|  |  |
| --- | --- |
| Universidad Don Bosco - Wikipedia, la enciclopedia libre |  |

**Universidad Don Bosco**

**Dirección de Educación a Distancia**

Ingeniería en Ciencias de la Computación

**Diseño y Programación de Software Multiplataforma (DPS941)**

Fase I de Proyecto de Cátedra

**Presentado por**

Ricardo Arturo De Paz Núñez (DN192246)

Kevin Alexander Fernandez Monge (FM150385)

Julio Danilo Flores Fuentes (FF201999)

Christian Alexander Hernández Funes (HF171856)

Kelvin Vladimir García Juárez (GJ111587)

**Fecha de Entrega** 27 de septiembre de 202

Tabla de contenido

[Introducción 2](#_Toc353329504)

[Objetivos del Proyecto 4](#_Toc1064501814)

[Objetivo General 4](#_Toc1704925302)

[Objetivos Específicos 4](#_Toc2076277870)

[UML - Modelado de Software 5](#_Toc1197917683)

[Diagrama de Casos de Uso 6](#_Toc1786018805)

[Diagrama de Clases 6](#_Toc824443109)

[Diagrama de Entidad-Relación (DER) 7](#_Toc459647148)

[Autenticación de Usuarios 7](#_Toc1604018951)

[Interfaces de la Aplicación Móvil 7](#_Toc2110322791)

[Diseño de la Base de Datos 9](#_Toc628467237)

[Conclusiones 12](#_Toc1401442200)

[Fuentes de Consulta 13](#_Toc1263119187)

# Introducción

El asilo de ancianos Esperanza de Santa Ana se ha enfrentado a desafíos significativos en la gestión de los registros de pacientes, doctores y citas médicas. Actualmente, estos registros se mantienen en hojas de cálculo de Excel separadas, lo que ha dado lugar a la pérdida de información crítica debido a la falta de uniformidad y al olvido ocasional de completar registros. En un esfuerzo por mejorar la eficiencia y la integridad de los datos, se ha propuesto la implementación de una aplicación móvil dedicada para el control de registros.

El objetivo de este proyecto es diseñar una base de datos robusta y segura utilizando herramientas como MySQL, XAMPP y PhpMyAdmin, y desarrollar una aplicación móvil mediante el uso de VSCode, Android Studio y Expo. Esta base de datos almacenará la información de pacientes, doctores, citas médicas y un historial de citas. La aplicación móvil, desarrollada con tecnologías como React Native, permitirá a los usuarios acceder a esta información de manera fácil y eficiente. La aplicación brindará la posibilidad de crear, actualizar y visualizar registros, así como también ofrecerá un historial completo de las citas médicas anteriores

**Descripción del proyecto**

El asilo de anciano Esperanza de Santa Ana actualmente lleva los registros de sus pacientes, doctores e historial de citas en una tabla de Excel, esto ocasiona perdida de información, ya que cada doctor tiene un archivo diferente, que algunas veces olvidan llenar al momento de la cita.

Por este motivo solicitan de una aplicación móvil para el control de los registros:

* Acceso a la aplicación por medio de usuario y contraseña + acceso por Gmail.
* Crear, actualizar, ver, pacientes, doctores y citas.
* Historial de citas.

# Objetivos del Proyecto

## Objetivo General

Diseñar e implementar un sistema de gestión de registros médicos que incluya una base de datos y una aplicación móvil para el acilo de ancianos Esperanza de Santa Ana, con el fin de mejorar la eficiencia en la gestión de pacientes, doctores y citas médicas, y garantizar un acceso seguro a la información.

## Objetivos Específicos

1. Diseñar una base de datos relacional que capture de manera eficiente la información de pacientes, doctores, citas médicas y un historial de citas.
2. Implementar una interfaz de usuario intuitiva y funcional en la aplicación móvil para permitir al personal del acilo crear, actualizar y visualizar registros de manera efectiva.
3. Desarrollar un sistema de autenticación que permita a los usuarios acceder de forma segura a la aplicación mediante usuario y contraseña, así como a través de cuentas de Gmail.
4. Proporcionar un historial completo de citas médicas anteriores para mejorar la toma de decisiones médicas y el seguimiento de la atención a los residentes del acilo.

# UML - Modelado de Software

## Diagrama de Casos de Uso

El primer paso en el modelado UML es crear un Diagrama de Casos de Uso (figura 1). Este diagrama identificará las diversas interacciones entre los actores y el sistema. En el proyecto, los actores podrían ser "Usuarios", "Doctores" y "Administradores", mientras que el sistema sería la aplicación móvil.

* **Actores:**
* **Usuario:** Representa a los residentes del acilo y sus familiares.
* **Doctores:** Profesionales médicos que utilizan la aplicación.
* **Administradores:** Personal interno que gestiona la base de datos y el sistema.
* **Casos de Uso:**
* **Iniciar Sesión:** Usuario, Doctores y Administradores pueden iniciar sesión en la aplicación.
* **Gestionar** Pacientes: Doctores y Administradores pueden crear, ver y actualizar información de pacientes.
* **Gestionar Citas:** Doctores y Administradores pueden programar y administrar citas médicas.
* **Gestionar Doctores:** Administradores pueden agregar y actualizar información de doctores.



***Figura 1.*** *Diagrama de casos de uso. Elaboración propia****.***

## Diagrama de Clases

**El Diagrama de Clases:** Las principales clases podrían incluir (figura 2):

* Usuario
* Doctor
* Paciente
* Cita
* Historial de Citas
* Autenticación
* Base de Datos

Diagrama

Descripción generada automáticamente

***Figura 2.*** *Diagrama de clases. Elaboración propia****.***

## Diagrama de Entidad-Relación (DER)

El diagrama de entidad-relación (DER) (figura 3) muestra la estructura de la base de datos que hemos diseñado. Las principales entidades son Usuarios, Pacientes, Doctores, Citas y Historial de Citas. Cada entidad tiene atributos específicos y relaciones con otras entidades para capturar la información de manera organizada.

Diagrama

Descripción generada automáticamente

***Figura 3.*** *Diagrama Entidad-Relación. Elaboración propia****.***

# Autenticación de Usuarios

La autenticación de usuarios se gestionará tanto por usuario y contraseña como por cuentas de Gmail. Esto garantizará que los usuarios autorizados puedan acceder a la aplicación de manera segura.

# Interfaces de la Aplicación Móvil

A continuación, se muestran algunas pantallas conceptuales de la aplicación móvil en desarrollo:

* **Pantalla de Inicio de Sesión**: Los usuarios podrán iniciar sesión con su nombre de usuario y contraseña o a través de su cuenta de Gmail.



* **Pantalla Principal**: Una vez autenticados, los usuarios accederán a la pantalla principal, desde donde podrán navegar a las secciones de pacientes, doctores y citas.
* **Gestión de Pacientes**: Los usuarios podrán ver la lista de pacientes, agregar nuevos pacientes y actualizar la información existente.
* **Gestión de Doctores**: Esta sección permitirá la administración de registros de doctores, incluyendo detalles como especialidad y contacto.
* **Programación de Citas**: Los usuarios podrán programar nuevas citas médicas y ver las citas existentes en un calendario.
* **Historial de Citas**: En esta pantalla, se mostrará el historial completo de citas médicas anteriores para referencia.

**Solucion al problema:**

La lógica detrás de esta solución implica la creación de una aplicación móvil con un diseño intuitivo que permita a los usuarios gestionar eficazmente los registros de pacientes, doctores y citas, evitando la pérdida de información y mejorando la eficiencia en la administración de un asilo de ancianos.

1. **Diseño de la Interfaz de Inicio de Sesión:**

* La interfaz de inicio de sesión debe permitir a los usuarios acceder a la aplicación mediante un nombre de usuario y contraseña.
* También debe proporcionar la opción de iniciar sesión mediante la cuenta de Gmail para mayor comodidad y seguridad.
* Al iniciar sesión, se debe verificar la autenticidad de las credenciales en la base de datos.

1. **Diseño de la Interfaz del Dashboard (Menú Principal):**
   * Después de iniciar sesión, los usuarios serán redirigidos al panel de control principal (dashboard).
   * El dashboard contendrá accesos rápidos a las funciones principales, como la gestión de pacientes, doctores y citas, así como un enlace al historial de citas y al perfil del usuario.
2. **Diseño de la Interfaz de Gestión de Pacientes:**
   * Esta interfaz permitirá a los usuarios crear, actualizar y ver registros de pacientes.
   * Se debe proporcionar un formulario para ingresar la información del paciente, como nombre, edad, dirección, número de teléfono, historial médico, etc.
   * Los registros de pacientes deben almacenarse en la base de datos MySQL.
3. **Diseño de la Interfaz de Gestión de Doctores:**
   * Similar a la gestión de pacientes, esta interfaz permitirá crear, actualizar y ver registros de doctores.
   * Debe incluir campos para detalles del doctor, como nombre, especialidad, horario de consulta, etc.
   * Los registros de doctores también se almacenarán en la base de datos.
4. **Diseño de la Interfaz de Gestión de Citas:**
   * Esta interfaz permitirá a los usuarios programar citas médicas.
   * Debe incluir un formulario para seleccionar un paciente, un doctor, una fecha y una hora para la cita.
   * Se deben implementar validaciones para evitar conflictos de horarios y citas duplicadas.
5. **Diseño de la Interfaz de Historial de Citas:**
   * En esta interfaz, los usuarios pueden ver un registro histórico de todas las citas pasadas.
   * Deben poder buscar citas por paciente, doctor o fecha.
6. **Diseño de la Interfaz de Perfil del Usuario:**
   * Esta interfaz permite a los usuarios ver y actualizar su perfil.
   * Debe mostrar la información de perfil actual, como nombre, dirección de correo electrónico, etc.
   * Los usuarios deben poder cambiar su contraseña o actualizar otros datos personales.
7. **Creación de la Base de Datos con MySQL:**
   * Debe diseñarse una base de datos MySQL que almacene la información de pacientes, doctores y citas.
   * Se crearán tablas separadas para cada entidad (pacientes, doctores, citas) y se establecerán relaciones entre ellas si es necesario.

# Diseño de la Base de Datos

-- Tabla de Usuarios

CREATE TABLE Usuarios (

IDUsuario INT AUTO\_INCREMENT PRIMARY KEY,

NombreUsuario VARCHAR(255) NOT NULL,

Contrasena VARCHAR(255) NOT NULL,

CorreoElectronico VARCHAR(255) NOT NULL UNIQUE

);

-- Tabla de Pacientes

CREATE TABLE Pacientes (

IDPaciente INT AUTO\_INCREMENT PRIMARY KEY,

NombrePaciente VARCHAR(255) NOT NULL,

FechaNacimiento DATE,

NumeroTelefono VARCHAR(20),

Direccion VARCHAR(255),

OtraInformacion TEXT

);

-- Tabla de Doctores

CREATE TABLE Doctores (

IDDoctor INT AUTO\_INCREMENT PRIMARY KEY,

NombreDoctor VARCHAR(255) NOT NULL,

EspecialidadMedica VARCHAR(255),

NumeroTelefono VARCHAR(20),

DireccionConsultorio VARCHAR(255),

OtraInformacion TEXT

);

-- Tabla de Citas

CREATE TABLE Citas (

IDCita INT AUTO\_INCREMENT PRIMARY KEY,

IDPaciente INT,

IDDoctor INT,

FechaHoraCita DATETIME NOT NULL,

DescripcionCita TEXT,

FOREIGN KEY (IDPaciente) REFERENCES Pacientes(IDPaciente),

FOREIGN KEY (IDDoctor) REFERENCES Doctores(IDDoctor)

);

-- Tabla de Historial de Citas

CREATE TABLE HistorialCitas (

IDHistorial INT AUTO\_INCREMENT PRIMARY KEY,

IDPaciente INT,

IDDoctor INT,

FechaCitaAnterior DATETIME,

DescripcionHistorial TEXT,

FOREIGN KEY (IDPaciente) REFERENCES Pacientes(IDPaciente),

FOREIGN KEY (IDDoctor) REFERENCES Doctores(IDDoctor)

);

-- Tabla de Credenciales Gmail

CREATE TABLE CredencialesGmail (

IDUsuario INT,

CorreoGmail VARCHAR(255) NOT NULL UNIQUE,

FOREIGN KEY (IDUsuario) REFERENCES Usuarios(IDUsuario));

**Presupuesto**

Para elaborar un presupuesto se deben tener en cuenta varios factores, como la duración del proyecto, el número de desarrolladores involucrados, sus tasas por hora, el costo de las herramientas y recursos adicionales, y cualquier otro gasto asociado. A continuación, presentamos un presupuesto estimado:

**Duración del Proyecto:** 3 meses

**Desarrolladores Involucrados:**

* Desarrollador Frontend (React Native)
* Scrum Master / Líder de Proyecto
* Diseñador UX/UI
* Administrador de GitHub

**Tasas por Hora:**

* Desarrollador Frontend: $40 por hora
* Scrum Master / Líder de Proyecto: $45 por hora
* Diseñador UX/UI: $35 por hora

**Administrador de GitHub:**  $30 por hora

**Herramientas y Recursos:**

* Figma (Diseño UX/UI): Gratis
* Trello: Gratis
* Git / GitHub: Gratis
* Licencias Creative Commons: Gratis

**Costos Estimados:**

1. Desarrollador Frontend (3 meses a 40 horas/semana): 3 meses x 4 semanas/mes x 40 horas/semana x $40/hora = $19,200

2. Scrum Master / Líder de Proyecto (3 meses a 40 horas/semana): 3 meses x 4 semanas/mes x 40 horas/semana x $45/hora = $21,600

3. Diseñador UX/UI (3 meses a 40 horas/semana): 3 meses x 4 semanas/mes x 40 horas/semana x $35/hora = $16,800

4. Administrador de GitHub (3 meses a 40 horas/semana): 3 meses x 4 semanas/mes x 40 horas/semana x $30/hora = $14,400

5. Herramientas y Recursos (Figma, Notion/Trello, Git/GitHub, Licencias CC): Gratis

**Costo Total Estimado:** $19,200 + $21,600 + $16,800 + $14,400 = $72,000

# Conclusiones

Durante el proceso de planificación y diseño teórico de la aplicación móvil en React Native, junto con la estructuración de la base de datos y el diseño del diagrama de entidad-relación (DER), se han logrado avances conceptuales significativos:

1. **Diseño de Interfaz**: El diseño teórico de la interfaz de la aplicación muestra un enfoque en la creación de una experiencia de usuario intuitiva y amigable.
2. **Estructura de la Base de Datos**: La estructura teórica de la base de datos y el diagrama DER han sido diseñados con una lógica clara de relaciones entre entidades, proporcionando una base sólida para la gestión de registros médicos.
3. **Enfoque en la Seguridad**: La consideración teórica de la autenticación de usuarios, así como las medidas de seguridad planificadas, demuestran un compromiso con la protección de datos médicos sensibles.

Sin embargo, es importante recordar que este trabajo teórico es solo el primer paso en el proceso de desarrollo.

# Fuentes de Consulta

* **Título de documento:**

Eisenman, B. (2016). *Learning React Native: Building native mobile apps*   *with Javascript*.

**Sitio web:**

<https://pepa.holla.cz/wp-content/uploads/2016/12/Learning-React-Native.pdf>

* **Título de documento:**

JavaScript tutorial. (s. f.).

**Sitio web:**

<https://www.w3schools.com/js/default.asp>

* **Título de documento:**

Documentación. (s. f.). *Firebase*.

**Sitio web:**

<https://firebase.google.com/docs?hl=es-419>

* **Título de documento:**

*MySQL : MySQL Documentation*. (s. f.).

**Sitio web:**

<https://dev.mysql.com/doc/>

* **Título de documento:**

Raman, A. (2023). Firebase to MySQL Integration: 2 easy methods. *Learn | Hevo*.

**Sitio web:**

<https://hevodata.com/learn/firebase-mysql-integration/>

* **Sitio web:** Workana

URL: <https://www.workana.com/es/freelancers/el-salvador/java>