



Universidad de San Carlos de Guatemala
Facultad de Ingeniería
Escuela de Ciencias y Sistemas

**DISEÑO E IMPLEMENTACIÓN DE PORTAL ADMINISTRATIVO Y DE CONTROL DE ACTIVIDADES DE
CÁTEDRA (TEO), DEL DEPARTAMENTO DE MATEMÁTICA DE LA FACULTAD DE INGENIERÍA DE
LA UNIVERSIDAD DE SAN CARLOS DE GUATEMALA**

Brayan Alexander Flores

César Estuardo Morales Toledo

Asesorado por el Ing. Herman Igor Véliz Linares

Guatemala, febrero de 2021

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA



FACULTAD DE INGENIERÍA

**DISEÑO E IMPLEMENTACIÓN DE PORTAL ADMINISTRATIVO Y DE CONTROL DE ACTIVIDADES DE
CÁTEDRA (TEO), DEL DEPARTAMENTO DE MATEMÁTICA DE LA FACULTAD DE INGENIERÍA DE
LA UNIVERSIDAD DE SAN CARLOS DE GUATEMALA**

TRABAJO DE GRADUACIÓN

PRESENTADO A LA JUNTA DIRECTIVA DE LA
FACULTAD DE INGENIERÍA
POR

Brayan Alexander Flores

César Estuardo Morales Toledo

ASESORADO POR EL ING. HERMAN IGOR VÉLIZ LINARES

AL CONFERÍRSELES EL TÍTULO DE

INGENIERO EN CIENCIAS Y SISTEMAS

GUATEMALA, FEBRERO DE 2021

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA
FACULTAD DE INGENIERÍA



NÓMINA DE JUNTA DIRECTIVA

DECANA	Inga. Aurelia Anabela Cordova Estrada
VOCAL I	Ing. José Francisco Gómez Rivera
VOCAL II	Ing. Mario Renato Escobedo Martínez
VOCAL III	Ing. José Milton de León Bran
VOCAL IV	Br. Christian Moisés de la Cruz Lea
VOCAL V	Br. Kevin Vladimir Armando Cruz
SECRETARIO	Ing. Hugo Humberto Rivera Pérez

TRIBUNAL QUE PRACTICÓ EL EXAMEN GENERAL PRIVADO

DECANA	Inga. Aurelia Anabela Cordova Estrada
EXAMINADOR	Ing. Pedro Pablo Hernández Ramírez
EXAMINADOR	Ing. Luis Fernando Espino Barrios
EXAMINADOR	Ing. César Augusto Fernández Cáceres
SECRETARIO	Ing. Hugo Humberto Rivera Pérez

HONORABLE TRIBUNAL EXAMINADOR

En cumplimiento con los preceptos que establece la ley de la Universidad de San Carlos de Guatemala, presentamos a su consideración nuestro trabajo de graduación titulado:

DISEÑO E IMPLEMENTACIÓN DE PORTAL ADMINISTRATIVO Y DE CONTROL DE ACTIVIDADES DE CÁTEDRA (TEO), DEL DEPARTAMENTO DE MATEMÁTICA DE LA FACULTAD DE INGENIERÍA DE LA UNIVERSIDAD DE SAN CARLOS DE GUATEMALA

Tema que nos fuera asignado por la Dirección de la Escuela de Ingeniería en Ciencias y Sistemas, con fecha febrero de 2019.



Brayan Alexander Flores



César Estuardo Morales Toledo



Guatemala, 13 de noviembre de 2019.

Escuela de Ciencias y Sistemas,
Facultad de Ingeniería USAC,

Respetables Ingenieros:

Por medio de la presenta hago de su conocimiento que los estudiantes César Estuardo Morales Toledo que se identifica con CUI 2950904130101 y Brayan Alexander Flores que se identifica con CUI 2686739420101, a quienes asesoré en el proceso de la realización de su trabajo de graduación, Diseño e Implementación del Portal Administrativo y de Control de Actividades de Cátedra (Teo), del Departamento de Matemática de la Facultad De Ingeniería de la Universidad de San Carlos de Guatemala, tienen mi aprobación para continuar en el proceso de revisión por la Escuela de Ciencias y Sistemas, tanto en la aplicación realizada como en el trabajo escrito.

Sin otro particular, me es grato suscribirme

F: 

Ing. Herman Igor Veliz Linares
Coordinador de SAE/SAP
herman.veliz@gmail.com
54022579



Universidad San Carlos de Guatemala
Facultad de Ingeniería
Escuela de Ingeniería en Ciencias y Sistemas

Guatemala, 19 de noviembre del 2019

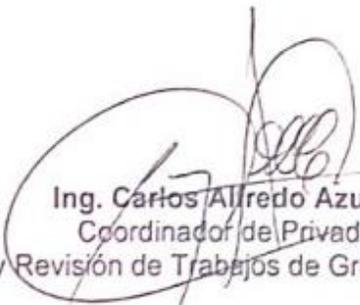
Ingeniero
Carlos Gustavo Alonzo
Director de la Escuela de Ingeniería
En Ciencias y Sistemas

Respetable Ingeniero Alonzo:

Por este medio hago de su conocimiento que he revisado el trabajo de graduación de los estudiantes **BRAYAN ALEXANDER FLORES** con carné 201403564 y CUI 2686 73942 0101, y **CÉSAR ESTUARDO MORALES TOLEDO** con carné 201403573 y CUI 2950 90413 0101, titulado: **DISEÑO E IMPLEMENTACIÓN DE PORTAL ADMINISTRATIVO Y DE CONTROL DE ACTIVIDADES DE CÁTEDRA (TEO), DEL DEPARTAMENTO DE MATEMÁTICA DE LA FACULTAD DE INGENIERÍA DE LA UNIVERSIDAD DE SAN CARLOS DE GUATEMALA**, y a mi criterio el mismo cumple con los objetivos propuestos para su desarrollo, según el protocolo.

Al agradecer su atención a la presente, aprovecho la oportunidad para suscribirme,

Atentamente,


Ing. Carlos Alfredo Azurdia
Coordinador de Privados
y Revisión de Trabajos de Graduación



UNIVERSIDAD DE SAN CARLOS
DE GUATEMALA



FACULTAD DE INGENIERÍA
ESCUELA DE INGENIERÍA EN
CIENCIAS Y SISTEMAS

*El Director de la Escuela de Ingeniería en Ciencias y Sistemas de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala, luego de conocer el dictamen del asesor con el visto bueno del revisor y del Licenciado en Letras, del trabajo de graduación **“DISEÑO E IMPLEMENTACIÓN DE PORTAL ADMINISTRATIVO Y DE CONTROL DE ACTIVIDADES DE CÁTEDRA (TEO), DEL DEPARTAMENTO DE MATEMÁTICA DE LA FACULTAD DE INGENIERÍA DE LA UNIVERSIDAD DE SAN CARLOS DE GUATEMALA”**, realizado por los estudiantes, BRAYAN ALEXANDER FLORES y CÉSAR ESTUARDO MORALES TOLEDO aprueba el presente trabajo y solicita la autorización del mismo.*

“ID Y ENSEÑAD A TODOS”

The block contains a handwritten signature in black ink over a circular official stamp. The stamp text includes 'UNIVERSIDAD DE SAN CARLOS DE GUATEMALA' and 'DIRECCIÓN DE INGENIERÍA EN CIENCIAS Y SISTEMAS'.

*Msc. Carlos Gustavo Atonzo
Director*

Escuela de Ingeniería en Ciencias y Sistemas

Guatemala, 08 de febrero de 2021



USAC
TRICENTENARIA
Universidad de San Carlos de Guatemala

Decanato
Facultad de Ingeniería
24189101- 24189102

DTG. 053.2021.

La Decana de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala, luego de conocer la aprobación por parte del Director de la Escuela de Ingeniería en Ciencias y Sistemas, al Trabajo de Graduación titulado: **DISEÑO E IMPLEMENTACIÓN DE PORTAL ADMINISTRATIVO Y DE CONTROL DE ACTIVIDADES DE CÁTEDRA (TEO), DEL DEPARTAMENTO DE MATEMÁTICA DE LA FACULTAD DE INGENIERÍA DE LA UNIVERSIDAD DE SAN CARLOS DE GUATEMALA**, presentado por los estudiantes universitarios: **Brayan Alexander Flores y César Estuardo Morales Toledo**, y después de haber culminado las revisiones previas bajo la responsabilidad de las instancias correspondientes, autoriza la impresión del mismo.

IMPRÍMASE:

Inga. Anabeía Cordova Estrada
Decana



Guatemala, febrero de 2021.

AACE/asga

ACTO QUE DEDICO A:

- Mi padre** Eliseo Subuyuj Molina, por formarme como un hombre con su cariño, consejo y correcciones.
- Mi madre** Madelainer Flores Sosa, por todo su amor y esfuerzo, al nunca darse por vencida y sacar adelante a nuestra familia y por su apoyo durante mi etapa universitaria.
- Mis hermanas** Maily Celene y Gerlyn Margarita Sánchez Flores, y Maria Elisa Subuyuj Flores, por ser un ejemplo en mi vida, por sus consejos y el cariño que siempre me han dado.
- Mis primos** Por estar en cualquier momento para mí y por todos los momentos alegres que hemos compartido.
- Mis tíos** Yesenia Siomara, Brayner Moisés, Ariel Enrique, Zulma y Adila Flores Sosa, por todos sus consejos y ser una parte muy importante en mi vida.

Demás familia

Víctor Daniel González Sánchez, Walter Augusto Ferguso Cabrera, Víctor Hugo González Corado, Blanca Leticia Sosa Mijangos, Reyes, Moisés Enrique Flores Guerra, por estar para mí en cualquier momento.

Mis amigos

Diana Melissa Tablas Padilla, Adolfo Rolando Salazar Urbina, César Estuardo Morales Toledo, Mario Alberto Alarcón, Marlis Karina Garrido Reyes, Cindi Viviana Lucero, Óscar Alejandro Luna, Diego Fernando Cruz, Leonel Eduardo Mansilla, Albin Avila, Javier Cabrera, Luis Gilberto Barrios, Javier Eduardo Cifuentes, Kimberly Paola de la Cruz, Brenda Alejandra Hernández, mis amigos del departamento de matemáticas y de la colonia Jardines de San Juan II y demás personas conocidas durante esta etapa de mi vida.

Brayan Alexander Flores

ACTO QUE DEDICO A:

- Dios** Por darme la bendición de la vida y la oportunidad de alcanzar una meta más.
- Mis padres** Ana Maritza Toledo Villagrán y Julio César Morales Sandoval, por su gran amor, sus cuidados, su incondicional apoyo y su gran entrega a su labor de padres.
- Mis hermanos** Julio Eduardo, Vivian Denisse y Luis Roberto, por todo el cariño, por sus enseñanzas, consejos, incondicional apoyo y por todos esos momentos que me hicieron crecer como persona.
- Mi novia** Fernanda Raquel Rosales Mazariegos, por su gran amor y por su compañía en cada una de las batallas para alcanzar esta meta.
- Mi cuñada y sobrinos** Ivina Acuña, Fabiola y Ángel Morales, por todo el cariño que me han dado.

Mis amigos

Brayan Alexander Flores, Cindi Viviana Lucero, Óscar Alejandro Luna, Adolfo Salazar Urizar, Mario Alberto Alarcón, Leonel Eduardo Mancilla, Albin Ávila, Diego Cruz, Javier Cabrera, Herber Fernando Sandoval, Juan Luis Baldelomar y mis amigos del Departamento de Matemática, por todo el apoyo y momentos compartidos.

Familia Rosales

Por todo el cariño y por siempre recibirme con los brazos abiertos.

César Estuardo Morales Toledo

AGRADECIMIENTOS A:

Universidad de San Carlos de Guatemala	Por permitirme pertenecer a esta casa de estudios y formarme como un profesional
Facultad de Ingeniería	Por darme la oportunidad y las herramientas necesarias para estudiar la carrera de Ingeniería en Ciencias y Sistemas
Asesor	Ingeniero Herman Igor Véliz Linares, por su apoyo durante la elaboración de este trabajo.
Pueblo de Guatemala	Por el apoyo económico que brindan a la Universidad de San Carlos.
Departamento de Matemáticas Facultad de Ingeniería	Por darme mi primera oportunidad laboral y por permitir realizar este trabajo.
Arquitectos e Ingenieros de Tecnología S.A.	Por abrirme las puertas y darme la oportunidad de desempeñarme como un profesional y el apoyo al final de mi carrera.

Brayan Alexander Flores

AGRADECIMIENTOS A:

Universidad de San Carlos de Guatemala	Por ser mi <i>alma máter</i> y por la oportunidad de formarme como un profesional.
Facultad de Ingeniería	Por ser guía para adquirir los conocimientos necesarios y por formarme en la rama de Ciencias y Sistemas.
Asesor	Ingeniero Herman Igor Véliz Linares, por su apoyo durante la elaboración de este trabajo.
Pueblo de Guatemala	Por el apoyo económico que brindan a la Universidad de San Carlos.
Departamento de Matemáticas de Facultad de Ingeniería	Por darme la confianza para realizar este trabajo de graduación y por la oportunidad laboral.

César Estuardo Morales Toledo

ÍNDICE GENERAL

ÍNDICE DE ILUSTRACIONES.....	V
LISTA DE SÍMBOLOS	VII
GLOSARIO	IX
RESUMEN.....	XI
OBJETIVOS.....	XIII
INTRODUCCIÓN	XV
1. MARCO TEÓRICO.....	1
1.1. Antecedentes.....	1
1.1.1. Trabajo de graduación de Pablo Alejandro Reyes Orozco	1
1.1.2. Matemática en línea	3
1.1.3. Página del Departamento de Matemática de la Facultad de Ingeniería	4
1.2. Diseño y desarrollo web	5
1.2.1. Desarrollo web.....	6
1.2.1.1. Clientes web	6
1.2.1.2. Aplicaciones web	6
1.2.1.3. Web service	7
1.2.2. Diseño web	7
1.2.2.1. Diseño adaptativo	8
1.3. Análisis y diseño de software	8
1.3.1. Metodologías de desarrollo.....	8

2.	DISEÑO DEL PORTAL TEO.....	15
2.1.	Definición del problema.....	15
2.2.	Análisis de requerimientos	16
2.2.1.	Requerimientos funcionales	16
2.2.1.1.	Usuarios	16
2.2.1.2.	Secciones de clase.....	17
2.2.1.3.	Matriculación por curso	17
2.2.1.4.	Notas y actividades de clase	18
2.2.1.5.	Estadísticas	19
2.2.1.6.	Calendario de actividades	19
2.2.1.7.	Reportes.....	20
2.2.1.8.	Comunicación auxiliar-estudiante.....	20
2.2.1.9.	Recursos didácticos	20
2.2.2.	Requerimientos no funcionales	21
2.2.2.1.	Usabilidad.....	21
2.2.2.2.	Multiplataforma	21
2.2.2.3.	Seguridad	21
2.2.2.4.	Disponibilidad	22
2.3.	Arquitectura utilizada.....	22
2.3.1.	Tecnologías utilizadas	23
2.3.2.	Diagrama general de la arquitectura	23
2.3.3.	Implementación del patrón de arquitectura	24
2.3.4.	Modelo de datos utilizado.....	25
3.	DOCUMENTACIÓN	27
3.1.	Manual del usuario	27
3.1.1.	Tipos de usuarios	28
3.1.1.1.	Usuario administrador	28
3.1.1.2.	Usuario auxiliar.....	28

	3.1.1.3.	Usuario alumno.....	29
3.1.2.		Inicio de sesión	29
	3.1.2.1.	Contraseña perdida	30
3.1.3.		Registro de usuarios.....	31
3.1.4.		Perfil	33
3.1.5.		Cerrar sesión	34
3.1.6.		Horario de cursos	35
3.1.7.		Olimpiada Interuniversitaria de Ciencia y Tecnología.....	35
3.1.8.		Pantalla de inicio.....	36
3.1.9.		Noticias	37
3.1.10.		Calendario	38
3.1.11.		Cursos	38
	3.1.11.1.	Listado de alumnos.....	39
	3.1.11.2.	Gestión de notas.....	40
	3.1.11.3.	Recursos.....	41
	3.1.11.4.	Estadísticas	42
3.1.12.		Administrador.....	42
	3.1.12.1.	Clases.....	42
	3.1.12.2.	Auxiliares	44
	3.1.12.3.	Catedráticos.....	45
3.2.		Manual técnico	46
3.2.1.		Características generales	47
3.2.2.		Definición de las reglas de negocio implementadas	47
3.2.3.		Fundamentación de la tecnología utilizada.....	49
3.2.4.		Vista funcional	50
	3.2.4.1.	Sobre la gestión de usuarios	50
	3.2.4.2.	Sobre las secciones de clase	51

3.2.4.3.	Sobre la gestión de notas y actividades de clase	51
3.2.4.4.	Sobre la gestión de recursos didácticos	52
3.2.4.5.	Sobre la comunicación entre tutor académico y estudiante	53
3.2.5.	Vista lógica	53
3.2.6.	Estructura de la codificación del proyecto	54
3.2.6.1.	Controladores	54
3.2.6.2.	Rutas	55
3.2.6.3.	Vistas.....	56
3.2.6.4.	Conexión con la base de datos	58
3.2.6.5.	Conexión con el servidor de correos	59
3.2.6.6.	Calendarización de tareas y envío de correos electrónicos	59
3.2.6.7.	Generación de estadísticas y descarga de archivos PDF.	63
4.	EVALUACIÓN DEL SISTEMA	67
4.1.	Análisis de resultados obtenidos	67
	CONCLUSIONES.....	75
	RECOMENDACIONES.....	77
	BIBLIOGRAFÍA.....	79

ÍNDICE DE ILUSTRACIONES

FIGURAS

1.	Estructura funcional del proyecto de Pablo Reyes	2
2.	Arquitectura general	24
3.	Modelo de tres capas	25
4.	Diagrama entidad-relación	26
5.	Pantalla de inicio de sesión	27
6.	Formulario de credenciales	30
7.	Formulario de recuperación de acceso	31
8.	Ventana de registro de usuario	32
9.	Mensaje de error durante el registro	33
10.	Datos personales	33
11.	Cerrar sesión.....	34
12.	Horario de cursos	35
13.	Pantalla de inicio	36
14.	Sección de noticias	37
15.	Calendario de actividades	38
16.	Listado de cursos asignados	39
17.	Pantalla para gestión de notas	40
18.	Pantalla de recursos.....	41
19.	Pantalla de administración de clases	43
20.	Pantalla de edición de clases	43
21.	Pantalla de administración de clases	44
22.	Pantalla de edición de auxiliares	45
23.	Pantalla de administración de clases	46

24.	Caso de uso de gestión de usuarios	50
25.	Caso de uso de secciones de clase.....	51
26.	Caso de uso de gestión de notas y actividades	52
27.	Caso de uso de gestión de recursos.....	52
28.	Caso de uso de comunicación tutor y estudiante.....	53
29.	Estructura de una ruta.....	56
30.	Solicitud post por medio de un formulario	57
31.	<i>Token</i> de verificación	58
32.	Uso de templete en una vista.....	58
33.	Configuración Cron Job	59
34.	Tarea Cron Job	60
35.	Código de tarea programada	61
36.	Importación de librería	62
37.	Schedule Cron Job.....	62
38.	Código de estadísticas.....	64
39.	Navegación dentro del portal	68
40.	Calificación de la forma de ingresar notas	69
41.	Utilidad en los recursos didácticos	70
42.	Forma de generar estadísticas de exámenes	71
43.	Comunicación entre tutor académico y estudiante.....	72
44.	Envío de notificaciones	73

LISTA DE SÍMBOLOS

Símbolo	Significado
@	Arroba

GLOSARIO

Apache	Software de servicio web, de código abierto, desarrollado y mantenido por Apache Software Foundation.
<i>Back-end</i>	Es la capa de acceso a los datos de un software, la cual no es accesible a los usuarios, esta capa contiene la lógica del sistema.
Base de datos	Conjunto de información persistente en un contexto, estructurada para su posterior recuperación, análisis o transmisión.
Cliente servidor	Arquitectura que organiza el software para que se pueda distribuir y ejecutar en dos partes. El cliente es el programa que utiliza los servicios proporcionados por el servidor.
CMS	Conjunto de programas (<i>Content Management System</i>) que son utilizados para administrar contenido digital.
<i>Code and fix</i>	Proceso para desarrollo de software muy simple, presente en muchos casos. Es un proceso cíclico que cuenta con dos pasos, codificar y solucionar los errores.

CSV	Tipo de documento que se encuentra estructurado en forma de tabla, en el que las columnas se encuentran separadas por comas.
<i>E-learning</i>	Aprendizaje que utiliza tecnología electrónica para permitir el acceso de manera remota al currículo educativo.
<i>Framework</i>	Área de trabajo conceptualmente estructurada, con asistencia de módulos estandarizados para el desarrollo de software.
<i>Graceful Degradation</i>	Capacidad que tiene un sistema para seguir siendo útil cuando una o gran parte de este ha dejado de funcionar o sido destruida.
Laravel	Framework de código abierto, para el desarrollo de sistemas en PHP.
MVC	Patrón de arquitectura de software (Modelo Vista Controlador.), que separa la lógica del negocio, los datos y las vistas del usuario final, para el desarrollo de un sistema.
Procesador	Componente electrónico donde se llevan a cabo los procesos lógicos.

RESUMEN

Con el fin de facilitar las labores que llevan a cabo los tutores académicos del Departamento de Matemática de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala, y mejorar el nivel de conocimiento que poseen los estudiantes sobre sus notas en los diversos cursos de dicho departamento, nace la necesidad de elaborar un portal que sea de fácil acceso y que colabore para que los procesos académicos sean más eficientes y transparentes.

El Portal Administrativo y de Control de Actividades de Cátedra (Teo), posee una serie de herramientas que ayudarán a optimizar el proceso de entrega de notas a los estudiantes, colaborará con las tareas que realizan los tutores académicos como la generación y entrega de estadísticas de las evaluaciones realizadas, y la comunicación con los estudiantes.

En el primer capítulo de esta tesis se hace referencia a los antecedentes del proyecto realizado y el marco teórico utilizado para la elaboración del portal. En este último se incluye información sobre servicios web, desarrollo web y metodologías de desarrollo. En el segundo capítulo se define el problema a resolver, además se hace un análisis de las funciones de la aplicación.

Se presenta en los siguientes capítulos la documentación y comprensión del sistema por parte de los distintos usuarios. Por último, se presentan los resultados de una evaluación realizada sobre el funcionamiento del sistema junto a los tutores académicos actuales. Los resultados obtenidos fueron bastante satisfactorios.

OBJETIVOS

General

Diseñar e implementar una plataforma que administre las actividades de los auxiliares de cátedra y los estudiantes del Departamento de Matemática de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala.

Específicos

1. Mejorar el proceso de entrega de estadísticas a los jefes de área.
2. Gestionar notas, recursos didácticos y usuarios.
3. Permitir la comunicación entre auxiliar y alumnos
4. Implementar un sistema de notificaciones a través de correos electrónicos

INTRODUCCIÓN

La administración de actividades académicas es un conjunto de procesos que permiten administrar tareas involucradas en un periodo educativo. Los procesos académicos pueden involucrar la presentación de reportes, comunicación entre el alumno y el tutor y la gestión de notas, recursos didácticos y usuarios. Todas estas actividades llevadas por una institución académica son de suma importancia para conocer el resultado de un ciclo educativo.

Desempeñar estas actividades de forma manual implica que los procesos sean deficientes, pérdida de información valiosa o que los resultados no sean aceptados. Dados estos factores es necesario sistematizar las actividades con apoyo de las tecnologías actuales, para reducir los problemas, aumentar la productividad al reducir tiempos de trabajo, logrando procesos eficientes y estandarizados.

En el Departamento de Matemática de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala se llevan a cabo todos los procesos antes mencionados de manera manual, dado que este departamento maneja uno de los grupos de estudiantes más grandes de la Facultad de Ingeniería, la información es de gran volumen, lo que conduce a caer en retrasos en la presentación de las notas a los estudiantes y en la entrega de estadísticas a los jefes de área del departamento. Otras Escuelas y Departamentos de la Facultad de Ingeniería cuentan con sistemas que optimizan las actividades académicas o herramientas de aprendizaje a distancia.

1. MARCO TEÓRICO

1.1. Antecedentes

En este capítulo se describen los antecedentes de proyectos con características funcionales similares a los que se desarrollarán en el Portal Teo.

1.1.1. Trabajo de graduación de Pablo Alejandro Reyes Orozco

En octubre de 2007 se le confirió el título de Ingeniero en Ciencias y Sistemas en la Facultad de Ingeniería en la Universidad de San Carlos a Pablo Alejandro Reyes Orozco por la realización de una tesis de aplicación en el Departamento de Matemática.

Cuando se realizó este proyecto, la forma en la que se llevaban a cabo los procesos de la gestión de notas de los estudiantes de los cursos del Área de Matemática es básicamente la misma que fue descrita en la Justificación del Problema del presente trabajo de tesis.

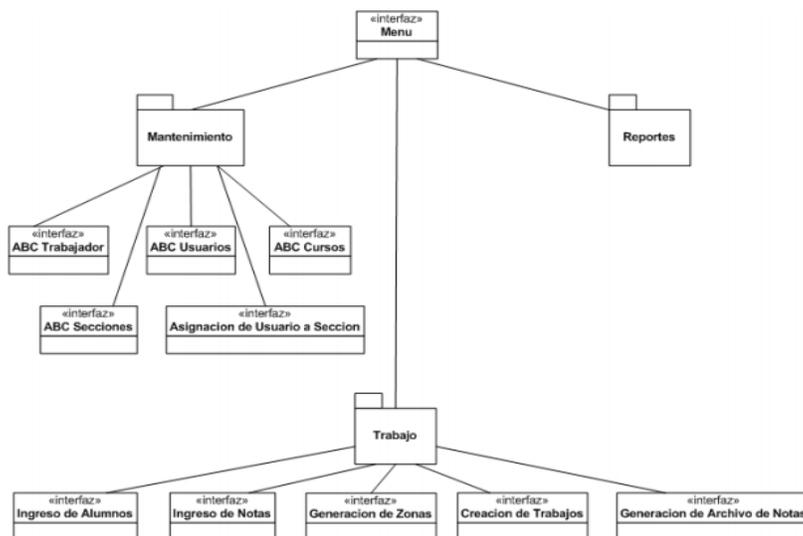
La solución que propuso Pablo Reyes en ese entonces fue la de un sistema informático que constaba de los siguientes módulos y características:

- Una base de datos centralizada.
- Un módulo de ingreso de usuarios: a través de este módulo se le permitía a un usuario que poseía el rol de administrador poder crear, modificar y eliminar usuarios según las necesidades que se presentaban.

- Un módulo para el ingreso de notas: este módulo brinda la facilidad de registrar las calificaciones de las actividades académicas como tareas, exámenes parciales y exámenes finales, además de permitir la exportación de estos registros hacia un archivo de con extensión xls (Excel).
- Un módulo para la generación de estadísticas: este módulo tenía la capacidad de generar automáticamente las estadísticas que se le entregan a los jefes de área del Departamento de Matemática.
- Módulo para el ingreso de estudiantes: este permitía llevar el control de los estudiantes asignados a cada curso.

Gráficamente, la estructura funcional del sistema desarrollado por Pablo Reyes puede observarse en el siguiente diagrama de presentación:

Figura 1. **Estructura funcional del proyecto de Pablo Reyes**



Fuente: REYES, Pablo. *Trabajo de graduación de Pablo Reyes*.

http://biblioteca.usac.edu.gt/tesis/08/08_0345_CS.pdf. Consulta: marzo de 2019.

El sistema fue desplegado en una red de área local (LAN) que contaba con un servidor central y siete computadoras que se conectaban a dicho servidor. Estas computadoras eran operadas por los catedráticos y algunos auxiliares. El servidor antes descrito poseía las siguientes características:

- Sistema operativo: Microsoft Windows Server 2003
- Base de datos: SQL Server 2005
- Memoria RAM: 1 GigaByte
- Disco duro: 160 GigaBytes
- Procesador: Intel Pentium Dual Core 2,8 Ghz.

La solución propuesta por Pablo Reyes cubría muchas de las necesidades que tenía el Departamento de Matemática respecto a la gestión de notas y la generación de estadísticas, sin embargo, actualmente no se utiliza el sistema y no se cuenta con evidencia que indique que el sistema se implementó para uso de los catedráticos, y si así lo fue, se desconoce cuánto tiempo fue empleado y por qué se discontinuó su uso.

1.1.2. Matemática en línea

Esta es una plataforma lanzada en el año 2013, desarrollada por una empresa guatemalteca del sector privado, cuyo nombre es Royale Studios.

Matemática en Línea es una plataforma de *e-learning* a través de la cual se proporciona recursos didácticos a los estudiantes de los cursos del Área de Matemática Básica, Matemática Intermedia y Matemática Aplicada. Estos recursos didácticos varían entre ejercicios resueltos y material de refuerzo complementario para que el estudiante posea un método adicional a las clases

magistrales para su aprendizaje. La forma en la que funciona esta plataforma es la siguiente:

- Los catedráticos de las diferentes secciones crean lo que en la plataforma lo que se denomina un Curso.
- Los estudiantes crean sus usuarios en la plataforma y pueden realizar el proceso de automatriculación en las secciones de su interés.
- El catedrático publica la información que desee a los estudiantes.
- Los estudiantes pueden resolver sus dudas haciendo consulta a los catedráticos.

Abordando el punto de vista técnico, la plataforma fue desarrollada con la herramienta WordPress, un sistema de gestión de contenidos (CMS). La plataforma muestra también a los usuarios una interfaz gráfica agradable e intuitiva.

Esta herramienta, a pesar de que puede resultar bastante útil tanto para catedráticos como para los estudiantes, no es muy utilizada. Son pocas las secciones que se encuentran registradas en la plataforma y no todas presentan mucha actividad por parte de los estudiantes. Es importante mencionar que esta plataforma no permite la gestión de las calificaciones en las diferentes actividades académicas realizadas por los estudiantes.

1.1.3. Página del Departamento de Matemática de la Facultad de Ingeniería

Este es el sitio oficial del Departamento de Matemática de la Facultad de Ingeniería. En este se encuentra toda la información como objetivos, misión y visión, historia y la estructura organizacional del Departamento de Matemática.

Una de las utilidades más importantes del sitio es que en este se publican los programas de todos los cursos impartidos por el Departamento de Matemática, para que estén disponibles para los estudiantes. Se encuentran también los horarios en los que se imparten las clases magistrales de las diferentes secciones.

Otra de las funciones importantes del sitio es que se encuentran publicadas las claves de los exámenes parciales, exámenes finales y exámenes de retrasada de la mayoría de los cursos. Las claves contienen la resolución detallada de los problemas evaluados en la mayoría de los exámenes realizados. Este es un recurso muy importante para los estudiantes debido a que son un gran apoyo al estudiar para los exámenes. Actualmente el sitio se encuentra vigente y sigue siendo útil para el Departamento de Matemática. Cabe mencionar también que el sitio no posee un módulo para la gestión de notas de los estudiantes.

1.2. Diseño y desarrollo web

Es muy habitual que el desarrollo y el diseño web sean confundidos, que se utilice un término para definir las funciones del otro o referirse a ambas como una sola. Está claro que ambas juegan un papel fundamental en la elaboración de un sitio web, pero cada una tiene un rol distinto y tiene diferentes objetivos. Esto no quiere decir que sean totalmente independientes una de la otra, es necesario un nivel de sinergia para mantener comunicado y hacer funcional el sistema.

1.2.1. Desarrollo web

El desarrollo web es el proceso de programación que permite la construcción de la lógica del sitio web. Este se encuentra repartido en dos partes, el cliente y el servidor, que en la mayoría de los casos se encuentran en comunicación constante. Tanto el cliente como el servidor constan de distintos lenguajes en los que pueden ser desarrollados, para el cliente existe Dom, Visual Basic, Script, entre otros, y para el servidor se cuenta con PHP, ASP.NET, JSP, entre otros, algunos lenguajes como JavaScript pueden estar en ambos lados. Todo este código que el usuario no ve es conocido como *back-end*. La interacción entre el cliente y el servidor permite el flujo de información bilateral, que tanto el cliente como el dueño del servicio utilizan para su beneficio y que se encuentra almacenado en una base de datos.

1.2.1.1. Clientes web

Es un dispositivo de cómputo perteneciente a una red, el cual es capaz de obtener información y aplicaciones provenientes de un servidor. Los navegadores son interpretados como los clientes de cualquier servicio en la web. A través de estas herramientas se dibujan los datos solicitados por un usuario, datos enviados desde el servidor, y de igual forma los usuarios son capaces de enviar información al servidor.

1.2.1.2. Aplicaciones web

Es un programa de aplicación que está alojado en un servidor remoto y que es entregado al usuario vía Internet en un navegador. Según Jarel Remick, cualquier componente en un sitio, que realice una función en la que el usuario interviene, es una aplicación web.

1.2.1.3. Web service

Un servidor web o *web service* cumple una tarea específica o un conjunto de tareas, que permiten la comunicación entre aplicaciones. El *web service* presta sus servicios y funcionalidades a los clientes a través de los protocolos estándar de Internet, de esta forma no es necesaria la intervención de un operario para terminar el proceso. Otra función del servidor web es interactuar con la base de datos para devolver la información solicitada y de igual forma enviar la información a este.

1.2.2. Diseño web

Por muchos años el diseño web jugaba un papel secundario en la creación de sitios web, principalmente por las limitantes que existían, no se contaba con librerías o herramientas que sirvieran para dar estilo al contenido. HTML como lenguaje de etiquetas en sus inicios permitían definir atributos básicos, que no permitían diseñar una interfaz muy agradable al usuario. Los diseñadores tenían que utilizar gráficas o imágenes para dar estilo a las aplicaciones. Con la integración de CSS, el diseño dio un giro importante y los diseñadores comenzaron a tener mayor control sobre los sitios.

De igual forma en los inicios de la web y las primeras aplicaciones, el programador jugaba el rol del diseñador, esto tampoco ayudaba en la generación de una interfaz agradable a usuario final, con el paso del tiempo se fue creando el rol del diseñador web. El diseñador web no solo se ocupa de la parte gráfica, tiene entre sus obligaciones más comunes determinar los objetivos del proyecto, las necesidades del usuario, define la arquitectura web y el contenido del sitio. Además, estudia la interacción de los usuarios con las

funciones del sitio. En resumen, es el encargado de la navegación, usabilidad, interacción, arquitectura de la información y de la parte gráfica del sitio web.

1.2.2.1. Diseño adaptativo

Los dispositivos móviles han representado un problema para el desarrollo web, que por muchos años solo se tenía que preocupar por un único diseño, en la actualidad se debe diseñar varias veces el mismo sitio para distintos dispositivos o diseñar soluciones que se adapten al tamaño de cada dispositivo, en el que se pueda interactuar con la aplicación. La técnica para hacer a un sitio adaptable a los distintos tamaños de pantalla es *graceful degradation*, que reduce el tamaño progresivamente con respecto a la disminución de las capacidades del navegador, con la finalidad de hacer la aplicación usable sin importar el tamaño del dispositivo.

1.3. Análisis y diseño de software

El portal ha sido diseñado bajo el patrón de desarrollo modelo vista controlador (MVC). Se ha implementado con el *framework* de desarrollo Laravel, con lenguaje PHP y JavaScript.

1.3.1. Metodologías de desarrollo

Una metodología de desarrollo está compuesta principalmente por una serie de técnicas, métodos y herramientas que facilitan el desarrollo de un sistema de software. Usualmente estas metodologías poseen cierta documentación que facilitan el entendimiento de las diversas técnicas a utilizar, lo cual es muy importante porque para que la metodología tenga éxito las

personas involucradas en el proyecto deben comprender perfectamente la metodología y el ciclo de vida del software que se planea seguir.

Actualmente, existe una gran cantidad de metodologías para el desarrollo de software, cada una con características distintas que hacen que se adapten a los diferentes requerimientos que pueda presentar un proyecto. A continuación, se detallan algunas de las metodologías:

- *Code and fix*: este modelo consta únicamente de dos pasos. El primero es la codificación, que es el acto de desarrollar software tratando de que el resultado proporcione los objetivos deseados. El segundo paso es la reparación, que consiste en corregir el código hasta que el software funcione. Esto se hace normalmente añadiendo más código.
 - Ventajas:
 - La aplicación de este modelo no requiere de mucha planificación.
 - Este puede ser útil en tareas o proyectos de software pequeños que no presentan mucha complejidad.
 - Desventajas:
 - Al no llevar a cabo una adecuada planificación o preparación previa puede caerse en el desperdicio de tiempo y recursos, tomando el gran riesgo de conducir el proyecto al fracaso.

- Modelo en cascada: este modelo ordena rigurosamente el proyecto de software en las siguientes etapas: toma y análisis de requerimientos, diseño del sistema, codificación, pruebas y mantenimiento. El inicio de cada una de las anteriores etapas debe esperar la finalización de la etapa anterior. Este fue el primer modelo en utilizarse formalmente.
 - Ventajas:
 - Este modelo impulsa una metodología de trabajo efectiva al definir.
 - Es uno de los modelos más eficientes en proyectos donde existe mucha burocracia, ya que debe existir una documentación extensiva antes de diseñar y diseñar antes de codificar.
 - Desventajas:
 - Es difícil poder desarrollar un proyecto de forma lineal, ya que en muchos casos los requerimientos cambian, entonces debe rediseñarse el sistema.
 - Tienen un costo más alto ya que se debe pasar por todas las fases para poder descubrir errores.
- Desarrollo evolutivo: este modelo consta del desarrollo de una versión inicial que luego se va detallando y mejorando hasta que se va cumpliendo con los requerimientos del cliente o el usuario final. Posee la característica de que las fases de especificación, desarrollo y validación se entrelazan en cada una de las iteraciones. Se vale de prototipos para ayudar al cliente a entender los requerimientos del cliente. Se van

entregando varias versiones intermedias hasta que se llega a una versión final.

- Ventajas:
 - Al usar prototipos puede finarse los requerimientos al poder recibir una mejor retroalimentación por parte del cliente.
 - El hecho de que las fases de especificación, desarrollo y validación vayan de la mano, ayuda a la identificación de posibles errores.
- Desventajas:
 - La realización de prototipos puede implicar mucho tiempo.
 - La corrección de las fallas no podrá presentarse hasta el lanzamiento de la siguiente versión.
- Modelo iterativo incremental: en este modelo el cliente indica al equipo de desarrollo del proyecto las funcionalidades que debe tener el software a desarrollar, así puede definirse varios incrementos dando prioridad a los requerimientos más importantes. Cada requerimiento suele desarrollarse utilizando un modelo de cascada, lo que implica que al inicio de cada incremento se debe tener una especificación detallada de la funcionalidad a desarrollar.
 - Ventajas:
 - Al desarrollarse el producto en incrementos aplicando el desarrollo en cascada, puede tenerse la certeza de que las

funcionalidades entregadas cumplan con los requerimientos del cliente.

- Aconsejable si el software a desarrollar tiene bien definidas sus funcionalidades.

- Desventajas:

- Al no estar bien definidos los requerimientos el software desarrollado puede estar mal estructurado, por lo que posteriormente será difícil de mantenerse.

- Modelo en espiral: este modelo de desarrollo se organiza en forma de espiral. En cada iteración se llevan a cabo un conjunto de actividades, en las cuales no existe ninguna prioridad definida. La característica principal de este modelo es que el orden de las actividades a realizar se elige haciendo un análisis de riesgo.

- Ventajas:

- La mayor ventaja es que al definir cada uno de los riesgos pueden evitarse ciertos problemas, ayudando a conducir al proyecto de software al éxito.

- Desventajas:

- Al realizarse un análisis de riesgo en cada iteración se puede incurrir en un costo muy elevado.
- Es aconsejado si se tiene bastante tiempo para llevar a cabo el proyecto de software.

- Modelos de desarrollo ágil: este modelo se basa en el modelo de desarrollo iterativo incremental, con la característica de que el trabajo es realizado mediante la colaboración de equipos autoorganizados. Se encuentran inmersos en un proceso compartido de toma de decisiones a corto plazo.
 - Ventajas:
 - Una de las principales ventajas es la rápida respuesta a los cambios.
 - El cliente interviene positivamente en el proceso.
 - Se eliminan tareas innecesarias como documentación extensiva.
 - Desventajas:
 - Al ser una metodología ágil la documentación es bastante escasa.

2. DISEÑO DEL PORTAL TEO

2.1. Definición del problema

En apoyo a los catedráticos del Departamento de Matemática, los auxiliares de cátedra I y II deben mantener un registro de las calificaciones obtenidas por los estudiantes en diversas actividades. Este registro suele hacerse en hojas de cálculo, pero por lo regular cada auxiliar de cátedra maneja la información en forma distinta y en periodos de tiempo diferentes, lo que representa, en algunos casos, inconvenientes debido a que no existe una forma estándar y sistematizada para tratar la información. Esto repercute en forma de atrasos o incumplimiento en actividades como la entrega de las estadísticas de los distintos exámenes realizados a los jefes de área, o la entrega de calificaciones a los jefes inmediatos.

Otra fragilidad que puede encontrarse en el sistema de evaluación es que, en muchos casos, por distintas razones, el estudiante no está al tanto de los resultados de su desempeño en los cursos del área de matemática. Esta desinformación puede ocasionar que el estudiante presente falta de compromiso con las actividades a realizar y en el peor de los casos la deserción en los cursos. Además, a veces no se utilizan medios que faciliten la comunicación estudiante-auxiliar, por lo que no se podrán solventar las dudas que se tengan en relación con los cursos.

Ante las dificultades anteriormente descritas, el proyecto TEO plantea implementar una solución que mejore los procesos académicos que involucren a los auxiliares de cátedra y a los alumnos pertenecientes a una o varias clases gestionadas por el Departamento de Matemática. El sistema deberá ser capaz de centralizar toda la información referente a las calificaciones en las diferentes actividades, para que de esta forma el estudiante pueda estar al tanto de su desempeño. Se pretende implementar una vía de comunicación a través del portal para que el estudiante pueda solventar sus dudas directamente con el auxiliar de cátedra y para que este último pueda dar anuncios o compartir recursos didácticos a los estudiantes.

2.2. Análisis de requerimientos

Para conocer las funcionalidades del portal se detallan a continuación las necesidades de los interesados en el proyecto.

2.2.1. Requerimientos funcionales

A continuación, se definirán los requerimientos de alto nivel de los interesados para la creación del Portal Teo.

2.2.1.1. Usuarios

El sistema requiere que se puedan realizar las siguientes actividades con respecto a los usuarios.

- Se requiere que se puedan crear, modificar, visualizar y eliminar usuarios.

- En el sistema deberán existir tres tipos de usuario: usuario administrador, usuario auxiliar y usuario estudiante.
- El usuario administrador deberá ser el usuario que cree a los usuarios con el rol de usuario auxiliar.
- Los usuarios podrán realizar el proceso de inicio de sesión a través de su usuario y una contraseña.

2.2.1.2. Secciones de clase

En la plataforma se deberá poder realizar las siguientes operaciones respecto a las secciones de los diferentes cursos:

- Se podrán crear las secciones al iniciar cada semestre. Se le deberá asignar uno de los cursos impartidos por el Departamento de Matemática, se le deberá colocar el catedrático que impartirá el curso, el año y el semestre, y se le deberá asignar también el usuario auxiliar que estará a cargo de dicha sección.
- Las secciones podrán modificarse o eliminarse dependiendo de las necesidades.
- La administración de las secciones de cursos podrá realizarse únicamente por el usuario administrador.

2.2.1.3. Matriculación por curso

Se requiere que los estudiantes puedan realizar los siguientes procesos respecto a las secciones de clase:

- Los estudiantes podrán matricularse en una sección por curso.

- El estudiante también podrá realizar el proceso de desmatriculación en un curso.
- El rol de usuario auxiliar podrá llevar a cabo un filtrado de los alumnos asignados en el Centro de Cálculo con los alumnos que se han matriculado en la sección que administran.

2.2.1.4. Notas y actividades de clase

El sistema requiere que se implemente un módulo para llevar control de las actividades realizadas en los cursos.

- Se podrán crear, visualizar, actualizar y eliminar actividades en cada una de las diferentes secciones. Estas actividades podrán pertenecer a las actividades de curso ponderadas o no ponderadas.
- Se requiere que las notas de las actividades puedan ser ingresadas a la plataforma mediante un proceso de carga masiva, utilizando un archivo de tipo csv.
- También podrán ingresarse las notas al sistema individualmente, seleccionando la actividad y al estudiante en cuestión.
- Se necesita que las notas puedan modificarse en todo momento.
- Las actividades de administración de notas y actividades de clase podrán ser llevadas a cabo únicamente por el usuario auxiliar asignado a la respectiva sección.

2.2.1.5. Estadísticas

Se deberá implementar un módulo para la generación de estadísticas.

- El sistema podrá generar estadísticas automáticamente de los exámenes parciales, exámenes finales y de los exámenes de retrasada en todos los cursos registrados en el sistema.
- También se podrán generar estadísticas de las notas finales de cada sección.
- Las estadísticas podrán ser enviadas automáticamente a los jefes de área.
- Cada usuario auxiliar será el encargado del manejo de las estadísticas de la sección que administra.

2.2.1.6. Calendario de actividades

Se requiere que se implemente un módulo para alertar a los estudiantes y auxiliares de las actividades que deben realizar.

- Cada sección deberá tener su propio calendario. El usuario auxiliar será el encargado de la creación de este para la sección que administra.
- El sistema deberá enviar alertas al usuario auxiliar recordándole que tiene que realizar el proceso de generación y entrega de estadísticas.
- También se le deberá enviar alertas al usuario auxiliar para que esté al tanto de cuándo debe entregar una actividad.
- Las alertas deberán enviarse a través de la plataforma y por medio de correo electrónico.

2.2.1.7. Reportes

Se necesita que se cree un módulo para la generación de reportes.

- El usuario auxiliar podrá generar reportes de notas por actividad, reportes de asistencia y reporte general de notas.
- A través de la plataforma podrá imprimirse el reporte generado.

2.2.1.8. Comunicación auxiliar-estudiante

Se requiere la implementación de un módulo que facilite la comunicación entre el auxiliar y los estudiantes.

- El usuario auxiliar podrá publicar información de eventos a realizarse.
- Deberá existir un chat entre el usuario estudiante y el usuario auxiliar.
- Se deberán enviar notificación vía correo electrónico de que hay un mensaje, tanto para el usuario auxiliar como para el usuario estudiante.

2.2.1.9. Recursos didácticos

Con el fin de apoyar al estudiante se desea que se implemente un módulo para la gestión de recursos didácticos.

- Se deberá poder crear, visualizar, modificar y eliminar recursos didácticos a lo largo de todo el semestre para cada una de las secciones.
- El encargado del manejo de los recursos didácticos deberá ser el usuario auxiliar.

2.2.2. Requerimientos no funcionales

A continuación, se definirán los requerimientos de bajo nivel de los interesados para la creación del Portal Teo.

2.2.2.1. Usabilidad

La plataforma debe presentar una interfaz de usuario agradable e intuitiva, con textos de ayuda e íconos ilustrativos para que el usuario pueda comprender el funcionamiento y sea fácil la interacción con el sistema.

2.2.2.2. Multiplataforma

Se requiere que el portal TEO sea una aplicación multiplataforma, es decir, que el acceso desde diferentes dispositivos no afecte la funcionalidad ni la usabilidad del sistema. Esto se logrará a través de un diseño responsivo de las pantallas de la aplicación.

2.2.2.3. Seguridad

Debido a la sensibilidad de la información que se va a almacenar en el sistema, se necesita que el acceso a la información sea restringido únicamente para los roles indicados, según fue descrito en los requerimientos funcionales. Para proveer de un nivel de seguridad más alto en la plataforma se deberá encriptar las contraseñas que sean ingresadas en el portal.

Se requiere también que la plataforma cuente con un certificado HTTPS para que la transferencia de datos sea de forma segura.

2.2.2.4. Disponibilidad

Ya que los usuarios pueden tener la necesidad de interactuar con la plataforma en diversos instantes del día, se necesita que la plataforma presente alta disponibilidad, funcionando a toda hora. Se requiere también que el sistema presente una alta capacidad para el manejo de la concurrencia debido a que muchos estudiantes y el auxiliar pueden solicitar el acceso a la plataforma al mismo tiempo.

2.3. Arquitectura utilizada

El tipo de arquitectura implementada para el sistema es una arquitectura de tipo cliente-servidor. De esta forma los usuarios podrán realizar peticiones por medio del protocolo https y recibirán una respuesta dependiendo del servicio web con el que interactúen.

El patrón de diseño utilizado es la programación por capas, el cual busca el desacoplamiento de las partes que componen el sistema. De esta forma será sencillo crear diferentes interfaces sobre el sistema sin la necesidad de realizar cambios en la capa lógica o de datos. Las capas que componen el sistema son las siguientes:

- Capa de presentación: esta capa será la que verán los diferentes usuarios, tanto usuarios administradores, como usuarios auxiliares y usuarios estudiantes. Es a través de esta capa por medio de la cual el usuario interactuará con el sistema.
- Capa lógica: en esta capa se define el funcionamiento de la lógica del sistema. Se colocarán todas las restricciones de seguridad de la aplicación.

- Capa de datos: es la última capa. En esta se define la forma en la cual se van a almacenar los datos, tanto en forma lógica como en forma física.

2.3.1. Tecnologías utilizadas

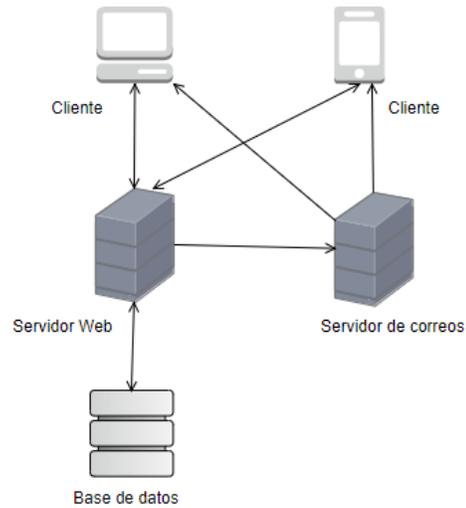
Las tecnologías utilizadas fueron cuidadosamente elegidas con el fin de satisfacer los requerimientos no funcionales descritos anteriormente, y también para agilizar el desarrollo de esta. A continuación, se mencionan las tecnologías utilizadas:

- Servidor Web: Apache 7
- Servidor de base de datos: MySQL
- Framework para el desarrollo: Laravel 5.8.3

2.3.2. Diagrama general de la arquitectura

A continuación, se muestra el diagrama general de la arquitectura:

Figura 2. **Arquitectura general**

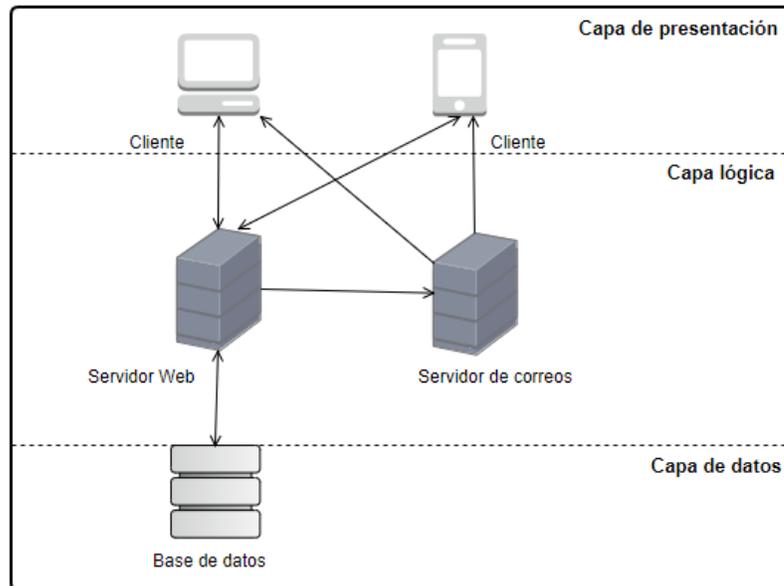


Fuente: elaboración propia.

2.3.3. Implementación del patrón de arquitectura

A continuación, se muestra el diagrama de la implementación del patrón de arquitectura:

Figura 3. **Modelo de tres capas**



Fuente: elaboración propia.

2.3.4. **Modelo de datos utilizado**

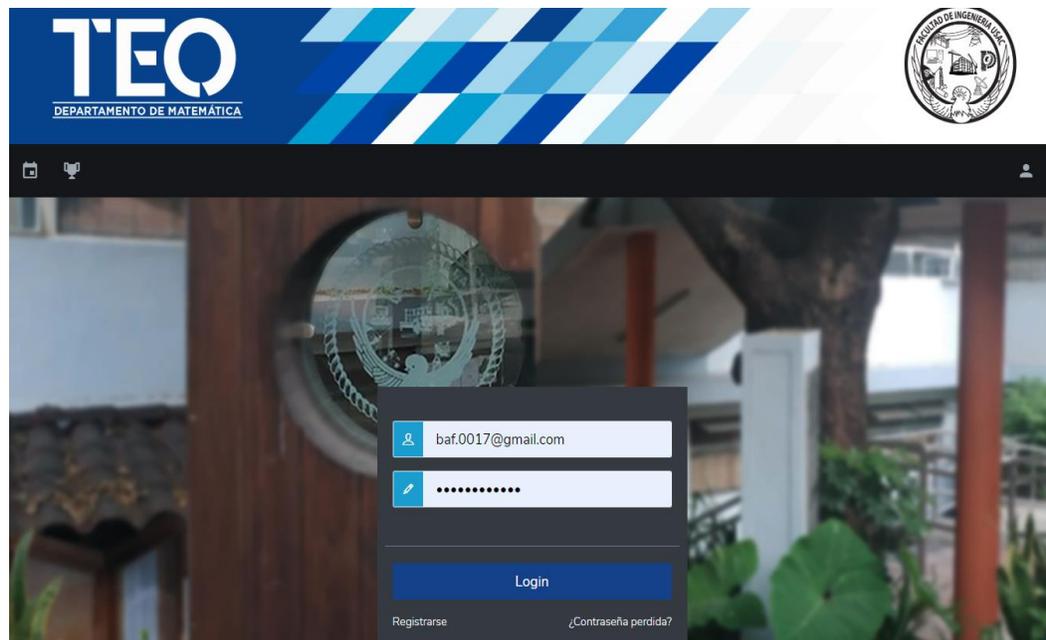
Se muestra el diagrama del modelo de datos utilizado:

3. DOCUMENTACIÓN

3.1. Manual del usuario

Se diseñó el portal TEO para la administración de las actividades de los auxiliares del Departamento de Matemática, de la Facultad de Ingeniería, Universidad de San Carlos de Guatemala, USAC, para que sea fácil de utilizar, tanto para los alumnos como para los auxiliares, haciendo más eficientes distintas tareas y procesos.

Figura 5. Pantalla de inicio de sesión



Fuente: elaboración propia.

La plataforma cuenta con distintas secciones, para auxiliares, alumnos y el administrador de la plataforma, las cuales serán descritas según su alcance y características.

3.1.1. Tipos de usuarios

El portal cuenta con tres distintos tipos de usuarios que son capaces de realizar distintas tareas dentro del portal, dependiendo de las credenciales con las que se cuenten. Estos tipos de usuarios permiten realizar todas las tareas que son permitidas en el portal y los usuarios dependen entre sí para un uso adecuado de las herramientas del portal

3.1.1.1. Usuario administrador

El sistema cuenta con un solo usuario administrador, para poder gestionar a los cursos, auxiliares, alumnos y catedráticos dentro del sistema. Son atribuciones y obligaciones del usuario administrador:

- Ingresar y mantener actualizados a los catedráticos
- Ingresar y mantener actualizados a los auxiliares
- Ingresar y mantener actualizados a las clases impartidas
- Dar de baja a un auxiliar
- Promover a un alumno a auxiliar
- Habilitar el periodo de matriculación de cursos

3.1.1.2. Usuario auxiliar

Un usuario auxiliar es un tipo de usuario que se encarga de la gestión de una o más clases. Son atribuciones y obligaciones del usuario auxiliar:

- Crear actividades dentro de sus clases
- Mantener actualizadas las notas de las actividades
- Publicar noticias sobre las clases
- Dar de baja a alumnos de sus clases
- Gestionar los recursos didácticos de sus clases

3.1.1.3. Usuario alumno

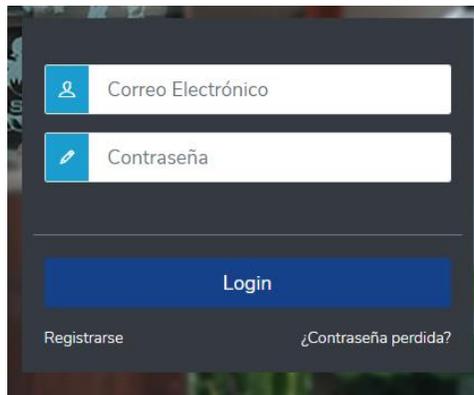
Un alumno es el usuario con menor privilegio dentro del sistema, es un usuario que solo puede ver los datos que le pertenecen. Son atribuciones y obligaciones del usuario alumno:

- Matricularse en las clases que le correspondan
- Ver sus notas
- Comentar noticias

3.1.2. Inicio de sesión

Para poder acceder al portal TEO, se deberá abrir cualquier explorador web y dirigirse a la siguiente dirección: teomateusac.ml. La página principal solicita el ingreso de las credenciales (figura 6), el correo electrónico y la contraseña registrada en el portal. Si estas credenciales concuerdan con las almacenadas se redirigirá a la pantalla de inicio dependiendo del tipo de usuario con el que se encuentre registrado.

Figura 6. **Formulario de credenciales**

El formulario de credenciales tiene un fondo oscuro. En la parte superior hay un campo de texto con un ícono de persona a la izquierda y el texto "Correo Electrónico". Debajo de eso hay otro campo de texto con un ícono de lápiz a la izquierda y el texto "Contraseña". En el centro hay un botón azul con el texto "Login". En la parte inferior izquierda hay un enlace "Registrarse" y en la parte inferior derecha hay un enlace "¿Contraseña perdida?".

Fuente: elaboración propia.

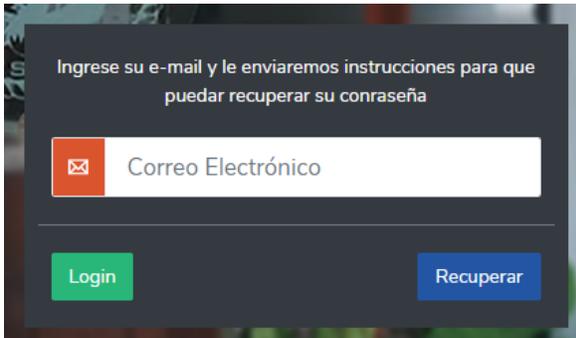
Si un usuario que no ha sido autorizado trata de ingresar a una página del portal, este será redirigido a la pantalla de inicio de sesión.

3.1.2.1. **Contraseña perdida**

Olvidar la contraseña para un usuario es un evento común, por lo que el portal cuenta con un proceso en el cual un usuario puede recuperar el acceso a este, mediante el correo que tenga registrado.

Para ello en la pantalla de inicio de sesión en la parte inferior derecha del formulario de credenciales, existe un enlace con el lema *¿Contraseña perdida?* Se abrirá un formulario (figura 7), que solicitará el correo electrónico registrado, al ingresarlo se enviará un mensaje al correo electrónico que se colocó en el formulario, con instrucciones para recuperar el acceso al portal TEO.

Figura 7. **Formulario de recuperación de acceso**



Formulario de recuperación de acceso. El formulario tiene un fondo oscuro con texto blanco. Encabezado: "Ingrese su e-mail y le enviaremos instrucciones para que puedan recuperar su contraseña". Campo de entrada: "Correo Electrónico" con un ícono de correo electrónico a la izquierda. Botones: "Login" (verde) y "Recuperar" (azul).

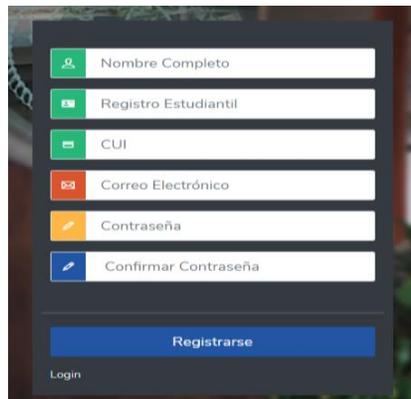
Fuente: elaboración propia.

3.1.3. **Registro de usuarios**

Cualquier alumno de la Facultad de Ingeniería, USAC, que esté cursando alguna materia del Departamento de Matemáticas, podrá registrarse en el portal TEO.

Para dirigirse a la pestaña de Registro debe ingresarse a la pantalla de inicio de sesión, posteriormente, en la parte inferior izquierda donde el usuario ingresa sus credenciales, podrá hacer clic en el vínculo con el texto *Registrarse*.

Figura 8. **Ventana de registro de usuario**



Fuente: elaboración propia.

Como requisitos para registrarse en el portal TEO, se deberá ingresar la siguiente información en el formulario de registro:

- Nombre completo.
- Registro estudiantil (número de carné).
- CUI.
- Correo electrónico.
- Contraseña, para brindar una mayor seguridad al usuario la contraseña debe de ser mayor a 8 caracteres, tener por lo menos una mayúscula, minúscula y un número.

Una vez ingresados estos datos un usuario puede terminar el proceso de registrarse en el portal.

Si todo salió bien se redirigirá a la pantalla de inicio y se le indicará con un mensaje que el proceso terminó con éxito, de no ser así un mensaje de error le indicará cuál ha sido el problema de su registro (figura 9).

Figura 9. **Mensaje de error durante el registro**



Fuente: elaboración propia.

3.1.4. Perfil

Todos los usuarios cuentan con datos que pueden ser editados por si ha ocurrido un error o si desea actualizarlos, para ello deberá dirigirse a la barra de menú en la parte superior del lado derecho, podrá encontrar tres íconos, para editar los datos del perfil deberá dar clic sobre el ícono más a la derecha, el ícono de usuario y luego elegir perfil.

Figura 10. **Datos personales**

Datos

Nombre	Teo
Registro Estudiantil	111111111
CUI	1111111111111
Correo	Teo@gmail.com
Contraseña Actual	Ingrese su contraseña actual
Contraseña Nueva	Ingrese una nueva contraseña

[Guardar](#) [Editar](#)

Fuente: elaboración propia.

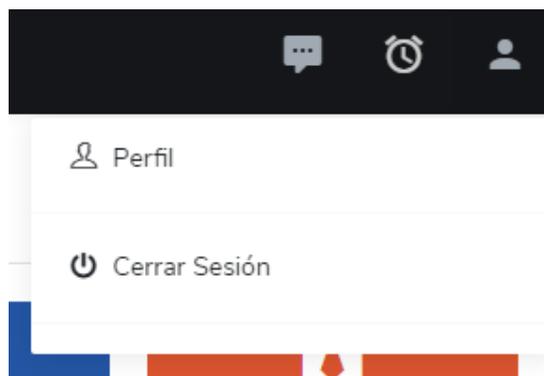
Los datos pueden ser editados, pero será solicitada la contraseña actual para efectuar cualquier cambio. Para habilitar la edición de datos se debe dar clic sobre el botón *Editar*. El registro estudiantil por cuestiones de seguridad no puede ser editado, el único usuario capaz de editar este registro es el administrador.

3.1.5. Cerrar sesión

Como medida de seguridad y contraparte de iniciar sesión, la opción cerrar sesión permite a los usuarios acabar con el privilegio de ejecutar las acciones a las que tienen acceso, según el tipo de usuario.

Al igual que Perfil, cerrar sesión se encuentra en el ícono de usuario, luego de seleccionarlo se puede encontrar la opción de cerrar sesión (figura 11), esta opción redirigirá a la pantalla de inicio de sesión. Si un usuario pasa 15 minutos inactivo la sesión caducará automáticamente.

Figura 11. Cerrar sesión



Fuente: elaboración propia.

3.1.6. Horario de cursos

En la barra de menú, en el segundo ícono de izquierda a derecha, puede encontrarse un enlace a los horarios de los cursos activos (figura 12), con el código de cada curso, nombre, sección, horario, catedrático y auxiliar.

Figura 12. Horario de cursos



The screenshot shows the website interface for the Department of Mathematics (TEQ). The header includes the TEQ logo and the text 'DEPARTAMENTO DE MATEMÁTICA' and 'Horarios'. A search bar is located at the top right. Below the header, there is a table with the following data:

Código	Curso	Sección	Horario	Salón	Catedrático	Auxiliar
101	Matemática Básica 1	C	7:00-9:00	107	Vera Marroquín	Cesar Morales
101	Matemática Básica 1	H	7:10-8:50	110	Renato Ponciano	Aux1
103	Matemática Básica 2	B	7:00- 9:50	310	Renato Ponciano	Brayan Alexander Flores
103	Matemática Básica 2	C	10:50-11:40	LII-6	Erika Cano	Cesar Morales

Fuente: elaboración propia.

3.1.7. Olimpiada Interuniversitaria de Ciencia y Tecnología.

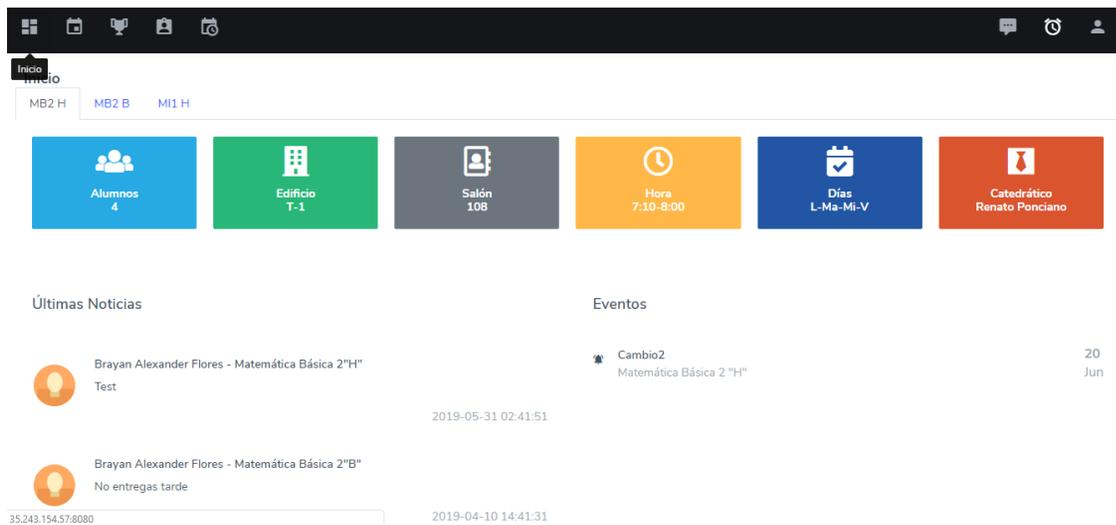
El portal cuenta con el enlace hacia la página oficial de la Olimpiada Interuniversitaria de Ciencia y Tecnología.

3.1.8. Pantalla de inicio

Al iniciar sesión el sistema redireccionará al usuario a la pantalla principal del portal (figura 13), esta pantalla cuenta con cierta información sobre el usuario. En la parte superior se podrá observar los cursos a los cuales se encuentra asignado, la siguiente fila de información se encuentra dividida en dos secciones, en el lado izquierdo puede encontrarse las noticias más recientes y por el lado derecho los eventos más próximos.

Se puede acceder a esta página en cualquier momento, al dirigirse a la barra de menú y seleccionar el botón de inicio, que se encuentra a la izquierda de todos los íconos.

Figura 13. Pantalla de inicio



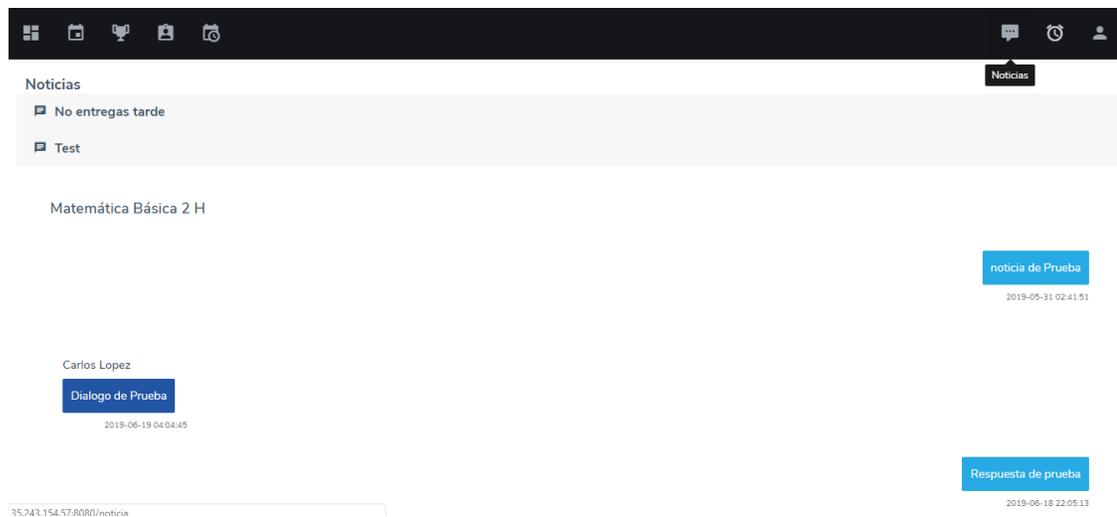
Fuente: elaboración propia.

3.1.9. Noticias

Como parte de la comunicación entre los auxiliares y alumnos, el portal cuenta con una sección para que estas dos partes puedan comunicarse adecuadamente (figura 14).

Un auxiliar crea una noticia sobre un curso y los alumnos pueden ver y comentar esta noticia, para mantener actualizadas a las partes, cuando se envía o publica la noticia o un mensaje, serán notificados por un correo electrónico.

Figura 14. Sección de noticias



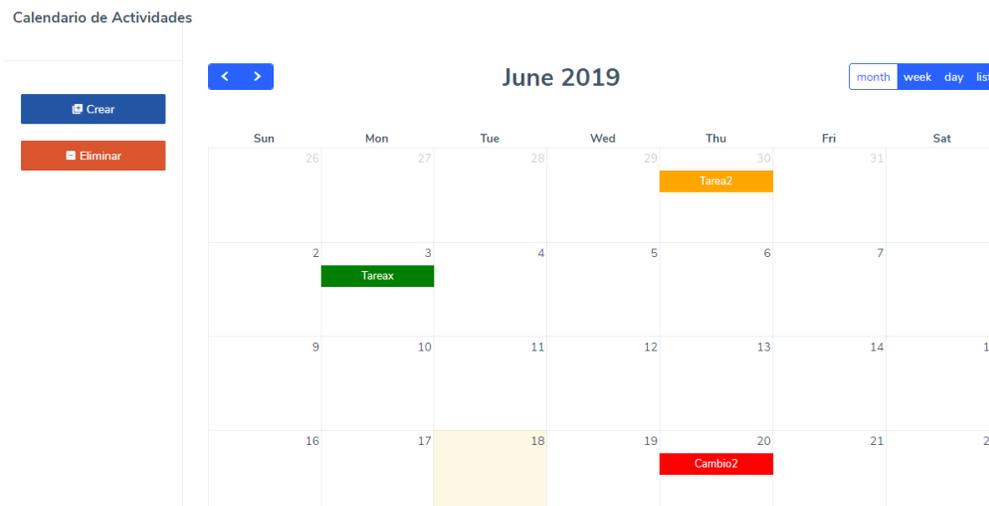
Fuente: elaboración propia.

Al igual que la pantalla de inicio, las noticias pueden ser accedidas en cualquier momento, al dirigirse al ícono de noticias, el primero de izquierda a derecha, en la parte derecha de la barra de menú.

3.1.10. Calendario

Al igual que las noticias, esta herramienta dentro del portal es compartida entre el usuario auxiliar y el alumno, con la diferencia que el auxiliar puede crear una actividad y el alumno solamente puede ver las actividades dentro del calendario (figura 15).

Figura 15. Calendario de actividades



Fuente: elaboración propia.

3.1.11. Cursos

Tanto los auxiliares como alumnos tienen cursos asignados, un auxiliar es asignado a un curso por el administrador y un alumno se matricula en un curso durante el periodo de asignación. La sección de cursos cuenta con diversas herramientas tanto para alumnos como para auxiliares.

Figura 16. **Listado de cursos asignados**



Fuente: elaboración propia.

Para elegir un curso al cual se está asignado, se puede dirigir a la barra de menú y dar clic sobre el cuarto ícono, de izquierda a derecha, este desplegará una lista de los cursos a los cuales el usuario se encuentra asignado (figura 16).

Al elegir uno de los cursos, el sistema redireccionará al usuario al módulo de los cursos y se habilitarán ciertas opciones si el usuario es auxiliar, de lo contrario, si el usuario se registra como alumno lo único que podrá verse en la sección de cursos serán las notas y los recursos del curso.

3.1.11.1. Listado de alumnos

Como una de las herramientas que son exclusivas del auxiliar, este podrá ver el listado de alumnos que estén asignados a uno de sus cursos, desde esta ventana el auxiliar será capaz de eliminar a uno de los alumnos del curso.

Podrá ordenar la lista de los alumnos de forma ascendente o descendente, por las cuatro columnas de datos que cuenta la lista, registro, CUI, nombre o correo electrónico. También podrá hacer búsquedas sobre cualquier campo.

3.1.11.2. Gestión de notas

La gestión de las notas es una herramienta de gran ayuda para el auxiliar, que le permite manejar las notas de todos sus cursos de diversas maneras. El auxiliar es capaz de modificar las notas directamente en la página o puede cargarlas de forma masiva con un archivo csv.

Las notas se muestran en una tabla (figura 17) y el auxiliar es capaz de filtrar por el nombre o registro de un estudiante, además puede crear o modificar las actividades de la clase.

Figura 17. Pantalla para gestión de notas

Mostrar 10 registros Buscar:

Registro	Nombre	HT1 - 2.00 pts	HT2 - 50.50 pts	Tarea1 - 1.50 pts
201402564	alumno test	0.00	0.00	0.00
201503528	alumno11	85.00	0.00	55.00
201503566	alumno2	15.00	0.00	75.00
201403568	alumno3	15.00	0.00	80.00

Mostrando registros del 1 al 4 de un total de 4 registros Anterior 1 Siguiente

Actividad
Guardar
Editar
Carga masiva

Fuente: elaboración propia.

Como medida de seguridad el sistema está configurado para guardar los cambios en las notas cada cinco minutos, para evitar cualquier pérdida de la información. La gestión de notas es una herramienta de uso exclusivo para los auxiliares.

3.1.11.3. Recursos

Los recursos son indispensables para mantener a los estudiantes al tanto en tareas, proyectos, apoyo didáctico, entre otros, por lo que el portal TEO tiene una sección para que un auxiliar pueda publicar un recurso sobre una clase (figura 18) y los alumnos asignados a esta puedan descargar los recursos compartidos.

Figura 18. Pantalla de recursos

MB2 H

Archivos disponibles

Show 10 entries

Search:

Nombre Recurso	Descripción	Descarga	Eliminación
Tarea1	Tarea1	Descargar	Eliminar

Showing 1 to 1 of 1 entries

Previous 1 Next

Subir recurso didáctico

Nombre del recurso

Descripción del recurso

Tipo de recurso

Seleccione archivo

Fuente: elaboración propia.

3.1.11.4. Estadísticas

Uno de los procesos más importantes para los auxiliares es la realización y entrega de las estadísticas de parciales, exámenes y notas finales. El portal cuenta con la automatización de este proceso, permite generar y enviar las estadísticas a los catedráticos correspondientes.

Las estadísticas es una herramienta de carácter exclusiva de los usuarios auxiliares, estos podrán generar las estadísticas de dos maneras, por tema o por nota final, además de poder descargar un archivo en formato PDF con los resultados de la estadística.

3.1.12. Administrador

La función principal del usuario administrador es mantener actualizada la información sobre los cursos, catedráticos y auxiliares, para que el portal funcione de manera adecuada.

3.1.12.1. Clases

En esta sección el administrador puede visualizar el listado de los cursos que se imparten actualmente (figura 19), además será capaz de añadir nuevas clases desde esta sección del portal.

Figura 19. Pantalla de administración de clases

Auxiliar	Curso	Sección	Horario	Edificio	Salón	Catedrático	Año	Semestre	Editar
Cesar Morales	Matemática Básica 2	C	10:50-11:40	T-3	LII-6	Erika Cano	2019	1	Editar
Brayan Alexander Flores	Matemática Básica 1	A	07:00 - 11:00	S-12	301	MARIO RENE DE LEON GARCIA	2019	1	Editar
Brayan Alexander Flores	Matemática Básica 1	C	7:00-9:00	S-12	107	Vera Marroquín	2019	1	Editar

Fuente: elaboración propia.

Para poder editar la información de un curso se debe dar clic sobre el botón *Editar* que se encuentra en la última casilla de cada registro. Esta acción redirigirá al usuario a una ventana donde se puede editar la información de la clase seleccionada (figura 20).

Figura 20. Pantalla de edición de clases

Datos

Auxiliar: Cesar Morales

Curso: Matemática Básica 2

Catedrático: Erika Cano

Sección: C

Edificio: T-3

Horario: 10:50-11:40

Salón: LII-6

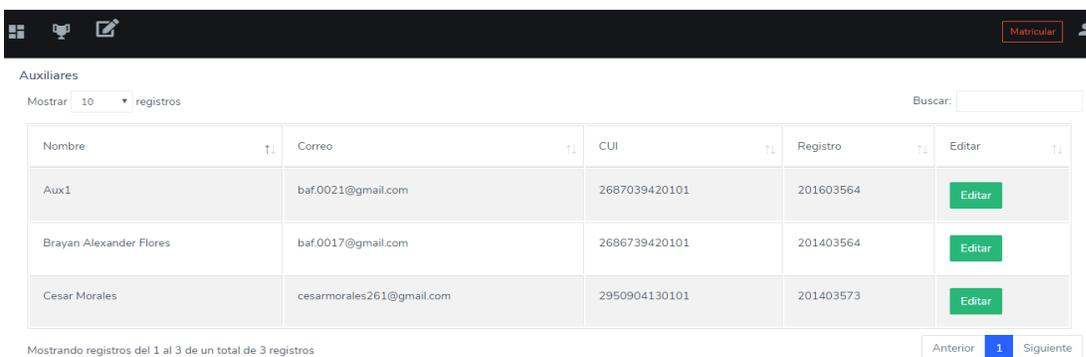
Guardar Editar

Fuente: elaboración propia.

3.1.12.2. Auxiliares

En esta sección el administrador puede visualizar el listado de los auxiliares activos dentro del sistema (figura 21), además será capaz de editar la información de un auxiliar y dar de baja si es necesario.

Figura 21. Pantalla de administración de clases



The screenshot shows a web interface for managing 'Auxiliares'. At the top right, there is a 'Matricular' button and a user profile icon. Below the header, the title 'Auxiliares' is displayed. A 'Mostrar' dropdown menu is set to '10 registros', and a 'Buscar:' search box is present. The main content is a table with the following data:

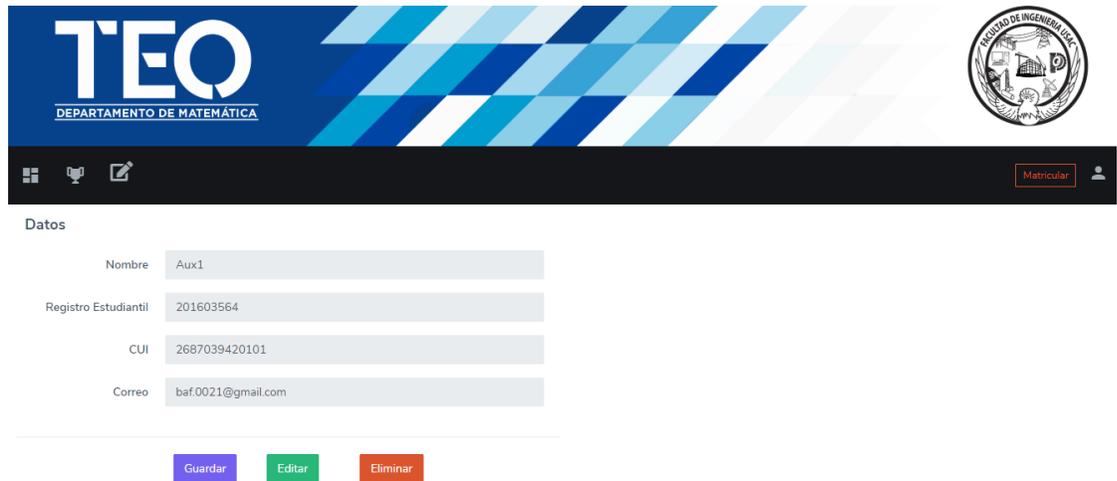
Nombre	Correo	CUI	Registro	Editar
Aux1	baf.0021@gmail.com	2687039420101	201603564	Editar
Brayan Alexander Flores	baf.0017@gmail.com	2686739420101	201403564	Editar
Cesar Morales	cesarmorales261@gmail.com	2950904130101	201403573	Editar

At the bottom left, it says 'Mostrando registros del 1 al 3 de un total de 3 registros'. At the bottom right, there are navigation buttons: 'Anterior', '1' (highlighted), and 'Siguiete'.

Fuente: elaboración propia.

Para editar la información de un auxiliar se debe dar clic sobre el botón *Editar* que se encuentra en la última casilla de cada registro. Esta acción redirigirá al usuario a una ventana donde se puede editar la información del auxiliar seleccionado o convertirlo a un usuario alumno (figura 22).

Figura 22. Pantalla de edición de auxiliares



Nombre	Aux1
Registro Estudiantil	201603564
CUI	2687039420101
Correo	baf.0021@gmail.com

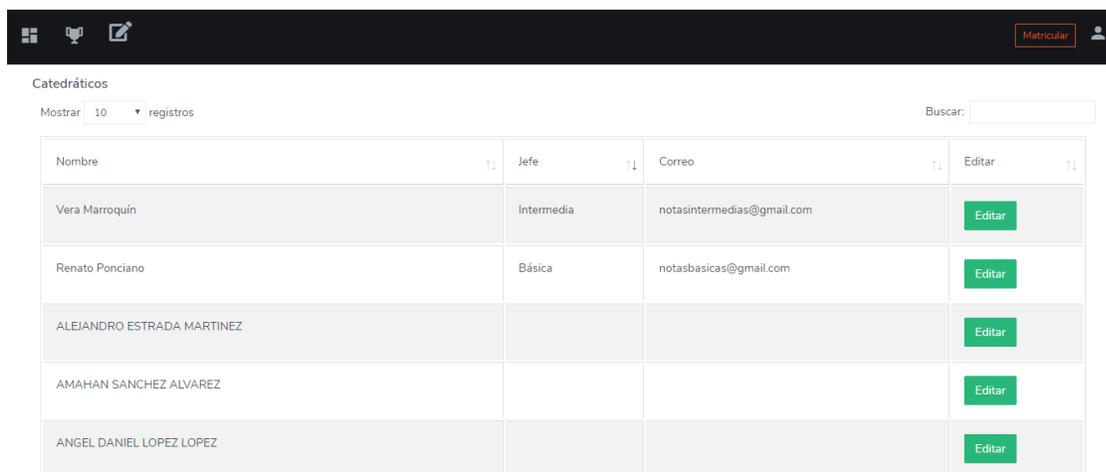
[Guardar](#) [Editar](#) [Eliminar](#)

Fuente: elaboración propia.

3.1.12.3. Catedráticos

El administrador puede visualizar el listado de los catedráticos en el sistema (figura 23), será capaz de editar, promover a jefe o dar de baja.

Figura 23. **Pantalla de administración de clases**



Nombre	Jefe	Correo	Editar
Vera Marroquín	Intermedia	notasintermedias@gmail.com	Editar
Renato Ponciano	Básica	notasbasicas@gmail.com	Editar
ALEJANDRO ESTRADA MARTINEZ			Editar
AMAHAN SANCHEZ ALVAREZ			Editar
ANGEL DANIEL LOPEZ LOPEZ			Editar

Fuente: elaboración propia.

Si se desea editar la información de un catedrático se debe dar clic sobre el botón *Editar*. Esto redireccionará al usuario a la ventana donde puede editarse la información del catedrático seleccionado o convertirlo en jefe de área, agregando una de las tres áreas permitidas y colocando su correo electrónico, de igual manera al quitar esta información de un jefe se volvería un usuario de tipo catedrático de nuevo.

3.2. **Manual técnico**

Con el fin de facilitar la comprensión de la estructura y la forma en la que se implementó el sistema se presenta este manual técnico, el cual contiene las especificaciones técnicas más importantes del sistema desarrollado. Este se encuentra dirigido a los encargados del mantenimiento y al administrador de dicho sistema.

3.2.1. Características generales

Se presentan algunas de las características del sistema:

- Nombre: Portal TEO
- Versión del sistema: 1.0
- Tipo de software: aplicación web
- Fecha de publicación: junio de 2019

3.2.2. Definición de las reglas de negocio implementadas

Para desarrollar el sistema se tomaron en cuenta ciertos lineamientos para protección y confidencialidad de la información del departamento de matemática, así como para comodidad de uso para los diferentes usuarios de la aplicación.

- Acceso a la información: debido a la importancia de la información que pueda ser registrada en el sistema, el acceso debe restringirse para las diferentes entidades que interactúen con el mismo. Entre las restricciones de acceso pueden encontrarse las siguientes:
 - Los usuarios que tengan el rol de Estudiante no podrán modificar notas, ni información correspondiente a tutores académicos, catedráticos o cursos. Únicamente podrán modificar su información personal y podrán visualizar sus notas en las actividades programadas por los tutores.
 - Los usuarios con rol de tutor académico podrán modificar notas de los estudiantes que estén asignados en sus cursos, así como programar actividades. No podrán modificar información respecto a otros tutores académicos o catedráticos.

- El usuario con rol de administrador sí podrá modificar información referente a cursos, catedráticos y tutores académicos. Puede visualizar información correspondiente a notas, pero no puede modificarlas.
- Acceso al sistema: los diferentes usuarios obtendrán los distintos roles de la siguiente forma:
 - Inicialmente en la base de datos se encuentra registrado un único usuario con el rol de administrador. Este ya cuenta con una contraseña predefinida, la cual después de la entrega del software deberá ser cambiada en el próximo inicio de sesión por motivos de seguridad. Si por razones administrativas se decide que debe haber más de un usuario administrador, el encargado de mantenimiento del sistema deberá agregar otro usuario a la base de datos directamente en el servidor de base de datos.
 - Los usuarios con rol de tutor académico únicamente podrán ser creados por el usuario Administrador por medio del portal. De igual manera, será el usuario administrador quien le asigne los cursos y secciones que podrá administrar. Una vez que se haya creado el usuario se deberá cambiar la contraseña con la cual podrá acceder al portal.
 - Los estudiantes sí podrán registrarse en el sistema y se les asignará automáticamente el rol de estudiante.
- Privacidad: uno de los requerimientos a los que se les dio mayor énfasis es que los usuarios tuvieran acceso únicamente a la información que les corresponde. Esto quiere decir que el usuario estudiante únicamente

podrá visualizar sus notas y no las de otros estudiantes matriculados en la misma sección de clase.

3.2.3. Fundamentación de la tecnología utilizada

Las tecnologías utilizadas para el desarrollo y funcionamiento del sistema fueron las siguientes:

- Base de datos: MySQL
 - Justificación: se decidió utilizar este servidor de base de datos debido a que presenta alto rendimiento bajo demanda y es bastante flexible si se necesitara hacer cambios en la base de datos del sistema. Presenta también fuerte protección de datos, lo cual es muy importante debido a la sensibilidad de la información.

- Lenguaje de programación principal: PHP
 - Justificación: se optó por utilizar este lenguaje debido a que permite muchas facilidades al desarrollar, hay bastante documentación y también porque se cuenta con bastante experiencia utilizando este lenguaje.

- Framework para el desarrollo: Laravel
 - Justificación: se tomó la decisión de utilizar un framework para codificar de forma más rápida y menos repetitiva, además, también permite llevar a cabo de forma simple varias tareas, como la gestión de rutas, solicitudes, respuestas, vistas, controladores, programación de tareas y autenticación.

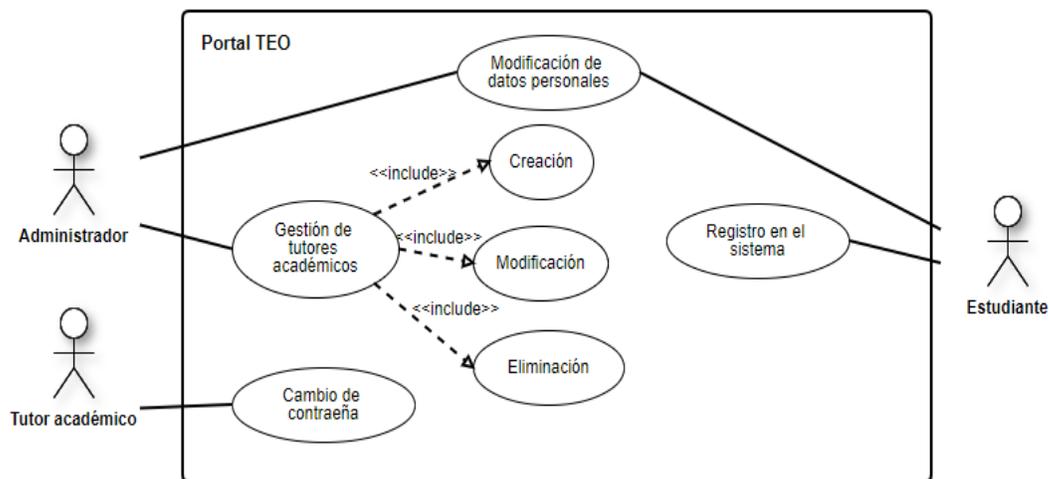
3.2.4. Vista funcional

Se ilustran por medio de casos de uso las diferentes operaciones que permite realizar el portal para cada actor en el sistema.

3.2.4.1. Sobre la gestión de usuarios

Las opciones que pueden realizar los diferentes tipos de usuario en el portal respecto a la gestión de datos de usuario, así como registro en el sistema, son los siguientes:

Figura 24. Caso de uso de gestión de usuarios

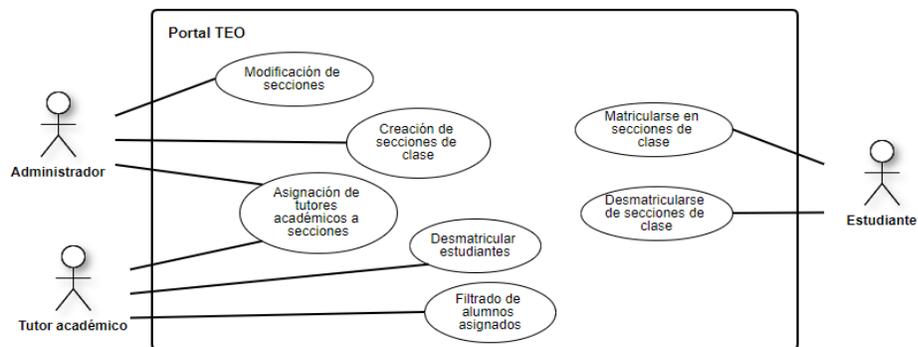


Fuente: elaboración propia.

3.2.4.2. Sobre las secciones de clase

Las operaciones que se pueden llevar a cabo respecto a las secciones de clase y la matriculación de los estudiantes en dichas secciones son las siguientes:

Figura 25. Caso de uso de secciones de clase

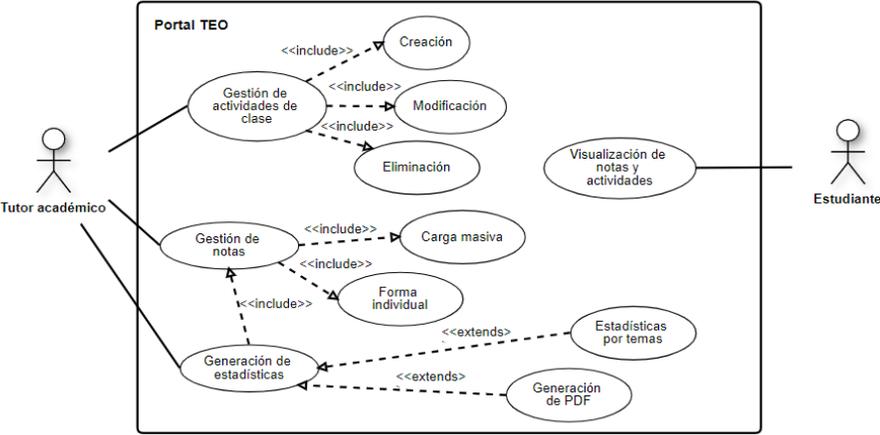


Fuente: elaboración propia.

3.2.4.3. Sobre la gestión de notas y actividades de clase

Los actores que participan en la gestión de notas y actividades de clase son solo los tutores académicos y estudiantes.

Figura 26. **Caso de uso de gestión de notas y actividades**

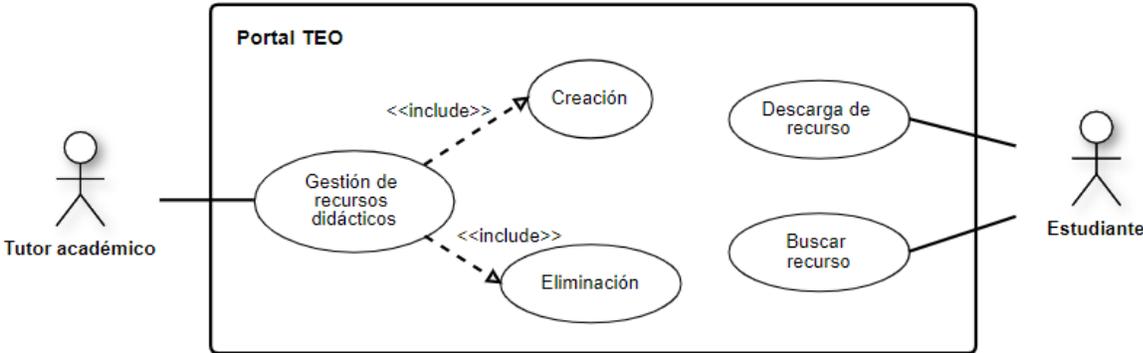


Fuente: elaboración propia.

3.2.4.4. Sobre la gestión de recursos didácticos

Las posibles opciones respecto al manejo de recursos didácticos son las siguientes:

Figura 27. **Caso de uso de gestión de recursos**

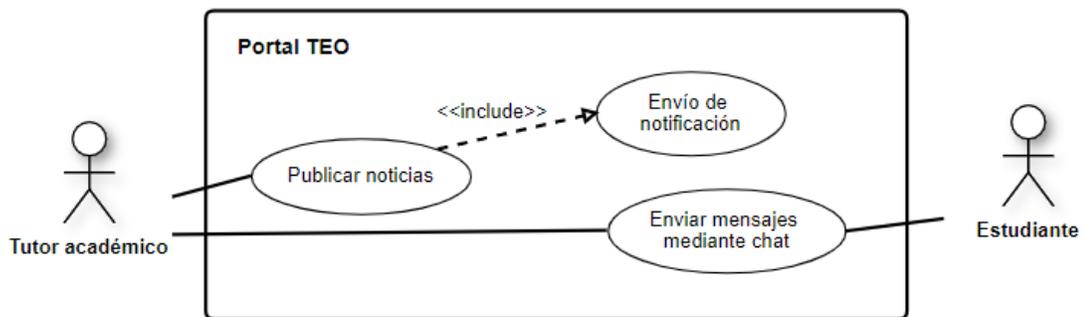


Fuente: elaboración propia.

3.2.4.5. Sobre la comunicación entre tutor académico y estudiante

La comunicación entre tutor académico y estudiante se da de la siguiente forma:

Figura 28. Caso de uso de comunicación entre tutor y estudiante



Fuente: elaboración propia.

3.2.5. Vista lógica

El tipo de arquitectura utilizado para el sistema desarrollado fue una arquitectura en capas, específicamente el modelo de 3 capas. La capa más inferior de modelo es la capa de Persistencia, la cual es la encargada de almacenar los datos. La siguiente es la capa de Negocio, la cual se encarga de gestionar la lógica de la aplicación, es decir, indicar qué se hace con los datos. La primera capa es la capa de Presentación, en la cual se maneja la interfaz con la que tiene contacto el usuario. Se tomó la decisión de utilizar este modelo debido a que realizar cambios en alguna de las capas es más sencillo, debido a que cada capa se comunica únicamente con la capa inmediatamente inferior (figura 3).

3.2.6. Estructura de la codificación del proyecto

Debido a que se decidió utilizar el *framework* Laravel para desarrollar el portal, el proyecto cuenta con la estructura que se describirá a continuación.

3.2.6.1. Controladores

El uso de controladores es un método que permite agrupar la lógica de las solicitudes HTTP que se generen desde la aplicación web y de esta forma se pueda estructurar mejor el código. Los controladores utilizados para manejar la lógica de la aplicación son los siguientes:

- **AdminController:** en este controlador se encuentra toda la lógica que utiliza el usuario Administrador para llevar a cabo sus tareas, como la creación de secciones de clase, el registro de tutores y su asignación correspondiente a las secciones de clase. Dentro de este controlador se encuentra una función que permite realizar una carga masiva de toda la información antes mencionada en el sistema.
- **AlumnoController:** en este controlador se localizan algunas de las funciones que le permiten al usuario Estudiante llevar a cabo sus tareas, como la visualización y descarga de recursos didácticos, así como la visualización de las actividades que tiene pendientes.
- **CalendarController:** este controlador contiene todos los métodos necesarios para manejar el calendario de actividades, como agregar actividades, eliminarlas y mostrar las ya programadas.
- **CursoController:** este controlador maneja la mayoría de la lógica respecto a las secciones de clase, en él se encuentran los métodos necesarios para poder gestionar las notas, los recursos didácticos y las actividades de cada clase.

- **EstadisticasController:** en este controlador se encuentran las funciones necesarias para la generación, descarga y envío de estadísticas. La mayoría de las funciones contienen consultas que permiten la generación de las estadísticas solicitadas por el departamento de matemática.
- **MailController:** en este controlador se encuentra parte de la lógica necesaria para permitir el envío de correos electrónicos para el restablecimiento de la contraseña.
- **MatricularController:** este contiene las funciones utilizadas para el proceso de matriculación de los estudiantes en las secciones de clase. Se realiza una serie de inserciones y verificaciones en la base de datos para llevar a cabo este proceso. También gestiona el proceso de desmatriculación.
- **NoticiasController:** contiene las funciones que permiten la publicación de noticias a los estudiantes, tanto el almacenado de las noticias y diálogos en la base de datos, como el envío de correos electrónicos con notificaciones a los estudiantes y tutores académicos.
- **PerfilController:** este controlador permite la visualización y modificación de los datos del usuario que se encuentra con sesión iniciada.
- **RegistroController:** este es el controlador que realiza las operaciones necesarias para que los estudiantes puedan registrarse en el sistema

3.2.6.2. Rutas

Laravel permite gestionar de forma sencilla las rutas a través de las cuales se puede acceder a las funciones descritas en los distintos controladores antes mencionados. Estas rutas son llamadas por medio de los diferentes formularios a la vez que se genera una interacción con el sistema por medio de los usuarios. Se encuentran descritas en un archivo llamado *web.php* dentro de la

carpeta *routes* del proyecto. Las rutas utilizadas para el portal cuentan con la siguiente estructura:

Figura 29. **Estructura de una ruta**

```
Route::post('/admin/cargamasiva', 'AdminController@cargamasiva')->name('admin.cargamasiva');
```

Fuente: elaboración propia.

En la anterior imagen la parte subrayada con color naranja indica el método HTTP por medio del cual se accederá al método del controlador que representa la ruta. En el proyecto la mayoría de las rutas utilizan los métodos *get* y *post*. La parte remarcada con color verde indica la ruta por medio de la cual se accederá al método del controlador, es decir, será la ruta que se mostrará en la barra de direcciones del navegador cada vez que el usuario interactúe con el portal. La parte remarcada con color amarillo indica el controlador y el método al cual se accederá por medio de la ruta; la primera parte indica el controlador y la segunda la función que se ejecutará. Por último, la parte remarcada con el color rojo indica el nombre de ruta. Dicho nombre puede ser utilizado en los formularios de las vistas para acceder a las rutas respectivas.

3.2.6.3. Vistas

Las vistas representan en el modelo de 3 capas la capa de presentación, es decir, es el componente de la aplicación con la que los usuarios tienen contacto directo. Dichas vistas en el *framework* Laravel son archivos PHP que contendrán en su mayoría código HTML, pero en este proyecto también

contienen código PHP que ayuda a completar la funcionalidad de la capa de presentación, añadiendo la lógica necesaria para mostrar la información que corresponde a cada tipo de usuario, según los datos que se encuentran almacenados en la capa de datos.

Las vistas que componen la aplicación se encuentran en el directorio */resources/views* del proyecto y se encuentran organizadas en subdirectorios dependiendo del fin que persiga dicha vista o bien, del usuario al cual se le mostrará.

Para estimular la lógica de la aplicación las vistas cuentan con ciertos elementos en el código. Uno de ellos es la forma de llamar a una ruta por medio de un formulario HTML (figura 30).

Figura 30. **Solicitud POST por medio de un formulario**

```
<form method="POST" action="{{ route('admin.cargamasiva') }}" enctype="multipart/form-data">
```

Fuente: elaboración propia.

Para llamar una función de un controlador, al formulario debe colocársele el atributo *action* seguido del comando entre comillas que sigue la siguiente estructura `{{ route('nombre de la ruta') }}`.

Por motivos de seguridad, Laravel necesita que las peticiones enviadas a través de formularios incluyan un *token* de verificación. Únicamente debe el agregarse el comando de la imagen siguiente dentro del formulario HTML, figura 31.

Figura 31. **Token de verificación**

```
{{ csrf_field() }}
```

Fuente: elaboración propia.

Laravel permite optimizar el proceso de codificación en las vistas, eliminando la necesidad de utilizar código repetitivo en estas, a través del uso de templates figura 32, heredándola en la vista a ser utilizada.

Figura 32. **Uso de template en una vista**

```
@extends('templates.main3')
```

Fuente: elaboración propia.

3.2.6.4. **Conexión con la base de datos**

Al utilizar el modelo de 3 capas se facilita la gestión de las conexiones con la base de datos, al utilizar el archivo de configuración que el *framework* posee en el directorio raíz del proyecto, llamado *env*. En dicho archivo es posible configurar la conexión con la base de datos, servidores de correo, entre otros.

Una vez configurada la conexión con la base de datos bastará con importar la clase denominada *DB* en cada uno de los controladores para realizar alguna operación con la base de datos.

3.2.6.5. Conexión con el servidor de correos

Por motivos de confiabilidad y simplicidad se optó por utilizar Gmail como servidor para enviar los diferentes correos con alertas en la aplicación. Al igual que la conexión con la base de datos, los parámetros para la configuración de la conexión con el servidor se colocan en el archivo `env`.

3.2.6.6. Calendarización de tareas y envío de correos electrónicos

Para llevar a cabo el proceso de enviar notificaciones a los usuarios por medio de correo electrónico cada vez que se encuentra próxima una entrega, se programó una tarea que se ejecuta diariamente verificando si hay una actividad a realizarse en el próximo día. Esto se llevó a cabo por medio de la utilización de lo que se conoce como *Cron Job*.

Cron Job es una herramienta de Linux que permite ejecutar un comando o una secuencia de comandos en una hora y fecha especificadas. Existen diversos servicios de *hosting* que permiten la calendarización de actividades, pero con el fin de consumir menos recursos se optó por utilizar *Cron Job*, ya que el servidor que se utiliza es un servidor Linux. Laravel facilita la conexión con *Cron Job* ya que es capaz de generar los comandos necesarios para que puedan ser ejecutados por Linux, figura 33.

Figura 33. Configuración Cron Job

```
1 | php artisan make:console MostrarMensaje
```

Fuente: elaboración propia.

En la figura 33, *MostrarMensaje* es el nombre del archivo que se generará. Este podrá encontrarse en el directorio *app/Console/Commands*. El archivo generado contará ya con código que facilitará la creación de la tarea a programar, figura 34.

Figura 34. **Tarea Cron Job**

```
protected $signature = 'enviar:correos';
```

Fuente: elaboración propia.

En el caso de la figura 34, *enviar:correos* es el nombre del comando que generará Laravel para que pueda ser ejecutado por un *Cron Job*. El archivo cuenta con una función predefinida llamada *handle*, en ella deberá colocarse el código de la tarea a programar. Para este caso en particular se utilizó el código de la figura 35.

Figura 35. Código de tarea programada

```
$consulta = "SELECT A.Fecha Fecha, A.Nombre as Nombreactividad, A.Ponderacion as
Ponderacion, E.Nombre as Nombrecurso, C.Nombre as Nombrealumno, C.email as Email
FROM Actividad A
INNER JOIN Alumno_Actividad B ON A.id_actividad = B.id_actividad
INNER JOIN users C ON B.id = C.id
INNER JOIN Clase D ON D.id_clase = A.id_clase
INNER JOIN Curso E ON E.id_curso = D.id_curso
WHERE ADDDATE(CURDATE(), 1) = Fecha AND id_actividad_padre IS NULL;";

$res = DB::select($consulta);

foreach ($res as $key => $item) {
    $to_name = $item->Nombrealumno;
    $to_email = $item->Email;
    $to_curso = $item->Nombrecurso;
    $data = array('Fecha' => $item->Fecha, 'Nombreactividad' => $item->Nombreactividad, '
    Ponderacion' => $item->Ponderacion, 'Nombrecurso' => $item->Nombrecurso, '
    Nombrealumno' => $item->Nombrealumno);
    Session::put('data',$data);
    Mail::send('email', $data, function($message) use ($to_name, $to_email, $to_curso) {
        $message->to($to_email, $to_name)
            ->subject('Recordatorio de actividad del curso de ' . $to_curso);
        $message->from('teo.mate.usac@gmail.com', 'Departamento de Matemática');
    });
}
```

Fuente: elaboración propia.

Puede observarse de la figura 35 que se realiza una consulta para conocer qué actividades están próximas a entregarse en cada una de las secciones de clase registradas en el portal, se extraen datos como el nombre del alumno, el nombre del curso y el nombre de la actividad para poder enviar un correo más personalizado en cada caso. Una vez realizada la consulta, por cada registro retornado de la base de datos se envía un correo. Para ello se utiliza la sentencia de control de flujo *foreach*. Dentro de esta última se puede observar el código necesario para el envío de un correo electrónico, siempre utilizando la configuración indicada para el servidor de correos electrónicos.

Básicamente lo que se envía en el correo electrónico es una vista, es decir código HTML. La vista que se envía se indica como parámetro en el comando *Mail::send*. En este caso la vista que se envía es la vista *email*. Para hacer el correo más personalizado se le puede enviar información a dicha vista por medio de un arreglo. Un detalle importante es que para hacer uso de la clase *Mail*, esta debe importarse al inicio del archivo en cuestión, figura 36.

Figura 36. **Importación de librería**

```
use Illuminate\Support\Facades\Mail;
```

Fuente: elaboración propia.

Una vez colocado el código necesario para la tarea a programarse, esta deberá añadirse al archivo *Kernel.php* que se encuentra en el directorio */app/Console*. Puede observarse que este archivo cuenta con cierta estructura. Deberá ubicarse la función llamada *schedule* y añadirse el comando de la figura 37.

Figura 37. **Schedule Cron Job**

```
protected function schedule(Schedule $schedule)
{
    // $schedule->command('enviar:correos')
    //                 ->everyMinute();

    $schedule->command('enviar:correos')
              ->dailyAt('06:00');
}
```

Fuente: elaboración propia

El comando a programarse será el creado anteriormente: *enviar:correos* y en este caso en particular se programará para ejecutarse diariamente a las 6:00 horas. Esto puede variar dependiendo de las necesidades ya que puede ejecutarse en intervalos de tiempo más cortos o bien programarse para una fecha en específico.

3.2.6.7. Generación de estadísticas y descarga de archivos PDF

El portal permite la generación de estadísticas de examen una vez ingresadas las notas de la evaluación correspondiente. Para la comodidad del usuario, estas estadísticas pueden ser descargadas en un archivo PDF para su fácil impresión y entrega.

El proyecto ya cuenta con el módulo de PHP denominado *DOMPDF*, el cual permite la generación de los documentos. El funcionamiento es bastante similar al de los correos electrónicos: se tiene una vista utilizada para generar los componentes que pueden observarse en el documento, y para que contengan la información computada por la capa lógica se le puede enviar información a la vista para así poder visualizar los datos resultantes del proceso estadístico.

Figura 38. Código de estadísticas

```
$sreprobados = "SELECT COUNT(*) as nreprobados FROM Alumno_Actividad WHERE Nota<60.5
AND Nota>=0 AND id_actividad=".$examen."";
$reprobados = DB::select($sreprobados);

$porcrepro = 0;
try{
    if($examinados[0]->nexaminados>0){
        $porcrepro = $reprobados[0]->nreprobados/$examinados[0]->nexaminados*100;
    }
}catch(Exception $e){
    $porcrepro = 0;
}

$sausentes = "SELECT COUNT(*) as nausentes FROM Alumno_Actividad WHERE Nota<0 AND
id_actividad=".$examen."";
$ausentes = DB::select($sausentes);

$porcausentes = 0;
try{
    if($inscritos[0]->ninscritos>0){
        $porcausentes = $ausentes[0]->nausentes/$inscritos[0]->ninscritos*100;
    }
}catch(Exception $e){
    $porcausentes = 0;
}

$scero = "SELECT COUNT(*) as ncero FROM Alumno_Actividad WHERE Nota=0 AND
id_actividad=".$examen."";
$cero = DB::select($scero);

$spromedio = "SELECT AVG(Nota) as npromedio FROM Alumno_Actividad WHERE Nota>=0 AND
id_actividad=".$examen."";
$promedio = DB::select($spromedio);

$sdesv = "SELECT STD(Nota) as ndesv FROM Alumno_Actividad WHERE Nota>=0 AND
id_actividad=".$examen."";
$desv = DB::select($sdesv);
```

Fuente: elaboración propia.

En el código mostrado en la figura 38 puede observarse una serie de consultas y cálculos. Se pueden observar los siguientes elementos:

- El segmento resaltado con color rojo se utiliza para calcular el número de estudiantes que reprobaron el examen, es decir, que poseen una nota inferior a los 61 puntos necesarios para aprobar la evaluación.
- El segmento resaltado con color naranja se utiliza para calcular el porcentaje de estudiantes examinados que reprobaron.
- La consulta remarcada con color amarillo se utiliza para calcular el número de estudiantes que no asistieron a la evaluación. Puede observarse que en la cláusula *WHERE* se verifica que el campo *Nota* sea menor a 0. Esto debido a que cuando se crea una evaluación en la base de datos se le coloca una nota de *-0,01* a todos los estudiantes. Esto se llevó a cabo para hacer una diferenciación de los estudiantes que se examinaron, pero que obtuvieron una nota de 0 puntos, de los que no se examinaron.
- En el segmento remarcado con color verde se hace el cálculo del porcentaje de estudiantes que se ausentaron de la evaluación.
- La consulta remarcada con color celeste se utiliza para obtener el número de estudiantes que obtuvieron nota de 0 puntos en la evaluación.
- En el segmento remarcado con color púrpura se calcula la nota promedio de los estudiantes en la evaluación. Para ello se utiliza la función de *SQL* llamada *AVG*.
- Por último, en el segmento remarcado con color rosa se utiliza una consulta para calcular la desviación estándar de las notas. Para ello se utiliza la función *STD*.

Para generar el documento PDF se utiliza la función de la librería *DOMPDF* llamada *loadView*. Dicha función recibe dos parámetros: el primero indica la adaptación de la vista en el documento; en este caso es la vista PDF que se encuentra en el directorio *Reportes*. El segundo parámetro son los datos

que serán mostrados en el documento, es decir, los números calculados en el proceso estadístico.

4. EVALUACIÓN DEL SISTEMA

La evaluación del sistema se llevó a cabo por medio de una encuesta realizada a los tutores académicos del Departamento de Matemática para conocer cómo fue la experiencia que tuvieron con el portal.

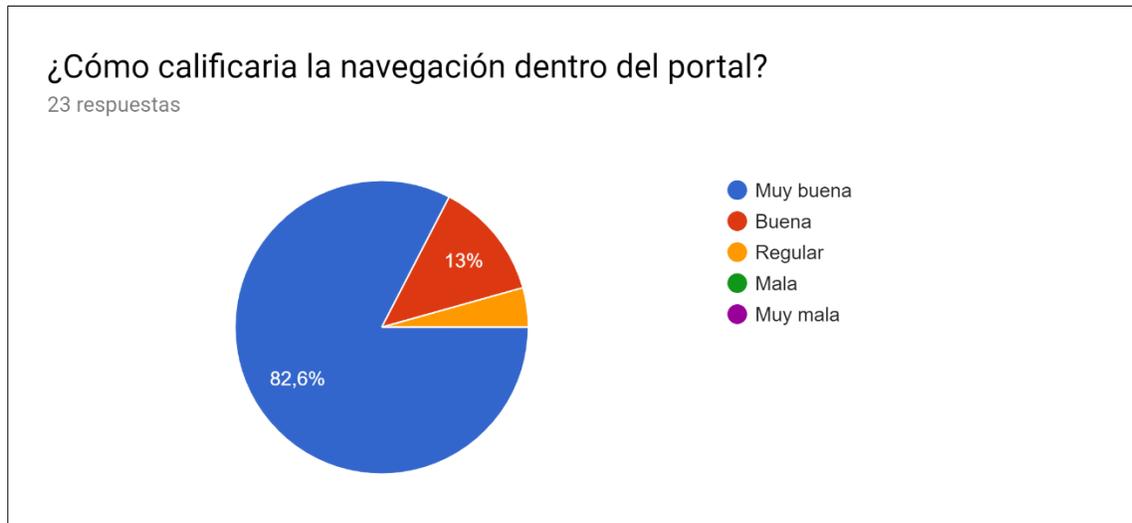
Para la realización de dicha encuesta se les brindó a los tutores académicos una pequeña inducción sobre cómo utilizar el portal, dándoles a conocer cuáles son las herramientas y las acciones que pueden llevarse a cabo por medio de este. Se les envió también por medios electrónicos un manual de usuario con explicaciones detalladas para respaldar los conocimientos aprendidos.

Se deseaba tener una muestra de tamaño 30, sin embargo, por cuestiones de disponibilidad de tiempo por parte de los tutores académicos se llenaron solamente 23 encuestas.

4.1. Análisis de resultados obtenidos

A continuación, se presenta el análisis de los resultados obtenidos para cada una de las preguntas realizadas en la encuesta.

Figura 39. **Navegación dentro del portal**



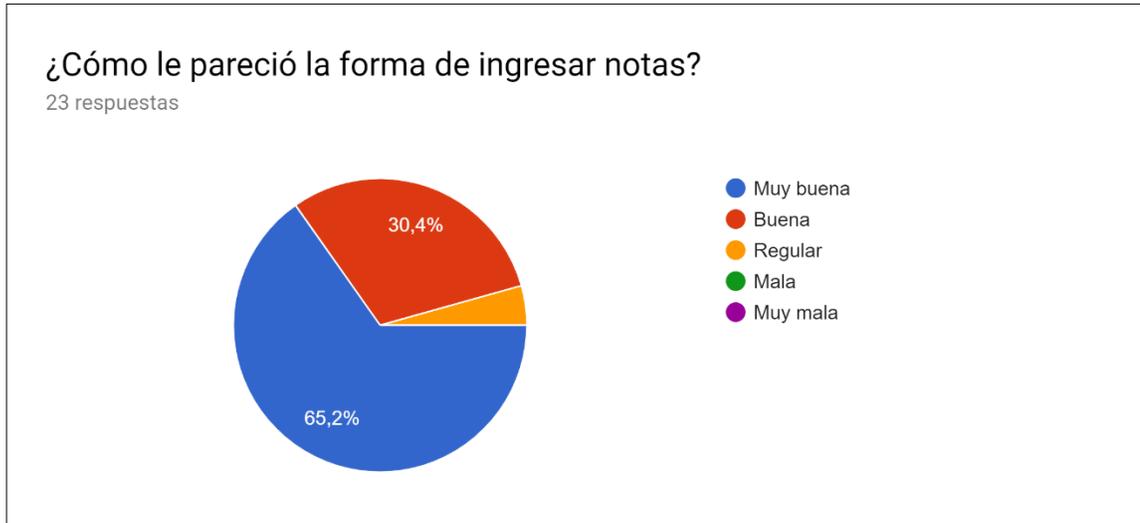
Fuente: Google Forms. *Calificación de usuarios respecto a la navegación del portal.*

[https://docs.google.com/forms/d/1nNcO5zd-4yn-](https://docs.google.com/forms/d/1nNcO5zd-4yn-Gqv6wJdJ3i5OZvShwWIWUHrtygGorHw/edit#responses)

[Gqv6wJdJ3i5OZvShwWIWUHrtygGorHw/edit#responses](https://docs.google.com/forms/d/1nNcO5zd-4yn-Gqv6wJdJ3i5OZvShwWIWUHrtygGorHw/edit#responses). Consulta: mayo de 2019.

En la figura 39 se muestra un gráfico que describe cómo fue la experiencia de los tutores académicos utilizando el Portal. En general, la encuesta refleja que para los usuarios fue bastante cómodo y entendible la utilización de las herramientas. Un comentario recibido por uno de los tutores fue que el entendimiento de algunas herramientas se facilitaría por medio de inducciones cada vez que entren nuevos tutores académicos, lo cual es bastante acertado.

Figura 40. **Calificación de la forma de ingresar notas**



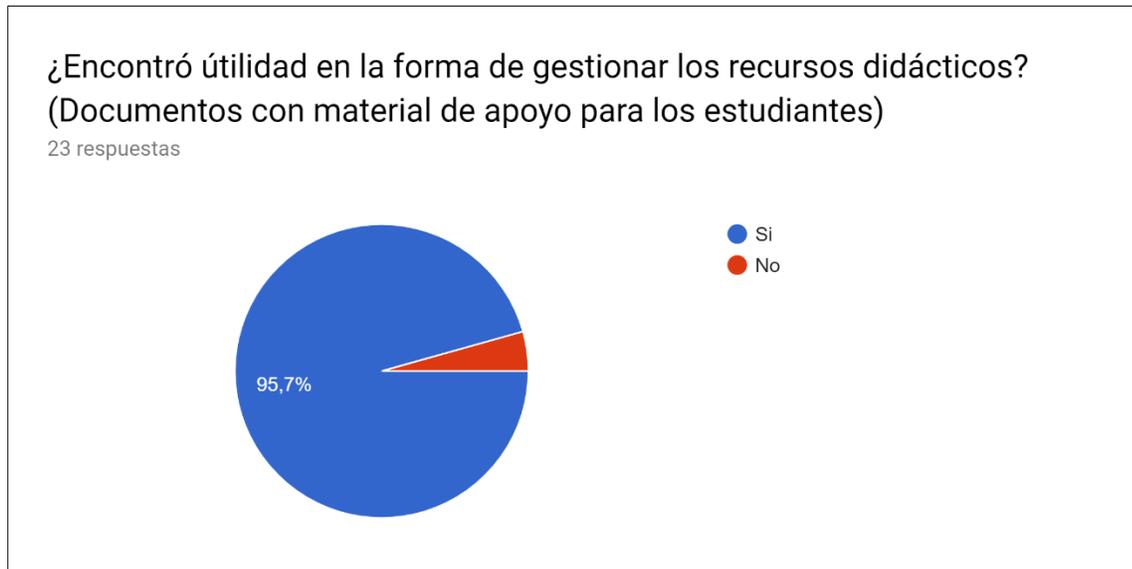
Fuente: Google Forms. *Calificación de los usuarios respecto al ingreso de notas.*

[https://docs.google.com/forms/d/1nNcO5zd-4yn-](https://docs.google.com/forms/d/1nNcO5zd-4yn-Gqv6wJdJ3i5OZvShwWIWUHrtygGorHw/edit#responses)

[Gqv6wJdJ3i5OZvShwWIWUHrtygGorHw/edit#responses](https://docs.google.com/forms/d/1nNcO5zd-4yn-Gqv6wJdJ3i5OZvShwWIWUHrtygGorHw/edit#responses). Consulta: mayo de 2019.

Uno de los problemas que debe superar un nuevo sistema es la resistencia al cambio por medio de los usuarios. En este caso los tutores académicos tienen la costumbre de gestionar las notas únicamente por medio del software Excel, es por ello por lo que al desarrollar el sistema se le dio una apariencia similar al cuadro donde suelen manejar las notas. Esta resistencia al cambio suele ser temporal y disminuye conforme se va utilizando el sistema y se va ganando experiencia en el uso de este.

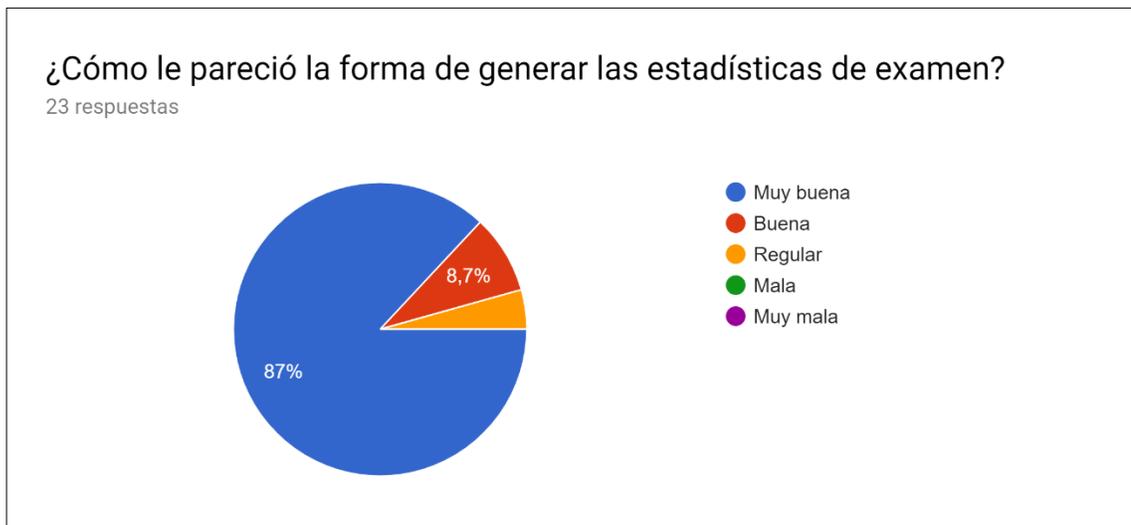
Figura 41. **Utilidad en los recursos didácticos**



Fuente: Google Forms. *Calificación de los usuarios respecto a los recursos didácticos.*
<https://docs.google.com/forms/d/1nNcO5zd-4yn-Gqv6wJdJ3i5OZvShwWIWUHrtygGorHw/edit#responses>. Consulta: mayo de 2019.

Uno de los fines principales con los que fue desarrollado el sistema fue el de servir como herramienta de apoyo en las actividades académicas, facilitando la interacción entre los tutores académicos y los estudiantes. El permitir que todos los documentos de una sección de clase puedan ser accedidos en un solo lugar y que se encuentren ordenados cronológicamente les facilitará a los estudiantes encontrar el material que necesiten para comprender mejor la materia. Es claro también que se debe tener cierto nivel de compromiso por parte de los tutores académicos para que puedan brindarle a sus estudiantes periódicamente material que sea de apoyo para ellos.

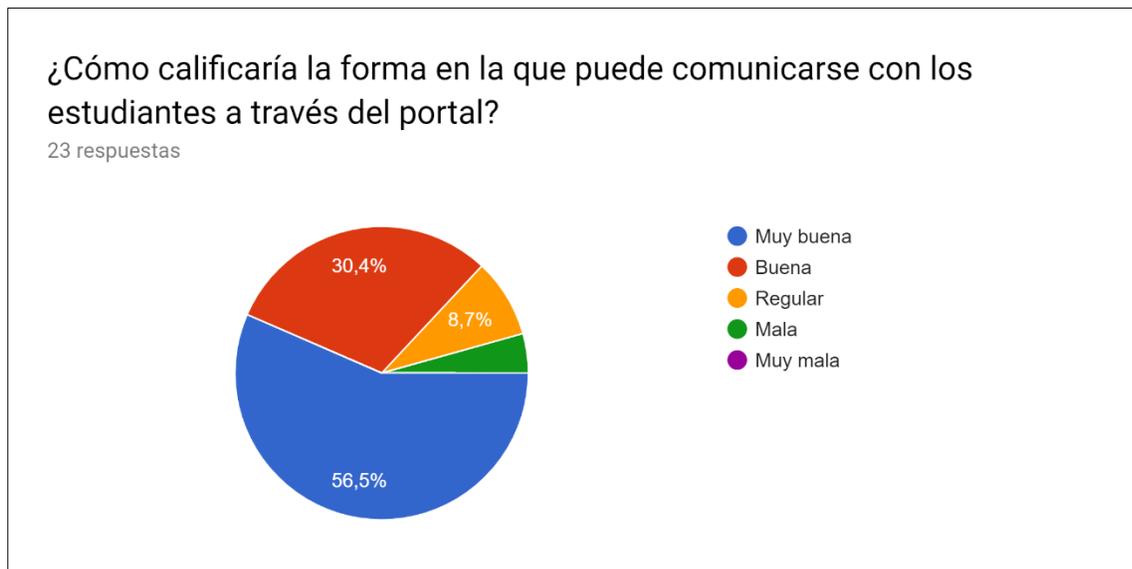
Figura 42. **Forma de generar estadísticas de exámenes**



Fuente: Google Forms. *Calificación de los usuarios respecto a la generación de estadísticas.*
<https://docs.google.com/forms/d/1nNcO5zd-4yn-Gqv6wJdJ3i5OZvShwWIWUHrtygGorHw/edit#responses>. Consulta: mayo de 2019.

Otro de los objetivos con que fue desarrollado el sistema fue el de facilitar el trabajo de los tutores académicos. Después de explicarle a los tutores la forma en la que pueden realizar esta labor y de que vieran el resultado obtenido se tuvo una aceptación bastante alta, un 87 % de los encuestados calificó la herramienta de *Muy buena* y un 8,7 % la calificó de *Buena*.

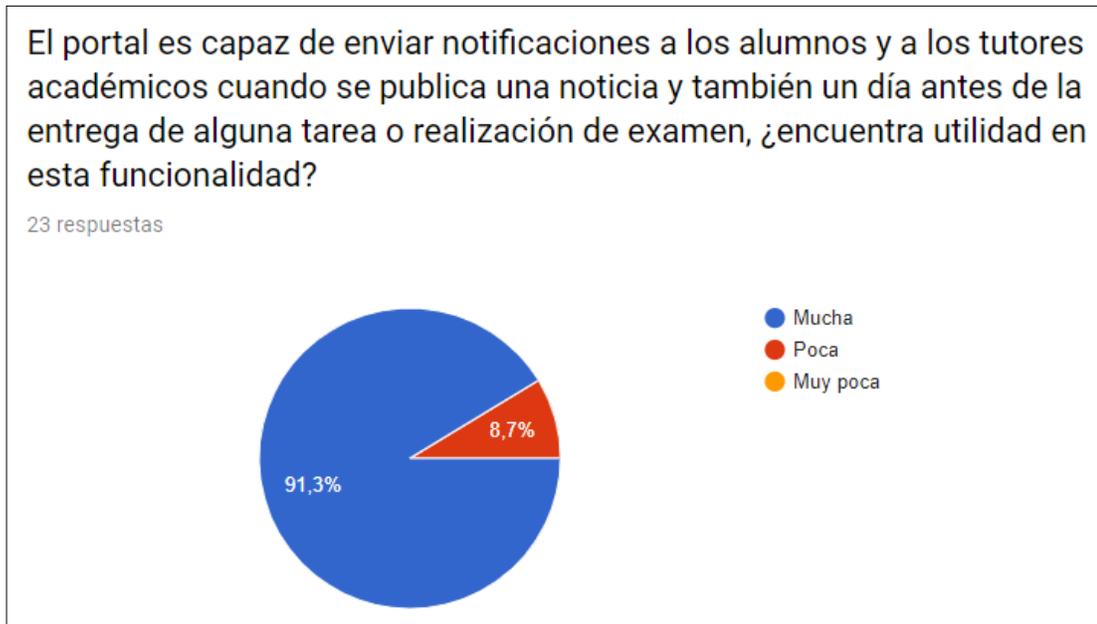
Figura 43. **Comunicación entre tutor académico y estudiante**



Fuente: Google Forms. *Calificación de los usuarios respecto a la comunicación con los estudiantes.* <https://docs.google.com/forms/d/1nNcO5zd-4yn-Gqv6wJdJ3i5OZvShwWUWUHrtygGorHw/edit#responses>. Consulta: mayo de 2019.

La comunicación entre el tutor académico y el estudiante es muy importante en algunos casos, tanto para resolución de dudas como para solventar algún problema que ocurra con el ingreso de notas. Es por lo que se decidió también proporcionar una herramienta que facilitara dicho proceso. Puede observarse que no se tuvo tanta aceptación como con las demás herramientas. Se recibió un comentario de un tutor académico indicando que no le parecía que no se permitiera la comunicación en privado, pero se llevó a cabo de esta forma ya que evitando estos contactos directos se ayuda a brindar transparencia al proceso de evaluación.

Figura 44. Envío de notificaciones



Fuente: Google Forms. *Calificación de los usuarios respecto a la utilidad de las notificaciones.*

[https://docs.google.com/forms/d/1nNcO5zd-4yn-](https://docs.google.com/forms/d/1nNcO5zd-4yn-Gqv6wJdJ3i5OZvShwWIWUHrtygGorHw/edit#responses)

[Gqv6wJdJ3i5OZvShwWIWUHrtygGorHw/edit#responses](https://docs.google.com/forms/d/1nNcO5zd-4yn-Gqv6wJdJ3i5OZvShwWIWUHrtygGorHw/edit#responses). Consulta: mayo de 2019.

En ocasiones, se dan incidentes en los que los estudiantes no entregan alguna tarea o hasta dejan de asistir a una evaluación porque olvidaron que tenían dicho compromiso. También es importante que los tutores académicos tengan presente cuáles son las fechas en las que se realizarán las evaluaciones ya que en algunos casos debe reservarse salones para la realización de estos. Los tutores encontraron bastante utilidad en el sistema de notificaciones, lo cual se refleja en los resultados de la encuesta, que indica que el 91 % encuentra utilidad en esta funcionalidad.

CONCLUSIONES

1. Se implementó una página web donde se provee a los auxiliares del Departamento de Matemática de Facultad de Ingeniería de la Universidad de San Carlos, herramientas para gestionar las notas, recursos y a los usuarios. Mediante una encuesta se comprobó que el 95,6 % de los usuarios consideran que la gestión de notas es mejor que la tienen actualmente y 95,7 % considera que es de utilidad la gestión de los recursos.
2. Se automatizó la generación de estadísticas para los exámenes y envío por medio de correo electrónico a los jefes de área, por lo cual el 95,7 % de los auxiliares consideran útil este proceso.
3. Se implementó una sección de noticias para que los auxiliares puedan comunicarse con los usuarios, la notificación de esta es por correo electrónico, 86,9 % de los auxiliares califica como buena o muy buena la forma de comunicarse con los alumnos.
4. Las notificaciones por exámenes, tareas y actividades próximas se implementaron vía correo electrónico, al igual que las noticias, por lo que el 91,3 % de los auxiliares considera de ayuda este sistema.

RECOMENDACIONES

1. El encargado de administrar el sistema en el Departamento de Matemática tendrá que dar mantenimiento al sistema antes de iniciar cada ciclo. Tendrá que crear usuarios a los nuevos tutores académicos, asignarles sus secciones de clase e ingresar los horarios correspondientes a cada semestre.
2. Para mantener la seguridad del sistema, se sugiere dar las credenciales del usuario administrador a una sola persona.
3. Dar breves inducciones sobre el sistema a los estudiantes que se incorporan a la Facultad de Ingeniería permitirá que las actividades se desarrollen sin complicaciones.
4. Se sugiere realizar copias de seguridad de la base de datos constantemente para asegurar la persistencia de la información.

BIBLIOGRAFÍA

1. BARBA, Juan. *Diseño y desarrollo web: análisis de casos*. Facultad de Belles Arts de Sant Carles, 2014. 38 p.
2. GARCÍA, Ivan. *Comparación de las tecnologías .Net y J2EE para el desarrollo de servicios web*. Universidad de San Carlos de Guatemala, 2007. 153 p.
3. *IBM KNOWLEDGE CENTER. Clientes web*. [en línea]. <https://www.ibm.com/support/knowledgecenter/es/SS5MD2_7.4.0.1/com.ibm.itcamt.doc/rt/Workspaces/T5_WRT/wksp_t5_clientsweb.html>. [Consulta: marzo de 2019].
4. RABIN, Jo. *Mobile web best practices 1.0, W3C, 2008*. [en línea]. <<https://www.w3.org/TR/mobile-bp/#d0e128>>. [Consulta: marzo de 2019].
5. REYES, Pablo. *Sistema para la gestión de notas del departamento de matemática de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala*. Trabajo de graduación de Ing. Ciencias y Sistemas, Facultad de Ingeniería, Universidad de San Carlos de Guatemala, 2007. 117 P.

6. RUIZ, José. *Diseño de aplicaciones web distribuidas en N capas utilizando patrones*. Trabajo de graduación de Ing. Ciencias y Sistemas, Facultad de Ingeniería, Universidad de San Carlos de Guatemala, 2006. 163 P.