



TECNOLÓGICO  
NACIONAL DE MÉXICO



**Instituto Tecnológico de Chilpancingo**  
Tecnológico Nacional De México

**Desarrollo de Aplicaciones  
Distribuidas**

# TaskZen:

Plataforma de Gestión  
de Proyectos y Tareas  
con React, Node.js,  
MySQL, Firebase y  
MongoDB

**DOCENTE: YANET EVANGELISTA ALCOCER**  
**ALUMNOS:**

- **DAVID MARIANO RUIZ EDGAR**
- **CARBAJAL PÉREZ DAVID GEOVANI**
- **SANTOS HONORATO ULISES SADAMI**

# INDICE

|                                     |        |
|-------------------------------------|--------|
| Introducción .....                  | 3      |
| Problemática .....                  | 4      |
| Objetivos.....                      | 5      |
| Alcance del proyecto .....          | 6 - 13 |
| Requerimientos Funcionales .....    | 14     |
| Requerimientos No Funcionales ..... | 15     |
| Restricciones .....                 | 15     |
| Declaración de trabajo .....        | 16     |
| Metodología .....                   | 17     |
| Diagrama de Arquitectura .....      | 19     |
| Bibliografía .....                  | 21     |

## INTRODUCCIÓN

La gestión de tareas y proyectos es un desafío común en el ámbito empresarial y personal, ya que la cantidad de responsabilidades que enfrentamos puede ser abrumadora y generar estrés. La falta de eficiencia en la gestión de tareas puede afectar negativamente el desempeño y la rentabilidad de las organizaciones.

La plataforma TaskZen es una solución innovadora que utiliza React, Node.js, MySQL, Firebase y MongoDB para gestionar proyectos y tareas de manera eficiente y organizada. La aplicación se presenta en un momento en que la gestión de tareas y proyectos es crucial para enfrentar el caos y la abrumación que experimentamos en nuestras vidas personales y profesionales.

Este caos a menudo se manifiesta en retrasos en la entrega de proyectos, tanto a nivel personal como profesional. Las fechas límite se acercan y, a pesar de nuestros mejores esfuerzos, las tareas se acumulan y se vuelven cada vez más difíciles de gestionar. Esta falta de eficiencia puede tener un impacto significativo en la calidad de nuestro trabajo y en nuestra capacidad para cumplir con los compromisos adquiridos.

Esta problemática puede resultar en la pérdida de oportunidades de negocio, la insatisfacción de clientes y la reducción de la rentabilidad de la empresa. Por lo tanto, es esencial abordar esta problemática de manera efectiva para mejorar tanto el desempeño individual como el rendimiento de las organizaciones.

## PROBLEMÁTICA

En un mundo caracterizado por la constante evolución y la creciente interconexión, las responsabilidades se multiplican exponencialmente, y la cantidad de tareas y proyectos que enfrentamos puede llegar a ser abrumadora. Este fenómeno no se limita al entorno laboral, ya que también se extiende a nuestras vidas personales, donde equilibrar las responsabilidades profesionales, familiares y personales puede resultar una tarea abrumadora.

La abrumación que experimentamos ante esta marea constante de tareas y proyectos puede ser paralizante. Nos encontramos atrapados en un ciclo interminable de demandas y pendientes, lo que a menudo se traduce en una falta de claridad y estructura. Las personas se ven envueltas en un caos de prioridades cambiantes y entregas atrasadas, lo que genera una sensación constante de no estar cumpliendo con nuestras responsabilidades. TaskZen aborda estos problemas ofreciendo una solución centralizada y organizada que permite:

- Planificar y ajustarse a las demandas cambiantes con facilidad
- Mantener una visibilidad clara sobre el progreso de los proyectos y la disponibilidad de recursos
- Priorizar y asignar tareas de manera eficiente a los miembros del equipo
- Mejorar la comunicación y la coordinación entre los miembros del equipo



## OBJETIVOS

### General:

- Desarrollar una plataforma de gestión de proyectos y tareas utilizando React para la interfaz de usuario, un servidor backend con Node.js, y bases de datos MongoDB, MySQL y Firebase, mejorando eficiencia y organización en entornos empresariales y personales.

### Específicos:

1. Diseño y Análisis del Sistema:
  - 1.1 Identificar actores y definir requisitos funcionales y no funcionales.
  - 1.2 Establecer los requisitos del sitio
2. Diseño del Sistema:
  - 2.1 Elaborar plantillas detalladas de casos de uso del sistema.
  - 2.2 Diseñar la arquitectura de la base de datos utilizando MongoDB, MySQL y Firebase.
  - 2.3 Crear prototipos de interfaz
3. Desarrollo del Sistema:
  - 3.1 Implementar un servidor backend robusto utilizando Node.js.
  - 3.2 Desarrollar módulos clave, como la gestión de usuarios, autenticación y autorización.
  - 3.3 Establecer conexiones seguras con las bases de datos MongoDB, MySQL y Firebase.
4. Implementación de Seguridad:
  - 4.1 Implementar funciones de autenticación y autorización para proteger la información.
  - 4.2 Garantizar la seguridad y escalabilidad de la plataforma en todas las interacciones.
5. Pruebas del Sistema:
  - 5.1 Realizar pruebas exhaustivas ejecutando módulos individuales del sistema.

5.2 Vincular de manera efectiva el sistema con las bases de datos para garantizar su funcionamiento correcto.

6. Documentación del Sistema:

6.1 Elaborar una guía detallada para el administrador, cubriendo todas las funciones del sistema.

6.2 Crear un manual de usuario comprensible que describa las características y el uso del sistema.

## ALCANCE DEL PROYECTO

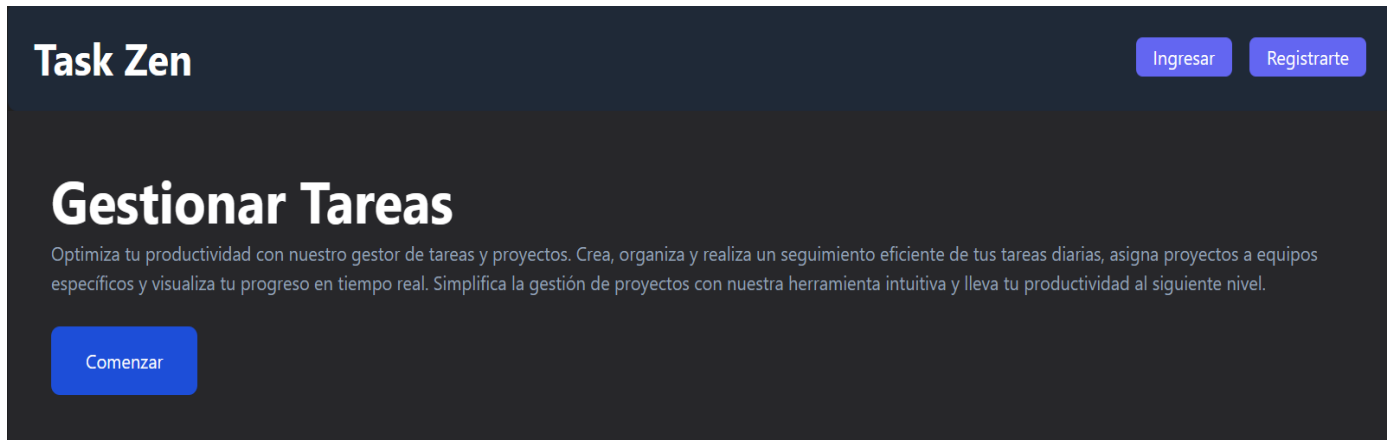
Descripción del alcance del proyecto:

Este proyecto tiene como objetivo principal desarrollar y poner en marcha TaskZen, una plataforma integral de gestión de proyectos y tareas. La finalidad es proporcionar a los usuarios, tanto en entornos empresariales como personales, una herramienta eficiente y organizada para abordar el desafío constante de la multiplicidad de tareas y proyectos. TaskZen empleará tecnologías modernas como React para la interfaz de usuario, Node.js para el servidor backend, y bases de datos MongoDB, MySQL y Firebase para garantizar un rendimiento escalable y seguro. La plataforma ofrecerá funcionalidades clave, incluyendo la planificación flexible, la visibilidad clara del progreso de proyectos, la asignación eficiente de tareas, y mejoras en la comunicación y coordinación entre equipos. Este proyecto busca transformar la forma en que los individuos y equipos gestionan sus responsabilidades diarias, mejorando significativamente la eficiencia y calidad de vida.

Existe una necesidad crítica en las empresas y en la vida cotidiana de las personas de contar con una herramienta de gestión de tareas y proyectos efectiva que permita la organización, el seguimiento y la colaboración en tiempo real. Esta herramienta debe ser fácil de usar y accesible a través de una plataforma web, adaptable a diferentes tipos de proyectos y necesidades.

Página de Inicio:

La interfaz principal de TaskZen será la página de inicio, donde los usuarios podrán acceder a información clave sobre la gestión de tareas y proyectos. El diseño incluirá elementos visuales que representen la identidad de TaskZen y proporcionará opciones claras para iniciar sesión, registrarse como nuevos usuarios.



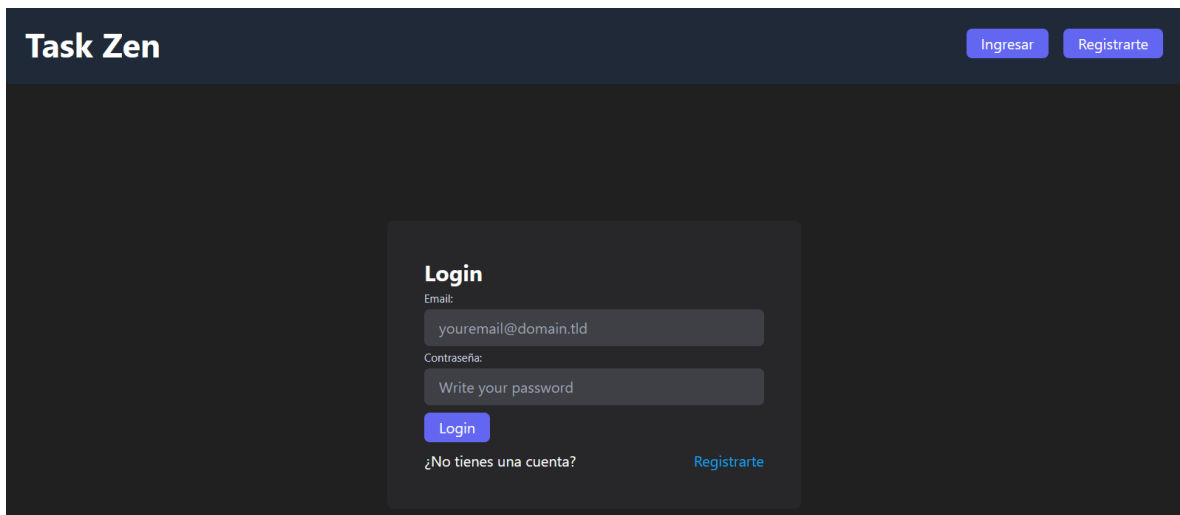
#### Registro de Usuarios:

La sección de registro permitirá a los usuarios crear cuentas en TaskZen proporcionando información personal esencial. El formulario solicitará detalles como nombre completo, correo electrónico y una contraseña segura de 6 dígitos. La validación de datos se realizará en tiempo real.

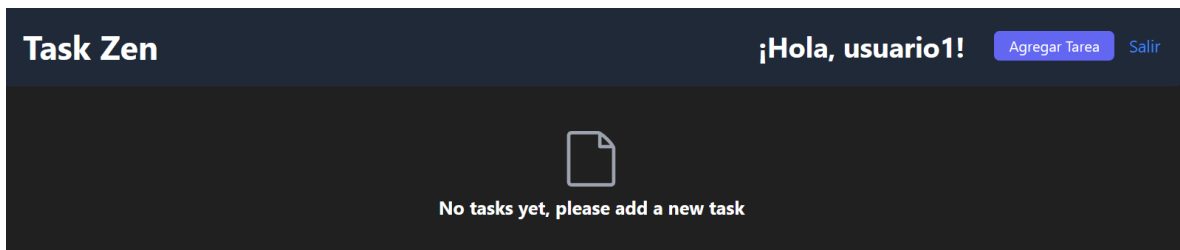
The screenshot shows the TaskZen registration form. It has a dark blue header with the 'Task Zen' logo and 'Ingresar' and 'Registrarte' buttons. The form is titled 'Register' and includes fields for 'Usuario:' (with placeholder 'Write your name'), 'Correo Electrónico:' (with placeholder 'youremail@domain.tld'), 'Contraseña:' (with placeholder '\*\*\*\*\*'), and 'Confirmar Contraseña:' (with placeholder '\*\*\*\*\*'). There is a blue 'Submit' button at the bottom of the form. Below the form, there is a link that says '¿Ya tienes una cuenta? Ingresar'.

#### Login:

Se presenta un módulo de login que garantiza un acceso seguro y eficiente a la plataforma. Los usuarios pueden ingresar con su correo electrónico y contraseña previamente registrados. Se incorpora un proceso de recuperación de contraseña que verifica la identidad a través del correo electrónico o número de teléfono registrado

The screenshot shows the 'Task Zen' login interface. At the top left is the 'Task Zen' logo. At the top right are two buttons: 'Ingresar' and 'Registrarte'. In the center is a 'Login' form with fields for 'Email' (containing 'youremail@domain.tld') and 'Contraseña' (containing 'Write your password'). Below the password field is a 'Login' button. At the bottom of the form are links for '¿No tienes una cuenta?' and 'Registrarte'.

- Agregar Tarea: Este botón te permite incorporar nuevas tareas a tu lista. Simplemente haz clic y completa la información necesaria.

The screenshot shows the 'Task Zen' dashboard after a user login. The top bar includes the 'Task Zen' logo, a greeting '¡Hola, usuario1!', and buttons for 'Agregar Tarea' and 'Salir'. The main content area features a document icon and the text 'No tasks yet, please add a new task'.

Al hacer clic en "Agregar Tarea":

- Campo de Título: Se despliega un campo para ingresar el título de la tarea.
- Campo de Descripción: Se presenta un área de texto para detallar la descripción de la tarea.
- Campo de Fecha: Aparece un selector de fecha para establecer la fecha límite de la tarea.
- Botón "Guardar": Un botón visualmente destacado para confirmar y almacenar la tarea.



Task Zen

¡Hola, usuario1!

Agregar Tarea

Salir

Title

Description

Date

dd/mm/aaaa

Save

Además de la capacidad para agregar tareas, en nuestra pantalla de inicio también ofreceremos la funcionalidad de incorporar proyectos. Esta característica permitirá a los usuarios organizar sus responsabilidades de manera más amplia y estructurada. Cada proyecto podrá contener múltiples tareas, brindando una visión holística de los objetivos y permitiendo una gestión más eficiente de las metas a largo plazo. Los proyectos serán elementos clave para la planificación estratégica y la organización general de las actividades, brindando una experiencia completa y versátil en la gestión de tareas y proyectos. Además, podrás asignar tareas a proyectos específicos, estableciendo una conexión clara entre tus objetivos generales y las acciones individuales requeridas para alcanzarlos. Esto facilitará aún más la coordinación y la comprensión de cómo las tareas contribuyen al progreso de los proyectos en curso..

Task Manager

Welcome Usuario David

Añadir Tarea

Añadir Proyecto

Cerrar Sesión

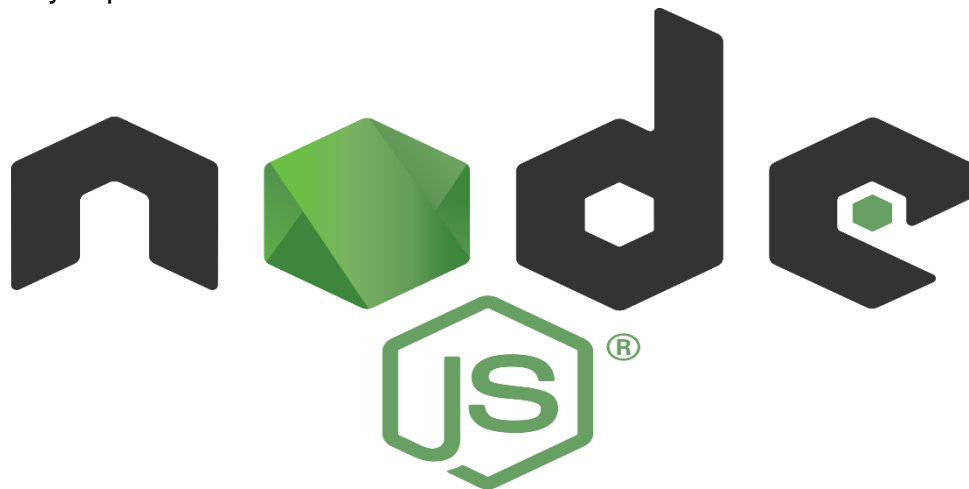
Nombre del Proyecto

Nombre del Proyecto

Guardar

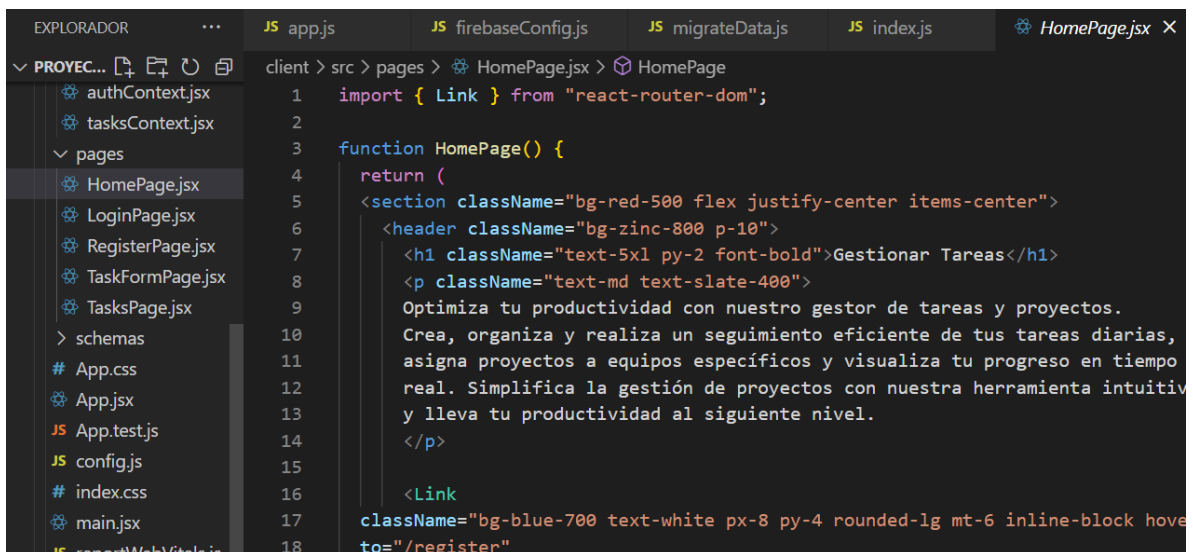
| Task Manager  |   |   | Welcome Usuario David                                   |   |   | Añadir Tarea  | Añadir Proyecto   | Cerrar Sesión   |
|---|---|---|---|---|---|---|---|---|
| ID Proyecto: 11<br>ID Usuario: 65725089deef30de46607c76 | ID Proyecto: 12<br>ID Usuario: 65725089deef30de46607c76 | ID Proyecto: 13<br>ID Usuario: 65725089deef30de46607c76 | ID Proyecto: 14<br>ID Usuario: 65725089deef30de46607c76 | ID Proyecto: 15<br>ID Usuario: 65725089deef30de46607c76 | ID Proyecto: 16<br>ID Usuario: 65725089deef30de46607c76 | ID Proyecto: 17<br>ID Usuario: 65725089deef30de46607c76 | ID Proyecto: 18<br>ID Usuario: 65725089deef30de46607c76 | ID Proyecto: 19<br>ID Usuario: 65725089deef30de46607c76 |
| <div>Abrir</div> <div>Editar</div> <div>Eliminar</div>  | <div>Abrir</div> <div>Editar</div> <div>Eliminar</div>  | <div>Abrir</div> <div>Editar</div> <div>Eliminar</div>  | <div>Abrir</div> <div>Editar</div> <div>Eliminar</div>  | <div>Abrir</div> <div>Editar</div> <div>Eliminar</div>  | <div>Abrir</div> <div>Editar</div> <div>Eliminar</div>  | <div>Abrir</div> <div>Editar</div> <div>Eliminar</div>  | <div>Abrir</div> <div>Editar</div> <div>Eliminar</div>  | <div>Abrir</div> <div>Editar</div> <div>Eliminar</div>  |

La razón de usar Node.js es porque es una plataforma de backend que permite ejecutar código JavaScript en el servidor. Es una opción popular para desarrollar servidores backend para aplicaciones web y móviles, incluyendo aplicaciones React. Node.js es escalable, eficiente y flexible, y su capacidad para usar JavaScript tanto en el frontend como en el backend permite una mayor eficiencia y reducción de costos en el desarrollo de aplicaciones. Para implementar un servidor backend con Node.js, se deben seguir los siguientes pasos: configurar el entorno de desarrollo, crear un nuevo proyecto de Node.js, desarrollar el servidor backend utilizando Node.js y Express, implementar módulos clave como la gestión de usuarios, autenticación y autorización, establecer conexiones seguras con las bases de datos MongoDB, MySQL y Firebase, y realizar pruebas exhaustivas para garantizar que el servidor backend funcione correctamente y cumpla con los requisitos y especificaciones definidas



La razón de utilizar React en el desarrollo de la plataforma TaskZen es que React es una biblioteca de JavaScript que permite crear interfaces de usuario interactivas y escalables. React es muy popular en el desarrollo de aplicaciones web y móviles, ya que permite la creación de componentes reutilizables y la actualización eficiente de la interfaz de usuario en tiempo real. Además, React es compatible con una amplia variedad de herramientas y bibliotecas, lo que lo hace muy flexible y

adaptable a diferentes necesidades y requisitos de proyectos, por lo que realizaremos nuestras plantillas con react de la siguiente manera:



```
client > src > pages > HomePage.jsx > HomePage
1  import { Link } from "react-router-dom";
2
3  function HomePage() {
4    return (
5      <section className="bg-red-500 flex justify-center items-center">
6        <header className="bg-zinc-800 p-10">
7          <h1 className="text-5xl py-2 font-bold">Gestionar Tareas</h1>
8          <p className="text-md text-slate-400">
9            Optimiza tu productividad con nuestro gestor de tareas y proyectos.
10           Crea, organiza y realiza un seguimiento eficiente de tus tareas diarias,
11           asigna proyectos a equipos específicos y visualiza tu progreso en tiempo
12           real. Simplifica la gestión de proyectos con nuestra herramienta intuitiva
13           y lleva tu productividad al siguiente nivel.
14         </p>
15       </section>
16     );
17   }
18   <Link
```



```
client > src > pages > LoginPage.jsx > LoginPage
28  return (
29    <div className="h-[calc(100vh-100px)] flex items-center justify-center">
30      <Card>
31        {loginErrors.map((error, i) => (
32          <Message message={error} key={i} />
33        ))}
34        <h1 className="text-2xl font-bold">Login</h1>
35
36        <form onSubmit={handleSubmit(onSubmit)}>
37          <Label htmlFor="email">Email:</Label>
38          <Input
39            label="Write your email"
40            type="email"
41            name="email"
42            placeholder="youremail@domain.tld"
43            {...register("email", { required: true })}
44          />
45          <p>{errors.email?.message}</p>
39  PROBLEMAS  CONSOLA DE DEPURACIÓN  TERMINAL  PUERTOS
```

```

PROYEC... client > src > pages > RegisterPage.jsx > ...
  authContext.jsx 24
  tasksContext.jsx 25
  pages
    HomePage.jsx 26
    LoginPage.jsx 27
    RegisterPage.jsx 28
    TaskFormPage.jsx 29
    TasksPage.jsx 30
  > schemas
  # App.css 31
  App.jsx 32
  App.test.js 33
  config.js 34
  # index.css 35
  main.jsx 36
  reportWebVitals.js 37

24 return (
25   <div className="h-[calc(100vh-100px)] flex items-center justify-center">
26     <Card>
27       {registerErrors.map((error, i) => (
28         <Message message={error} key={i} />
29       ))}
30       <h1 className="text-3xl font-bold">Register</h1>
31       <form onSubmit={handleSubmit(onSubmit)}>
32         <FormGroup label="Usuario" name="username" errors={errors}>
33           <Input
34             type="text"
35             placeholder="Write your name"
36             {...register("username")}
37             autoFocus
38           />
39         </FormGroup>
40
41       <FormGroup label="Correo Electrónico" name="email" errors={errors}>

```

```

PROYEC... client > src > pages > TaskFormPage.jsx > TaskFormPage
  authContext.jsx 70
  tasksContext.jsx 71
  pages
    HomePage.jsx 72
    LoginPage.jsx 73
    RegisterPage.jsx 74
    TaskFormPage.jsx 75
    TasksPage.jsx 76
  > schemas
  # App.css 77
  App.jsx 78
  App.test.js 79
  config.js 80
  # index.css 81
  main.jsx 82
  reportWebVitals.js 83

70 {errors.title && (
71   <p className="text-red-500 text-xs italic">Please enter a title.</p>
72 )}
73
74 <Label htmlFor="description">Description</Label>
75 <Textarea
76   name="description"
77   id="description"
78   rows="3"
79   placeholder="Description"
80   {...register("description")}
81 ></Textarea>
82
83 <Label htmlFor="date">Date</Label>
84 <Input type="date" name="date" {...register("date")} />
85 <Button>Save</Button>
86 </form>
87 </Card>

```

```

PROYECTODISTRIBUIDAS client > src > pages > TasksPage.jsx > ...
  authContext.jsx 4
  tasksContext.jsx 5
  pages
    HomePage.jsx 6
    LoginPage.jsx 7
    RegisterPage.jsx 8
    TaskFormPage.jsx 9
    TasksPage.jsx 10
  > schemas
  # App.css 11
  App.jsx 12
  App.test.js 13
  config.js 14
  # index.css 15
  main.jsx 16

4 import { ImFileEmpty } from 'react-icons/im';
5
6 export function TasksPage() {
7   const { tasks, getTasks } = useTasks();
8
9   useEffect(() => {
10     getTasks();
11   }, []);
12
13   return (
14     <>
15       {tasks.length === 0 && (
16         <div className="flex justify-center items-center p-10">
17           <div>
18             <ImFileEmpty className="text-6xl text-gray-400 m-auto my-2" />
19             <h1 className="font-bold text-xl">
20               No tasks yet, please add a new task
21             </h1>

```

Una vez que los usuarios se han registrado en la plataforma TaskZen, sus datos se visualizarán en la base de datos de MongoDB, donde se almacenarán en una colección llamada "USERS"

Esta colección contendrá la información personal de los usuarios, como nombres, direcciones de correo electrónico, contraseñas cifradas, entre otros datos relevantes. El uso de MongoDB permitirá un almacenamiento seguro y eficiente de la información, facilitando su posterior recuperación y manipulación por parte del sistema.

Además, en la colección de "TASKS", se almacenarán las tareas y proyectos asignados a los usuarios. Cada tarea y proyecto tendrá su propia información, como título, descripción, fecha límite, estado de completado, entre otros detalles relevantes. La estructura de la colección de tasks permitirá la planificación flexible, la visibilidad clara del progreso de proyectos, la asignación eficiente de tareas y mejoras en la comunicación y coordinación entre equipos

En resumen, la arquitectura de la plataforma TaskZen se basa en la siguiente estructura:

- Usuarios: Almacenados en la colección "usuarios" de MongoDB.
- Tareas y proyectos: Almacenados en la colección "tasks" de MongoDB.

Esta estructura permitirá a los usuarios gestionar sus tareas y proyectos de manera eficiente y organizada, contando con una base de datos segura y confiable para el almacenamiento de la información.

test

LOGICAL DATA SIZE: 1.04KB   STORAGE SIZE: 72KB   INDEX SIZE: 108KB   TOTAL COLLECTIONS: 2

CREATE COLLECTION

| Collection Name | Documents | Logical Data Size | Avg Document Size | Storage Size | Indexes | Index Size | Avg Index Size |
|-----------------|-----------|-------------------|-------------------|--------------|---------|------------|----------------|
| tasks           | 3         | 487B              | 163B              | 36KB         | 1       | 36KB       | 36KB           |
| users           | 3         | 580B              | 194B              | 36KB         | 2       | 72KB       | 36KB           |

## Requerimientos Funcionales

- El sistema permitirá a los usuarios elegir entre iniciar sesión o registrarse como nuevos usuarios, proporcionando información personal como nombre completo, correo electrónico y una contraseña segura de al menos 6 dígitos para el registro.
- El sistema permitirá el inicio de sesión con el correo electrónico y contraseña previamente registrados, verificando la autenticidad de las credenciales del usuario.
- El sistema permitirá proporcionar opciones claras para iniciar sesión, registrarse como nuevos usuarios y recuperar la contraseña, a través de un proceso sencillo y seguro.
- El sistema permitirá a los usuarios agregar nuevas tareas a su lista, incluyendo campos para el título, descripción y fecha límite, facilitando la creación y gestión de tareas.
- El sistema permitirá a los usuarios planificar y ajustarse a las demandas cambiantes con facilidad, brindando herramientas intuitivas para la planificación y organización de tareas y proyectos.
- El sistema permitirá la comunicación y la coordinación entre los miembros del equipo, ofreciendo funciones de mensajería, notificaciones y colaboración en tiempo real.
- El sistema permitirá establecer conexiones seguras con las bases de datos MongoDB, MySQL y Firebase para garantizar la seguridad y escalabilidad de la plataforma, protegiendo la información del usuario y garantizando un rendimiento óptimo.
- El sistema permitirá implementar funcionalidades de autenticación y autorización para garantizar la protección de la información, asegurando que solo los usuarios autorizados puedan acceder y modificar los datos.



## Requerimientos No Funcionales

- El sistema deberá ser fácil de usar y accesible a través de una plataforma.
- El sistema deberá ser adaptable a diferentes tipos de proyectos y necesidades de los usuarios.
- El servidor backend deberá ser escalable y capaz de manejar un gran número de usuarios y solicitudes.
- El sistema deberá garantizar la seguridad y protección de la información de los usuarios, incluyendo la autenticación, autorización y protección contra ataques de hackers.
- El sistema permitirá la colaboración en tiempo real entre los miembros del equipo y proporcionar una comunicación efectiva.
- El sistema deberá ser fácil de mantener y actualizar, permitiendo a los desarrolladores realizar mejoras y correcciones rápidas.
- El sistema deberá ser capaz de integrarse con otras herramientas y servicios de terceros, como bases de datos, API y aplicaciones

## Entregables:

- Sitio web funcional y seguro para la gestión de proyectos y tareas.
- Base de datos MongoDB, MySQL y Firebase.
- Plan de trabajo detallado.
- Manual del administrador que cubra todas las funciones del sistema.
- Manual del usuario que describa las características y el uso del sistema.

## Características de los entregables:

- Interfaz de usuario intuitiva y fácil de usar.
- Base de datos segura y confiable para el almacenamiento de la información de proyectos, tareas y usuarios.
- Cronograma de actividades detallado que establezca la duración del proyecto, la fecha de inicio y final.
- Manual del administrador que cubra todas las funciones del sistema.

- Manual del usuario que describa las características y el uso del sistema.

## Restricciones

- El sistema debe cumplir con las regulaciones de protección de datos y privacidad, como el RGPD, para garantizar la seguridad y privacidad de la información del usuario.
- El sistema debe ser compatible con una amplia variedad de navegadores web y dispositivos móviles para garantizar una experiencia de usuario óptima.
- El sistema debe tener una disponibilidad del 99.9% para garantizar que los usuarios puedan acceder al sistema en todo momento.
- El sistema debe tener una capacidad de escalabilidad para manejar un gran número de usuarios y tareas a medida que el sistema crece.
- El sistema debe tener una capacidad de recuperación para garantizar que los datos del usuario estén seguros en caso de un fallo del sistema.
- El sistema debe tener una capacidad de rendimiento para garantizar que el sistema funcione de manera eficiente y rápida, incluso en momentos de alta demanda.

## Declaración del trabajo:

El objetivo principal del proyecto es desarrollar e implementar la plataforma TaskZen, una herramienta de gestión de proyectos y tareas que permita a los usuarios planificar y ajustarse a las demandas cambiantes con facilidad, mantener una visibilidad clara sobre el progreso de los proyectos y la disponibilidad de recursos, priorizar y asignar tareas de manera eficiente a los miembros del equipo, y mejorar la comunicación y la coordinación entre los miembros del equipo.

La plataforma TaskZen utilizará tecnologías modernas como React para la interfaz de usuario, Node.js para el servidor backend, y bases de datos MongoDB, MySQL y Firebase para garantizar un rendimiento escalable y seguro.

Los entregables del proyecto incluirán una plataforma web funcional y segura para la gestión de proyectos y tareas, una base de datos centralizada y segura para almacenar la información de proyectos, tareas, usuarios y otros datos relevantes, un plan de trabajo detallado, un manual del administrador y un manual del usuario.

La plataforma TaskZen cumplirá con los estándares de calidad y usabilidad para brindar una experiencia fluida a los usuarios, y se enfocará únicamente en cumplir con los requisitos establecidos sin realizar cambios, adiciones o funcionalidades adicionales no previamente acordadas.

Las restricciones del proyecto incluyen cumplir con los estándares de seguridad y privacidad de la información, ser compatible con diferentes navegadores web y dispositivos móviles, y enfocarse únicamente en cumplir con los requisitos establecidos sin realizar cambios, adiciones o funcionalidades adicionales no previamente acordadas.

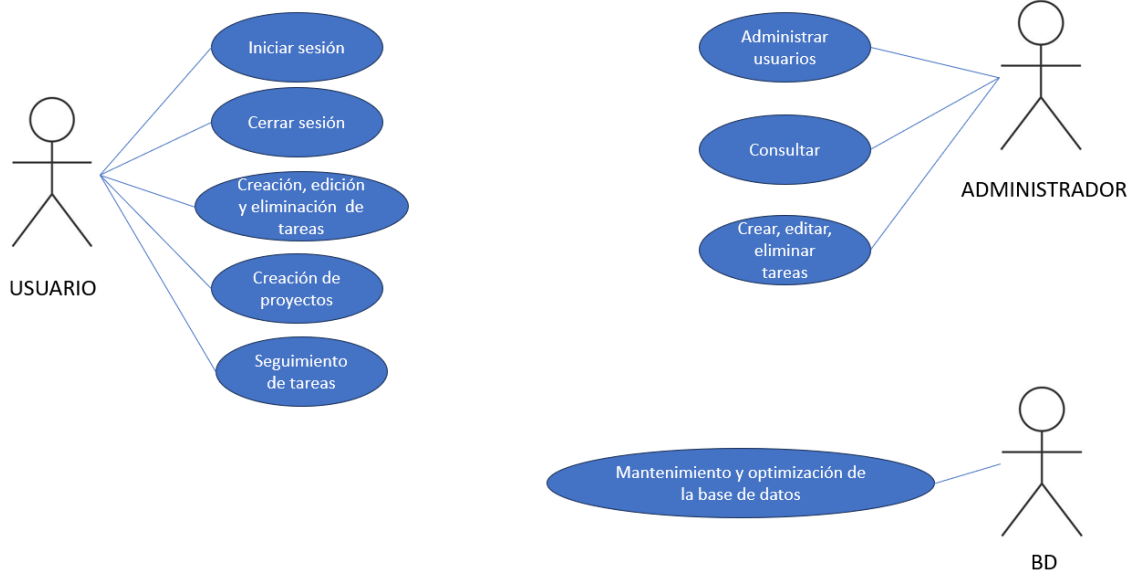
## Metodología

La metodología utilizada en el proyecto TaskZen será Scrum, una metodología ágil que se enfoca en la flexibilidad, la colaboración y la entrega rápida de valor. Scrum es un enfoque iterativo e incremental, lo que significa que el desarrollo se divide en ciclos cortos llamados sprints, permitiendo entregar incrementos funcionales en cada iteración.

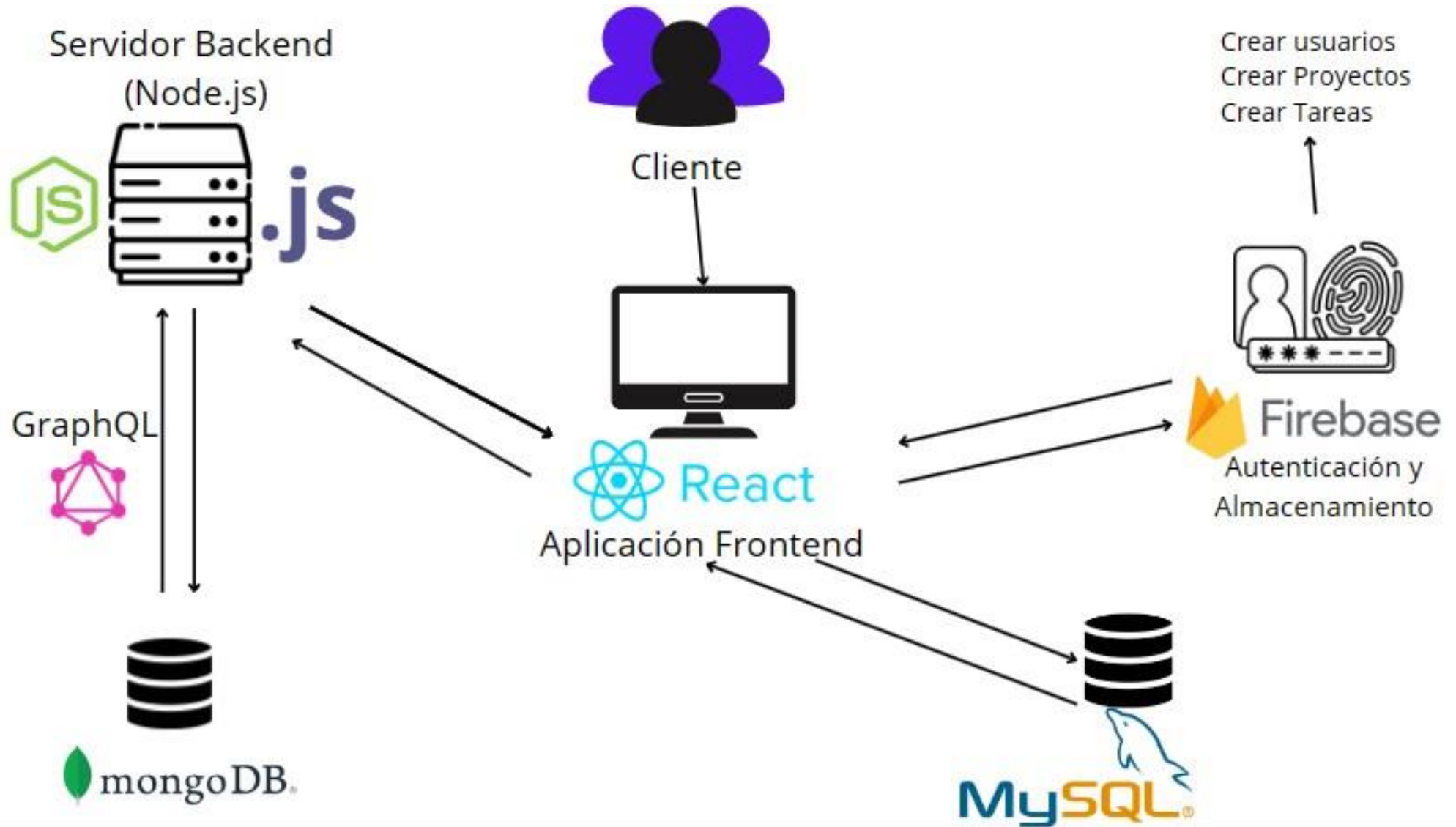
Scrum también se destaca por su adaptabilidad, ya que permite ajustar los requisitos y prioridades a lo largo del proyecto, lo cual es beneficioso considerando los cambios a las necesidades y requerimientos del proyecto TaskZen.

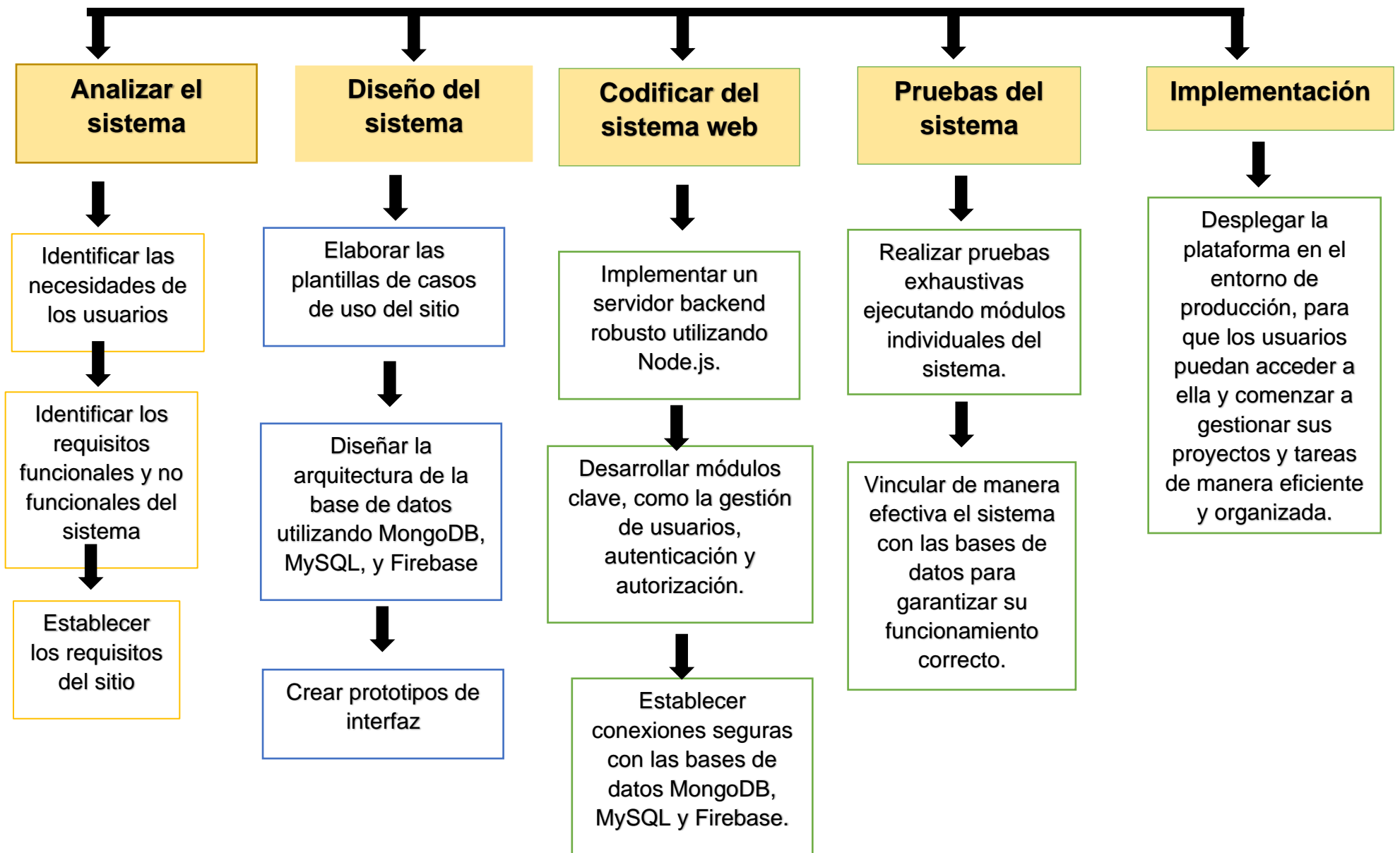
El equipo se ajustó a las horas de clase para realizar pruebas entre el equipo y ver los errores que surgían. Se enfocó en la entrega rápida de valor y la adaptabilidad a los cambios en las necesidades y requerimientos del proyecto.

Se realizaron reuniones regulares para proporcionar resultados visibles y utilizables, lo que brindó visibilidad sobre el progreso del proyecto y ayudó a identificar cualquier problema o desviación a tiempo.



## DIAGRAMA DE ARQUITECTURA







## BIBLIOGRAFÍA

- <https://cloud.mongodb.com/v2/655e499b09ecaa76b87f61e4#/overview>
- <https://firebase.google.com/?hl=es>
- <https://visualstudio.microsoft.com/es/>
- <https://graphql.org/>