



**PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR**

**Unidad Académica de Formación Técnica y Tecnológica – PUCE TEC**

**SISTEMA WEB DE AGENDAMIENTO DE CITAS MÉDICAS PARA EL  
CENTRO DE ESTIMULACIÓN TEMPRANA FONOLANDIA**

**Proyecto de titulación previo a la obtención del título de: Tecnólogo Superior en  
Desarrollo de Software**

**Autores: Kevin Ricardo Cañizares Guanoluisa, Elicio Alejandro Perrazo**

**Cartagena**

**Tutor: Ing. Andrés Sebastián Páez Mora**

**Quito, Ecuador**

**2024**

## **Dedicatoria**

Este proyecto es el resultado del esfuerzo compartido y el compromiso de todos los que nos han acompañado en esta aventura, Gracias por ser parte fundamental de este logro.

## Tabla de contenidos

<b>Dedicatoria</b> .....	2
<b>Lista de tablas</b> .....	4
<b>Lista de figuras</b> .....	5
<b>Capítulo I Levantamiento de Requisitos y Diseño del Sistema</b> .....	12
1.    Objetivos.....	12
1.1.    Objetivo General.....	12
1.2.    Objetivos Específicos.....	12
2.    Análisis de Requisitos.....	12
2.1.    Recolección de requisitos.....	12
2.2.    Diseño del Sistema.....	15
2.3.    Modelado.....	17
<b>Capítulo II Construcción del Sistema</b> .....	44
3.    Sprints.....	44
3.1.    Sprint 1: Realización de Mockups del Sistema.....	44
3.2.    Sprint 2: Desarrollo Frontend.....	46
3.3.    Sprint 3: Desarrollo Backend.....	48
3.4.    Sprint 4: Integración de Componentes y Pruebas Finales.....	50
<b>Capítulo III</b> .....	52
4.    Pruebas y Estabilización.....	52
4.1.    Introducción.....	52
4.2.    Tipos de Pruebas.....	52
4.3.    Pruebas.....	53
<b>Conclusiones:</b> .....	57
<b>Referencias bibliográficas:</b> .....	59

## **Lista de tablas**

Tabla 1. Gestión Pacientes.....	17
Tabla 2. Gestión Pacientes.....	18
Tabla 3. Gestión Pacientes.....	19
Tabla 4. Gestión Citas .....	21
Tabla 5. Gestión Citas .....	22
Tabla 6. Gestión Citas .....	24
Tabla 7. Recordatorio de Citas .....	25
Tabla 8. Recordatorio de Citas .....	26
Tabla 9. Gestión Especialista .....	27
Tabla 10. Gestión Especialista.....	28
Tabla 11. Gestión Especialista.....	29

## Lista de figuras

Figura 1. Gestión de Pacientes .....	17
Figura 2. Gestión Citas .....	20
Figura 3. Recordatorio de Citas .....	25
Figura 4. Gestión Especialista .....	27
Figura 5. Modelo Entidad Relación.....	31
Figura 6. Pantalla de Inicio.....	32
Figura 7. Crear Paciente .....	33
Figura 8. Modificar paciente ingreso historia clínica .....	34
Figura 9. Modificar paciente ingreso datos .....	34
Figura 10. Eliminar paciente ingreso historia clínica .....	35
Figura 11. Eliminar paciente .....	35
Figura 12. Crear Especialista.....	36
Figura 13. Modificar especialista ingreso cédula de identidad .....	36
Figura 14. Modificar Especialista.....	37
Figura 15. Eliminar especialista ingreso cédula de identidad .....	37
Figura 16. Eliminar especialista .....	38
Figura 17. Asignar horario ingreso de cédula de identidad .....	38
Figura 18. Asignar turno.....	39
Figura 19. Crear citas ingreso historia clínica del paciente .....	39
Figura 20. Crear cita elegir especialista.....	40
Figura 21. Crear cita elegir especialidad .....	40
Figura 22. Crear cita elegir día .....	41
Figura 23. Modificar cita ingreso número de cita .....	41
Figura 24. modificar cita ingreso de nuevos datos .....	42
Figura 25. Eliminar cita ingreso de número de cita.....	42
Figura 26. Eliminar cita .....	43
Figura 27. Consultar cita ingreso de número de historia clínica .....	43
Figura 28. Consultar cita datos actualizados .....	44
Figura 29. Login .....	47
Figura 30. Página Principal .....	47
Figura 31. Base de Datos.....	49
Figura 32. Archivo api.....	49
Figura 33. Implementación Express.JS .....	50

Figura 34. Query de pruebas de integración.....	53
Figura 35. Resultado query de integración.....	53
Figura 36. Ruta de información de cada paciente .....	53
Figura 37. URL.....	54
Figura 38. Validación .....	54
Figura 39. Respuesta .....	54
Figura 40. Validador de rutas API.....	54
Figura 41. Módulo Modificar Paciente .....	55
Figura 42. Ingreso de datos en módulo Modificar Paciente.....	55
Figura 43. Formulario de Modificar Paciente .....	56
Figura 44. Resultado Obtenido.....	56

## DECLARACIÓN y AUTORIZACIÓN

Yo, **Kevin Ricardo Cañizares Guanoluisa** con C.I. 1720270618 autor del trabajo de aplicativo web intitulado: **“Sistema Web de Agendamiento de Citas Médicas para el Centro de Estimulación Temprana Fonolandia”**, previa a la obtención del título de **Tecnólogo Superior en Desarrollo de Software** en la Unidad Académica de Formación Técnica y Tecnológica PUCE TEC:

1.- Declaro conocer la obligación de la Pontificia Universidad Católica del Ecuador, según el artículo 144 de la Ley Orgánica de Educación Superior, de entregar a la SENESCYT en formato digital una copia del trabajo de graduación para integrarse al Sistema Nacional de Información de la Educación Superior del Ecuador para su difusión pública respetando los derechos de autor.

2.- Autorizo a la Pontificia Universidad Católica del Ecuador a difundir a través de sitio web de la Biblioteca de la PUCE el referido trabajo de titulación, respetando las políticas de propiedad intelectual de Universidad.

Quito, 8 de agosto del 2024

Kevin Ricardo Cañizares Guanoluisa  
C.I. 1720270618

## DECLARACIÓN y AUTORIZACIÓN

Yo, **Elicio Alejandro Perrazo Cartagena** con C.I. 172336921-9 autor del trabajo de aplicativo web intitulado: **“Sistema Web de Agendamiento de Citas Médicas para el Centro de Estimulación Temprana Fonolandia”**, previa a la obtención del título de **Tecnólogo Superior en Desarrollo de Software** en la Unidad Académica de Formación Técnica y Tecnológica PUCE TEC:

1.- Declaro conocer la obligación de la Pontificia Universidad Católica del Ecuador, según el artículo 144 de la Ley Orgánica de Educación Superior, de entregar a la SENESCYT en formato digital una copia del trabajo de graduación para integrarse al Sistema Nacional de Información de la Educación Superior del Ecuador para su difusión pública respetando los derechos de autor.

2.- Autorizo a la Pontificia Universidad Católica del Ecuador a difundir a través de sitio web de la Biblioteca de la PUCE el referido trabajo de titulación, respetando las políticas de propiedad intelectual de Universidad.

Quito, 8 de agosto del 2024

Elicio Alejandro Perrazo Cartagena

C.I. 1723369219-9

## **Agradecimientos**

Quiero expresar mi agradecimiento a todas aquellas personas que me han apoyado en la realización de esta tesis.

En primer lugar, a mi familia, por su amor incondicional, comprensión y por creer en mí en todo momento. Gracias por su paciencia y por brindarme todo su apoyo emocional a lo largo de la carrera y mientras realizaba.

A mis profesores quienes me acompañaron a lo largo de estos dos años en la carrera, quienes me brindaron las herramientas necesarias y el conocimiento para enfrentar este gran desafío académico.

A los que contribuyeron, de alguna manera, a la culminación de esta tesis, mi agradecimiento.

Kevin Cañizares

Quisiera expresar mi más sincero agradecimiento a todas y cada una de las personas que han sido parte fundamental en este proceso y cuya ayuda a sido crucial para alcanzar la culminación de esta etapa.

A mis padres y hermana, mi profunda gratitud por su apoyo incondicional durante estos dos años. Han sido pilares esenciales en mi vida, y gracias a su constante ánimo y aliento, hemos podido superar desafíos y concluir con éxito esta fase de mi formación. A mis amigos más cercanos, por su compañía leal y honesta. Su presencia ha sido una fuente de motivación y fortaleza a lo largo de este camino.

A mis profesores, quienes han sido faros de orientación y conocimiento. No solo he adquirido conocimientos valiosos y herramientas, sino también importantes principios y valores que han enriquecido mi experiencia académica y profesional.

A todos, muchas gracias por su apoyo y contribuciones invaluable.

Alejandro Perrazo

## **Introducción**

Frente al acelerado avance tecnológico en varios campos administrativos y empresariales, es común observar que la gestión de la industria se tecnifica para un mejor desenvolvimiento en los servicios prestados y en la influencia en su atención en los nichos en que se desenvuelven.

Por eso, el proyecto solucionará un problema específico de muchas empresas actualmente. El sistema que se desarrollará es un sistema web de agendamiento de citas diseñada específicamente para el centro de estimulación temprana Fonolandia.

Este sistema busca automatizar las operaciones administrativas del centro, además de facilitar la programación de citas, el sistema tendrá una funcionalidad de notificación automática, que enviará correos electrónicos de recordatorio a especialistas y clientes un día antes de cada cita programada. Esta característica asegura una mejor preparación y asistencia a las citas, reduciendo el número de inasistencias y optimizando la planificación diaria del centro. Se espera que esta solución digital no solo mejore la gestión de tiempo del personal, sino que también ofrezca una experiencia más ordenada y proactiva a los clientes, contribuyendo significativamente a la eficiencia operativa del centro.

## **Capítulo I Levantamiento de Requisitos y Diseño del Sistema**

### **1. Objetivos**

#### **1.1.Objetivo General**

Desarrollar un sistema web de agendamiento de citas médicas para el centro de estimulación temprana Fonolandia, el cual automatice y optimice las operaciones administrativas de agendamiento, mejorando la gestión de citas y la experiencia del usuario final.

#### **1.2.Objetivos Específicos**

- Realizar un análisis detallado de los requisitos funcionales y no funcionales para asegurar que el sistema de agendamiento de citas cumpla con las necesidades del centro de estimulación temprana Fonolandia.
- Diseñar una interfaz de usuario de manera que sea intuitiva y accesible para el usuario, permitiendo así una gestión eficiente de pacientes, citas y especialistas.
- Implementar una funcionalidad la cual sea de notificaciones automáticas por correo electrónico para reducir las inasistencias de pacientes y mejorar de esta manera la planificación diaria del centro de estimulación temprana Fonolandia.
- Crear un módulo en el sistema que permita el registro, modificación y eliminación de la información de los pacientes, garantizando que el manejo de los datos del centro sea eficiente y organizado.
- Asegurar que el sistema sea escalable y capaz de manejar un incremento en la demanda sin comprometer su rendimiento.

### **2. Análisis de Requisitos**

#### **2.1.Recolección de requisitos**

Para el levantamiento de requisitos del sistema de agendamiento de citas, se realizó varias entrevistas de manera virtual vía Zoom, en las cuales se pudieron evidenciar la necesidad de la Dra. Dayana Montenegro, la cual es propietaria de Fonolandia, el sistema de agendamiento de citas es su prioridad, donde se remarcó sobre todo la importancia del envío de correos electrónicos recordando las citas asignadas tanto a los especialistas como a los clientes.

### 2.1.1. Requisitos Funcionales

En el desarrollo de software los requisitos funcionales son muy importantes pues así vamos a poder especificar los comportamientos y las capacidades que un sistema de tener, para así llegar a cumplir con las expectativas del usuario final. Estos requisitos detallan las funciones y servicios que el software debe proporcionar y definen cómo el sistema debe responder a ciertas entradas y comportarse en diversas situaciones (Sommerville, 2011). Así, los requisitos funcionales son un componente fundamental ya que proporcionan una comprensión compartida entre todas las partes sobre lo que el sistema debe llegar a lograr.

- **Registro de Pacientes:** El sistema debe permitir el registro de nuevos pacientes con información básica como nombre, edad, género, teléfono, correo electrónico y datos del representante.
- **Historial de Citas:** El sistema debe mantener un registro histórico de todas las citas agendadas, incluyendo detalles como fecha, hora, especialista asignado y paciente.
- **Agendamiento de Citas:** El sistema debe permitir a la doctora agendar nuevas citas para los pacientes, especificando la fecha, hora, especialidad médica y especialista asignado. Debe ser posible verificar la disponibilidad de citas para un especialista y una fecha específica.
- **Gestión de Recursos:** Debe ser posible asignar recursos adicionales a las citas, como salas de consulta específicas.
- **Modificación y Cancelación de Citas:** El sistema debe permitir la modificación o cancelación de citas existentes por parte de la doctora.
- **Recordatorios de Citas:** El sistema debe enviar recordatorios automáticos a los pacientes y al personal médico antes de la fecha de la cita, mediante correo electrónico.

### 2.1.2. Requisitos No Funcionales

Estos requisitos son de suma importancia para definir las características y criterios que afectan la operación y el rendimiento del sistema. Estos requisitos abarcan aspectos como la usabilidad, la seguridad, el rendimiento, la fiabilidad y la escalabilidad del sistema. Según (Sommerville, 2011), los requisitos no funcionales “especifican criterios que pueden usarse para juzgar la operación de un sistema, en lugar de comportamientos específicos”. Estos requisitos aseguran que se cumpla las expectativas de calidad y rendimiento en diferentes condiciones y escenarios.

- **Seguridad:** El sistema debe garantizar la confidencialidad de la información del paciente.
- **Rendimiento:** El sistema debe ser capaz de manejar múltiples accesos concurrentes y mantener una respuesta rápida, incluso durante períodos de alta demanda, para evitar retrasos en el agendamiento de citas.
- **Usabilidad:** La interfaz de usuario debe ser intuitiva y fácil de usar tanto para la doctora como para el personal administrativo, minimizando la necesidad de capacitación adicional.
- **Escalabilidad:** El sistema debe ser escalable para poder manejar un aumento en el número de pacientes y médicos sin comprometer su rendimiento.
- **Integración:** Debe ser posible integrar el sistema de agendamiento de citas con otros sistemas existentes en el centro de salud, como el sistema de historias clínicas electrónicas.

## **2.2.Diseño del Sistema**

El diseño del sistema de agendamiento de citas para Fonolandia se basó en los diferentes requisitos tanto funcionales como no funcionales los que fueron obtenidos del análisis de los objetivos de la organización. Este diseño busca ofrecer una solución eficiente y completa para los requerimientos del usuario final, asegurando así un uso óptimo y rendimiento confiable.

### **2.2.1. Interfaz de Usuario**

El diseño de interfaz de usuario se centra en la experiencia y usabilidad del usuario, se partió de un diseño previamente creado con Mockitt, asegurando una navegación fácil de comprender.

La interfaz gráfica constará con un Login inicial para el usuario administrador y tendrá una interfaz sencilla e intuitiva donde podrá gestionar pacientes, gestionar las citas o gestionar a los especialistas, donde cada botón llevará a un formulario para el módulo seleccionado. Desde la interfaz de gestionar pacientes se podrá registrar un nuevo paciente, modificar un paciente o eliminar paciente mediante. La interfaz de gestión de citas tendrá una funcionalidad parecida al módulo de pacientes, donde se podrá agendar una cita, modificar una cita y eliminar una cita mediante la utilización de formularios. Por último, estará el módulo de gestionar especialistas, donde se podrá registrar un especialista, modificar al especialista, eliminar un especialista y asignar horario de atención al especialista.

### **2.2.2. Arquitectura del sistema**

El sistema estará compuesto por componentes principales como: la interfaz gráfica, la lógica de programación y la base de datos. La arquitectura del sistema de agendamiento de citas para Fonolandia se diseñó empleando el patrón de diseño Modelo-Vista-Controlador (MVC), que facilita la separación de responsabilidades y mejora la mantenibilidad y escalabilidad del sistema (Gamma, 1994). Esta arquitectura utilizada está compuesta por los siguientes componentes principales:

- **Modelo:** Se utiliza PostgreSQL para la gestión de datos, Según (Kroening, 2020), “PostgreSQL es un sistema de base de datos relacional de código abierto que ofrece características robustas para el manejo de datos, garantizando la integridad y consistencia de información” (p. 134).

La capacidad de PostgreSQL es fundamental para asegurar integridad en los datos.

- **Vista:** fue desarrollada con React y CoreUI, que proporciona una interfaz de usuario moderna y a su vez eficiente. (Jackson, 2019) destaca que “React es una biblioteca de JavaScript que facilita la creación de interfaces de usuario interactivas y dinámicas, mientras CoreUI ofrece componentes visuales que mejoran la experiencia de usuario y la usabilidad” (p. 75). Esta combinación ayuda a una mejor optimización de la experiencia de usuario y la flexibilidad de la interfaz gráfica.
- **Controlador:** Implementado con Express.js, maneja la lógica de negocio y la comunicación entre la vista y el modelo. Según (Hoffman, 2018), “Express.js es un framework de Node.js que facilita la construcción de aplicaciones web robustas y escalables, permitiendo una gestión eficiente de las solicitudes y respuestas del servidor” (p.56). Esto asegura un rendimiento eficiente y una integración fluida entre los componentes del sistema.

### 2.3.Modelado

- **Diagramas UML:** Casos de uso sistema de agendamiento de citas centro médico Fonolandia.

#### CASO DE USO 1. GESTIÓN DE PACIENTES

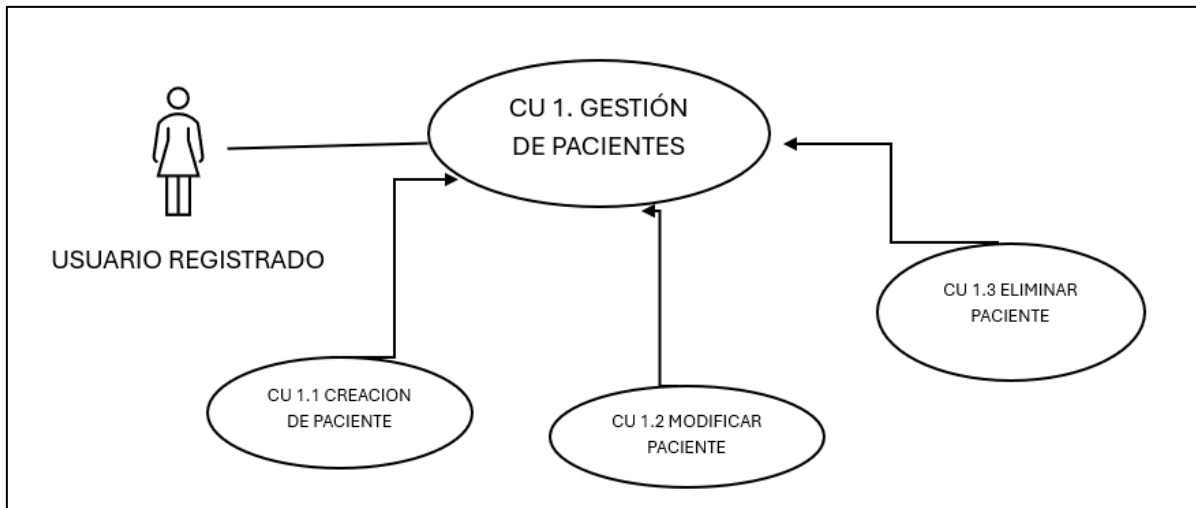


Figura 1. Gestión de Pacientes

Casos de uso extendido gestión de pacientes:

Tabla 1. Gestión Pacientes

Nombre	CU 1.1 Creación de paciente
Descripción	Permite la creación de un nuevo paciente en el sistema
Actores	Usuario Registrado
Precondiciones	Ingreso de credenciales en login.
Requisitos no Funcionales	Usabilidad

Flujo de Eventos	<ol style="list-style-type: none"> <li>1. Usuario registrado selecciona en la interfaz la opción de gestión de paciente.</li> <li>2. Sistema despliega menú con las opciones de: creación de paciente, modificar paciente, eliminar paciente.</li> <li>3. Usuario registrado selecciona la opción creación de paciente.</li> <li>4. Sistema muestra formulario con datos requeridos a ingresar (nombre, edad, género, correo electrónico, dirección, teléfono).</li> <li>5. Usuario registrado ingresa los datos del paciente.</li> <li>6. Usuario registrado selecciona la opción de guardar cambios.</li> <li>7. Sistema muestra mensaje de si está segura de guardar los cambios</li> <li>8. Sistema procede a validar la información y guardar en la base de datos</li> </ol>
Flujo alternativo	6.1 Si la doctora no está segura de guardar los cambios se regresa al paso 5

Tabla 2. Gestión Pacientes

Nombre	CU 1.2 Modificar Paciente
Descripción	La doctora puede modificar campos del registro del paciente sea cambio de domicilio, teléfono de contacto o correo electrónico, etc.
Actores	Usuario Registrado
Precondiciones	Que el paciente este registrado en el sistema
Requisitos no Funcionales	Ninguno
Flujo de Eventos	<ol style="list-style-type: none"> <li>1. Usuario registrado ingresa a la opción de gestión de paciente.</li> </ol>

	<ol style="list-style-type: none"> <li>2. Sistema despliega menú con las opciones de: creación de paciente, modificar paciente, eliminar paciente.</li> <li>3. Usuario registrado selecciona opción de modificar paciente.</li> <li>4. Sistema despliega campo donde se encuentra la información actual del paciente.</li> <li>5. Usuario registrado puede modificar los campos que desee y aceptar.</li> <li>6. Sistema muestra mensaje si está segura de guardar los cambios realizados.</li> <li>7. Sistema da a la opción de aceptar para guardar los cambios realizados.</li> <li>8. Sistema da mensaje de datos guardados satisfactoriamente.</li> </ol>
Flujo alternativo	5.1 Se dará una opción que diga si el usuario registrado este de acuerdo en guardar los cambios realizados, caso contrario regresa al paso 4

Tabla 3. Gestión Pacientes

Nombre	CU 1.3 Eliminar Paciente
Descripción	Se podrá eliminar los datos le paciente del sistema
Actores	Usuario registrado
Precondiciones	Que el paciente este registrado en el sistema
Requisitos no Funcionales	Ninguno
Flujo de Eventos	<ol style="list-style-type: none"> <li>1. Usuario registrado ingresa a la opción de gestión de paciente.</li> <li>2. Sistema despliega menú con las opciones de: creación de paciente, modificar paciente, eliminar paciente.</li> <li>3. Usuario registrado selecciona opción de eliminar paciente.</li> </ol>

	<ol style="list-style-type: none"> <li>4. Sistema despliega campo para ingresar la identificación del paciente y buscarlo en la base de datos.</li> <li>5. Usuario registrado ingresa el ID del paciente</li> <li>6. Sistema muestra la información del paciente</li> <li>7. Usuario registrado da a la opción de eliminar registro de paciente.</li> <li>8. Usuario registrado da la opción de eliminar y se borran los registros de la base de datos</li> </ol>
Flujo alternativo	7.1 Sistema muestra una ventana emergente dando el mensaje de si desea realizar la eliminación

CASO DE USO 2. GESTIÓN DE CITAS

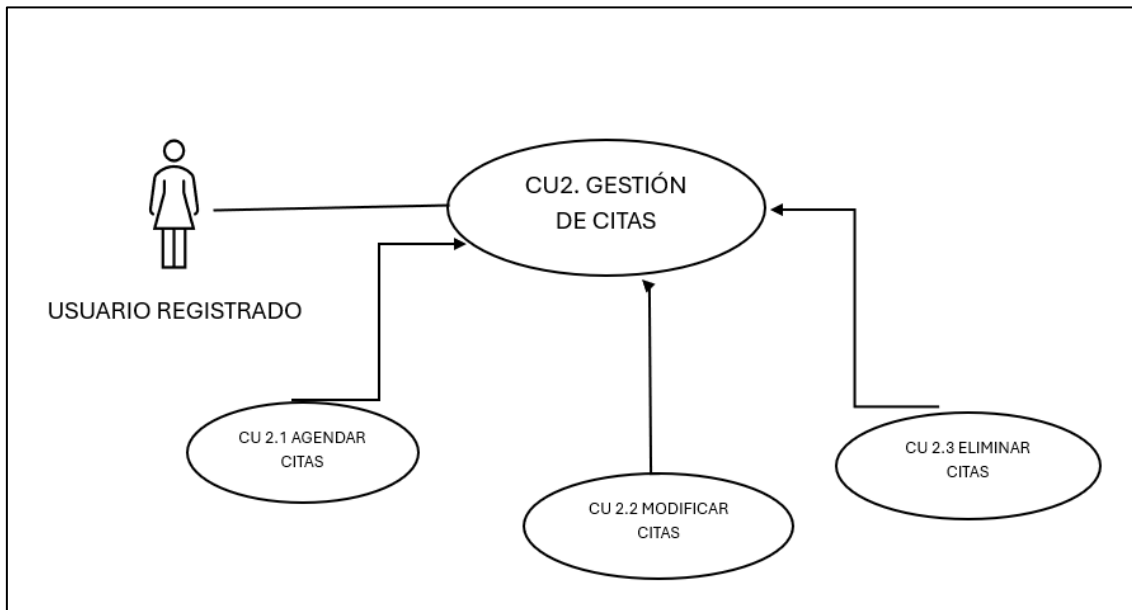


Figura 2. Gestión Citas

## Casos de uso extendido gestión de citas

Tabla 4. Gestión Citas

Nombre	CU 2.1 Agendar Citas
Descripción	Aquí se realiza el agendamiento de las citas médicas
Actores	Usuario Registrado
Precondiciones	Ingreso de las credenciales del usuario registrado, que el paciente este registrado
Requisitos no Funcionales	Usabilidad
Flujo de Eventos	<ol style="list-style-type: none"> <li>1. Usuario registrado ingresa a la opción de Gestión de citas.</li> <li>2. Sistema despliega el menú con las opciones: Agendar citas, Modificar citas, Eliminar citas</li> <li>3. Usuario registrado ingresa a la opción de Agendar citas.</li> <li>4. Sistema muestra ventana para ingresar ID o nombre del paciente.</li> <li>5. Usuario registrado ingresa el ID o nombre del paciente y especialidad a tratar.</li> <li>6. Sistema muestra un calendario con los horarios disponibles para agendar con el especialista asignado.</li> <li>7. Usuario registrado elige el horario de agendamiento de cita.</li> <li>8. Sistema muestra la opción de las salas disponibles en el horario elegido.</li> <li>9. Usuario registrado selecciona la sala disponible.</li> <li>10. De ser necesario el usuario registrado puede ingresar algún recurso adicional a la información de la cita médica, sea algún equipo especial como observación.</li> </ol>

	<p>11. Usuario registrado guarda la información en la base de datos.</p> <p>12. Sistema muestra mensaje si desea mantener los datos actuales.</p> <p>13. Usuario registrado da en aceptar y la información es guardada exitosamente</p>
Flujo alternativo	<p>4.1 Si algún campo mandatorio no está llenado correctamente regresa al paso 3</p> <p>7.1 SI no hay disponibilidad de horario se regresa al paso 6</p>

Tabla 5. Gestión Citas

Nombre	CU 2.2 Modificar Citas
Descripción	EL usuario registrado podrá modificar el día y la hora de las citas que requieran ser modificadas
Actores	Usuario Registrado
Precondiciones	Que el paciente tenga agendada una cita médica previamente
Requisitos no Funcionales	Ninguno
Flujo de Eventos	<ol style="list-style-type: none"> <li>1. Usuario registrado ingresa a la opción de Agendamiento de citas.</li> <li>2. Sistema despliega el menú con las opciones: Agendar citas, Modificar citas, Eliminar citas.</li> <li>3. Usuario registrado ingresa a la opción de modificar cita.</li> <li>4. Sistema muestra modal para ingresar ID o nombre del paciente</li> <li>5. Usuario registrado ingresa el número de identificación del paciente.</li> <li>6. Sistema muestra la información actual de la cita del paciente.</li> <li>7. Usuario registrado elige opción de modificar.</li> </ol>

	<ol style="list-style-type: none"> <li>8. Sistema despliega un calendario con los horarios disponibles para agendar con el especialista asignado.</li> <li>9. Usuario registrado elige el nuevo horario de agendamiento de cita.</li> <li>10. Sistema muestra la opción de las salas disponibles en el horario elegido.</li> <li>11. Usuario registrado selecciona la sala disponible y acepta.</li> <li>12. De ser necesario la Doctora puede ingresar algún recurso adicional a la información de la cita médica, sea algún equipo especial como observación.</li> <li>13. Usuario registrado guarda la información en la base de datos.</li> <li>14. Sistema muestra mensaje si está de acuerdo en los cambios realizados.</li> <li>15. Usuario registrado Acepta y la información es guardada en la base de datos.</li> </ol>
Flujo alternativo	<ol style="list-style-type: none"> <li>5.1 Si el paciente no está registrado aparecerá el error de que no está registrado en el sistema y regresa a paso 4.</li> <li>11.1 SI no hay disponibilidad de horario se regresa al paso 9.</li> </ol>

Tabla 6. Gestión Citas

Nombre	CU 2.3 Eliminar Citas
Descripción	Se puede realizar la eliminación definitiva de la cita médica
Actores	Usuario Registrado
Precondiciones	Que el paciente tenga una cita médica agendada
Requisitos no Funcionales	Ninguno
Flujo de Eventos	<ol style="list-style-type: none"> <li>1. Usuario registrado ingresa a la opción de Agendamiento de citas.</li> <li>2. Sistema despliega el menú con las opciones: Agendar citas, Modificar citas, Eliminar citas.</li> <li>3. Usuario registrado ingresa a la opción de eliminar cita.</li> <li>4. Sistema muestra modal para ingresar ID o nombre del paciente.</li> <li>5. Usuario registrado ingresa el ID o nombre del paciente.</li> <li>6. Sistema muestra información de la cita actual del paciente y de su especialista.</li> <li>7. Usuario registrado da la opción de Eliminar.</li> <li>8. Sistema muestra mensaje “Esta segura con eliminar esta cita”.</li> <li>9. Usuario registrado da opción de Aceptar y la cita médica queda borrada de la base de datos.</li> </ol>
Flujo alternativo	5.1 Si el número de identificación del paciente no está correctamente llenado, aparece mensaje de error y regresa al paso 4.

### CASO DE USO 3. RECORDATORIO DE CITAS

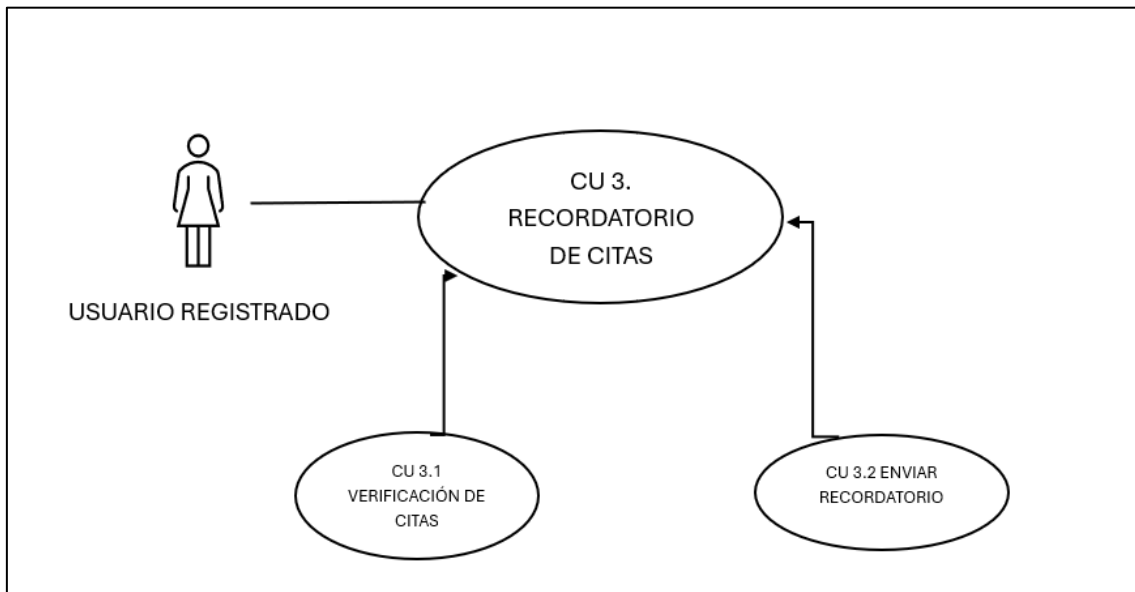


Figura 3. Recordatorio de Citas

#### Casos de uso extendido recordatorio de citas

Tabla 7. Recordatorio de Citas

Nombre	CU 3.1 VERIFICACION DE CITAS
Descripción	La doctora puede realizar una consulta rápida de las citas agendadas que desee buscar.
Actores	Usuario registrado
Precondiciones	El paciente debe tener agendada una cita
Requisitos no Funcionales	Usabilidad
Flujo de Eventos	<ol style="list-style-type: none"> <li>1. Usuario registrado ingresa en opción Recordatorio de citas.</li> <li>2. Sistema despliega menú con las opciones de Verificación de citas y Enviar Recordatorio.</li> <li>3. Usuario registrado selecciona la opción de Verificación de citas.</li> <li>4. Sistema despliega un modal para ingresar ID o nombre del paciente.</li> <li>5. Usuario registrado ingresa ID o nombre del paciente.</li> </ol>

	<p>6. Al dar aceptar el sistema despliega la información de la cita o citas que el paciente tenga pendientes.</p> <p>7. Usuario registrado puede dar la opción de Salir.</p>
Flujo alternativo	4.1 Si algún de los campos mandatorios no están correctamente aparece error y regresa al paso 3

Tabla 8. Recordatorio de Citas

Nombre	CU 3.2 Enviar Recordatorio
Descripción	El usuario registrado puede enviar recordatorios al mail registrado del representante del paciente para indicar que su cita médica esta próxima a cumplirse
Actores	Usuario Registrado
Precondiciones	Que el paciente este con una cita agendada activa
Requisitos no Funcionales	Ninguno
Flujo de Eventos	<ol style="list-style-type: none"> <li>1. Usuario registrado ingresa en opción Recordatorio de citas.</li> <li>2. Sistema despliega menú con las opciones de Verificación de citas y Enviar Recordatorio.</li> <li>3. Usuario registrado selecciona la opción de Enviar Recordatorio</li> <li>4. Usuario registrado ingresa el ID o nombre del paciente.</li> <li>5. Al dar aceptar el sistema despliega la información de la cita o citas que el paciente tenga pendientes, con la información respectiva de contacto.</li> <li>6. Al dar la opción de enviar recordatorio el sistema enviará un mail al contacto que este guardado en el sistema.</li> </ol>
Flujo alternativo	4.1 Si algún de los campos mandatorios no están correctamente aparece error y regresa al paso 3.

## CASO DE USO 4. GESTIÓN ESPECIALISTAS

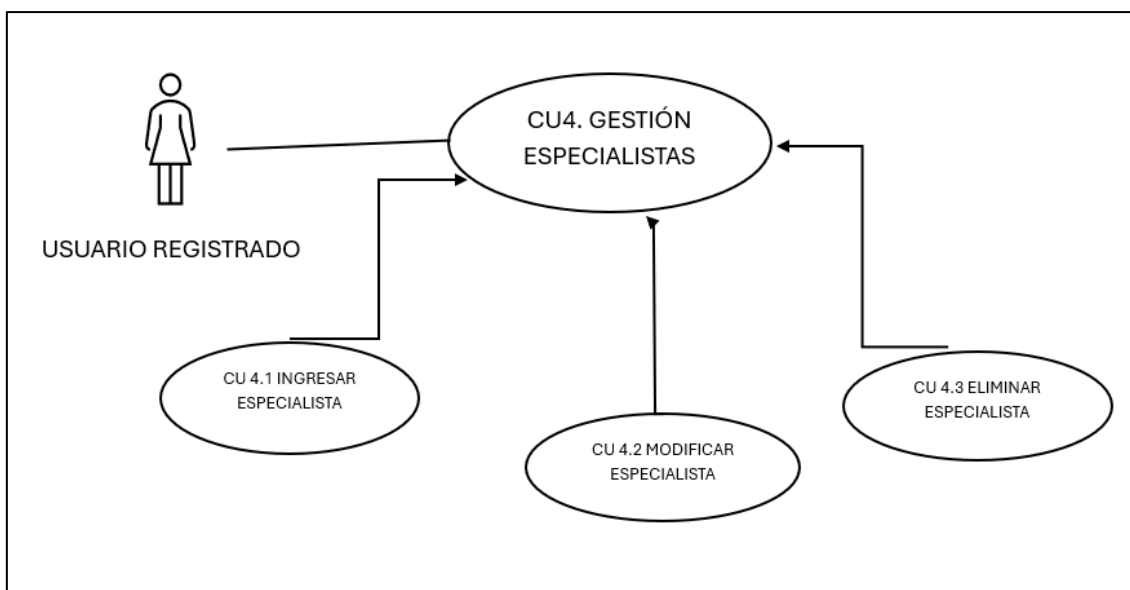


Figura 4. Gestión Especialista

### Casos de uso extendido

Tabla 9. Gestión Especialista

Nombre	CU 4.1 Ingresar Especialista
Descripción	Ingreso de la información del médico especialista
Actores	Usuario Registrado
Precondiciones	Ingreso del usuario al sistema con sus credenciales
Requisitos no Funcionales	Usabilidad
Flujo de Eventos	<ol style="list-style-type: none"> <li>1. Usuario registrado ingresa en opción Gestión Especialistas.</li> <li>2. Sistema despliega menú con las opciones: Ingresar Especialista, Modificar Especialista, Eliminar Especialistas.</li> <li>3. Usuario registrado ingresa en opción Ingresar Especialista.</li> <li>4. Sistema muestra formulario para ingresar los datos del nuevo especialista.</li> <li>5. Usuario registrado ingresa los datos del nuevo especialista.</li> </ol>

	<p>6. Al elegir la opción Aceptar el sistema despliega modal con mensaje de si está de acuerdo en guardar los cambios.</p> <p>7. Usuario registrado acepta y los datos del nuevo especialista son guardados satisfactoriamente</p>
Flujo alternativo	6.1 Si algún de los campos mandatorios no están correctamente aparece error y regresa al paso 5.

Tabla 10. Gestión Especialista

Nombre	CU 4.2 Modificar Especialista
Descripción	El usuario registrado podrá modificar los datos de los doctores especialistas
Actores	Usuario Registrado
Precondiciones	Que el Especialista este registrado en el sistema
Requisitos no Funcionales	Ninguno
Flujo de Eventos	<ol style="list-style-type: none"> <li>1. Usuario registrado ingresa en opción Gestión Especialistas.</li> <li>2. Se despliega menú con las opciones: Ingresar Especialista, Modificar Especialista, Eliminar Especialista.</li> <li>3. Usuario registrado ingresa en opción Modificar Especialista.</li> <li>4. Sistema despliega opción para Ingresar ID o nombre del especialista.</li> <li>5. Usuario registrado ingresa datos pedidos.</li> <li>6. Aparece información actual del especialista.</li> <li>7. Usuario registrado ingresa los datos a modificar.</li> <li>8. Sistema muestra la opción Aceptar y se despliega modal con mensaje de si está de acuerdo en guardar los cambios.</li> </ol>

	9. Usuario registrado Acepta y los datos del nuevo especialista son guardados satisfactoriamente
Flujo alternativo	7.1 Si algún de los campos mandatorios no están correctamente aparece error y regresa al paso 6.

Tabla 11. Gestión Especialista

Nombre	CU 4.3 Eliminar Especialista
Descripción	Se puede borrar los datos del especialista.
Actores	Usuario Registrado
Precondiciones	Que el especialista este registrado en el sistema
Requisitos no Funcionales	Ninguno
Flujo de Eventos	<ol style="list-style-type: none"> <li>1. Usuario registrado ingresa en opción Gestión Doctores.</li> <li>2. Sistema despliega menú con las opciones: Ingresar Especialista, Modificar Especialista, Eliminar Especialista.</li> <li>3. Usuario registrado ingresa en opción Eliminar Especialista.</li> <li>4. Sistema muestra opción para ingresar ID o nombre del especialista.</li> <li>5. Usuario registrado ingresa ID o nombre del especialista</li> <li>6. Sistema despliega la información actual del especialista.</li> <li>7. Usuario registrado tiene la opción de eliminar.</li> <li>8. Sistema muestra modal con mensaje “Desea eliminar los datos”.</li> <li>9. Usuario registrado da en aceptar y la información del especialista es eliminada de la base de datos.</li> </ol>
Flujo alternativo	3.1 Si algún de los campos mandatorios no están correctamente aparece error y regresa al paso 2.

- **Modelo Entidad Relación:**

**Tabla Especialistas:** Esta tabla servirá para registrar a cada especialista que trabaja en el centro con la información básica para el correcto funcionamiento del sistema.

ID\_Especialista(PK), Nombres, Apellidos, Email y Num\_Telefonico

**Terapias\_Ofrecidas:** Esta tabla permitirá registrar todas las terapias que se ofrecen actualmente en el centro.

ID\_Terapia(PK), Nombre

**Salas:** Esta tabla permitirá registrar cada sala del centro con su respectivo número.

ID\_Sala(PK), Num\_Sala

**Horarios:** Esta tabla nos permitirá registrar las distintas franjas horarias con las que se trabajará en el centro.

ID\_Horario(PK), Hora\_Inicio, Hora\_Fin

**Pacientes:** Esta tabla nos permitirá registrar la información relacionada con los pacientes y, al tratarse de menores de edad, también la información del representante legal.

ID\_Paciente(PK), Nombres, Apellidos, Representante, Email, Num\_Telefonico

**Especialista\_Terapia:** Esta tabla nos permitirá vincular a cada especialista con la terapia que puede realizar, ya que no todos los especialistas podrán realizar todas las terapias.

ID\_Especialista(FK Especialistas), ID\_Terapia(FK Terapias\_Ofrecidas)

**Terapia\_Sala:** Esta tabla servirá para seleccionar ciertas salas que servirán únicamente para un tipo de terapia, como puede ser la sala de hidroterapia, que debido a los instrumentos especiales requeridos tendrá que ser siempre la misma.

ID\_Terapia (FK Terapias\_Ofrecidas), ID\_Sala (FK Salas)

**Especialista\_Horario\_Terapia:** Esta tabla nos ayudará a evitar que alguna cita no sea posible realizar ya que sirve para marcar que un especialista tiene una terapia asignada en un cierto horario y también que una sala en ese horario está ocupada.

ID\_Especialista (FK Especialistas, Clave única), ID\_Terapia (FK Terapias\_Ofrecidas), ID\_Horario (FK Horarios, Clave única), ID\_Sala (FK Salas)

**Citas:** Esta tabla permitirá tener el histórico de las citas agendadas y también tener la información de cada cita.

ID\_Cita (PK), ID\_Paciente (FK Pacientes), ID\_Terapia (FK Terapias\_Ofrecidas), ID\_Horario (FK Horarios), Fecha.

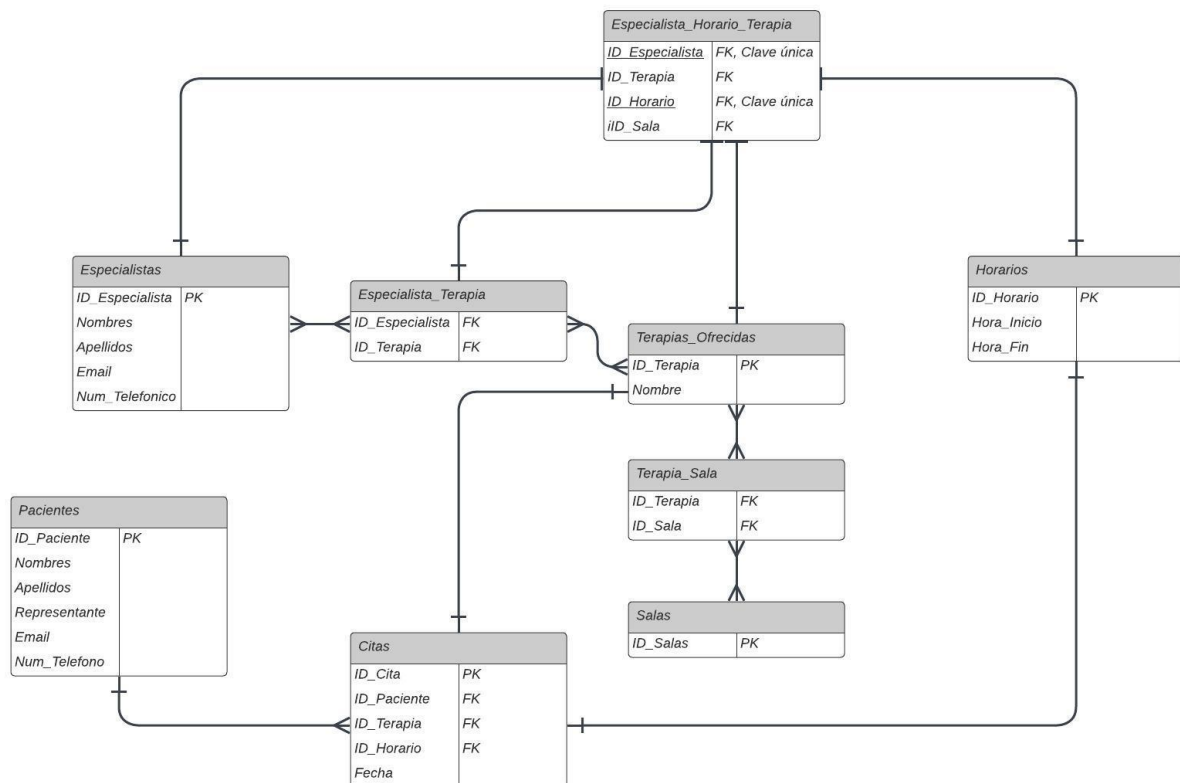


Figura 5. Modelo Entidad Relación

- **Prototipado de diseño:**

Se inicia a desarrollar el prototipo de las diferentes interfaces las cuales van a tener la interacción según las necesidades del usuario, se utiliza para esto la herramienta Mockiit, los prototipos son los siguientes:

Pantalla de Inicio donde se puede tener las distintas opciones las cuales son las necesidades básicas:

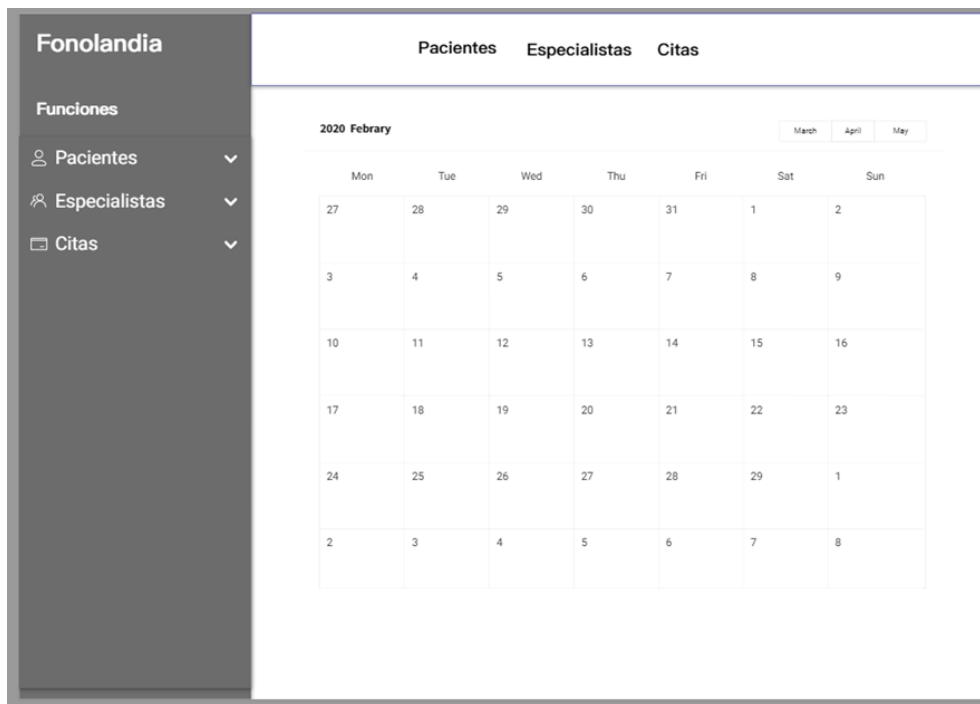


Figura 6. Pantalla de Inicio

Pacientes/ Crear Pacientes: aquí el usuario puede crear el paciente para el ingreso a la base de datos con los campos necesarios para tener la información adecuada.

The screenshot displays the 'Fonolandia' web application interface. On the left is a dark sidebar with the title 'Fonolandia' and a 'Funciones' menu containing 'Pacientes', 'Especialistas', and 'Citas'. The 'Pacientes' section is expanded, showing 'Crear Paciente' as the active option, along with 'Modificar Paciente' and 'Eliminar Paciente'. The main content area has a top navigation bar with 'Pacientes', 'Especialistas', and 'Citas', and a breadcrumb 'Home / Crear Paciente'. The form contains five input fields: 'Nombre del Paciente', 'Cédula del Paciente', 'Nombre del Representante', 'Teléfono', and 'Correo Electrónico', each with a 'Please enter' placeholder. A 'Confirmar Datos' button is located at the bottom of the form.

Figura 7. Crear Paciente

Pacientes/ Modificar Pacientes: se tiene la opción de poder modificar algún dato según sea la necesidad del usuario.

The screenshot shows the 'Fonolandia' web application interface. On the left is a dark sidebar with the title 'Fonolandia' and a 'Funciones' menu containing 'Pacientes', 'Especialistas', and 'Citas'. The 'Pacientes' section is expanded, showing 'Crear Paciente', 'Modificar Paciente' (highlighted), and 'Eliminar Paciente'. The main content area has a top navigation bar with 'Pacientes', 'Especialistas', and 'Citas'. Below this, the page title is 'Home / Modificar Paciente'. A section titled 'Buscar por historia clínica' contains a text input field with the placeholder 'Please enter' and a 'Buscar' button.

Figura 8. Modificar paciente ingreso historia clínica

The screenshot shows the 'Fonolandia' web application interface, similar to Figure 8. The sidebar and navigation are the same. The main content area is titled 'Home / Modificar Paciente'. Below the title, there are five form fields, each with a label and a 'Please enter' placeholder: 'Nombre del Paciente', 'Cédula del Paciente', 'Nombre del Representante', 'Teléfono', and 'Correo Electrónico'. At the bottom of the form is a 'Confirmar Cambios' button.

Figura 9. Modificar paciente ingreso datos

Pacientes/ Eliminar Paciente: el usuario tiene la posibilidad de eliminar los datos del paciente.

The screenshot shows the 'Fonolandia' web application interface. On the left is a dark sidebar with the title 'Fonolandia' and a 'Funciones' menu. The menu items are: 'Pacientes' (with a blue arrow), 'Crear Paciente', 'Modificar Paciente', 'Eliminar Paciente' (highlighted in blue), 'Especialistas' (with a dropdown arrow), and 'Citas' (with a dropdown arrow). The main content area has a top navigation bar with 'Pacientes', 'Especialistas', and 'Citas' tabs, and a home icon. Below the navigation, the page title is 'Home / Eliminar Paciente'. The main section is titled 'Buscar por historia clínica' and contains a single text input field with the placeholder 'Please enter' and a 'Buscar' button below it.

Figura 10. Eliminar paciente ingreso historia clínica

The screenshot shows the 'Fonolandia' web application interface, similar to Figure 10. The sidebar and navigation are identical. The main content area is titled 'Home / Eliminar Paciente'. Below the title, there are five form fields, each with a label and a placeholder 'Please enter': 'Nombre del Paciente', 'Cédula del Paciente', 'Nombre del Representante', 'Teléfono', and 'Correo Electrónico'. At the bottom of the form is a button labeled 'Eliminar Paciente'.

Figura 11. Eliminar paciente

Especialistas/Crear Especialista: el usuario puede ingresar un especialista que es el encargado de la atención al paciente en el horario definido después.

The screenshot shows the 'Fonolandia' web application interface. On the left is a dark sidebar with a menu titled 'Funciones' containing 'Pacientes', 'Especialistas', 'Crear Especialista', 'Modificar Especialista', 'Eliminar Especialista', and 'Asignar Turno'. Below this is a 'Citas' section. The main content area has a top navigation bar with 'Pacientes', 'Especialistas', and 'Citas'. The page title is 'Home / Crear Especialista'. The form includes the following fields: 'Nombre Especialista' (text input), 'Cédula del Especialista' (text input), 'Terápias de Especialización' (checkboxes for: Evaluación Inicial, Terapia del Lenguaje, Estimulación Temprana, Psicología Educativa, Clínica Psicología, Terapia Física, Neuropsicología, Tutorías Académicas, Terapia Ocupacional, Audometrías, Hidroterapias, Terapia Conductual, Psicopedagogía, Psicología Infantil), 'Teléfono' (text input), and 'Correo electrónico' (text input). A 'Confirmar Datos' button is at the bottom.

Figura 12. Crear Especialista

Modificar Especialistas: si se necesita un cambio en los datos del especialista o en los horarios de atención.

The screenshot shows the 'Fonolandia' web application interface. The sidebar is the same as in Figure 12, but 'Modificar Especialista' is highlighted. The main content area has a top navigation bar with 'Pacientes', 'Especialistas', and 'Citas'. The page title is 'Home / Modificar Especialista'. The form includes a 'Buscar por cédula' section with a text input field and a 'Buscar' button.

Figura 13. Modificar especialista ingreso cédula de identidad

**Fonolandia**

Pacientes Especialistas Citas

Home / Modificar Especialista

**Nombre Especialista**  
Please enter

**Cédula del Especialista**  
Please enter

**Terápias de Especialización**

- Evaluación Inicial
- Terapia del Lenguaje
- Estimulación Temprana
- Psicología Educativa
- Clínica Psicología
- Terapia Física
- Neuropsicología
- Tutorías Académicas
- Terapia Ocupacional
- Audimetrías
- Hidroterapias
- Terapia Conductual
- Psicopedagogía
- Psicología Infantil

**Teléfono**  
Please enter

**Correo electrónico**  
Please enter

**Confirmar Cambios**

Figura 14. Modificar Especialista

Especialistas/ Eliminar Especialista: si algún especialista llegara a terminar su relación con el centro médico el usuario tiene la posibilidad de eliminar el especialista.

**Fonolandia**

Pacientes Especialistas Citas

Home / Eliminar Especialista

**Buscar por cédula**  
Please enter

**Buscar**

Figura 15. Eliminar especialista ingreso cédula de identidad

**Fonolandia**

Pacientes Especialistas Citas

Home / Eliminar Especialista

**Nombre Especialista**  
Please enter

**Cédula del Especialista**  
Please enter

**Terápias de Especialización**

- Evaluación Inicial
- Terapia del Lenguaje
- Estimulación Temprana
- Psicología Educativa
- Clínica Psicología
- Terapia Física
- Neuropsicología
- Tutorías Académicas
- Terapia Ocupacional
- Audimetrías
- Hidroterapias
- Terapia Conductual
- Psicopedagogía
- Psicología Infantil

**Teléfono**  
Please enter

**Correo electrónico**  
Please enter

**Eliminar Especialista**

Figura 16. Eliminar especialista

Especialistas/ Asignar Horarios: en esta sección se realizar la asignación de horarios de atención del especialista para guardar en la base de datos y poder saber la disponibilidad de los especialistas.

**Fonolandia**

Pacientes Especialistas Citas

Home / Asignar Horario

**Buscar por cédula**  
Please enter

**Buscar**

Figura 17. Asignar horario ingreso de cédula de identidad

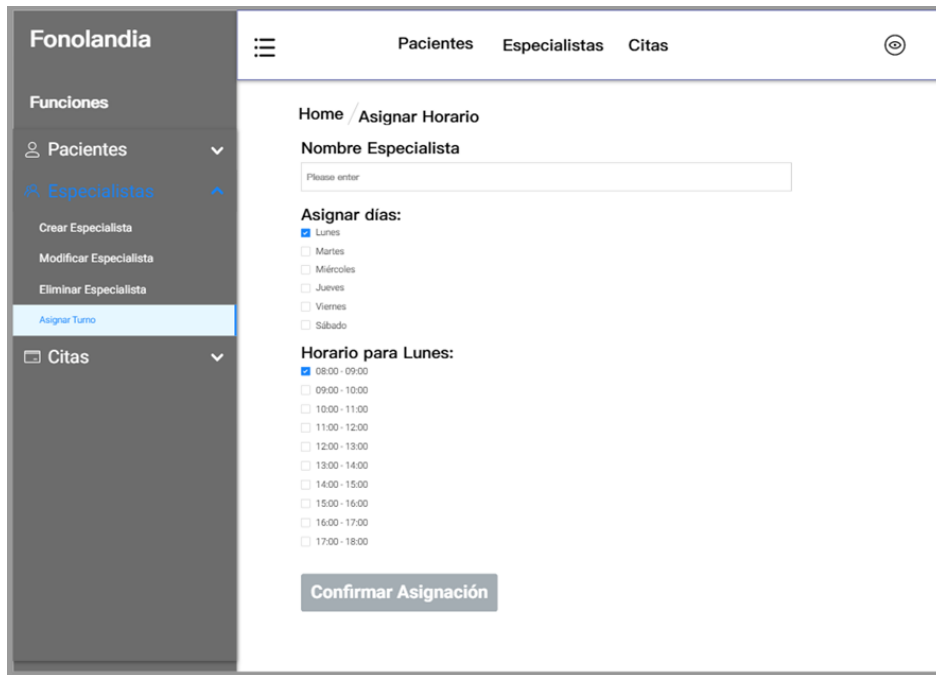


Figura 18. Asignar turno

Citas/ Crear Citas: en esta opción el usuario puede ya ingresar a crear la cita médica del paciente, sabiendo la historia clínica del paciente y la especialidad a la cual se la desea agendar.

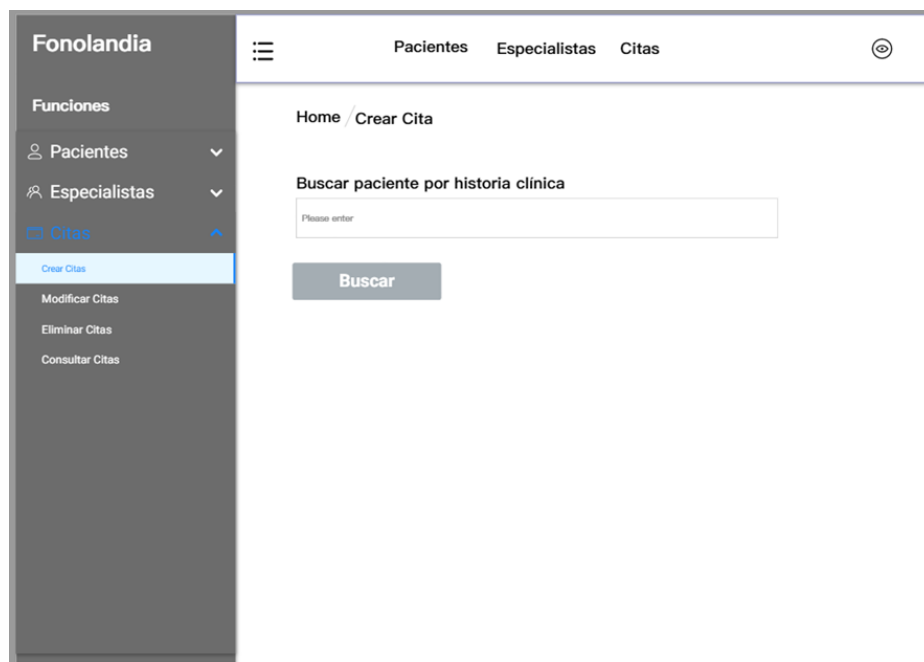


Figura 19. Crear citas ingreso historia clínica del paciente

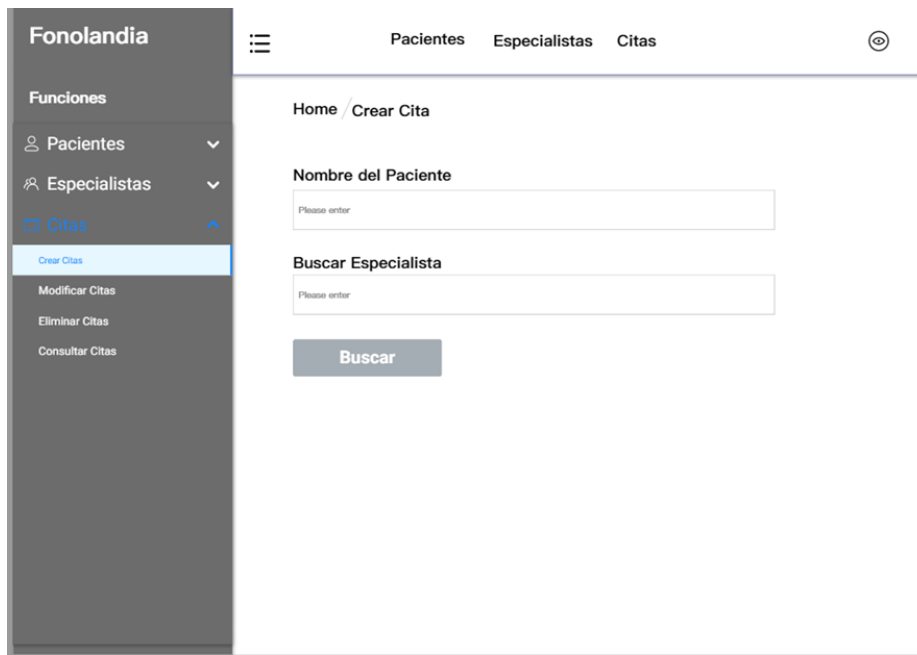


Figura 20. Crear cita elegir especialista

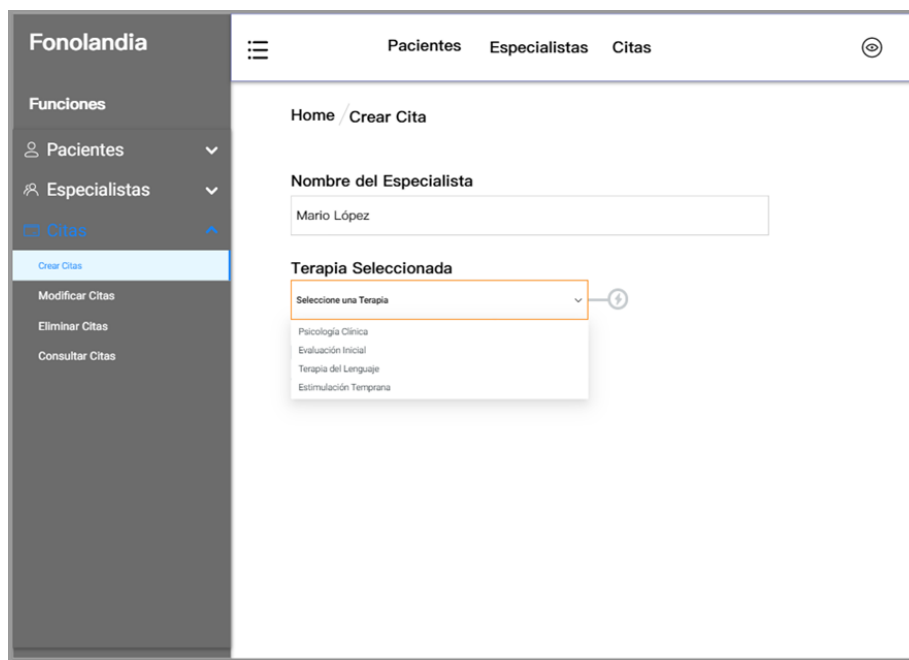


Figura 21. Crear cita elegir especialidad

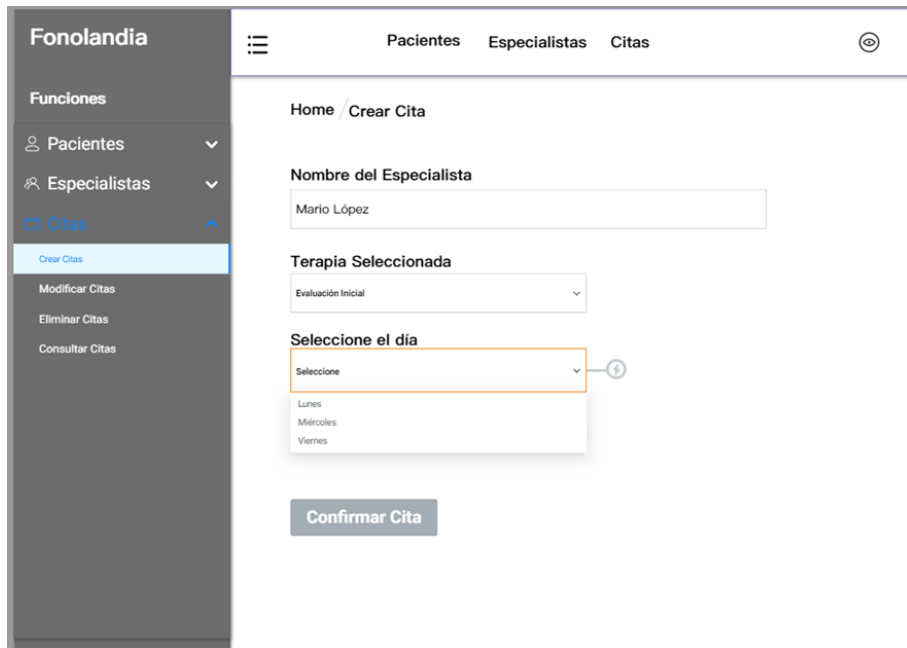


Figura 22. Crear cita elegir día

Citas/ Modificar Citas: tenemos la opción de modificar las citas según la disponibilidad o algún cambio que se pueda realizar.

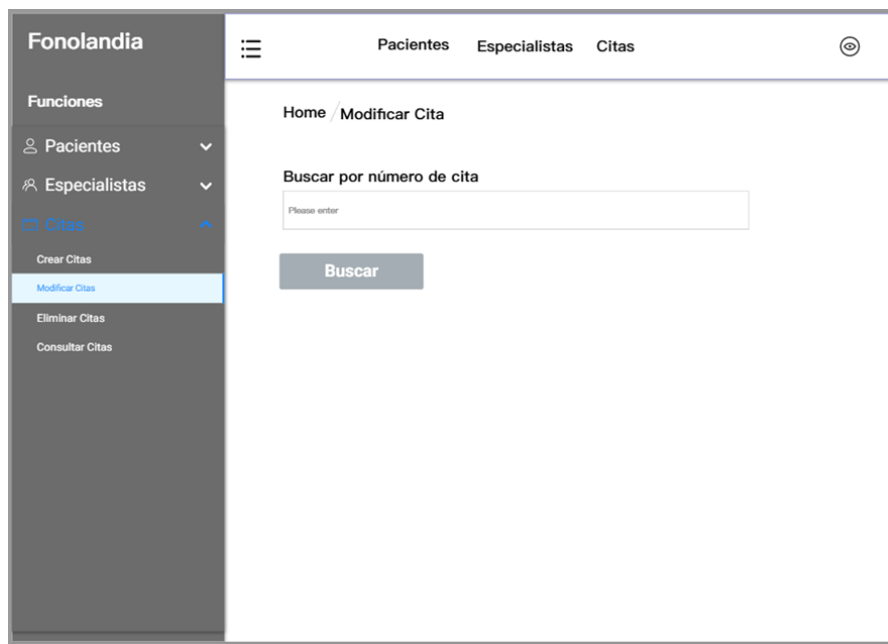


Figura 23. Modificar cita ingreso número de cita

The screenshot shows the 'Fonolandia' application interface. On the left is a dark sidebar with 'Fonolandia' at the top and a 'Funciones' (Functions) menu containing 'Pacientes', 'Especialistas', 'Citas', 'Crear Citas', 'Modificar Citas' (highlighted), 'Eliminar Citas', and 'Consultar Citas'. The main content area has a top navigation bar with 'Pacientes', 'Especialistas', and 'Citas' tabs. Below this is a breadcrumb 'Home / Crear Cita'. The form contains the following fields:
 

- Nombre del Paciente:** Text input with 'Pedro Alarcón'.
- Nombre del Especialista:** Text input with 'Mario Lopez'.
- Terapia Seleccionada:** Dropdown menu with 'Evaluación Inicial' selected.
- Seleccione el día:** Dropdown menu with 'Martes' selected.
- Seleccione el horario:** Dropdown menu with '9:00 - 10:00' selected.
- Confirmar Modificación:** A grey button at the bottom.

Figura 24. modificar cita ingreso de nuevos datos

Citas/ Eliminar Citas: en el caso que el paciente ya no desee tener la cita médica agendada se podrá realizar una eliminación de la cita y así quede libre la sala y el horario para otro paciente.

The screenshot shows the 'Fonolandia' application interface for deleting an appointment. The sidebar is identical to the previous screenshot, with 'Eliminar Citas' highlighted in the 'Funciones' menu. The main content area has a top navigation bar with 'Pacientes', 'Especialistas', and 'Citas' tabs. Below this is a breadcrumb 'Home / Eliminar Cita'. The form contains:
 

- Buscar por número de cita:** A text input field with the placeholder text 'Please enter'.
- Buscar:** A grey button below the search field.

Figura 25. Eliminar cita ingreso de número de cita

**Fonolandia** Pacientes Especialistas Citas

Home / Eliminar Citas

**Nombre del Paciente**  
Please enter

**Nombre del Especialista**  
Please enter

**Terapia Seleccionada**  
Please enter

**Día Seleccionado**  
Please enter

**Horario Seleccionado**  
Please enter

Eliminar Cita

Figura 26. Eliminar cita

Citas/ Consultar Citas: el usuario tiene la opción de puede consultar con el número de cita respectivo para que se le despliegue la información actualizada.

**Fonolandia** Pacientes Especialistas Citas

Home / Consultar Citas

**Buscar por número de historia clínica**  
Please enter

Buscar

Figura 27. Consultar cita ingreso de número de historia clínica

The screenshot displays the 'Fonolandia' application interface. On the left is a dark sidebar with the title 'Fonolandia' and a menu under 'Funciones' containing 'Pacientes', 'Especialistas', 'Citas', 'Crear Citas', 'Modificar Citas', 'Eliminar Citas', and 'Consultar Citas'. The main content area has a top navigation bar with 'Pacientes', 'Especialistas', and 'Citas' tabs. Below this, the page title is 'Home / Consultar Citas'. The form consists of five input fields, each with a 'Please enter' placeholder: 'Nombre del Paciente', 'Nombre del Especialista', 'Terapia Seleccionada', 'Día Seleccionado', and 'Horario Seleccionado'.

Figura 28. Consultar cita datos actualizados

- **Evaluación del Diseño:** Se socializó el diseño propuesto con la doctora Dayana Montenegro, que quedó muy satisfecha con él, solicitando dos cambios para los formularios y las tablas.

## Capítulo II Construcción del Sistema

### 3. Sprints

#### 3.1.Sprint 1: Realización de Mockups del Sistema

**Duración:** 1 mes

**Fechas:** del 20 de marzo al 20 de abril de 2024

**Objetivos y metas:**

- Desarrollar los mockups del sistema para el agendamiento de citas para Fonolandia.
- Incorporar sugerencias de la doctora sobre el cambio del ingreso de número de cédula a ID de historia clínica.

### **Historias de Usuario:**

- Historia de Usuario 1, Como desarrolladores, queremos preparar nuestro entorno de desarrollo para asegurarnos de que todas las herramientas que se necesiten estén disponibles y con su configuración correcta.
- Historia de Usuario 2, Como desarrolladores, queremos crear mockups del sistema utilizando Mockitt para definir claramente la interfaz y las interacciones de usuario.
- Historia de Usuario 3, como usuario, quiero que el diseño de ingreso de datos sea lo más intuitivo.

### **Criterios de Aceptación:**

1. Preparación del Entorno: El entorno de desarrollo debe incluir PostgreSQL, Node.js, React y Express.js. Las herramientas deben estar configuradas.
2. Diseño de Mockups: Los mockups deben reflejar todas las vistas y componentes del sistema. Los mockups deben incorporar el cambio del ingreso de número de cédula del paciente por el ID de la historia clínica del paciente. Los mockups deben ser revisados y aprobados por el usuario final.

### **Entregables:**

- Mockups del sistema.
- Cambios sugeridos por la doctora.

### **Desafíos y Soluciones:**

- Desafío: la Adaptación de los mockups a los nuevos requisitos dados por la doctora.
- Solución: Realizar las respectivas revisiones y ajustes con la colaboración de la doctora para así garantizar que el nuevo diseño cumpla con lo pedido.

### **3.2. Sprint 2: Desarrollo Frontend**

**Duración:** 1 mes

**Fechas:** del 21 de abril al 20 de mayo de 2024

**Objetivos y metas:**

- Implementar el diseño de los mockups en el frontend.
- Desarrollar los respectivos componentes e interacciones para la interfaz de usuario utilizando React con CoreUI.

**Historias de Usuario:**

- Historia de Usuario 1, Como usuario, quiero ver un frontend funcional que me muestre la interfaz del sistema según los mockups diseñados en la primera fase.
- Historia de Usuario 2, Como usuario, quiero interactuar con los componentes de la aplicación para navegar entre vistas y enviar datos.

**Criterios de Aceptación:**

1. Implementación de Componentes: todos los componentes se deben implementar de acuerdo con los diseños de los mockups. Los componentes deben ser funcionales y consistentes.
2. Interacción del Usuario: las interacciones deben funcionar correctamente como por ejemplo la navegación entre vistas y el envío de formularios. No deben existir errores visibles o fallos en la interfaz de usuario.

**Entregables:**

- Componentes frontend desarrollados y funcionales.

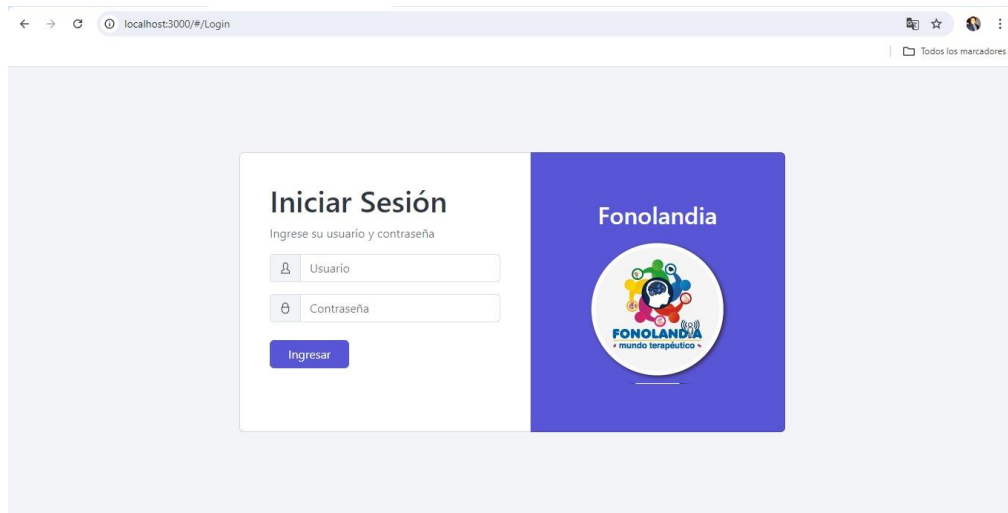


Figura 29. Login

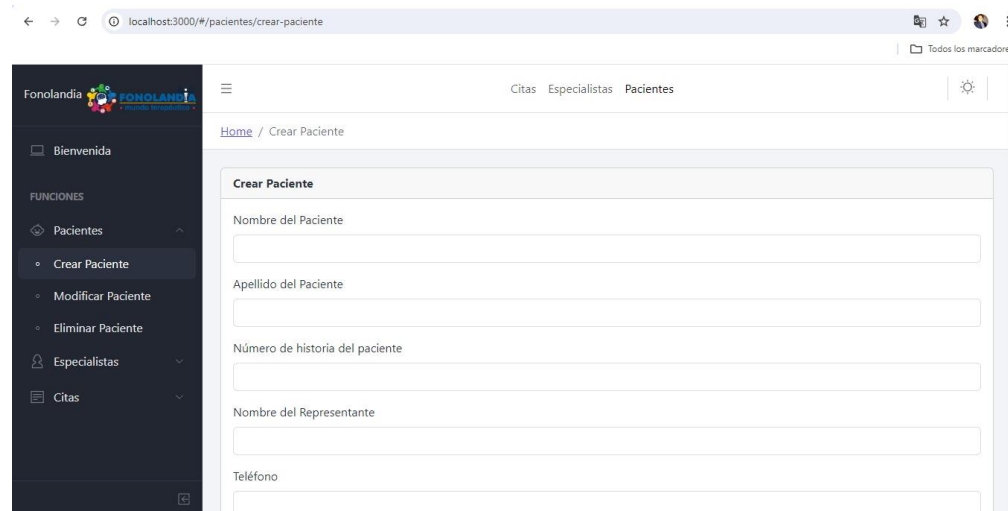


Figura 30. Página Principal

### Desafíos y Soluciones:

- **Desafío:** Integrar los componentes del frontend con el diseño visual y las interacciones requeridas.
- **Solución:** Uso de bibliotecas y herramientas de React y CoreUI para asegurar una integración coherente con los mockups diseñados.

## **Planificación y Gestión del Tiempo:**

- Creación de un cronograma detallado para el desarrollo de cada componente.
- Reuniones periódicas para el seguimiento del progreso y resolución de problemas.

### **3.3. Sprint 3: Desarrollo Backend**

**Duración:** 1 mes

**Fechas:** del 21 de mayo al 20 de junio de 2024

#### **Objetivos y metas:**

- Configurar la base de datos utilizando PostgreSQL.
- Desarrollar el backend con Express.js e integrar con la base de datos.
- Realizar pruebas de integración con Postman.

#### **Historias de Usuario:**

- Historia de Usuario 1, Como desarrolladores, queremos diseñar y configurar la base de datos para que esta pueda almacenar la información necesaria para el sistema.
- Historia de Usuario 2, Como desarrolladores, queremos implementar el backend utilizando Express.js para manejar las solicitudes del frontend y realizar operaciones en la base de datos.
- Historia de Usuario 3, Como desarrolladores, queremos realizar pruebas de integración con Postman para asegurarnos que las APIs funcionan correctamente-

#### **Criterios de Aceptación:**

1. Diseño y Configuración de la Base de Datos: La base de datos debe incluir todas las tablas y relaciones necesarias. Las consultas y operaciones deben funcionar correctamente.
2. Desarrollo del Backend: Los endpoints deben estar correctamente implementados. Las APIs deben ser accesibles y retornar datos en el formato esperado. Las pruebas con Postman deben mostrar que toda las APIS funcionan como se espera.

## Entregables:

- Base de Datos configurada y funcional.
- API REST implementada con Express.js

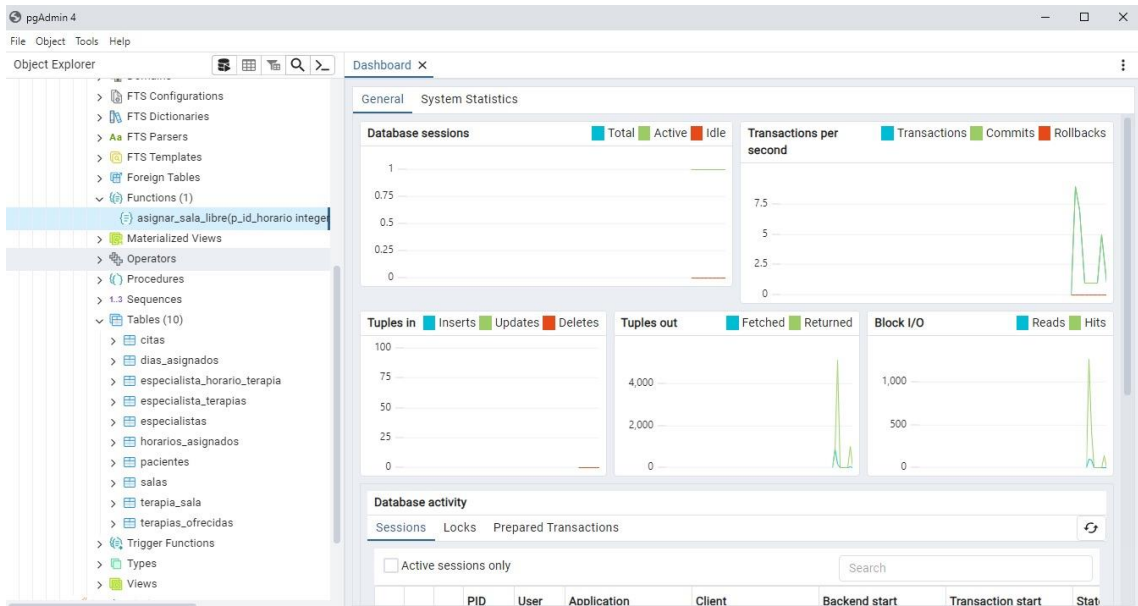


Figura 31. Base de Datos



Figura 32. archivo api

```
JS apijs x JS server.js
src > routes_api > JS apijs > router.get('/pacientes/num_historial') callback
33 pool.connect((err, client, release) => {
34   if (err) {
35     | return console.error('Error adquiriendo cliente', err.stack)
36   }
37   client.query('SELECT NOW()', (err, result) => {
38     | release()
39     | if (err) {
40     | | return console.error('Error ejecutando consulta', err.stack)
41     | }
42     | console.log('Conectado a la base de datos:', result.rows)
43     | })
44   })
45 }
46 router.get('/pacientes/:num_historial', async (req, res) => {
47   const { num_historial } = req.params
48   try {
49     | const result = await pool.query('SELECT * FROM Pacientes WHERE num_historial = $1', [
50     | | num_historial,
51     | ])
52     | if (result.rows.length === 0) {
53     | | return res.status(404).json({ error: 'Paciente no encontrado' })
54     | }
55     | res.status(200).json(result.rows[0])
56   } catch (err) {
57     | res.status(500).json({ error: err.message })
58   }
}
```

Figura 33. Implementación Express.JS

### Desafíos y Soluciones:

- Desafío: Integrar el backend con la base de datos y asegurar una comunicación efectiva.
- Solución: Implementar prácticas de desarrollo y pruebas exhaustiva para garantizar la integridad y el rendimiento del sistema.

### 3.4. Sprint 4: Integración de Componentes y Pruebas Finales

**Duración:** 1 mes

**Fechas:** del 21 de junio al 20 de julio de 2024

#### Objetivos y metas:

- Integrar todos los componentes frontend con el backend.
- Realizar pruebas finales para asegurar la funcionalidad completa del sistema.
- Resolver problemas detectados y ajustar el sistema según sea necesario.

### **Historias de Usuario:**

- Historia de Usuario 1, Como usuario, quiero que el sistema completo funcione correctamente con la integración del frontend y backend.
- Historia de Usuario 2, Como desarrolladores queremos realizar pruebas finales para así asegurar de que el sistema sea robusto y funcione correctamente cumpliendo los requisitos establecidos.

### **Criterios de Aceptación:**

1. Integración de Frontend y Backend: la aplicación debe funcionar como un sistema completo con la integración. Todos los componentes deben interactuar correctamente con el backend.
2. Pruebas Funcionales de Usabilidad: El sistema debe pasar todas las pruebas funcionales y de usabilidad sin errores significativos. La experiencia de usuario debe ser intuitiva y fluida.
3. Corrección de Errores y Ajustes Finales: Todos los errores identificados durante las pruebas deben ser corregidos.

### **Entregables:**

- Sistema integrado con todos los componentes funcionando correctamente.
- Informe de pruebas finales.

### **Desafíos y Soluciones:**

- Desafío: Resolver problemas de integración entre el frontend y el backend.
- Solución: Implementar pruebas de integración exhaustivas y ajustes basados en las pruebas obtenidas.

## Capítulo III

### 4. Pruebas y Estabilización

#### 4.1. Introducción

Las pruebas de software son esenciales en el proceso del desarrollo del sistema, debido a que aseguran que el sistema funcione de manera adecuada y cumpla con los requisitos que fueron establecidos. En este capítulo, se describe las pruebas realizadas en el sistema de agendamiento de citas para el centro pediátrico Fonolandia, según (Gregory, 2009) “Las pruebas de software son una parte integral del desarrollo ágil. Proporcionando retroalimentación rápida y continua sobre la calidad del software.”

En particular se detallan las pruebas de integración que se llevaron a cabo para así garantizar que los diferentes módulos del sistema funcionen de manera adecuada en su combinación. Estas pruebas han sido cruciales para la calidad y estabilidad del software, asegurando así que el sistema maneje las diferentes interacciones entre sus componentes sin errores.

#### 4.2. Tipos de Pruebas

En el desarrollo del sistema de agendamiento de citas, se realizaron algunos tipos de pruebas para así asegurar el correcto funcionamiento. Entre estas pruebas, las más destacadas fueron las pruebas de integración.

##### 4.2.1. Pruebas de Integración

Las pruebas de integración se enfocan en verificar la interacción entre los diferentes módulos del sistema, asegurando que funcionen correctamente cuando se combinan. En el sistema de agendamiento de citas, estas pruebas fueron de suma importancia para así garantizar, que los módulos interactúen de manera correcta.

### 4.3. Pruebas

Se desarrollo pruebas de integración con la base de datos, en las cuales se realizó un query para así asegurar la conexión con la base de datos mediante un Select simple de la fecha, en caso de que la conexión sea exitosa nos dará un resultado por consola.

```
pool.connect((err, client, release) => {
  if (err) {
    return console.error('Error adquiriendo cliente', err.stack)
  }
  client.query('SELECT NOW()', (err, result) => {
    release()
    if (err) {
      return console.error('Error ejecutando consulta', err.stack)
    }
    console.log('Conectado a la base de datos:', result.rows)
  })
})
```

Figura 34. Query de pruebas de integración

Al Inicializar el servidor se ejecutará ese query, con el siguiente resultado:

```
Servidor corriendo en http://localhost:3001
Conectado a la base de datos: [ { now: 2024-08-05T15:03:51.358Z } ]
```

Figura 35. Resultado query de integración

Se realizaron diferentes rutas con la conexión a la base de datos, por ejemplo, cuando recoge la información de cada paciente.

```
router.get('/pacientes/:num_historial', async (req, res) => {
  const { num_historial } = req.params
  try {
    const result = await pool.query('SELECT * FROM Pacientes WHERE num_historial = $1', [
      num_historial,
    ])
    if (result.rows.length === 0) {
      return res.status(404).json({ error: 'Paciente no encontrado' })
    }
    res.status(200).json(result.rows[0])
  } catch (err) {
    res.status(500).json({ error: err.message })
  }
})
```

Figura 36. Ruta de información de cada paciente

Al realizar una prueba en postman del método get de la ruta anterior se consigue el siguiente resultado:



Figura 37. URL

Hay un validador en el código que nos enviará un mensaje por consola de cada consulta, en caso de haber errores para poder validar cual fue la solicitud hecha al Backend, aunque no sea realizada por parte del Frontend.



Figura 38. Validación

Para el mensaje validamos dos cosas directamente desde el servidor

```
Solicitud recibida: GET /api/v1/pacientes/18
Ruta /api/v1 alcanzada
```

Figura 39. Respuesta

La primera que cual es la solicitud realizada y la segunda que acceda a la ruta principal de cada API.

```
// Middleware para logging
server.use((req, res, next) => {
  console.log(`Solicitud recibida: ${req.method} ${req.url}`)
  next()
})

// Usar las rutas de la API
server.use(
  '/api/v1',
  (req, res, next) => {
    console.log('Ruta /api/v1 alcanzada') // Log para verificar que la ruta base se está alcanzando
    next()
  },
  apiRoutes,
)
```

Figura 40. Validador rutas API

### 4.3.1. Pruebas de Integración Frontend

Las validaciones realizadas en Postman también fueron realizadas directamente en el Frontend de la siguiente manera:

Accedemos a la sección “Modificar paciente” donde se pedirá ingresar el número de historia clínica.

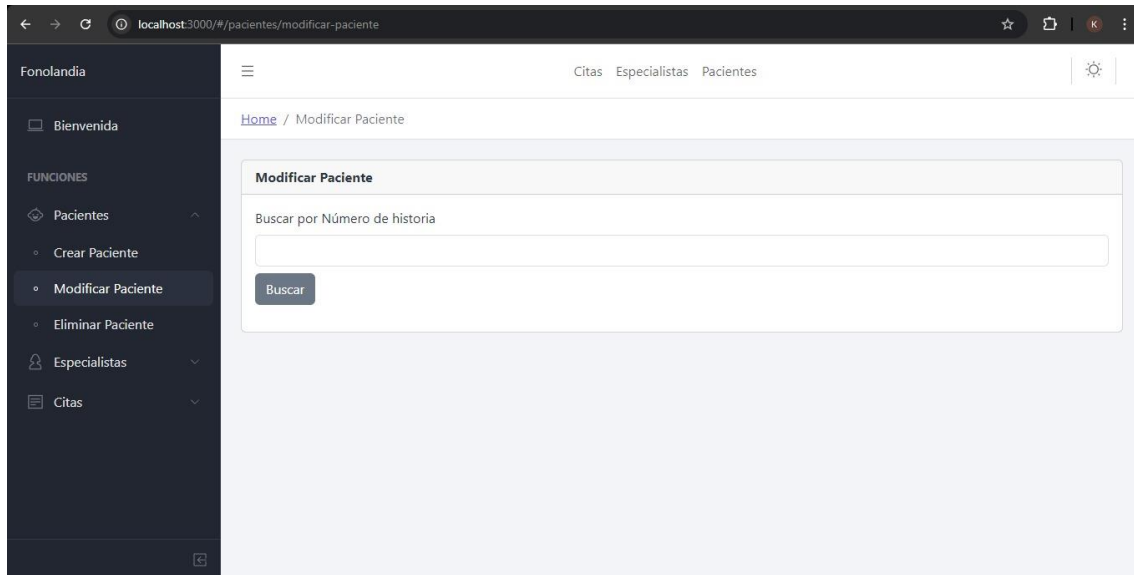


Figura 41. Módulo Modificar Paciente

Colocamos el número de historia clínica y seleccionamos buscar.

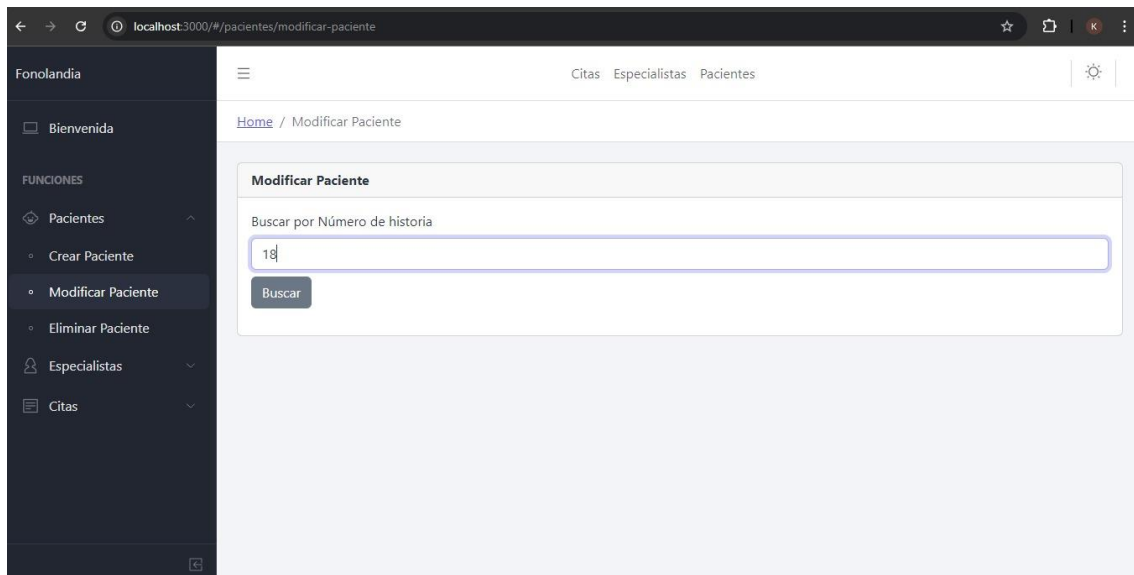


Figura 42. Ingreso de datos en módulo Modificar Paciente

Se despliega la misma información que en Postman

Nombre del Paciente  
Alejandro

Apellido del Paciente  
Perrazo

Número de Historia  
18

Nombre del Representante  
Elicio

Teléfono  
999999999

Correo Electrónico  
eaperrazo@puce.edu.ec

Figura 43. Formulario de Modificar Paciente

Así podemos observar que nos da el mismo mensaje en le Backend

```
Solicitud recibida: GET /api/v1/pacientes/18  
Ruta /api/v1 alcanzada
```

Figura 44. Resultado Obtenido

## **Conclusiones:**

- El desarrollo del sistema web de agendamiento de citas para el centro de estimulación temprana Fonolandia ha permitido optimizar y automatizar las operaciones administrativas relacionadas con el agendamiento de citas. Como resultado, se ha mejorado la eficiencia del usuario final, logrando una satisfacción entre los pacientes y el personal del centro.
- El análisis de los requisitos tanto funcionales como no funcionales ha sido clave para así garantizar que el sistema de agendamiento de citas cumpla con las necesidades del usuario. Este enfoque ha hecho desarrollar una solución que se adapta a las demandas operativas y tecnológicas del centro.
- El diseño de una interfaz intuitiva y accesible ha facilitado una eficiente gestión de pacientes, citas y especialistas. Esto ha simplificado la interacción con los usuarios, haciendo una experiencia más agradable y productiva.
- La implementación de notificaciones automáticas por correo electrónico ha reducido significativamente las inasistencias de los pacientes, mejorando la planificación del centro diariamente.
- El módulo que se desarrollo para el registro, modificación y eliminación de la información de los pacientes ha mejorado la eficiencia y organización de datos en el centro de estimulación temprana Fonolandia. Este sistema ha permitido un control riguroso y actualizado de la información, asegurando la disponibilidad e integridad de los datos.
- El sistema se ha desarrollado de manera escalable, esto asegura que podrá manejar un incremento en la demanda sin comprometer su rendimiento. El sistema está diseñado dejando espacio hacia futuras implementaciones como por ejemplo el añadir otra sede o posibles crecimientos que tenga directamente el centro.

**Recomendaciones:**

- Es recomendable establecer un plan de revisión y actualización periódica del sistema, para así asegurar que este se mantenga actualizado a las necesidades del usuario final, esto incluye la incorporación de nuevas funcionalidades y mejoras que sean basadas en el feedback continuo del usuario final.
- Capacitar a los usuarios finales sobre las nuevas funcionalidades y mejoras que se puedan dar, esto es crucial ya que así no solo aumentará la eficiencia y efectividad del sistema, sino que asegurará una transición óptima cuando se implemente alguna actualización.
- Explorar la posibilidad de ampliar funcionalidades del sistema para así incluir características adicionales que beneficie al usuario final, esto podría aumentar significativamente el valor y la utilidad del sistema desarrollado.
- Continuar con la optimización de la experiencia del usuario final mediante realización de pruebas periódicas e incorporación de mejoras basadas en los resultados de dichas pruebas. Esto asegurará que el sistema siga siendo intuitivo, fácil de usar y cumpla con las expectativas cambiantes del usuario.

## Referencias bibliográficas:

- Gamma, E. H. (1994). *Design patterns: Elements of reusable object-oriented software*. Addison-Wesley.
- Gregory, L. C. (2009). *Agile Testing: A Practical Guide for Testers and Agile Teams*. Boston, MA: Addison-Wesley Professional.
- Hoffman, J. (2018). *Pro Express.js: Master Express.js, the powerful Node.js framework for building web applications*. Apress.
- Jackson, J. (2019). *React and CoreUI: Building modern user interfaces with React and CoreUI*. Packt Publishing.
- Kroening, D. &. (2020). *Database systems: Concepts, desingn, and aplications*. Srpinger.
- Sommerville, I. (2011). *Software Engineering*. Bostón: Addison-Wesley.

## Anexos

Socialización del sistema con la Dra. Dayana Montenegro, propietaria Fonolandia

