

Proyecto de Sistema de Información

Versión del Framework Laravel

Este proyecto fue desarrollado utilizando la versión 11 del framework Laravel, el cual permite una arquitectura limpia y moderna con soporte para componentes, rutas, migraciones y controladores estructurados.

Sistema Gestor de Base de Datos (SGBD)

Se utilizó PostgreSQL como sistema gestor de base de datos debido a su robustez, eficiencia en consultas complejas y compatibilidad con Laravel.

Autenticación con Breeze y Tailwind

Se implementó el paquete Breeze, el cual facilita la autenticación de usuarios incluyendo rutas para login, registro y restablecimiento de contraseña. Breeze también integra Tailwind CSS, proporcionando un diseño moderno, limpio y completamente responsive sin necesidad de configurar Bootstrap.

Seguridad con Spatie

Para la gestión de roles y permisos se usó el paquete Spatie Laravel Permission. Esto permitió asignar distintos niveles de acceso a usuarios según su rol (administrador, docente, evaluador, etc.), asegurando que solo usuarios autorizados accedan a funcionalidades específicas.

Diccionario de Datos

El diccionario de datos fue implementado para todas las entidades del sistema, detallando los atributos, tipos de datos, restricciones y relaciones clave entre tablas como usuarios, proyectos, asignaturas, docentes, evaluaciones, entre otros, teniendo en cuenta las entidades que eran obligatorias y teniendo en cuenta el diccionario de los compañeros de base de datos.

Menú de Opciones

Se desarrolló un menú con acceso a todos los módulos del sistema: Proyectos, Asignaturas, Docentes, Evaluaciones, Instituciones, Programas, Facultades, Departamentos, etc.

Experiencia Técnica del Desarrollo

La experiencia técnica fue interesante pero estresante. Se aplicaron conceptos de MVC, uso de migraciones, relaciones Eloquent, validaciones, controladores REST y Blade components. Además, trabajar con Laravel 11 proporcionó estabilidad, y Breeze nos facilitó mucho todo pero Spatie nos generó muchos problemas.

Importancia del Trabajo en Equipo

El trabajo colaborativo permitió distribuir tareas, detectar errores con mayor facilidad y mantener la motivación. Cada integrante aportó desde su conocimiento y fortaleció las decisiones técnicas, logrando cumplir con los objetivos del proyecto de forma eficaz.

Utilidad del Trabajo Colaborativo con GIT

Git y GitHub fueron esenciales para organizar el trabajo. Si bien comenzamos a trabajar con ramas y commits organizados hacia el final del proyecto, logramos integrar todo en un repositorio específico de esta entrega, ya que el proyecto lo teníamos en el repositorio donde teníamos todos los trabajos del semestre, por lo que la mayoría de commits quedaron ahí.