Manual de Usuario: SEDES FARMACIA

**Introducción**

Bienvenido al manual de usuario de la aplicación móvil SEDES FARMACIA, aquí te enseñaremos a como poder iniciar el proyecto y el cómo funciona los diferentes módulos dentro de este proyecto. (TOMAR EN CUENTA QUE EL PROYECTO NO ESTA TERMINADO Y RECIBIRA ACTUALIZACIONES)

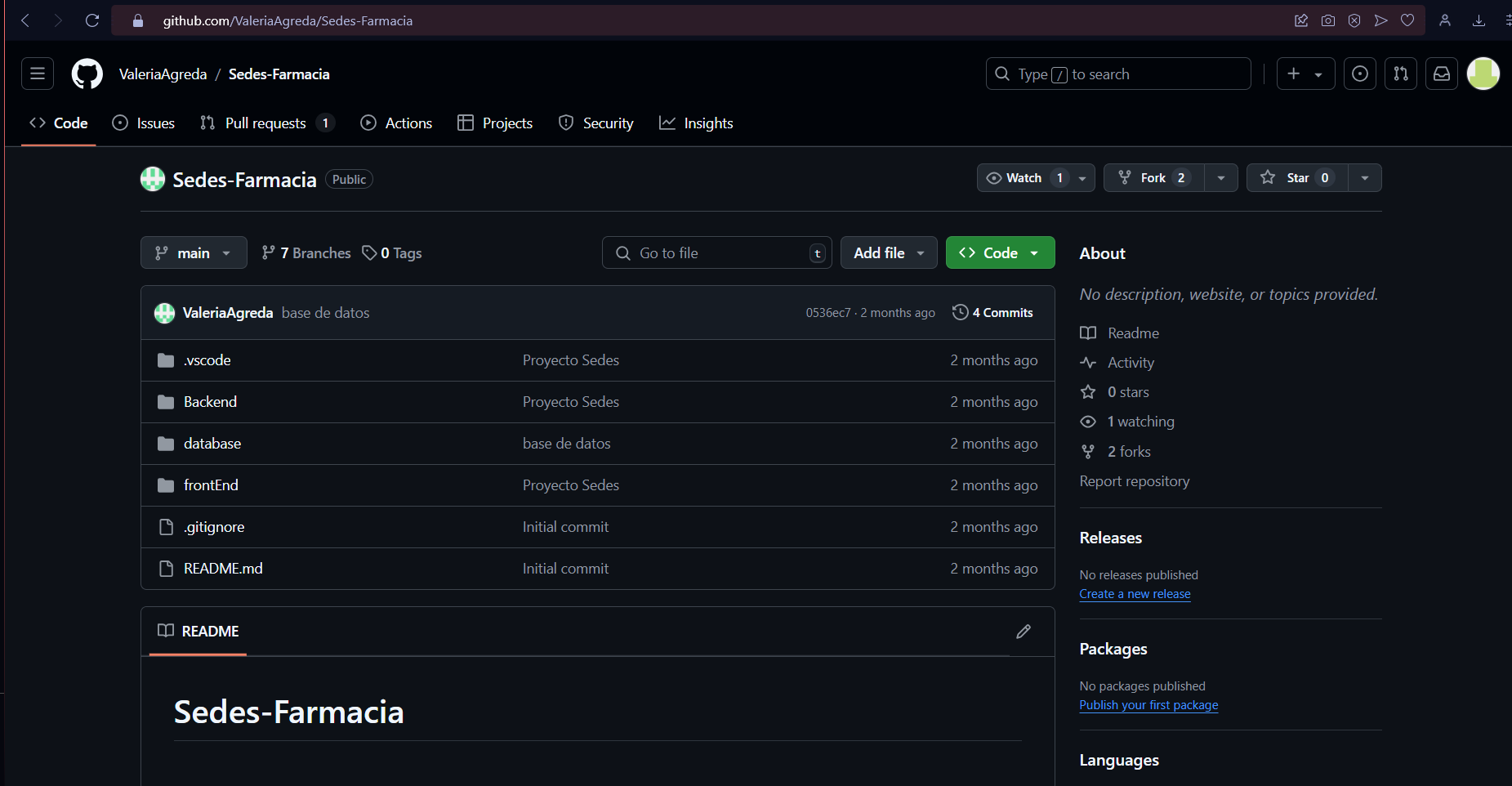
**Contenido**

1. Iniciar el proyecto
2. Página principal (Usuario final)
3. Inicio de Sesión
4. Página Olvidaste tu Contraseña
5. Página de vista principal (Usuario del sistema)
6. Modulo Gestión Farmacia
7. Modulo Gestión de Usuario
8. Modulo Gestión Turno
9. Herramientas Utilizadas

**1.-INICIAR EL PROYECTO (USUARIO FINAL)**

**Paso 1: Clonar o descargar el proyecto del GITHUB**

[**https://github.com/ValeriaAgreda/Sedes-Farmacia.git**](https://github.com/ValeriaAgreda/Sedes-Farmacia.git)

****

**PASO 2 : Cambiar al nombre de la base de datos que extrayeron y poner sus credenciales de usuario (database.js)**

**Texto

Descripción generada automáticamente**

**PASO 2 : Abrir dos terminales dentro del proyecto (Visual Studio Code) y ejecutar los siguientes comandos**

**Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente**

**TERMINAL 1**

**Cd backend**

**Npm install**

**Npm start**

**TERMINAL 2**

**Cd frontend**

**Npm install**

**Npm run dev**

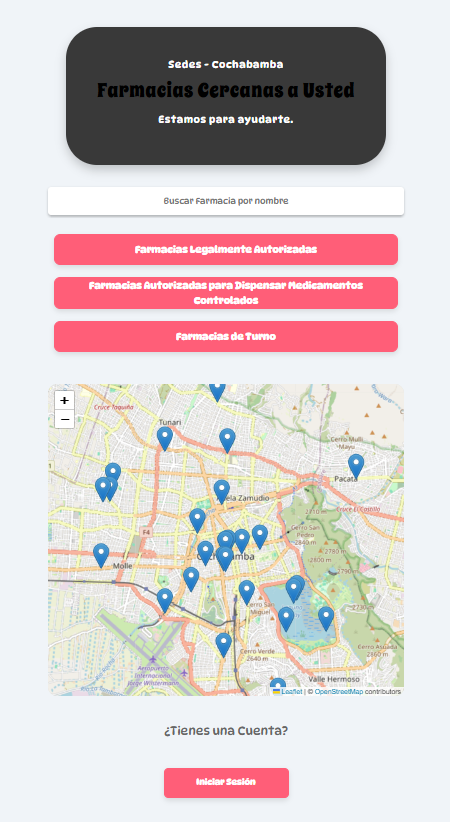
**COSAS EXTRAS**

**Cuenta Gmail (se puede cambiar) : gmail: sedesfarmacias@gmail.com contrasenia: SedesFarmacias123 contrasenia de aplicacion: fdoo nums yeoi nuet**

**2.-PAGINA PRINCIPAL (USUARIO FINAL)**

**Descripción**

Este sistema permite a los usuarios buscar farmacias cercanas en la ciudad de Cochabamba, Bolivia, mediante un mapa interactivo. Los usuarios pueden filtrar farmacias según su disponibilidad, servicios ofrecidos y turnos. Además, ofrece información detallada de cada farmacia seleccionada, como horarios de atención, estado de autorización y ubicación.



**Funcionalidades Principales**

**1. Visualización de Farmacias en el Mapa**

* Muestra un mapa interactivo con las ubicaciones de las farmacias disponibles.
* Cada farmacia está marcada con un **ícono de marcador** en el mapa.
* Mapa

  Descripción generada automáticamente

**2. Filtros de Farmacias**

* **Farmacias Legalmente Autorizadas:** Muestra solo farmacias que cumplen con las normativas legales.
* **Farmacias con Medicamentos Controlados:** Filtra las farmacias que dispensan sustancias reguladas.
* **Farmacias de Turno:** Visualiza las farmacias que están en turno el día actual.

Interfaz de usuario gráfica

Descripción generada automáticamente con confianza baja

**3. Búsqueda por Nombre**

* Permite buscar farmacias ingresando parte o el total de su nombre.

Interfaz de usuario gráfica, Texto, Aplicación, Word

Descripción generada automáticamente

**4. Detalles de Farmacia**

* Al seleccionar una farmacia en el mapa:
  + Se muestra su nombre, estado de autorización, horario de atención y fotografía (si está disponible).
  + Proporciona un enlace directo para abrir la ubicación en Google Maps.

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

**5. Inicio de Sesión**

* Redirige al usuario registrados a una página de inicio de sesión si desea acceder a funcionalidades adicionales.

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente con confianza media

**Cómo Usar el Sistema**

**Paso 1: Iniciar la Aplicación**

1. Abre la página principal del sistema.
2. Verás un mapa interactivo centrado en la ciudad de Cochabamba.

**Paso 2: Buscar una Farmacia**

1. Usa el campo de búsqueda ubicado en la parte superior.
   * Escribe el nombre (o parte del nombre) de la farmacia.
   * Los resultados se actualizan automáticamente en el mapa.

**Paso 3: Aplicar Filtros**

1. Haz clic en cualquiera de los botones disponibles:
   * **Farmacias Legalmente Autorizadas**
   * **Farmacias con Medicamentos Controlados**
   * **Farmacias de Turno**
2. El mapa actualizará automáticamente las farmacias mostradas según el filtro seleccionado.

**Paso 4: Ver Detalles de una Farmacia**

1. Haz clic en un marcador en el mapa.
2. Se mostrará un cuadro emergente con el nombre de la farmacia.
3. Haz clic en el cuadro emergente para cargar los detalles:
   * Nombre de la farmacia.
   * Estado de autorización (ej. **AUTORIZADA**).
   * Horarios de atención (12h, 24h, o predeterminados).
   * Fotografía de referencia (si está disponible).
   * Un enlace para abrir la ubicación en Google Maps.

**Paso 5: Actualizar Ubicación**

1. Si deseas resetear la selección, puedes actualizar el mapa con el botón de actualización de ubicación (si está configurado en tu versión).

**Paso 6: Iniciar Sesión**

1. Si tienes una cuenta, haz clic en **"Iniciar Sesión"** al final de la página.
2. Serás redirigido a la página de login.

**Navegación Técnica**

**Componentes Clave**

1. **Home.jsx**
   * Administra la interfaz principal del sistema.
   * Maneja los filtros, búsqueda y carga de detalles de farmacias.
2. **MapView.jsx**
   * Renderiza el mapa interactivo con Leaflet.
   * Realiza peticiones a la API para cargar las farmacias disponibles según el filtro o búsqueda.

**Interacción con el Backend (Farmacia.js)**

* **API URLS:**
  + /farmacia/all para obtener todas las farmacias.
  + /farmacia/farmacias-con-sustancias para farmacias con medicamentos controlados.
  + /farmacia/turnosDia/{fecha} para farmacias de turno.
  + /farmacia/cargarfarmacia/{id} para detalles de una farmacia específica.

3.-PAGINA DE INICIO DE SESIÓN (USUARIO DEL SISTEMA )

DESCRIPCIÓN

La página de inicio de sesión permite a los usuarios autenticarse en el sistema proporcionando un nombre de usuario y una contraseña. Si las credenciales son correctas, el usuario será redirigido al menú de administración. También incluye opciones para recuperación de contraseña y para regresar a la página principal.

Interfaz de usuario gráfica

Descripción generada automáticamente

**Funcionalidades Principales**

**1. Inicio de Sesión**

* Permite ingresar un **usuario** y una **contraseña**.
* Valida las credenciales enviando una solicitud al servidor.
* Si las credenciales son correctas, redirige al menú de administración.
* Si las credenciales son incorrectas, muestra un mensaje de error.

**2. Recuperación de Contraseña**

* Incluye un botón para redirigir a la página de recuperación de contraseña.

Imagen que contiene Logotipo

Descripción generada automáticamente

**3. Volver a la Página Principal**

* Proporciona un botón para regresar a la página principal del sistema.

Interfaz de usuario gráfica

Descripción generada automáticamente con confianza baja

**Cómo Usar la Página**

**Paso 1: Ingresar Credenciales**

1. En la página de inicio de sesión:
   * Introduce tu **usuario** en el campo de texto correspondiente.
   * Introduce tu **contraseña** en el campo de texto protegido.
2. Haz clic en el botón **"Iniciar sesión"**.

**Paso 2: Manejo de Errores**

* Si las credenciales son incorrectas o ocurre un error en el servidor:
  + Aparecerá un mensaje debajo de los campos de entrada indicando el problema.
  + Revisa las credenciales ingresadas y vuelve a intentarlo.

**Paso 3: Recuperar Contraseña**

1. Si olvidaste tu contraseña, haz clic en el botón **"¿Olvidaste tu contraseña?"**.
2. Serás redirigido a una página donde podrás restablecerla.

**Paso 4: Volver a la Página Principal**

1. Si deseas regresar a la página principal, haz clic en el botón **"Volver"**.
2. Esto te llevará a la página de inicio del sistema.

**Navegación Técnica**

**Componentes Clave**

1. **useState:**
   * Maneja el estado de los campos de entrada (user, password) y del mensaje de error (error).
2. **useNavigate:**
   * Redirige a las rutas específicas:
     + /MenuAdmin para el menú de administración después de un inicio de sesión exitoso.
     + /Contrasenia para la recuperación de contraseña.
     + / para volver a la página principal.
3. **Axios:**
   * Realiza solicitudes HTTP al servidor.
   * En este caso, envía los datos de inicio de sesión al endpoint http://localhost:8082/usuario/login.

**Interacción con el Backend (usuario.js)**

**Endpoint Usado:**

* **POST /usuario/login**
  + Envia un objeto con los campos user y password al backend.
  + Si las credenciales son válidas, responde con los datos del usuario.

**Formato de la Solicitud:**

{

"user": "nombre\_usuario",

"password": "contraseña\_segura"

}

**Posibles Respuestas:**

**Éxito (200):**

{

"message": "Inicio de sesión exitoso"

}

* + La aplicación redirige al usuario a /MenuAdmin.

**Error (401 o 400):**

{

"error": "Credenciales inválidas"

}

* + La página muestra el mensaje de error.

1. **Otros errores:**
   * Si no hay respuesta del servidor o ocurre un error desconocido, muestra el mensaje: **"Error al intentar iniciar sesión"**.

**Requisitos**

* Backend configurado en http://localhost:8082.
* Rutas disponibles:
  + /usuario/login
  + /MenuAdmin
  + /Contrasenia

4.-PAGINA OLVIDASTE TU CONTRASEÑA (USUARIO DEL SISTEMA)

**Descripción**

La página de **Recuperación de Contraseña** permite a los usuarios solicitar un restablecimiento de contraseña ingresando su correo electrónico. Si el correo es válido, recibirán un email con las instrucciones para restablecer su contraseña. También incluye una opción para regresar a la página de inicio de sesión.

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

**Funcionalidades Principales**

**1. Recuperación de Contraseña**

* Permite al usuario ingresar su correo electrónico.
* Envía una solicitud al servidor para iniciar el proceso de recuperación de contraseña.
* Notifica al usuario si el correo fue enviado con éxito o si ocurrió un error.

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

**2. Volver a la Página de Inicio de Sesión**

* Proporciona un botón para regresar a la página de inicio de sesión.

Imagen que contiene Interfaz de usuario gráfica

Descripción generada automáticamente

**Cómo Usar la Página**

**Paso 1: Ingresar el Correo Electrónico**

1. En la página de recuperación de contraseña:
   * Introduce tu correo electrónico en el campo de texto.
2. Haz clic en el botón **"Enviar"** para solicitar el restablecimiento.

**Paso 2: Verificar Mensajes**

* **Si el correo es válido:**
  + Aparecerá un mensaje indicando: **"Email enviado, revisa tu correo."**
  + Verifica tu bandeja de entrada o carpeta de spam para continuar con el restablecimiento.
* **Si hay un error:**
  + Aparecerá un mensaje como **"Error en la recuperación de la contraseña"**.
  + Asegúrate de que el correo ingresado sea correcto y vuelve a intentarlo más tarde.

**Paso 3: Volver a la Página de Inicio de Sesión**

1. Si deseas regresar a la página de inicio de sesión, haz clic en el botón **"Volver"**.
2. Serás redirigido a la página de inicio de sesión.

**Navegación Técnica**

**Componentes Clave**

1. **useState:**
   * Maneja el estado del correo electrónico (gmail) y del mensaje (message).
2. **useNavigate:**
   * Redirige a las rutas específicas:
     + /Login para regresar a la página de inicio de sesión.
3. **fetch:**
   * Realiza solicitudes HTTP al servidor.
   * En este caso, envía los datos del correo electrónico al endpoint http://localhost:8082/usuario/recover-password.

**Interacción con el Backend (usuario.js)**

**Endpoint Usado:**

* **POST /usuario/recover-password**
  + Envía un objeto con el correo electrónico al backend.
  + Si el correo es válido, el backend enviará un email al usuario con las instrucciones para recuperar la contraseña.

**Formato de la Solicitud:**

{

"gmail": "correo@ejemplo.com"

}

**Posibles Respuestas:**

**Éxito (200):**

{

"message": "Email enviado con éxito"

}

* + Muestra el mensaje: **"Email enviado, revisa tu correo."**

**Error (400 o 404):**

{

"error": "Correo no encontrado"

}

* + Muestra el mensaje de error devuelto por el servidor.

**Otros errores:**

* + Si no hay respuesta del servidor o ocurre un error desconocido, muestra el mensaje: **"Hubo un error, intenta más tarde."**

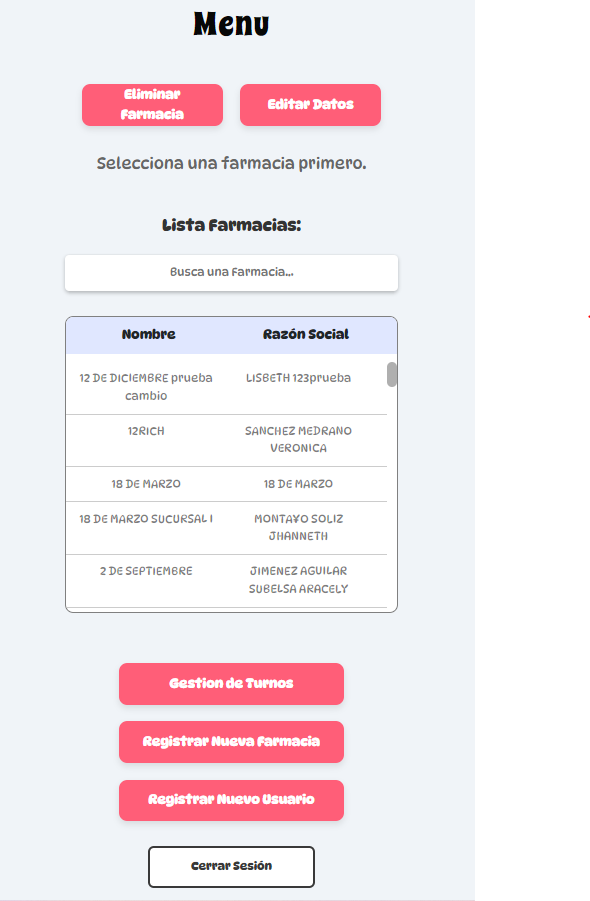
**Requisitos**

* Backend configurado en http://localhost:8082.
* Rutas disponibles:
  + /usuario/recover-password
  + /Login

5.-PÁGINA DE VISTA PRINCIPAL MENÚ (USUARIO DEL SISTEMA)

**Descripción**

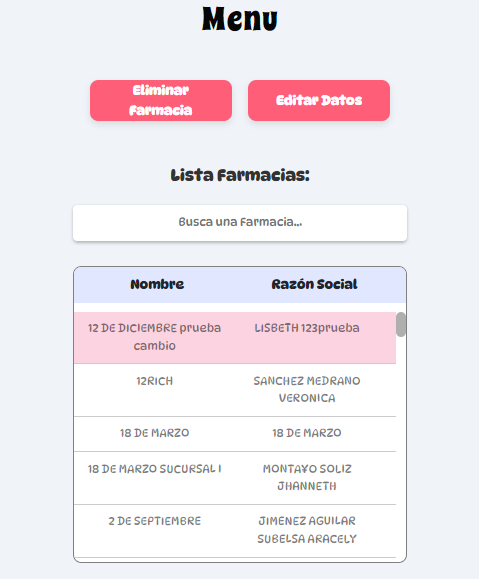
La página de **Menú Admin** permite a los usuarios acceder a distintos módulos. desde la creación, edición y eliminación de farmacias. También ver una lista de farmacias para seleccionarlas de ahí, como también un buscador para filtrar las farmacias por su nombre. Después tenemos en la parte inferior botones para acceder al módulo de turnos y el de usuarios. finalizando con el de cerrar sesión.



**Funcionalidades Principales**

**1. Selección de Farmacia para Eliminación y Edición de Datos**

* Permite al usuario seleccionar una farmacia de la lista.
* Permite al usuario buscar o filtrar una farmacia por su nombre en el cuadro “Busca una Farmacia”
* Envía una solicitud al servidor para iniciar el proceso de recuperación de la farmacia seleccionada para editar o eliminar.
* Notifica al usuario si se seleccionó una farmacia antes de apretar los botones.

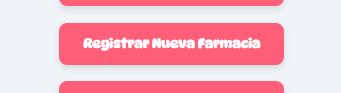


* Para eliminar una farmacia deberás confirmar la eliminación.



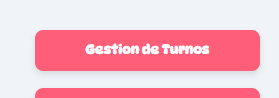
**2. Registrar Nueva Farmacia**

* Proporciona un botón para registrar una nueva farmacia que le llevará al formulario.



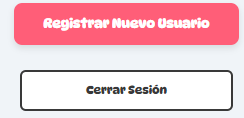
**2. Ir al Módulo Gestión de Turnos**

* Proporciona un botón para ir al modulo gestion turnos que le llevará a otro menú.



**2. Registrar Nuevo Usuario y Cerrar Session**

* Proporciona un botón para ir al modulo gestion usuarios que le llevará a otro menú.
* Proporciona un botón para ir al login que le llevará a la ventana de login de usuario.



**Cómo Usar la Página**

**Paso 1: Ingresar el Correo Electrónico**

En la página de menú admin:

* + Selecciona una farmacia de la lista para poder editarla o eliminarla.
  + Filtra la farmacia colocando su nombre en el cuadro de texto.

Haz clic en los otros botones para navegar por los módulos, crear una nueva farmacia o cerrar sesión.

**Paso 2: Verificar Mensajes**

* **Si se seleccionó una farmacia:**
  + Desaparecerá el texto indicando: **"Selecciona una farmacia primero."**
* **Si se eliminó una farmacia:**
  + Aparecerá un texto indicando: **"¿Estás seguro de que deseas eliminar esta farmacia?"** con las opciones de sí y no.

**Paso 3: Navegar por las opciones o cerrar session**

Si deseas regresar a la página de login, haz clic en el botón **"Cerrar Sesión"**.

Serás redirigido a la página de inicio de sesión.

**Navegación Técnica**

**Componentes Clave**

**useState:**

* + Maneja el estado de las farmacias (farmacias), búsqueda (searchTerm), farmacia seleccionada (selectedFarmacia), confirmación de eliminación (showConfirm), y mensajes de éxito/error.

**useEffect:**

* + Realiza una solicitud al backend para obtener todas las farmacias desde el endpoint:
  + GET http://localhost:8082/farmacia/all.
  + Actualiza las listas de farmacias (farmacias y filteredFarmacias).

**useNavigate:**

* + Redirige a las rutas específicas:
    - /Login para cerrar sesión.
    - /RegistroFarmacia para registrar una nueva farmacia.
    - /RegistroUsuario para registrar un usuario.
    - /Turno para gestionar turnos.
    - /EditarFarmacia/:id para editar una farmacia seleccionada.

**fetch:**

* + Endpoint usado:
    - PUT http://localhost:8082/farmacia/farmacia/eliminar/{id}
  + Si la eliminación es exitosa, actualiza las listas de farmacias.

**Interacción con el Backend (usuario.js)**

**Endpoint Usado:**

* **GET /farmacia/all: Obtiene todas las farmacias.**
* **PUT /farmacia/farmacia/eliminar/{id}: Marca una farmacia como eliminada en el sistema.**

**Formato de la Solicitud:**

{

"id": "ID de la farmacia"

}

**Posibles Respuestas:**

**Éxito (200):**

{

"message": "Email enviado con éxito"

}

* + La farmacia se elimina correctamente.
  + Muestra el mensaje: "Farmacia eliminada correctamente."

**Error (400 o 404):**

* + Si la farmacia no se encuentra o no puede eliminarse, muestra:
  + "Ocurrió un error al eliminar la farmacia."

**Otros errores:**

* + Si ocurre un problema en la comunicación con el servidor, muestra:
  + "Hubo un error, intenta más tarde."

6.-MODULO GESTION FARMACIA

El módulo de Gestión Farmacia permite al administrador crear y actualizar farmacias registradas en el sistema. A continuación, se detallan las funcionalidades principales y los pasos para realizar ambas acciones.

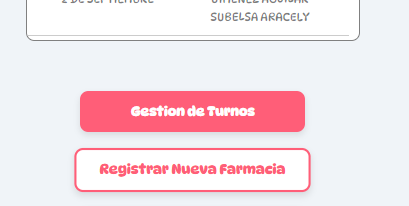
6.1.-CREACIÓN FARMACIA

**Descripción**

La funcionalidad de creación de farmacias permite registrar una nueva farmacia ingresando datos básicos, la ubicación, y los detalles del propietario.

**Cómo Usar la Página**

**Paso 1: Accede a la página de creación de farmacias desde el menú principal.**

****

**Paso 2: Completa los siguientes campos obligatorios:**

Datos de la farmacia:

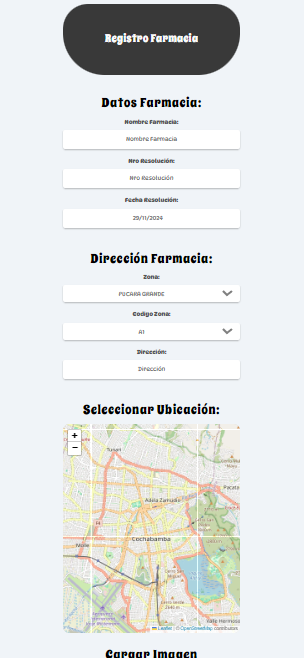
* + Nombre de la farmacia.
  + Número de resolución y fecha de resolución.
  + Dirección, zona, y código de zona.
  + Tipo de farmacia y horas de atención.
  + Imagen (subir archivo en formato JPG/PNG).

Datos del propietario:

* + Nombre, apellidos, y número de carnet de identidad.
  + Número de NIT, celular, y correo electrónico.

Ubicación geográfica:

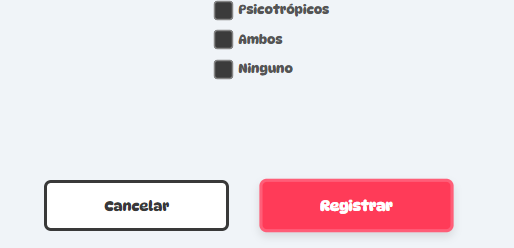
* + Selecciona la ubicación haciendo doble clic en el mapa interactivo.



**Paso 3: Haz clic en el botón "Registrar".**

Si todos los datos son válidos, aparecerá un mensaje indicando que la farmacia fue registrada exitosamente.

En caso de error, revisa los campos resaltados y corrige la información.



6.2.-ACTUALIZAR FARMACIA

**Descripción**

La funcionalidad de actualización permite modificar los datos de una farmacia previamente registrada.

**Cómo Usar la Página**

**Paso 1: Desde el menú principal, selecciona la farmacia que deseas editar.**

****

**Paso 2: Modifica los campos necesarios. Los campos obligatorios son los mismos que en la creación de farmacias:**

**Datos básicos:** Nombre, resolución, dirección, zona, etc.

**Propietario:** Nombre, apellidos, carnet, etc.

**Ubicación:** Actualiza la ubicación en el mapa si es necesario.



**Paso 3: Haz clic en el botón "Actualizar".**

Si los datos son válidos, aparecerá un mensaje indicando que la farmacia fue actualizada exitosamente.

En caso de error, corrige los campos señalados.



**Navegación Técnica**

**Componentes Clave**

**useState:**

* + Maneja el estado de:
    - Datos de los formularios:
      1. Nombre de la farmacia, propietario, dirección, etc.
      2. Campos relacionados con la ubicación, tipo de farmacia, horarios y categoría.
    - Mensajes de notificación: Mensajes de éxito o errores (error, successMessage).
    - Estados auxiliares:
      1. Listas desplegables para zonas, sectores, y códigos (zonas, sectores, codigosZona).
      2. Archivos cargados (fileBase64).

**useEffect:**

* + Carga inicial de datos:
  + Ejecuta funciones para obtener las opciones de zonas, sectores y categorías desde el servidor.
  + Edición: Carga los datos de la farmacia seleccionada al montar el componente EditarFarmacia.

**Fetch API / Axios:**

* + Utilizado para realizar solicitudes HTTP al servidor.

**Interacción con el Backend (usuario.js)**

**Endpoint Usado:**

* **POST /farmacia/nuevafarmacia**Crea una nueva farmacia con los datos proporcionados.
* **PUT /farmacia/actualizarfarmacia/:id**Actualiza los datos de una farmacia específica.
* **GET /farmacia/cargarfarmacia/:id**Carga los datos de una farmacia para su edición.
* **GET /farmacia/zonas**Obtiene la lista de zonas disponibles.
* **GET /farmacia/codigoszonas**Obtiene los códigos de zonas.
* **POST /farmacia/farmacia\_sustancias**Registra las sustancias controladas asociadas a una farmacia.

**Validaciones:**

**Creación y Edición:**

* + Todos los campos obligatorios deben estar completos.
  + Validación de formatos:
    - Nombre del propietario y apellidos: Solo letras y espacios.
    - CI: 7 u 8 dígitos.
    - NIT: Formato numérico válido.
    - Celular: 8 dígitos.
    - Correo electrónico: Formato estándar nombre@dominio.com.
  + La ubicación (latitud y longitud) debe seleccionarse en el mapa.
  + La imagen debe ser menor a 5 MB y estar en formato válido.

**Errores de Validación:**

* + "Por favor, complete los siguientes campos: [campos faltantes]."
  + "El formato del CI no es válido."
  + "El correo electrónico no es válido."
  + "Seleccione una imagen válida (hasta 5 MB)."

**Validaciones:**

* + Acciones Exitosas:
    - "Farmacia registrada exitosamente."
    - "Farmacia actualizada exitosamente."
  + Errores de Servidor:
    - "Error al registrar la farmacia."
    - "Error al actualizar la farmacia."
    - "Ocurrió un error al comunicarse con el servidor."
  + Errores de Validación:
    - "Todos los campos son obligatorios."
    - "El archivo seleccionado es demasiado grande."
    - "Formato de CI inválido."

7.-MODULO GESTIÓN DE USUARIO (USUARIO DEL SISTEMA)

**Descripción**

La página de **Gestión de Usuarios** permite al administrador visualizar, crear, editar y eliminar usuarios del sistema. Incluye validaciones para asegurar que los datos ingresados sean correctos, y muestra notificaciones para las acciones realizadas.

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

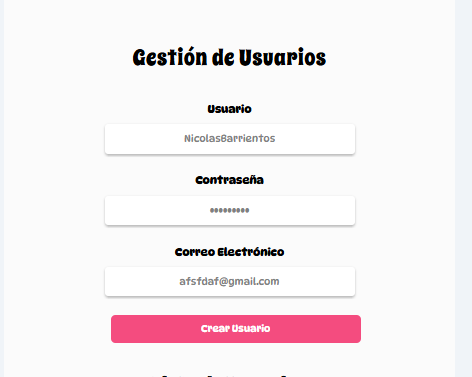
**Funcionalidades Principales**

**1. Visualización de Usuarios**

* Lista todos los usuarios registrados, mostrando:
  + Nombre de usuario.
  + Correo electrónico.

**2. Creación de Usuarios**

* Permite ingresar un nuevo usuario proporcionando:
  + Nombre de usuario.
  + Contraseña.
  + Correo electrónico.
* Valida los datos ingresados.



**3. Edición de Usuarios**

* Permite seleccionar un usuario existente y modificar su información.
* Actualiza los datos en el servidor.

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

**4. Eliminación de Usuarios**

* Elimina un usuario seleccionado, marcándolo como inactivo en la base de datos.

Interfaz de usuario gráfica, Texto, Aplicación, Chat o mensaje de texto

Descripción generada automáticamente

**5. Volver al Menú Administrativo**

* Proporciona un botón para regresar al menú principal del administrador.

Interfaz de usuario gráfica

Descripción generada automáticamente con confianza media

**Cómo Usar la Página**

**Paso 1: Visualizar Usuarios**

1. Al cargar la página, se listarán todos los usuarios disponibles.
2. Cada usuario se muestra con sus datos básicos.

**Paso 2: Crear un Usuario**

1. En el formulario:
   * Ingresa un **nombre de usuario** (entre 5 y 16 caracteres).
   * Ingresa una **contraseña** (entre 5 y 16 caracteres).
   * Ingresa un **correo electrónico** válido.
2. Haz clic en el botón **"Crear Usuario"**.
3. Si los datos son correctos, aparecerá un mensaje de éxito:
   * **"Usuario creado exitosamente"**.
4. Si hay errores en los datos, aparecerá un mensaje indicando el problema.

**Paso 3: Editar un Usuario**

1. En la lista de usuarios:
   * Haz clic en el botón **"Editar"** del usuario que deseas modificar.
   * Los datos del usuario se cargarán en el formulario.
2. Modifica los campos necesarios.
3. Haz clic en el botón **"Actualizar Usuario"**.
4. Si la actualización es exitosa, aparecerá el mensaje:
   * **"Usuario actualizado exitosamente"**.

**Paso 4: Eliminar un Usuario**

1. En la lista de usuarios:
   * Haz clic en el botón **"Eliminar"** del usuario que deseas borrar.
2. Si la eliminación es exitosa, aparecerá el mensaje:
   * **"Usuario eliminado"**.

**Paso 5: Volver al Menú**

1. Haz clic en el botón **"Volver"**.
2. Serás redirigido al menú principal del administrador.

**Navegación Técnica**

**Componentes Clave**

1. **useState:**
   * Maneja el estado de:
     + Lista de usuarios (usuarios).
     + Campos del formulario (user, password, gmail).
     + Mensajes de notificación (message, error).
     + ID del usuario seleccionado para editar (selectedUserId).
2. **useEffect:**
   * Ejecuta la función fetchUsuarios al cargar la página para obtener la lista inicial de usuarios.
3. **axios:**
   * Se utiliza para realizar solicitudes HTTP al servidor:
     + **GET**: Obtener usuarios.
     + **POST**: Crear un usuario.
     + **PUT**: Actualizar un usuario.
     + **DELETE**: Eliminar un usuario.

**Interacción con el Backend (usuario.js)**

**Endpoints Usados:**

1. **GET /usuario/all**
   * Obtiene la lista de todos los usuarios.
2. **POST /usuario**
   * Crea un nuevo usuario con los datos proporcionados.
3. **PUT /usuario/:id**
   * Actualiza la información del usuario especificado por su ID.
4. **DELETE /usuario/:id**
   * Elimina un usuario especificado por su ID.

**Validaciones**

1. **Usuario:**
   * Debe tener entre 5 y 16 caracteres.
2. **Contraseña:**
   * Debe tener entre 5 y 16 caracteres.
3. **Correo Electrónico:**
   * Debe cumplir con un formato válido (nombre@dominio.com).
4. **Campos Vacíos:**
   * Todos los campos son obligatorios.

**Mensajes y Notificaciones**

1. **Errores de Validación:**
   * **"Todos los campos son obligatorios."**
   * **"El usuario debe tener entre 5 y 16 caracteres."**
   * **"La contraseña debe tener entre 5 y 16 caracteres."**
   * **"El correo debe ser válido."**
2. **Acciones Exitosas:**
   * **"Usuario creado exitosamente."**
   * **"Usuario actualizado exitosamente."**
   * **"Usuario eliminado."**
3. **Errores de Servidor:**
   * **"Error creando usuario."**
   * **"Error actualizando usuario."**
   * **"Error eliminando usuario."**

**8.- MÓDULO GESTIÓN DE TURNOS (TURNO DE FARMACIAS)**

**Descripción**  
La página de Gestión de Turnos permite al administrador generar, visualizar, y enviar turnos asignados a las farmacias. Incluye validaciones para asegurar que los datos ingresados sean correctos, y muestra notificaciones para las acciones realizadas, los turnos se manejan de acuerdo a la cantidad de farmacias registradas en la base de datos y la cantidad de días que tiene el mes.



**FUNCIONALIDADES PRINCIPALES**

1. **Visualización de Turnos**
   * Lista todos los turnos asignados a las farmacias de una zona y un mes seleccionados, mostrando:
     + Nombre de la farmacia.
     + Dirección.
     + Fecha asignada.
2. **Generación de Turnos**
   * Permite asignar turnos a las farmacias seleccionadas, siguiendo estas reglas:
     + Asignación equitativa según el número de farmacias y días del mes.
     + Validación de asignaciones previas con opción de reemplazo.
     + Persistencia de datos en la base de datos.
3. **Envío de Notificaciones**
   * Envía correos electrónicos a los dueños de las farmacias notificando sus turnos asignados.
4. **Volver al Menú Administrativo**
   * Proporciona un botón para regresar al menú principal del administrador.

**CÓMO USAR LA PÁGINA**

**Paso 1: Seleccionar Zona y Mes**

1. Selecciona una zona de la lista desplegable "Filtrar por Zona".
2. Selecciona un mes de la lista desplegable "Filtrar por Mes".

**Paso 2: Cargar Turnos Existentes**

1. Haz clic en "Cargar Turnos" para ver los turnos asignados previamente.
2. Si no hay turnos, aparecerá un mensaje:
   * "No se encontraron turnos para la zona y mes seleccionados."

**Paso 3: Generar Turnos**

1. Haz clic en "Generar Turnos" para asignar turnos a las farmacias:
   * Si ya existen turnos, aparecerá un mensaje de confirmación para reemplazarlos.
2. Si la generación es exitosa, aparecerá el mensaje:
   * "Turnos generados y guardados exitosamente."

**Paso 4: Enviar Notificaciones**

1. Haz clic en "Mandar Gmails" para notificar a los dueños de las farmacias sus turnos asignados.
2. Aparecerá un mensaje indicando el éxito o error del envío.

**Paso 5: Volver al Menú**

1. Haz clic en "Volver" para regresar al menú principal.

**NAVEGACIÓN TÉCNICA**

**Componentes Clave**

1. **useState:**
   * Maneja el estado de:
     + Zonas (zonas).
     + Mes seleccionado (filtroMes).
     + Turnos cargados (turnos).
     + Validación de turnos existentes (turnosExistentes).
2. **useEffect:**
   * Ejecuta la función fetchZonas al cargar la página para obtener las zonas disponibles.
3. **fetch:**
   * Realiza solicitudes HTTP al backend para las operaciones CRUD y envío de correos.

**Interacción con el Backend**  
**Endpoints Usados:**

1. **GET /codigo/all**
   * Obtiene todas las zonas disponibles.
2. **GET /horas/turnosZonaMes/:codigoId/:mes/:anio**
   * Obtiene turnos asignados para una zona y mes específicos.
3. **POST /horas/guardarTurnos**
   * Guarda los turnos generados.
4. **DELETE /horas/eliminarTurnosZonaMes/:codigoId/:mes/:anio**
   * Elimina turnos existentes para una zona y mes específicos.
5. **GET /horas/enviarturnos**
   * Envía notificaciones por correo a los dueños de farmacias.

**VALIDACIONES**

1. **Zona y Mes:**
   * Selección obligatoria antes de cargar o generar turnos.
2. **Turnos Existentes:**
   * Confirmación antes de reemplazar turnos previos.

**MENSAJES Y NOTIFICACIONES**

1. **Errores de Validación:**
   * "Por favor selecciona una zona y un mes."
   * "No se encontraron farmacias para la zona seleccionada."
2. **Acciones Exitosas:**
   * "Turnos generados y guardados exitosamente."
   * "Correos enviados con éxito."
3. **Errores de Servidor:**
   * "Error al cargar los turnos."
   * "Error guardando turnos."
   * "Error enviando correos."

**9.- HERRAMIENTAS UTILIZADAS**

**Frontend:**

1. **React.js**
   * Biblioteca principal para construir interfaces de usuario interactivas y dinámicas.
   * Uso de hooks como useState y useEffect para manejar el estado y efectos secundarios.
2. **React Router**
   * Para navegación entre páginas y manejo de rutas dinámicas.
3. **React Bootstrap**
   * Para estilizar componentes y estructurar la interfaz con un diseño responsivo.
4. **Leaflet.js**
   * Biblioteca para la integración y manipulación de mapas interactivos.
5. **React Leaflet**
   * Componente de integración entre React y Leaflet para la visualización de mapas y marcadores.
6. **CSS Personalizado**
   * Para personalizar la apariencia y el diseño de las páginas según las necesidades del proyecto.

**Backend:**

1. **Node.js**
   * Plataforma de ejecución del backend, que actúa como servidor para gestionar la lógica de la aplicación.
2. **Express.js**
   * Framework para estructurar las rutas y manejar solicitudes HTTP en el backend.
3. **Nodemailer**
   * Librería utilizada para enviar correos electrónicos con notificaciones desde el backend.
4. **Fetch API**
   * Integrado en el frontend para interactuar con las APIs del backend.

**Base de Datos:**

1. **MySQL**
   * Sistema de gestión de bases de datos utilizado para almacenar información de farmacias, turnos, propietarios, y sus relaciones.
2. **MySQL Workbench**
   * Herramienta utilizada para administrar, diseñar y optimizar la base de datos.

**Gestión de Archivos:**

1. **FileReader API**
   * Utilizado en el frontend para cargar, procesar y convertir imágenes en formato Base64.
2. **Canvas API**
   * Utilizado para redimensionar imágenes antes de enviarlas al backend.

**Herramientas de Desarrollo y Testing:**

1. **Visual Studio Code**
   * IDE principal utilizado para escribir y depurar el código del proyecto.
2. **Postman**
   * Herramienta para probar y verificar las APIs del backend de manera eficiente.
3. **Console Logs**
   * Implementados para depuración y seguimiento de errores en el frontend y backend.

**Servidor Local:**

1. **Localhost**
   * Configuración del servidor local ejecutado en el puerto 8082 para pruebas y desarrollo.

**Control de Versiones:**

1. **Git**
   * Herramienta utilizada para gestionar versiones del código y facilitar la colaboración en equipo.
2. **GitHub**
   * Repositorio remoto para almacenar el código y facilitar el trabajo en equipo.

**Mapas y Geolocalización:**

1. **OpenStreetMap**
   * Fuente de datos de mapas para mostrar la ubicación de farmacias en el frontend.
2. **Map View Custom Components**
   * Componentes personalizados para centrar el mapa y manejar eventos de clic en los marcadores.

**Validación y Seguridad:**

1. **Expresiones Regulares**
   * Utilizadas para validar entradas de datos como correos electrónicos, CI, NIT, y números de teléfono.
2. **Manejo de Errores**
   * Validaciones en el frontend y backend para garantizar que los datos sean correctos y completos.

**Otros:**

1. **Axios**
   * Utilizado en el login para realizar solicitudes HTTP de manera más sencilla y estructurada.
2. **HTML5 Date Picker**
   * Para capturar y validar fechas en el formulario de registro de farmacias.
3. **Google Maps URL Integration**
   * Para proporcionar un enlace directo a la ubicación de cada farmacia en Google Maps.

Estas herramientas han sido seleccionadas para garantizar que el proyecto sea funcional, seguro y fácil de usar, con un diseño moderno y un backend robusto.