return;
    }

    $tempPassword = Str::random(8);

    $user->update([
        'password' => Hash::make($tempPassword),
    ]);

    $user->notify(new UserResetPassword($request->user(), $tempPassword));
}
```

Controlador de acta

Consta de su método index que permite filtrar un acta mediante el request que se envié, además de un método store que permite guardar un acta.


```

class ActaController extends Controller
{
    public function index(Request $request)
    {
        $query = Acta::query();

        $query->applyFilters($request->all());

        return ResourceCollection::make($query->get());
    }

    public function store(StoreActaRequest $request,
                        ActaService $actaService)
    {
        try {
            $validated = $request->validated();
            $consejo = Consejo::find($validated['consejo']);
            $acta = $actaService->procesarDocumentos($consejo);

            return ResourceObject::make($acta);
        } catch (\Exception $e) {
            return response()->json([
                'errors' => $e->getMessage(),
            ], Response::HTTP_UNPROCESSABLE_ENTITY);
        }
    }
}

```

Método show y descargar de controlador acta

Se evidencia dos métodos: show y descargar. El primer método sirve para mostrar las actas creadas en la base de datos, en cambio el segundo método almacena dentro del servidor en la carpeta storage la información de los documentos generados (Resoluciones o Acuerdos) y la unificación de ellos.

```

public function show(Acta $acta)
{
    return ResourceObject::make($acta);
}

public function descargar(Acta $acta)
{
    $dir = $acta->output_path;
    $files = glob(storage_path("app/" . $dir) . '/generados/*.docx');

    if (!count($files)) {
        return response()->noContent(Response::HTTP_NOT_FOUND);
    }

    $file = "/unificado.docx";
    $dm = new DocxMerge();
    $dm->merge($files, storage_path("app/" . $dir . $file));
    $contentType = MimeType::DOCX;

    return Storage::download($dir . $file, pathinfo($file)['basename'],
        ["Content-Type: $contentType"]);
}

```

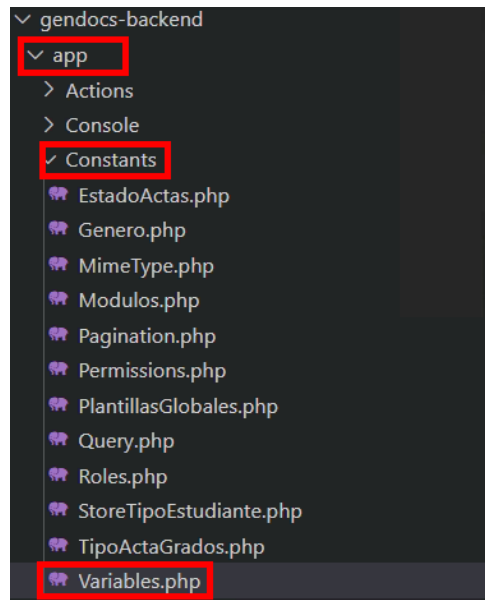
Método crear plantilla

Se muestra un método crear plantilla para guardar las actas en el repositorio de Google Drive.

```
public function crearPlantilla(Acta $acta,
                               ActaService $actaPlantillaService)
{
    try {
        $acta = $actaPlantillaService->generarPlantilla($acta);
        return ResourceObject::make($acta);
    } catch (\Exception $e) {
        return response()->json([
            'errors' => $e->getMessage(),
        ], Response::HTTP_UNPROCESSABLE_ENTITY);
    }
}
```

Constants

Por otra parte, dentro de carpeta app, se tiene una subcarpeta llamada Constants (Constante) la cual contiene una clase denominada Variables.php, y muestra las parametrizaciones de las etiquetas que son utilizadas para la generación de documentos, se puede evidenciar en la siguiente figura.



Ejemplo, clase de variables

Se evidencia los comandos que son utilizados para la generación de documentos.

```

abstract class Variables
{
    const FROM = '{{FROM}}';
    const TO = '{{TO}}';

    /**
     * Variables utilizadas para reemplazar datos para CONSEJO
     */
    const PREFIX_CONSEJO = 'consejo';
    const FECHA = "{{FECHA}}";
    const FECHAUP = "{{FECHAUP}}";
    const SESION = "{{SESION}}";
    const RESPONSABLE = "{{RESPONSABLE}}";

    /**
     * Variables utilizadas para reemplazar datos para información en GENERAL
     */
    const PREFEX_GENERAL = 'general';
    const CREADOPOR = "{{CREADOPOR}}";
    const NUMDOC = "{{NUMDOC}}";

    /**
     * Variables utilizadas para reemplazar datos de DOCENTES
     */
    const PREFIX_DOCENTES = 'docentes';
    const DOCENTE_N = "{{DOCENTE_N_$i}}";

    /**
     * Variables utilizadas para reemplazar datos para de ESTUDIANTE
     */
    const PREFIX_ESTUDIANTE = 'estudiante';
    const ESTUDIANTE = "{{ESTUDIANTE}}";
    const ESTUDIANTEUP = "{{ESTUDIANTEUP}}";
    const CEDULA = "{{CEDULA}}";
    const MATRICULA = "{{MATRICULA}}";
    const FOLIO = "{{FOLIO}}";
    const TELEFONO = "{{TELEFONO}}";
    const CELULAR = "{{CELULAR}}";
    const CORREO = "{{CORREO}}";
    const CORREOUTA = "{{CORREOUTA}}";
    const NOMBRECARRERA = "{{NOMBRECARRERA}}";
    const NOMBRECARRERAUP = "{{NOMBRECARRERAUP}}";
}

```

Variables utilizadas para generar la plantilla del acta

Cada una de estas variables se usan en la generación de plantillas al momento de crear un documento ya sea una resolución o acuerdo.

```

/**
 * Variables utilizadas para generar la plantilla del acta
 */
const FECHA_U = "{{FECHA_U}}"; // 16 DE JULIO DE 2021
const SESIONUP = "{{SESIONUP}}"; // ORDINARIA
const SESION_L = "{{SESION_L}}"; // ordinaria
const NUMACT = "{{NUMACT}}"; // 001
const Y = "{{Y}}"; // 2022
const DIASEM_T = "{{DIASEM_T}}"; // VIERNES
const NUMMES_T_U = "{{NUMMES_T_U}}"; // JULIO
const MES_T_L = "{{MES_T_L}}"; // julio
const NUMDIA_T = "{{NUMDIA_T}}"; // VEINTICINCO
const NUMANIO_T = "{{NUMANIO_T}}"; // DOS MIL VEINTIUNO
const NUMANIO_T_L = "{{NUMANIO_T_L}}"; // dos mil veintiuno
const DIAS_T = "{{DIAS_T}}"; // veinticinco días
const HORA_T_L = "{{HORA_T_L}}"; // ocho
const MINUTOS_T_L = "{{MINUTOS_T_L}}"; // treinta
const ASISTIERON = "{{ASISTIERON}}"; // miembros que asistieron
const NO_ASISTIERON = "{{NO_ASISTIERON}}"; // miembros que no asistieron

```

Roles del sistema

Los roles me permiten dar permisos de escritura y lectura para los archivos de Google Docs y Google Drive.

```
abstract class Roles
{
    /**
     * ROLES DEL SISTEMA
     *
     * Los roles declarados a continuación deben de estar mapeados
     * exactamente en el lado del cliente (front-end).
     */
    const Admin = 'Administrador';
    const AdminTemp = 'Administrador-Temporal';
    const Reader = 'Lector';
    const Writer = 'Escritor';

    /**
     * ROLES PARA GOOGLE DRIVE
     *
     * Estos roles son utilizados para otorgar permisos de lectura/escritura
     * por el servicio de google drive y sus dependencias.
     */
    const GReader = 'reader';
    const GWriter = 'writer';
}
```

Migraciones de base de datos

En el proyecto de Laravel es importante la sección de migraciones, ya que en esta se tiene un control de versiones de la base de datos, todos los campos y tablas que conforman la estructura del sistema, tal como lo muestra a continuación.

```
▼ migrations
  2014_05_21_213752_create_modules_table.php
  2014_10_12_000000_create_users_table.php
  2014_10_12_100000_create_password_resets_table.php
  2014_10_12_200000_add_two_factor_columns_to_users_table.php
  2019_08_19_000000_create_failed_jobs_table.php
  2019_12_14_000001_create_personal_access_tokens_table.php
  2021_11_24_025725_create_directorios_table.php
  2021_11_24_025726_create_carreras_table.php
  2021_11_24_030355_create_estudiantes_table.php
  2021_11_24_032134_create_procesos_table.php
  2021_11_24_032141_create_plantillas_table.php
  2021_11_24_124556_create_tipo_consejos_table.php
  2021_11_24_124557_create_consejos_table.php
  2021_11_28_014023_create_actas_table.php
```

```

2022_01_30_203653_create_permission_tables.php
2022_02_27_235525_create_drive_apis_table.php
2022_03_13_180132_create_documentos_table.php
2022_03_14_030801_create_numeraciones_table.php
2022_03_26_145835_create_docentes_table.php
2022_03_26_145837_create_miembros_table.php
2022_04_10_162549_create_jobs_table.php
2022_04_10_172028_create_job_batches_table.php
2022_04_22_133440_create_plantillas_globales_table.php
2022_05_21_213814_create_user_modules_table.php
2022_06_04_180403_create_documento_docentes_table.php
2022_06_09_210616_create_model_has_modules_table.php
2022_07_26_012600_create_cargo_table.php

```

Ejemplo de una migración

El ejemplo de una migración, en este caso la tabla docente, crea la tabla con sus respectivos campos o elimina si es que ya existe.

```

class CreateDocentesTable extends Migration
{
    /** Run the migrations. */
    public function up()
    {
        Schema::create('docentes',
            function (Blueprint $table) {
                $table->id();

                $table->string('cedula')->unique();
                $table->string('nombres');
                $table->string("correo")->nullable();
                $table->string('correo_uta');
                $table->string('celular');
                $table->string("telefono")->nullable();

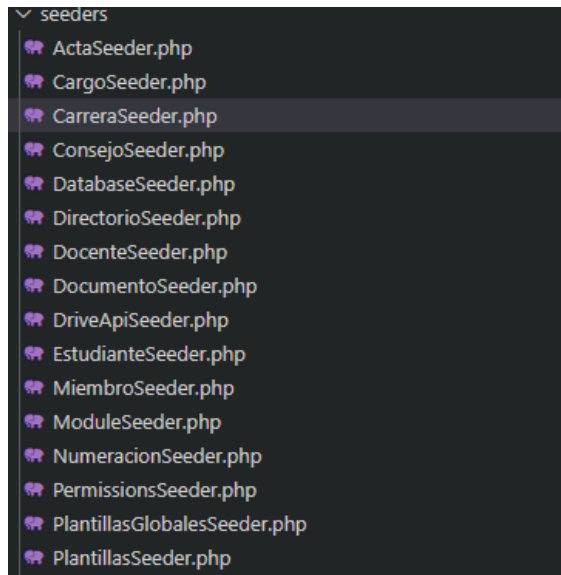
                $table->timestamps();
            });
    }

    /** Reverse the migrations.*/
    public function down()
    {
        Schema::dropIfExists('docentes');
    }
}

```

Carpeta seeders

La carpeta seeders, permite insertar datos al momento de crear las tablas ejecutando las migraciones, depende de la lógica de negocio si se utiliza los seeders al momento de la implementación del sistema.



Ejemplo en el caso de carreras

Se muestra un ejemplo de las carreras que ya existen las cuales se crean por defecto con los Seeder.

```
class CarreraSeeder extends Seeder
{
    /** Run the database seeds. */
    public function run()
    {
        Carrera::create([
            'nombre' => 'Ingeniería en Sistemas Computacionales e Informaticos'
        ]);

        Carrera::create([
            'nombre' => 'Ingeniería en Electrónica y Comunicaciones'
        ]);

        Carrera::create([
            'nombre' => 'Ingeniería Industrial en Procesos de Automatización'
        ]);

        Carrera::create([
            'nombre' => 'Tecnologías de la Información'
        ]);

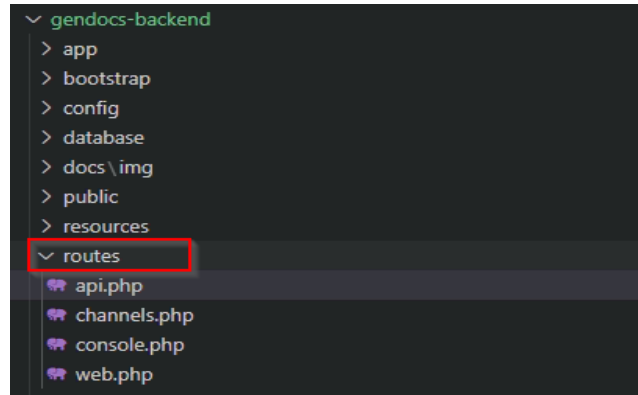
        Carrera::create([
            'nombre' => 'Ingeniería en Telecomunicaciones'
        ]);

        Carrera::create([
            'nombre' => 'Ingeniería Industrial'
        ]);

        Carrera::create([
            'nombre' => 'Software'
        ]);
    }
}
```

Ejemplo de Seeder de carreras

En la carpeta routes (rutas) es el componente más importante del framework de Laravel ya que se encarga de manejar el flujo de las solicitudes HTTP, desde y hacia el cliente.



Rutas API REST

Se verifican todas las API que se encuentran creadas de cada uno de los modelos del sistema.

```

Route::middleware(["auth:sanctum", "isUserActive"])->group(function () {

    Route::apiResource('docentes', DocenteController::class)->except(['destroy']);
    Route::apiResource('carreras', CarreraController::class)->except(['destroy', 'store']);
    Route::apiResource('procesos', ProcesoController::class)->except(['destroy']);
    Route::apiResource('plantillas', PlantillasController::class)->except(['destroy']);
    Route::apiResource('tipo-consejos', TipoConsejoController::class)->only(['index']);
    Route::apiResource('documentos', DocumentoController::class)->except(['show', 'update']);
    Route::apiResource('numeracion', NumeracionController::class)->only(['index', 'store']);
    Route::apiResource('batch', BatchController::class)->only(['show']);
    Route::apiResource("plantillas-globales", PlantillasGlobalesController::class)->only(['index']);

    Route::apiResource('miembros', MiembroController::class)->except(['show']);

    Route::apiResource('notificaciones', NotificacionController::class)->only('store');

    Route::apiResource('estudiantes', EstudianteController::class)->except(['destroy']);
    Route::post('estudiantes/email-notificacion', [EstudianteController::class, 'sendNotificationEmail']);

    Route::apiResource('consejos', ConsejoController::class);
    Route::patch('consejos/{consejo}/cerrar', [ConsejoController::class, 'cerrar']);

    Route::apiResource('actas', ActaController::class)->only(['index', 'store', 'show']);
    Route::put('actas/{acta}/plantilla', [ActaController::class, 'crearPlantilla']);
    Route::get('actas/{acta}/descargar', [ActaController::class, 'descargar']);

    Route::apiResource('cargos', CargoController::class);

    // AUTH
    Route::apiResource('modulos', ModuleController::class)->only(['index']);
    Route::get('me', [UserController::class, 'me']);
    Route::put('user/profile', [UserController::class, 'updateProfile']);
    Route::put('user/password', [UserController::class, 'updatePassword']);
    Route::post('user/reset-password', [UserController::class, 'resetPassword']);
    Route::apiResource('user', UserController::class)->except(['destroy']);

    Route::get('roles', [RoleController::class, 'index']);
});

```

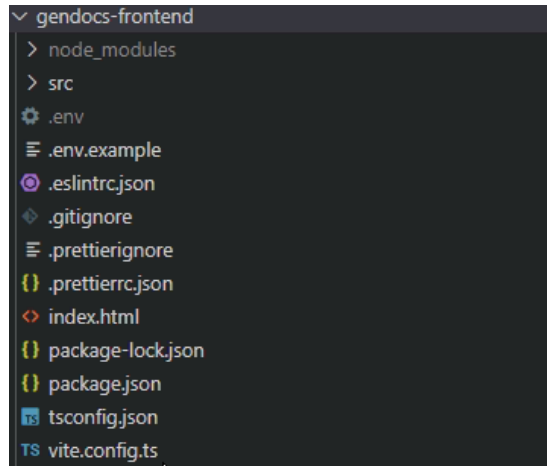
Métodos API REST

En la siguiente figura se muestra de forma más detallada las API con sus métodos API REST que son GET, POST, PUT, DELETE.

Method	URI	Action
GET HEAD	/	Closure
GET HEAD	api/actas	App\Http\Controllers\ActaController@index
POST	api/actas	App\Http\Controllers\ActaController@store
GET HEAD	api/actas/{acta}	App\Http\Controllers\ActaController@show
GET HEAD	api/actas/{acta}/descargar	App\Http\Controllers\ActaController@descargar
PUT	api/actas/{acta}/plantilla	App\Http\Controllers\ActaController@crearPlantilla
GET HEAD	api/batch/{batch}	App\Http\Controllers\BatchController@show
GET HEAD	api/cargos	App\Http\Controllers\CargoController@index
POST	api/cargos	App\Http\Controllers\CargoController@store
GET HEAD	api/cargos/{cargo}	App\Http\Controllers\CargoController@show
PUT PATCH	api/cargos/{cargo}	App\Http\Controllers\CargoController@update
DELETE	api/cargos/{cargo}	App\Http\Controllers\CargoController@destroy
GET HEAD	api/carreras	App\Http\Controllers\CarreraController@index
GET HEAD	api/carreras/{carrera}	App\Http\Controllers\CarreraController@show
PUT PATCH	api/carreras/{carrera}	App\Http\Controllers\CarreraController@update
GET HEAD	api/consejos	App\Http\Controllers\ConsejoController@index
POST	api/consejos	App\Http\Controllers\ConsejoController@store
GET HEAD	api/consejos/{consejo}	App\Http\Controllers\ConsejoController@show
PUT PATCH	api/consejos/{consejo}	App\Http\Controllers\ConsejoController@update
DELETE	api/consejos/{consejo}	App\Http\Controllers\ConsejoController@destroy
PATCH	api/consejos/{consejo}/cerrar	App\Http\Controllers\ConsejoController@cerrar
GET HEAD	api/csrf-cookie	Laravel\Sanctum\Http\Controllers\CsrfCookieController@show
GET HEAD	api/docentes	App\Http\Controllers\DocenteController@index
POST	api/docentes	App\Http\Controllers\DocenteController@store
GET HEAD	api/docentes/{docente}	App\Http\Controllers\DocenteController@show
PUT PATCH	api/docentes/{docente}	App\Http\Controllers\DocenteController@update
GET HEAD	api/documentos	App\Http\Controllers\DocumentoController@index
POST	api/documentos	App\Http\Controllers\DocumentoController@store
DELETE	api/documentos/{documento}	App\Http\Controllers\DocumentoController@destroy
GET HEAD	api/estudiantes	App\Http\Controllers\EstudianteController@index
POST	api/estudiantes	App\Http\Controllers\EstudianteController@store
POST	api/estudiantes/email-notificacion	App\Http\Controllers\EstudianteController@sendNotificationEmail
GET HEAD	api/estudiantes/{estudiante}	App\Http\Controllers\EstudianteController@show
PUT PATCH	api/estudiantes/{estudiante}	App\Http\Controllers\EstudianteController@update
POST	api/login	Laravel\Fortify\Http\Controllers\AuthenticatedSessionController@store
POST	api/logout	Laravel\Fortify\Http\Controllers\AuthenticatedSessionController@destroy
GET HEAD	api/me	App\Http\Controllers\UserController@me
GET HEAD	api/miembros	App\Http\Controllers\MiembroController@index
POST	api/miembros	App\Http\Controllers\MiembroController@store
PUT PATCH	api/miembros/{miembro}	App\Http\Controllers\MiembroController@update
DELETE	api/miembros/{miembro}	App\Http\Controllers\MiembroController@destroy
GET HEAD	api/modulos	App\Http\Controllers\ModuleController@index
POST	api/notificaciones	App\Http\Controllers\NotificacionController@store
GET HEAD	api/numeracion	App\Http\Controllers\NumeracionController@index
POST	api/numeracion	App\Http\Controllers\NumeracionController@store
GET HEAD	api/plantillas	App\Http\Controllers\PlantillasController@index
POST	api/plantillas	App\Http\Controllers\PlantillasController@store
GET HEAD	api/plantillas-globales	App\Http\Controllers\PlantillasGlobalesController@index
GET HEAD	api/plantillas/{plantilla}	App\Http\Controllers\PlantillasController@show
PUT PATCH	api/plantillas/{plantilla}	App\Http\Controllers\PlantillasController@update
GET HEAD	api/procesos	App\Http\Controllers\ProcesoController@index
POST	api/procesos	App\Http\Controllers\ProcesoController@store
GET HEAD	api/procesos/{proceso}	App\Http\Controllers\ProcesoController@show
PUT PATCH	api/procesos/{proceso}	App\Http\Controllers\ProcesoController@update
GET HEAD	api/roles	App\Http\Controllers\RoleController@index
GET HEAD	api/tipo-consejos	App\Http\Controllers\TipoConsejoController@index
GET HEAD	api/user	App\Http\Controllers\UserController@index
POST	api/user	App\Http\Controllers\UserController@store
POST	api/user/confirm-password	Laravel\Fortify\Http\Controllers\ConfirmablePasswordController@store
GET HEAD	api/user/confirmed-password-status	Laravel\Fortify\Http\Controllers\ConfirmedPasswordStatusController@show
PUT	api/user/password	App\Http\Controllers\UserController@updatePassword
PUT	api/user/profile	App\Http\Controllers\UserController@updateProfile
POST	api/user/reset-password	App\Http\Controllers\UserController@resetPassword
GET HEAD	api/user/{user}	App\Http\Controllers\UserController@show
PUT PATCH	api/user/{user}	App\Http\Controllers\UserController@update

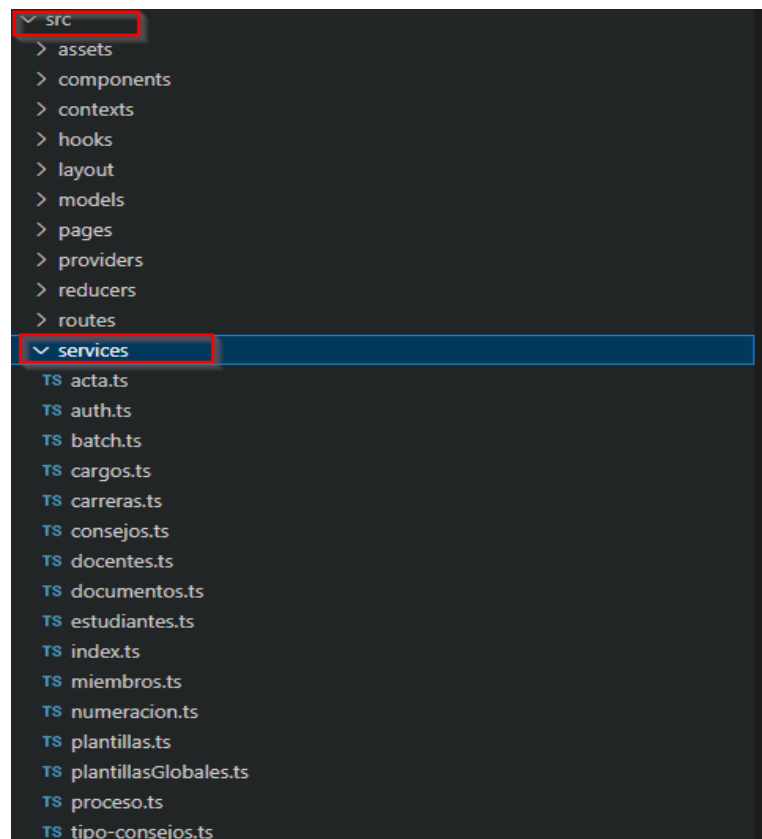
Proyecto frontend

Se evidencia en la figura siguiente la segunda sección que es el proyecto frontend el cual tiene la siguiente estructura.



Estructura de la carpeta src

Se observa la carpeta src, en el cual hay una subcarpeta denominada services (Servicios) donde se encuentra el consumo de todos los servicios expuestos en nuestro proyecto de Laravel (BackEnd) de cada una de las API REST creadas.



Carpeta Axios

Para el consumo de estos servicios se utiliza Axios que es una librería de JavaScript, en el cual permite realizar llamadas de API REST con retorno JSON (JavaScript Object Notation).

```
TS docentes.ts X
gendocs-frontend > src > services > TS docentes.ts > ...
1 import axios from "axios";
2 import { HTTP_STATUS } from "models/enums";
3 import {
4   DocenteForm,
5   IDocente,
6   IFilterPagingProps,
7   IResponse,
8 } from "models/interfaces";
```

Método GET para obtener todos los docentes

Como ejemplo de la librería Axios se evidencia el consumo de cada uno de los servicios de las API REST, tanto para GET, POST y PUT.

```
export async function getDocentes(
  props?: IFilterPagingProps
): Promise<IDocente[]> {
  try {
    const params = parseFilterPagingProps(props);

    const {
      data: { data },
    } = await axios.get(`docentes?${params}`);

    return data;
  } catch (error) {
    return [];
  }
}
```

Método GET por ID

El método getDocente me permite obtener los profesores buscando por un parámetro de entrada como es el ID del docente.

```
export async function getDocente(
  docenteId: string
): Promise<IResponse<IDocente>> {
  try {
    const {
      data: { data },
    } = await axios.get("docentes/" + docenteId);

    return {
      data,
      message: "",
      status: HTTP_STATUS.ok,
    };
  } catch (error) {
    return handleErrors(error, null);
  }
}
```

Método POST o guardar docentes

El método saveDocente me permite guardar un docente con sus respectivos datos.

```
export async function saveDocente(
  form: DocenteForm
): Promise<IResponse<IDocente>> {
  try {
    const payload = {
      type: "simple",
      ...form,
    };

    const {
      data: { data },
    } = await axios.post("docentes", payload);

    return {
      status: HTTP_STATUS.created,
      data: data,
      message: HTTP_MESSAGES[201],
    };
  } catch (error) {
    return handleErrors(error);
  }
}
```

Método PUT, para actualizar docentes

El método updateDocente me permite actualizar un docente siempre y cuando se mande como parámetro de entrada un ID.

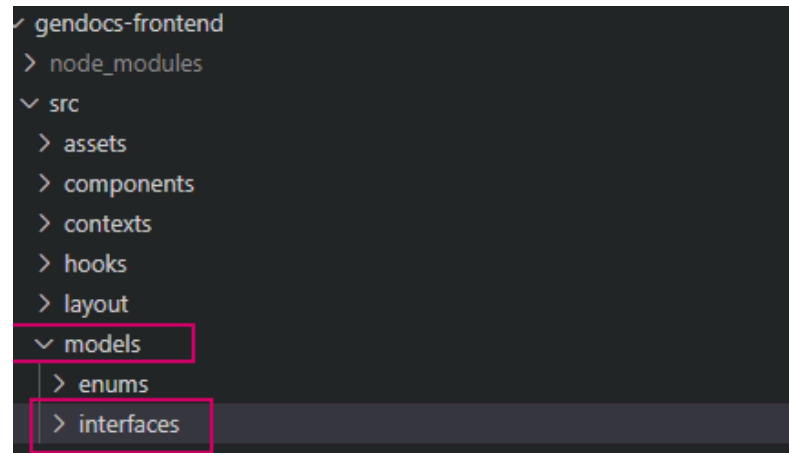
```
export async function updateDocente(
  form: IDocente
): Promise<IResponse<IDocente>> {
  try {
    const {
      data: { data },
    } = await axios.put("docentes/" + form.id, form);

    return {
      status: HTTP_STATUS.ok,
      data: data,
      message: HTTP_MESSAGES[200],
    };
  } catch (error) {
    return handleErrors(error);
  }
}
```

Por otra parte, dentro de la carpeta SRC se encuentra los modelos en donde se tiene las interfaces.

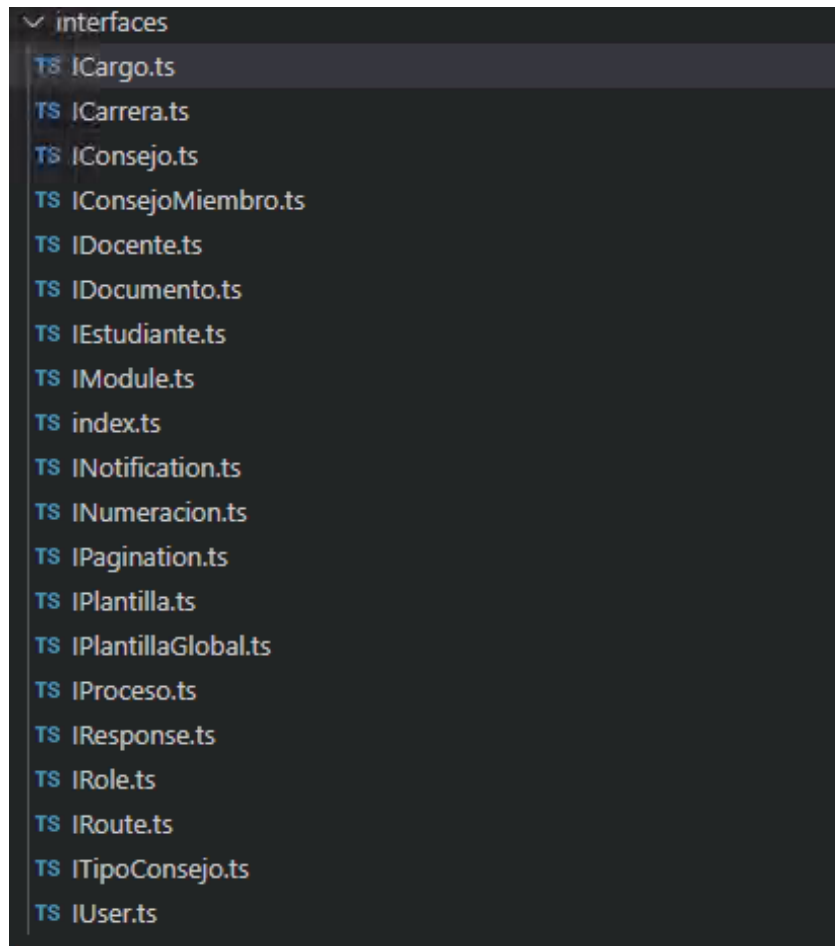
Interfaces

Las interfaces están compuestas por los tipos de datos que tienen cada clase.



Clases de las Interfaces

En las interfaces se tiene las clases que terminan en una extensión ts y su nomenclatura empieza con INombreDeLaClase.ts para diferenciar que se trata de una interfaz.



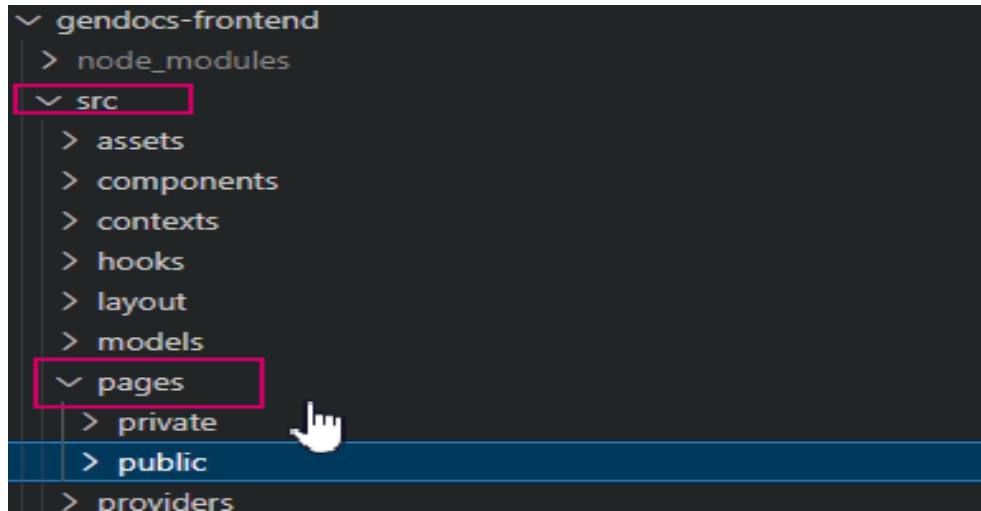
Ejemplo de interfaces

Ejemplo de interfaz de la clase IDocentes, se definen cada uno de los campos con su respectivo tipo de dato.

```
export interface IDocente {  
  id: number;  
  cedula: string;  
  nombres: string;  
  celular: string;  
  correo_uta: string;  
  correo: string;  
  telefono: string;  
}  
  
export type DocenteForm = Omit<IDocente, "id">;
```

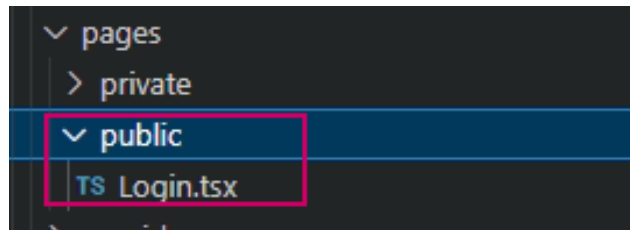
Private-public

Por otra parte, se tiene los pages(páginas) en donde se divide en dos private y public (privado y público).



Clase Login.tsx de la carpeta public

La carpeta public se tiene una clase denominada Login.tsx como se muestra en la siguiente figura, en donde se encuentra desarrollada la interfaz de usuario, para ingresar al sistema.



Ejemplo de Login.tsx

Ejemplo de Login.tsx, en este caso se diseña los textBox en donde se va a escribir el correo y la contraseña y un botón para ingresar.

```

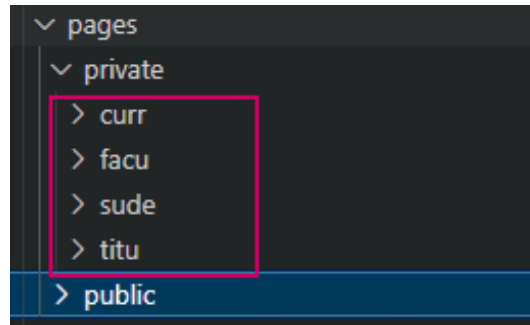
<TextField
  required
  autoFocus
  fullWidth
  margin="normal"
  id="email"
  name="email"
  label="Email"
  autoComplete="email"
  value={formik.values.email}
  onChange={formik.handleChange}
  error={formik.touched.email && Boolean(formik.errors.email)}
  helperText={formik.touched.email && formik.errors.email}
/>
<TextField
  margin="normal"
  required
  fullWidth
  id="password"
  name="password"
  type="password"
  label="Contraseña"
  autoComplete="current-password"
  value={formik.values.password}
  onChange={formik.handleChange}
  error={formik.touched.password && Boolean(formik.errors.password)}
  helperText={formik.touched.password && formik.errors.password}
/>

<Button
  fullWidth
  type="submit"
  variant="contained"
  sx={{ mt: 3, mb: 2 }}
  disabled={submitting}
  loading={submitting}
>
  Ingresar
</Button>
</Box>

```

División de cada área

Por otra parte, la subcarpeta private se encuentra dividido para módulos del sistema como son: Curr (Unidad Curricular), Facu (Secretaría de Facultad), Sude (Secretaría de Consejo Académico de Facultad) y Titu (Secretaría de Titulación), tal y como se muestra en la figura siguiente.



Funcionalidades e interfaces de usuario

Dentro de cada uno de estos módulos se tienen las funcionalidades e interfaces de usuario, como lo indica en la siguiente figura, donde se tiene el módulo de facu y se puede observar que dentro de ello se tiene la carpeta docente que está compuesto por la carpeta componentes, hooks y su archivo base donde se encuentra la interfaz de usuario maestro Docente.tsx.

