

Parcial - Ingeniería Web

Proyecto Silabo con VUE

Estudiante	Escuela	Asignatura
Melany YasminLazo Arana mlazoa@ulasalle.edu.pe	Carrera Profesional de Ingeniería de Software	Ingeniería Web Semestre 2025-I ING-WEB

Estudiante	Escuela	Asignatura
Alvaro Andre Machaca Melendez amachacam@ulasalle.edu.pe	Carrera Profesional de Ingeniería de Software	Ingeniería Web Semestre 2025-I ING-WEB

Estudiante	Escuela	Asignatura
Carla Fernanda Ropa Calizaya cropac@ulasalle.edu.pe	Carrera Profesional de Ingeniería de Software	Ingeniería Web Semestre 2025-I ING-WEB

Índice

1. Introducción	3
1.1. Tecnologías y herramientas utilizadas	3
2. Marco Teórico	3
2.1. Vue.js	3
2.2. Aplicaciones de Página Única (SPA)	3
2.3. Gestión de Sílabos Académicos	3
3. Descripción del Proyecto	3
3.1. Características Principales	4
4. Estructura del Proyecto	4
5. Instalación y Ejecución	5
5.1. Requisitos previos	5
5.2. Instalación local (modo desarrollo)	5
5.2.1. Clonar el repositorio	5
5.2.2. Instalar dependencias	5
5.2.3. Ejecutar el servidor de desarrollo	6
5.3. Ejecución con Docker	6
5.3.1. Construir y levantar el contenedor	6
6. Descripción Detallada de los Componentes	6
6.1. Vistas Principales	6
6.2. Componentes Reutilizables	7
6.3. Estructura de Datos	7

7. Funcionalidades Desarrolladas	7
7.1. Interfaz de Visualización vs. Edición	7
7.2. Gestión de Datos Generales	7
7.3. Gestión de Competencias	8
7.4. Gestión de Unidades Didácticas	8
7.5. Sistema de Evaluación	8
7.6. Gestión de Bibliografía	9
7.7. Firma Digital	11
8. Personalización y Configuración	13
9. Próximas Mejoras	13
10. Conclusiones	13
11. Referencias	13

1. Introducción

Este documento describe el desarrollo de **Silabo App**, una aplicación web interactiva diseñada para facilitar la gestión, edición y visualización dinámica de sílabos académicos universitarios. El proyecto se ha desarrollado utilizando tecnologías web modernas como Vue.js 3 y Bootstrap, con el objetivo de proporcionar una herramienta eficiente para que los docentes puedan estructurar contenidos por unidades y semanas, definir competencias, agregar metodologías, bibliografía y criterios de evaluación de manera intuitiva y estandarizada.

En el contexto académico actual, la gestión eficiente de sílabos es fundamental para garantizar la calidad educativa y facilitar los procesos administrativos. Silabo App busca resolver esta necesidad mediante una interfaz amigable y funcionalidades específicamente diseñadas para el entorno universitario.

1.1. Tecnologías y herramientas utilizadas

- **Vue.js 3**: Framework JavaScript progresivo utilizado como base para el desarrollo frontend.
- **Vue Router**: Sistema de enrutamiento oficial para la navegación entre vistas de Vue.js.
- **Bootstrap 5**: Framework CSS para el desarrollo de interfaces responsive.
- **Docker**: Plataforma de contenedores para facilitar el despliegue.

2. Marco Teórico

2.1. Vue.js

Vue.js es un framework JavaScript progresivo para la construcción de interfaces de usuario. Está diseñado para ser adoptado gradualmente y puede ser integrado con otros proyectos o bibliotecas existentes. Vue 3, la versión utilizada en este proyecto, introduce mejoras significativas en términos de rendimiento, compatibilidad con TypeScript y una API de composición que facilita la reutilización de código.

2.2. Aplicaciones de Página Única (SPA)

Las SPA (Single Page Applications) son aplicaciones web que cargan una única página HTML y actualizan dinámicamente su contenido según la interacción del usuario sin necesidad de recargar la página completa. Esto proporciona una experiencia más fluida y rápida para el usuario, similar a la de una aplicación de escritorio.

2.3. Gestión de Sílabos Académicos

Un sílabo académico es un documento que detalla los objetivos, contenidos, metodologías, criterios de evaluación y bibliografía de un curso universitario. La gestión efectiva de estos documentos es esencial para la organización académica y la comunicación entre docentes y estudiantes. La digitalización y estandarización de este proceso a través de aplicaciones web como Silabo App permite optimizar recursos, facilitar actualizaciones y mejorar la accesibilidad.

3. Descripción del Proyecto

Silabo App es una aplicación web interactiva que permite a los docentes universitarios crear, editar y visualizar sílabos académicos de manera estructurada y estandarizada.

3.1. Características Principales

- **Visualización de sílabos:** Interfaz amigable para consultar información académica estructurada como:
 - Datos generales del curso (nombre, código, créditos, docente, etc.)
 - Competencias del curso
 - Unidades didácticas con sus semanas y contenidos
 - Sistema de evaluación
 - Bibliografía y recursos
- **Edición completa:** Permite modificar todos los aspectos del sílabo:
 - Edición de datos generales
 - Gestión de competencias
 - Creación y modificación de unidades didácticas
 - Configuración del sistema de evaluación
 - Carga de firma digital del docente
- **Gestión de unidades:** Organización de contenidos por unidades y semanas.
- **Sistema de evaluación:** Definición de criterios y fechas de evaluación.
- **Firma digital:** Carga de firma del docente para oficializar el documento.
- **Navegación por páginas:** Interfaz dividida en secciones accesibles mediante un sistema de paginación.

4. Estructura del Proyecto

La estructura del proyecto se organiza de manera modular para facilitar su mantenimiento y extensión:

```
silabo-app/  
|----public/          # Archivos públicos estáticos  
|----src/             # Código fuente principal  
|   |----assets/      # Recursos estáticos (CSS, imágenes)  
|   |----components/  # Componentes Vue reutilizables  
|       |----SilaboEditor.vue # Editor de secciones del sílabo  
|       +----ModalBase.vue # Componente para ventanas modales  
|   |----data/        # Datos de la aplicación  
|       +----silabo.json # Estructura de datos del sílabo  
|   |----views/       # Vistas principales  
|       |----HomeVi.vue # Vista de visualización del sílabo  
|       +----EditView.vue # Vista de edición del sílabo  
|   |----router/      # Configuración de rutas  
|   |----App.vue      # Componente raíz  
|   +----main.js      # Punto de entrada de la aplicación  
|----Dockerfile       # Instrucciones para construir la imagen Docker  
|----docker-compose.yml # Configuración para ejecución con Docker  
|----package.json     # Dependencias y scripts de la aplicación  
+----README.md        # Documentación del proyecto
```

5. Instalación y Ejecución

5.1. Requisitos previos

Antes de comenzar, asegúrese de tener instalado lo siguiente en su máquina:

- Node.js 14+ (<https://nodejs.org/>)
- npm (<https://www.npmjs.com/>) o yarn (<https://yarnpkg.com/>)
- Git (<https://git-scm.com/>)
- Docker & Docker Compose (<https://docs.docker.com/compose/>) (opcional para contenedores)

5.2. Instalación local (modo desarrollo)

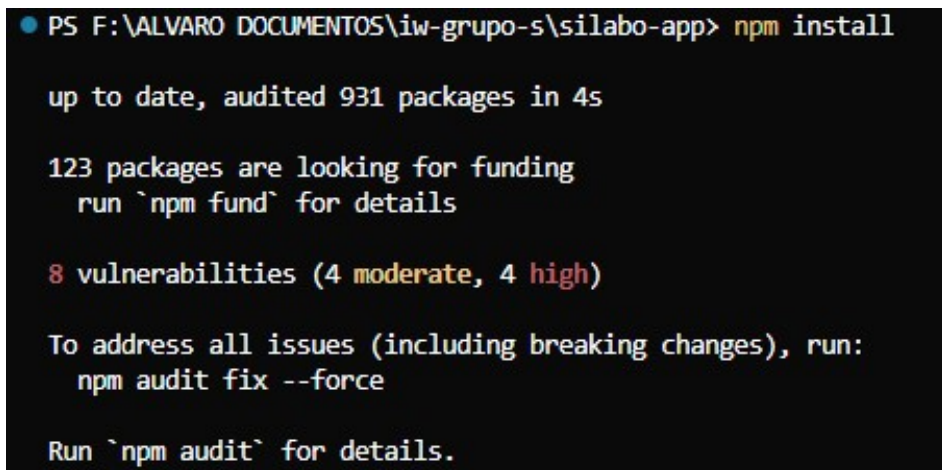
5.2.1. Clonar el repositorio

```
git clone https://github.com/CarlaSalle23/iw-grupo-s.git  
cd silabo-app
```

5.2.2. Instalar dependencias

```
npm install  
# o con yarn  
yarn install
```

A continuación, se muestra el resultado de la ejecución del comando de instalación:



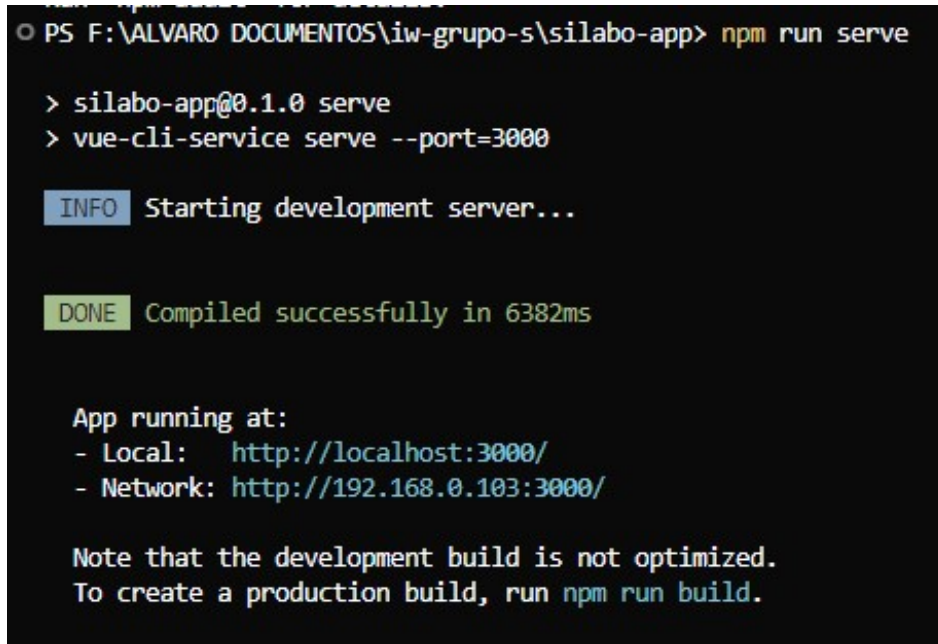
```
PS F:\ALVARO DOCUMENTOS\iw-grupo-s\silabo-app> npm install  
  
up to date, audited 931 packages in 4s  
  
123 packages are looking for funding  
run `npm fund` for details  
  
8 vulnerabilities (4 moderate, 4 high)  
  
To address all issues (including breaking changes), run:  
npm audit fix --force  
  
Run `npm audit` for details.
```

Figura 1: Resultado de la instalación de dependencias con npm. Se observan 8 vulnerabilidades (4 moderadas, 4 altas) que podrían solucionarse ejecutando `npm audit fix --force`.

5.2.3. Ejecutar el servidor de desarrollo

```
npm run serve  
# o con yarn  
yarn serve
```

El resultado de la ejecución del servidor se muestra a continuación:



```
PS F:\ALVARO DOCUMENTOS\iw-grupo-5\silabo-app> npm run serve  
  
> silabo-app@0.1.0 serve  
> vue-cli-service serve --port=3000  
  
[INFO] Starting development server...  
  
[DONE] Compiled successfully in 6382ms  
  
App running at:  
- Local:   http://localhost:3000/  
- Network: http://192.168.0.103:3000/  
  
Note that the development build is not optimized.  
To create a production build, run npm run build.
```

Figura 2: Ejecución del servidor de desarrollo. La aplicación está disponible en <http://localhost:3000> (local) y <http://192.168.0.103:3000> (red).

La aplicación estará disponible en <http://localhost:3000>

5.3. Ejecución con Docker

5.3.1. Construir y levantar el contenedor

```
docker-compose up --build
```

La aplicación estará disponible en <http://localhost:3000>

6. Descripción Detallada de los Componentes

6.1. Vistas Principales

El proyecto cuenta con dos vistas principales:

- **HomeVi.vue**: Vista de visualización del sílabo completo. Presenta toda la información del sílabo de manera estructurada y organizada para su consulta.
- **EditView.vue**: Vista de edición del sílabo. Proporciona una interfaz para modificar todos los aspectos del sílabo de manera intuitiva.

6.2. Componentes Reutilizables

Los componentes más importantes del sistema son:

- **SilaboEditor.vue**: Componente principal para editar las diferentes secciones del sílabo de manera modular.
- **ModalBase.vue**: Componente para mostrar ventanas modales, utilizado para confirmar acciones o añadir información adicional.

6.3. Estructura de Datos

El archivo `silabo.json` contiene la estructura principal de datos del sílabo, organizando la información en secciones como:

- Información general del curso
- Competencias y resultados de aprendizaje
- Unidades didácticas
- Contenidos por semanas
- Metodología de enseñanza
- Sistema de evaluación
- Bibliografía y recursos

7. Funcionalidades Desarrolladas

La aplicación Silabo App ofrece dos modos principales de interacción: el modo de visualización (Home) y el modo de edición (Edit), cada uno con interfaces adaptadas para sus respectivos propósitos.

7.1. Interfaz de Visualización vs. Edición

La aplicación cuenta con dos perspectivas principales:

- **Modo Home**: Permite visualizar toda la información del sílabo de manera estructurada y amigable para consulta.
- **Modo Edit**: Ofrece formularios interactivos para la modificación de cada sección del sílabo.

7.2. Gestión de Datos Generales

La aplicación permite editar información básica del curso como:

- Nombre y código del curso
- Créditos y horas lectivas
- Prerrequisitos
- Información del docente
- Periodo académico

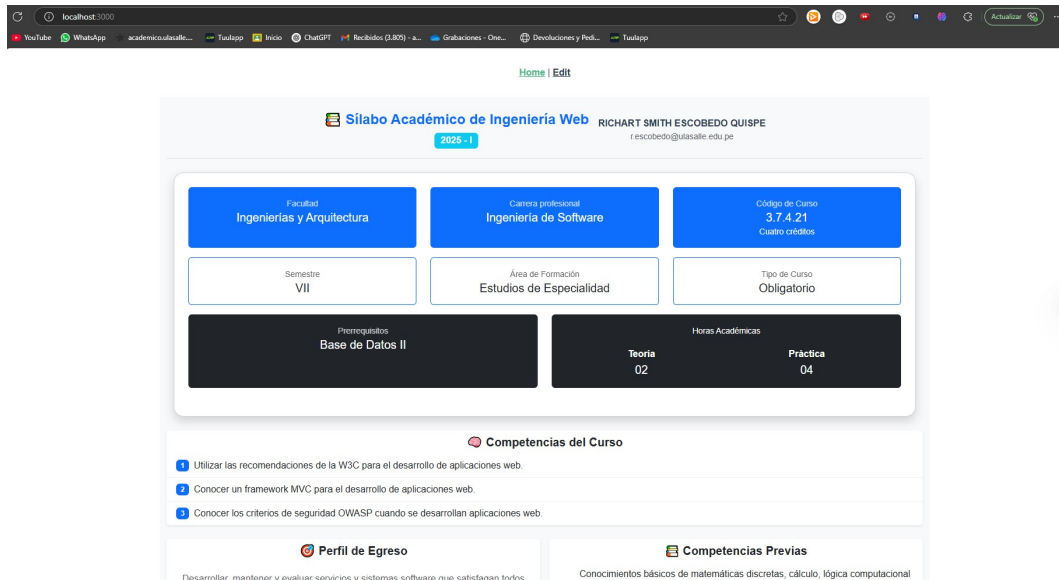


Figura 3: Interfaz de visualización con datos generales del curso (Home)

7.3. Gestión de Competencias

Esta funcionalidad permite:

- Añadir/eliminar competencias generales
- Añadir/eliminar competencias específicas
- Definir resultados de aprendizaje por competencia
- Especificar el perfil de egreso
- Establecer competencias previas requeridas

7.4. Gestión de Unidades Didácticas

Permite la estructuración del contenido académico:

- Creación de unidades con título y descripción
- Organización por semanas con fechas específicas
- Asignación de contenidos específicos por semana
- Definición de actividades y recursos por unidad
- Opción para añadir nuevas unidades

7.5. Sistema de Evaluación

Esta sección permite configurar:

- Criterios de evaluación (evidencias, exámenes)

[Home](#) | [Edit](#)

Sílabo Académico de Ingeniería Web

2025 - I

Richart Smith Escobedo Qu

r.escobedo@ulasalle.edu.pr

Facultad	Carrera profesional	Código de Curso
Ingenierías y Arquitectura	Ingeniería de Software	3.7.4.21
		Créditos
Semestre	Área de Formación	Tipo de Curso
VII	Estudios de Especialidad	Obligatorio
Prerrequisitos		Horas Teoría
Base de Datos II		02
		Horas Práctica
		04
Firma del docente		
Elegir archivo No se eligió ningún archivo		

Competencias del Curso

1 Utilizar las recomendaciones

2 Conocer un framework MVC

3 Conocer los criterios de selección

Perfil de Egreso

Desarrollar, mantener y evaluar servicios y sistemas software que satisfagan todos los requisitos del usuario y se comporten de forma fiable y eficiente, sean asequibles de desarrollar y

Competencias Previas

1 Conocimientos básicos de

2 Comprensión de la responsabilidad

3 Gestionar el desarrollo de software

4 Capacidad de aprender y aplicar

Figura 4: Interfaz de edición de datos generales del curso (Edit)

- Ponderaciones de cada criterio (porcentajes de calificación)
- Fechas límite de evaluaciones
- Descripción detallada de cada actividad evaluativa
- Configuración de la fórmula de calificación final

7.6. Gestión de Bibliografía

Permite organizar referencias bibliográficas:

- Bibliografía básica
- Bibliografía complementaria


Competencias del Curso

- 1 Utilizar las recomendaciones de la W3C para el desarrollo de aplicaciones web.
- 2 Conocer un framework MVC para el desarrollo de aplicaciones web.
- 3 Conocer los criterios de seguridad OWASP cuando se desarrollan aplicaciones web.


Perfil de Egreso

Desarrollar, mantener y evaluar servicios y sistemas software que satisfagan todos los requisitos del usuario y se comporten de forma fiable y eficiente, sean asequibles de desarrollar y mantener y cumplan normativas de calidad, aplicando las teorías, principios, métodos y prácticas de la informática.


Competencias Previas

Conocimientos básicos de matemáticas discretas, cálculo, lógica computacional como base para la estructuración de datos, algoritmos, programación de computadoras, diseño y arquitectura de software.


- 1
- 2 Comprensión de la responsabilidad y ética profesional.
- 3 Gestionar el desarrollo de sistemas de software.
- 4 Capacidad de aprender y actualizarse permanentemente.

Anterior


Página 1 de 3

Siguiente


Figura 5: Visualización de competencias del curso y perfil de egreso (Home)


Competencias del Curso

- 1 Utilizar las recomendaciones
- 2 Conocer un framework MVC
- 3 Conocer los criterios de seguridad


Perfil de Egreso

Desarrollar, mantener y evaluar servicios y sistemas software que satisfagan todos los requisitos del usuario y se comporten de forma fiable y eficiente, sean asequibles de desarrollar y mantener y cumplan normativas de calidad, aplicando las teorías, principios, métodos y prácticas de la informática.


Competencias Previas

- 1 Conocimientos básicos de matemáticas discretas, cálculo, lógica computacional como base para la estructuración de datos, algoritmos, programación de computadoras, diseño y arquitectura de software.
- 2 Comprensión de la responsabilidad y ética profesional.
- 3 Gestionar el desarrollo de sistemas de software.
- 4 Capacidad de aprender y actualizarse permanentemente.

Anterior

Página 1 de 3

Siguiente

Figura 6: Interfaz de edición de competencias (Edit)

- Recursos digitales
- Enlaces web relevantes

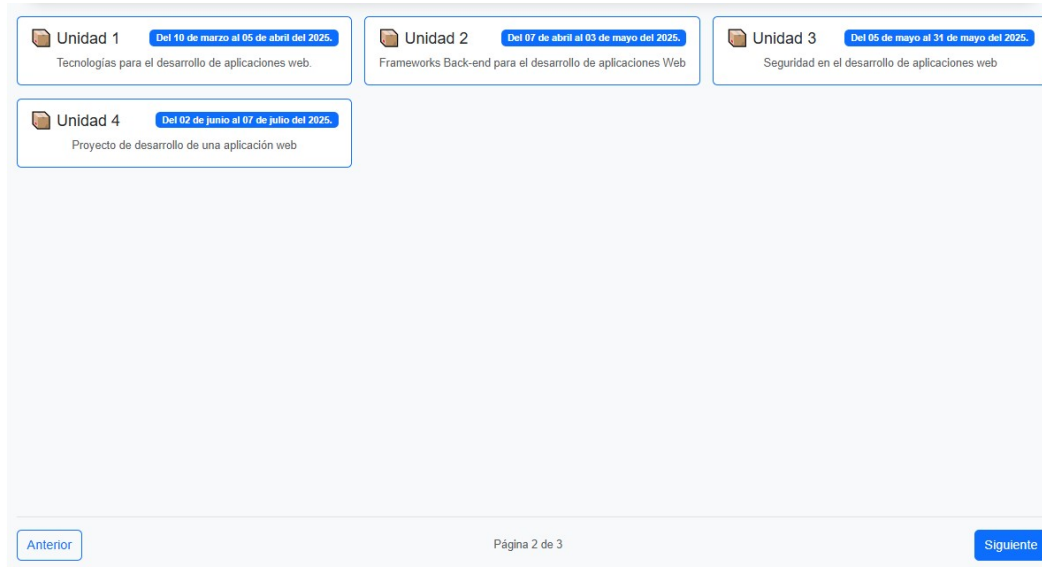


Figura 7: Visualización de unidades didácticas con fechas (Home)

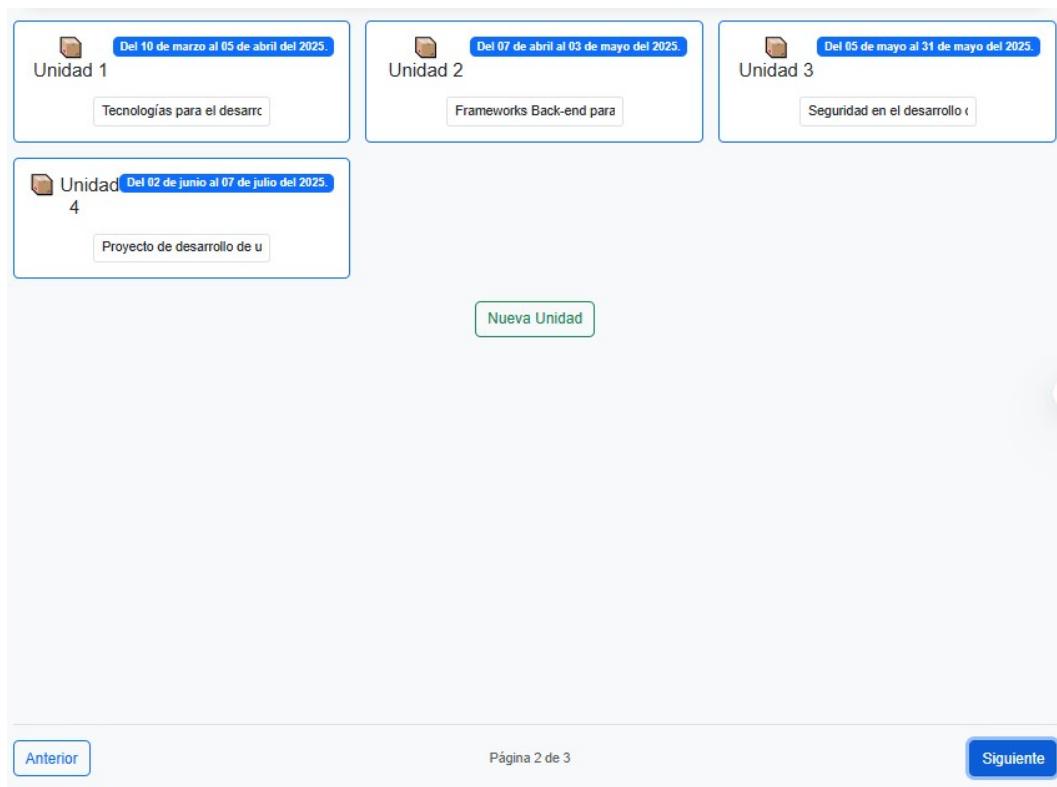


Figura 8: Interfaz de edición de unidades didácticas (Edit)

7.7. Firma Digital

Funcionalidad para oficializar el documento:

Sistema de Evaluación			
Tipo	Peso	Fecha	Descripción
Evidencia 1	El promedio de la evidencia 1 y la evidencia 2 corresponde al 20% de la nota final. %	Hasta el 05 de abril.	Prácticas de laboratorio sobre uso de tecnologías y plataformas de desarrollo web. Exposiciones individuales y grupales.
Evidencia 2	El promedio de la evidencia 1 y la evidencia 2 corresponde al 20% de la nota final. %	Hasta el 26 de abril.	Prácticas de laboratorio sobre uso de tecnologías y plataformas de desarrollo web. Exposiciones individuales y grupales.
EXAMEN PARCIAL	La evaluación parcial corresponde al 30% de la nota final. %	Evaluación del 05 al 10 de mayo Ingreso de notas hasta el 18 de mayo.	Presentación de avance de proyecto: Desarrollo incremental de una aplicación web Examen - rubrica
Evidencia 3	El promedio de la evidencia 3 y la evidencia 4 corresponde al 20% de la nota final. %	Hasta el 31 de mayo.	Prácticas de laboratorio sobre uso de tecnologías y plataformas de desarrollo web. Exposiciones individuales y grupales.
Evidencia 4	El promedio de la evidencia 3 y la evidencia 4 corresponde al 20% de la nota final. %	Hasta el 21 de junio.	Prácticas de laboratorio sobre uso de tecnologías y plataformas de desarrollo web. Exposiciones individuales y grupales.
EXAMEN FINAL	La evaluación final corresponde al 30% de la nota final. %	Evaluación del 30 de junio hasta el 05 de julio Ingreso de notas hasta el 06 de julio. Entrega y revisión de exámenes 07 de julio.	Presentación de avance de proyecto: Desarrollo incremental de una aplicación web Examen - rubrica

Anterior Página 3 de 3 Siguiente

Figura 9: Visualización del sistema de evaluación (Home)

Sistema de Evaluación			
Tipo	Peso	Fecha	Descripción
<input type="text" value="Evidencia 1"/>	<input type="text" value=""/> %	<input type="text" value="Hasta el 05 de abril."/>	<input type="text" value="Prácticas de laboratorio sot"/>
<input type="text" value="Evidencia 2"/>	<input type="text" value=""/> %	<input type="text" value="Hasta el 26 de abril."/>	<input type="text" value="Prácticas de laboratorio sot"/>
<input type="text" value="EXAMEN PARCIAL"/>	<input type="text" value=""/> %	<input type="text" value="Evaluación del 05 al 10 de i"/>	<input type="text" value="Presentación de avance de"/>
<input type="text" value="Evidencia 3"/>	<input type="text" value=""/> %	<input type="text" value="Hasta el 31 de mayo."/>	<input type="text" value="Prácticas de laboratorio sot"/>
<input type="text" value="Evidencia 4"/>	<input type="text" value=""/> %	<input type="text" value="Hasta el 21 de junio."/>	<input type="text" value="Prácticas de laboratorio sot"/>
<input type="text" value="EXAMEN FINAL"/>	<input type="text" value=""/> %	<input type="text" value="Evaluación del 30 de junio I"/>	<input type="text" value="Presentación de avance de"/>

Anterior Página 3 de 3 Siguiente

Figura 10: Interfaz de edición del sistema de evaluación (Edit)

- Carga de imagen de firma
- Validación del documento

- Timestamp de última modificación

8. Personalización y Configuración

Silabo App ofrece varias opciones de personalización a través de archivos de configuración:

- **vue.config.js**: Configuración del proyecto Vue
- **tailwind.config.js**: Personalización de Tailwind CSS
- **postcss.config.js**: Configuración de PostCSS

9. Próximas Mejoras

Se han identificado las siguientes áreas de mejora para futuras versiones:

- Almacenamiento persistente de datos
- Exportación a PDF
- Sistema de autenticación
- Historial de versiones
- Plantillas predefinidas

10. Conclusiones

Silabo App representa una solución efectiva para la gestión académica de sílabos universitarios, ofreciendo una interfaz moderna e intuitiva para docentes. El uso de tecnologías como Vue.js 3, Bootstrap y TailwindCSS proporciona una base sólida para el desarrollo de funcionalidades avanzadas y una experiencia de usuario optimizada.

La aplicación responde a necesidades reales del entorno académico, facilitando la estandarización de contenidos, la organización por competencias y la visualización estructurada de información relevante para el proceso educativo.

Las principales ventajas de Silabo App incluyen:

- **Usabilidad**: Interfaz intuitiva que facilita la creación y edición de sílabos.
- **Flexibilidad**: Estructura modular que permite adaptarse a diferentes formatos de cursos.
- **Estandarización**: Promueve un formato unificado de sílabos a nivel institucional.
- **Eficiencia**: Reduce el tiempo dedicado a tareas administrativas.

Con las mejoras planificadas, Silabo App tiene el potencial de convertirse en una herramienta integral para la gestión docente, contribuyendo a la calidad académica y a la optimización de procesos administrativos en entornos universitarios.

11. Referencias

- Vue.js Documentation. (2023). <https://vuejs.org/>
- Vue Router Documentation. (2023). <https://router.vuejs.org/>
- Bootstrap Documentation. (2023). <https://getbootstrap.com/>
- Docker Documentation. (2023). <https://docs.docker.com/>