



# **PROYECTO FIN DE CICLO**

**APLICACIÓN GVP  
(GESTIÓN VISITAS PLANTA)**

**GVP APP  
(PLANT VISITS MANAGEMENT)**

Autor

**Pablo Álava Lasheras**

Tutor

**Cesáreo Anía Gérez**

**DESARROLLO DE APLICACIONES  
MULTIPLATAFORMA**

**2022/2023**

## **RESUMEN**

Este Trabajo de Fin de Grado ha sido realizado por Pablo Álava Lasheras, estudiante del Grado Superior de Desarrollo de Aplicaciones Multiplataforma en el IES Tubalcain de Tarazona en el curso 2021/2022.

El proyecto consta de una aplicación móvil para Android y una aplicación Web, se basa en la gestión de visitas planta de los hospitales.

A partir de la pulsera del paciente, el enfermero de planta podrá acceder con rapidez a los datos de dicho paciente gracias a la aplicación móvil.

## **PALABRAS CLAVE**

Hospital, Enfermería, Planta, Código barras, Android, Kotlin, Java, Android Studio, Visual Studio Code, IntelliJ IDEA, Spring Boot, Api Rest, JavaScript.

## **ABSTRACT**

This Final Degree Project has been carried out by Pablo Álava Lasheras, a student of the Higher Degree in Multiplatform Application Development at the IES Tubalcain de Tarazona in the 2021/2022 academic year.

The project consists of a mobile application for Android and a Web application, it is based on the management of hospital floor visits.

From the patient's bracelet, the floor nurse will be able to quickly access said patient's data thanks to the mobile application.

## **KEYWORDS**

Hospital, Nursing, Plant, Barcode, Android, Kotlin, Java, Android Studio, Visual Studio Code, IntelliJ IDEA, Spring Boot, Api Rest, JavaScript.

## TABLA DE CONTENIDOS

<b>DESCRIPCIÓN DEL PROYECTO .....</b>	<b>1</b>
CONTEXTO DEL PROYECTO .....	1
ÁMBITO Y ENTORNO .....	1
ANÁLISIS DE LA REALIDAD .....	1
SOLUCIÓN Y JUSTIFICACIÓN DE LA SOLUCIÓN PROPUESTA .....	2
DESTINATARIOS .....	2
OBJETIVO DEL PROYECTO .....	3
OBJETIVO DEL PROYECTO EN LENGUA EXTRANJERA .....	4
<b>ACUERDO DEL PROYECTO .....</b>	<b>5</b>
REQUISITOS FUNCIONALES Y NO FUNCIONALES .....	5
TAREAS .....	6
METODOLOGÍA A SEGUIR PARA LA REALIZACIÓN DEL PROYECTO .....	7
PLANIFICACIÓN TEMPORAL DE TAREAS .....	8
PRESUPUESTO (GASTOS, INGRESOS, BENEFICIO) .....	10
<b>ANÁLISIS Y DISEÑO .....</b>	<b>11</b>
MODELADO DE DATOS .....	11
ANÁLISIS Y DISEÑO DE LA INTERFAZ DE USUARIO .....	23
ANÁLISIS Y DISEÑO DEL SISTEMA FUNCIONAL .....	50
DISEÑO DE LA ARQUITECTURA DE LA APLICACIÓN .....	59
TECNOLOGÍAS/HERRAMIENTAS USADAS Y SU DESCRIPCIÓN .....	59
ARQUITECTURA DE COMPONENTES DE LA APLICACIÓN .....	61
<b>IMPLEMENTACIÓN E IMPLANTACIÓN DEL SISTEMA.....</b>	<b>63</b>
IMPLEMENTACIÓN .....	63
PRUEBAS .....	63
<b>DOCUMENTO DE CIERRE .....</b>	<b>64</b>
DOCUMENTO DE DESPLIEGUE Y CONFIGURACIÓN .....	64
MANUAL DE USUARIO .....	66
RESULTADOS OBTENIDOS Y CONCLUSIONES .....	68
SEGUIMIENTO Y DESVIACIÓN RESPECTO A LA PLANIFICACIÓN .....	69
<b>BIBLIOGRAFÍA .....</b>	<b>70</b>

## **DESCRIPCIÓN DEL PROYECTO**

### **CONTEXTO DEL PROYECTO**

La idea de GVP (Gestión visitas planta) surge a raíz de la necesidad de la empresa Viamed Salud S.L. para mejorar la eficacia y rapidez de los enfermeros en planta para realizar sus tareas.

### **ÁMBITO Y ENTORNO**

El entorno de dispositivos de este proyecto está enfocado para dispositivos Android y cualquier navegador Web.

Por otra parte, el proyecto y la aplicación se utiliza en el ámbito y entorno Hospitalario y de Salud.

La aplicación y el proyecto “GVP” ha sido enfocado específicamente para la empresa Viamed Salud S.L pero puede ser adaptado a otros grupos hospitalarios.

### **ANÁLISIS DE LA REALIDAD**

Investigando otras aplicaciones similares que puedan ser competencia de GVP pueden ser:

- iClinic: es una aplicación médica preparada para buscar y gestionar los historiales clínicos y llevar al día la facturación y la contabilidad del centro médico. Como el resto de aplicaciones médicas seleccionadas, ofrece plantillas que facilitan el día a día de los profesionales sanitarios.
- EpicCare EMR: es una aplicación destinada para médicos que trabajan en grandes hospitales y centros sanitarios, de ahí que no exista una versión gratuita, sino que se tenga que solicitar un presupuesto a medida. El personal sanitario cuenta con funcionalidades donde pueden organizar y seguir sus flujos de trabajo y gestionar la documentación.

Respecto las aplicaciones encontradas, ninguna se asemeja a la idea principal de este proyecto la cual se basa en la lectura de las pulseras de los pacientes mediante un código de barras para ver los datos del paciente.

Son aplicaciones más complejas y difíciles de gestionar.

## **SOLUCIÓN Y JUSTIFICACIÓN DE LA SOLUCIÓN PROPUESTA**

Como se ha comentado antes, la idea del proyecto ha surgido de la necesidad de mejorar la rapidez de la enfermería de planta.

A raíz de usar una aplicación móvil, el personal médico podrá llevar encima tablets o móviles para poder escanear con rapidez y ver los datos del paciente.

Por otra parte, el administrador podrá registrar los episodios de pacientes y usuarios (Personal de Planta) mediante una aplicación web.

Gracias a los conocimientos adquiridos en el curso DAM se propuso desarrollar estas aplicaciones.

Al principio se decidió desarrollar la aplicación en el lenguaje de programación Java, pero revisando bien el lenguaje, se decidió que Kotlin era más efectivo, más nuevo y fácil de aprender, para que pueda crear aplicaciones poderosas de inmediato.

La aplicación de administración ha sido desarrollada en Java Web con el framework Spring Boot.

Respecto al almacenamiento de datos, se debía adaptar al sistema Microsoft SQL Server, ya que este sistema de almacenamiento es que utiliza en Viamed Salud.

Para ello se ha pensado en crear una Api Rest para la comunicación de datos entre la aplicación móvil y la base de datos de Viamed Salud.

El lenguaje que se ha decidido para la Api Rest ha sido JavaScript y el entorno Node.js.

## **DESTINATARIOS**

GPV está destinado para ser utilizado en el sector hospitalario, más específicamente a la empresa Viamed Salud, ya que, la idea surgió gracias a esta y por ende, son colaboradores en el proyecto.

## OBJETIVO DEL PROYECTO

### **GVP Administración Web**

GVP Administración Web es una aplicación que tiene como principales funciones, crear episodios y crear usuarios.

Por otra parte, también nos permite visualizar, eliminar y modificar todos los episodios y usuarios.

También nos permite visualizar e imprimir la pulsera del paciente.

Para poder manejar todos los datos de la base de datos se necesitará iniciar sesión.

La conexión de esta aplicación y la base de datos será mediante el jdbc de Sql Server.

### **GVP Móvil**

GVP móvil es una aplicación Android que tiene como principales funciones ver con rapidez los datos del paciente, crear evoluciones y mostrar las estadísticas y datos del personal sanitario.

También tiene incorporada un ToDo en la aplicación para que los sanitarios puedan crear tareas pendientes y ponerlas como completadas.

La conexión de la aplicación y la base de datos SQL Server se realiza mediante un Api Rest.

El almacenamiento de las tareas se almacena de forma local en Android con SQLite.

## OBJETIVO DEL PROYECTO EN LENGUA EXTRANJERA

### **GVP Web Administration**

GVP Web Administration is an application whose main functions are creating episodes and creating users.

On the other hand, it also allows us to view, delete and modify all episodes and users.

It also allows us to view and print the patient's bracelet.

In order to manage all the data in the database, you will need to log in.

The connection of this application and the database will be through the jdbc of Sql Server.

### **Mobile GVP**

GVP mobile is an Android application whose main functions are to quickly view patient data, create evolutions and display the statistics and data of health personnel.

It also has a ToDo incorporated into the application so that health workers can create pending tasks and mark them as completed.

The connection between the application and the SQL Server database is done through a Rest API.

Task storage is stored locally on Android with SQLite.



## **ACUERDO DEL PROYECTO**

### **REQUISITOS FUNCIONALES Y NO FUNCIONALES**

#### **Requisitos funcionales Web Administrator GVP:**

- El usuario puede iniciar sesión.
- El usuario puede cerrar sesión.
- El usuario puede crear episodios.
- El usuario puede eliminar episodios.
- El usuario puede modificar episodios.
- El usuario puede visualizar los episodios existentes.
- El usuario puede ver los detalles de episodio.
- El usuario puede generar pdf de la pulsera del paciente.
- El usuario puede crear usuarios.
- El usuario puede eliminar usuarios.
- El usuario puede modificar episodios.
- El usuario puede visualizar los episodios existentes.

#### **Requisitos funcionales GVP Móvil:**

- El usuario puede iniciar sesión.
- El usuario puede cerrar sesión.
- El usuario puede escanear la pulsera del paciente.
- El usuario puede visualizar los pacientes.
- El usuario puede filtrar los pacientes.
- El usuario puede ver sus estadísticas.
- El usuario puede ver sus trabajos.
- El usuario puede crear una evolución.
- El usuario puede elegir el tipo de evolución.
- El usuario puede ver el historial clínico del paciente.
- El usuario puede crear tareas.
- El usuario puede marcar como completada la tarea.
- El usuario puede eliminar la tarea.

#### **Requisitos no funcionales:**

- La aplicación móvil es exclusiva para dispositivos Android.
- La aplicación de administración se puede utilizar en cualquier dispositivo con un navegador web.
- Los datos se pueden almacenar externamente:
  - SQL SERVER mediante Api Rest y JDBC
- Los datos se pueden almacenar internamente:
  - Android (SQLite con Room)

**TAREAS**

APP MOVIL GVP			
Autenticación de usuarios	Inicio de sesión		
Home	Escanear Pulsera	Ver el paciente escaneado	Ver datos del Paciente
			Crear Evolución
	Listado Pacientes	Filtrar Paciente	Ver historial clínico del paciente
		Seleccionar Paciente	Ver visitas realizadas hoy
	Mis trabajos	Ver trabajos realizados	
		Filtrar trabajos realizados	
	Tareas	Crear tarea	
		Ordenar tareas	
		Completar tarea	
		Eliminar tarea	
Perfil	Mis estadísticas	Ver las estadísticas del usuario	
	Cerrar Sesión		

Aplicación Web Administración		
Autenticación	Iniciar Sesión	
Episodios	Crear episodio	
	Listar episodios	Imprimir episodio
		Ver detalles
		Modificar episodio
		Eliminar episodio
Usuarios	Crear usuario	
	Listar <u>usuarios</u>	Modificar usuario
		Eliminar usuario
Cerrar sesión		

## METODOLOGÍA A SEGUIR PARA LA REALIZACIÓN DEL PROYECTO

Para realizar este proyecto dentro del periodo de tiempo indicado, es necesaria una buena organización. Por este motivo, se decidió utilizar la metodología Scrumban (Scrum y Kanban). Primero, se clasificaron las tareas según su grado de importancia, y las más importantes se anotaron en Google Keep.

Se realizaron todas las tareas a la vez, intentando alternarlas si en algunas no se conseguía el rendimiento de trabajo esperado.

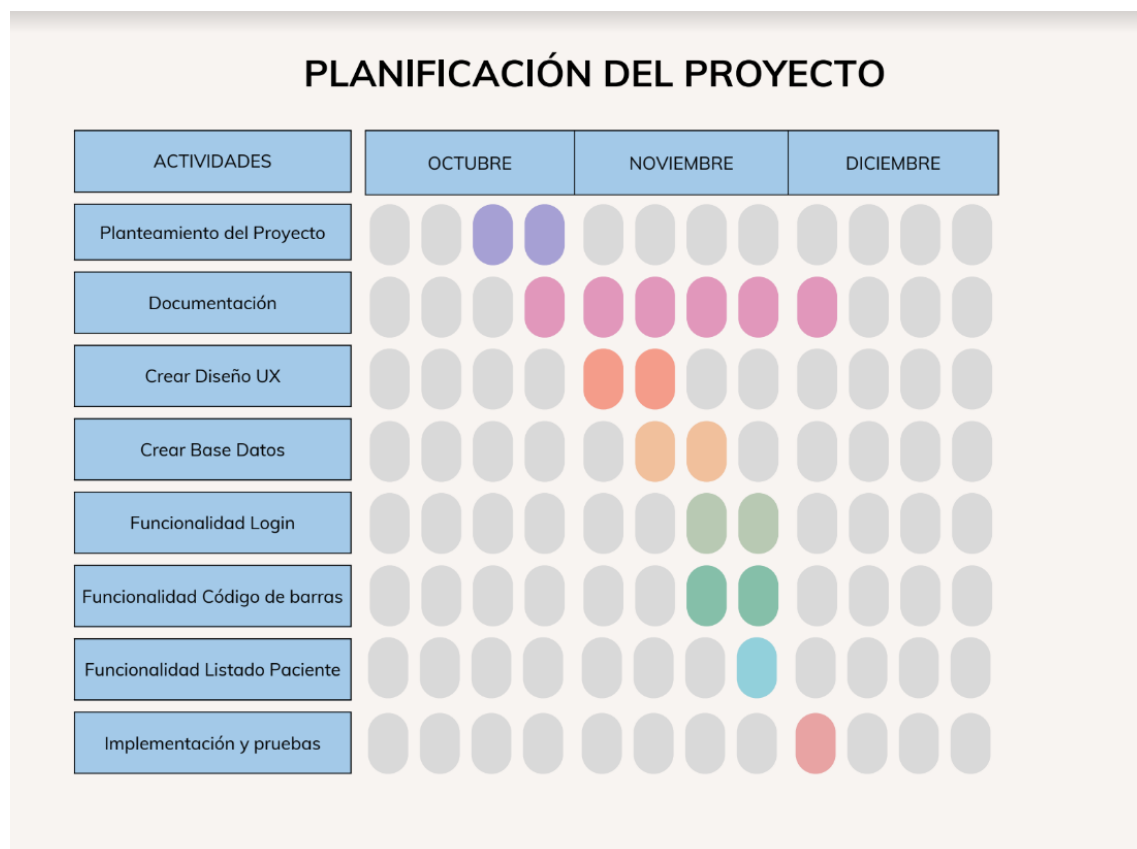
Una vez finalizada la tarea, se añadían otras siempre teniendo en cuenta el tiempo restante y siguiendo el orden de prioridad establecido anteriormente.

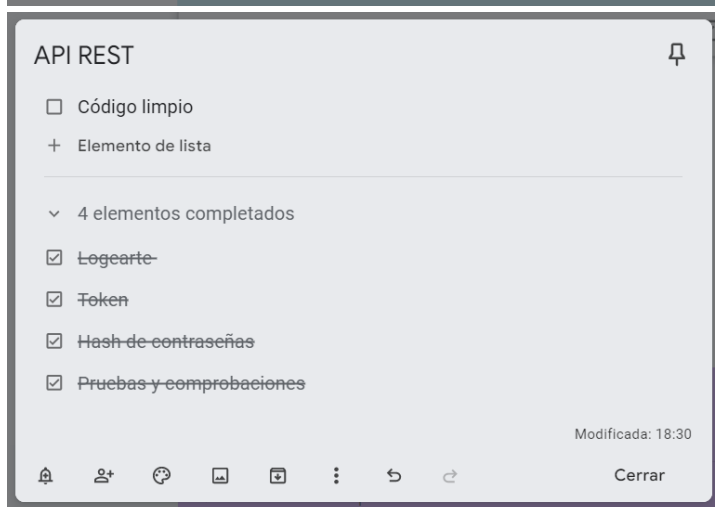
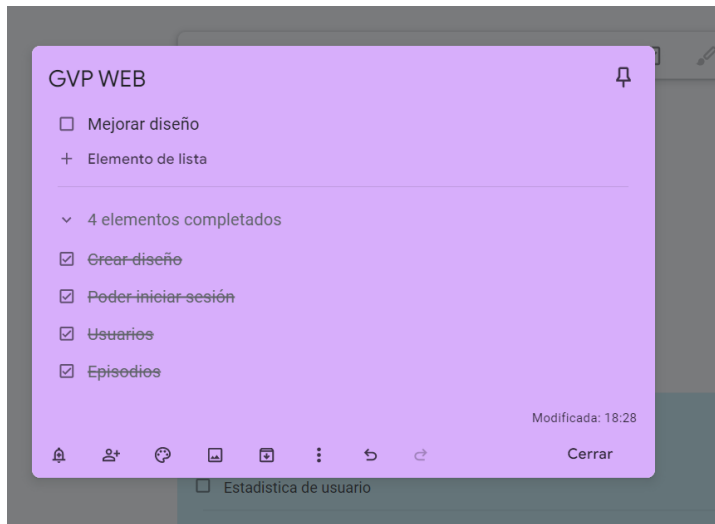
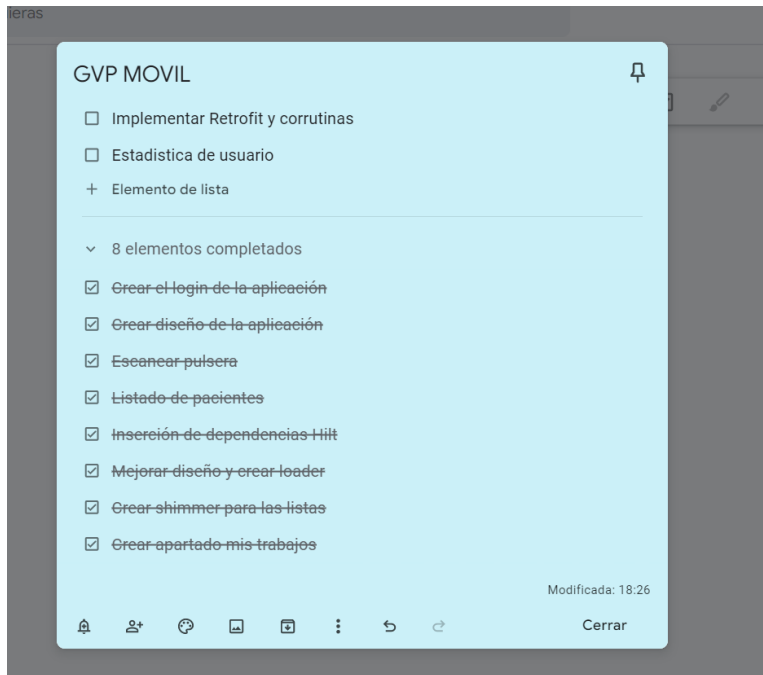
## PLANIFICACIÓN TEMPORAL DE TAREAS

Para la planificación temporal de tareas se estudió el tiempo máximo que debía dedicarse a cada una. Por ese motivo, saber la fecha de entrega era necesario para la planificación general. Siendo la fecha de entrega el 7/12/2022 y empezando el proyecto aproximadamente el 2/11/2022, una buena organización gracias a la metodología Scrumban, era necesaria para poder terminar el proyecto.

Primero, se planificó el proyecto y se establecieron las tareas y funcionalidades obligatorias que debían ser trabajadas. Una vez planificado el trabajo de forma general, se pudo especificar más en la planificación de otras tareas siguiendo el orden de prioridad. Sabiendo la gran complejidad del proyecto, se supuso que los tiempos de cada una de las tareas iban a ser inexactos. Pudiéndose atrasar debido a cualquier tipo de complicación no prevista.

La primera tarea fue crear los diseños de las dos aplicaciones. Una vez terminados, se empezó a planear y programar la lógica.





**PRESUPUESTO (GASTOS, INGRESOS, BENEFICIO)**

<b>PRESUPUESTO PROYECTO GVP</b>		
<b>DESCRIPCIÓN</b>	<b>UNIDADES</b>	<b>INGRESOS</b>
Horas trabajadas	40	800€
Seguro y Mantenimiento	1	250€
Total		1050€
IVA	21%	
Suma Total		1270,5€

## ANÁLISIS Y DISEÑO

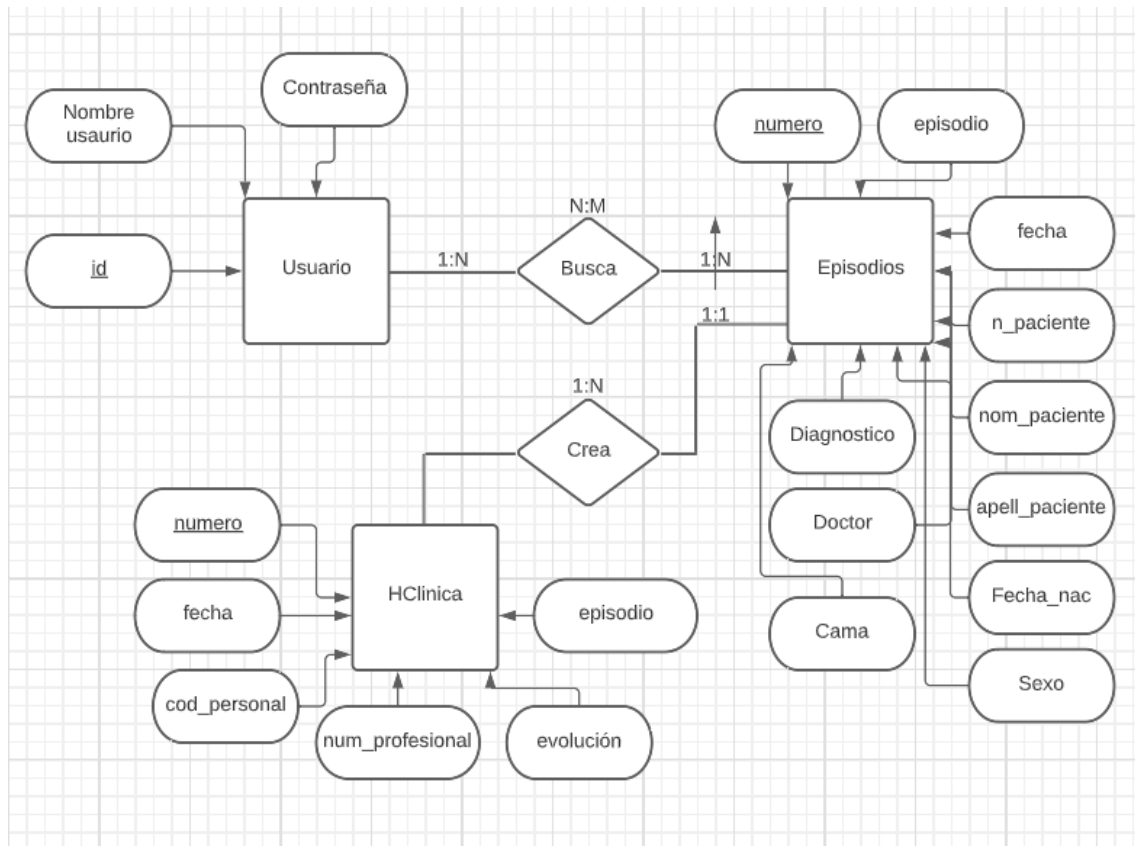
### MODELADO DE DATOS

La base de datos utilizada para este proyecto es Microsoft SQL Server.

Se ha elegido esta base de datos, ya que es la utilizada por la empresa Viamed Salud.

Para facilitar la gestión de la base de datos se ha decidido crear una Api Rest que se encarga de hacer las gestiones necesarias.

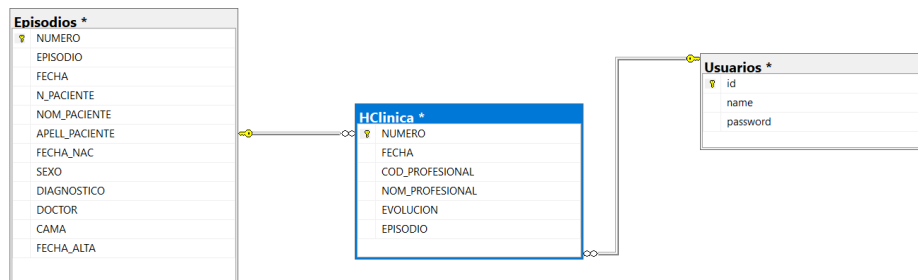
### Modelo Entidad Relación:



## Modelo Relacional:

En el proyecto no se ha utilizado ninguna relación en la base de datos, las relaciones entre tablas se hacían mediante programación.

Un posible modelo relacional sería este:

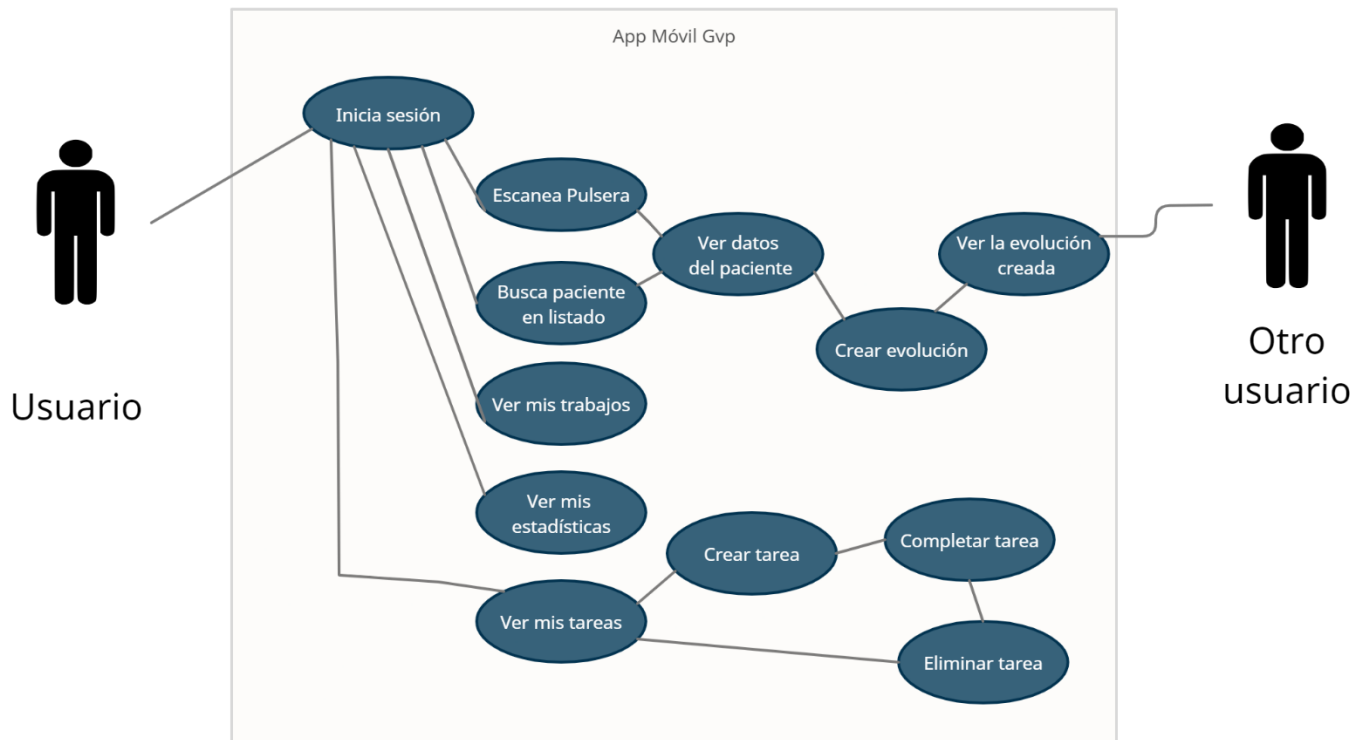


Algunas solicitudes a la base de datos mediante la Api Rest:

filter activity	
GET	localhost:4000/hclinica/episodio/AMN2242419
5 days ago	
POST	localhost:4000/users/authenticate
5 days ago	
GET	localhost:4000/users/
12 days ago	
GET	localhost:4000/hclinica/2
12 days ago	
GET	localhost:4000/hclinica/
12 days ago	
GET	localhost:4000/episodios/AMN2242419
14 days ago	
GET	localhost:4000/episodios/
14 days ago	
GET	localhost:4000/users/2
14 days ago	
POST	localhost:4000/users/register
14 days ago	



## Casos de uso:



CU-1	Iniciar Sesión
<b>Descripción:</b> El usuario inicia sesión.	
<b>Actores:</b> Usuario	
<b>Predicciones:</b> <ul style="list-style-type: none"> <li>- El usuario tiene que estar registrado.</li> <li>- Se requiere conexión a la red de Viamed.</li> </ul>	
<b>Caso normal de uso:</b> <ul style="list-style-type: none"> <li>- El usuario introduce sus datos.</li> <li>- El usuario pulsa “Iniciar Sesión”.</li> <li>- El usuario ya esta logeado y se muestra la pantalla de inicio.</li> </ul>	
<b>Poscondiciones:</b> Se inicia sesión y entra a la pantalla de inicio.	
<b>Alternativa:</b> Aparece mensaje de error.	

CU-2	Escanear Pulsera
<b>Descripción:</b> El usuario quiere escanear una pulsera.	
<b>Actores:</b> Usuario	
<b>Predicciones:</b> <ul style="list-style-type: none"> <li>- El usuario tiene que apuntar con la cámara a la pulsera.</li> <li>- El episodio del paciente existe en la base de datos.</li> </ul>	
<b>Caso normal de uso:</b> <ul style="list-style-type: none"> <li>- El usuario con la cámara apunta a la pulsera.</li> </ul>	
<b>Poscondiciones:</b> Se muestran los datos del paciente.	
<b>Alternativa:</b> Aparece mensaje de que no se ha encontrado ese episodio.	

CU-3	Listado Pacientes
<b>Descripción:</b> El usuario quiere ver el listado de pacientes	
<b>Actores:</b> Usuario	
<b>Predicciones:</b> <ul style="list-style-type: none"> <li>- El usuario elige un paciente de la lista.</li> <li>- Busca un paciente filtrándolo.</li> </ul>	
<b>Caso normal de uso:</b> <ul style="list-style-type: none"> <li>- Busca y selecciona un paciente de la lista.</li> </ul>	
<b>Poscondiciones:</b> Se muestran los datos del paciente. Se muestra la lista de pacientes filtrada	
<b>Alternativa:</b> Aparece mensaje de que no se ha encontrado ese episodio.	

CU-4	Mis trabajos
<b>Descripción:</b> El usuario quiere ver los trabajos que ha realizado	
<b>Actores:</b> Usuario	
<b>Predicciones:</b> <ul style="list-style-type: none"><li>- El usuario busca en los trabajos realizados</li><li>- Busca un paciente filtrándolo.</li></ul>	
<b>Caso normal de uso:</b> <ul style="list-style-type: none"><li>- Busca un trabajo realizado.</li></ul>	
<b>Poscondiciones:</b> Se muestran los trabajos realizados	
<b>Alternativa:</b> No se encuentra ningún trabajo	

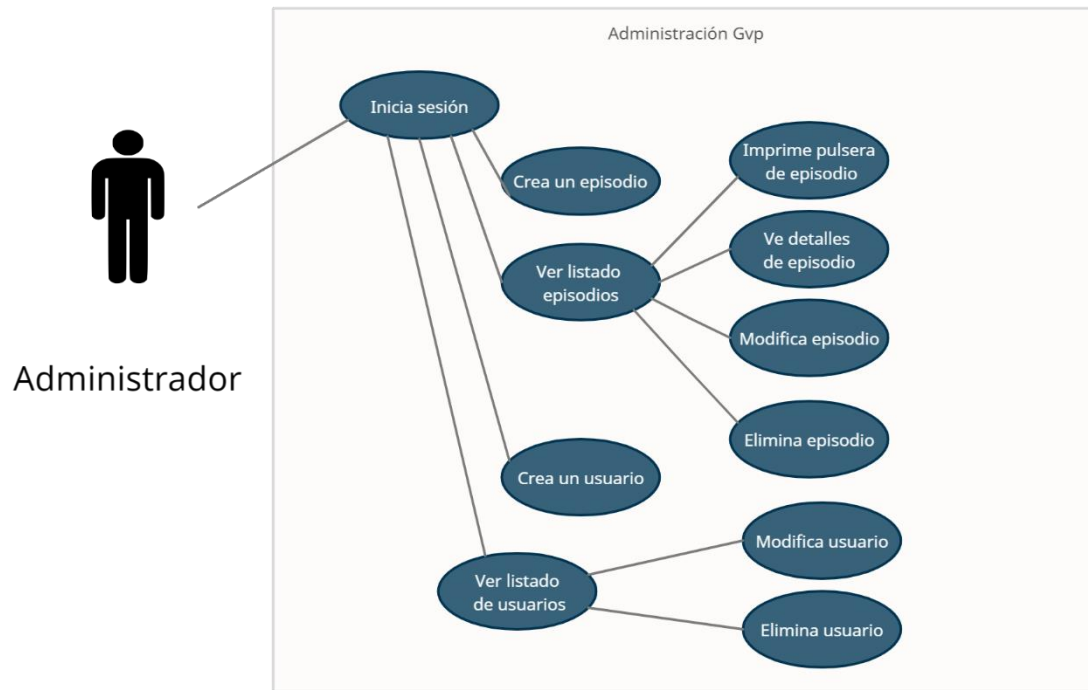
CU-5	Ver estadísticas
<b>Descripción:</b> El usuario quiere ver sus estadísticas	
<b>Actores:</b> Usuario	
<b>Predicciones:</b> <ul style="list-style-type: none"><li>- Para mostrar las estadísticas el usuario tiene que tener datos</li></ul>	
<b>Caso normal de uso:</b> <ul style="list-style-type: none"><li>- El usuario pulsa en mis estadísticas</li></ul>	
<b>Poscondiciones:</b> Se muestran las estadísticas el usuario	
<b>Alternativa:</b> No se encuentran estadísticas	

CU-6	Realizar Tareas
<b>Descripción:</b> El usuario quiere realizar una tarea	
<b>Actores:</b> Usuario	
<b>Predicciones:</b> - Debe escribir la tarea	
<b>Caso normal de uso:</b> - El usuario presiona el botón para realizar una tarea	
<b>Poscondiciones:</b> Se crea la tarea	
<b>Alternativa:</b> No se crea la tarea y muestra error	

CU-7	Completar Tarea
<b>Descripción:</b> El usuario quiere completar una tarea	
<b>Actores:</b> Usuario	
<b>Predicciones:</b> - Debe deslizar la tarea para completarla	
<b>Caso normal de uso:</b> - El usuario desliza y completa la tarea	
<b>Poscondiciones:</b> La tarea sale como completada	
<b>Alternativa:</b> Ninguna	

CU-8	Eliminar Tarea
<b>Descripción:</b> El usuario quiere eliminar una tarea	
<b>Actores:</b> Usuario	
<b>Predicciones:</b> - Debe estar la tarea completada para ser eliminada	
<b>Caso normal de uso:</b> - El usuario desliza y elimina la tarea	
<b>Poscondiciones:</b> La tarea se elimina correctamente	
<b>Alternativa:</b> Ninguna	

CU-9	Crear evolución
<b>Descripción:</b> El usuario quiere crear una evolución al paciente	
<b>Actores:</b> Usuario	
<b>Predicciones:</b> - Debe elegir entre las opciones de evolución	
<b>Caso normal de uso:</b> - Elige una opción y pulsa crear evolución	
<b>Poscondiciones:</b> La evolución se crea correctamente	
<b>Alternativa:</b> Se muestra error al crear evolución	



CU-10	Crear episodio
<b>Descripción:</b> El usuario quiere crear un episodio	
<b>Actores:</b> Usuario	
<b>Predicciones:</b> - Debe rellenar todas las opciones obligatorias	
<b>Caso normal de uso:</b> - Rellena todo y crea un episodio	
<b>Poscondiciones:</b> El episodio se crea correctamente	
<b>Alternativa:</b> Se muestra error al crear el episodio	

CU-11	Ver listado episodios
<b>Descripción:</b> El usuario quiere ver todos los episodios	
<b>Actores:</b> Usuario	
<b>Predicciones:</b> - Presiona listado de episodios	
<b>Caso normal de uso:</b> - Visualiza todos los episodios	
<b>Poscondiciones:</b> Se visualizan todos los episodios	
<b>Alternativa:</b> No existen episodios o muestra error	

CU-12	Imprimir pulsera
<b>Descripción:</b> El usuario quiere imprimir pulsera	
<b>Actores:</b> Usuario	
<b>Predicciones:</b> - Presiona en la lista el episodio que quiere imprimir	
<b>Caso normal de uso:</b> - Elige un episodio y presiona el botón imprimir	
<b>Poscondiciones:</b> Se le genera un pdf para poder imprimir	
<b>Alternativa:</b> Ninguna	

CU-13	Ver detalles episodio
<b>Descripción:</b> El usuario quiere ver los detalles de un episodio	
<b>Actores:</b> Usuario	
<b>Predicciones:</b> - Presiona en la lista el episodio que ver los detalles	
<b>Caso normal de uso:</b> - Elige un episodio y presiona el botón ver detalles	
<b>Poscondiciones:</b> Se le muestra en una pantalla los detalles del episodio seleccionado	
<b>Alternativa:</b> Ninguna	

CU-14	Eliminar episodio
<b>Descripción:</b> El usuario quiere eliminar un episodio	
<b>Actores:</b> Usuario	
<b>Predicciones:</b> - Presiona en la lista el episodio que quiere eliminar	
<b>Caso normal de uso:</b> - Elige un episodio y presiona el botón de eliminar	
<b>Poscondiciones:</b> Se le muestra un mensaje de episodio eliminado	
<b>Alternativa:</b> Se le muestra error al eliminar episodio	



CU-15	Crear Usuario
<b>Descripción:</b> El usuario quiere crear un nuevo usuario	
<b>Actores:</b> Usuario	
<b>Predicciones:</b> - Presiona en crear usuario	
<b>Caso normal de uso:</b> - Rellena todos los datos para crear el usuario	
<b>Poscondiciones:</b> Se le muestra un mensaje de usuario creado correctamente	
<b>Alternativa:</b> Se le muestra error al crear usuario	

CU-16	Modificar Usuario
<b>Descripción:</b> El usuario quiere modificar un usuario	
<b>Actores:</b> Usuario	
<b>Predicciones:</b> - Presiona modificar usuario en la lista de usuarios	
<b>Caso normal de uso:</b> - Rellena los datos para poder modificar el usuario	
<b>Poscondiciones:</b> Se le muestra un mensaje de usuario modificado correctamente	
<b>Alternativa:</b> Se le muestra error al modificar usuario	

CU-17	Eliminar usuario
<b>Descripción:</b> El usuario quiere eliminar un usuario	
<b>Actores:</b> Usuario	
<b>Predicciones:</b> - Presiona en la lista el usuario que quiere eliminar	
<b>Caso normal de uso:</b> - Elige un usuario y presiona el botón de eliminar	
<b>Poscondiciones:</b> Se le muestra un mensaje de usuario eliminado	
<b>Alternativa:</b> Se le muestra error al eliminar usuario	

## ANÁLISIS Y DISEÑO DE LA INTERFAZ DE USUARIO

Para poder analizar y diseñar la interfaz de usuario se ha creado un diseño UX. (User Experience Design) con el propósito de definir el diseño de la experiencia de usuario.

### **Aplicación GVP Móvil:**

Al iniciar la aplicación, aparece un “Splash Screen” (Pantalla de carga) con el logo de Viamed Salud.



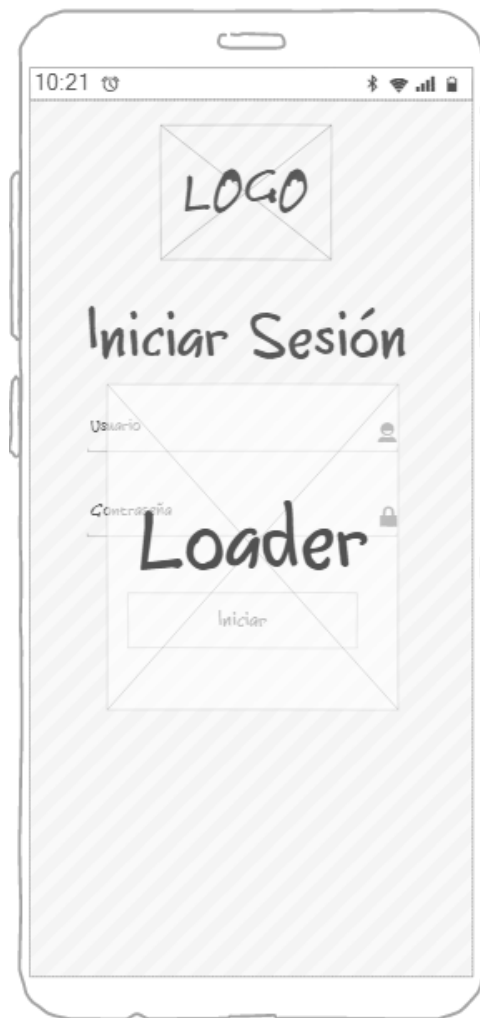
*Ilustración 1: Splash Screen*

Después de la pantalla de carga, en el caso de no tener la sesión iniciada con anterioridad, aparece el inicio de sesión:



*Ilustración 2: Pantalla de Inicio de Sesión*

Tras la pantalla de carga después de iniciar sesión, se muestra un loader (pantalla de carga):



*Ilustración 3: Pantalla Loader*

Una vez cargada, se muestra la pantalla de home con múltiples opciones.



*Ilustración 4: Pantalla Home*

En caso de pulsar el botón de escanear pulsera aparece la pantalla de escáner:



*Ilustración 5: Pantalla Escanear Pulsera*

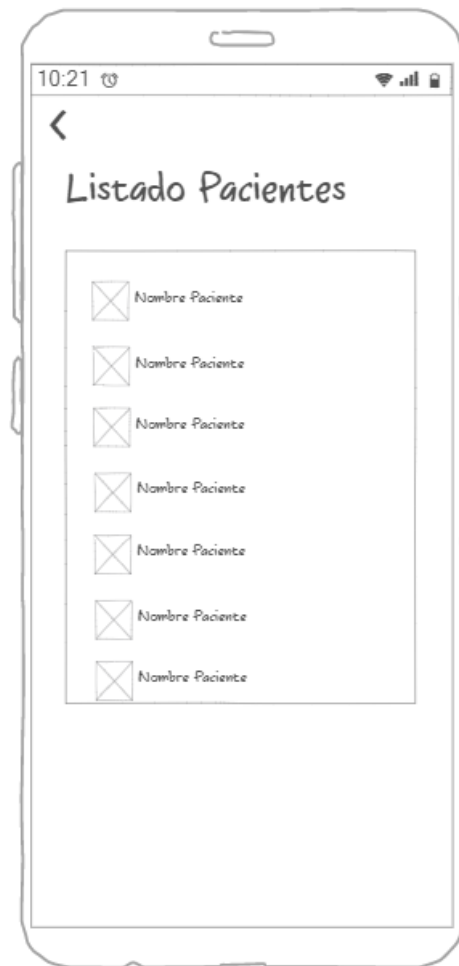
Si se escanea la pulsera correctamente se muestra la pantalla de datos del paciente:



*Ilustración 6: Pantalla Datos Paciente*

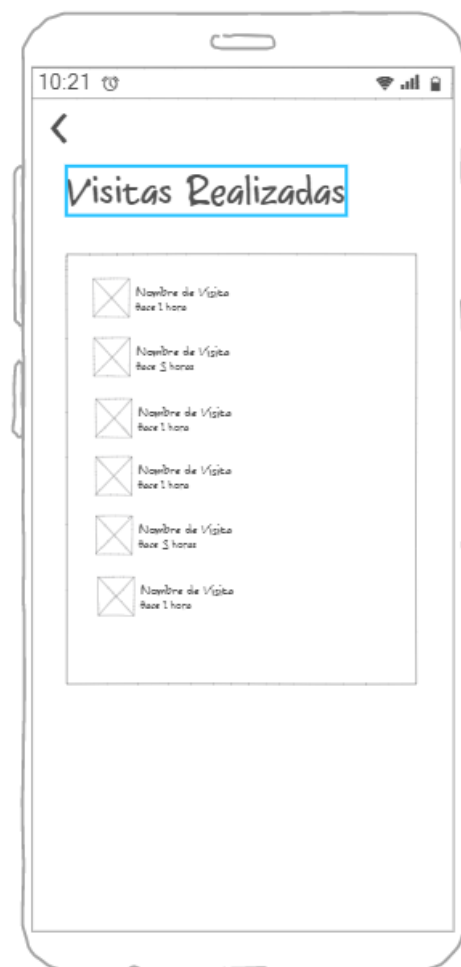


Si en la pestaña de home el usuario pulsa en el botón “listado de paciente” se muestra la pantalla de listado de pacientes:



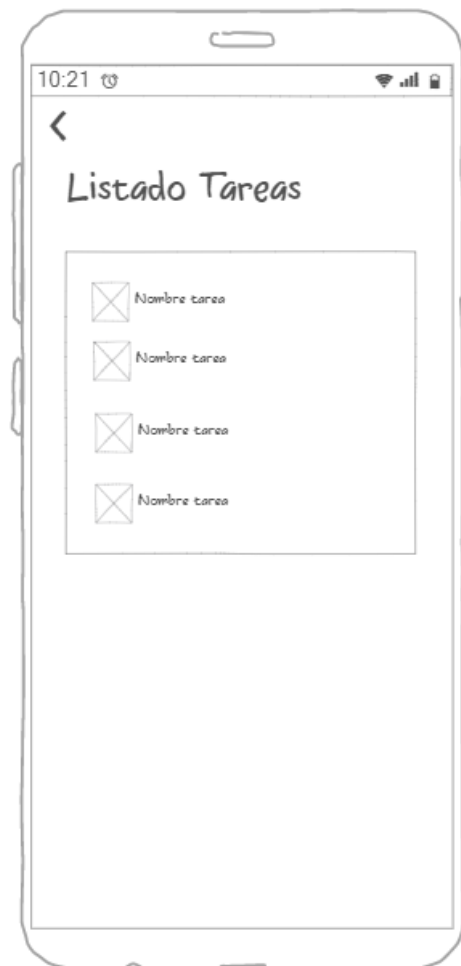
*Ilustración 7: Pantalla Listado de pacientes*

Si el usuario en la pantalla de home pulsa en visitas se muestran las visitas o trabajos realizados por el usuario:



*Ilustración 8: Pantalla Visitas o trabajos realizados*

Si el usuario en la pantalla home pulsa en el botón “Tareas” se muestra la pantalla de las tareas:



*Ilustración 9: Pantalla mis tareas*

Si el usuario pulsa abajo a la derecha para ver su perfil se muestra la pantalla de perfil:



*Ilustración 10: Pantalla mi perfil*

Si pulsa en el botón cerrar sesión, el usuario vuelve a la pantalla de iniciar sesión y se cierra la sesión

Si pulsa en el botón mis estadísticas se visualiza las estadísticas del usuario:



*Ilustración 10: Pantalla mis estadísticas*

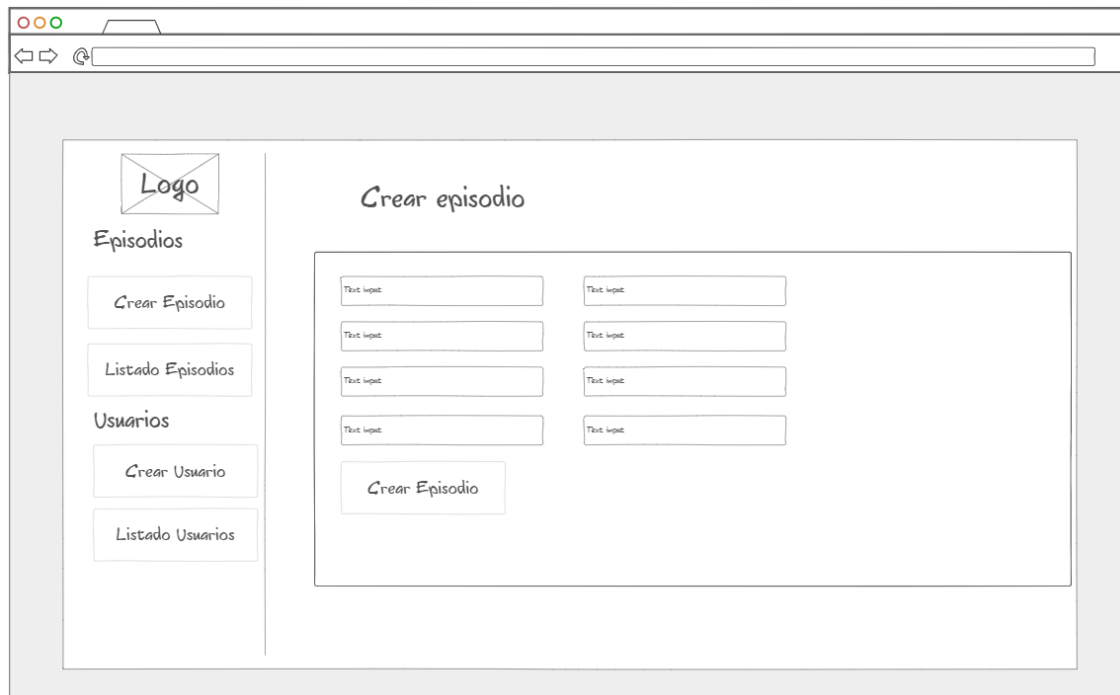
## Aplicación Administración GVP Web:

Pantalla de login o inicio de sesión esta pantalla se mostrará si el usuario no ha iniciado sesión.



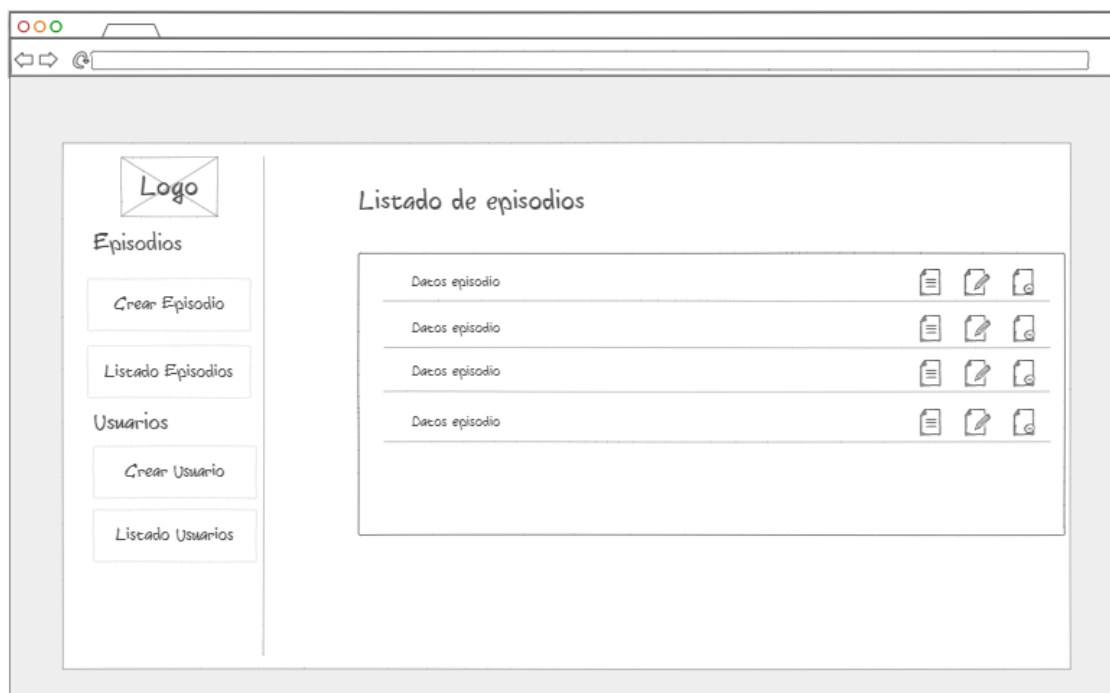
*Ilustración 11: Pantalla mis estadísticas*

La pantalla que se muestra al administrador después de iniciar sesión correctamente, es la pantalla de crear episodio.



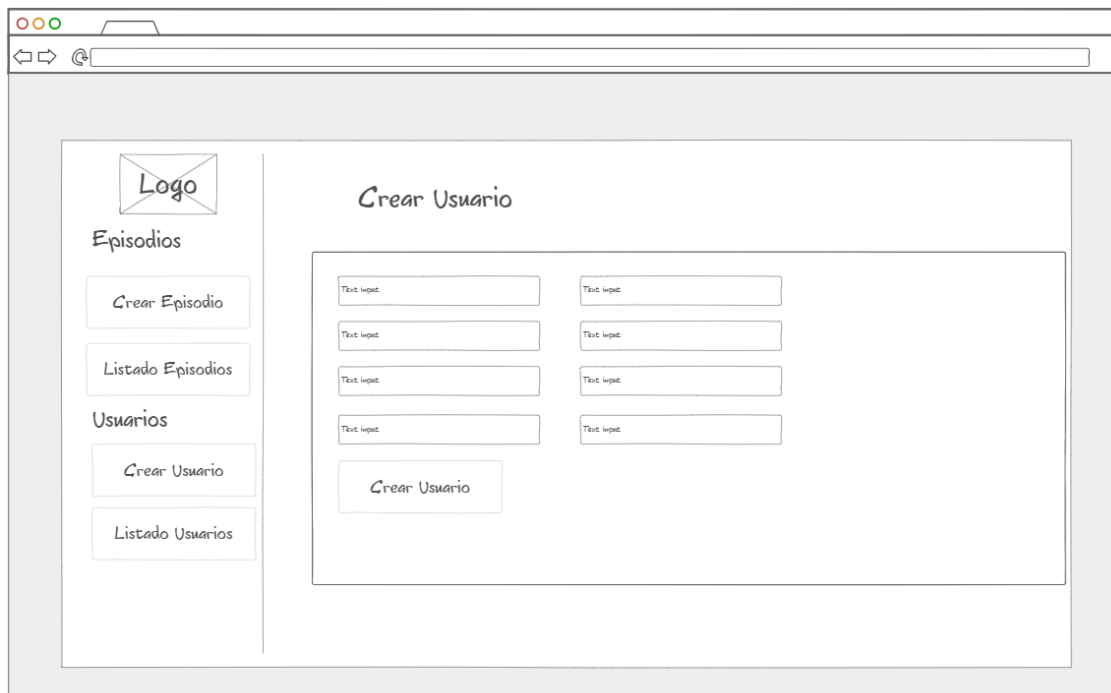
*Ilustración 12: Pantalla crear episodio*

Si el usuario pulsa en listado de episodios se muestra la pantalla de listados de episodios.



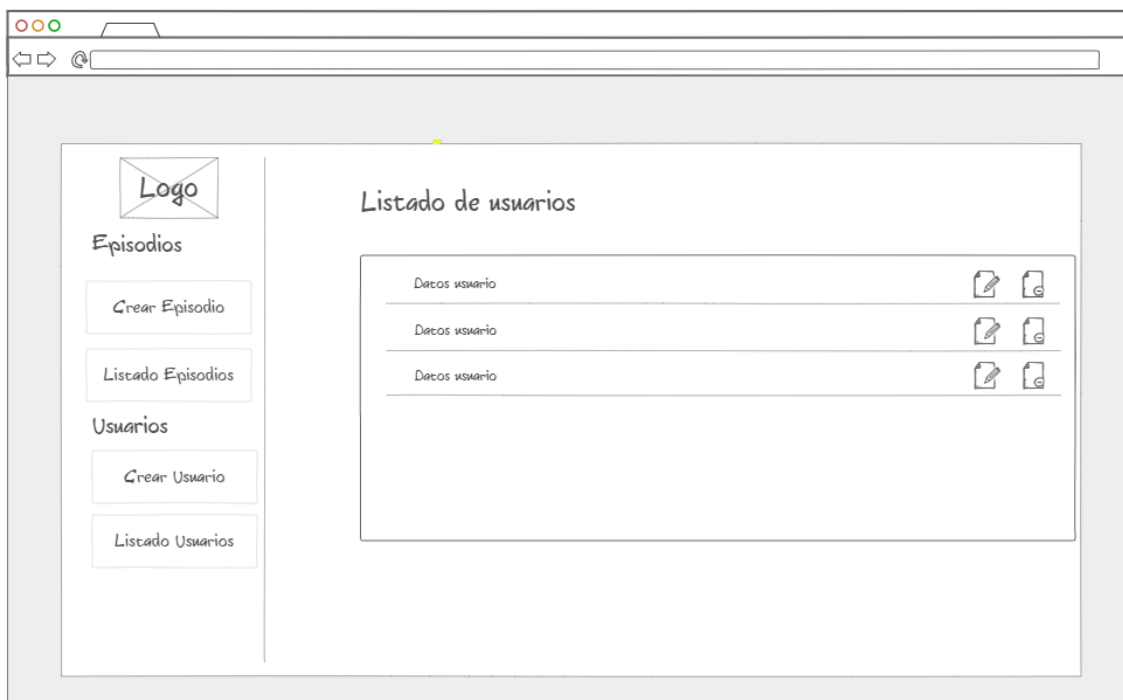
*Ilustración 13: Pantalla listado de episodios*

Si el usuario pulsa en el menú crear usuario se mostrar la siguiente pantalla.



*Ilustración 14: Pantalla crear usuario*

Al presionar en listado de usuarios se muestra la pantalla de listado de usuarios (Ilustración 15).

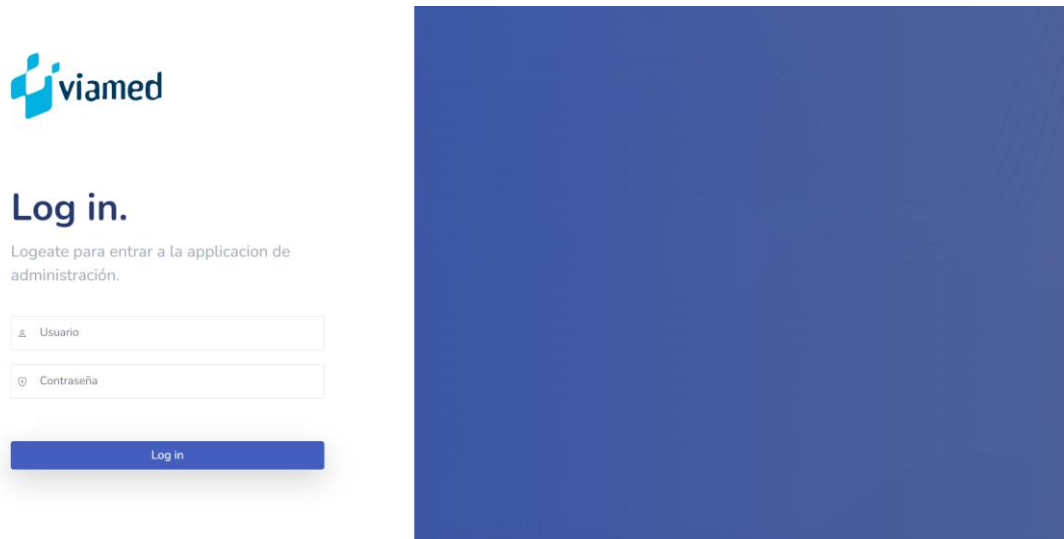


*Ilustración 15: Pantalla listado usuarios*

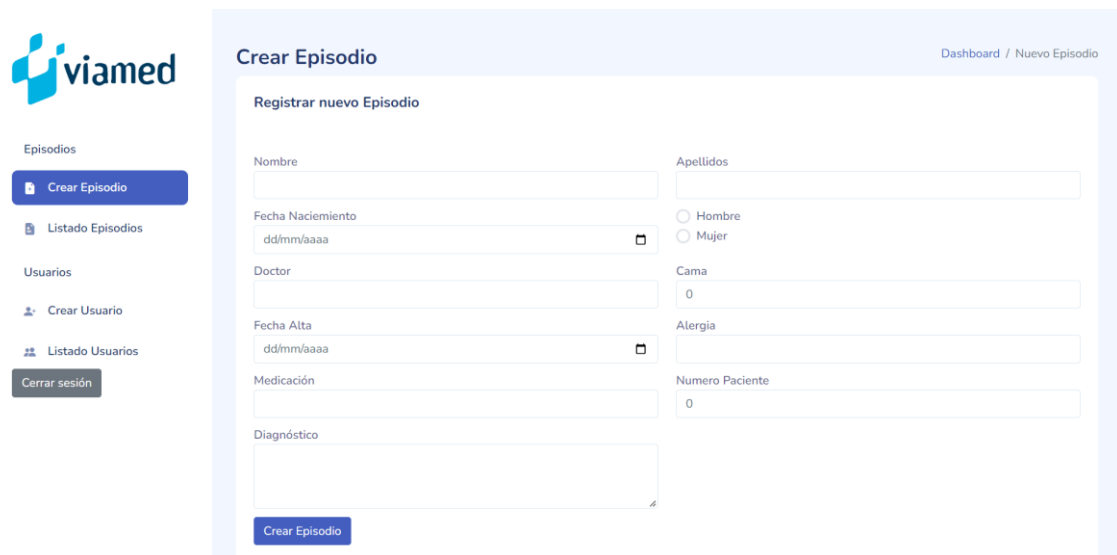


Una vez se ha creado el diseño UX se procedió al diseño real UI de la aplicación quedado como se muestra a continuación:

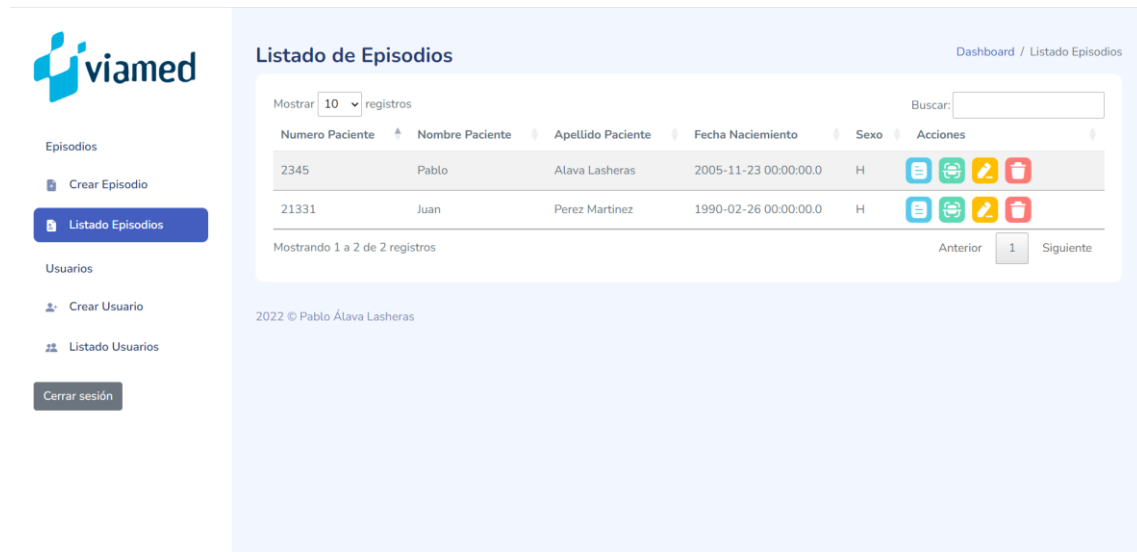
### Aplicación GVP Administración Web



*Ilustración 16: Pantalla inicio de sesión*



*Ilustración 17: Pantalla crear episodio*



**viamed**

Episodios

Crear Episodio

Listado Episodios

Usuarios

Crear Usuario

Listado Usuarios

Cerrar sesión

**Listado de Episodios** Dashboard / Listado Episodios

Mostrar 10 registros

Buscar:

Numero Paciente	Nombre Paciente	Apellido Paciente	Fecha Nacimiento	Sexo	Acciones
2345	Pablo	Alava Lasheras	2005-11-23 00:00:00.0	H	[Iconos de acciones]
21331	Juan	Perez Martinez	1990-02-26 00:00:00.0	H	[Iconos de acciones]

Mostrando 1 a 2 de 2 registros

Anterior 1 Siguiente

2022 © Pablo Álava Lasheras

*Ilustración 18: Pantalla listado episodios*



**viamed**

Episodios

Crear Episodio

Listado Episodios

Usuarios

Crear Usuario

Listado Usuarios

Cerrar sesión

**Detalles Episodio** Dashboard / Detalles Episodio

**Detalles Episodio**

Nombre	Pablo	Apellidos	Alava Lasheras
Fecha Nacimiento	23/11/2005	Sexo	H
Doctor	Pepito	Cama	208
Fecha Alta	28/12/2022	Alergia	A algo
Medicación		Numero Paciente	2345
Diagnóstico	Tiene las rodillas mal		

*Ilustración 19: Pantalla detalles episodio*

The screenshot shows the 'Modificar Episodio' (Modify Episode) form in the viamed application. The form is divided into two columns. The left column contains fields for 'Nombre' (Pablo), 'Fecha Nacimiento' (23/11/2005), 'Doctor' (Pepito), 'Fecha Alta' (28/12/2022), 'Medicación', and 'Diagnóstico' (Tiene las rodillas mal). The right column contains fields for 'Apellidos' (Alava Lasheras), 'Sexo' (H), 'Cama' (208), 'Alergia' (A algo), and 'Numero Paciente' (2345). A 'Modificar Episodio' button is at the bottom left of the form. The sidebar on the left shows navigation options: 'Episodios' (Crear Episodio, Listado Episodios), 'Usuarios' (Crear Usuario, Listado Usuarios), and 'Cerrar sesión'. The breadcrumb at the top right reads 'Dashboard / Modificar Episodio'.

*Ilustración 20: Pantalla modificar episodio*

The screenshot shows the 'Listado de Episodios' (List of Episodes) page in the viamed application. A confirmation dialog is displayed in the center, asking 'Esta seguro de eliminar?' (Are you sure you want to delete?). The dialog includes a warning icon, the text 'Si elimina el episodio no se podrá recuperarlo!' (If you delete the episode, it cannot be recovered!), and two buttons: 'Eliminar!' (Delete!) and 'Cancelar' (Cancel). The background shows a table with columns: 'Numero Paciente', 'Nombre Paciente', 'Apellido Paciente', 'Fecha Nacimiento', 'Sexo', and 'Acciones'. The table lists two episodes: one with 'Numero Paciente' 2345 and another with 'Numero Paciente' 21331. The sidebar on the left shows navigation options: 'Episodios' (Crear Episodio, Listado Episodios), 'Usuarios' (Crear Usuario, Listado Usuarios), and 'Cerrar sesión'. The breadcrumb at the top right reads 'Dashboard / Listado Episodios'.

*Ilustración 21: Alerta eliminar episodio*

viamed

Episodios

- Crear Episodio
- Listado Episodios

Usuarios

- Crear Usuario
- Listado Usuarios
- Cerrar sesión

Crear Usuario

Dashboard / Nuevo Usuario

Registrar nuevo Usuario

Nombre Usuario

Contraseña

Crear

2022 © Pablo Álava Lasheras

*Ilustración 22: Pantalla crear usuario*

viamed

Episodios

- Crear Episodio
- Listado Episodios

Usuarios

- Crear Usuario
- Listado Usuarios
- Cerrar sesión

Listado de Usuarios

Dashboard / Listado Usuarios

Mostrar 10 registros

Buscar:

Usuario	Contraseña	Fecha Creación	Fecha Modificación	Acciones
pablo	*****	1990-12-12 00:00:00.0	1990-12-12 00:00:00.0	

Mostrando 1 a 1 de 1 registros

Anterior 1 Siguiente

2022 © Pablo Álava Lasheras

*Ilustración 23: Pantalla listado usuarios*

viamed

Episodios

- Crear Episodio
- Listado Episodios

Usuarios

- Crear Usuario
- Listado Usuarios
- Cerrar sesión

Modificar Usuario

Dashboard / Modificar Usuario

Modificar Usuario

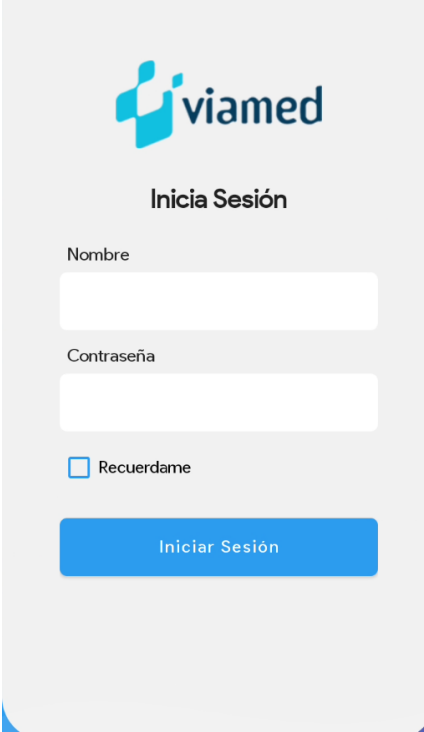
Nombre Usuario

Contraseña

Actualizar

2022 © Pablo Álava Lasheras

*Ilustración 24: Pantalla modificar usuario*

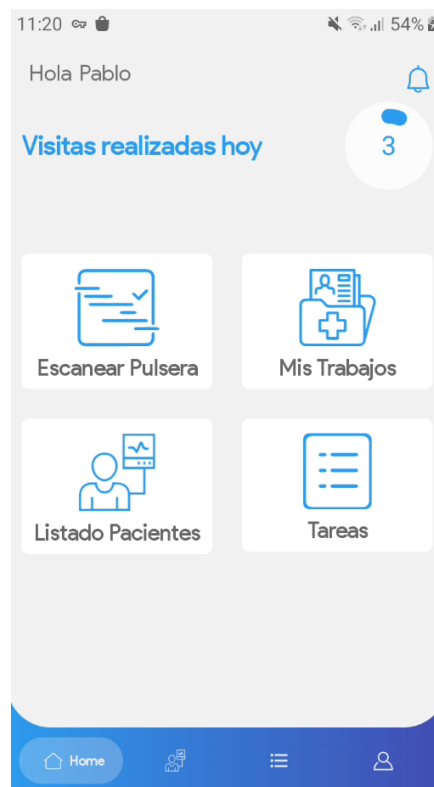
**GVP Aplicación móvil:**

The image shows a mobile application login screen for 'viamed'. At the top is the 'viamed' logo, which consists of a blue stylized icon followed by the text 'viamed'. Below the logo is the title 'Inicia Sesión'. There are two input fields: 'Nombre' (Name) and 'Contraseña' (Password). Below the password field is a checkbox labeled 'Recuerdame' (Remember me). At the bottom is a blue button with the text 'Iniciar Sesión' (Log In).

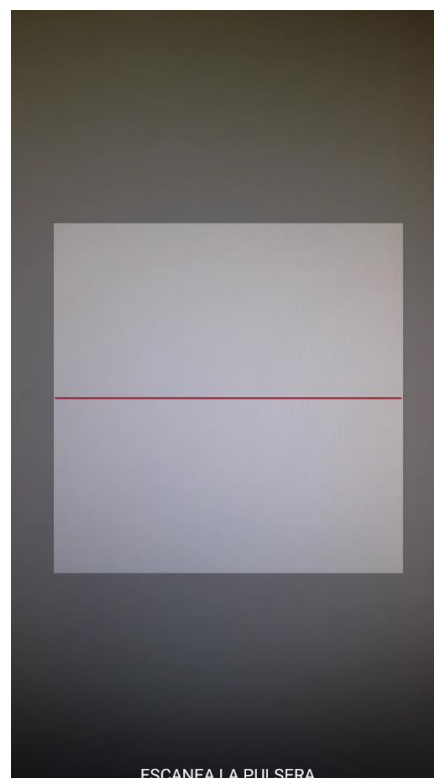
*Ilustración 24: Pantalla iniciar sesión*

This image is a second mockup of the same login screen, but with sample data and a loading overlay. The 'Nombre' field contains the text 'pablo' and the 'Contraseña' field contains five dots. A large, light blue, stylized 'viamed' logo is overlaid on the input fields. The 'Recuerdame' checkbox is unchecked, and the 'Iniciar Sesión' button is still visible at the bottom.

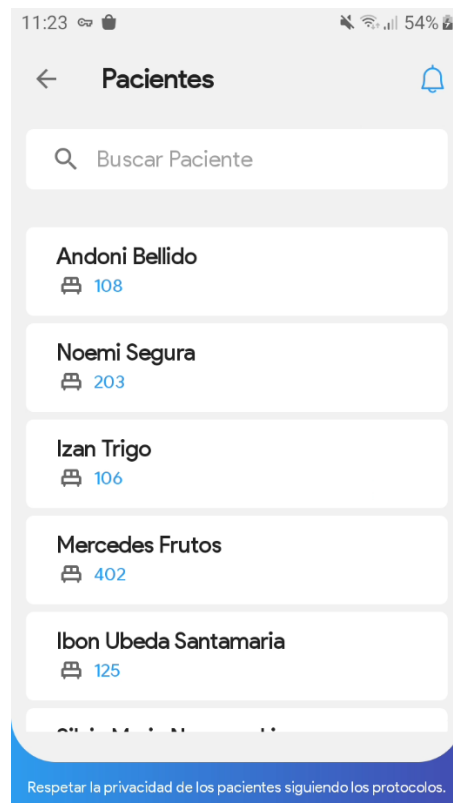
*Ilustración 25: Pantalla loader*



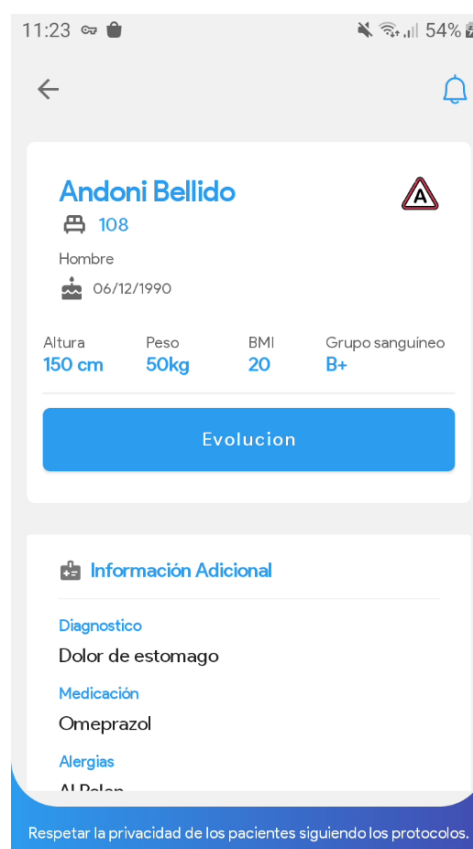
*Ilustración 26: Pantalla home*



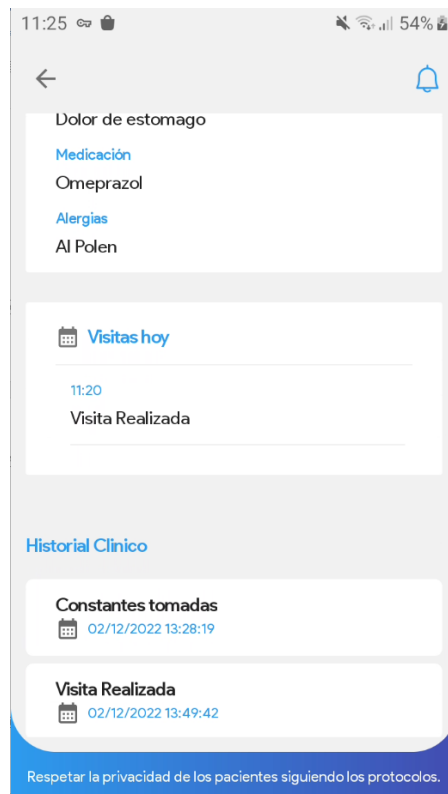
*Ilustración 27: Pantalla escanear pulsera*



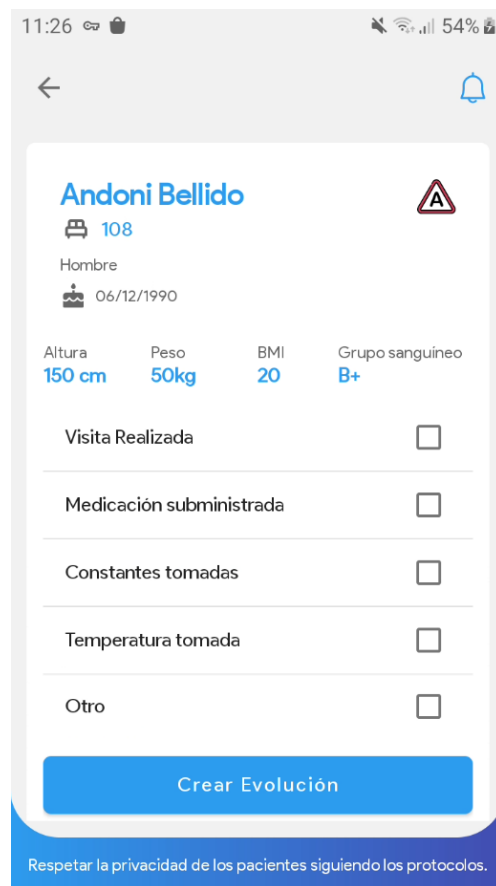
*Ilustración 28: Pantalla listado pacientes*



*Ilustración 29: Pantalla datos paciente*

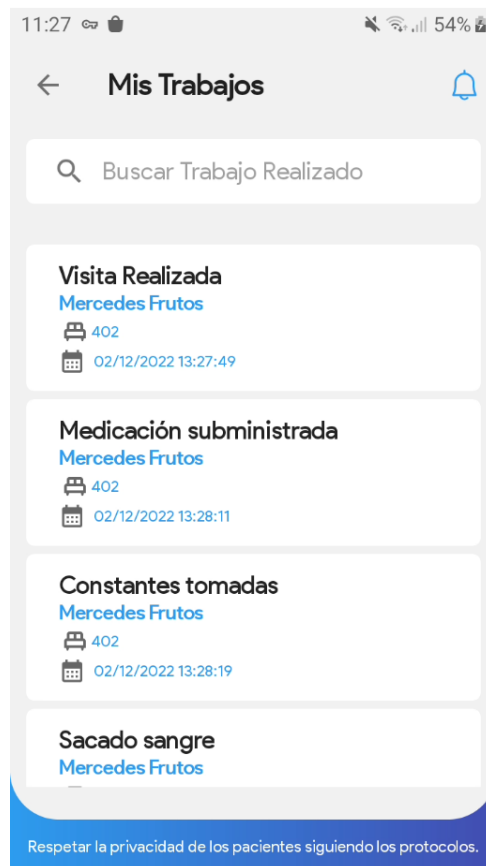


*Ilustración 30: Pantalla datos paciente 2*

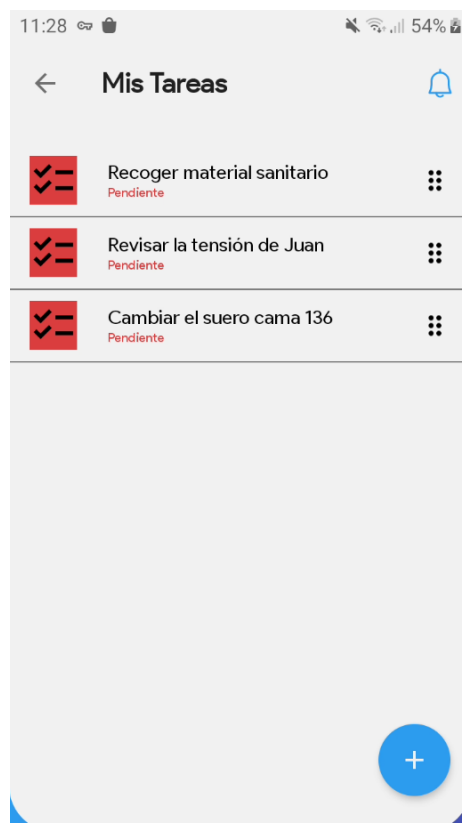


*Ilustración 31: Opción crear evolución*

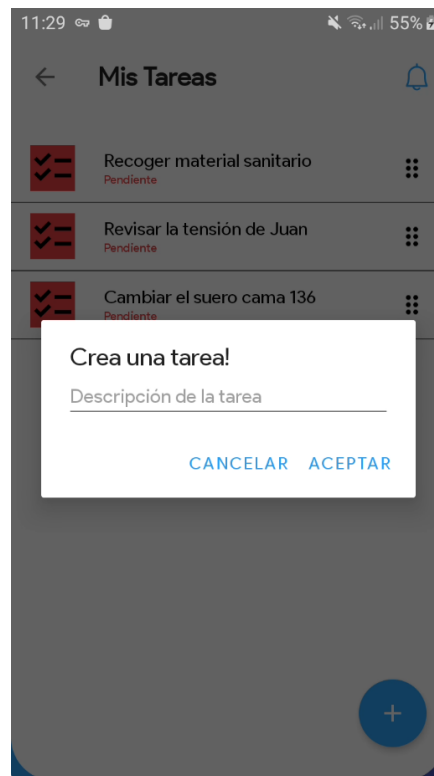




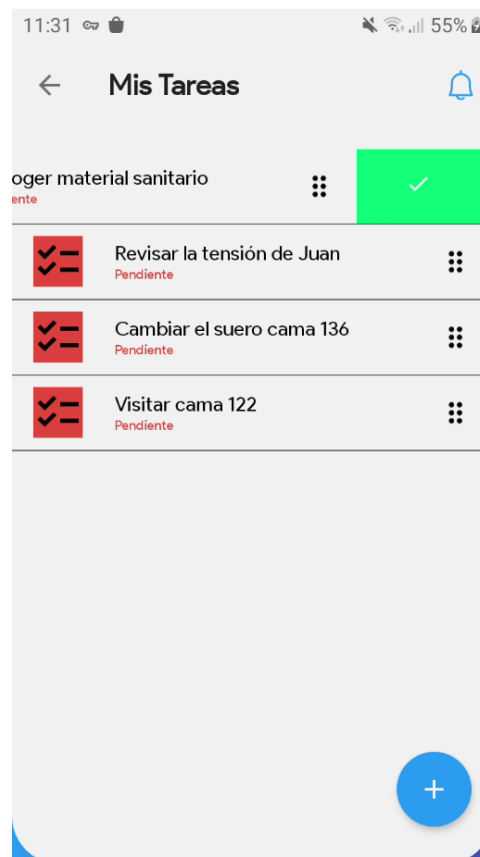
*Ilustración 32: Pantalla mis trabajos*



*Ilustración 33: Pantalla mis tareas*



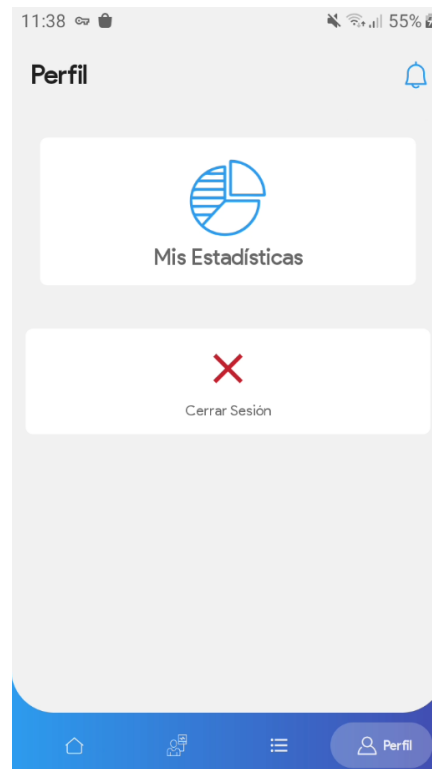
*Ilustración 34: Pop-up crear una tarea*



*Ilustración 35: Completando una tarea*



*Ilustración 36: Eliminando una tarea*



*Ilustración 37: Pantalla perfil*

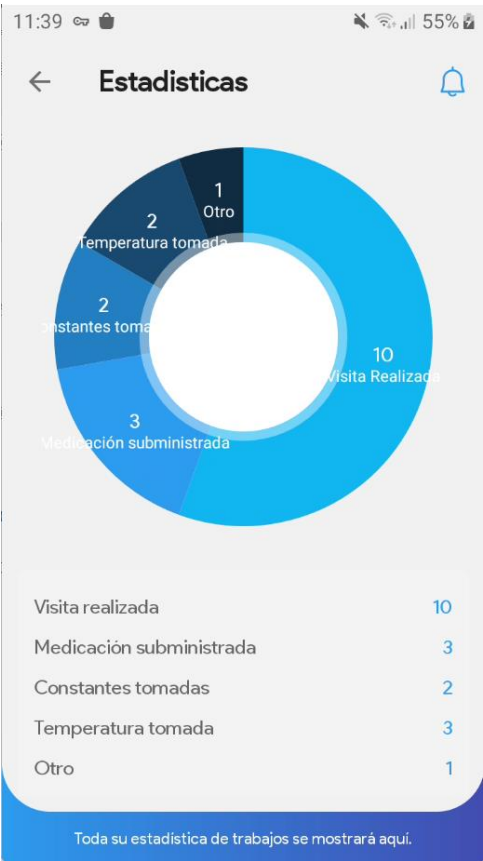


Ilustración 38: Pantalla estadísticas

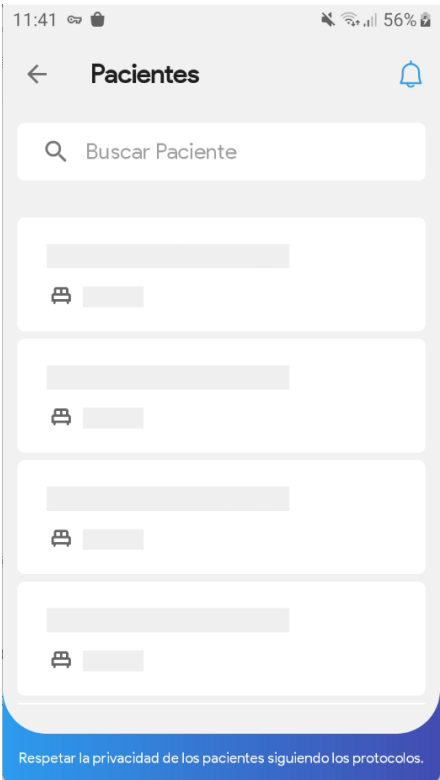


Ilustración 39: Shimmer lista pacientes

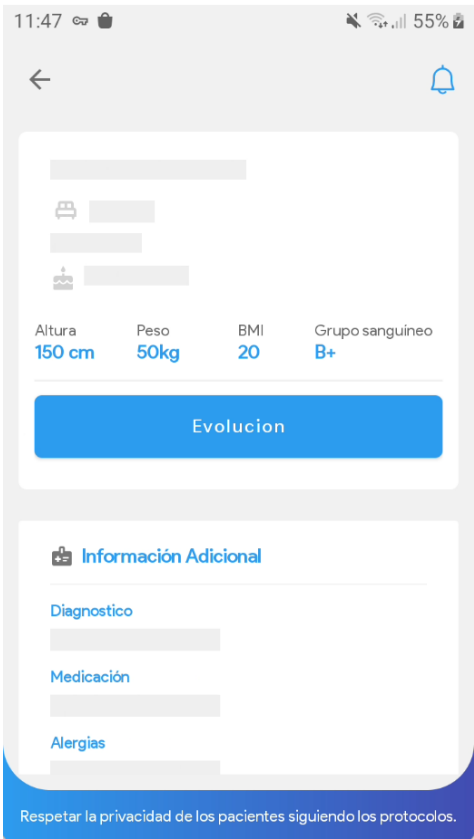


Ilustración 40: Shimmer datos paciente



Ilustración 41: Shimmer mis trabajos

## ANÁLISIS Y DISEÑO DEL SISTEMA FUNCIONAL

En el análisis y diseño del sistema funcional se describe el funcionamiento que deberá realizar cada componente para poder desarrollarlo y hacer las aplicaciones funcionales.

### GVP Administrador Web:



## Log in.

Logeate para entrar a la aplicacion de administración.

Log in 3

*Ilustración 42: Pantalla inició de sesión*

1. El usuario debe introducir el nombre de usuario.
2. El usuario debe introducir su contraseña.
3. Al pulsar, si los campos 1 y 2 son correctos se redirigirá a la pantalla principal.

Pantalla que se muestra al iniciar sesión correctamente:

**viamed**

Episodios

- Crear Episodio
- Listado Episodios 13

Usuarios

- Crear Usuario 14
- Listado Usuarios 15
- Cerrar sesión 16

**Crear Episodio**

Dashboard / Nuevo Episodio

Registrar nuevo Episodio

Nombre 1

Apellidos 7

Fecha Nacimiento dd/mm/aaaa 2

☐ Hombre ☐ Mujer 8

Doctor 3

Cama 0 9

Fecha Alta dd/mm/aaaa 4

Alergia 10

Medicación 5

Numero Paciente 0 11

Diagnóstico 6









Crear Episodio 12

*Ilustración 43: Pantalla crear episodio*

1. Introducir el nombre del paciente.
2. Introducir fecha nacimiento con el formato específico.
3. Introducir el nombre del doctor.
4. Introducir la fecha de alta del paciente.
5. Introducir la medicación.
6. Introducir el diagnóstico del paciente.
7. Introducir los apellidos del paciente.
8. Introducir el sexo.
9. Introducir el número de cama.
10. Introducir las alergias del paciente.
11. Introducir el numero identificador del paciente.
12. Al pulsar si se han rellenado los campos correctamente se creará un episodio.
13. Muestra el listado de episodios.
14. Muestra crear usuario.
15. Muestra el listado de usuarios.
16. Al pulsar se cierra sesión.

**Listado de Episodios** Dashboard / Listado Episodios

Mostrar  registros Buscar:

Numero Paciente	Nombre Paciente	Apellido Paciente	Fecha Nacimiento	Sexo	Acciones
2345	Pablo	Alava Lasheras	2005-11-23 00:00:00.0	H	<div>1 2 3 4</div> <div>   </div>
21331	Juan	Perez Martinez	1990-02-26 00:00:00.0	H	<div>   </div>

Mostrando 1 a 2 de 2 registros Anterior  Siguiente 6

2022 © Pablo Álava Lasheras

*Ilustración 44: Pantalla listado episodios*

1. Al pulsar se muestran los detalles del episodio.
2. Se crea un pdf con la pulsera para poder imprimirla.
3. Se muestra la pantalla modificar episodio con los datos del episodio.
4. Al pulsar se mostrará pop-up preguntando si se desea eliminar el episodio seleccionado.
5. Si escribimos se filtrará la tabla a tiempo real.
6. Paginación de la tabla.



Crear Usuario

Dashboard / Nuevo Usuario

Registrar nuevo Usuario

Nombre Usuario

Nombre Usuario 1

Contraseña

Contraseña 2

3 Crear

2022 © Pablo Álava Lasheras

*Ilustración 45: Pantalla crear usuario*

1. Introducir el nombre del nuevo usuario.
2. Introducir la contraseña.
3. Al pulsar se creará un nuevo usuario.

Listado de Usuarios

Dashboard / Listado Usuarios

Mostrar 10 registros

Buscar: 3

Usuario	Contraseña	Fecha Creación	Fecha Modificación	Acciones
pablo	*****	1990-12-12 00:00:00.0	1990-12-12 00:00:00.0	1 2

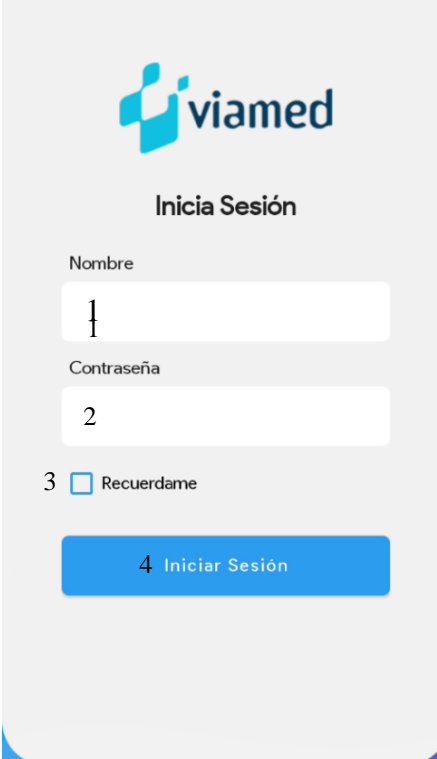
Mostrando 1 a 1 de 1 registros

Anterior 1 Siguiente 4

2022 © Pablo Álava Lasheras

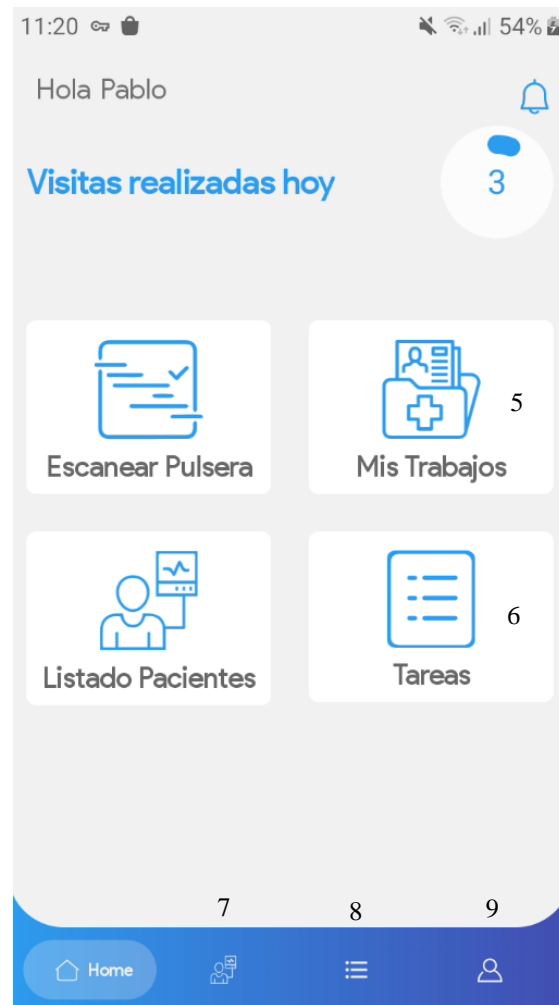
*Ilustración 46: Pantalla listado usuarios*

1. Al pulsar se mostrará la pantalla modificar usuario con los datos del usuario seleccionado.
2. Al pulsar se mostrará pop-up preguntando si se desea eliminar el usuario seleccionado.
3. Si escribimos se filtrará la tabla a tiempo real.
4. Paginación de la tabla.

**GVP Aplicación móvil:**

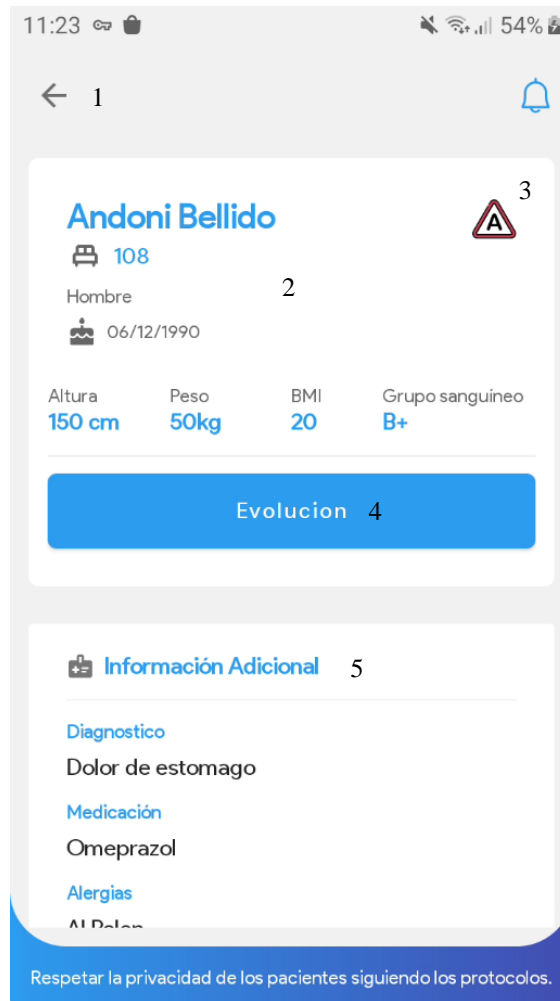
*Ilustración 47: Pantalla inicio sesión*

1. El usuario debe introducir el nombre de sesión.
2. El usuario debe introducir su contraseña.
3. Al pulsar se le recordaran los datos.
4. Al pulsar si se ha introducido bien el usuario y contraseña se le mostrará la pantalla de home.



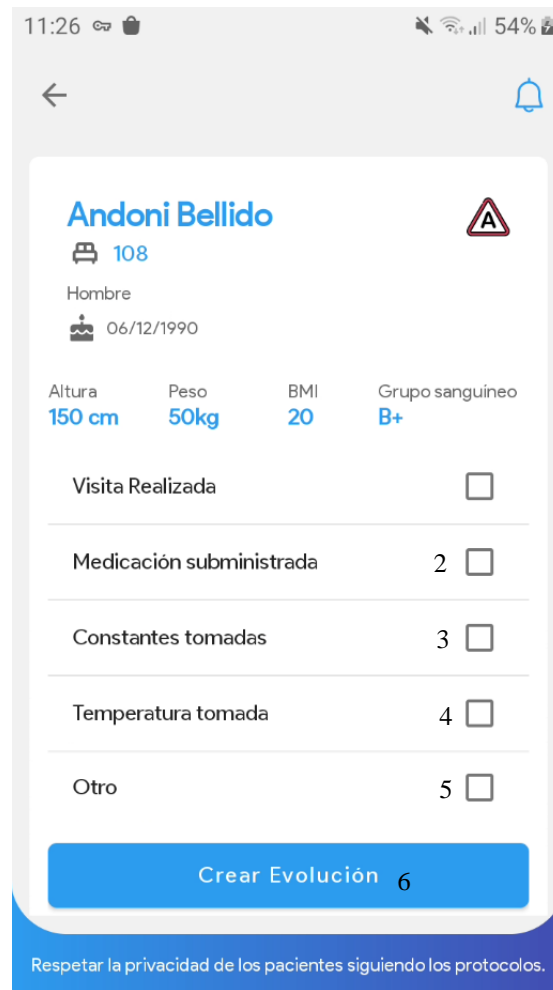
*Ilustración 48: Pantalla home*

1. Se mostrará el nombre de usuario.
2. Se mostrará el numero de visitas realizadas ese día por el usuario.
3. Muestra la pantalla escanear pulsera.
4. Muestra la pantalla listado de pacientes.
5. Muestra la pantalla mis trabajos.
6. Muestra la pantalla tareas.
7. Muestra la pantalla listado de pacientes.
8. Muestra la pantalla tareas.
9. Muestra la pantalla mi perfil.



*Ilustración 49: Pantalla datos paciente*

1. Al pulsar se volverá atrás.
2. Se muestra los datos del paciente seleccionado o escaneado.
3. Se muestran alertas del paciente (alérgico o menor).
4. Al pulsar se muestran las opciones crear evolución.
5. Muestra la información adicional del paciente.



11:26 54%

←

**Andoni Bellido**

108

Hombre

06/12/1990

Altura **150 cm**    Peso **50kg**    BMI **20**    Grupo sanguíneo **B+**

Visita Realizada ☐

Medicación subministrada 2 ☐

Constantes tomadas 3 ☐

Temperatura tomada 4 ☐

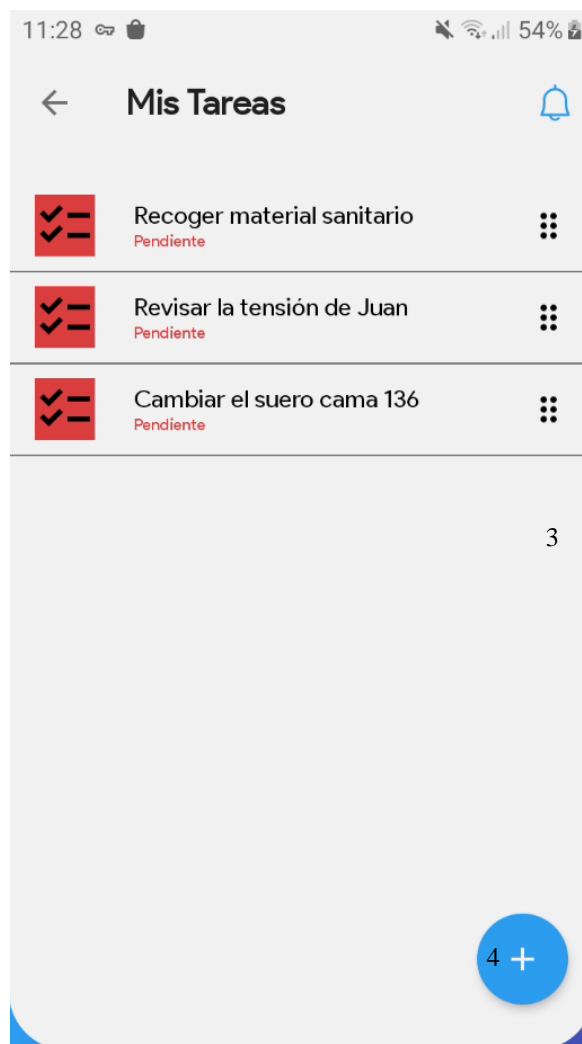
Otro 5 ☐

**Crear Evolución 6**

Respetar la privacidad de los pacientes siguiendo los protocolos.

*Ilustración 50: Opción crear evolución*

1. Opción que al pulsar se seleccionará visita realizada.
2. Opción que al pulsar se seleccionará medicación subministrada.
3. Opción que al pulsar se seleccionará constantes tomadas.
4. Opción que al pulsar se seleccionará temperatura tomada.
5. Opción que al pulsar se seleccionará otro.
6. Al pulsar se creará una nueva evolución.



*Ilustración 51: Pantalla mis tareas*

1. Al mantener pulsado se puede reordenar la lista.
2. Al deslizar de derecha a izquierda una tarea pendiente se completará.
3. Al deslizar de derecha a izquierda una tarea completada se eliminará.
4. Al pulsar mostrará un pop-up de crear nueva tarea.

## DISEÑO DE LA ARQUITECTURA DE LA APLICACIÓN

### TECNOLOGÍAS/HERRAMIENTAS USADAS Y SU DESCRIPCIÓN

#### Aplicación GVP Administración Web:

La aplicación ha sido desarrollada en el lenguaje Java utilizando el IDE IntelliJ IDEA. Para que la ampliación de Java sea web se ha utilizado el framework Spring con la tecnología Spring Boot.

La conexión a la base de datos es mediante el JDBC de SQL Server.

Spring Boot. 

Spring Boot es una tecnología para crear aplicaciones autocontenidas. De este modo, el equipo se puede centrar en el desarrollo a medida. Dejando a un lado la arquitectura. Es decir, se delega en Spring Boot las labores de configuración de dependencias y el despliegue del servicio de aplicaciones. (*Qué es Spring Boot*, 2022).

Para el diseño de la aplicación se ha utilizado la plantilla de Bootstrap Mazer.

#### Api Rest.

Para poder conectar la base de datos SQL Server y la aplicación Android se ha creado una API REST con el lenguaje de programación JavaScript y el entorno Node.js.

Para la autenticación de contraseñas se ha utilizado la función de hash bcrypt.

BCrypt. 

bcrypt es una función de hash de contraseñas diseñada por Niels Provos y David Mazières, basada en el cifrado Blowfish y presentada en USENIX en 1999. (tok.wiki, s. f.)

También han sido utilizados *tokens* de autenticación para una mayor seguridad; estos expiran pasado un tiempo específico.

#### Token Api.

En la autenticación por token, cuando el cliente se ha podido validar como un usuario de la aplicación, recibe una cadena encriptada como respuesta. Esa cadena es el token y sirve para que en los siguientes accesos, el usuario pueda informar al servidor que ya ha pasado por el proceso de autenticación. (Zúñiga, 2022)

## GVP Aplicación móvil:

La aplicación se a desarrollado en el lenguaje Kotlin, se ha utilizado el IDE Android Studio.

Kotlin.



Es un lenguaje de programación pragmático pensado para funcionar con Máquina Virtual de Java (JVM) y Android. Además, puede ser compilado a código fuente de JavaScript.

Se caracteriza por una perfecta combinación de características claramente orientadas a la funcionalidad durante la programación, centrándose en la seguridad, la claridad y la interoperabilidad. (*¿Qué es Kotlin y por qué deberías empezar a aprenderlo ya?*, 2022)

Para la conexión a la Api Rest se ha utilizado la librería Retrofit.

Retrofit es un cliente de servidores REST para Android y Java.

Para poder leer los códigos de barras se ha utilizado la librería Zxing.

También se han utilizado librerías externas para mejorar el front end de la aplicación:

Librerías:

- SmoothBottomBar: Barra de navegación.
- MPAndroidChart: Graficos.
- Lottie: Componentes animados.
- Shimmer: Pantalla de carga.
- DragDrop: Drag and Drop en Lista
- Logger: Un estilo mejorado del log.
- Bumptech: Procesamiento de imágenes.
- FancyToast: Toasts modernos.
- Material: Material Components.



## ARQUITECTURA DE COMPONENTES DE LA APLICACIÓN

### Aplicación GVP Administración Web:

La arquitectura utilizada para esta aplicación ha sido MVC (Model, View, Controller).

- Model: define los datos que debe contener la aplicación y es responsable de gestionar estos datos para enviar al View la información que necesita.
- View: define cómo se deben mostrar los datos de la aplicación. Es una representación gráfica del modelo.
- Controller: Es un intermediario entre Vista y Modelo. Él está a cargo responder a los eventos del usuario.

Al utilizar el framework Spring se detalla como construir una aplicación con esta arquitectura.

### GVP Aplicación móvil:

La arquitectura utilizada para la aplicación móvil ha sido MVVM (Model, View, View, Model).

Se ha decidido utilizar este tipo de arquitectura porque es un patrón de diseño muy potente y fácil de implementar, también permite que nuestra aplicación sea escalable y más fácil de mantener.

- Model: define los datos que debe contener la aplicación y es responsable de gestionar estos datos para enviar al View la información que necesita.
- View: la vista se vincula con variables "observables" y "acciones" expuestas por el ViewModel de forma flexible.
- ViewModel: es el responsable de ajustar el modelo y preparar los datos observables que necesita la vista. También proporciona hooks para que la vista pase eventos al modelo. Sin embargo, el ViewModel no está vinculado a la vista.  
(Ramos, s. f.)

Para un código mas limpio y mejor rendimiento de la aplicación se ha utilizado inyección de dependencias (Hilt).

También en las consultas se ha utilizado corrutinas estas son útiles cuando tienes trabajos que se deben hacer solo si ViewModel está activo.

## **IMPLEMENTACIÓN E IMPLANTACIÓN DEL SISTEMA**

### **IMPLEMENTACIÓN**

Las dos aplicaciones se han subido a repositorios de GitHub mediante este enlace:  
Para poder acceder a los repositorios la url que se ha entregado junto al proyecto.

### **PRUEBAS**

Se han realizado pruebas mientras se desarrollaba los diferentes apartados del proyecto.

Por cada funcionalidad que se desarrollaba se realizaban las pruebas necesarias.

En la aplicación web la mayoría de pruebas han sido rellenar los inputs con datos incorrectos y correctos.

También que en caso de error se notificara correctamente al usuario.

En la aplicación móvil las funcionalidades eran el visionado de datos por lo que se ha comprobado que los datos mostrados eran los correctos.

Al finalizar las aplicaciones se han hecho pruebas con diferente personal del grupo hospitalario Viamed Salud.

Si en el proceso de pruebas saltaba algún error se me notificaba para poder solucionarlo.

## DOCUMENTO DE CIERRE

## DOCUMENTO DE DESPLIEGUE Y CONFIGURACIÓN

Para el despliegue de la aplicación web es necesario tener instalado el JDK de java.

Una vez instalado dentro del directorio iniciaremos un terminal y introducimos el siguiente comando:

```
mvmw spring-boot:run
```

```
C:\Users\pablo\OneDrive\Escritorio\AdministradorWebGvp>mvnw spring-boot:run
[INFO] Scanning for projects...
[INFO]
[INFO] -----< com.crud:GvpWebAdministrator >-----
[INFO] Building GvpWebAdministrator 0.0.1-SNAPSHOT
[INFO] -----[ jar ]-----
[INFO]
[INFO] >>> spring-boot-maven-plugin:2.2.2.RELEASE:run (default-cli) > test-compile @ GvpWebAdministrator >>>
[INFO]
[INFO] --- maven-resources-plugin:3.1.0:resources (default-resources) @ GvpWebAdministrator ---
[INFO] Using 'UTF-8' encoding to copy filtered resources.
[INFO] Copying 1 resource
[INFO] Copying 2985 resources
[INFO]
[INFO] --- maven-compiler-plugin:3.8.1:compile (default-compile) @ GvpWebAdministrator ---
[INFO] Changes detected - recompiling the module!
[INFO] Compiling 15 source files to C:\Users\pablo\OneDrive\Escritorio\AdministradorWebGvp\target\classes
[INFO]
[INFO] --- maven-resources-plugin:3.1.0:testResources (default-testResources) @ GvpWebAdministrator ---
[INFO] Using 'UTF-8' encoding to copy filtered resources.
[INFO] skip non existing resourceDirectory C:\Users\pablo\OneDrive\Escritorio\AdministradorWebGvp\src\test\resources
[INFO]
[INFO] --- maven-compiler-plugin:3.8.1:testCompile (default-testCompile) @ GvpWebAdministrator ---
[INFO] Changes detected - recompiling the module!
[INFO] Compiling 1 source file to C:\Users\pablo\OneDrive\Escritorio\AdministradorWebGvp\target\test-classes
[INFO]
[INFO] <<< spring-boot-maven-plugin:2.2.2.RELEASE:run (default-cli) < test-compile @ GvpWebAdministrator <<<
[INFO]
[INFO] --- spring-boot-maven-plugin:2.2.2.RELEASE:run (default-cli) @ GvpWebAdministrator ---
[INFO] Attaching agents: []
```

[illegible]

El puerto por defecto de Spring boot es el 8080.

Para poder desplegar la Api Rest debemos tener instalado en la maquina el entorno Node Js.

Una vez lo tengamos instalados en un terminal dentro del directorio ejecutaremos estos comandos:

```
npm install
```

```
npm start
```

El primer comando sirve para instalar todos los módulos necesarios.

El segundo comando iniciaremos el api rest mediante el servicio express.

El puerto configurado para la api rest es el 4000.

Para el despliegue de la aplicación móvil solo se requiere la instalación en el dispositivo móvil de la misma.

Se genera un fichero APK mediante el IDE Android Studio, una vez se ha obtenido ese fichero se transfiere al dispositivo móvil y se procede a su instalación.

## MANUAL DE USUARIO

### Aplicación GVP Administración Web

- Sesiones de usuario:
  - ¿Es necesario iniciar sesión?  
Si, para poder utilizar las funciones de la aplicación es necesario iniciar sesión.
- Episodios:
  - Para crear un episodio. ¿Se deben rellenar todos los campos?  
No todos los campos son obligatorios la propia aplicación te avisara si el campo es obligatorio.
  - ¿Se pueden crear varios episodios con el mismo paciente?  
Si, los episodios son completamente individuales unos de otros.
  - ¿Se pueden eliminar los episodios?  
Si, es posible eliminar episodios.
  - ¿Se pueden modificar episodios?  
Si, en caso de equivocación es posible modificar los episodios.
- Usuarios:
  - Para crear un usuario. ¿Se deben rellenar todos los campos?  
Si, es obligatorio rellenar todos los campos para poder crear un usuario.
  - ¿Se puede eliminar los usuarios?  
Si, es posible eliminar usuarios.
  - ¿Se pueden modificar usuarios?  
Si, en caso de equivocación es posible modificar los usuarios.

### GVP Aplicación móvil:

- Sesiones de usuario:
  - ¿Es necesario iniciar sesión?  
Si, para poder utilizar las funciones de la aplicación es necesario iniciar sesión.
- Escanear pulsera:
  - ¿Qué necesito para escanear?  
La primera vez será necesario dar permisos a la cámara para poder escanear la pulsera del paciente.
  - ¿Sin cámara no puedo escanear?  
No, sin cámara no puedes escanear la pulsera, pero podrás buscar el paciente la lista de pacientes.

- Lista de pacientes:
  - ¿Puedo buscar un paciente por cama?  
Si, es posible filtrar los pacientes ya sea por nombre y apellidos o por cama.
- Datos Paciente:
  - ¿Dónde están las alertas del paciente?  
Las alertas tanto si el paciente es alérgico como si es menor de edad se encuentran en la equina superior derecha.
  - ¿Qué datos puedo consultar del paciente?  
Los datos que se pueden consultar son: datos personales, datos clínicos y evoluciones.
  - ¿Cómo creo una evolución?  
Para poder crear una evolución es necesario elegir el tipo de evolución y después pulsar en el botón “Crear Evolución”.
- Mis trabajos:
  - ¿Cómo puedo saber si he realizado un trabajo?  
Es posible buscar tu trabajo mediante la barra de búsqueda, te servirá para filtrar entre tus tareas.
- Mis estadísticas:
  - ¿Las estadísticas se actualizan constantemente?  
Si, las estadísticas son constantemente actualizadas.
- Mis tareas:
  - ¿Cómo se crea una tarea?  
Sera necesario pulsar el botón de añadir que se encuentra en la equina inferior derecha
  - ¿Se pueden crear tareas con varias filas?  
Si, es posible crear la tarea con las filas que te sean necesarias.
  - ¿Cómo se completa una tarea?  
Para completar una tarea es tan sencillo como deslizar de derecha a izquierda esa tarea.
  - ¿Al completar una tarea puedo ponerla en pendiente?  
No, una vez completes una tarea solo podrás eliminarla.
  - ¿Se pueden guardar mis tareas en otro móvil?  
No, las tareas son almacenadas por dispositivo.

## RESULTADOS OBTENIDOS Y CONCLUSIONES

Una vez se han finalizado las aplicaciones se ha obtenido la primera versión de GVP Administración y GVP Móvil.

La versión obtenida realiza las funciones planteadas para este proyecto.

En relación a la interfaz he dedicado mucho tiempo al front end para que la aplicación se parezca mas a una aplicación profesional.

He tenido especial cuidado en la seguridad de las aplicaciones.

He llegado a aprender un lenguaje nuevo y ha sido una experiencia bastante tediosa, pero con el empeño aplicado y la similitud al lenguaje Java he conseguido resolver los problemas que me surgían en esta nueva experiencia con el lenguaje.

La conclusión que he sacado para este proyecto se basa en la dedicación y empeño e ilusión por el proyecto, por otra parte, he tenido que ser lo más productivo posible para poder terminarlo, ya que el tiempo era reducido.

Por otra parte, gracias a Viamed Salud que me ha aportado la idea. Gracias a que he trabajado en más proyectos con ellos tenía las ideas claras y eso a ayudado que se avance con más rapidez en el proyecto.



## SEGUIMIENTO Y DESVIACIÓN RESPECTO A LA PLANIFICACIÓN

En general he seguido la planificación que me había propuesto, he tenido algunos problemas que han hecho que modifique el proyecto a mitad.

Uno de los problemas que mas me han retrasado las tareas planificadas es el programa de administración.

Al principio fue planificado en desarrollarlo en el lenguaje Java y Swing, pero dado al retraso que me estaba ofreciendo tuve que decidir un cambio para poder terminar esa tarea a tiempo.

Otro cambio fue en la creación de la Api Rest, esta fue planificada en desarrollarla en el lenguaje PHP, pero dado a la poca similitud de los leguajes utilizados en el proyecto decidí cambiarla.

Por otro lado, uno de los errores que mas se me ha dado, ha sido el formateo de las fechas ya que no tenia conocimiento en los lenguajes utilizados y tuve que perder más tiempo en estudiar el formateo de fechas.

## **BIBLIOGRAFÍA**

Fazt Code. (2019, 5 enero). *Tu primer REST API usando Node.js, ¿Que es una REST*

*API?* [Vídeo]. YouTube. <https://www.youtube.com/watch?v=bK3AJfs7qNY>

Gerald Versluis. (2022, 16 agosto). *Barcode Scanner in .NET MAUI with*

*ZXing.Net.Mau*i [Vídeo]. YouTube.

[https://www.youtube.com/watch?v=ostgj2xB\\_ok](https://www.youtube.com/watch?v=ostgj2xB_ok)

MoureDev by Brais Moure. (2020, 23 marzo). *KOTLIN: Curso ANDROID desde CERO*

*para PRINCIPIANTES* [Vídeo]. YouTube.

<https://www.youtube.com/watch?v=ebQphhLpJG0>


Programación Android by AristiDevs. (2021a, febrero 18). *[Tutorial] RETROFIT 2 en*

*KOTLIN con CORRUTINAS - Consumir API JSON en Android Studio en*

*Español 2022* [Vídeo]. YouTube. [https://www.youtube.com/watch?v=aQP-](https://www.youtube.com/watch?v=aQP-mUGWh1U)

[mUGWh1U](https://www.youtube.com/watch?v=aQP-mUGWh1U)

Programación Android by AristiDevs. (2021b, julio 29). *Dagger HILT en KOTLIN con*

*MVVM*  - *Tutorial android inyección de dependencias con Dagger Hilt*

[Vídeo]. YouTube. <https://www.youtube.com/watch?v=t6ZuzSu2UHI>

*¿Qué es Kotlin y por qué deberías empezar a aprenderlo ya?* (2022, 5 mayo).

IfgeekthenNTTdata. [https://ifgeekthen.nttdata.com/es/que-es-kotlin-y-por-que-](https://ifgeekthen.nttdata.com/es/que-es-kotlin-y-por-que-deberias-empezar-aprenderlo-ya)

[deberias-empezar-aprenderlo-ya](https://ifgeekthen.nttdata.com/es/que-es-kotlin-y-por-que-deberias-empezar-aprenderlo-ya)

*Qué es Spring Boot.* (2022, 3 junio). Alto Servicios Comunicación.

<https://altoservicios.com/crear-paginas-web/que-es-spring-boot/>

Ramos, J. (s. f.). *Android: ¿Qué es MVC, MVP y MVVM?* Programación y más.

<https://programacionymas.com/blog/android-mvc-mvp-mvvm>

SinFloo. (2019, 30 diciembre). *Como hacer un CRUD en Java Spring Boot con Mysql -*

*Listar* [Vídeo]. YouTube. <https://www.youtube.com/watch?v=KTPH5Cryl24>

The Android Factory. (2022, 14 julio). *Facebook Shimmer == GOAT loading state / Android 2022 / Kotlin* [Vídeo]. YouTube. [https://www.youtube.com/watch?v=-\\_am5AqBBgc](https://www.youtube.com/watch?v=-_am5AqBBgc)

tok.wiki. (s. f.). *bcrypt FondoyDescripción*. <https://hmong.es/wiki/Bcrypt>

Zúñiga, F. G. de. (2022, 6 julio). *API REST y su autenticación basada en token*. Blog de arsys.es. <https://www.arsys.es/blog/autenticacion-api-rest-token>