

Universidad Don Bosco



“Perfil del proyecto”

Asignatura:

Diseño y Programación de Software Multiplataforma

Docente:

Ing. Karens Medrano

Integrantes:

- Cristian Ricardo González González GG222955
- Dania Gorety Tejada Campos TC232020
- Franly Azael Marroquín Carmona MC230204
- Katherine Yamileth Flores Ávila FA231908
- Miguel Ángel Herrera HA160257

Fecha de entrega:

Domingo 11 febrero de 2024

Índice

PERFIL DEL PROYECTO	4
1. <i>Nombre del Proyecto</i>	4
2. <i>Problema para resolver</i>	4
3. <i>Objetivo general</i>	4
4. <i>Metas</i>	4
5. <i>Resultados esperados</i>	5
6. <i>Metodología</i>	5
7. <i>Área geográfica y Beneficiarios del proyecto</i>	5
Área geográfica:	5
Beneficiarios del proyecto:	6
8. <i>Equipo Participante</i>	6
9. <i>Posibles Fuentes de Financiamiento</i>	7
10. <i>Factores críticos de éxito</i>	7
11. <i>Duración</i>	7
12. <i>Presupuesto</i>	7
Mokups	8
Inicio de sesión	8
Rol de Docente	8
Rol de Alumno	9
Rol de Administrador	10
1. <i>Paleta de Colores</i>	12
2. <i>Iconografía</i>	12
3. <i>Tipografías</i>	12
Inicio de sesión: Rigtheous	12
• <i>Mockups apartado Docente: Inter, Roboto, Bold</i>	12
• <i>Mockups apartado Alumno: Rigtheous, Inter</i>	12
• <i>Mockups apartado Administrador: Rigtheous, Inter</i>	12
4. <i>Imágenes</i>	12
5. <i>Navegación Web</i>	13
6. <i>Diseño de Pantallas</i>	13
Diagrama UML	14
Clase Docente:	14
Clase Alumno:	14

Clase Administrador:	14
Clase Materia:	14
Clase Tarea:	14
<i>Diagrama grafico de Arquitectura de Software</i>	<i>16</i>
Lógica de Negocio:	16
Lenguaje de programación:.....	16
Frameworks Beckend:.....	16
Base de Datos:.....	16
Seguridad:	17
Gestión de Sesiones y Autorización:.....	17
Gestión de Configuración:	17
<i>herramientas para utilizar Para el desarrollo de la aplicación</i>	<i>17</i>
Visual estudio code	17
Android Estudio	17
Trello	18
Figma	18
Draw.io	18
React Native.....	18
GitHub	18
ExpoSnack	18
<i>Cronograma</i>	<i>19</i>
<i>Fuentes de consulta</i>	<i>20</i>
<i>Links:</i>	<i>21</i>
GitHub.....	21
Mockups :.....	21
Trello:.....	21

PERFIL DEL PROYECTO

1. Nombre del Proyecto

- *Sistema de control estudiantil*

Es una propuesta para desarrollar una página web que ayudara a administrar un buen manejo de información estudiantil realizando y organizando tareas como entrega de calificaciones, facilitando la asistencia y la toma de decisiones sobre el seguimiento académico

2. Problema para resolver

La ausencia de un sistema de control estudiantil podría generar un problema de planificación académica tanto como en los estudiantes, profesores y administradores de la institución además puede resultar en una falta de datos precisos sobre la participación de los estudiantes, la transmisión de información importante y colaborativa y la dificultad de una buena organización.

- **¿Cómo planeamos solucionar?**

Desarrollando un sistema de control estudiantil bien implementado y dando eficiencia en cuanto la mejoría del entorno educativo ayudando a abordar estos problemas y optimizar la gestión académica además podríamos reducir el riesgo de pérdida de datos y garantizar mejoría hacia la institución.

3. Objetivo general

Elaborar, desarrollar e implementar un sistema moderno para gestionar la información académica. Nuestro propósito al realizar este sistema es mejorar la administración escolar, fomentar una comunicación más efectiva entre todos los involucrados y facilitar la toma de decisiones informadas en la institución educativa. Este sistema cubrirá áreas como el monitoreo de estudiantes, la gestión de calificaciones y la asistencia, contribuyendo al progreso general de los procesos educativos.

4. Meta

s

- Aprovechamiento de las funcionalidades del sistema y que puedan interactuar eficientemente.
- Implementar tecnologías de seguimiento de asistencia que logren una precisión proporcionando datos fiables para evaluar la participación estudiantil.
- Reducir los errores y optimizando el tiempo dedicado por los docentes a esta tarea.
- Identificar posibles áreas de mejora para mantener la eficiencia del sistema.
- Integrar el sistema con plataformas educativas externas para compartir información de manera eficiente, como la transferencia segura de expedientes académicos o la colaboración en proyectos interinstitucionales.
- Respaldo de datos con el fin de asegurar la protección y accesibilidad de la información académica en situaciones de fallos técnicos o pérdida accidental.

5. Resultados esperados

El proyecto llamado “**Sistema de Control Estudiantil**” tiene como objetivo buscar una serie de beneficios y resultados óptimos para mejorar la gestión académica en instituciones educativas, algunos resultados esperados serian:

- **Eficiencias en la Administración Escolar:** mediante la automatización de tarea como las entregas de notas en su determinado tiempo y el seguimiento de la asistencia se espera una eficiencia en la administración escolar
- **Comunicación Efectiva:** la implementación de una herramienta de comunicación interna facilitara la relación entre estudiantes, Docentes y administradores, promoviendo un ambiente más dinámico y fácil
- **Reducción de errores y optimización de tiempo:** al simplificar y mejorar algunos puntos sobre la gestión de información académica, se espera reducir los errores y optimizar tiempo que utilizaban los docentes para hacer dicha tarea y así permitiéndoles enfocarse en otras actividades educativas más eficaces.

Mediante la recopilación y análisis de datos se podrá identificar posible área de mejoramiento en los procesos educativos y así poder garantizar el funcionamiento del sistema eficiente a largo plazo.

6. Metodología

Para desarrollar el proyecto “**sistema de control Estudiantil**” se utilizarán metodología que nos permitirán la gestión eficiente y avances en nuestro proyecto y se propone utilizar un enfoque basado en lo aprendido tiempo atrás y lo que estaremos aprendiendo en la materia actual que es Diseño y Programación de Software Multiplataforma. Algunos métodos por utilizar serian:

- **Herramienta trello:** se utilizar para determinar el tiempo utilizado en cada punto del trabajo y poder organizar los avances del proyecto para poner garantizar una entrega completa y satisfactoria
- **Reuniones programadas:** se llevará a cabo reuniones en un determinado tiempo para discutir dudas, proceso, etc. sobre el avance que se llevará en eso momentos. y también seguir planificando cada vez más los puntos para poder concluir el proyecto.
- **Almacenamiento sobre el desarrollo del proyecto:** como equipo de trabajo usaremos la plataforma Git Hub para poner ir almacenando los procesos que se irán haciendo en nuestro proyecto con el fin de poner garantizar que todo el equipo tengamos los mismos avances y no generemos diversas versiones erróneas.

7. Área geográfica y Beneficiarios del proyecto

Área geográfica:

el proyecto “sistema de control estudiantil” se llevará a cabo en instituciones educativa ubicadas en el área urbana y rural. estas instituciones pueden varias en tamaño y nivel educativo. Se

priorizará iniciar en escuela pilotos dentro de la zona más factible para todos los integrantes del equipo que trabajaremos en el proyecto.

Beneficiarios del proyecto:

El proyecto tiene como objetivo beneficiar a diversos grupos de la comunidad educativa incluyendo:

- **Estudiantes:** Tendrá acceso a una plataforma que le permitirá consultar sus calificaciones y revisar su historial académico y mantenerse informado sobre evento importantes
- **Docentes:** Podrán gestionar las entregas de notas más rápido y podrá llevar las asistencias de los alumnos más controlada y sobre todo se tendrá una comunicación con los alumnos sobre algunos puntos, eventos, etc próximos a tener.
- **Administradores escolares:** contarán con la herramienta para monitorear el rendimiento académico de los estudiantes, identificar las tendencias y así tomar decisiones informadas para mejorar el proceso educativo.
- **Padres de Familia:** Podrán mantenerse informados sobre el progreso académico de sus hijos y participar de manera más activa en su educación, recibiendo actualizaciones periódicas sobre calificaciones, asistencia y eventos escolares.

En resumen, el proyecto "**Sistema de Control Estudiantil**" buscará mejorar la experiencia educativa para todos los involucrados, promoviendo una comunicación efectiva, una gestión académica eficiente y una toma de decisiones informadas en las instituciones educativas.

8. Equipo Participante

El éxito del proyecto "**Sistema de Control Estudiantil**" dependerá en gran medida del compromiso y la colaboración de un equipo que lleva como objetivo beneficiar la comunidad educativa. Actores clave que formarán parte del equipo participante:

- **Los Desarrolladores de Software:** Serán los responsables de diseñar, desarrollar e implementar la plataforma web del proyecto sistema de control estudiantil. Teniendo la experiencia en tecnologías web como HTML, CSS, JavaScript y frameworks como React.js para el frontend, así como Node.js para el backend.
- **Coordinador del proyecto:** Tendrá como objetivo gestionar todas las actividades relacionadas con el proyecto, incluyendo la planificación de cada punto, asignación de recursos y seguimiento del progreso en el proyecto.
- **Usuarios Finales** (Profesores, Estudiantes, Administradores Escolares): Serán parte integral del equipo participante, ya que proporcionarán comentarios y retroalimentación clave durante la ejecución del proyecto. Su participación nos garantizará que el sistema satisfaga adecuadamente sus necesidades y requisitos.

9. Posibles Fuentes de Financiamiento

Dado que el proyecto en cuestión tiene por objetivo implementar un sistema de gestión estudiantil en una institución educativa, nuestra meta consiste en obtener financiamiento gubernamental para adquirir los recursos necesarios destinados al desarrollo del proyecto. Es fundamental contar con servidores de alto rendimiento, dada la exigencia de capacidad de procesamiento requerida para manejar eficientemente la información del programa. La complejidad del proyecto demanda infraestructura informática de última generación, ya que su ejecución no es factible en equipos convencionales.

10. Factores críticos de éxito

Dentro del marco del proyecto "**Control Estudiantes**", se identifican diversos riesgos potenciales durante su ejecución, entre los cuales se destaca el mal funcionamiento de alguna de las unidades de cómputo designadas para la realización de las tareas. Ante esta contingencia, cada miembro del equipo de trabajo dispondrá de una estación de trabajo individual, garantizando así la continuidad de las labores asignadas en caso de fallo de alguna de las máquinas. Esta medida de contingencia asegura la integridad y disponibilidad del trabajo completo para cada integrante, mitigando el riesgo de pérdida del proyecto. Asimismo, se implementará un sistema de versionamiento utilizando la plataforma GitHub, donde se irán guardando de manera progresiva y sistemática los avances del trabajo. Esta práctica salvaguarda los datos y el progreso del proyecto, incluso en el improbable escenario de que todas las unidades de cómputo experimenten fallos simultáneos.

11. Duración

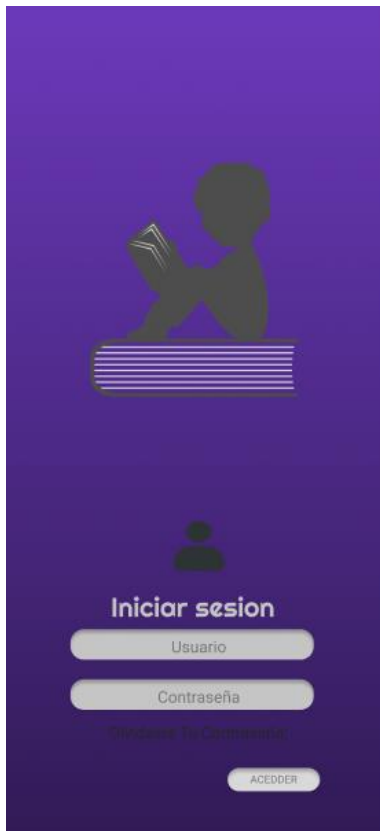
El proyecto "**Control Estudiantes**" está programado para una duración de 5 meses, desde febrero hasta junio, con inicio el domingo 11 de febrero y culminación en la fecha de presentación y defensa el 26 de mayo. Dentro de este período, está prevista la entrega de la primera fase del proyecto para el 24 de marzo, mientras que la entrega final del proyecto completo está programada para el 5 de mayo y finalizando el proceso de instalación del proyecto en la institución educativa en el mes de junio.

12. Presupuesto

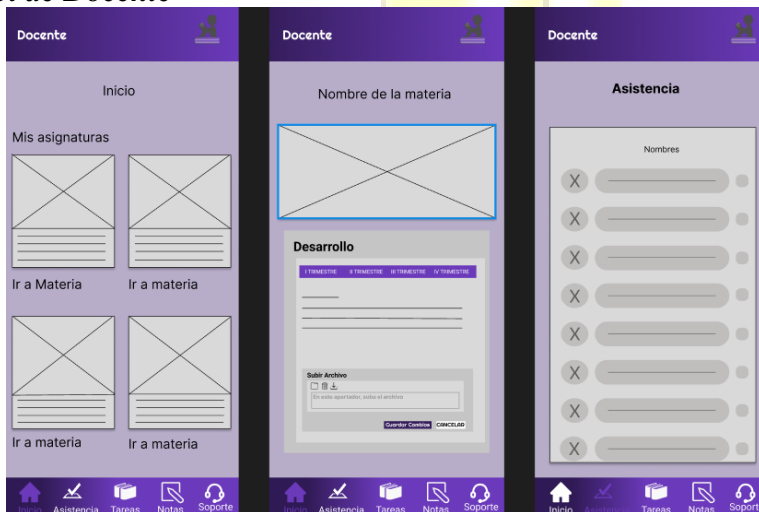
	Costo	Total
Pago mensual a cada integrante del equipo por cada mes de trabajo en la duración del proyecto (5 integrantes)	\$600.00 c/u	\$15,000
Costo Diarios (Alimentación)	\$5.00 c/u	\$3,750
Transporte	\$2.00 c/u	\$1,500
Mantenimiento de equipo		\$1,000
Total		\$21,250

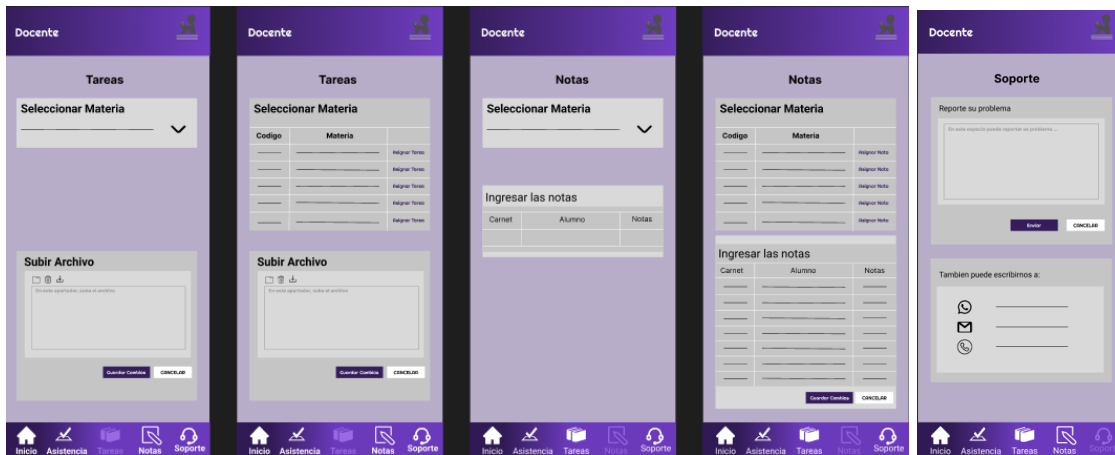
Mokups

Inicio de sesión

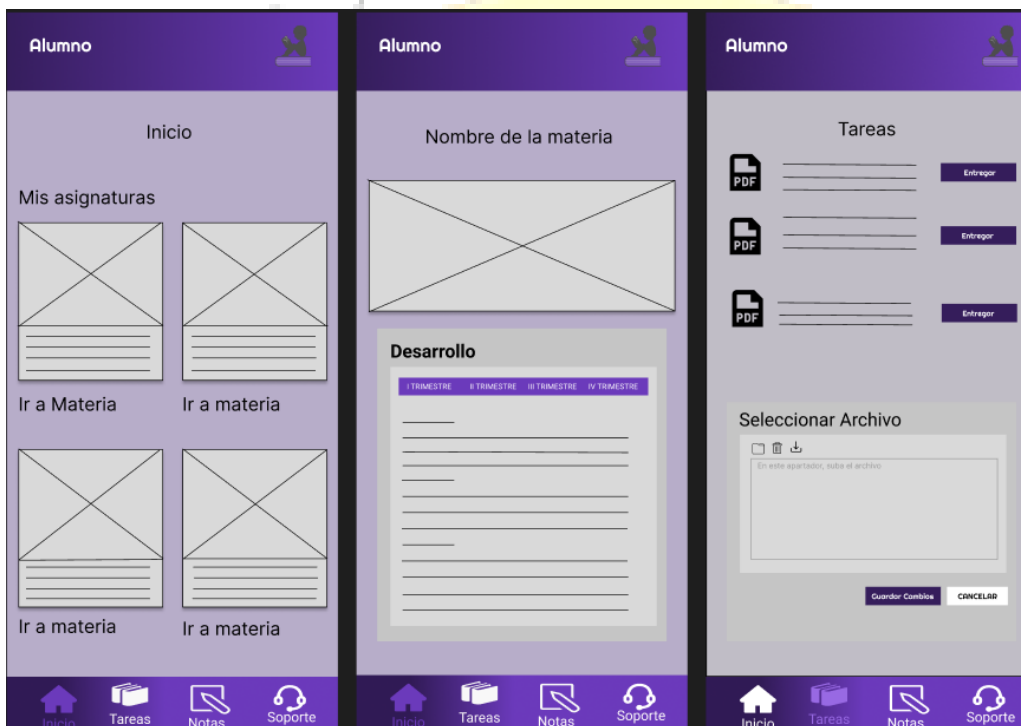


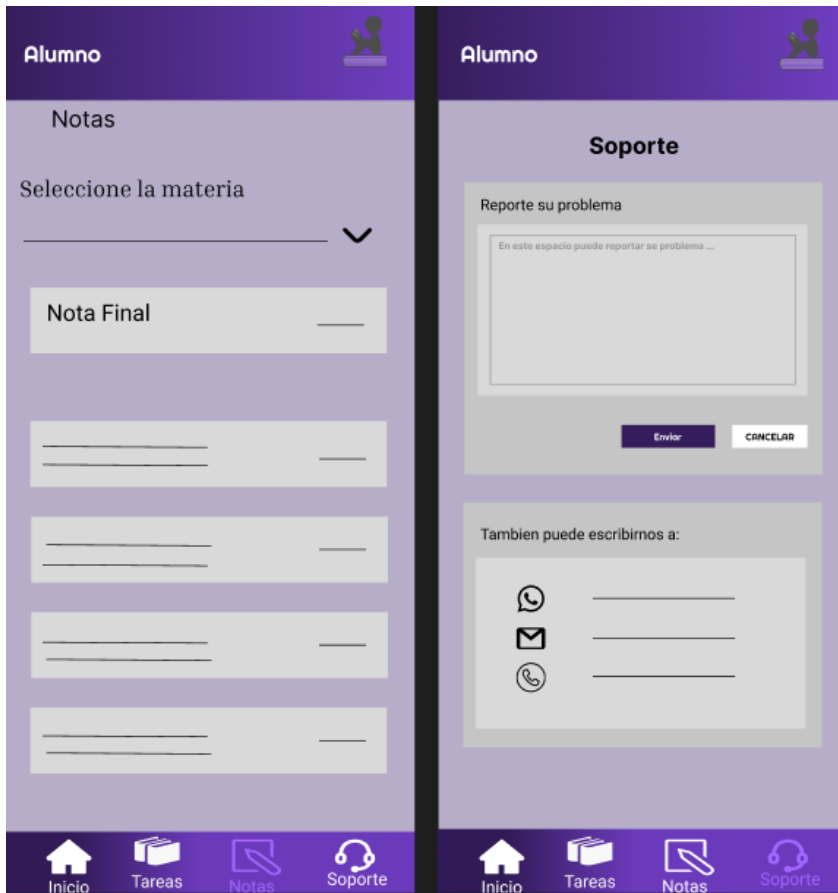
Rol de Docente



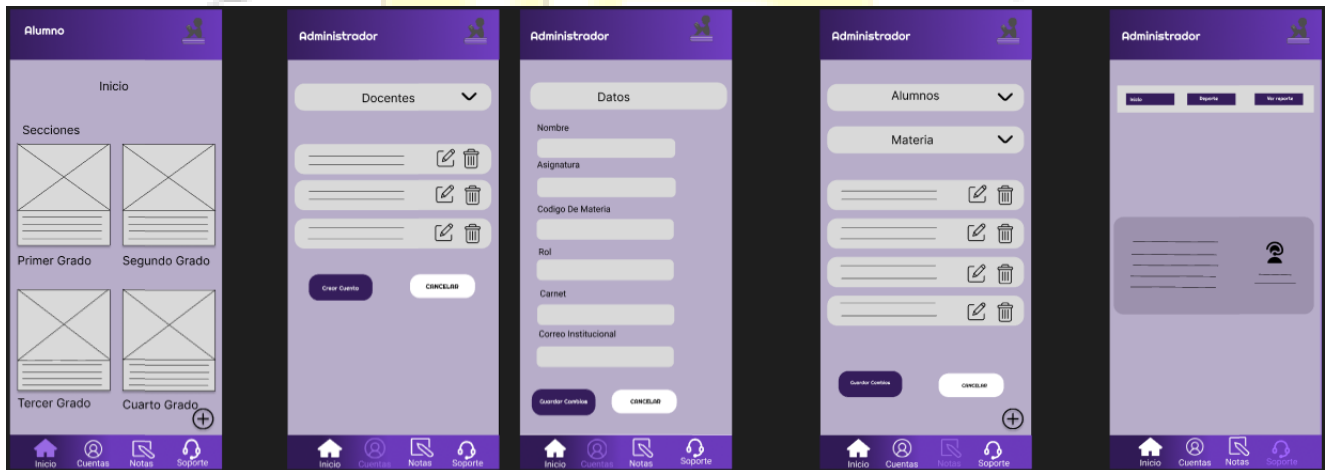


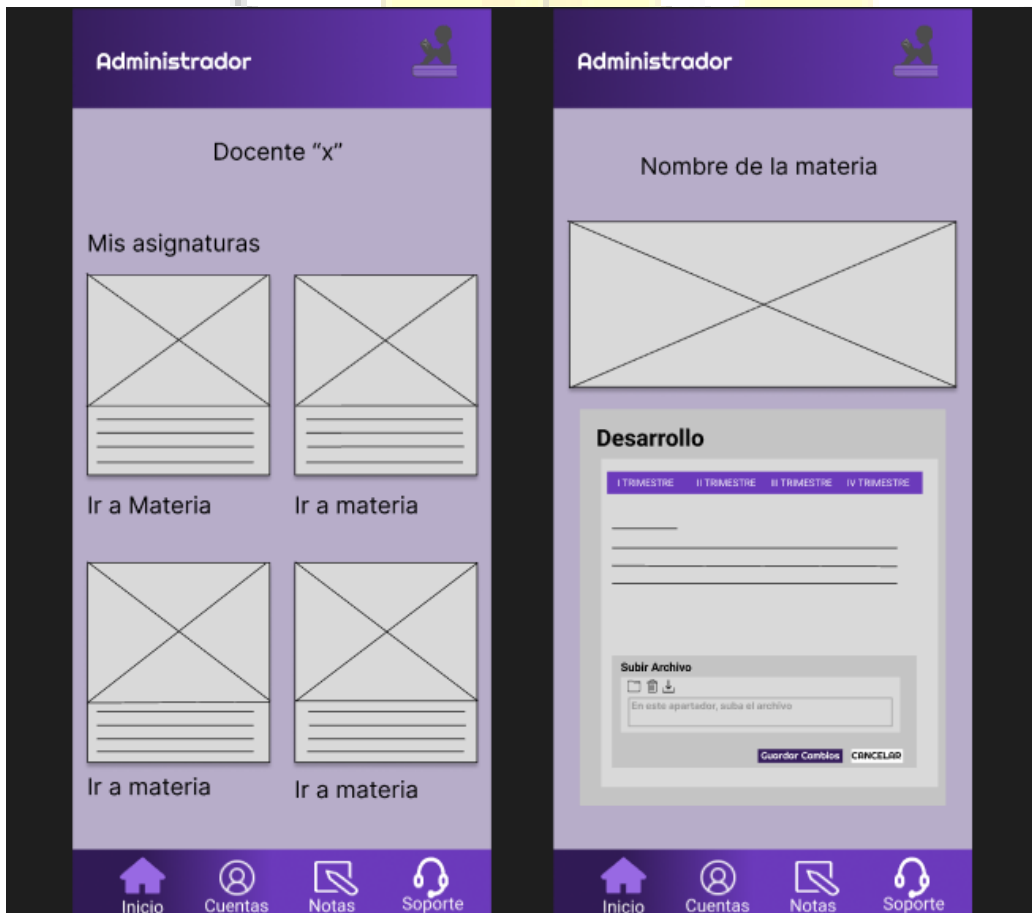
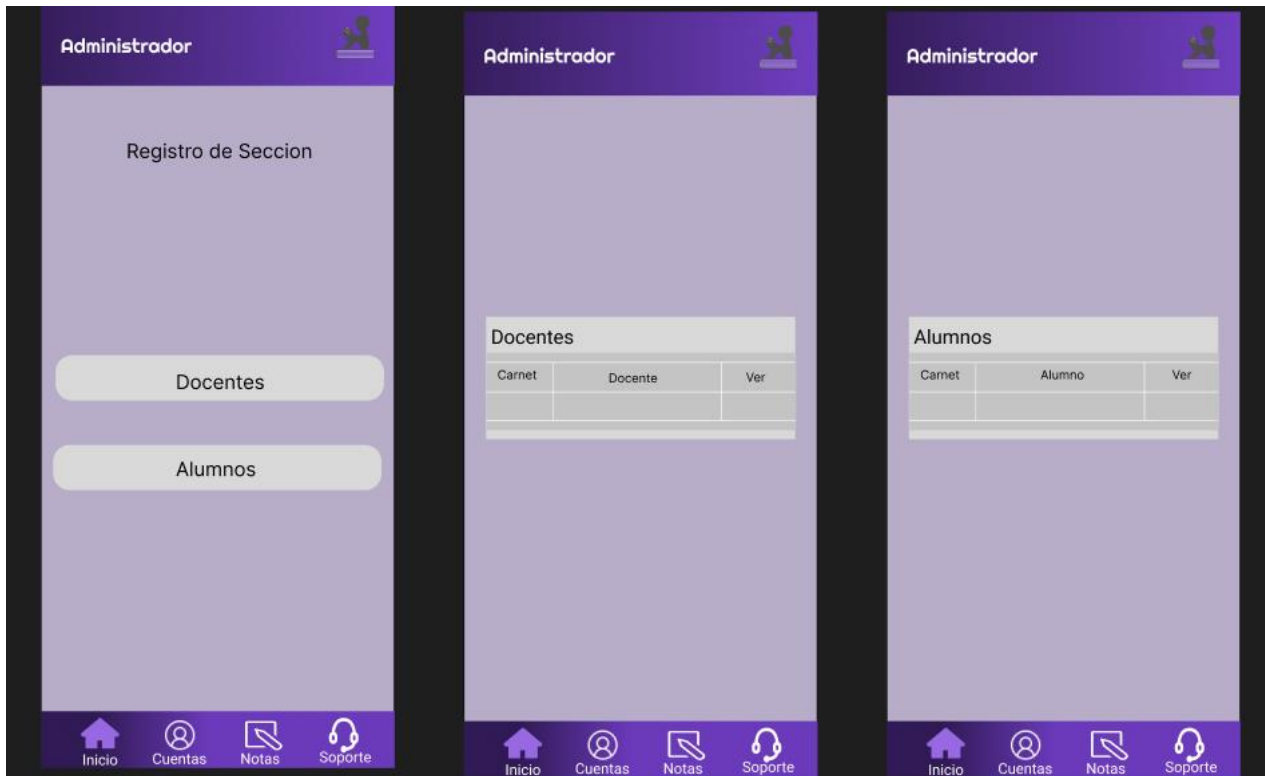
Rol de Alumno



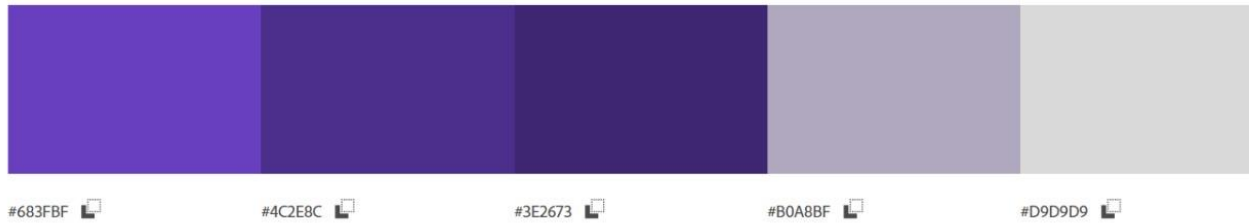


Rol de Administrador





1. Paleta de Colores:



2. Iconografía

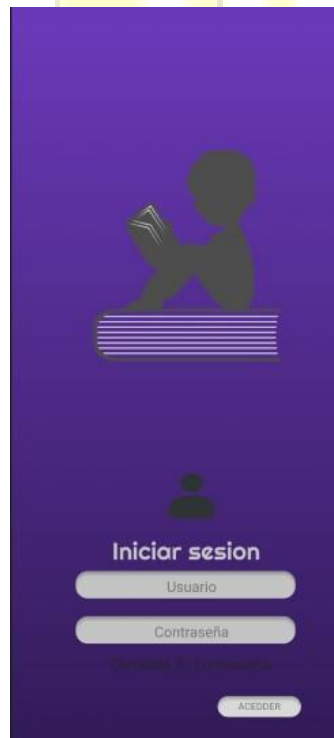


3. Tipografías

Inicio de sesión: Rigtheous

- **Mockups apartado Docente:** Inter, Roboto, Bold
- **Mockups apartado Alumno:** Rigtheous, Inter
- **Mockups apartado Administrador:** Rigtheous, Inter

4. Imágenes



Tamaño: 430x932.

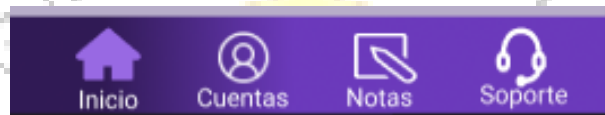


Iconos: 36 x 36

5. Navegación Web

Nuestra aplicación utiliza una Barra de Navegación Inferior para la navegación web. Esta barra contiene cuatro iconos, cada uno representando una sección principal de la aplicación.

Los usuarios pueden navegar fácilmente a través de la aplicación tocando en estos iconos. Cada icono está acompañado de un texto descriptivo para asegurar que los usuarios entiendan a qué sección conduce cada icono. Creemos que este tipo de navegación se alinea bien con los objetivos de nuestros usuarios y la estructura de nuestra aplicación.



6. Diseño de Pantallas

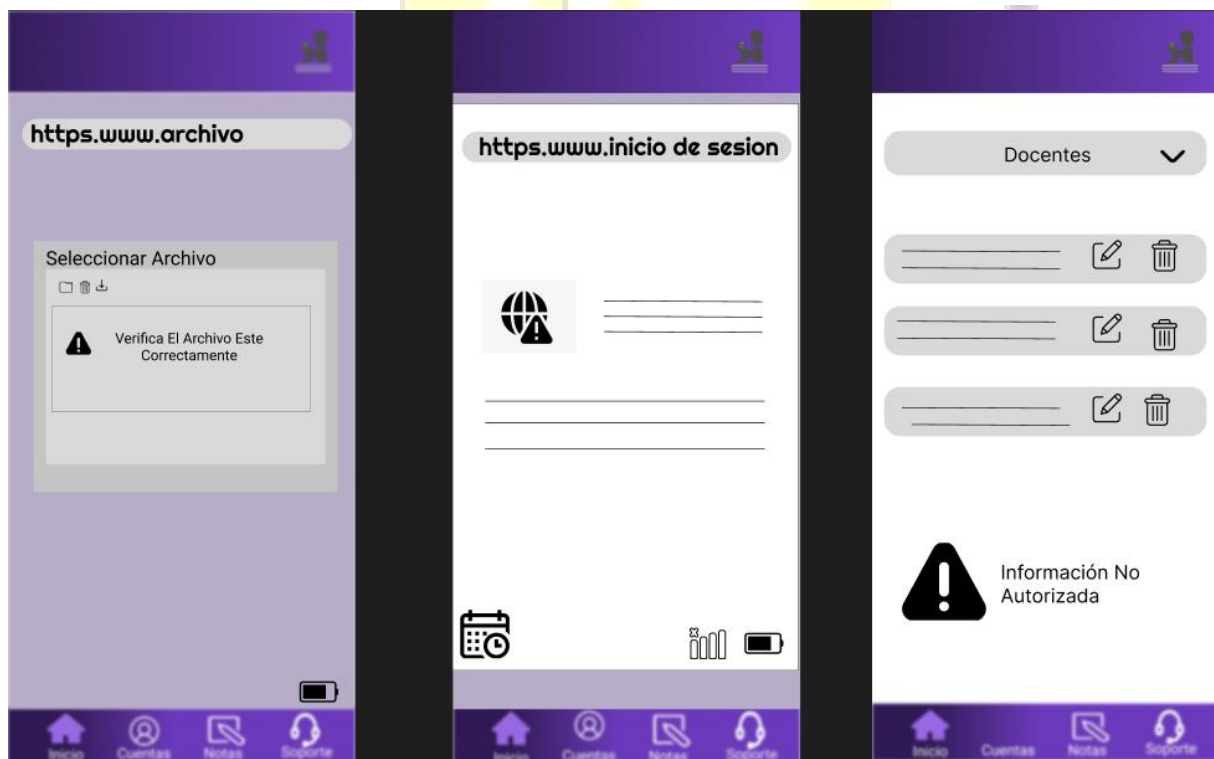


Diagrama UML

Clase Docente:

Esta clase representa a los docentes. Cada docente tiene un nombre, una lista de materia que Enseña a sus alumnos, un código de materia, un carnet y un correo institucional.

Los docentes pueden pasar asistencia, subir tareas, ingresar notas y ver un resumen la tarea que a Dejado.

Clase Alumno:

Esta clase representa a los alumnos. Cada alumno tiene un nombre, una lista de materias que está cursando y sección de notas (donde la clave es la materia y el valor es la nota). Los alumnos pueden subir tareas y ver su nota final y el detalle de sus notas.

Clase Administrador:

Esta clase representa a los administradores. Cada administrador tiene un nombre, una lista de todos los docentes y una lista de todos los alumnos. Los administradores pueden ver el contenido de cada grado, agregar, editar y eliminar docentes, agregar alumnos, ver y hacer reportes, ver registros de sesiones, y ver las materias que cada docente y alumno tiene.

Clase Materia:

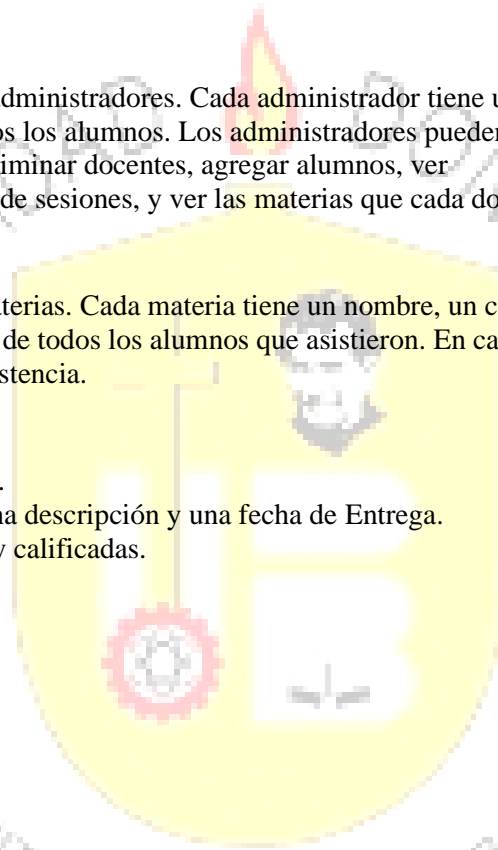
Esta clase representa las materias. Cada materia tiene un nombre, un código, una lista de todas las tareas asignadas y una lista de todos los alumnos que asistieron. En cada materia se pueden agregar tareas y verificar la asistencia.

Clase Tarea:

Esta clase representa las tareas.

Cada tarea tiene un nombre, una descripción y una fecha de Entrega.

Las tareas pueden ser subidas y calificadas.



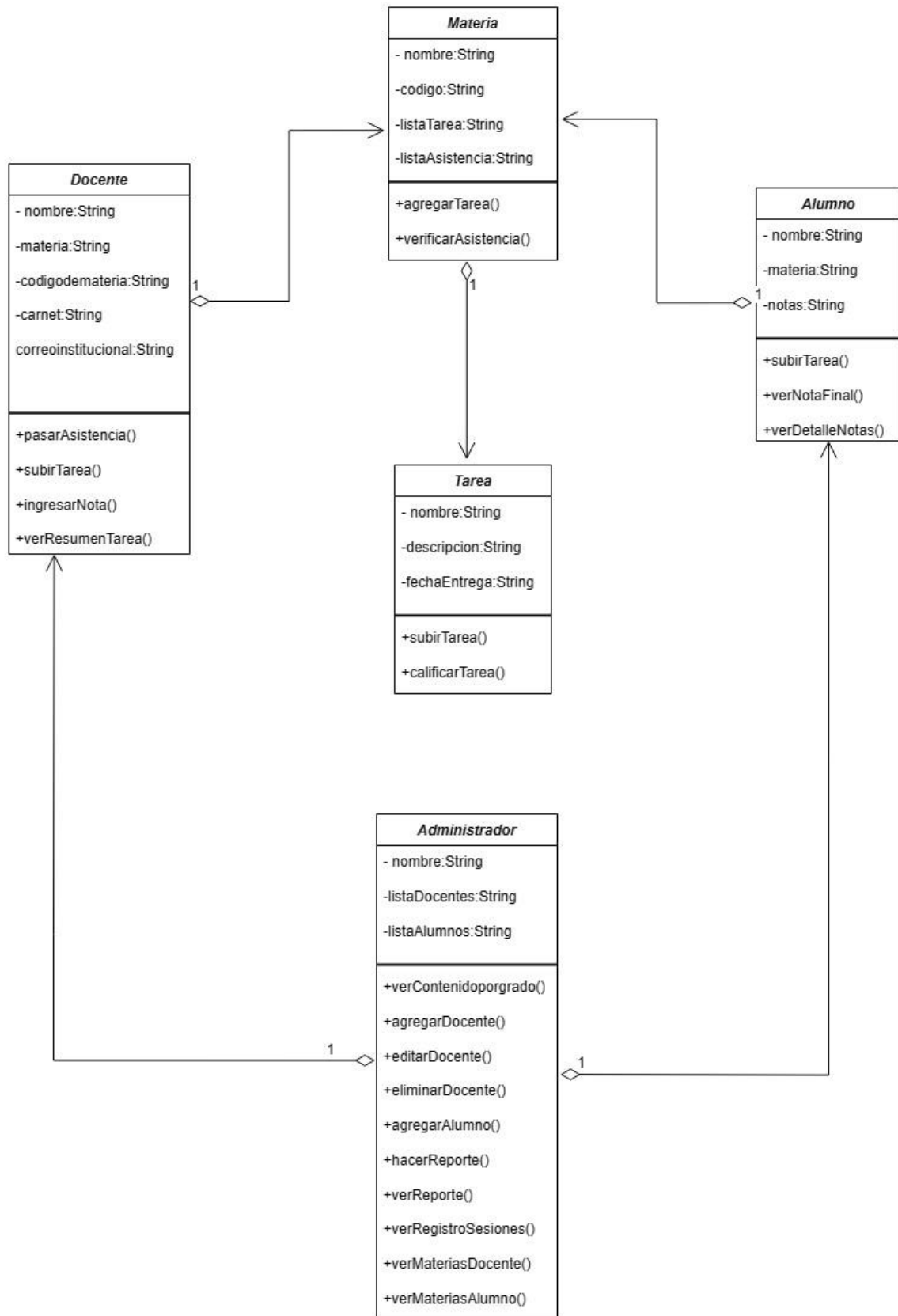
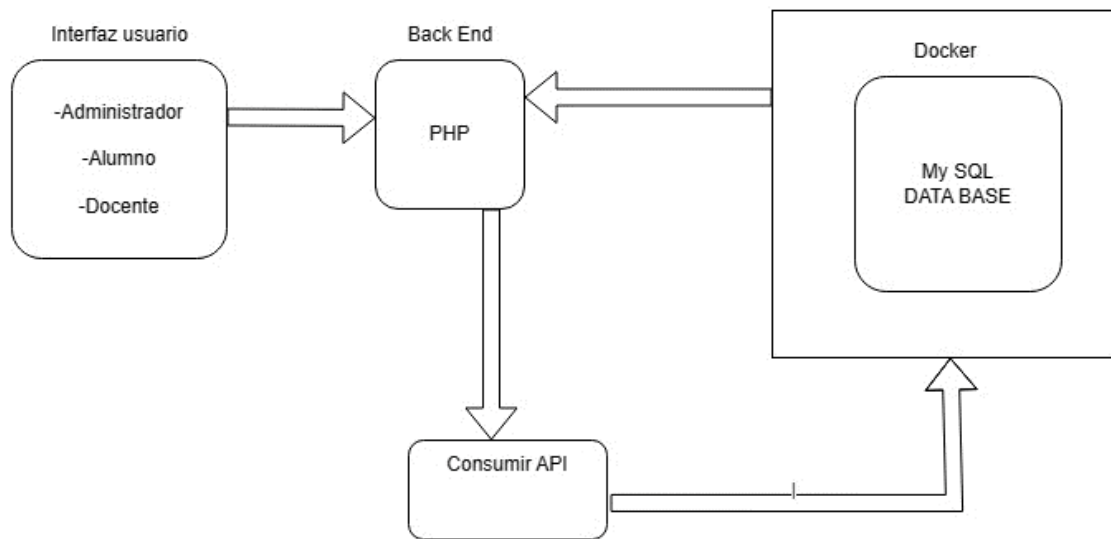


Diagrama grafico de Arquitectura de Software



Lógica de Negocio:

Servidores: Para este punto la lógica de negocio que usaremos para la comunicación con la base de datos es Mysql debido a que es ampliamente utilizado en la industria, desde pequeñas aplicaciones web hasta grandes sistemas empresariales, debido a su fiabilidad, rendimiento y facilidad de uso.

Lenguaje de programación:

El lenguaje de programación que se estará usando para este trabajo es React

Frameworks Backend:

Para este punto se estará utilizando el de frameworks de PHP por que son herramientas poderosas que ayudan a los desarrolladores a construir aplicaciones web de manera más rápida, organizada, segura y escalable, permitiéndoles concentrarse en la lógica de la aplicación en lugar de en detalles técnicos y repetitivos.

Base de Datos:

Sistema de Gestion de Base de Datos (DBMS): para este caso estaremos usando el DBMS de MySQL por que es un sistema de gestión de bases de datos relacional de código abierto muy popular, conocido por su velocidad, confiabilidad y facilidad de uso. Es ampliamente utilizado en aplicaciones web y empresariales.

Comunicación entre componentes

API (Interfaz de Programación de Aplicaciones): para este caso estaremos creando una API con javascript

Protocolos de Comunicación: Para este punto se estará utilizando el protocolo de HTTPS porque es una extensión del protocolo HTTP, pero con una capa adicional de seguridad proporcionada por SSL/TLS.

HTTPS cifra la comunicación entre el navegador y el servidor, protegiendo así la privacidad y la integridad de los datos transmitidos.

Seguridad:

Autenticación y Autorización: en este caso se hará tanto del lado del Backend para prevenir inyecciones de SQL y de datos que no estén bien almacenados

Gestión de Sesiones y Autorización:

Cookies y Tokens de Sesión: para nuestro caso en la Aplicación de Sistema de control estudiantil podemos utilizar Cookies como Tokens de sesión, pero nos inclinamos por cookies por que se pueden usar cookies para almacenar un identificador de sesión único en el navegador del usuario cuando inicia sesión en la aplicación. Este identificador se puede utilizar para recuperar la información del estudiante de la base de datos en cada solicitud subsiguiente, lo que permite mantener la sesión del usuario activa mientras navega por la aplicación.

Gestión de Estado del Cliente: para este caso al estar trabajando nosotros con React estaremos utilizando Redux ya que es especialmente útil en aplicaciones con una gran cantidad de interacciones complejas entre componentes, o en aplicaciones que necesitan compartir estado entre múltiples componentes en diferentes partes de la jerarquía de componentes. Ayuda a mantener el estado de la aplicación organizado y predecible, lo que facilita la depuración y el mantenimiento del código.

Gestión de Configuración:

Configuración del Entorno: Para la gestión de configuración de nuestro proyecto hay varias herramientas y prácticas que pueden ser útiles para asegurar un entorno de desarrollo estable y consistente, pero para este caso utilizaremos Docker permite empaquetar la aplicación y todas sus dependencias en contenedores, lo que facilita la creación de entornos de desarrollo consistentes y reproducibles.

Autorización de Implementación: para este punto usaremos Docker ya que es una plataforma de código abierto que permite a los desarrolladores crear, implementar y ejecutar aplicaciones de manera eficiente en entornos aislados llamados "contenedores". Estos contenedores son unidades estandarizadas de software que incluyen todo lo necesario para ejecutar una aplicación, como el código, las bibliotecas, las herramientas y las configuraciones del sistema.

herramientas para utilizar Para el desarrollo de la aplicación

Visual estudio code

Es una plataforma altamente versátil que nos ayuda a procesar los datos de programación de una manera más dinámica. Con su amplia gama de extensiones y características personalizables, brinda una experiencia de desarrollo eficiente y ágil, permitiendo una codificación más rápida y precisa. Su entorno de trabajo intuitivo y su soporte integrado para múltiples lenguajes de programación hacen que sea una herramienta indispensable para desarrolladores de software de todos los niveles. Además, su capacidad para integrarse con servicios en la nube y herramientas de control de versiones lo convierte en una opción ideal para equipos de desarrollo colaborativo.

Android Estudio

Es una plataforma integral que nos ayuda a interpretar el código y visualizar gráficamente el proceso de resolución de la aplicación. Con su conjunto de herramientas avanzadas, facilita el desarrollo, depuración y optimización de aplicaciones Android, mejorando la eficiencia del desarrollo y la calidad del producto

final.

Trello

Esta plataforma nos permite organizar nuestras tareas de manera eficiente, lo que nos ayuda a mantenernos ordenados y enfocados en cada una de las tareas que se asignen. Además, facilita la colaboración entre miembros del equipo, proporcionando herramientas para el seguimiento del progreso y la comunicación fluida.

Figma

Es una herramienta esencial para la creación de mockups de interfaz gráfica de usuarios. Con una amplia variedad de componentes y una interfaz intuitiva, facilita la organización y el diseño eficiente de proyectos. Su colaboración en tiempo real permite un trabajo fluido y colaborativo entre equipos, mejorando la productividad y la calidad del diseño.

Draw.io

Es una plataforma versátil que empleamos para la creación de diagramas UML, esenciales para mostrar la lógica y la estructura de nuestros programas. Con su interfaz intuitiva y sus amplias opciones de personalización, nos permite representar de manera clara y concisa los distintos aspectos de nuestro proyecto. Además, al ser una herramienta basada en la web, nos brinda la flexibilidad de trabajar en nuestros diagramas desde cualquier lugar y en cualquier momento. Su integración con otras herramientas de productividad, como Google Drive y Microsoft Teams, también facilita la colaboración y el intercambio de ideas entre miembros del equipo.

React Native

Es una potente herramienta basada en JavaScript, desempeña un papel fundamental en la creación de interfaces gráficas dinámicas (Front-End). Esta tecnología nos permite desarrollar aplicaciones móviles multiplataforma con un rendimiento excepcional y una experiencia de usuario fluida. Al combinar React Native con PHP en el lado del servidor, conseguimos optimizar el consumo de servicios y mejorar la capacidad de respuesta del cliente, lo que resulta en una interacción más rápida y eficiente para los usuarios. Además, su naturaleza modular y su amplia comunidad de desarrollo ofrecen una gran variedad de componentes y bibliotecas que facilitan la creación de interfaces de usuario sofisticadas y funcionales.

GitHub

Es una plataforma líder en alojamiento de código, es fundamental para gestionar todos los procesos relacionados con la creación de la aplicación. Nos permite organizar la información en repositorios con ramas individuales, facilitando la gestión específica del contenido de la página y asegurando un desarrollo colaborativo eficiente.

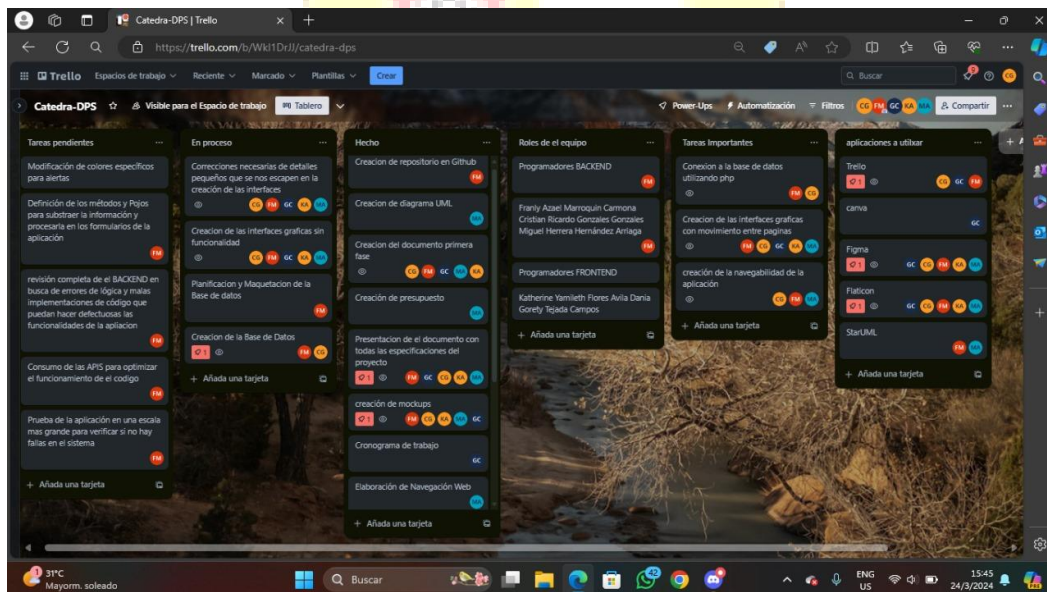
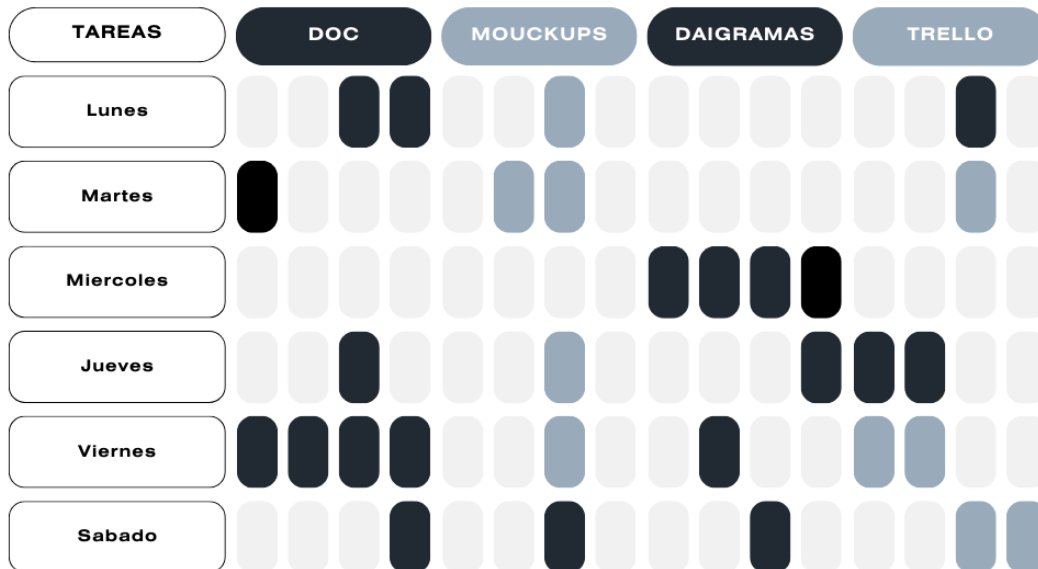
ExpoSnack

Esta plataforma se destaca como una plataforma en línea integral que simplifica el desarrollo móvil, brindando una solución valiosa para los equipos que necesitan colaborar sin acceso a Visual Studio. Su interfaz intuitiva y características versátiles no solo fomentan la eficiencia,

sino que también aseguran la continuidad del desarrollo de la aplicación al facilitar la colaboración entre los miembros del equipo, permitiéndoles avanzar y probar ideas en tiempo real. Con ExpoSnack, se establece un entorno de trabajo dinámico y productivo que impulsa el progreso del proyecto de manera fluida y eficaz.

Cronograma

Sistema de control estudiantil



Fuentes de consulta

Deyimar, A. (2020, febrero 4). *Qué es React: definición, características y funcionamiento*.

Tutoriales Hosting. <https://www.hostinger.es/tutoriales/que-es-react>

Figma. (s/f). Figma. Recuperado el 24 de marzo de 2024, de

<https://www.figma.com/files/recents-and-sharing/recently-viewed?fuid=1219677382660308405>

Iconos y stickers gratuitos - Millones de recursos para descargar. (s/f). Flaticon. Recuperado el 24 de marzo de 2024, de <https://www.flaticon.es/>

StarUML. (s/f). Staruml.Io. Recuperado el 24 de marzo de 2024, de <https://staruml.io/>

(S/f-a). Canva.com. Recuperado el 24 de marzo de 2024, de

<https://www.canva.com/templates/?query=cronograma>

(S/f-b). Cloudflare.com. Recuperado el 24 de marzo de 2024, de <https://www.cloudflare.com/es-es/learning/ssl/what-is-https/>

(S/f-c). Amazon.com. Recuperado el 24 de marzo de 2024, de

<https://aws.amazon.com/es/docker/>

Links:

GitHub

https://github.com/ZANK123-F/Catedra_DPS

Mockups :

<https://www.figma.com/file/6YKXEYihWoiN2jCt3g9rJn/muckups-DPS?type=design&node-id=0-1&mode=design&t=5487zTi41m25lTJv-0>

Trello:

<https://trello.com/b/Wkl1DrJJ/catedra-dps>

