

Universidad de San Carlos de Guatemala

Escuela de Ciencias y Sistemas

Facultad de Ingeniería

Introducción a la programación y Computación 1

Segundo Semestre 2024

Catedrático:

Tutor académico:



FIUSAC
FACULTAD DE INGENIERÍA
UNIVERSIDAD DE SAN CARLOS DE GUATEMALA

PROYECTO 2

Sistema ECYS

Objetivos

Generales

- Familiarizar al estudiante con el desarrollo de aplicaciones web.
- El estudiante aplique los conocimientos adquiridos en el curso de Introducción a la programación y computación 1.
- Elaborar la lógica para presentar una solución a la problemática planteada.

Específicos

- Familiarizar al estudiante con el concepto de la programación web.
- Lograr que el estudiante comprenda el concepto de backend y frontend empleando una arquitectura cliente - servidor.
- Adentrar al estudiante a la programación web por medio de APIs.
- Que el estudiante se relacione con el uso de NodeJS.
- El estudiante se familiarice con librerías y frameworks para el desarrollo de aplicaciones frontend.

Descripción General

La Escuela de Ciencias y Sistemas de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala enfrenta varios desafíos en la gestión eficiente de los cursos impartidos en la carrera de Ingeniería en Ciencias y Sistemas. Con una creciente demanda de estudiantes y la complejidad de administrar múltiples cursos, es crucial implementar un sistema web robusto que facilite la administración de cursos, profesores y alumnos. Este sistema permitirá la gestión precisa de los cursos, asegurando que cada asignatura esté adecuadamente registrada, con profesores asignados, y que los estudiantes puedan inscribirse sin dificultades, mejorando la organización y reduciendo los errores administrativos.

El sistema garantizará una trazabilidad completa de la información académica, permitiendo al administrador cargar y actualizar los cursos y profesores de manera eficiente, mientras que los profesores podrán gestionar sus clases, actividades y calificaciones de los estudiantes. Además, la plataforma proporcionará a los alumnos una interfaz intuitiva para inscribirse en cursos, consultar sus notas y actualizar su información personal, todo en tiempo real. La capacidad de modificar y eliminar registros de manera segura será fundamental para mantener la base de datos actualizada y evitar inconsistencias.

Más allá de la gestión básica de usuarios y cursos, el sistema también ofrecerá la capacidad de generar reportes académicos que ayuden a visualizar el rendimiento de los estudiantes a lo largo del tiempo. Esta herramienta será clave para mejorar la toma de decisiones en la administración educativa y para proporcionar a los profesores una visión clara del progreso de sus alumnos. Implementada con Express en el backend y React en el frontend, la aplicación garantizará un rendimiento óptimo y una experiencia de usuario moderna y fluida, adecuada para un entorno académico de alto nivel.

Aplicación

La Escuela de Ciencias y Sistemas tiene la necesidad de desarrollar una aplicación web para la gestión de los cursos impartidos en la carrera y lo ha contratado a usted para que con sus habilidades de programación le de solución al problema. A este sistema podrán acceder los siguientes tipos de usuarios:

Rol	Descripción
Administrador	Cargará profesores y cursos al sistema, y podrá visualizar a todos los alumnos dentro de la escuela. La plataforma contará con un único usuario administrador
Profesor	Podrá ser asignado a los distintos cursos que imparte, y dentro de estos, puede cargar alumnos, crear actividades, y generar reportes sobre las actividades.
Alumno	Podrá ser asignado a los distintos cursos que esté recibiendo, ver sus notas en cada curso y actualizar su información.

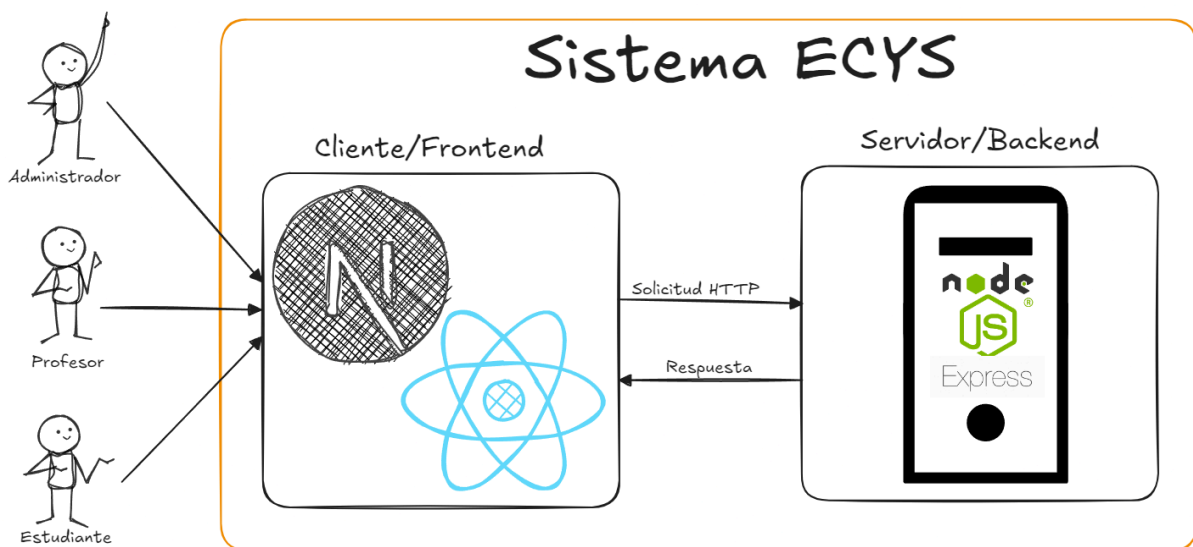
La aplicación contará con las siguientes funcionalidades:

- Ingreso de datos
- Modificación de datos.
- Visualización de datos.
- Eliminación de datos.

Arquitectura de la aplicación

El sistema estará conformado por dos partes: frontend y backend:

- La parte de backend debe ser una API desarrollada en NodeJS haciendo uso de la librería de Express.js.
- La parte de frontend debe ser una aplicación web, haciendo el uso de la librería React y también puede utilizar NextJS para complementar el desarrollo de la aplicación Frontend, además de las herramientas que considere necesarias.



El puerto a utilizar por lado del Backend tiene que ser:

- Puerto 4000

Flujo de la aplicación

1. Cualquiera de los usuarios entra a la página web ingresando la ruta local de la aplicación frontend.
2. Los usuarios pueden realizar acciones como ver, agregar, modificar y eliminar datos.
3. Por cada acción se realiza una solicitud HTTP a la API que a su vez retornará los datos pertinentes en su respuesta HTTP.
4. La API en NodeJS es la que almacenará todos los datos por medio de listas.
5. Para el envío y recepción de datos, se utilizará el formato JSON, permitiendo una comunicación eficiente y estructurada entre el frontend y el backend.

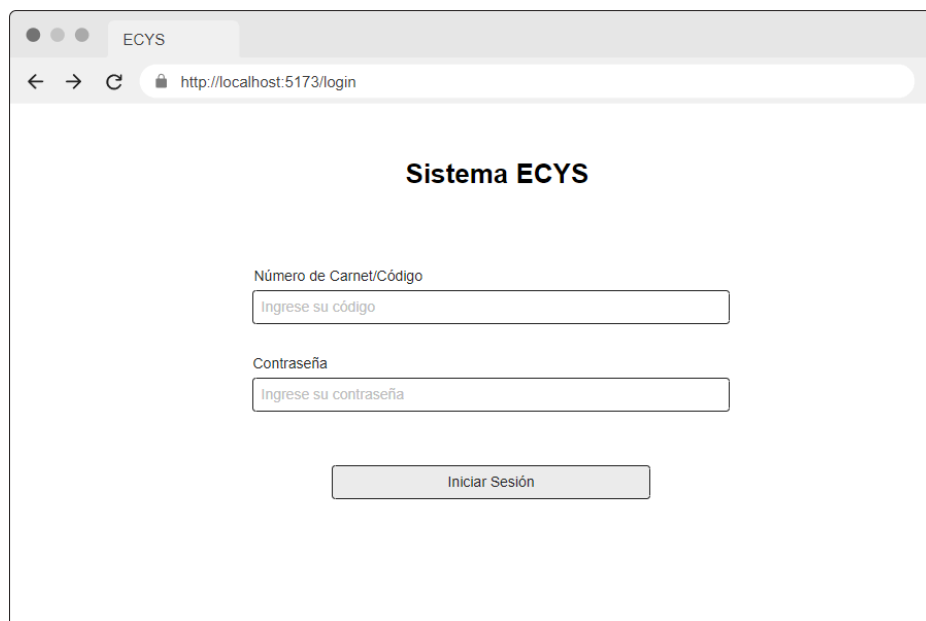
Página Principal

Deberá diseñar una página de inicio de la aplicación, de esta página se partirá para ir al módulo de autenticación. Deberá contener la siguiente información de manera creativa (esto con el fin de promocionar la página de ECYS):

- Nombre de la aplicación
- Logo
- Descripción de a lo que se dedica la página web
- Acerca de
- Contacto de soporte con la información del estudiante quien desarrolló la página.

Módulo de Autenticación

La función principal de este módulo es darle el acceso a los usuarios por medio de su número de carnet o código y contraseña.



The screenshot displays a web browser window with the title 'ECYS'. The address bar shows 'http://localhost:5173/login'. The main content area has the heading 'Sistema ECYS'. Below the heading, there are two input fields. The first is labeled 'Número de Carnet/Código' and contains the placeholder text 'Ingrese su código'. The second is labeled 'Contraseña' and contains the placeholder text 'Ingrese su contraseña'. Below these fields is a button labeled 'Iniciar Sesión'.

Se tomarán en cuenta las siguientes consideraciones:

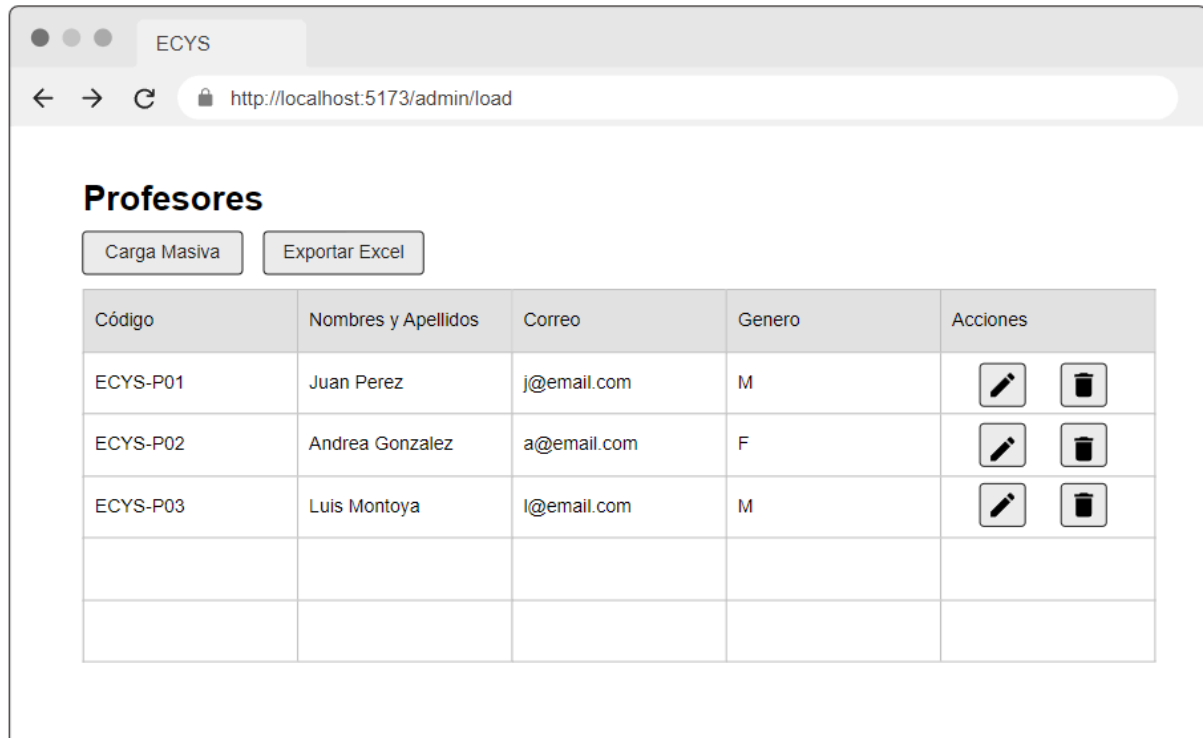
- Para poder ingresar como administrador se tendrá un usuario único con el código **“admin”** y la contraseña **“admin”**.
- Este es un sistema cerrado por lo que no hay un registro de usuarios, sino que el administrador debe crear a los estudiantes y profesores una cuenta para iniciar sesión.
- En caso de que alguno de los datos sea erróneo, se debe mostrar un error.
- Cada vista dentro del sistema debe tener la opción de *cerrar sesión*, la cual regresará a esta primera ventana de login.







Módulo de Administración

usar un navbar para poder navegadr en profesor y estudiante

Profesores

En este módulo se encargará el administrador de gestionar los profesores dentro de la plataforma.



Código	Nombres y Apellidos	Correo	Genero	Acciones
ECYS-P01	Juan Perez	j@email.com	M	 
ECYS-P02	Andrea Gonzalez	a@email.com	F	 
ECYS-P03	Luis Montoya	l@email.com	M	 

Este módulo contará con las siguientes funcionalidades:

- **Carga Masiva:** Para cargar los profesores es necesario cargar un JSON donde contendrá el siguiente formato:

Creo que es un solo json

```
[
  {
    "codigo": "ECYS-P01",
    "nombre": "Juan Pérez",
    "correo": "j@gmail.com",
    "genero": "m",
    "contrasenia": "1234"
  },
  {
    "codigo": "ECYS-P02",
    "nombre": "Andrea Gonzalez",
    "correo": "a@gmail.com",
    "genero": "f",
    "contrasenia": "1234"
  }
]
```

En esta debe de obtener el email, ya que el email es unico y el codigo

- **Exportar Excel:** El administrador es capaz de exportar un archivo de tipo excel donde mostrará todos los datos de los profesores, así como el siguiente ejemplo:

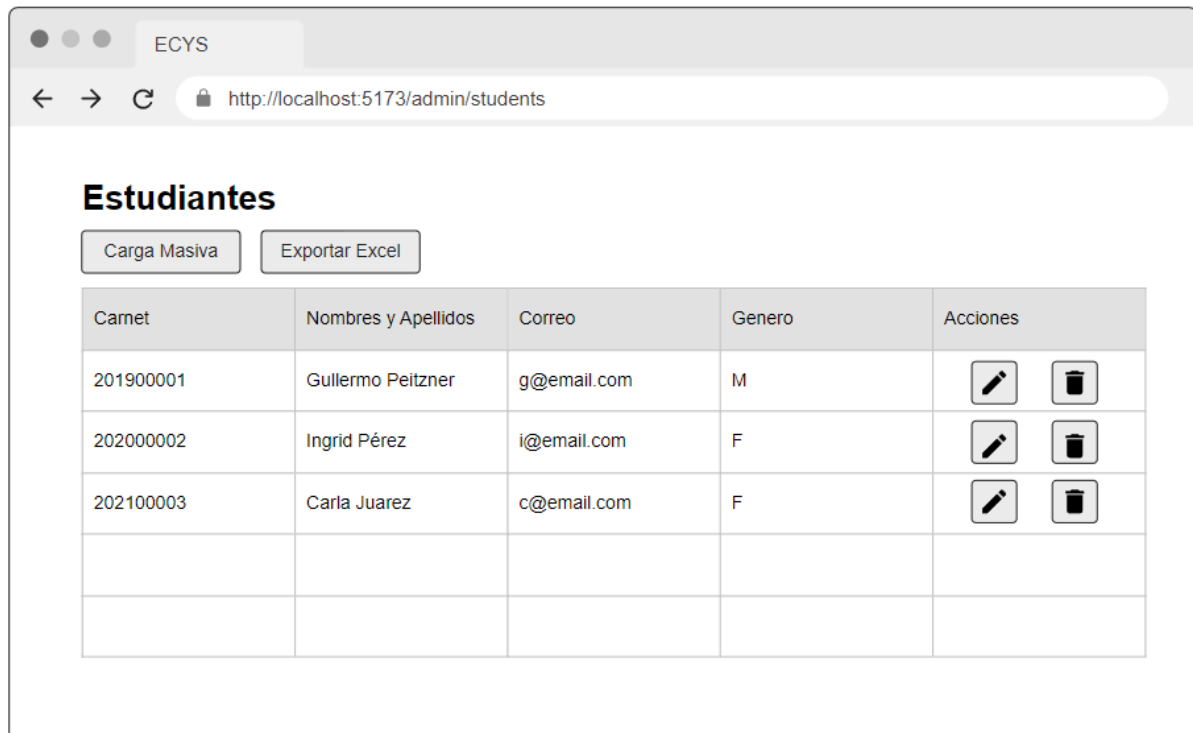
	A	B	C	D	E
1	codigo	nombre	correo	genero	contrasenia
2	ECYS-P01	Juan Pérez	j@email.com	m	1234
3	ECYS-P02	Andrea González	a@email.com	f	1234
4					
5					

Tip: Para realizar esto, puede usar la librería `xlsx` y `file-saver`

- **Editar Profesor:** Para editar un profesor, únicamente puede cambiar los siguientes atributos:
 - Nombre
 - Correo
 - Contraseña
- **Eliminar Profesor:** El botón de eliminar se encargará de desaparecer el profesor del sistema.

Estudiantes

En este módulo se encargará el administrador de gestionar los estudiantes dentro de la plataforma.



Este módulo contará con las siguientes funcionalidades:

- **Carga Masiva:** Para cargar los estudiantes es necesario cargar un JSON donde contendrá el siguiente formato:

```
[
  {
    "carnet": "201900001",
    "nombre": "Guillermo Peitzner",
    "correo": "g@gmail.com",
    "genero": "m",
    "contrasenia": "1234"
  },
  {
    "carnet": "202000002",
    "nombre": "Ingrid Perez",
    "correo": "c@gmail.com",
    "genero": "f",
    "contrasenia": "1234"
  }
]
```

lo mismo que profesor

- **Exportar Excel:** El administrador es capaz de exportar un archivo de tipo excel donde mostrará todos los datos de los estudiantes, así como el siguiente ejemplo:

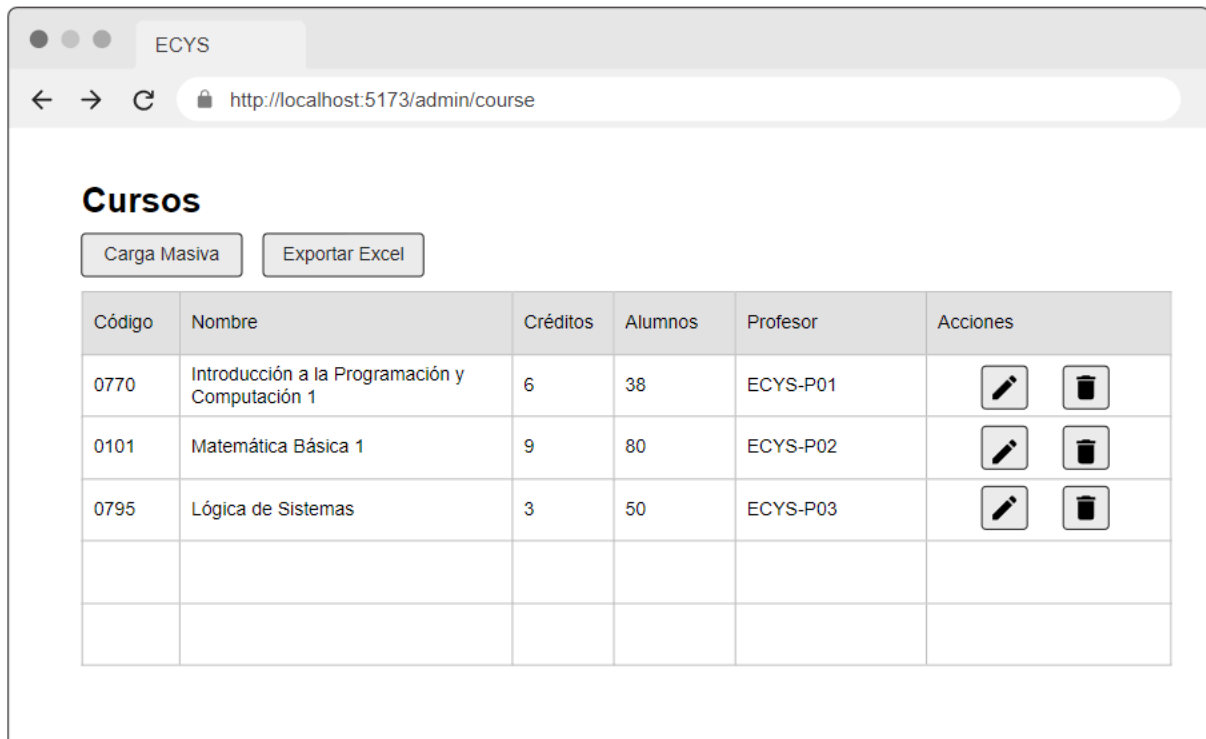
	A	B	C	D	E
1	carnet	nombre	correo	genero	contrasenia
2	201900001	Guillermo Peitzner	g@email.com	m	1234
3	202000002	Ingrid Perez	i@email.com	f	1234
4					
5					







Tip: Para realizar esto, puede usar la librería `xlsx` y `file-saver`

- **Editar Estudiante:** Para editar un estudiante, únicamente puede cambiar los siguientes atributos:
 - Nombre
 - Correo
 - Contraseña
- **Eliminar Estudiante:** El botón de eliminar se encargará de desaparecer el estudiante del sistema.

Cursos

En este módulo se encargará el administrador de gestionar los cursos de la carrera de ECYS dentro de la plataforma.



Código	Nombre	Créditos	Alumnos	Profesor	Acciones
0770	Introducción a la Programación y Computación 1	6	38	ECYS-P01	 
0101	Matemática Básica 1	9	80	ECYS-P02	 
0795	Lógica de Sistemas	3	50	ECYS-P03	 

Este módulo contará con las siguientes funcionalidades:

- **Carga Masiva:** Para cargar los estudiantes es necesario cargar un JSON donde contendrá el siguiente formato:

```
[
  {
    "codigo":0770,
    "nombre":"Introducción a la Programación y Computación 1",
    "creditos":6,
    "profesor":"ECYS-P01"
  },
  {
    "codigo":0101,
    "nombre":"Matemática Básica 1",
    "creditos":9,
    "profesor":"ECYS-P02"
  },
]
```

Por defecto todos contendrán 0 alumnos.

- **Exportar Excel:** El administrador es capaz de exportar un archivo de tipo excel donde mostrará todos los datos de los cursos, así como el siguiente ejemplo:

	A	B	C	D	E
1	codigo	nombre	creditos	profesor	alumnos
2	770	Introducción a la Programación y Computación 1	6	ECYS-P01	38
3	101	Matemática Básica 1	9	ECYS-P02	80
4					
5					

Tip: Para realizar esto, puede usar la librería `xlsx` y `file-saver`

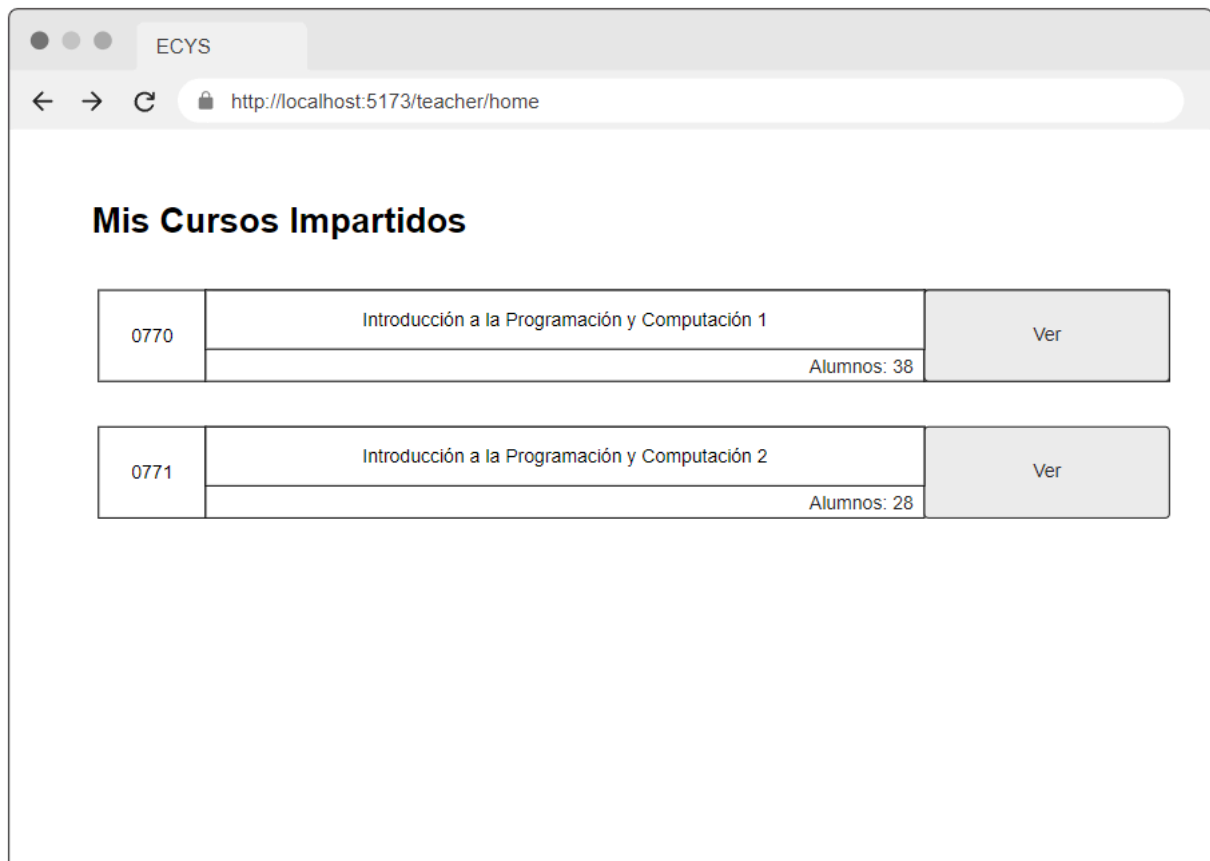
- **Editar Curso:** Para editar un curso, únicamente puede cambiar los siguientes atributos:
 - Nombre
 - Créditos
 - Profesor
- **Eliminar Curso:** El botón de eliminar se encargará de desaparecer el curso del sistema.

Módulo de Profesores

Este será el módulo que se mostrará al profesor cuando inicie sesión en el sistema. En este apartado se mostrarán los cursos y se podrán cargar los alumnos a los cursos asignados.

Inicio

En este apartado el profesor podrá visualizar todos los cursos que imparte. Por cada curso se tiene que mostrar el código y nombre del curso, además de la cantidad de alumnos, junto con un botón de “Ver”, para entrar a visualizar el curso.

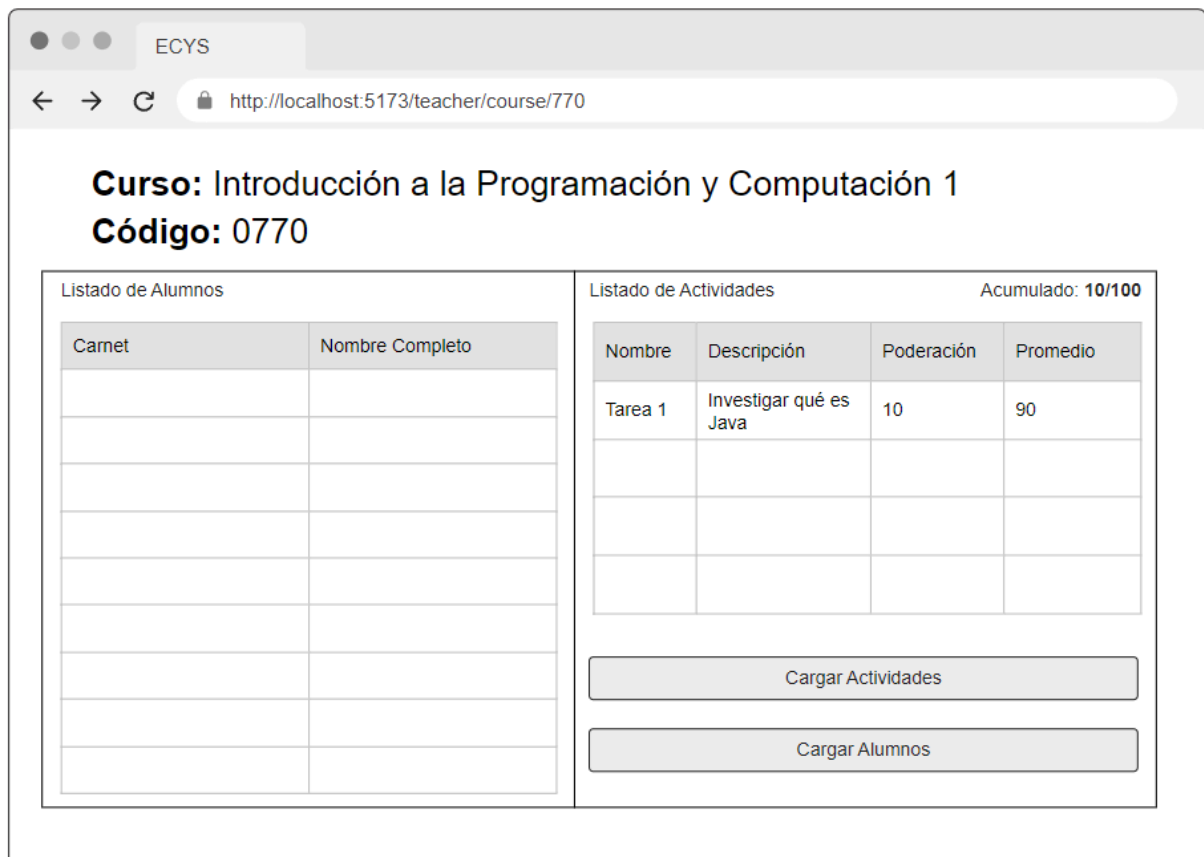


The screenshot shows a web browser window with the title 'ECYS' and the URL 'http://localhost:5173/teacher/home'. The page displays a section titled 'Mis Cursos Impartidos' (My Courses Taught). Below this title, there is a table with two rows of course information. Each row contains a course code, the course name, the number of students, and a 'Ver' (View) button.

Código	Nombre del Curso	Alumnos	Acción
0770	Introducción a la Programación y Computación 1	Alumnos: 38	Ver
0771	Introducción a la Programación y Computación 2	Alumnos: 28	Ver

Administración del curso

En este apartado podrá gestionar cada curso que imparte el profesor.



The screenshot shows a web browser window with the address bar displaying 'http://localhost:5173/teacher/course/770'. The page title is 'Curso: Introducción a la Programación y Computación 1' with code 'Código: 0770'. Below this, there are two main sections: 'Listado de Alumnos' and 'Listado de Actividades'. The 'Listado de Alumnos' section contains a table with columns 'Carnet' and 'Nombre Completo', which is currently empty. The 'Listado de Actividades' section contains a table with columns 'Nombre', 'Descripción', 'Poderación', and 'Promedio'. It shows one activity: 'Tarea 1' with description 'Investigar qué es Java', a weight of 10, and an average of 90. Below the activity table are two buttons: 'Cargar Actividades' and 'Cargar Alumnos'. The total accumulated score is shown as 'Acumulado: 10/100'.

Listado de Alumnos	
Carnet	Nombre Completo

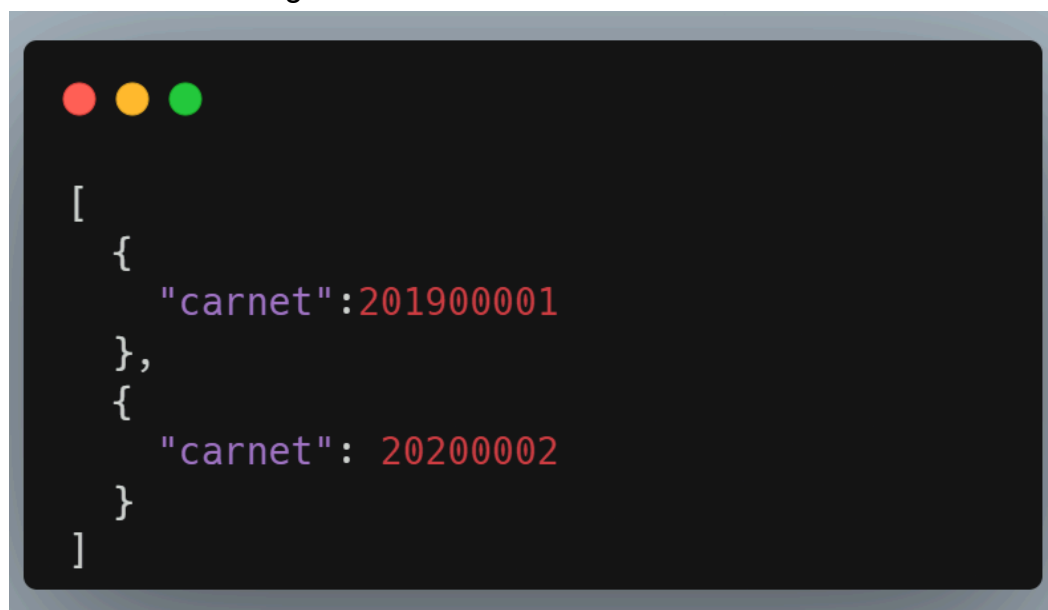
Listado de Actividades		Acumulado: 10/100	
Nombre	Descripción	Poderación	Promedio
Tarea 1	Investigar qué es Java	10	90

Cargar Actividades

Cargar Alumnos

Este módulo contará con las siguientes funcionalidades:

- **Cargar Alumnos:** Para cargar los estudiantes es necesario cargar un JSON donde contendrá el siguiente formato:



```
[
  {
    "carnet": 201900001
  },
  {
    "carnet": 20200002
  }
]
```

ese es aparte de estudiantes

- **Cargar Actividades:** El profesor es capaz de cargar las actividades del curso, el cuál se cargará un archivo JSON donde contendrá el siguiente formato:

```
[
  {
    "nombre": "Tarea 1",
    "descripción": "Investigación ¿Qué es Java?",
    "ponderación": 10,
    "profesor": "ECYS-P01",
    "notas": [
      {
        "carnet": 201900001,
        "nota": 100
      },
      {
        "carnet": 20200002,
        "nota": 80
      }
    ]
  }
]
```

Acá para obtener la información es parametro profesor , creo

Se mostrará también una tabla en donde se encuentren los alumnos inscritos en el curso y con ello otra tabla que mostrará el detalle de las actividades cargadas, así mismo los puntos acumulados de las actividades sobre 100.

Reportes

En este apartado se pueden generar los siguientes reportes:

- Top 5 estudiantes con mejor rendimiento (filtrándose por curso), por medio de una gráfica de barras.
- Top 5 estudiantes con peor rendimiento (filtrándose por curso), por medio de una gráfica de barras.

Para mostrar el rendimiento, se obtendrá el número de carnet y además la nota acumulada en el curso.

Solo abran 2 tablas (las de arriba) y el de actividades solo contara a los alumnos inscritos gracias a cargar alumnos

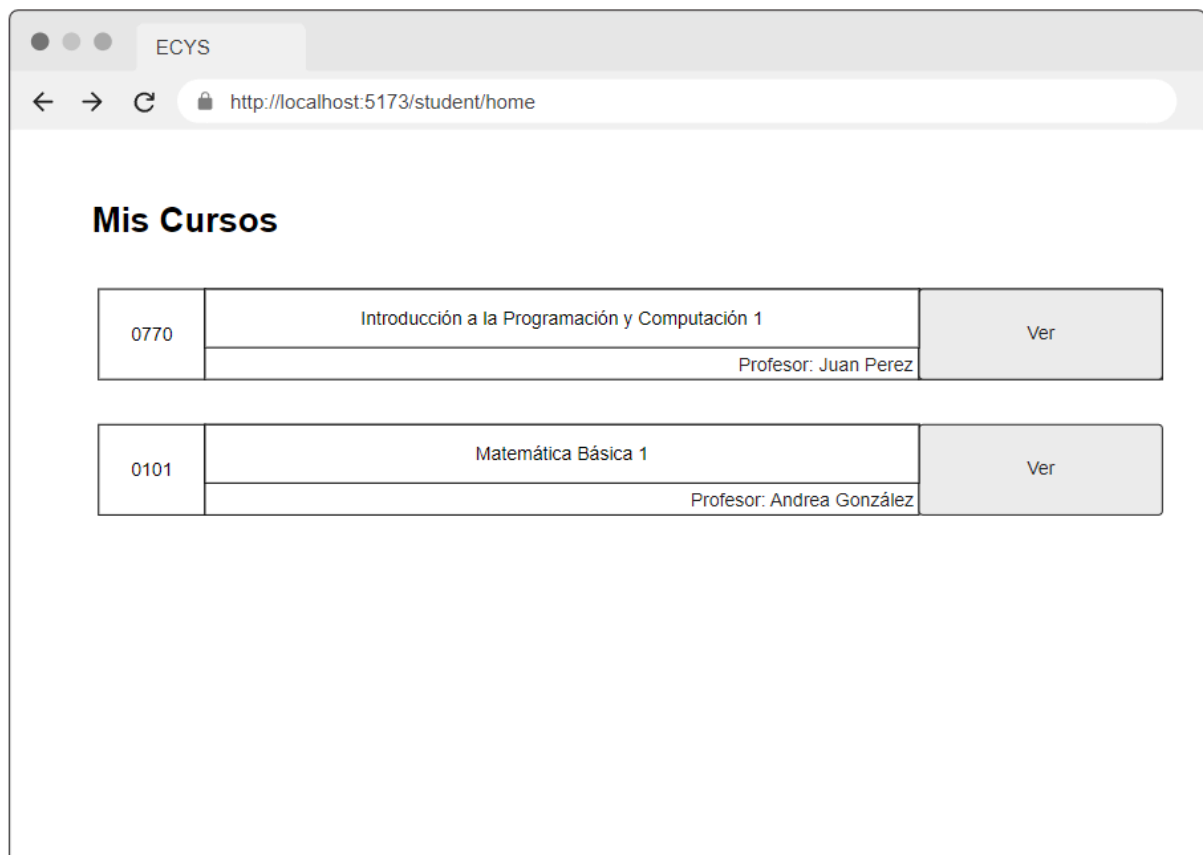
sd

Módulo de Estudiantes

Este será el módulo que se mostrará al estudiante cuando inicie sesión en el sistema. En este apartado se mostrarán los cursos y se podrá visualizar el profesor quien imparte el curso y los datos del curso.

Inicio

En este apartado el estudiante podrá ver todos los cursos a los cuáles está asignado. Por cada curso se tiene que mostrar el nombre del curso y el nombre completo del profesor que lo imparte.



Mis Cursos		
0770	Introducción a la Programación y Computación 1	Ver
	Profesor: Juan Perez	
0101	Matemática Básica 1	Ver
	Profesor: Andrea González	

Información del curso

En este apartado podrá mostrar la información de cada curso.

Curso: Introducción a la Programación y Computación 1
Código: 0770

Listado de Actividades

Nombre	Descripción	Ponderación	Nota obtenida
Tarea 1	Investigar qué es Java	10	90
	TOTAL	10	9

Exportar mis notas

Este módulo contará con las siguientes funcionalidades:

- **Exportar mis notas:** El estudiante es capaz de exportar un archivo de tipo excel donde mostrará las actividades, junto con su ponderación y su nota en la actividad, así como el siguiente ejemplo:

	A	B	C	D
1	nombre	descripcion	ponderación	nota
2	Tarea 1	Investigación ¿Qué es Java?	10	90
3	Tarea 2	Investigación ¿Qué es React?	10	100
4				
5				

Tip: Para realizar esto, puede usar la librería `xlsx` y `file-saver`

Se mostrará también una tabla en donde se encuentren las actividades y las notas de las actividades dentro del curso.

Reportes

En este apartado se pueden generar el siguiente reporte:

- Gráfica de barras de las notas totales en cada curso.

Requerimientos

Documentación

- Manual Técnico de la API, para esto se solicita que usted como desarrollador genere su documentación por medio de Postman, subiendo un archivo de texto con el link que se generó de la documentación.
- Manual de Usuario en donde explique cómo se interactúa con la aplicación por lado del Frontend en PDF.
- Diagrama de Clases en PDF.

Restricciones

- El Backend debe ser desarrollado en NodeJS, utilizando la librería de ExpressJS.
- El Frontend debe ser desarrollado con la librería de ReactJS y también puede ser complementado con el uso de NextJS.
- No se permite utilizar código copiado o bajado de internet.
- El editor de texto por utilizar queda a discreción del estudiante (se recomienda el uso de Visual Studio Code).
- Las copias obtendrán nota de 0 y reporte a la Escuela de Ciencias y Sistemas.
- Durante la calificación se le solicitará al estudiante modificar el código del proyecto con el objetivo de validar la creación de este.
- Su repositorio debe estar privado.
- El estudiante no tendrá derecho a calificación si no muestra su aplicación Frontend y Backend, no se calificará únicamente Backend o únicamente Frontend.
- El estudiante no tendrá derecho a calificación si no manejó su código dentro de su repositorio privado de github.
- El estudiante no tendrá derecho a calificación si no muestra 2 copias de la hoja de calificación impresas según la fecha y hora escogida.
- Tiene que contar con mínimo 2 commits semanales y tener agregado al auxiliar de colaborador del repositorio para monitorear los avances realizados en el proyecto, de lo contrario obtendrá una penalización.

Habilidades por evaluar

- Interacción con objetos en entornos de API.
- Manejo de propiedades en componentes gráficos web.
- Uso de peticiones HTTP.
- Conocimiento de los conceptos de backend y frontend para el desarrollo web.

Entrega

- **FECHA DE ENTREGA:** 26/10/2024 antes de las 23:59 (No se aceptarán entregas, ni commits a partir de esa fecha y hora).
- En su repositorio adjuntar código fuente y la documentación solicitada.
- El repositorio debe ser privado y en Github, teniendo el nombre de *IPC1_Proyecto2_carnet*, por ejemplo: *IPC1_Proyecto2_202300000*.
- Subir el enlace del repositorio de Github en la tarea asignada en UEDi.
- Agregar de colaborador del repositorio al auxiliar correspondiente a su sección:

Sección	Auxiliar	Usuario de github
A	Rodrigo Alejandro Hernández de León	rodrialeipc
B	Josué Rodolfo Morales Castillo	RMorales202010033
C	David Augusto Maldonado Hurtarte	DavidMaldo02
D	Esteban Humberto Valdez Ennati	ennoxSG
E	Douglas Alexander Soch Catalán	DAlexSC
F	Ayeser Cristián OxlaJ Juárez	Ayeser-Cristian
G	Federico David Zet Pajoc	fede2510