

21-5-2023

UNIVERSIDAD DON BOSCO



Proyecto DSP – Segunda fase

Asignatura:

Diseño y Programación de Software Multiplataforma

Docente:

Ing. Alexander Sigüenza

Alumnos:

| | |
|------------------------------------|----------|
| Gerardo Antonio Cabezas Vaquero | CV152055 |
| Yosselin Beatriz Martínez Alvarado | MM110166 |
| Gabriela María Pineda González | PG120866 |

Tabla de contenido

| | |
|--|----|
| Introducción..... | 2 |
| Diseño UX/UI (Mock ups) | 3 |
| Lógica de la aplicación. | 11 |
| diagrama uml..... | 12 |
| Detalle de las herramientas a utilizar | 13 |
| Asignación de actividades | 14 |
| cronograma de actividades | 15 |
| presupuesto de costos de la aplicación..... | 16 |
| Bibliografía | 17 |

INTRODUCCIÓN

En la actualidad para poder estar al día con la información más reciente y actualizada es indispensable tener un dispositivo móvil conectado al internet. Por lo que las empresas han tenido que adaptarse en mayor o menor medida a la era digital que estamos viviendo.

En el presente trabajo se hará una herramienta para la consulta de pacientes de una clínica médica consiste en una aplicación móvil capaz de administrar la información necesaria referente a los pacientes y cada consulta realizada en la clínica médica, se pretende facilitar el acceso a la información, tanto para los pacientes como para los empleados de esta, con el objetivo de mejorar la experiencia de la clínica.

La aplicación estará diseñada para análisis, diseño y programación de un aplicativo que optimice los procesos de registro, búsqueda y control de las historias clínicas de los pacientes del centro médico, la app se lleva bajo una base de datos normalizada la cual nos da la integridad de nuestra información, dándonos así la confiabilidad de los datos

La aplicación se desarrollará con React Native (Node.JS/Visual Studio Code), se implementará Cloud Firestore que es una base de datos flexible y escalable para el desarrollo en servidores, dispositivos móviles y la Web desde Firebase y Google Cloud.

DISEÑO UX/UI (MOCK UPS)

Un mockup o bosquejo es un fotomontaje a través del cual los diseñadores del frontend presentan las propuestas de las aplicaciones; Con los mockups se puede valorar “cómo funcionan” o cómo lucen bajo un contexto particular. Al mismo tiempo gracias a ellos, se puede estudiar la capacidad de la propuesta y el cliente podrá obtener una visión mucho más acertada y realista de la aplicación.

El presente trabajo pretende mostrar el proceso de creación de la aplicación para un centro médico, por lo que vamos a iniciar con los mockups:

- 1- Página de inicio, esta será la que pueden visualizar todos los usuarios al iniciar el aplicativo. Debido a que es una aplicación de seguimiento se debe de tener un usuario, por lo que puede iniciar sesión para ver el historial medico en el hospital, si es la primera vez, debe de registrarse.





Bienvenidos/as Centro Medico

Ingrese Correo

Nombre Completo

Usuario

Contraseña

Registrar

2. Para poder crear un usuario con el perfil de paciente debe de ingresar el correo, el nombre completo, usuario y contraseña:

3. Los usuarios de medico deben de ser creados por el administrador, por lo que se le debe de brindar el usuario y la contraseña para poder ingresar



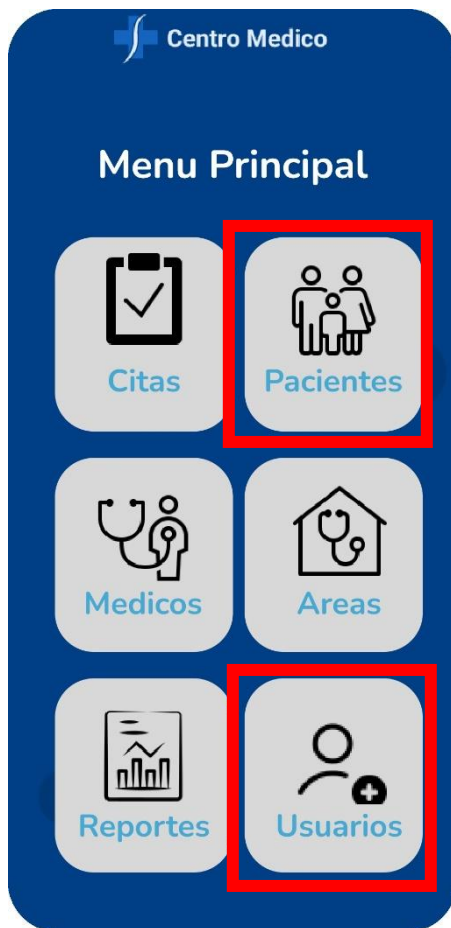
 Centro Medico

Bienvenidos/as Centro Medico

Usuario

Contraseña

Inicio

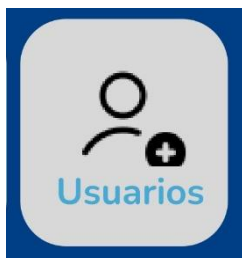


4. Cuando ya se han logeado con alguno de los usuarios de clientes; ingresa a la pantalla principal por lo que puede agregar pacientes, citas, áreas, reportes.

Los botones marcados en rojo se van a manejar en función del perfil.



Esto solo aparecerá en el perfil administrador y en el perfil de médico.



Este solo aparecerá en el perfil de administrador,

5. Posterior se regreara la creación del usuario de paciente este ya ya puede: crear, eliminar o editar la cita, según la necesidad.

Centro Medico

Editar / Eliminar Cita

Paciente

Medico

Fecha y Hora

Sintomas

Medicamentos

Tipo de Pago

Costo

Editar Cita Eliminar Cita

Centro Medico

Ingreso de Citas

Paciente

Medico

Fecha y Hora

Sintomas

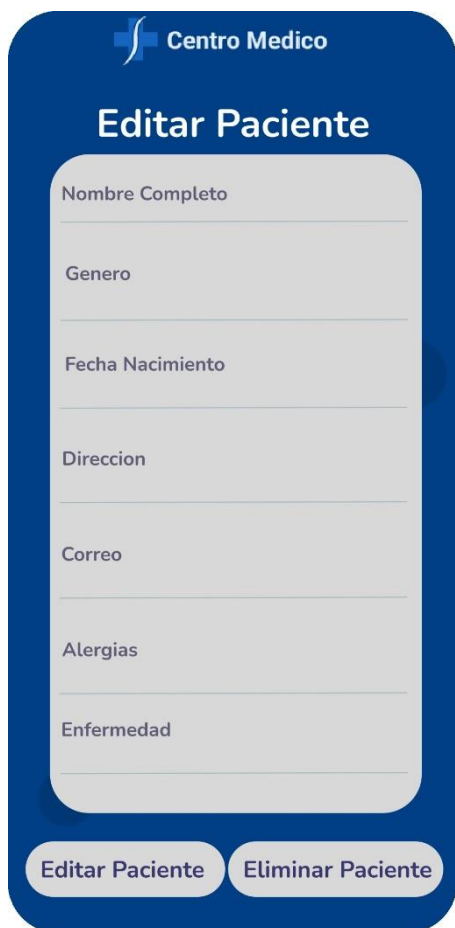
Medicamentos

Tipo de Pago

Costo

Agregar Cita

5. Posterior se regreara la creación del medico y pueden crear y editar el registro de pacientes.



Centro Medico

Editar Paciente

Nombre Completo

Genero

Fecha Nacimiento

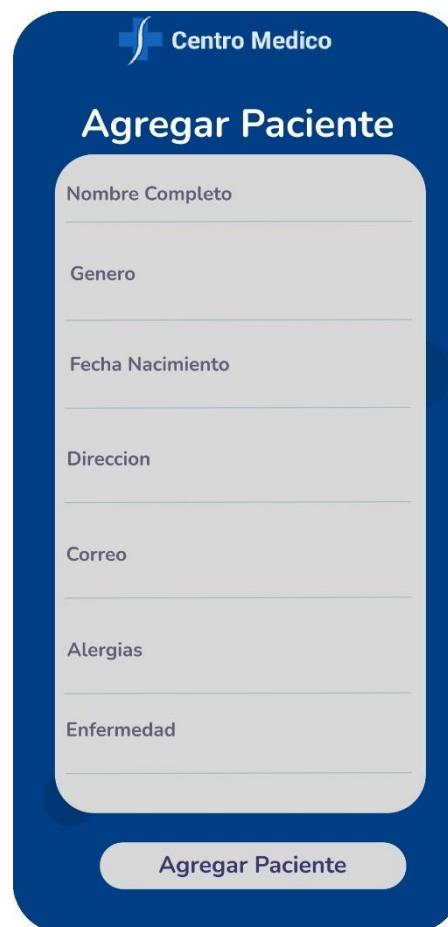
Direccion

Correo

Alergias

Enfermedad

Editar Paciente Eliminar Paciente



Centro Medico

Agregar Paciente

Nombre Completo

Genero

Fecha Nacimiento

Direccion

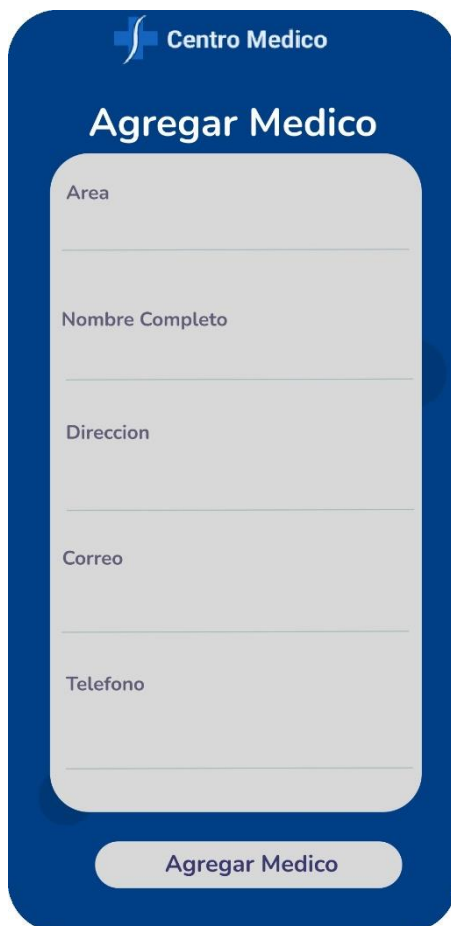
Correo

Alergias

Enfermedad

Agregar Paciente

6. En el perfil de administrador se podran crear, editar o eliminar registros con el perfil de medico.



Centro Medico

Agregar Medico

Area

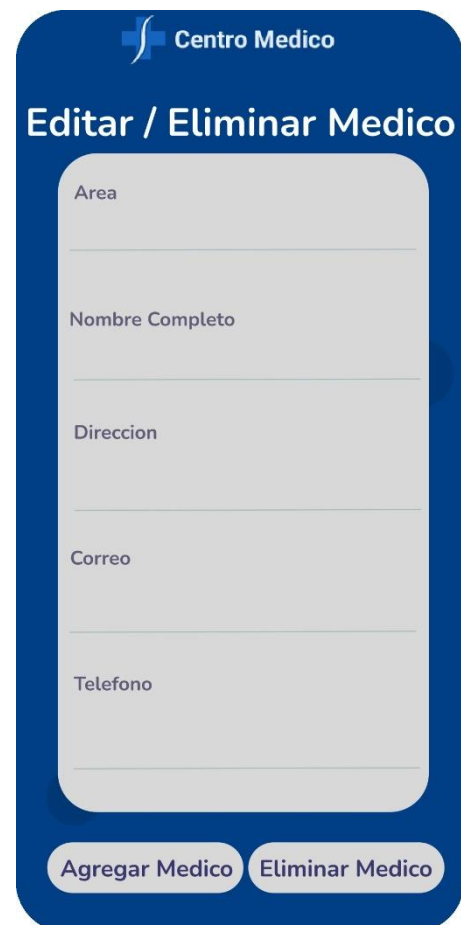
Nombre Completo

Direccion

Correo

Telefono

Agregar Medico



Centro Medico

Editar / Eliminar Medico

Area

Nombre Completo

Direccion

Correo

Telefono

Agregar Medico Eliminar Medico

6. En el perfil de administrador se podran crear, editar o eliminar registros con el perfil de pacientes.

Centro Medico

Agregar Usuario

Nombre Completo

Usuario

Correo

Contraseña

Tipo de Rol

Agregar Usuario

Centro Medico

Editar/Eliminar Usuario

Nombre Completo

Usuario

Correo

Contraseña

Tipo de Rol

Agregar Usuario

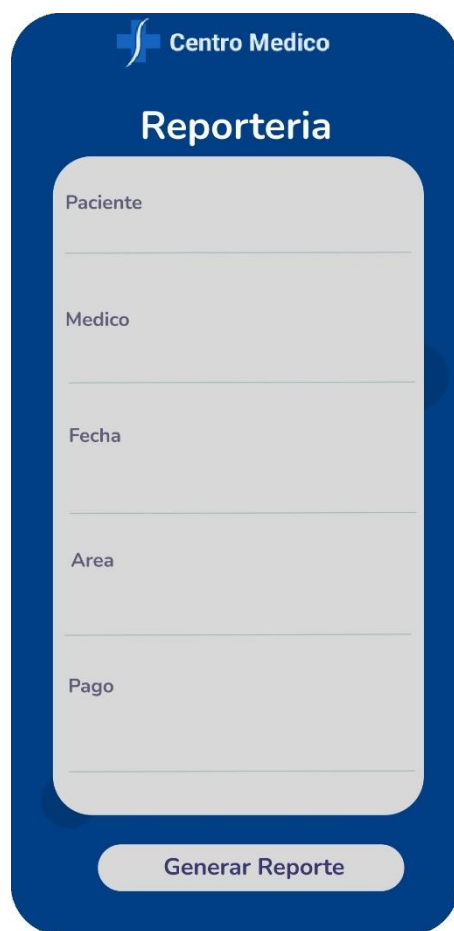
Eliminar Usuario

7. En el perfil de administrador podrá agregar nuevas categorias/area:

The screen has a dark blue header with the 'Centro Medico' logo and title. Below the header, the title 'Agregar Area' is displayed in white. A light gray rounded rectangle contains two input fields: 'Area' and 'Sub-Area'. Below this form is a white button with the text 'Agregar Area'. A dark blue circular button is located at the bottom left of the screen.

The screen has a dark blue header with the 'Centro Medico' logo and title. Below the header, the title 'Editar / Eliminar Area' is displayed in white. A light gray rounded rectangle contains two input fields: 'Area' and 'Sub-Area'. Below this form are two white buttons: 'Agregar Area' and 'Eliminar Area'. A dark blue circular button is located at the bottom left of the screen.

6. En el perfil de medicos, pacientes y administrador se podran visualizar un resumen de la visita.



Centro Medico

Reporteria

Paciente

Medico

Fecha

Area

Pago

Generar Reporte

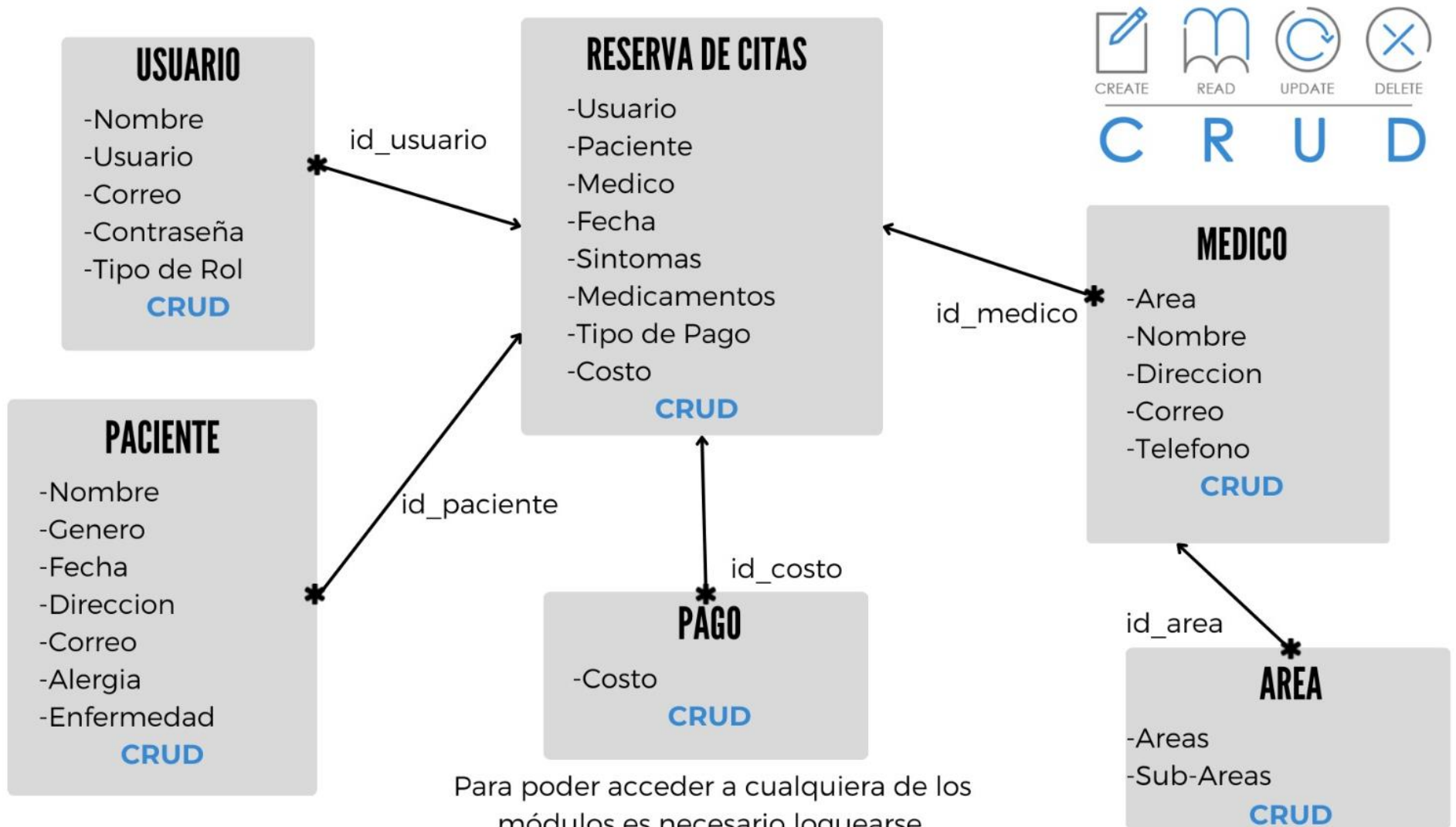
LÓGICA DE LA APLICACIÓN.

La aplicación propuesta pretende brindar soluciones efectivas y rápidas a los pacientes que quieren monitorear el histórico de sus condiciones de salud
















La aplicación debe de tener las funciones fundamentales de un registro de pacientes, médicos, citas, histórico de citas y pagos; la aplicación debe de ahorrar tiempo a la hora de programar, cancelar o reagendar una cita.

Para que la aplicación sea de uso frecuente de los usuarios finales debe de ser fácil de utilizar, acelerar procesos en un sitio que guarde los datos con la confidencialidad necesaria.

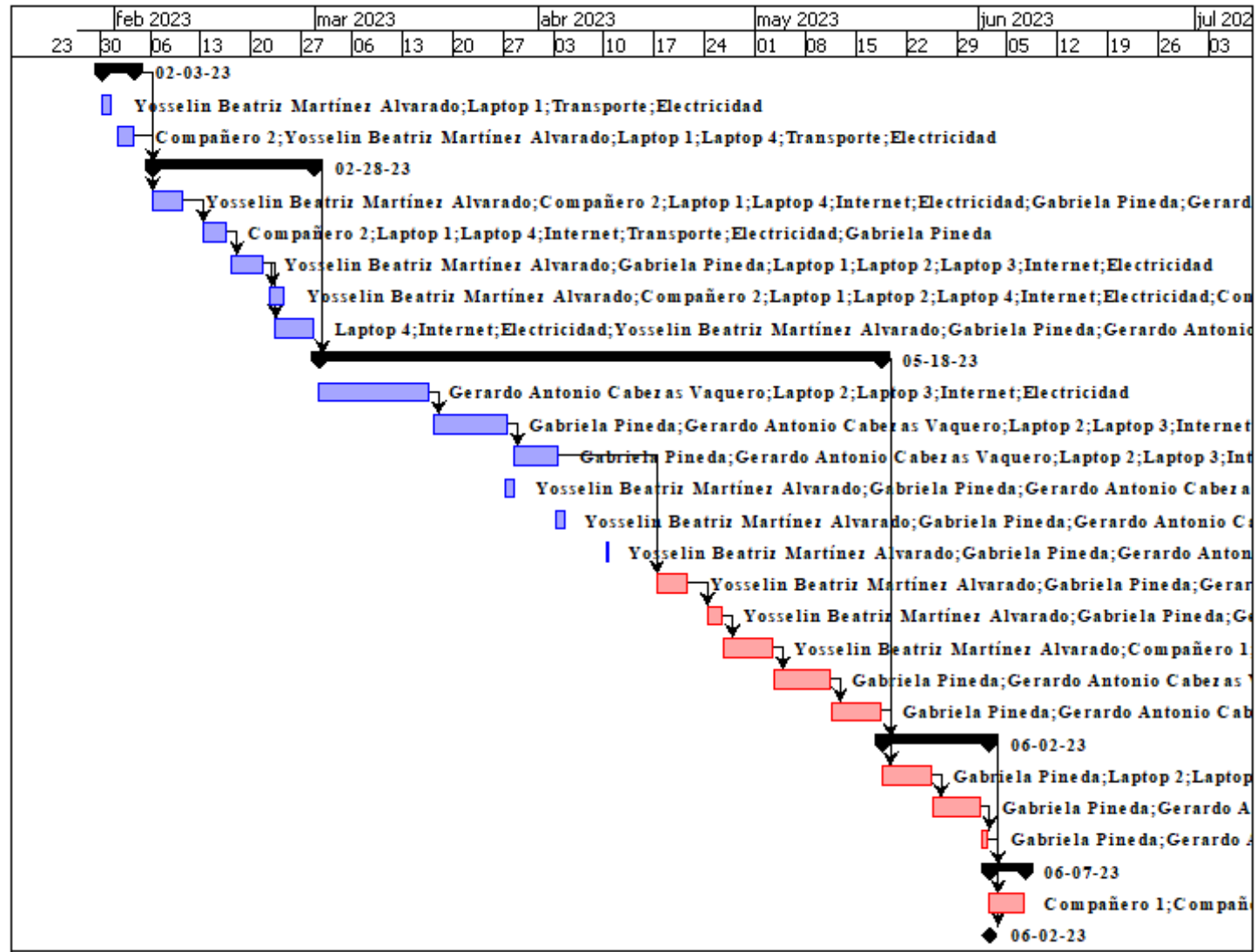
DIAGRAMA UML



DETALLE DE LAS HERRAMIENTAS A UTILIZAR

| |  | Nombre | Duración | Inicio | Terminado |
|----|---|---|---------------------|--------------------------|--------------------------|
| 1 |  | <input checked="" type="checkbox"/> Identificación de los problemas, oportunidades y objetivos | 5 days? | 01-30-23 08:00 AM | 02-03-23 05:00 PM |
| 2 | | Entrevistas con Jefaturas de la organización | 2 days? | 01-30-23 08:00 AM | 01-31-23 05:00 PM |
| 3 |  | Recolección de la información y procesos | 3 days? | 02-01-23 08:00 AM | 02-03-23 05:00 PM |
| 4 | | <input type="checkbox"/> Análisis y Diseño del Sistema | 17 days? | 02-06-23 08:00 AM | 02-28-23 05:00 PM |
| 5 | | Identificar los requerimientos tecnológicos necesarios (Hardware, software, redes) | 5 days? | 02-06-23 08:00 AM | 02-10-23 05:00 PM |
| 6 | | Diseño de Mock Ups UX/UI | 4 days | 02-13-23 08:00 AM | 02-16-23 05:00 PM |
| 7 |  | Diseño de Diagrama UML | 3 days? | 02-17-23 08:00 AM | 02-21-23 05:00 PM |
| 8 | | Narración y Escenarios | 3 days? | 02-22-23 08:00 AM | 02-24-23 05:00 PM |
| 9 |  | Herramientas a utilizar y analisis del desarrollo | 4 days? | 02-23-23 08:00 AM | 02-28-23 05:00 PM |
| 10 | | <input type="checkbox"/> Desarrollo y documentación del Software | 56.5 days? | 03-01-23 08:00 AM | 05-18-23 01:00 PM |
| 11 | | Desarrollo de base de datos | 12 days? | 03-01-23 08:00 AM | 03-16-23 05:00 PM |
| 12 | | Inicio de Sesión, Registro de usuarios y opciones del sistema | 7 days? | 03-17-23 08:00 AM | 03-27-23 05:00 PM |
| 13 |  | Formularios para mantenimiento de pacientes | 5 days? | 03-28-23 08:00 AM | 04-03-23 05:00 PM |
| 14 |  | Formularios para mantenimiento de medicos | 1.667 days? | 03-27-23 08:00 AM | 03-28-23 02:20 PM |
| 15 |  | Formularios para mantenimiento citas | 1.667 days? | 04-03-23 08:00 AM | 04-04-23 02:20 PM |
| 16 |  | Formularios para mantenimiento areas | 1 day? | 04-10-23 08:00 AM | 04-10-23 05:00 PM |
| 17 |  | Formularios para mantenimiento de usuarios | 5 days? | 04-17-23 08:00 AM | 04-21-23 05:00 PM |
| 18 | | Generación de Informes del Sistema | 2.5 days? | 04-24-23 08:00 AM | 04-26-23 01:00 PM |
| 19 | | Redacción de Manual de Procedimientos | 5 days? | 04-26-23 01:00 PM | 05-03-23 01:00 PM |
| 20 |  | Creación de Manual de Programador (Diccionario de Datos) | 6 days? | 05-03-23 01:00 PM | 05-11-23 01:00 PM |
| 21 | | Manual de Usuario | 5 days? | 05-11-23 01:00 PM | 05-18-23 01:00 PM |
| 22 |  | <input type="checkbox"/> Prueba del Software | 11.017 da... | 05-18-23 01:00 PM | 06-02-23 01:07 PM |
| 23 | | Prueba de interfaz del sistema | 5.25 days? | 05-18-23 01:00 PM | 05-25-23 03:00 PM |
| 24 | | Prueba de los procesos principales del sistema | 4.267 days? | 05-25-23 03:00 PM | 06-01-23 08:07 AM |
| 25 | | Verificar informes generados en el sistema | 1.5 days? | 06-01-23 08:07 AM | 06-02-23 01:07 PM |
| 26 |  | <input type="checkbox"/> Implementación del sistema | 2.5 days? | 06-02-23 01:07 PM | 06-07-23 08:07 AM |
| 27 |  | Implementación del sistema | 2.5 days? | 06-02-23 01:07 PM | 06-07-23 08:07 AM |
| 28 |  | Instalación del Sistema | 0 days? | 06-02-23 01:07 PM | 06-02-23 01:07 PM |

ASIGNACIÓN DE ACTIVIDADES



CRONOGRAMA DE ACTIVIDADES

| ACTIVIDAD | Enero | | | | Febrero | | | | Marzo | | | | Abril | | | | Mayo | | | | Junio | | | |
|--|-------|----|----|----|---------|----|----|----|-------|----|----|----|-------|----|----|----|------|----|----|----|-------|----|----|----|
| | S1 | S2 | S3 | S4 | S1 | S2 | S3 | S4 | S1 | S2 | S3 | S4 | S1 | S2 | S3 | S4 | S1 | S2 | S3 | S4 | S1 | S2 | S3 | S4 |
| Entrevistas con Jefaturas de la organización | | | | | | | | | | | | | | | | | | | | | | | | |
| Recolección de la información y procesos | | | | | | | | | | | | | | | | | | | | | | | | |
| Análisis y Diseño del Sistema | | | | | | | | | | | | | | | | | | | | | | | | |
| Identificar los requerimientos tecnológicos necesarios (Hardware, software, redes) | | | | | | | | | | | | | | | | | | | | | | | | |
| Diseño de Mock Ups UX/UI | | | | | | | | | | | | | | | | | | | | | | | | |
| Diseño de Diagrama UML | | | | | | | | | | | | | | | | | | | | | | | | |
| Narración y Escenarios | | | | | | | | | | | | | | | | | | | | | | | | |
| Herramientas a utilizar | | | | | | | | | | | | | | | | | | | | | | | | |
| Desarrollo y documentación del software | | | | | | | | | | | | | | | | | | | | | | | | |
| Desarrollo de base de datos | | | | | | | | | | | | | | | | | | | | | | | | |
| Inicio de Sesión, Registro de usuarios y opciones del sistema | | | | | | | | | | | | | | | | | | | | | | | | |
| Formularios para mantenimiento de pacientes | | | | | | | | | | | | | | | | | | | | | | | | |
| Formularios para mantenimiento de medicos | | | | | | | | | | | | | | | | | | | | | | | | |
| Formularios para mantenimiento citas | | | | | | | | | | | | | | | | | | | | | | | | |
| Formularios para mantenimiento areas | | | | | | | | | | | | | | | | | | | | | | | | |
| Formularios para mantenimiento de usuarios | | | | | | | | | | | | | | | | | | | | | | | | |
| Generación de Informes del Sistema | | | | | | | | | | | | | | | | | | | | | | | | |
| Redacción de Manual de Procedimientos | | | | | | | | | | | | | | | | | | | | | | | | |
| Creación de Manual de Programador (Diccionario de Datos) | | | | | | | | | | | | | | | | | | | | | | | | |
| Manual de Usuario | | | | | | | | | | | | | | | | | | | | | | | | |
| Prueba del Software | | | | | | | | | | | | | | | | | | | | | | | | |
| Prueba de interfaz del sistema | | | | | | | | | | | | | | | | | | | | | | | | |
| Prueba de los procesos principales del sistema | | | | | | | | | | | | | | | | | | | | | | | | |
| Verificar informes generados en el sistema | | | | | | | | | | | | | | | | | | | | | | | | |
| Implementación del sistema | | | | | | | | | | | | | | | | | | | | | | | | |
| Implementación del sistema | | | | | | | | | | | | | | | | | | | | | | | | |
| Instalación del Sistema | | | | | | | | | | | | | | | | | | | | | | | | |

PRESUPUESTO DE COSTOS DE LA APLICACIÓN

Nombre del Proyecto: Presupuesto para la clínica dental comunitaria de Santa Ana
Del 30 de enero al 2 de Junio 2023

| | |
|------------------------|-----------|
| Utilidad | 20.00% |
| % Costo de Riesgo | 1.00% |
| Electricidad | 15% |
| Depreciación de Equipo | 0.10% |
| Horas disponibles | 704 |
| Precio de Laptop | \$ 900.00 |

| RECURSOS | HORAS SOLICITADAS | COSTO POR HORA | COSTO TOTAL | PRECIO NETO | COSTO NETO |
|----------------------|----------------------|-------------------|----------------|----------------|--------------------|
| Programador Senior 1 | 650 | \$ 7.50 | \$4,875.00 | \$ 9.08 | \$ 5,898.75 |
| Programador Junior 2 | 650 | \$ 3.33 | \$2,166.45 | \$ 4.03 | \$ 2,621.40 |
| Programador Junior 3 | 650 | \$ 3.33 | \$2,166.45 | \$ 4.03 | \$ 2,621.40 |
| Laptop 1 | 650 | | \$ 288.49 | 0 | 0 |
| Laptop 2 | 650 | | \$ 288.49 | 0 | 0 |
| Laptop 3 | 650 | | \$ 288.49 | 0 | 0 |
| Total | | | | | \$11,141.56 |

| CONCEPTO | TOTAL |
|------------------------|---------------------|
| Costo de Recursos | \$ 11,141.56 |
| Electricidad | \$ 1,671.23 |
| Depreciación de Equipo | \$ 865.48 |
| Hosting (Anual) | \$ 150.00 |
| Soporte Técnico | \$ 1,200.00 |
| Total | \$ 15,028.27 |

BIBLIOGRAFÍA

Creative commons. (2023). *Sobre las licencias*. Obtenido de <https://creativecommons.org/licenses/?lang=es>

GitHub. (2020). Obtenido de <https://github.com/>

Google. (s.f.). *Firebase*. Obtenido de https://console.firebase.google.com/u/0/?_gl=1*1g9x114*_ga*MTM1NjU4Mjk5Ni4xNjg0NzI0OTcz*_ga_CW55HF8NVT*MTY4NDcyNDk3My4xLjEuMTY4NDcyNTAxMy4wLjAuMA..

Llasera, J. (05 de 06 de 2021). *Mockup: qué es y para qué se usa*. Obtenido de Imborrable: <https://imborrable.com/blog/mockup-que-es/>

Portal de transparencias gobierno. (20 de Febrero de 2023). *PROCEDIMIENTO PARA EJECUTAR DEPRECIACIÓN*. (s. f.). Obtenido de <https://www.transparencia.gob.sv/institutions/dnm/documents/260776/download>

MANUAL DE USUARIO

CENTRO MEDICO

Pantalla de inicio o de bienvenida:

En el cual para avanzar a la pantalla de logueo se le da clic en el botón acceder:



Pantalla de inicio de sesión:

En esta pantalla aparecen los campos para poder ingresar al sistema en el cual es necesario ingresar correo y contraseña y si aún no se cuenta con un usuario registrado, se le puede dar clic a la opción de crear cuenta o registrar con Google.



Pantalla para crear cuenta:

Desde esta pantalla podremos crear el usuario.

FormularioCrearCuent

Centro Medico

Crear Cuenta

Ingresa tus datos personales

Ingresa tu Correo

Ingresa una Contraseña

Confirma tu Contraseña

Regístrate

O regístrate con

Continúa con Google

Continúa con Apple

© 2023 Centro Medico. Todos los derechos reservados.

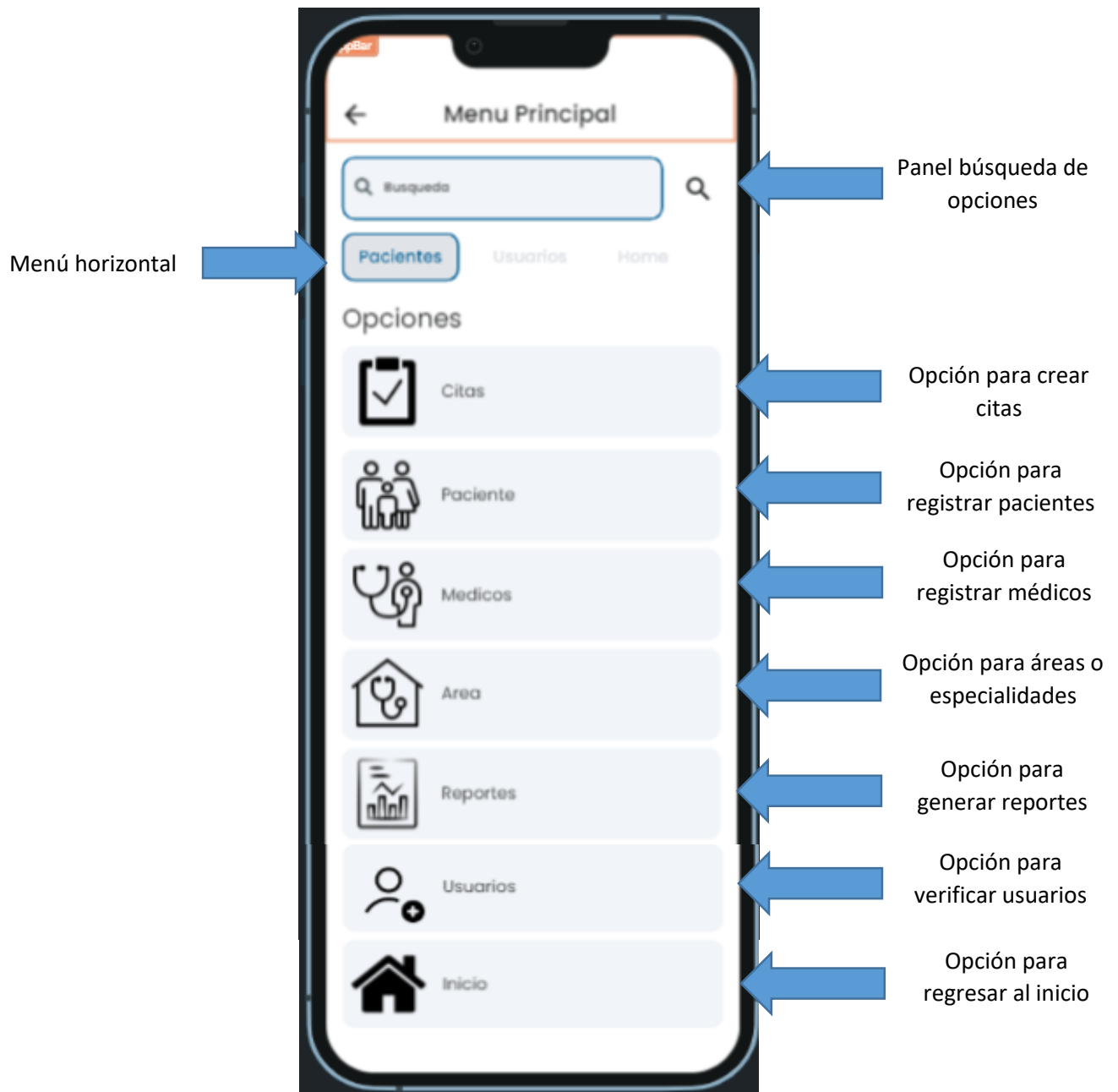
Campos para poder crear el usuario donde se pide el correo y la contraseña.

Botón para realizar la acción.

Opciones extras para registrarse.

Pantalla de menú principal:

Menú principal de la aplicación donde se visualizan las diferentes opciones que tiene el usuario para interactuar de acuerdo a sus necesidades:



Pantallas de administración de citas:

Pantallas para administrar todas las opciones de citas donde se puede ingresar cita , editar cita y eliminar cita:

The image displays three mobile application screens for appointment management, each with a blue header bar containing a medical cross icon and the text "Centro Medico".

- Editar Cita:** The screen features a title "Editar Cita" and seven input fields: "Paciente:", "Medico:", "Medicamentos:", "Síntomas:", "Fecha y Hora:", "Costo:", and "Tipo de Pago" (a dropdown menu). A blue "Editar" button is at the bottom.
- Eliminar Cita:** The screen features a title "Eliminar Cita" and the same seven input fields as the "Editar Cita" screen. A blue "Eliminar" button is at the bottom. An orange rectangular box highlights the header area.
- Ingreso de Cita:** The screen features a title "Ingreso de Cita" and the same seven input fields. A blue "Agregar" button is at the bottom.



Opción para editar
los campos
relacionados para
citas



Opción para revisar
la cita y si es
necesario poder
eliminarla



Opción para
ingresar citas según
la necesidad

Pantallas de administración de pacientes:

Pantallas para administrar todas las opciones de pacientes donde se puede ingresar paciente , editar paciente y eliminar paciente:

The image displays three mobile application screens for patient management, each with a blue header bar containing the 'Centro Medico' logo and name. The screens are titled 'Agregar Paciente', 'Eliminar Paciente', and 'Actualización de Paciente'. Each screen contains a series of white input fields with blue borders for the following fields: 'Nombre Completo:', 'Genero:', 'Fecha de Nacimiento:', 'Dirección:', 'Correo:', 'Alergias:', and 'Enfermedad:'. At the bottom of each screen is a blue button with white text: 'Agregar', 'Eliminar', and 'Actualizar' respectively.



Opción para
agregar los campos
relacionados para
ingresar un
paciente



Opción para revisar
los campos
relacionados para
decidir eliminar un
paciente



Opción para editar
los campos
relacionados para
pacientes

Pantallas de administración de médicos:

Pantallas para administrar todas las opciones de medicos donde se puede ingresar medico , editar medico y eliminar medico:

The image displays three mobile application screens for managing doctors, each with a blue header bar labeled 'Centro Medico'.

- Screen 1 (Left):** Titled 'Agregar Medico'. It contains seven text input fields: 'Area:', 'Nombre Completo:', 'Genero:', 'Dirección:', 'Correo:', 'Telefono:', and 'Especialidad:'. A blue button labeled 'Agregar' is at the bottom.
- Screen 2 (Middle):** Titled 'Modificar Medico'. It contains the same seven text input fields. A blue button labeled 'Eliminar' is at the bottom.
- Screen 3 (Right):** Titled 'Modificar Medico'. It contains the same seven text input fields. A blue button labeled 'Modificar' is at the bottom.



Opción para
agregar los campos
relacionados para
ingresar un medico



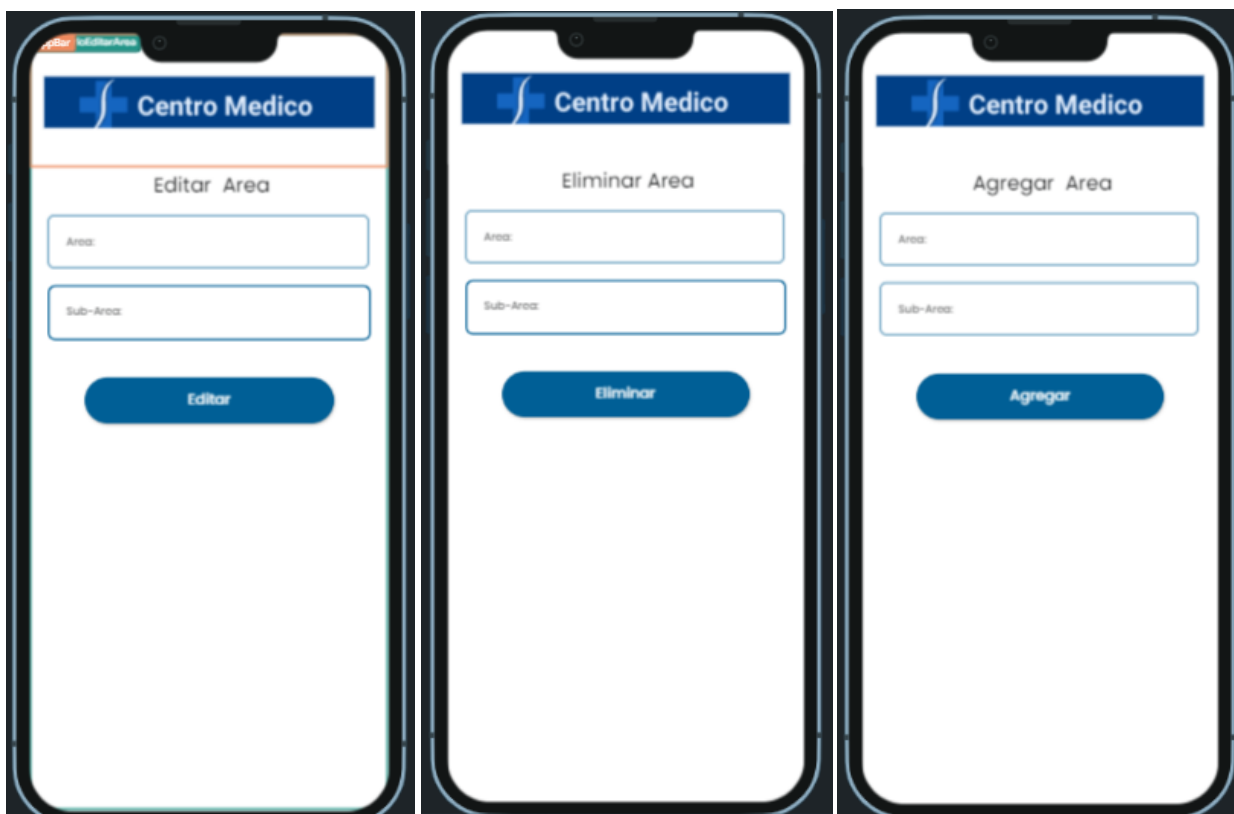
Opción para revisar
los campos
relacionados para
tomar la decisión
de eliminar un
doctor



Opción para editar
los campos
relacionados para
un doctor

Pantallas de administración de áreas:

Pantallas para administrar todas las opciones de las areas del centro medico donde se puede ingresar area , editar area y eliminar area:



Opción para editar
los campos
relacionados para
las áreas



Opción para revisar
los campos de la
administración de
áreas y decidir
eliminarlo



Opción para
agregar los campos
relacionados para
ingresar las áreas

Pantallas de administración de usuarios:

Pantallas para administrar todas las opciones de las áreas del centro medico donde se puede ingresar area , editar area y eliminar area:

The image displays three mobile application screens for user management, each with a blue header featuring a medical cross icon and the text "Centro Medico".

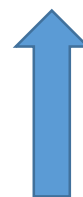
- Agregar Usuario:** This screen contains input fields for "Usuario", "Nombre Completo", "Genero", "Contraseña", "Correo", and "Tipo de Rol". A blue button labeled "Agregar" is positioned at the bottom.
- Editar Usuario:** This screen contains input fields for "Usuario", "Nombre Completo", "Genero", "Contraseña", "Correo", and "Tipo de Rol". A blue button labeled "Editar" is positioned at the bottom.
- Eliminar Usuario:** This screen contains input fields for "Usuario", "Nombre Completo", "Genero", "Contraseña", "Correo", and "Tipo de Rol". A blue button labeled "Eliminar" is positioned at the bottom.



Opción para
agregar los campos
relacionados para
el ingreso de
usuarios



Opción para editar
los campos
relacionados para
la administración
de usuarios



Opción para revisar
los campos en la
administración de
usuarios y decidir si
eliminarlo

Manual Técnico

- **StatefulWidget**

Clase que se utiliza para crear widgets cuyo estado puede cambiar durante la vida útil del widget, permiten actualizar y redibujar la interfaz de usuario en respuesta a cambios en el estado.

Es responsable de crear una instancia de su clase de estado y administrarla, mientras que el objeto State se encarga de almacenar y actualizar el estado del widget.

```
class InicioSesionWidget extends StatefulWidget {
  const InicioSesionWidget({Key? key}) : super(key: key);

  @override
  _InicioSesionWidgetState createState() => _InicioSesionWidgetState();
}

class _InicioSesionWidgetState extends State<InicioSesionWidget>
  with TickerProviderStateMixin {
  late InicioSesionModel _model;

  final scaffoldKey = GlobalKey<ScaffoldState>();

  final animationsMap = {
    'containerOnPageLoadAnimation': AnimationInfo(
      trigger: AnimationTrigger.onPageLoad,
      effects: [
        VisibilityEffect(duration: 1.ms),
        FadeEffect(
          curve: Curves.easeInOut,
          delay: 0.ms,
          duration: 400.ms,
          begin: 0,
          end: 1,
```

- **Void initState**

Se utiliza en clases de widgets StatefulWidget para realizar tareas de inicialización cuando el widget es creado y agregado al árbol de widgets.

En el método initState() se suelen inicializar variables y estados para el widget, también se puede realizar llamadas a servicios externos o API para obtener datos iniciales necesarios para el widget.

Se configuran controladores y suscripciones a eventos, como escuchadores de cambios o eventos de interacción

- **void dispose**

Se utiliza en clases de widgets StatefulWidget para realizar tareas de limpieza y liberación de recursos cuando el widget se elimina del árbol de widgets; se liberan los recursos asignados, como controladores, suscripciones a eventos, conexiones a servicios o API, o cualquier recurso que pueda estar en uso por el widget.

También se cancelan tareas o temporizadores activos para evitar fugas de memoria o comportamientos inesperados.

```
@override
void initState() {
  super.initState();
  _model = createModel(context, () => InicioSesionModel());

  _model.emailAddressController1 ??= TextEditingController();
  _model.passwordController1 ??= TextEditingController();
  _model.emailAddressController2 ??= TextEditingController();
  _model.passwordController2 ??= TextEditingController();
}

@override
void dispose() {
  _model.dispose();

  super.dispose();
}
```

- **Widget build(BuildContext context)**

Se utiliza en clases de widgets StatefulWidget o StatelessWidget para construir y retornar la representación visual del widget en la interfaz de usuario, también se puede diseñar y configurar la apariencia visual del widget y generar automáticamente el código correspondiente en el método build().

```
super.initState();
_model = createModel(context, () => MenuModel());

_model.busquedaController ??= TextEditingController();
}

@override
void dispose() {
  _model.dispose();

  super.dispose();
}

@override
Widget build(BuildContext context) {
  return GestureDetector(
    onTap: () => FocusScope.of(context).requestFocus(_model.unfocusNode),
    child: Scaffold(
      key: scaffoldKey,
      backgroundColor: FlutterFlowTheme.of(context).secondaryBackground,
      appBar: AppBar(
        backgroundColor: FlutterFlowTheme.of(context).secondaryBackground,
        automaticallyImplyLeading: false,
        leading: FlutterFlowIconButton(
          borderColor: Colors.transparent,
          borderRadius: 30,
          borderWidth: 1,
          buttonSize: 60,
          icon: Icon(
            Icons.arrow_back_rounded,
            color: FlutterFlowTheme.of(context).primaryText,
```

- **createState()**

Se utiliza en una clase StatefulWidget para crear una instancia de su clase de estado asociada. La clase de estado es responsable de almacenar y actualizar el estado del widget y es donde se implementa la lógica y la funcionalidad del widget; está asociada al diseño y las configuraciones realizadas en la interfaz visual de la herramienta.

```
12
13 class FormularioIngresoCitasWidget extends StatefulWidget {
14   const FormularioIngresoCitasWidget({Key? key}) : super(key: key);
15
16   @override
17   _FormularioIngresoCitasWidgetState createState() =>
18     _FormularioIngresoCitasWidgetState();
19 }
20
21 class _FormularioIngresoCitasWidgetState
22   extends State<FormularioIngresoCitasWidget> {
23   late FormularioIngresoCitasModel _model;
24
25   final scaffoldKey = GlobalKey<ScaffoldState>();
26
27   @override
28   void initState() {
29     super.initState();
30     _model = createModel(context, () => FormularioIngresoCitasModel());
31
32     _model.pacienteController ??= TextEditingController();
33     _model.medicoController ??= TextEditingController();
34     _model.medicamentosController ??= TextEditingController();
35     _model.sintomasController ??= TextEditingController();
```

- **Evento de Toque onPressed**


Se utiliza en muchos widgets para definir una acción o una función que se ejecutará cuando el usuario interactúe con el widget presionándolo. Es una forma de capturar eventos de interacción y responder a ellos

Se asigna una acción o función que se ejecutará cuando el usuario interactúe con un widget presionándolo.


```
Align(  
  alignment: AlignmentDirectional(0, 0.05),  
  child: Padding(  
    padding: EdgeInsetsDirectional.fromSTEB(0, 24, 0, 0),  
    child: FFBUTTONWidget(  
      onPressed: () {  
        print('Button pressed ...');  
      },  
      text: 'Agregar',  
      options: FFBUTTONOptions(  
        width: 270,  
        height: 50,  
        padding: EdgeInsetsDirectional.fromSTEB(0, 0, 0, 0),  
        iconPadding: EdgeInsetsDirectional.fromSTEB(0, 0, 0, 0),  
        color: Color(0xFF005F96),  
        textStyle: FlutterFlowTheme.of(context)  
          .titleMedium  
          .override(  
            fontFamily: 'Poppins',  
            color: FlutterFlowTheme.of(context).primaryBtnText,  
          ),  
        elevation: 2,  
        borderSide: BorderSide(  
          color: Colors.transparent,  
          width: 1,  
        ),  
      ),  
    ),  
  ),  
)
```


Configuración con Firebase


Apps para Android

 **Proyecto Centro Medico Android**
com.mycompany.projectocentromedico

Apps de Apple


 **Proyecto Centro Medico iOS**
com.mycompany.projectocentromedico


Apps web


 **Proyecto Centro Medico Web**
Web App

Configuración del SDK

¿Necesitas volver a configurar los SDK de Firebase en tu app? Revisa las instrucciones de configuración del SDK o descarga el archivo de configuración con las claves y los identificadores de tu app.


 [Ver las instrucciones del SDK](#)

 [google-services.json](#)

ID de la app 



1:960867439904:android:9e19be77525873dbcdf345

Sobrenombre de la app

Proyecto Centro Medico Android 

Nombre del paquete


com.mycompany.projectocentromedico

| Huellas digitales del certificado SHA  | Tipo  |
|---|--|
| 9b:bb:31:86:b2:18:9e:f1:6d:8f:27:2b:da:b5:93:62:c3:95:cb:b4 | SHA-1 |
| 07:b9:2e:02:75:d5:e2:f3:b3:51:ef:6c:44:c5:a6:74:c3:6b:e6:f4:36:9f:01:e0:13:34:af:a3:7b:6d:6e:ef | SHA-256 |

[Agregar huella digital](#)

Nombre del proyecto


CentroMedico 

ID del proyecto 

centromedico-cbbd4

Número de proyecto 

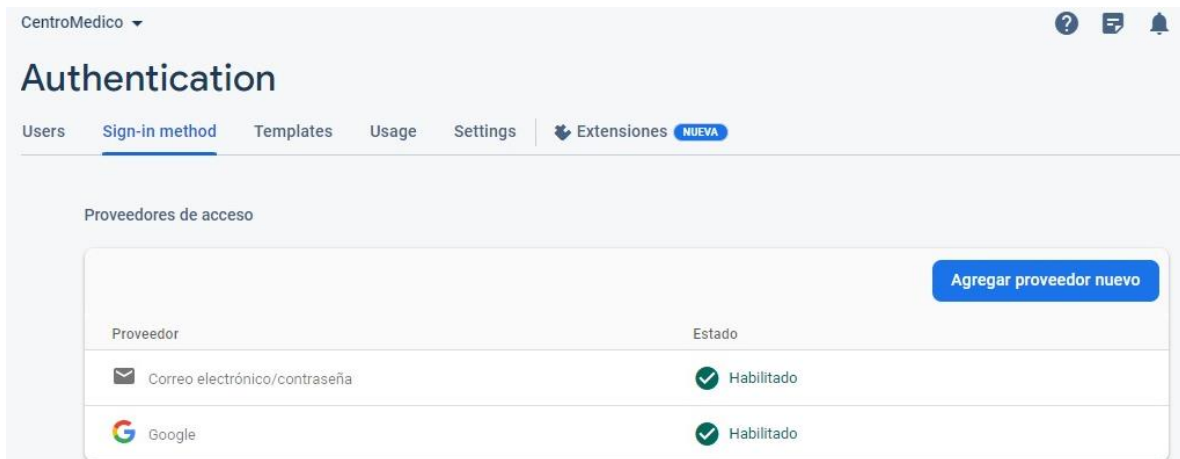
960867439904

Ubicación predeterminada de los recursos de GCP 

Aún sin seleccionar 

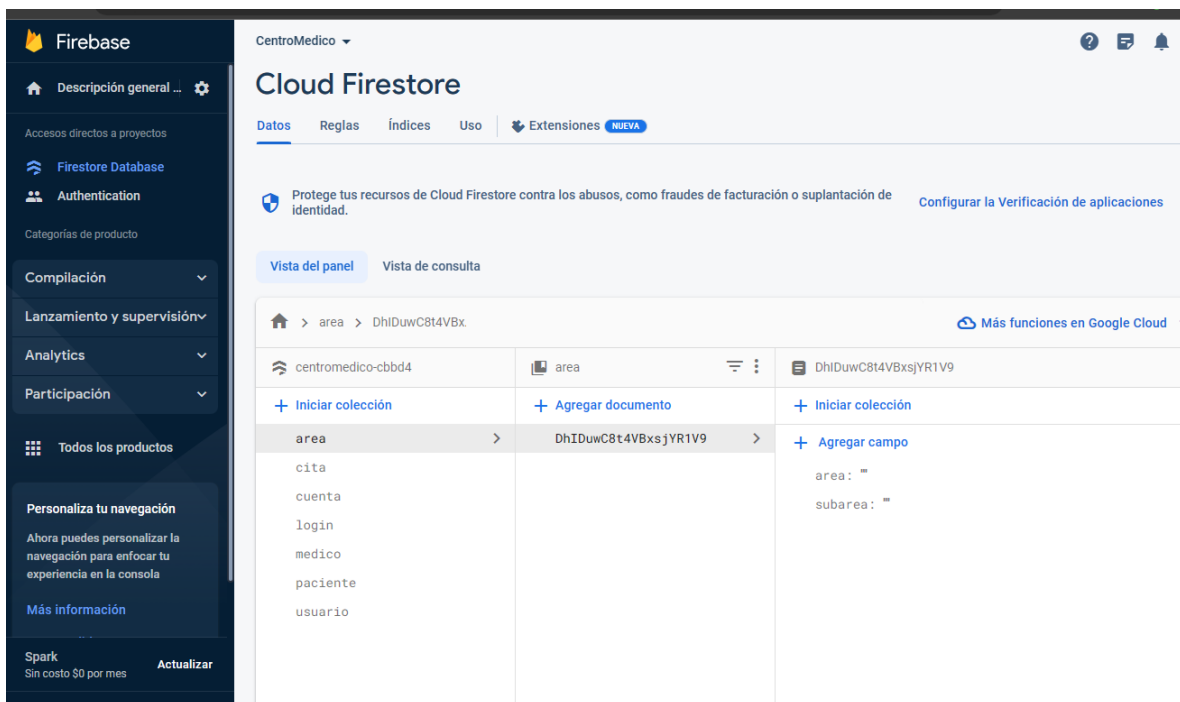
Clave de API web

No hay clave de API web para este proyecto



Se crea BD utilizando Firebase con la opción de Firestore que permite realizar consultas más complejas y rápidas, y el escalamiento se ajusta a un nivel más alto que Realtime Database.

La Base de Datos se llama: CentroMedico



Se crearon diferentes colecciones que son lo equivalente a crear una tabla

- Area
- Cita
- Cuenta
- Login
- Medico
- Paciente
- Usuario

También se agregaron los campos a cada colección, que se establecen diferente tipo de datos que son similares a los de otros gestores de BD

| Tipo de datos | Orden de clasificación | Notas |
|------------------|---|--|
| Arreglo | Según los valores de los elementos | <p>Un arreglo no puede contener otro valor de arreglo como uno de sus elementos.</p> <p>En los arreglos, los elementos conservan la posición que se les asignó. Para ordenar dos o más arreglos, se debe realizar con base en los valores de sus elementos.</p> <p>Cuando se comparan dos arreglos, se comparan los primeros elementos de cada uno de ellos. Si son iguales, se deben comparar los segundos elementos y así sucesivamente, hasta que se encuentre una diferencia. Si un arreglo no tiene más elementos que comparar, pero es igual al otro arreglo hasta ese punto, debe ordenarse antes del arreglo más largo.</p> <p>Por ejemplo, $[1, 2, 3] < [1, 2, 3, 1] < [2]$. El arreglo $[2]$ tiene el valor más alto en el primer elemento. El arreglo $[1, 2, 3]$ tiene elementos iguales a los primeros tres de $[1, 2, 3, 1]$, pero su longitud es menor.</p> |
| Punto geográfico | Por latitud, luego por longitud | <p>Por el momento, no recomendamos usar este tipo de datos debido a las limitaciones de consultas. En general, es mejor almacenar la latitud y longitud como campos numéricos separados. Si tu app necesita consultas geográficas simples basadas en la distancia, revisa Consultas geográficas.</p> |
| Número entero | Numérico | 64 bits, firmado |
| Mapa | Por claves y, luego, por valor | <p>Representa un objeto incorporado en un documento. Cuando se indexa, puedes hacer consultas por subcampos. Si excluyes este valor de la indexación, también se excluyen todos los subcampos.</p> <p>Siempre se realiza el ordenamiento por clave. Por ejemplo, si escribes <code>{c: "foo", a: "bar", b: "qux"}</code>, el mapa se ordena por clave y se guarda como <code>{a: "bar", b: "qux", c: "foo"}</code>.</p> <p>Los campos de mapas se ordenan por clave y se comparan con pares clave-valor. Primero se comparan las claves y, luego, los valores. Si los primeros pares clave-valor son iguales, se compararán los siguientes, y así sucesivamente. Si dos mapas comienzan con los mismos pares clave-valor, se considerará la longitud del mapa. Por ejemplo, estos mapas se ordenan en sentido ascendente:</p> <pre>{a: "aaa", b: "baz"} {a: "foo", b: "bar"} {a: "foo", b: "bar", c: "qux"} {a: "foo", b: "baz"} {b: "aaa", c: "baz"} {c: "aaa"}</pre> |
| Nulo | Ninguno | — |
| Referencia | Por elementos de ruta de acceso (colección, ID de documento, colección, ID de documento...) | Por ejemplo, <code>projects/[PROJECT_ID]/databases/[DATABASE_ID]/documents/[DOCUMENT_PATH]</code> . |