



DOCUMENTO SOBRE REMI

José Antonio Benítez Chacón

DESCRIPCIÓN BREVE

Remi: aplicación para el seguimiento de tratamientos en pacientes

Rodrigo Bernal Gómez, José Fuentes La Borda y Rubén Romero Aponte

2º Grado Superior de Desarrollo de Aplicaciones Multiplataforma

Proyecto Remi

Contenido

Descripción del proyecto elegido	2
Diagrama de pantallas.....	2
.....	4
Diagrama de clases	5
Persistencia empleada	7
Permisos y justificación	8
Capturas de pantallas y emuladores	8
Dificultades encontradas.....	8
Mejoras futuras	9

Proyecto Remi

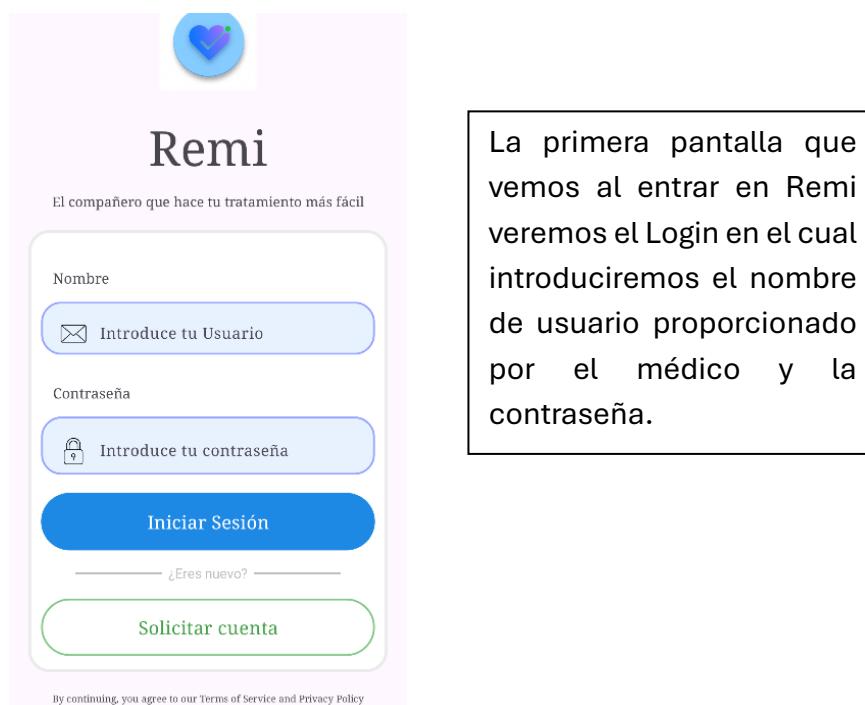
Descripción del proyecto elegido

Nuestro proyecto se llama Remi y es una aplicación para ayudar a los médicos a seguir los tratamientos de sus pacientes. La idea es que los médicos puedan ver qué medicamentos han tomado los pacientes y cuáles no, así como organizar sus citas de manera sencilla.

Además, los pacientes reciben notificaciones y recordatorios sobre los horarios en los que deben tomar cada medicamento, para que no se olviden de nada. La aplicación también permite que los médicos controlen varios pacientes al mismo tiempo y tengan toda la información de su tratamiento en un solo lugar.

En resumen, Remi busca facilitar la gestión de los tratamientos y citas médicas, haciendo más fácil el seguimiento tanto para los médicos como para los pacientes, de manera clara y sencilla.

Diagrama de pantallas



Proyecto Remi



Una vez que iniciemos sesión tendremos dos diferentes tipos de usuarios los pacientes y los médicos. En la vista de médicos la primera pantalla que nos sale será está en la cual podremos ver los usuarios que tiene dicho médico asociados a él. Ahora si clicásemos en el botón de “Entrar al usuario”.



Está es la pantalla que nos sale a la hora de clicar “Entrar al usuario”. En la cual podemos ver los diferentes tratamientos que tiene el usuario actualmente. Tanto el médico como el paciente pueden marcar que el tratamiento está tomado.

Proyecto Remi



Además, en la pestaña de médico se puede añadir el medicamento que tenga que tomar el paciente.



Este es el formulario que nos aparece si quisiera el médico añadir un nuevo tratamiento al paciente.

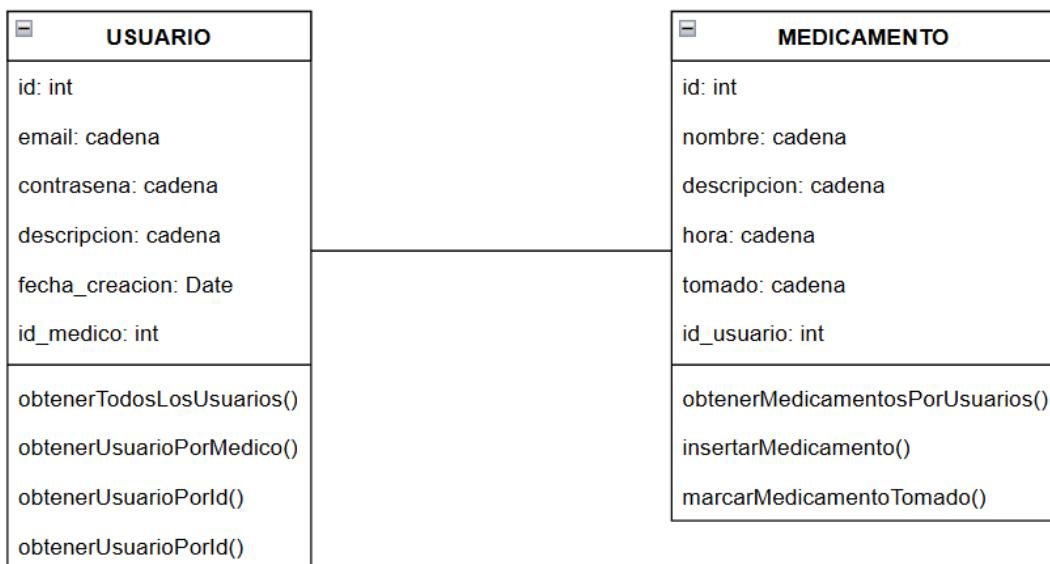
Proyecto Remi



Y esta es la vista siendo paciente en el cuál podemos ver los tratamientos que tenemos actualmente, con diferentes informaciones sobre él. Además de que si le diésemos al botón de “Tomado”, se cambiaría el TextView azul por Tomado (si lo tuviéramos en “No tomado”).

Diagrama de clases

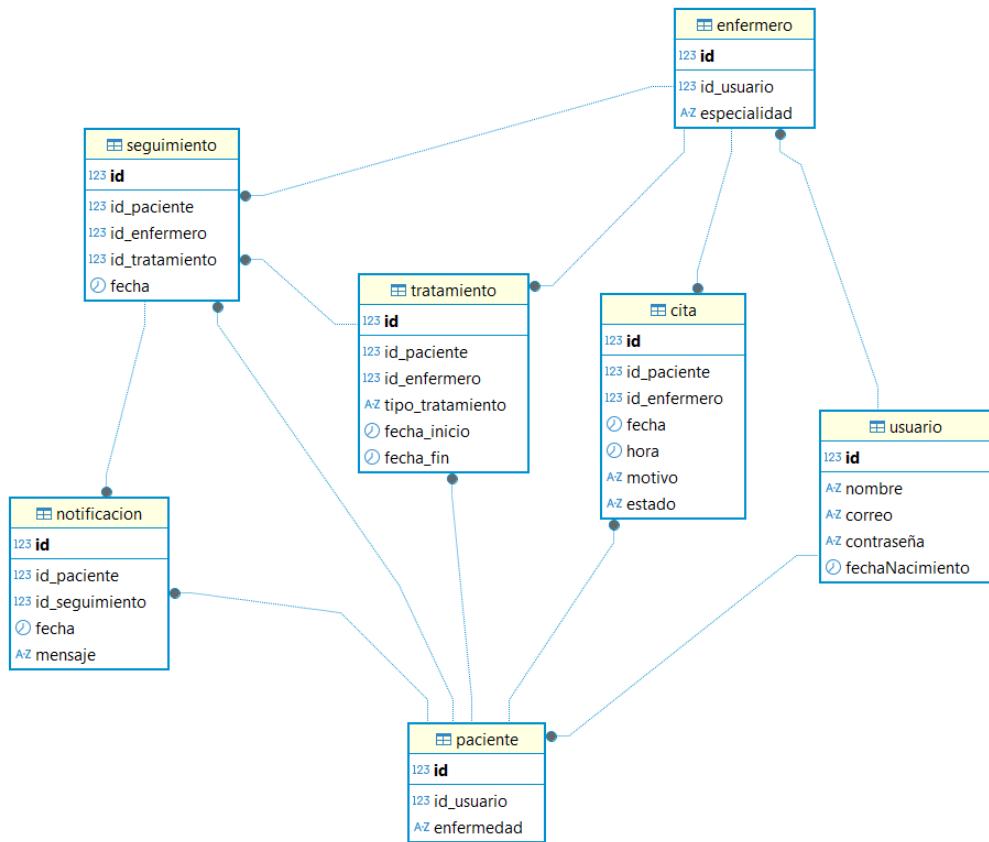
Para este diagrama hemos tenido ciertas dudas ya que tenemos dos bases de datos referidas a las dos diferentes versiones de Remi. La primera en la cual no acortamos absolutamente ninguna función ni nada, y en la cual la base de datos esta al completo. Y la segunda en la cual hemos hecho funcional la información y hemos optado por las dos tablas que enseñamos en el diagrama de clases (usuarios y medicamentos)



Proyecto Remi

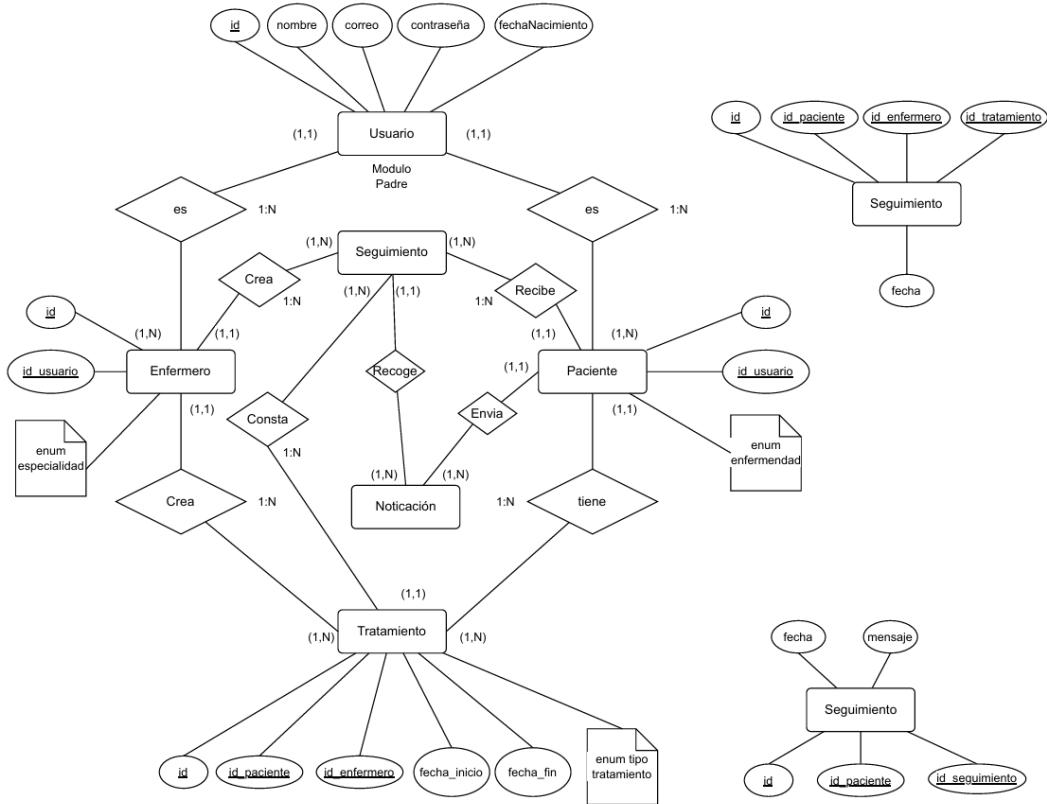
Y la primera versión de la base de datos que es en la que tenemos todas las tablas de la base de datos hechas. Hemos decidido poner en el informe el diseño de base de datos que hicimos para ella en la cual teníamos más tablas que finalmente no integraron en el proyecto por falta de tiempo.

Este es el Diagrama:



Proyecto Remi

Y este es diagrama entidad:relación:



Persistencia empleada

La persistencia de datos se implementa mediante la creación de una base de datos local en el dispositivo móvil, lo que permite almacenar y recuperar información incluso sin conexión a internet. Aunque inicialmente se contempló el uso de MariaDB, por limitaciones de tiempo se optó por SQLite en esta versión de REMI, debido a su ligereza, facilidad de integración y eficiencia en entornos móviles.

La base de datos se gestiona a través de un DBManager, que centraliza todas las operaciones de consulta, inserción, actualización y eliminación de datos. La información se organiza en tablas predefinidas diseñadas específicamente para la aplicación, garantizando un acceso ordenado y seguro a los registros.

Como ejemplo práctico, mediante el DBManager se pueden realizar consultas específicas que generan listas útiles para la aplicación, como “medicamentosPorUsuario()”, que devuelve los medicamentos asociados a un usuario concreto, o “usuariosPorMedico()”, que permite obtener todos los pacientes asignados a un médico determinado. Estas funciones muestran cómo la interacción con la base de datos se abstrae a través del DBManager, facilitando el

Proyecto Remi

manejo de la información y asegurando que todas las operaciones se realicen de forma eficiente y segura.

En conjunto, esta arquitectura permite un manejo sólido y escalable de los datos locales, manteniendo la integridad de la información y sentando las bases para futuras mejoras o migraciones a sistemas de base de datos más complejos.

Permisos y justificación

En Remi nos preocupamos por la seguridad y privacidad de nuestros usuarios, así como por la comodidad y facilidad de uso de la aplicación. Por esta razón, hemos decidido que la aplicación no solicite permisos sensibles como la cámara, la ubicación o el acceso al almacenamiento, que podrían poner en riesgo la información personal de los pacientes.

La única funcionalidad que requiere permiso son las notificaciones, ya que son esenciales para recordar a los usuarios los horarios de sus medicamentos y citas médicas. De esta manera, garantizamos que los usuarios puedan cumplir con sus tratamientos sin tener que preocuparse por la pérdida de datos o accesos no autorizados.

Esta decisión también forma parte de nuestro compromiso de ofrecer una app responsable y confiable, donde los usuarios puedan interactuar de manera segura, y donde la información médica se mantenga protegida en todo momento. Además, al limitar los permisos, hacemos que la instalación y uso de la app sea más rápida y sencilla, evitando solicitar permisos innecesarios que podrían generar desconfianza o rechazo por parte de los usuarios.

Capturas de pantallas y emuladores

Para este apartado hemos decidido que mejor que grabar la pantalla de un móvil y enseñar insitú como funciona. Por ello, está adjuntado en la entrega el video también. Lo hemos probado en un Honor 400.

Dificultades encontradas

Contando aparte de dificultades con errores cometidos, podemos decir varios, pero centrándonos en los graves decidimos decir:

1. Organización de trabajo; con esto nos queremos referir tanto a las tareas que tenemos que hacer cada uno como a las comunicaciones sobre lo hecho entre nosotros.

Proyecto Remi

2. Nombre de los archivos; el error ha sido que cada uno tenía diferentes tareas para hacer y que llamamos archivos iguales o muy diferentes hasta el punto de que hemos tenido que invertir un día en refactorizar nombres.
3. Añadir muchos archivos innecesarios; en este caso hemos introducidos gran cantidad de archivos, los cuales usamos para hacer los diseños estáticos pero que a la hora de hacer la aplicación funcional no sirven para nada y nos desorientaban.
4. Archivos caché de Git; nos referimos al archivo llamado .idea, por el cuál hemos recibido más de 20 errores.
5. Dudas sobre CRUD; muchísimas pruebas y errores a la hora de querer formar la base de datos y querer introducir, sacar, borrar y actualizar datos sobre la base de datos. Hemos tenido que echar muchísimas horas ya que en los apuntes venía por encima visto.

Mejoras futuras

Haciendo hincapié en que hemos tenido 2 semanas para desarrollar la aplicación ya hemos desarrollado 2 versiones de Remi. En la primera hemos desarrollado todas las pantallas al completo, pero sin ser funcional es decir siendo información estática. En la segunda hemos reducido la cantidad de pantallas, pero si hemos hecho la información funcional, en esta hemos añadido la diferencia entre tipos de usuarios tanto pacientes como médicos y además hemos añadido la pantalla de toma de medicamentos en los pacientes. Y en la pantalla de médicos hemos desarrollado una vista de pacientes relacionados con ese medicamento en la cuál puede añadir medicamentos, calendar citas...

Como mejoras futuras podríamos plantear infinitas ya que el campo de la aplicación es infinito. Pero achicando mejoras planteamos ampliar las funcionalidades de la aplicación como por ejemplo introducir en el apartado de médicos que, en el caso de no tener el fármaco registrado en la base de datos, el propio médico pueda escanear el código de barras del medicamento y se añada automáticamente.