

**UNIVERSIDAD DE EL SALVADOR  
FACULTAD MULTIDISCIPLINARIA ORIENTAL  
DEPARTAMENTO DE INGENIERIA Y ARQUITECTURA  
SECCIÓN DE INGENIERÍA DE SISTEMAS INFORMÁTICOS**



**INFORME FINAL DE TRABAJO DE GRADO:  
MODALIDAD PASANTÍA DE PRÁCTICA PROFESIONAL**

**TÍTULO DEL PLAN:**

**DESARROLLADOR BACKEND DEL SISTEMA DE PLANIFICACIÓN DE CARGA  
ACADÉMICA EN EL DEPARTAMENTO DE MEDICINA.**

**PARA OPTAR AL GRADO ACADÉMICO DE:**

**INGENIERÍA DE SISTEMAS INFORMÁTICOS**

**PRESENTADO POR:**

**DANYS EMANUEL GUEVARA ARRIAZA N.º CARNET GA19014**

**DOCENTE ASESOR:**

**ING. HÉCTOR JAVIER PAIZ RAMOS**

**NOVIEMBRE DE 2025**

**SAN MIGUEL, EL SALVADOR CENTROAMÉRICA**

**UNIVERSIDAD DE EL SALVADOR**  
**AUTORIDADES**



**MSC. JUAN ROSA QUINTANILLA**  
**RECTOR**

**DRA. EVELYN BEATRIZ FARFÁN**  
**VICERRECTORA ACADÉMICA**

**MSC. ROGER ARMANDO ARIAS ALVARADO**  
**VICERRECTOR ADMINISTRATIVO**

**LIC. PEDRO ROSALÍO ESCOBAR CASTANEDA**  
**SECRETARIO GENERAL**

**LIC. CARLOS AMILCAR SERRANO RIVERA**  
**FISCAL GENERAL**

**UNIVERSIDAD DE EL SALVADOR  
FACULTAD MULTIDISCIPLINARIA ORIENTAL  
AUTORIDADES**



**MSC. CARLOS IVÁN HERNÁNDEZ FRANCO  
DECANO**

**DRA. NORMA AZUCENA FLORES RETANA  
VICEDECANA**

**LIC. CARLOS JESÚS SÁNCHEZ  
SECRETARIO**

**ING. JOSÉ LUIS CASTRO CORDERO  
JEFE DEL DEPARTAMENTO DE INGENIERÍA Y ARQUITECTURA**

**ING. MILAGRO ALICIA GONZÁLEZ DE REYES  
COORDINADORA DE LOS PROCESOS DE GRADO DE LA CARRERA DE INGENIERÍA  
DE SISTEMAS INFORMÁTICOS**

## Índice

Resumen.....	6
Introducción.....	8
Información de La Institución .....	9
Datos generales.....	9
Localización .....	9
Antecedentes .....	9
Recursos .....	11
Instalaciones y equipos .....	11
<i>Mobiliario</i> .....	11
Tabla 1 .....	11
<i>Equipo</i> .....	11
Tabla 2 .....	11
<i>Humanos</i> .....	12
Tabla 3 .....	12
Actividades Actuales .....	12
Producción Principal y Otras .....	12
Situación Técnica .....	12
Situación Administrativa.....	13
Metodología .....	13
Tabla 4 .....	14
Resultados y Discusión .....	15
Fase 1: Construcción de la Lógica Fundamental y Arquitectura Backend .....	15
Tabla 5 .....	15
Figura 1 .....	18
Figura 2 .....	19

Figura 3 .....	20
Figura 4 .....	22
Figura 5 .....	22
Fase 2: Optimización, Desarrollo de Lógicas Complejas y Refactorización .....	23
Tabla 6 .....	23
Figura 6 .....	26
Figura 7 .....	26
Fase 3: Integración de Servicios, Seguridad y Consolidación del Sistema .....	29
Tabla 7 .....	29
Figura 8 .....	32
Figura 9 .....	32
Figura 10 .....	34
Figura 11 .....	35
Figura 12 .....	35
Conclusiones.....	37
Recomendaciones.....	38
Referencias .....	39
Visto Bueno del Tutor de la Institución .....	40
Anexos .....	41

## Resumen

El presente informe documenta la implementación del Backend del Sistema de Planificación de Carga Académica para el Departamento de Medicina de la Facultad Multidisciplinaria Oriental (FMO). Este sistema representa un componente fundamental en la estrategia de modernización de la gestión universitaria. Su relevancia reside en establecer un mecanismo funcional para organizar la distribución de la carga académica y optimizar los recursos administrativos de la facultad.

El desarrollo de la arquitectura del lado del servidor se abordó mediante un proceso estructurado enfocado en la integridad y funcionalidad de los datos. Este enfoque se tradujo en la creación de un sistema de gestión (CRUD) con validaciones rigurosas y en el diseño de un esquema de roles y permisos. Además, se integraron funcionalidades específicas para la planificación, como la gestión de grupos comunes y la funcionalidad de numeración automática, elementos esenciales para la precisión en la organización académica.

Este trabajo busca explorar el valor de una infraestructura tecnológica estable para el soporte de procesos académicos críticos. El proyecto demostró la capacidad del *framework* Laravel para el desarrollo de soluciones a nivel institucional y permitió consolidar habilidades avanzadas en la creación de aplicaciones web. El sistema final entregado es el soporte tecnológico que facilita las operaciones de planificación, siendo un elemento clave para la gestión administrativa de la FMO.

Palabras Clave: Desarrollo Backend; Laravel; Sistema de Carga Académica; Planificación Universitaria; Seguridad Web; Sistemas de Información.

## **Abstract**

This professional internship report addresses the implementation of the Backend of the Academic Load Planning System for the Medicine Department of the Oriental Multidisciplinary Faculty (FMO). This system represents a fundamental component in the strategy for modernizing university management. Its relevance lies in establishing a functional mechanism for organizing the distribution of the academic load and optimizing the faculty's administrative resources.

The development of the server-side architecture was approached through a structured process focused on data integrity and functionality. This approach resulted in the creation of a management system (CRUD) with rigorous validations and the design of a roles and permissions scheme. Furthermore, specific functionalities for planning were integrated, such as common group management and the automatic numbering functionality, which are essential for precision in academic organization.

This work seeks to explore the value of a stable technological infrastructure for supporting critical academic processes. The project demonstrated the capability of the Laravel framework for developing institutional-level solutions and allowed for the consolidation of advanced skills in web application creation. The final system delivered is the technological support that facilitates planning operations, making it a key element for the FMO's administrative management.

**Keywords:** Backend Development; Laravel; Academic Load System; University Planning; Web Security; Information Systems.

## Introducción

Este documento presenta un recuento detallado del trabajo realizado y las competencias desarrolladas durante la pasantía profesional en la Unidad de Tecnología de la Información (UTI) de la Facultad Multidisciplinaria Oriental. Como proyecto de culminación de la carrera de Ingeniería de Sistemas Informáticos, esta experiencia se enfocó en el desarrollo del Sistema de Planificación de Carga Académica, una iniciativa de gran valor estratégico para la institución.

La relevancia de esta pasantía se encuentra en la oportunidad de aplicar los fundamentos de la ingeniería de software a un desafío real. La labor como desarrollador Backend se centró en la construcción de la espina dorsal del sistema para el módulo de Medicina: la lógica, la seguridad y la gestión de datos que operan detrás de la interfaz de usuario. El propósito fue crear una base tecnológica confiable que permitiera optimizar los procesos administrativos y académicos del departamento.

El objetivo general fue, por tanto, Contribuir al desarrollo e implementación del Sistema de Planificación de Carga Académica para el módulo de medicina, Los objetivos específicos incluyeron la integración con bases de datos, la optimización de procesos, la realización de pruebas técnicas A continuación, se expone la metodología, los logros y las habilidades adquiridas, en correspondencia con el plan de trabajo original.

## Información de La Institución

### Datos generales

#### **Localización**

Las prácticas profesionales se llevaron a cabo en la Unidad de Tecnología de la Información (UTI), perteneciente a la Facultad Multidisciplinaria Oriental (FMO), localizada en Cantón el Jute, a 6.5 kms al Sur Oriente de la ciudad de San Miguel.

(Wikipedia)

#### **Antecedentes**

El 17 de junio de 1966, en Sesión No. 304, el Consejo Superior Universitario fundó el Centro Universitario de Oriente (CUO), en la Ciudad de San Miguel, como una extensión de los estudios universitarios de la Universidad Nacional de El Salvador hacia la Zona Oriental. Los objetivos principales que motivaron su creación fueron:

- Ampliar la capacidad de servicio docente de la Unidad.
- Satisfacer las necesidades educativas y culturales de la Zona Oriental.
- Contribuir con el desarrollo y progreso.
- Crear los instrumentos técnicos y culturales para sectores de la población que no tienen acceso a la Educación Universitaria.
- Descentralizar los servicios de Educación Superior.

Estos objetivos aún siguen vigentes. En abril de 1967, se adquirió un terreno de 108 manzanas de extensión en el Cantón el Jute, a 6.5 kms al Sur Oriente de la Ciudad de San Miguel, donde se pretendía construir el Campus Universitario del CUO. Las actividades académicas se iniciaron el 17 de mayo de 1969.

Por lo tanto, hasta el día de hoy, se cumplen 56 años de funcionamiento académico (desde 1969 hasta 2025) y 59 años desde su creación (desde el 17 de junio de 1966 hasta el 17 de junio de 2025)." (Universidad de El Salvador FMO Historia, Facultad Multidisciplinaria Oriental. (s. f.), 2025)

En este contexto, la Unidad de Tecnología de la Información (UTI) nació en 2024 como una iniciativa estratégica para la digitalización de la facultad. Su misión se centra en la modernización de operaciones críticas, como la administración académica y el seguimiento estudiantil, alineándose con la visión de la UES de ser un líder en la educación superior innovadora y sostenible.

- La Misión de la UES-FMO es: “Ser la institución de educación superior que forma profesionales con capacidad académica e intelectual transfiriendo conocimientos científico y técnico, para la formulación de alternativas de solución a los problemas que obstaculizan el desarrollo de la región oriental con una perspectiva integral, promoviendo la sostenibilidad y la armonía con el medio ambiente y la identidad cultural”. (Universidad de El Salvador FMO Mision, Facultad Multidisciplinaria Oriental. (s. f.), 2025)
- Su Visión es: “Ser una universidad transformadora de la educación superior y desempeñar un papel protagónico relevante, en la transformación de la conciencia crítica y prepositiva de la sociedad salvadoreña y en especial de la Zona Oriental del País, con liderazgo en la innovación educativa y excelencia académica, a través de la integración de las funciones básicas de docencia la investigación y la proyección social”. (Universidad de El Salvador FMO Vision, Facultad Multidisciplinaria Oriental. (s. f.), 2025)

## Recursos

### Instalaciones y equipos

#### **Mobiliario**

**Tabla 1**

*Mobiliario Asignado en la Unidad de Tecnología*

<b>Mobiliario</b>	<b>Cantidad</b>	<b>Descripción</b>
Escritorio	1	Proporcionado por la unidad de tecnología de la información.
Silla	1	Proporcionado por la unidad de tecnología de la información.

*Nota: El espacio de trabajo y el mobiliario que se detallan fueron facilitados por la institución anfitriona durante el período de la pasantía.*

#### **Equipo**

**Tabla 2**

*Equipo de Cómputo Utilizado en el Desarrollo*

<b>Equipo</b>	<b>Cantidad</b>	<b>Descripción</b>
Modelo: Razer Blade Memoria: 16 GIB Procesador: Intel(R) Core (TM) i7-8750H x 12 Gráficos: Intel(R) UHD Graphics 630 (CFL GT2) Gráficos 1 NVIDIA GeForce GTX 1060 with Max-Q Desing Capacidad del disco: 2.0 TB	1	Equipo personal

*Nota: Las especificaciones del computador personal empleado para el desarrollo del proyecto se detallan en la tabla anterior.*

## Humanos

**Tabla 3**

*Recursos Humanos y Equipo de Asesoría*

Personal	Cantidad	Descripción
Docente Asesor	1	Realizó el control y seguimiento del desarrollo del proceso de pasantía.
Responsable de la unidad de tecnología de la información	1	Brindó la inducción a los procedimientos, guió las tareas a realizar y supervisó el trabajo técnico.

*Nota: El proceso de pasantía fue supervisado tanto por un asesor académico de la universidad como por un asesor responsable de la unidad de tecnología.*

### Actividades Actuales

#### Producción Principal y Otras

El quehacer fundamental de la Unidad de Tecnología de la Información abarca el ciclo de vida completo del software, desde su concepción y construcción hasta su puesta en marcha y soporte continuo. Su enfoque productivo consiste en la ingeniería de herramientas de software, como el Sistema de Planificación de Carga Académica, con el fin de actualizar y mejorar la eficiencia de los procesos académicos y administrativos de la facultad. (Unidad de Tecnología de la Información, 2025)

#### Situación Técnica

La filosofía técnica de la UTI es actual y estratégica, enfocada en la digitalización de la facultad. La unidad da preferencia al uso de tecnologías de código abierto y a estructuras de software que puedan crecer a futuro, permitiendo la creación de soluciones duraderas y de fácil mantenimiento.

Su conjunto de herramientas tecnológicas es variado, empleando frameworks modernos para el desarrollo tanto del servidor como de la interfaz de usuario, lo que posibilita la construcción de aplicaciones web robustas con interfaces dinámicas. Sus procesos de trabajo incorporan prácticas de colaboración, apoyándose en sistemas de control de versiones para garantizar la calidad y la eficiencia del equipo. (Unidad de Tecnología de la Información, 2025)

### **Situación Administrativa**

Establecida en 2024, la UTI opera como un núcleo estratégico para la modernización tecnológica de la facultad. Su estructura no es la de un departamento autónomo, sino la de una unidad adscrita a la carrera de Ingeniería de Sistemas Informáticos, la cual forma parte del Departamento de Ingeniería y Arquitectura. En términos de gestión, la unidad articula sus proyectos en comunicación directa con los decanatos y las diversas áreas académicas para asegurar que las herramientas desarrolladas respondan a los requerimientos funcionales de los usuarios finales. (Unidad de Tecnología de la Información, 2025)

### **Metodología**

El desarrollo del Sistema de Planificación de Carga Académica se basó en una metodología integral que combinó el paradigma de Programación Orientada a Objetos (OOP) con un enfoque colaborativo. Se trabajó con el framework Laravel 9, cuya arquitectura Modelo-Vista-Controlador (MVC) permitió mantener una separación clara entre la lógica de negocio y la presentación, asegurando un código más organizado.

El lenguaje de programación principal fue PHP en su versión 8. El enfoque fue el desarrollo Backend garantizando la lógica, seguridad y el rendimiento del servidor, el éxito del proyecto requirió una estrecha colaboración con la desarrolladora Frontend. Esta coordinación fue fundamental para proveer los datos necesarios, construir los *endpoints* y asegurar que la lógica del sistema respondiera correctamente a las interacciones del usuario.

Las herramientas principales utilizadas fueron:

**Tabla 4**

*Marco Tecnológico Utilizado en el Proyecto*

<b>Tecnología</b>	<b>Versión</b>	<b>Descripción y justificación de uso</b>
PHP	8.x	Lenguaje de programación del lado del servidor. Se utilizó como base para el framework Laravel, permitiendo una lógica de negocio robusta y segura.
Laravel	9	Framework de PHP con arquitectura MVC (Modelo-Vista-Controlador). Fue elegido por la institución para estandarizar el desarrollo, agilizar la creación de funcionalidades y garantizar un código organizado y mantenible.
Liveware	2	Framework full-stack para Laravel que permite crear interfaces dinámicas con PHP, reduciendo la necesidad de escribir código JavaScript complejo. Fue fundamental para la reactividad de los formularios.
MySQL	10.4.32	Sistema de gestión de bases de datos relacional. Se utilizó para almacenar toda la información del sistema, desde los datos de los docentes hasta los registros de los formatos académicos.
Git / GitHub	Git: 2.43.0	Sistema de control de versiones distribuido. Se utilizó para gestionar el código fuente de manera colaborativa, permitiendo el trabajo





de las franjas horarias, asegurando la correcta actualización y consistencia de los horarios.											
---	--	--	--	--	--	--	--	--	--	--	--

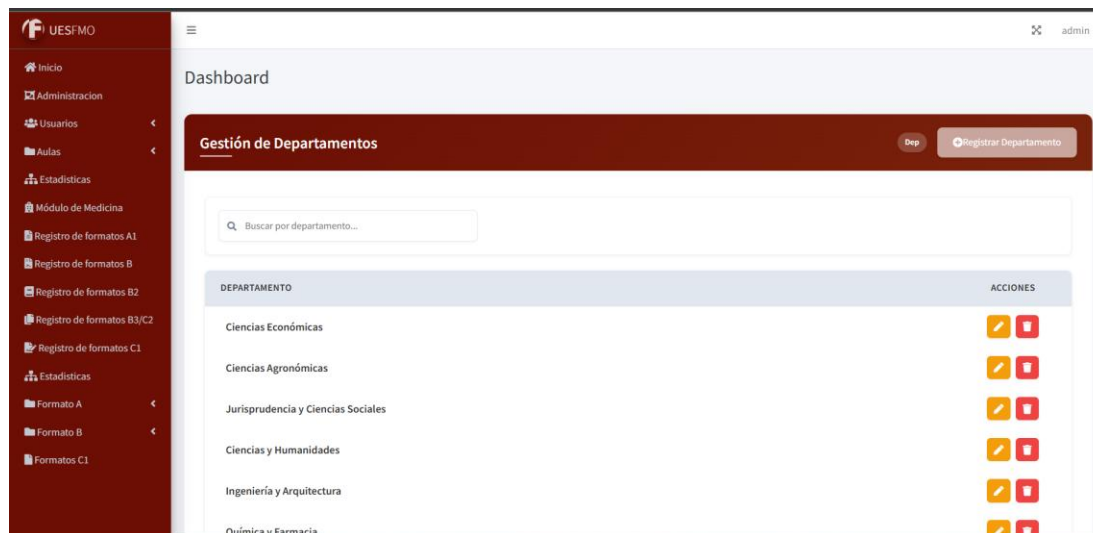
*Nota: Cronograma de ejecución semanal para las tareas de desarrollo Backend de la Fase 1.*

En este apartado, se documenta la ejecución técnica y funcional de las actividades del bimestre. Esta sección expone detalladamente las soluciones implementadas para cada una de las tareas listadas en el cronograma, con el propósito de construir la lógica robusta y el control centralizado de la información académica desde la arquitectura del servidor.

- Implementación de Operaciones CRUD y Modelos de Datos** El primer paso consistió en establecer las bases para la gestión de la información. Se implementaron desde cero las operaciones CRUD (Crear, Leer, Actualizar, Eliminar) para la tabla de departamentos. Este trabajo no se limitó a la lógica básica, sino que incluyó la creación del Modelo Eloquent, el Controlador con los métodos correspondientes (index, create, store, edit, update, destroy) y la definición de las rutas necesarias, sentando un precedente arquitectónico para los futuros módulos del sistema.

Figura 1

*Interfaz de Usuario para las Operaciones CRUD de Departamentos*



*Nota: Interfaz que consume la lógica CRUD implementada en el servidor, permitiendo al administrador visualizar, editar y eliminar los registros de los departamentos.*

- Desarrollo de un Sistema de Validación Robusto** Una prioridad desde el inicio fue garantizar la integridad y calidad de los datos. Para ello, se implementaron estrategias de validación complementarias:
  - Manejo de Restricciones a Nivel de Base de Datos:** Se desarrollaron validaciones en el backend para los formularios de docentes y aulas que incluían el manejo específico de restricciones por integridad referencial. Se implementó la captura del error SQL con código "23000", lo que permitió interceptar intentos de eliminación de registros que tenían dependencias (ej. un docente asignado a una materia) y devolver una notificación amigable al usuario en lugar de un error crítico del sistema.

## Figura 2

### *Gestión de Eliminación de registro con control de integridad referencial*

```
public function eliminarAula($id)
{
    try {
        try {
            $aula = Aulas::find($id);
            if ($aula) {
                $aula->delete();
                $path = public_path("aulas/{$aula->foto}");
                if (File::exists($path)) {
                    File::delete($path);
                }
            }
            // Emitir evento de eliminación exitosa
            $this->emit('aulaEliminada');
        } catch (QueryException $e) {
            // Capturar violaciones de restricción de clave foránea
            if ($e->getCode() == '23000') {
                $this->emit('errorEliminarAula', 'No se puede eliminar esta aula porque está asignada a otros registros.');
            }
        }
    }
}
```

*Nota: La función eliminar aula esta implementada de una manera robusta y sigue buenas practicas de desarrollo especialmente en el contexto de frameworks como Laravel.*

- **Arquitectura de Seguridad Basada en Roles y Permisos.** Se construyó desde cero la arquitectura de seguridad y control de acceso del sistema. Se implementó la lógica para la gestión de roles y la asignación de permisos específicos a cada rol. Este componente fundamental permite a los administradores del sistema definir con gran detalle qué acciones puede realizar cada tipo de usuario, asegurando que la información sensible solo sea accesible para el personal autorizado.

Figura 3

### Modelo de Datos para el Sistema de Roles y Permisos.

SELECT \* FROM `permissions`

Perfilando [ Editar en línea ] [ Editar ] [ Explicar SQL ] [ Crear código PHP ] [ Actualizar ]

1 > >> |  Mostrar todo | Número de filas: 25 | Filtrar filas: Buscar en esta tabla | Ordenar según

Opciones extra

	id	name	guard_name	created_at	updated_at
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	1	ver-usuarios	web	2024-04-11 22:45:18	2024-04-11 22:45:18
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	2	editar-usuarios	web	2024-04-11 22:45:18	2024-04-11 22:45:18
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	3	guardar-usuarios	web	2024-04-11 22:45:18	2024-04-11 22:45:18
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	4	eliminar-usuarios	web	2024-04-11 22:45:18	2024-04-11 22:45:18
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	5	ver-departamentos	web	2024-04-11 22:45:18	2024-04-11 22:45:18
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	6	editar-departamentos	web	2024-04-11 22:45:18	2024-04-11 22:45:18
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	7	guardar-departamentos	web	2024-04-11 22:45:18	2024-04-11 22:45:18
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	8	eliminar-departamentos	web	2024-04-11 22:45:18	2024-04-11 22:45:18
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	9	ver-niveles	web	2024-04-11 22:45:18	2024-04-11 22:45:18
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	10	editar-niveles	web	2024-04-11 22:45:18	2024-04-11 22:45:18
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	11	guardar-niveles	web	2024-04-11 22:45:18	2024-04-11 22:45:18
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	12	eliminar-niveles	web	2024-04-11 22:45:18	2024-04-11 22:45:18
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	13	ver-general-formatos	web	2024-04-11 22:45:18	2024-04-11 22:45:18
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	14	previa-general-formatos	web	2024-04-11 22:45:18	2024-04-11 22:45:18
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	15	guardar-formatos	web	2024-04-11 22:45:18	2024-04-11 22:45:18
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	16	ver-mis-formatos	web	2024-04-11 22:45:18	2024-04-11 22:45:18
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	17	previa-formatos	web	2024-04-11 22:45:18	2024-04-11 22:45:18
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	18	descargar-formatos	web	2024-04-11 22:45:18	2024-04-11 22:45:18

SELECT \* FROM `roles`

Perfilando [ Editar en línea ] [ Editar ] [ Explicar SQL ] [ Crear código PHP ] [ Actualizar ]

Mostrar todo | Número de filas: 25 | Filtrar filas: Buscar en esta tabla | Ordenar según la clave: Ning

Opciones extra

	id	name	guard_name	created_at	updated_at
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	1	Super Administrador	web	2024-04-11 22:45:18	2024-04-11 22:45:18
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	2	Administrador	web	2024-04-11 22:45:18	2024-04-11 22:45:18
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	3	Docente	web	2024-04-11 22:45:18	2024-04-11 22:45:18
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	4	Jefe Medicina	web	2024-10-09 09:55:33	2024-10-09 09:55:33
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	5	Docente Medicina	web	2024-10-10 06:25:44	2024-10-10 06:25:44
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	7	Administrador Decanato	web	2024-12-09 18:56:32	2024-12-09 18:56:32

*Nota: Estructura de las tablas roles y permissions en la base de datos. Este modelo relacional es la base de la arquitectura de seguridad y control de acceso del sistema.*

- **Lógica de Negocio para Módulos de Planificación Académica.** Se tradujeron los complejos requerimientos del Departamento de Medicina en código funcional del lado del servidor.
  - Gestión de Componentes Académicos: Se desarrolló la lógica para la gestión de componentes, lo que incluyó la definición de las relaciones en los modelos de Eloquent y la configuración de las rutas y funciones en los controladores para manejar el registro y la edición. Se implementaron también funciones para la liberación de registros y la resolución de conflictos en la lógica.
  - Soporte Backend para Formatos y Reportes: Se proveyó el soporte lógico para los formatos institucionales. Esto incluyó el desarrollo del reporte en Word para el Formato A (oferta académica), el análisis y la implementación de la lógica de negocio y las consultas a la base de datos para el Formato C (Carga Laboral), y la implementación de la lógica para el Formato B1 (Oferta Académica), que incluía el filtrado de docentes por tipo de contratación y el cálculo del tiempo de servicio.
  
- **Soporte de Consultas para Franja Horaria.** Se brindó apoyo crucial al equipo de desarrollo mediante el diseño e implementación de las consultas a la base de datos necesarias para alimentar la vista previa y la gestión de las franjas horarias. Asimismo, se realizaron las modificaciones pertinentes en el backend para asegurar la correcta actualización y consistencia de los horarios. La importancia de estas franjas horarias radica en que permiten deducir la carga laboral semanal del docente, detallando los días y las horas impartidas al día. son un componente esencial para la gestión académica a nivel de todos los demás departamentos de la facultad, aunque estas franjas también son utilizadas por el Departamento de Medicina para su planificación interna, funcionando como un sistema general de asignación de recursos.

Figura 4

## Gestión de Franjas Horarias: Listado de Docentes y Filtros

The screenshot shows the 'Gestión de Franjas Horarias' interface. At the top, there is a search bar and a 'Generar Reporte' button. Below the search bar, there are filters for 'Modalidad de Contrato' with options: 'Ley de Salario', 'Contrato Permanente', 'Eventual', and 'Servicios Profesionales No Personales'. The main content is a table with columns: 'DOCENTE', 'FRANJA', 'ESTADO', and 'ACCIONES'.

DOCENTE	FRANJA	ESTADO	ACCIONES
AC ANA LUCY CANALES DE BONILLA	ASIGNADO	ACTIVA	[Iconos de acciones]
AP ARISTIDES MAURICIO PERLA LOPEZ	ASIGNADO	INACTIVA	[Iconos de acciones]
CB CID MILAGRO BÉNÉTEZ DE CASTRO	NO ASIGNADO	INACTIVA	[Iconos de acciones]
CR CÉSAR MISael RODRÍGUEZ FRANCO	ASIGNADO	ACTIVA	[Iconos de acciones]
CS CÉSAR NEFTALI SÁENZ ROMERO	ASIGNADO	INACTIVA	[Iconos de acciones]

*Nota: Esta vista consume el resultado de la lógica y consultas implementadas en el Backend, mostrando un listado de docentes con filtros por modalidad de contrato.*

Figura 5

## Gestión Detallada de Franjas Horarias por Docente

The screenshot shows the 'Gestión Detallada de Franjas Horarias por Docente' interface for Ana Lucy Canales de Bonilla. It features two main tabs: 'Consultar Franjas' and 'Crear Nueva'. The 'Consultar Franjas' tab shows a dropdown for 'Selección de Ciclo Académico' set to 'II-2025' and a count of '2 Franja(s) Encontradas'. The 'Crear Nueva' tab shows a dropdown for 'Selección de Ciclo para Nueva Franja' set to 'II-2025' and a 'Crear Franja' button. A yellow warning box states: 'Solo se puede tener una franja activa por ciclo', 'Las franjas nuevas se crean inactivas', and 'Después de crear la franja, podrás asignar horarios y activarla'. At the bottom, there are 'Cerrar y Volver' buttons for both tabs.

*Nota: La vista donde se cual consume el Backend desarrollado presenta dos pestañas principales: Consultar Franjas, que despliega las franjas existentes (activas e inactivas) por ciclo, y Crear Nueva, que permite asignar un nuevo ciclo académico y una franja, enfatizando la regla de una franja activa por ciclo.*

**Fase 2: Optimización, Desarrollo de Lógicas Complejas y Refactorización (Mayo - Junio 2025).** Durante el segundo bimestre, el enfoque del desarrollo backend transitó de la construcción de funcionalidades iniciales a la optimización del rendimiento y la implementación de lógicas de negocio significativamente más complejas. Esta etapa fue crucial para robustecer la plataforma, mejorar su mantenibilidad y dar soporte a los requerimientos avanzados de la gestión académica, como el manejo de asignaturas compartidas y la gestión por ciclos.

**Tabla 6**

*cronograma con el listado de tareas asignadas para la fase 2.*

Descripción de la tarea solicitada	Cronograma Segundo Bimestre							
	Mayo				Junio			
Diseño e Implementación de Grupos Comunes (Arquitectura): Desarrollar la lógica principal y el modelo de datos para permitir que un grupo principal tenga múltiples grupos comunes asociados.								
Grupos Comunes (Numeración y Validación): Implementar el esquema de numeración automática y las validaciones de consistencia a nivel de backend para Grupos Comunes, previniendo duplicidades y conflictos.								
Refactorización Integral del Servidor: Realizar la refactorización profunda de componentes Livewire y del sistema de generación de PDF, reestructurando métodos para mayor claridad y eficiencia.								
Optimización de Consultas y Estructura de Datos: Optimizar consultas SQL								

(incluyendo DocumentosController) y agregar columnas a las tablas principales (ciclo académico, tipo de grupo) para mejorar el rendimiento general.								
Lógica de Gestión por Ciclo y Reportes Avanzados: Integrar el soporte backend para la gestión por ciclos académicos. Mejorar la lógica de generación de reportes (PDF/Word) para visualizar correctamente la estructura de Grupos Comunes.								
Mejoras en Lógica de Asignación y Eliminación: Fortalecer la lógica de asignación de tutores (considerando ciclo/departamento) e implementar condiciones en el <i>backend</i> para prevenir la eliminación accidental de asignaciones en el Formato B1.								

*Nota: Se muestra el listado de las tareas asignadas para la fase 2 con su periodo de tiempo.*

En la presente sección, se documenta la ejecución técnica y las soluciones de software desarrolladas en este bimestre. Este segmento expone las implementaciones de las lógicas de negocio complejas (como Grupos Comunes) y el proceso de refactorización llevado a cabo para mejorar significativamente el rendimiento y la mantenibilidad de la arquitectura del servidor, dando cumplimiento a todas las tareas detalladas en el cronograma.

***Implementación de la Lógica de Negocio para Grupos Comunes y Compartidos*** Esta fue la funcionalidad más estratégica del periodo, diseñada para resolver el complejo escenario operativo de la asignación de una misma materia a docentes de diferentes carreras. Los Grupos Comunes son un concepto, una nueva

característica clave implementada en el formato A “Oferta Académica”, que permite crear un grupo de una materia y compartirlo con una o más carreras, ya sean del mismo o de diferentes departamentos.

La lógica de negocio aplicada considera la restricción fundamental de que la materia debe estar al mismo nivel académico en todas las carreras con las que se pretende compartir el grupo. El funcionamiento técnico es el siguiente: al momento de crear un Grupo Común, se seleccionan las carreras involucradas. La asignación del número de grupo se realiza de manera estructurada; por ejemplo, si se crean tres grupos comunes (1, 2 y 3), el número '1' siempre corresponderá al grupo de la carrera que inició la creación del Grupo Común. Los números de grupo subsiguientes se asignan secuencialmente según el orden de selección de las otras carreras (la primera carrera seleccionada obtiene el grupo '2', la segunda el '3', y si hay más carreras que números de grupo disponibles, estas se asociarán al último número de grupo disponible, en este caso, el '3'). Esta lógica asegura la correcta vinculación y gestión de los grupos compartidos.

- **Diseño de la Arquitectura de Datos y de Negocio:** Se diseñó y desarrolló desde el backend la lógica para permitir que un grupo principal tuviera múltiples grupos comunes asociados. Esto implicó la modificación del modelo de datos para soportar estas relaciones complejas y la implementación de la lógica en los controladores para gestionar la creación, vinculación y eliminación de dichos grupos.
- **Esquema de Numeración Automática:** Para garantizar la coherencia y evitar la ambigüedad en los reportes, se creó un esquema de numeración automática para los grupos comunes. Esta lógica, implementada en el servidor, aseguraba que cada grupo común tuviera un identificador único y estandarizado, facilitando su gestión por parte de los usuarios y su correcta representación en los documentos.
- **Validación y Pruebas de Consistencia:** Se incorporaron validaciones a nivel de backend para asegurar la integridad de los datos al momento de crear y asignar grupos comunes, previniendo duplicidades o asignaciones conflictivas.

## Figura 6

Listado de Formatos A (Oferta académica) registrados “Formato A General”.

**Formatos registrados**

Ciclo seleccionado 2-2025

Si el estado esta en borrador puedes editar el formato, de lo contrario solo podras descargar el documento.

Seleccione el departamento  
Ciencias Económicas

Seleccione el ciclo del cual desea ver los informes  
2-2025

ID	Fecha de creación	Código carrera	Carrera	Ciclo	estado	Acción
288	2025-10-07 21:32:46	LS0804	Licenciatura en Mercadeo Internacional (2004)	2-2025	Prevalido	Ver vista previa, Descargar pdf, Eliminar formato, Editar formato
289	2025-10-07 21:33:10	LS0803	Licenciatura en Administración de Empresas (1994)	2-2025	Prevalido	Ver vista previa, Descargar pdf, Eliminar formato, Editar formato
290	2025-05-28 08:13:21	LS0801	Licenciatura en Economía (1994)	2-2025	Prevalido	Ver vista previa, Seguir editando el formato, Eliminar formato
291	2025-05-28 08:13:47	LS0802	Licenciatura en Contaduría Pública (1994)	2-2025	Prevalido	Ver vista previa, Descargar pdf, Eliminar formato, Editar formato
319	2025-05-25 15:54:27	LS0808EL	Licenciatura en Logística Comercial Internacional (2023)	2-2025	Prevalido	Ver vista previa, Descargar pdf, Eliminar formato, Editar formato

Fin del contenido

*Nota: Muestra la interfaz de Formatos registrados (Formato A Oferta Académica). Esta vista permite seleccionar el departamento y ciclo, y visualiza el listado de carreras con sus formatos asociados, indicando el estado y las acciones disponibles, como Editar Formato, que dirige a la vista de Registro de un nuevo formato A (Figura 5).*

## Figura 7

Interfaz para la Asignación de Materias y Grupos Comunes.

**Registro de un nuevo formato A1**

Seleccione un nivel para cargar las materias correspondientes  
Primer año

Vista previa Descargar en pdf

Se muestran las materias de Primer año

Registrar una nueva materia

Código: FRI254	Código: III154	Código: DGR154	Código: TCI254
Francés Intensivo II	Inglés Intermedio Intensivo I	Didáctica General I (Electiva)	Teoría de la Comunicación
Unidades Val: 8.0 Grupo: 40 Grupo: 1	Unidades Val: 8.0 Grupo: 40 Grupo: 1	Unidades Val: 4.0 Grupo: 40 Grupo: 1	Unidades Val: 4.0 Grupo: 40 Grupo: 1
Tipo de grupo: Grupo Teorico	Tipo de grupo: Grupo Teorico	Tipo de grupo: Grupo Teorico	Tipo de grupo: Grupo Teorico
Prerrequisito: Francés Intensivo I	Prerrequisito: Inglés Básico Intensivo	Prerrequisito: Psicopedagogía I	Prerrequisito: Teoría de la Comunicación y de la Información II (Electiva)

*Nota: Interfaz de registro del Formato A (Oferta Académica), donde se visualiza el resultado de las consultas de materias y se expone la funcionalidad para "Asignar como grupo común", la cual es soportada por la lógica de negocio implementada en el servidor.*

### ***Optimización y Refactorización del Backend para Rendimiento y***

***Mantenibilidad.*** Se dedicó un esfuerzo técnico considerable a mejorar la calidad y la eficiencia del código base existente, con una visión a largo plazo.

- **Refactorización Integral de Componentes del Servidor:** Se llevó a cabo una refactorización profunda y sistemática de los componentes Livewire y del sistema de generación de PDF. Este proceso no fue meramente cosmético; implicó reestructurar los métodos responsables de la lógica de negocio para hacerlos más claros, reutilizables y eficientes. Se consolidaron funciones duplicadas y se estandarizó la forma en que se procesaban los datos de niveles y asignaturas en todos los formatos.
- **Optimización de Consultas SQL:** Se identificaron y optimizaron varias consultas a la base de datos para mejorar la velocidad de carga del sistema, especialmente en módulos con alta densidad de información. Se realizaron modificaciones en el DocumentosController para que las consultas incluyeran identificadores de formatos y datos de grupo de manera más eficiente, reduciendo el número de queries necesarias.
- **Mejora de la Estructura de Datos:** Se agregaron columnas adicionales en las tablas de la base de datos para mostrar información clave como el ciclo académico y el tipo de grupo directamente en las tablas principales, evitando la necesidad de realizar consultas complejas (joins) en tiempo real y mejorando así el rendimiento general.

***Soporte Backend para Funcionalidades Avanzadas de Gestión.*** Se implementaron nuevas lógicas para dar soporte a requerimientos administrativos y de experiencia de usuario más sofisticados.

- **Lógica de Gestión por Ciclo Académico:** Se integró en todo el backend el soporte para la gestión por ciclos académicos. Se añadieron filtros en las consultas que permitían a la lógica de negocio identificar el ciclo al que pertenecía cada docente y cada formato, como en el caso del Formato B3(propuesta de contratación de personal docente). Esto fue fundamental para una administración de datos más precisa y ordenada.
- **Mejoras en la Lógica de Asignación y Eliminación:** Se mejoró significativamente la lógica de asignación de tutores para que considerara el ciclo académico activo y otras variables relevantes, como el departamento o el tipo de grupo. Un tutor es la persona encargada que se encuentra en los centros de práctica externos, como hospitales y clínicas, y su gestión se refleja en el sistema a través del Formato B3 (Carga Académica). En la lógica de negocio, esta funcionalidad está específicamente diseñada para el Departamento de Medicina, permitiendo crear un reporte detallado solamente del personal de tutoría asignado a dicho departamento. Adicionalmente, se fortaleció la seguridad del sistema al implementar condiciones adicionales en el backend para prevenir la eliminación accidental de asignaciones en el Formato B1 (Oferta Académica), añadiendo una capa de protección a la integridad de los datos. Es importante destacar que el Formato B1 (Oferta Académica) cuenta con dos variantes de lógica de negocio distintas: una que se aplica al sistema general para los demás departamentos, y una segunda variante que fue desarrollada específicamente para atender las necesidades particulares del Departamento de Medicina. Esta diferenciación en la lógica subraya el diseño robusto del sistema y la protección específica implementada en sus funcionalidades clave.
- **Optimización de la Generación de Reportes (PDF y Word):** Se mejoró sustancialmente la lógica del servidor para la generación de reportes, asegurando que la información de los nuevos grupos comunes y compartidos se visualizara correctamente en los documentos oficiales. Se trabajó en la adaptación del formato para que los grupos se visualizaran agrupados por asignatura, con su respectiva numeración, niveles y detalles, un requerimiento clave para el personal académico.

**Fase 3: Integración de Servicios, Seguridad y Consolidación del Sistema (Julio - Septiembre 2025).** El último bimestre de la pasantía representó la etapa de madurez y consolidación del proyecto. El enfoque se centró en robustecer la aplicación mediante la integración de servicios externos, la configuración de una arquitectura de archivos escalable y la implementación de funcionalidades de gestión avanzadas que requerían optimizaciones complejas a nivel de base de datos e interconexión entre sistemas.

**Tabla 7**

*cronograma con el listado de tareas asignadas para la fase 3.*

Descripción de la tarea solicitada	Cronograma Tercer Bimestre									
	Julio			Agosto			Septiembre			
Filtro Avanzado de Aulas: Implementar el filtro avanzado en AulasController para el cálculo de disponibilidad en tiempo real, excluyendo aulas ya asignadas, y añadir la búsqueda directa por nombre.										
Integración de Base de Datos para Tutores: Realizar la integración del módulo de docentes con la segunda base de datos para gestionar el estado de tutoría por ciclo, incluyendo habilitación de procesos CRUD y manejo de errores.										
Implementación de Autenticación Externa (Google): Desarrollar e integrar el inicio de sesión con Google (OAuth 2.0 / Laravel										

Socialite), incluyendo la configuración de Google Cloud Console y las validaciones de seguridad (CSRF).										
Configuración de Gestión Documental y Consolidación: Configurar el sistema de archivos de Laravel (filesystems) para la carga y almacenamiento seguro de documentos, establecer reglas de nomenclatura y preparar la arquitectura para escalabilidad en la nube.										

*Nota: Cronograma que muestra las tareas de ejecución semanal para el desarrollo Backend de la Fase 3.*

El siguiente segmento tiene como finalidad exponer la ejecución técnica y las soluciones de arquitectura desarrolladas en el cierre del proyecto. En este punto, se detallan las tareas enfocadas en la integración de servicios externos (OAuth, Bases de Datos) y la configuración de una plataforma segura y escalable, elementos que consolidaron la madurez del sistema, tal como se listan en el cronograma.

**Gestión Avanzada de Módulos: Consultas Optimizadas e Integración de Datos.** Se trabajó intensamente en la optimización de los módulos de aulas y tutores, implementando funcionalidades que requerían un manejo de datos en tiempo real y la conexión con fuentes de datos externas.

El módulo de Tutores consistió en la integración de la información de los encargados de los centros de práctica externos (hospitales y clínicas), la cual se gestiona a través del Formato B3 (Carga Académica) del Departamento de Medicina. Para este módulo, se desarrolló una lógica de negocio específica que permite crear

reportes detallados del personal de tutoría, asegurando una gestión precisa y alineada con las necesidades de dicho departamento.

El módulo de Aulas se enfocó en la implementación de funcionalidades para una gestión inteligente de los espacios físicos de la facultad. Esto incluyó la capacidad de filtrar y visualizar aulas disponibles según criterios como el día, bloque horario, previniendo así conflictos de calendarización y optimizando el uso de los recursos institucionales."

- **Filtro Avanzado de Aulas con Cálculo de Disponibilidad en Tiempo Real:**

Se implementó una de las herramientas más complejas a nivel de backend: un filtro avanzado para la gestión de aulas modulo que es utilizado por el sistema base (Sistemas de los formatos generales que usan los demás departamentos a excepción de medicina) y el sistema de medicina. Este desarrollo incluyó:

- La creación de una consulta SQL optimizada en el AulasController que permitía filtrar aulas no solo por día y bloque horario, sino que también calculaba su disponibilidad en tiempo real, excluyendo automáticamente aquellas que ya estaban asignadas en la planificación académica.
- Se añadió una funcionalidad de búsqueda directa por nombre, mejorando la rapidez con que los administradores pueden localizar espacios específicos.

## Figura 8

### Gestión de Aulas: Búsqueda y Disponibilidad Horaria

The screenshot shows the 'Aulas' management interface. On the left is a navigation menu with options like 'Inicio', 'Administración', 'Usuarios', 'Aulas', 'Gestión de Aulas', 'Asignación de aulas', 'Estadísticas', 'Módulo de Medicina', and various 'Registro de formatos' (A1, B, B2, B3/C2, C1, A, B, C1). The main content area is titled 'Aulas' and 'Asignación de Aulas'. It features two dropdown menus: 'Seleccione un día:' (set to 'Lunes') and 'Seleccione una hora:' (set to '07:00:00 - 09:00:00'). Below these is a search box 'Buscar por nombre de aula:' with 'Aula 20' entered and a 'Buscar Aula' button. A table titled 'Aulas disponibles para el día Lunes en horario 07:00:00 - 09:00:00' lists available classrooms with columns for ID, Nombre del Aula, and Capacidad.

ID	Nombre del Aula	Capacidad
21	AULA 20	80 personas
51	AULA 201	50 personas
53	AULA 203	60 personas
54	AULA 204	60 personas
55	AULA 205	60 personas
57	AULA 207	50 personas

*Nota: Esta vista permite al usuario filtrar las aulas por día y realizar una búsqueda por nombre, presentando un listado de las aulas disponibles que cumplen con los criterios especificados, junto con su capacidad.*

## Figura 9

### Interfaz del Filtro Avanzado para la Gestión de Aulas

The screenshot shows the 'Gestión de aulas de medicina' interface. At the top right is a button 'Agregar Nueva Aula'. Below is a 'Filtrar por' section with a dropdown 'Nombre del Aula' and a search box 'Buscar...'. The main part is a table with columns: Nombre Aula, Capacidad, Foto, Pantalla, Coordenadas, and Acciones. The 'Foto' column contains 'No disponible' for all rows. The 'Acciones' column contains edit and delete icons for each row.

Nombre Aula	Capacidad	Foto	Pantalla	Coordenadas	Acciones
Auditorio 2		No disponible			[Edit] [Delete]
Auditorio 3		No disponible			[Edit] [Delete]
Auditorio Hospital		No disponible			[Edit] [Delete]
AULA 32		No disponible			[Edit] [Delete]
AULA 34		No disponible			[Edit] [Delete]
AULA 35		No disponible			[Edit] [Delete]
AULA 36		No disponible			[Edit] [Delete]
AULA 37		No disponible			[Edit] [Delete]
AULA 38		No disponible			[Edit] [Delete]

*Nota: El resultado visual del filtro de aulas en tiempo real. La lógica del backend realiza las consultas optimizadas para mostrar únicamente los espacios disponibles según los criterios seleccionados.*

- **Integración con Base de Datos para la Gestión de Tutores:** Para centralizar la información y mantener la consistencia, se llevó a cabo la integración del módulo de docentes con una segunda base de datos.
  - Se habilitaron procesos completos de creación, edición y validación robusta de datos de tutores, reduciendo significativamente los errores de registro.
  - El núcleo de esta funcionalidad fue la gestión del estado de tutoría por ciclo académico, enlazado directamente con una segunda base de datos que almacena información crítica como hospitales, modalidades de contrato y tiempos de vinculación.
  - Se implementaron mecanismos de manejo de errores y trazabilidad para que cualquier inconsistencia entre la base de datos local y la segunda base de datos pudiera ser identificada y corregida con facilidad.

***Implementación de Autenticación Externa con Google (Flujo OAuth 2.0)*** Un logro técnico fundamental de esta fase fue la modernización de la seguridad y el acceso al sistema mediante la integración de un proveedor de identidad externo.

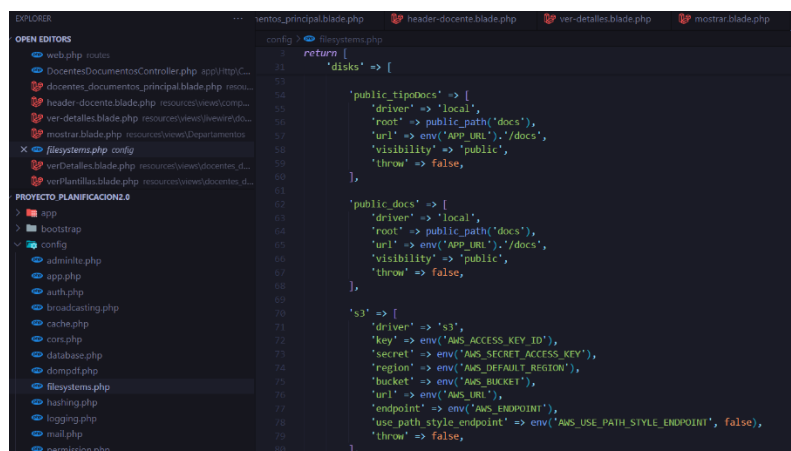
- Se desarrolló e integró el inicio de sesión con Google mediante el paquete Laravel Socialite.
- El desarrollo incluyó la configuración completa del servicio OAuth 2.0 en Google Cloud Console, la gestión de las credenciales de cliente y la configuración de las rutas de callback en la aplicación Laravel.
- Se implementaron todas las validaciones de seguridad necesarias, incluyendo la verificación de estado para prevenir ataques de falsificación de solicitudes entre sitios (CSRF), garantizando un proceso de login seguro.
- Se realizaron pruebas exhaustivas para asegurar el correcto registro de usuarios nuevos y el inicio de sesión de usuarios existentes, implementando la asignación automática del departamento correspondiente al usuario basado en la información provista por Google.

**Configuración de un Sistema de Gestión Documental Escalable** Se diseñó y configuró la arquitectura del lado del servidor para la carga y almacenamiento de documentos de docentes, preparando la aplicación para un crecimiento futuro.

- Se configuró el sistema de archivos (filesystems) de Laravel para la carga y almacenamiento seguro de documentos de docentes.
- Se establecieron rutas y carpetas específicas (public/docs), con la visibilidad adecuada para que los archivos fueran accesibles desde la aplicación. La arquitectura para este módulo se definió en el archivo de configuración de Laravel, como se detalla en la **Figura 7**.
- Crucialmente, la configuración se preparó para una futura escalabilidad con servicios de almacenamiento en la nube, demostrando una visión a largo plazo.
- Se establecieron reglas de nomenclatura estandarizadas para los archivos subidos, lo que mejoró notablemente la identificación, consulta y almacenamiento de documentos en el sistema.

**Figura 10**

### Configuración de Discos de Almacenamiento en Laravel



```

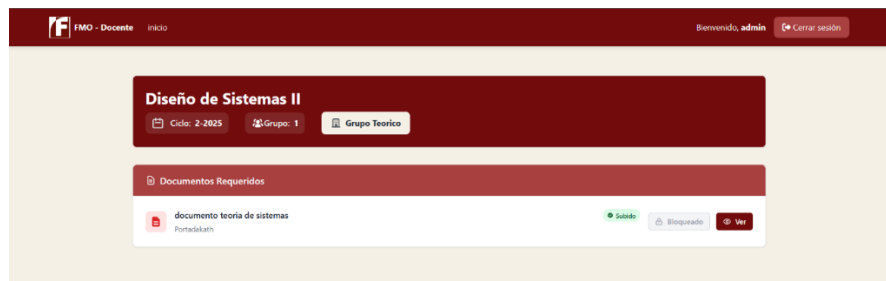
return [
    'disks' => [
        'public tipodocs' => [
            'driver' => 'local',
            'root' => public_path('docs'),
            'url' => env('APP_URL').'/docs',
            'visibility' => 'public',
            'throw' => false,
        ],
        'public docs' => [
            'driver' => 'local',
            'root' => public_path('docs'),
            'url' => env('APP_URL').'/docs',
            'visibility' => 'public',
            'throw' => false,
        ],
        's3' => [
            'driver' => 's3',
            'key' => env('AWS_ACCESS_KEY_ID'),
            'secret' => env('AWS_SECRET_ACCESS_KEY'),
            'region' => env('AWS_DEFAULT_REGION'),
            'bucket' => env('AWS_BUCKET'),
            'url' => env('AWS_URL'),
            'endpoint' => env('AWS_ENDPOINT'),
            'use_path_style_endpoint' => env('AWS_USE_PATH_STYLE_ENDPOINT', false),
            'throw' => false,
        ],
    ],
];

```

*Nota: Código del archivo de configuración filesystems.php, donde se definen los discos de almacenamiento locales y se prepara la arquitectura para una futura integración con servicios en la nube (S3).*

Figura 11

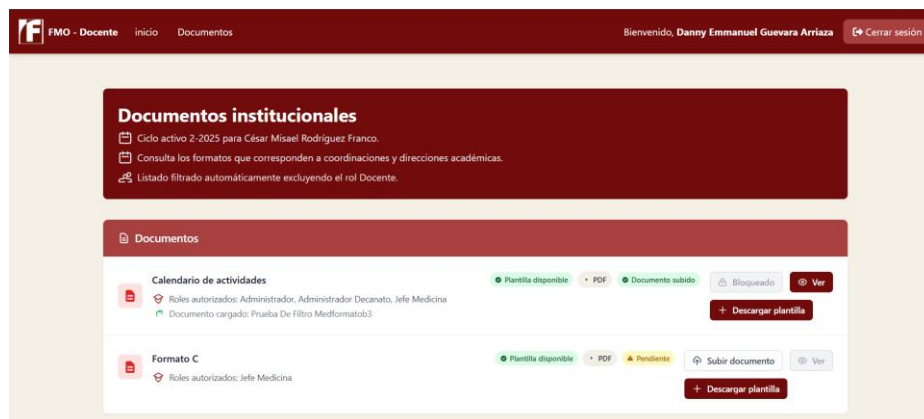
### Interfaz del Docente para la Carga de Documento



*Nota: Interfaz final que consume el servicio de almacenamiento configurado en el backend, permitiendo a los docentes subir y gestionar su documento.*

Figura 12

### Módulo FMO - Docente: Documentos Institucionales



*Nota: La vista presenta los **Documentos institucionales** correspondientes al ciclo 2-2025. Se visualiza un listado de documentos clave indicando su estado, roles autorizados para su gestión y las acciones disponibles como descargar la plantilla o subir el documento.*

**Discusión y Competencias Adquiridas.** La pasantía como Desarrollador Backend representó una inmersión profunda en la arquitectura del lado del servidor, donde el rol principal fue construir lógica y funcional del Sistema de Planificación de Carga Académica. La experiencia se centró en traducir los requerimientos del negocio en un código seguro, eficiente y escalable, estableciendo los cimientos sobre los cuales opera toda la interfaz de usuario. Tareas como el diseño del modelo de datos para la gestión de roles y permisos, la implementación de validaciones robustas y la integración con una base de datos externa, fueron cruciales para garantizar la integridad y seguridad de la información.

En el plano técnico, esta experiencia permitió consolidar un dominio avanzado del framework Laravel, la gestión de bases de datos relacionales con MySQL y la implementación de lógicas de negocio complejas. Los desafíos más formativos incluyeron la optimización de consultas SQL complejas para evitar cuellos de botella en el rendimiento y la refactorización de controladores para mejorar la mantenibilidad del código. La colaboración constante con la desarrolladora Frontend fue fundamental, proveyendo los datos necesarios para que la interfaz cobrara vida. Esta pasantía solidificó la comprensión de que un backend sólido y bien diseñado es el pilar indispensable sobre el que se construye cualquier aplicación web exitosa.

## Conclusiones

- Se cumplió a cabalidad con el plan de trabajo, entregando una arquitectura de backend robusta, segura y escalable, para todas las funcionalidades del Sistema de Planificación de Carga Académica.
- Se estableció una base de software altamente segura y confiable, un logro evidenciado por la implementación de un sistema de roles y permisos, la creación de validaciones de datos rigurosas en el servidor y la integración de métodos de autenticación modernos y seguros como Google (OAuth 2.0).
- Se logró optimizar significativamente el rendimiento y la eficiencia del sistema mediante la refactorización estratégica de controladores y la optimización de consultas SQL complejas, como la que permite el filtrado de aulas en tiempo real, lo que garantiza tiempos de respuesta rápidos para el usuario final.
- Se consolidaron competencias avanzadas y específicas del desarrollo backend, destacando la integración exitosa con servicios de terceros, la conexión y sincronización con bases de datos externas para la gestión de tutores y el diseño de una arquitectura de almacenamiento de archivos preparada para la nube, habilidades cruciales en el desarrollo de software moderno.

## Recomendaciones

- **Implementar Pruebas Automatizadas en el Backend:** Para fortalecer la fiabilidad del sistema, se recomienda desarrollar un conjunto de pruebas unitarias y de integración (utilizando herramientas como PHPUnit). Esto permitiría validar automáticamente la lógica de negocio y con cada nuevo cambio, reduciendo la posibilidad de regresiones y errores.
- **Adoptar un Estándar de Codificación Formal para PHP:** Para mejorar la consistencia y mantenibilidad del código a largo plazo, sería beneficioso adoptar y documentar un estándar de codificación para PHP (como PSR-12). Esto aseguraría que todo el equipo de desarrollo siga las mismas convenciones, facilitando las revisiones de código y la incorporación de nuevos miembros.
- **Estructurar un Proceso de Inducción (Onboarding) para Desarrolladores:** Se propone diseñar un programa de inducción formal que ofrezca una capacitación inicial sobre la arquitectura del servidor, el esquema de la base de datos. Esto permitiría a los futuros pasantes comprender la estructura del backend de manera más rápida y eficiente.

## Referencias

Laravel. (s.f.). *Laravel documentation*. <https://laravel.com/docs>

Livewire. (s.f.). *Laravel Livewire documentation*. <https://livewire.laravel.com/docs>

Universidad de El Salvador, Facultad Multidisciplinaria Oriental. (s. f.). *Historia de la Facultad Multidisciplinaria Oriental*. <https://www.fmoues.edu.sv/historia>

Universidad de El Salvador, Facultad Multidisciplinaria Oriental. (s. f.). *Visión*. <https://www.fmoues.edu.sv/vision>

User, S. (s.f.). *Historia*. <http://www.fmoues.edu.sv/index.php/fmo/historia>

Git. (s.f.). *Git documentation*. <https://git-scm.com/book/en/v2>

GitHub. (2025). *GitHub*. <https://github.com/>

Visual Studio Code. (2021, 3 de noviembre). *Visual Studio Code*. <https://code.visualstudio.com/>

XAMPP installers and downloads for Apache Friends. (s.f.). *XAMPP*. <https://www.apachefriends.org/index.html>

**Visto Bueno del Tutor de la Institución**

Jefa de la Unidad de Tecnología de la Información

**Tutor Externo:**

Vo. Bo. Ing. \_\_\_\_\_

Ligia Astrid Hernández Bonilla



# Anexos

**Anexo A: Constancia de finalización de la pasantía**

**UNIVERSIDAD DE EL SALVADOR  
FACULTAD MULTIDISCIPLINARIA ORIENTAL  
DEPARTAMENTO DE INGENIERIA Y ARQUITECTURA  
UNIDAD DE TECNOLOGIA DE LA INFORMACION**

**CONSTANCIA DE LA INSTITUCIÓN**

**Ciudad Universitaria Oriental, San Miguel, 08 de septiembre de 2025**

**Ing. Milagro Alicia González de Reyes**

Coordinadora de Procesos de Grado De la Carrera de Ingeniería de Sistemas Informáticos de la Facultad Multidisciplinaria Oriental.

Presente.

El suscrito: **Ing. Ligia Astrid Hernández Bonilla**, Jefa de la Unidad de Tecnología de la Información de la Facultad Multidisciplinaria Oriental, Universidad de El Salvador HACE CONSTAR que, El Br. **DANYS EMANUEL GUEVARA ARRIAZA**, ha realizado satisfactoriamente su PASANTIA DE PRACTICA PROFESIONAL en el proyecto: Desarrollador Backend del Sistema de Planificación de Carga Académica en el Departamento de Medicina habiendo iniciado en la fecha comprendida del **03 de Marzo de 2025**, hasta **08 de Septiembre de 2025** por un total de **6 meses**.

Y para los efectos que el interesado estime conveniente, se firma y Sella la presente, en Ciudad Universitaria Oriental, San Miguel a las **10:00** horas del día **08 de septiembre de 2025**.  
Atentamente:

F: 

Ing. Ligia Astrid Hernández Bonilla  
Universidad de El Salvador Facultad Multidisciplinaria Oriental  
Jefa de la Unidad de Tecnología de la Información



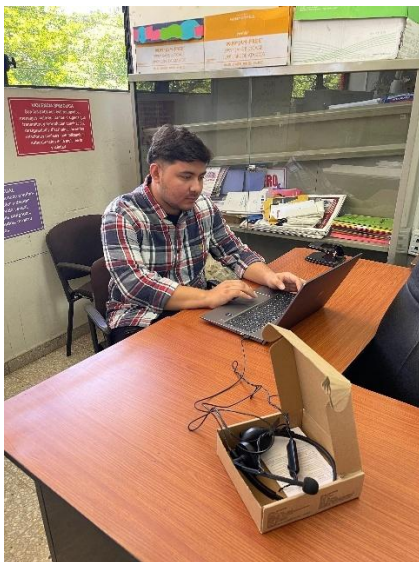
## Anexo B: Evidencia del Proceso de Trabajo

**Figura B.** *Desarrollo de Lógica de Negocio en el Entorno de la UTI*



*Nota: Durante una jornada de trabajo, realizando labores de programación y análisis para la implementación de módulos del sistema.*

**Figura B.2** *Implementación de Funcionalidades del Backend*



*Nota: Concentrado en el desarrollo y depuración de código del lado del servidor para el Sistema de Planificación de Carga Académica.*

**Figura B.3** Planificación de tareas del sistema



*Nota: Equipo de trabajo analizando los requerimientos y la asignación de tareas.*