

# RecuperARte

Michael Gabriel GIORDANO, German MERKEL, Leonel Alejandro SAN  
MARTIN  
DNI: 41.665.401, DNI: 40.853.806, DNI: 36.528.387  
Miércoles, Grupo N° 15

<sup>1</sup>Universidad Nacional de La Matanza,  
Departamento de Ingeniería e Investigaciones Tecnológicas,  
Florencio Varela 1903 - San Justo, Argentina

**Resumen.** Para ayudar a las personas luego de la pandemia vivida, hemos diseñado una aplicación denominada “RecuperARte”. Dicha aplicación está orientada a recuperar a las personas (ya sean atletas o no) del sedentarismo ocasionado por la misma. Para lograr este objetivo, vamos a controlar el movimiento que realiza una persona durante su actividad física, ya sea caminando o corriendo, y luego de analizar esos datos, daremos pautas que indiquen su progreso y de qué manera mejorarlo.

**Palabras claves:** Pandemia, RecuperARte, Sedentarismo, Progreso.

## 1 Introducción

Queremos ayudar a las personas a mejorar su condición física luego de todo este tiempo en el que nos volvimos más sedentarios. Para eso desarrollamos la aplicación RecuperARte, donde nuestro principal objetivo es que la gente vuelva a entrenar y a ponerse en forma tal y como estaban antes de este acontecimiento.

Luego de observar a personas frustrarse por sus malos intentos de volver a la actividad física, ya sea por fatiga o por no poder rendir como esperaban, se nos ocurrió ayudarlos creando una aplicación la cual va a controlar su actividad, analizar los datos e indicar si el progreso es o no como lo esperado.

Para esto, la app RecuperARte, se basará en dos estadísticas. La primera es la cantidad de metros que realiza por día el usuario. La segunda es con qué frecuencia realiza dicha actividad, esto es en que horarios y días lo hace. Estos parámetros serán captados por dos sensores, uno es el acelerómetro, el cual nos indicará los metros recorridos y la velocidad del usuario. El otro sensor que utilizaremos es el de proximidad, con el cual cuando la aplicación este en uso, se verificara si el usuario lleva su smartphone a hacer actividad física, en caso contrario se enviara una alerta para que lo utilice a la hora de realizar la actividad.

Por otro lado, RecuperARte, cuenta con varias pantallas. Vamos a contar con

un sistema de usuario y contraseña. Por lo tanto, en la primera pantalla de la aplicación, se debe mostrar al usuario la posibilidad de iniciar sesión o de registrarse. En el caso de registrarse, pedimos nombre, apellido, dni email y contraseña, la guardamos en persistencia, y listo. En el caso de iniciar sesión, el proceso se realiza de a dos pasos: Uno es la autenticación, utilizando un factor externo, el cual es un Código QR predeterminado. El otro paso es la autenticación interna, donde tenemos un típico login con email y contraseña. Una vez dentro de la aplicación, se mostrará una pantalla que muestra la actividad del día y el estado de batería del smartphone. También habrá un menú de estadísticas donde se verán todos los datos recopilados de la actividad física del usuario. Para concluir, se comenta que toda la actividad de los sensores será guardada en un determinado servidor.

## 2 Desarrollo

Repositorio GitHub

<https://github.com/SOA-2021/RecuperARte>

Diagrama funcional/navegación de las Activities

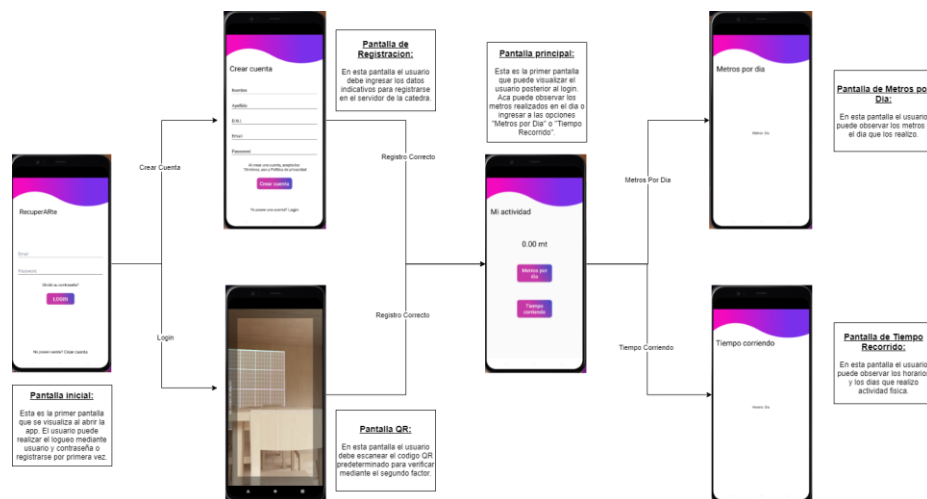


Imagen 1.1

Implementación de la ejecución concurrente del programa y mecanismo de sincronización.

Para realizar la ejecución concurrente de la aplicación, se utilizó un mecanismo de sincronización llamado Synchronized, el cual nos ayuda a

indicar que a ciertas regiones del código solamente podrá acceder un subproceso a la vez. también se utiliza `runOnUiThread` el cual nos permite ejecutar una acción específica desde un thread.

#### Comunicación entre los componentes

Para realizar la comunicación entre los componentes se utilizó `Intent`. Esto fue utilizado para movernos de una activity a otra. `Intent` nos ayuda para invocar a las diferentes activities de nuestra aplicación.

#### Técnica que utilizó para la comunicación con el servidor

Para la comunicación con el servidor se utilizó `URLConnection`. Primero se establece la conexión a una url específica. Para esto es necesario un token con el que se va a realizar la autenticación. Debemos tener una request a enviar. Luego el servicio nos devolverá un tipo de clase de respuesta esperada y con esto retornaremos la respuesta de la API. Si algo en este proceso falla se lanza una `HttpException`.

#### Implementación de la persistencia de los datos en la aplicación.

La persistencia de los datos de la aplicación se realizó en `SQLite`. Cuando se cierra la aplicación, la base de datos es almacenada en un archivo dentro del almacenamiento interno del teléfono que es privado y solamente puede acceder el usuario.

#### Manual de Usuario de la Aplicación Android

La primera pantalla que se nos aparece al iniciar sesión es la pantalla de login, en la cual podemos ingresar directamente el mail y contraseña para iniciar sesión o cambiar a la pantalla de registración. (Imagen 2.1).

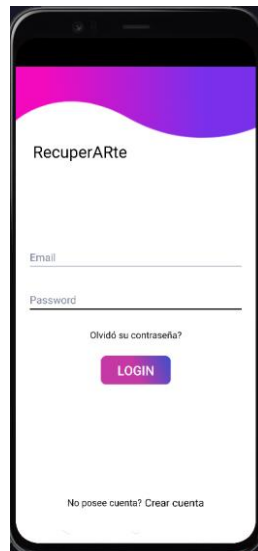


Imagen 2.1

Si es la primera vez en la aplicación, debemos elegir la opción “Crear Cuenta”, esta nos llevara a una nueva pantalla en donde debemos completar con el nombre, apellido, DNI, email y contraseña, luego presionamos “Crear Cuenta” (Imagen 2.2).

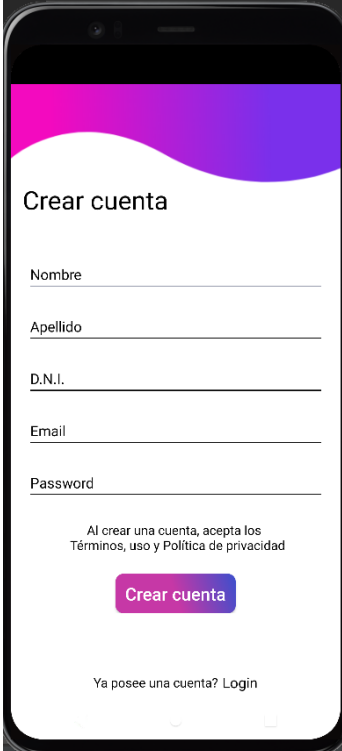


Imagen 2.2

Al presional el botón “Crear Cuenta”, se va a enviar una petición al servidor de la cathedra con los datos a crear. En caso de que el usuario ya se encuentre registrado, nos informara mediante la siguiente leyenda:

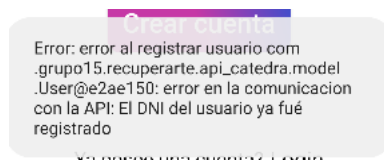


Imagen 2.3

Si la generación es correcta, nos lleva a la pantalla de inicio de la aplicación. (Imagen 2.6)

Una vez ya registrado el usuario, al completar la pantalla de login, nos lleva al segundo factor del inicio de sesión en donde debemos escanear el Código QR que figura dentro del celular de la imagen. Por defecto, se nos va a activar la cámara para poder realizar el escaneo. (Imagen 2.4).



Imagen 2.4

El QR por defecto utilizado en la aplicación para verificar el segundo factor es el siguiente: (Imagen2.5)



Imagen 2.5

Una vez validado el escaneo del código QR, se da por completa la autenticación en dos factores, y se procede a ingresar a la pantalla de inicio del usuario. (Imagen 2.6)



En la pantalla principal podemos observar la cantidad de metros que realizamos por día junto a dos botones que nos indican las funcionalidades de la aplicación. En primer lugar, podemos observar el botón “Metros por Día”. Al presionar el mismo, nos lleva hacia la imagen 2.7.



Imagen 2.7

Si presionamos el botón “Tiempo Corriendo”, nos dirige hacia la pantalla donde podemos observar el horario y el día de las veces que hemos realizado actividad física.

### 3 Conclusiones

Como conclusión podemos comentar que los problemas que surgieron durante el desarrollo, más allá de ir probando los componentes y comprendiendo su uso a medida que íbamos avanzando con el TP, es que se nos complicó para poder obtener los metros que realiza una persona a través del sensor del acelerómetro. Comenzamos probando con las fórmulas básicas que vimos en física calculando los metros a través de las aceleraciones y los segundos, pero el resultado era demasiado erróneo y alejado de la realidad. Posteriormente, probamos con realizar la integral de la aceleración obtenida y dentro de todo se asimilaba bastante. Para finalizar, aplicamos un cálculo matemático a la fórmula y pudimos aproximarnos bastante. Leyendo un poco más en la Internet pudimos observar que para calcular los metros recorridos se recomienda utilizar el GPS.

Además, para poder garantizar la tolerancia a fallos, tuvimos en cuenta por ejemplo que si el usuario no dispone de conexión a internet se lanza un mensaje indicando que no posee conexión y la aplicación no genera la conexión con la API. Por otro lado, en el caso de que los sensores que

necesita la aplicación no estén disponibles, se le avisa al usuario también que los mismos no están disponibles y no se procede a registrar los mismos. Para finalizar, si falla la conexión con la api se le notifica al usuario que reintente en unos segundos y vuelva a realizar la operación.

## 4 Referencias

1. Developers Android: <https://developer.android.com/>
2. Publico-Android: <http://www.sounlam.com.ar/wiki/index.php/PUBLICO:Android>
3. Tutorial de Bases de Datos SQLite en aplicaciones Android: <https://www.develou.com/android-sqlite-bases-de-datos/>
4. Cómo crear un lector de códigos QR en Android con Android Studio: <https://www.desarrollolibre.net/blog/android/como-crear-un-lector-de-codigos-qr-en-android-con-android-studio>
5. Generador QR: <https://www.codigos-qr.com/generador-de-codigos-qr/>