

Aplicación para el monitoreo de la frecuencia cardíaca

Application for monitoring heart rate

Lina F. Parra Rojas

Universidad Distrital Francisco José de Caldas
lfparrar@correo.udistrital.edu.co

Andrés C. Torres M.

Universidad Distrital Francisco José de Caldas
actorrresm@correo.udistrital.edu.co

En este artículo se muestra detalladamente la aplicación para llevar a cabo la medición de frecuencia cardíaca. Este sistema se desarrolla utilizando una pulsera de la compañía Mio Global conocida como Mio Link, para tomar los datos que son enviados mediante comunicación bluetooth a un dispositivo Smartphone con la aplicación previamente instalada, esta se encarga de llevar el manejo de una base de datos con los valores tomados, graficarlos, compararlos con los valores de frecuencia cardíaca normalizada y de enviar una alarma mediante un mensaje texto informando de cualquier anomalía, también envía la base de datos vía correo electrónico si se desea.

Palabras clave: Bluetooth, gestión de la información de salud, ritmo cardíaco, teléfono celular

This paper shows in detail the steps to carry out the realization of a system for monitoring heart rate. This system is developed using a wristband Global Mio company known as Mio Link, to take the data that is sent via Bluetooth communication to a Smartphone device with pre-installed application, it is responsible for keeping managing a database with the values taken, plot them, compared with normalized heart rate values and send an alarm via text message reporting any abnormality, also send the database with a e-mail.

Keywords: Bluetooth, cellular phones, circadian rhythm, health information management

Tipología del artículo: Investigación

Fecha recepción del manuscrito: Mayo 1, 2015

Fecha aceptación del manuscrito: Junio 5, 2015

Investigación financiada por: Universidad Distrital Francisco José de Caldas.

Edición digital: <http://revistas.udistrital.edu.co/ojs/index.php/tekhne/issue/view/755>

Cómo citar: Parra, L., y Torres, A. (2015). *Aplicación para el monitoreo de la frecuencia cardíaca*. Revista Tekhnê, 12(2), 55-64.

Introducción

Este proyecto se plantea como solución a un creciente problema del manejo de enfermedades cardiovasculares (ECV), considerando que estas son la principal causa de muerte en todo el mundo. Cada año mueren más personas por ECV que por cualquier otra causa. Se calcula que en 2012 murieron por esta causa 17,5 millones de personas, lo cual representa un 31 % de todas las muertes registradas en el mundo. De estas más de tres cuartas partes de las defunciones por ECV se producen en los países de ingresos bajos y medios. Se calcula que en 2030 morirán cerca de 23,3 millones de personas por ECV, sobre todo por cardiopatías, y se prevé que siga siendo la principal causa de muerte (OMS, 2015).

Para la completa realización de este proyecto se llevó a cabo procesos de investigación, programación y diseño (Cortes, Leguizamón, y Jacinto, 2015). El dispositivo sensor seleccionado fue el Mio Link (MIO, 2015), el cual logra tomar el valor de la frecuencia cardíaca sin muchas molestias, y su apariencia similar a la de un reloj le otorga un uso sencillo y cómodo. La aplicación se desarrolla para dispositivos tipo Android debido a la facilidad en diseño y la gran cantidad de personas que los utilizan.

Metodología

El proyecto como tal se encuentra dividido en tres bloques básicos los cuales pueden ser observados en la Fig. 1.



Figura 1. Diagrama de bloques principal.

Para la realización de cada bloque el objetivo que se buscaba era utilizar los materiales y/o software más adecuados para que fuera un producto final lo más completo posible, pero al mismo tiempo disminuyendo el costo y la complejidad de su uso, a continuación se procederá a explicar cada uno de los bloques con más profundidad, dando a conocer los diferentes aspectos y procesos que fueron tenidos en cuenta para el desarrollo de estos.

Medición de la frecuencia

El objetivo principal de este bloque es llevar a cabo la medición de los valores de la frecuencia cardíaca, para después poder realizar diferentes procesos con estos, este consta únicamente de la pulsera Mio Link (MIO, 2015), la cual fue seleccionada después de un estudio de las diferentes opciones que se podían obtener en el mercado que realizaran esta operación.

Como resultado se encontró que actualmente existen una gran variedad dependiendo de las necesidades y requerimientos, para este proyecto se buscó tomar el

valor de la frecuencia cardíaca y que este fuera enviado inalámbricamente a un dispositivo móvil, teniendo en cuenta estos criterios se logró descartar todo tipo de dispositivo que no contara con la posibilidad de comunicación inalámbrica, además al no ser necesario visualizar directamente en este los latidos por minuto o BPM por sus siglas en inglés (*beats per minute*), fue posible elegir uno de menor costo, después de este proceso de selección el dispositivo escogido al final fue la pulsera Mio Link citada anteriormente. Las pulseras encontradas con posibilidad de uso se encuentran brevemente explicadas en la tabla 1.

Teniendo en cuenta los datos obtenidos y los requerimientos del proyecto, y anotando que todas los dispositivos nombrados en la tabla 1 cuentan con comunicación vía Bluetooth 4.0, se llegó a la conclusión de que la mejor opción sería la pulsera Mio Link esto debido a que se requiere puntualmente solo la toma de valores de la frecuencia cardíaca, y se busca el uso más cómodo sabiendo que esta no requiere banda pectoral, además de ser la de menor costo que no la requiere.

Es de aclarar que a pesar de haberse escogido este dispositivo, es posible el funcionamiento con otros nombrados en la tabla debido a que se maneja el mismo protocolo de comunicación.

En la Fig. 2 se puede observar el dispositivo elegido. La pulsera Mio Link esta cuenta con un sistema de toma de datos continuo y envió de estos inalámbricamente. A continuación se enuncian más a fondo sus características.



Figura 2. Pulsera Mio Link (MIO, 2015).

Características Mio Link.

- Batería de litio recargable vía USB.
- Resistente al agua (30 m aproximadamente).
- Sensor de ritmo cardíaco óptico.
- Envío de datos vía Bluetooth Smart 4.0 y ANT+.

Como su principal característica predomina la manera en la cual adquiere los valores de los latidos por minuto, esta es sencilla y no presenta ningún inconveniente ni molestia

Tabla 1
 Comparación medidores de pulso cardíaco.

NOMBRE	COSTO	POSITIVOS	NEGATIVOS
POLAR LOOP	95.40 Dólares	Pantalla led, sumergible [3]	Requiere banda pectoral
GARMIN VIVO SMART	74.99 Dólares	Pantalla led, Funciones extras con Smartphone, sumergible, [4]	Requiere Banda Pectoral
WITHINGS PULSE O2 [5]	99.99 Dólares	Pantalla led, Medición de oxígeno en la sangre, Monitoreo del sueño, [5]	No es sumergible
FITBIT CHARGE HR	147,67 Dólares	Pantalla led, Contador pasos y calorías quemadas, Sumergible, [6]	Costo elevado
SAMSUNG GEAR FIT	104.19 Dólares	Pantalla led, Funciones con Smartphone, diferentes sensores [7]	Costo Elevado
MICROSOFT BAND	199 Dólares	Pantalla táctil, monitorea sueño, calorías, estrés, temperatura, luz, aceleración, [8]	No sumergible, el mayor costo
MIO LINK	79 Dólares	Sumergible, sensor de precisión [9]	No cuenta con pantalla

ya que no se debe mantener a presión tomando la medida gracias a como captura estos valores.

La parte más esencial en este caso es la comunicación inalámbrica vía Bluetooth Smart 4.0 (Bluetooth, 2015). Este basa su funcionamiento en el bluetooth de baja energía (BLE por sus siglas en ingles). Para entender cómo funciona la transmisión de datos se debe entender el protocolo GATT (*Generic Attribute Profile*), pero esta parte será explicada a continuación en lo concerniente al segundo bloque principal (Fig. 3). Algo a tener en cuenta es que debido a que se trabaja este protocolo en la pulsera, el dispositivo Android que se planea utilizar debe contar como mínimo con el mismo protocolo Bluetooth lo que quiere decir que debe tener una versión de Android 4.3 o superior para poder establecer la comunicación de manera correcta.

Comunicación

La comunicación se logró por medio de Bluetooth de baja energía, es decir Bluetooth Smart 4.0 (Bluetooth, 2015). La pulsera Mio global link se enlaza con un dispositivo smartphone. Sus características principales son favorables para la aplicación ya que el consumo de energía es reducido, lo que aporta un gran tiempo de utilización sin necesidad de recargar la energía de alguno de los dispositivos. Otras ventajas incluyen soporte multi-conexión, transmisión automática de datos en ciertos intervalos de tiempo y menor tiempo de inicialización.

Para utilizar esta comunicación se debe tener en cuenta conceptos claves, ya que esta comunicación se lleva a



Figura 3. Sistema sensado Mio Continuous Technology (MIO, 2015).

cabo haciendo uso del protocolo GATT/ATT. Este utiliza el menor número de bytes que sea posible. Cada atributo es identificado por un identificador universal llamado (UUID). Este es un formato de 128 bits normalizados para un ID de cadena utilizando para emparejar la información. Los atributos tienen un formato, una característica y un servicio.

- Característica: es un conjunto de descriptores los cuales tienen un valor de datos transferidos entre el cliente y el servicio.

- Servicio: es una acumulación de características que operan juntos para realizar una función específica.
- Descriptor: proporciona información adicional acerca de una característica.
- Servidor: es un dispositivo que recibe comando GATT y solicitan y retornan respuesta.
- Cliente: es un dispositivo que inicia comandos GATT y acepta respuestas.

Estos conceptos deben estar claros ya que el protocolo GATT funciona haciendo uso de ellos para poder transmitir los datos, esto se realiza llevando a cabo los ciertos pasos: escanear el dispositivo, conectarse a este y recibir los datos.

Tratamiento de datos

Buscando cumplir los objetivos planteados para este proyecto se hace uso del software de programación de aplicaciones Android Studio (Google Inc., 2015). Este es el IDE oficial para el desarrollo de aplicación Android, el cual está basado en el Intelligent Java IDE, esto debido a que para llevar a cabo programación para dispositivos Android se debe hacer uso del lenguaje de programación Java. Entre las principales características de este software se encuentran las siguientes:

- Sistema de construcción flexible basado en Gradle.
- Generación de variantes y múltiples archivos Apk.
- Plantillas de código como ayuda para construir aplicaciones.
- Editor de Layout gráfico y simple.

A pesar de ser un software relativamente nuevo la cantidad de usuarios que actualmente lo escogen como su sistema de diseño y programación va en aumento debido a que las características anteriormente mencionadas lo hacen una opción viable y bastante útil. Además de este también se debe mencionar el programa Eclipse, el cual también permite la programación de aplicaciones para dispositivos Android, aunque no es su principal uso, y lleva más tiempo siendo utilizado. Sin embargo, las opciones dedicadas de Android Studio lo vuelven la opción idónea para trabajar, teniendo también en cuenta que la programación es básicamente igual en ambos. En la Fig. 4 se puede observar el entorno de trabajo de Android Studio.

Para poder diseñar una aplicación en Android se deben tener diferentes conceptos claros, referentes a las partes que componen una aplicación, estas son:

- Vista: Una vista es todo aquello que compone la interfaz de usuario, ya sea un botón, un texto, una imagen, una entrada de texto, etc.
- Layout: Estos son un conjunto de vistas para organizar la interfaz, pueden ser relativos, lineales entre otros.
- Actividad: Las actividades son aquellas que se encargan de mostrarnos la interfaz de usuario y poder trabajar en ella, en ellas se encuentran los procesos que podremos realizar especificados en código.

• Servicio: Los servicios son procesos que se llevan a cabo de manera *escondida*, es decir trabajan sin que el usuario se dé cuenta y mientras se realizan otras tareas.

• Intent o Intención: Estos son los que se encargan de realizar acciones en la aplicación tales como lanzar otras actividades, comunicarlás entre sí, entre otras.

• TextView: Es el elemento con el cual se puede visualizar un texto simple y modificarlo.

• EditText: Este es un campo en el cual se puede ingresar texto para trabajarlo internamente.

• Button o botón: cumple la función de cualquier botón normal realizando una acción asociada cuando es pulsado.

• ListView: Permite observar una lista de datos de forma organizada.

Estas son a grandes rasgos las partes de una aplicación, a su vez cada una está compuesta por sus propios componentes.

La serie de procesos que lleva a cabo la aplicación empieza tomando los valores enviados vía Bluetooth, y almacenándolos en una base de datos, la cual también almacenara la hora de la medida, y el número de la posición del dato correspondiente. Esta es diseñada usando el sistema de gestión de base de datos SQLite, el cual es de dominio público, y por lo tanto libre para su uso.

SQLite es un motor de base de datos de SQL que se incorpora directamente con la aplicación. A diferencia de la mayoría de las otras bases de datos SQL, SQLite no tiene un proceso servidor independiente. SQLite lee y escribe directamente a archivos de disco ordinarios, no requiere de dependencias externas, las librerías son compatibles con una gran variedad de lenguajes de programación, soporta diferentes tipos de texto y recibe órdenes de SQL personalizadas. Una base de datos completa de SQL puede contar con varias tablas, índices, triggers y vistas todas contenidas en un único archivo.

Para el manejo de la base de datos se utilizó el *SQLiteOpenHelper* la cual es una clase auxiliar utilizada en Android Studio que facilita la creación y manejo de bases de datos. En este caso, se hace uso de dos bases de datos, una para registrar los valores tomados y la otra para almacenar permanentemente los valores normales.

Los datos tomados y almacenados tienen dos objetivos, además de ser mostrados en la interfaz principal, ser visualizados tanto a manera de lista como gráficamente y activar una alarma en caso de que los valores presenten anomalías. Para el segundo caso los valores son comparados con los valores normales, los cuales se pueden obtener de acuerdo a la siguiente fórmula, dando un valor con el cual se calcularán los límites: $valor = 208 - (0.7 \times Edad)$, dando como resultado la tabla 2. Para esta aplicación esos límites fueron agrupados en tres más generales.

Esta alarma es enviada vía mensaje de texto al número previamente dado en el registro del usuario. Esto se

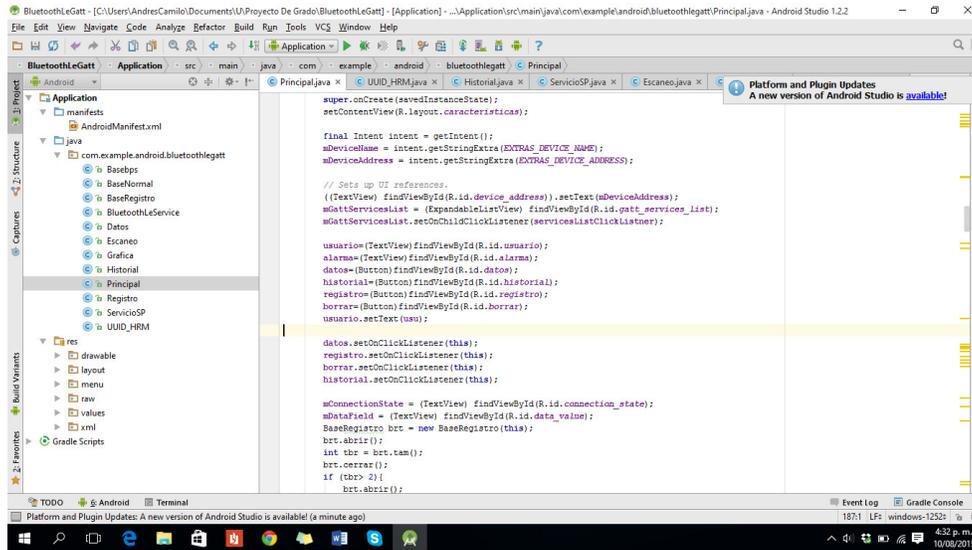


Figura 4. Entorno de desarrollo Android Studio.

Tabla 2
 Valores normalizados.

DESCRIPCIÓN	LÍMITE INFERIOR RC	LÍMITE SUPERIOR RC
EN REPOSO	60 BPM	<50% de RCM
MUY SUAVE	50% de RCM	<60% de RCM
SUAVE	60% de RCM	<70% de RCM
MODERADA	70% de RCM	<80% de RCM
INTENSA	80% de RCM	<90% de RCM
MÁXIMA	90% de RCM	220BPM

realizó gracias a un *Smsmanager* que permite lograr todas las acciones correspondientes al envío de mensajes de texto, el cual es un objeto previamente declarado en la implementación del lenguaje Java utilizado en el software.

La gráfica es una manera más fácil y didáctica de visualizar los datos tomados. Es necesario el uso de una librería externa llamada *Android Plot* la cual es de uso libre y permite declarar, trabajar y manipular diferentes tipos de gráficos dependiendo de la necesidad.

Resultados

En la Fig. 5 se puede observar el diagrama de casos de uso general para la aplicación, en el cual se muestran las diferentes funciones que esta lleva a cabo.

Al iniciar la aplicación se realizan los pasos que se evidencian en el diagrama de secuencia de la Fig. 6. Justo después de que el usuario da inicio a la aplicación, esta se asegura de la disponibilidad y activación del Bluetooth de baja energía, que en el caso de estar desactivado inicia un *intent* que lo solicita. Luego que la activación esta asegurada,

se procede a encontrar todos los dispositivos disponibles al alcance del smartphone usando los métodos ya definidos para esto en el desarrollo de aplicaciones, y los añade a un arreglo que es visible a través de un *listview* en la pantalla principal, Fig. 7. En este se puede seleccionar la pulsera Mio Link o el dispositivo que se desee para realizar la conexión, dando inicio y enviando el UUID de este a la actividad denominada principal desde la cual se manejan las principales características de la aplicación.

Después de haber iniciado la actividad principal relacionada al *layout* del mismo nombre, el cual se puede observar en la Fig. 8, se debe realizar el registro de los datos de usuario. Esto se realiza pulsando el botón con esta instrucción, el cual dirige a la actividad que se encarga de guardar todos los datos ingresados en una base de datos para sus posteriores usos. En esta actividad se encuentran los diferentes campos que son nombre, edad, sexo, peso, estatura, enfermedades, persona encargada, numero de celular y correo electrónico, los cuales se ingresan gracias a un *edit text* (Fig. 9). Posterior a esto, y luego de

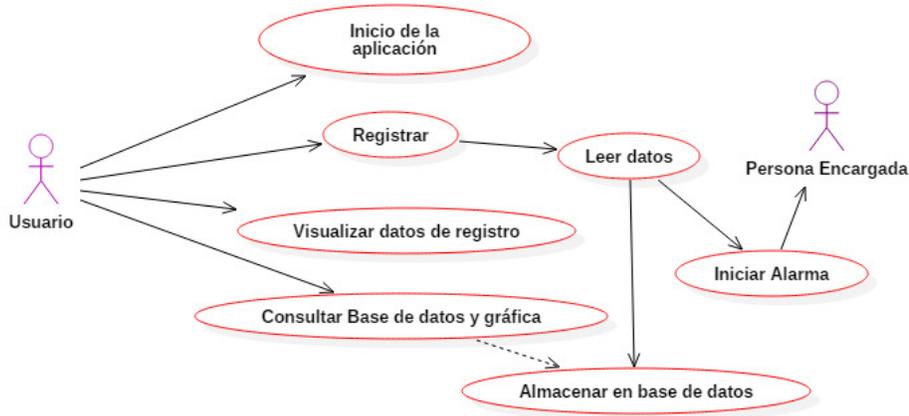


Figura 5. Diagrama de casos de uso.

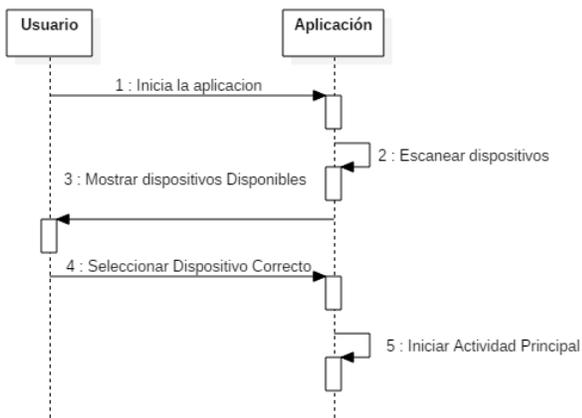


Figura 6. Diagrama de secuencia Inicio.

pulsar el botón de *Ok*, el proceso termina y los datos son almacenados. Estos pasos se pueden observar en la Fig. 10.

Es de aclarar que al ingresar a la aplicación, si los datos fueron dados previamente, estos serán cargados sin necesidad de repetir el proceso de registro. Después de realizar este, el sistema tomará el valor de los BPM de la pulsera. Para esto solicita el valor de acuerdo al UUID característico visualizándolo en el layout principal y almacenándolo en una base de datos. Esta puede ser accedida de diferentes maneras, y guarda el valor de BPM para su posterior uso. Estos dos procesos se pueden ver gráficamente en las Fig. 11 y 12 respectivamente. Para almacenar los valores se crea una base de datos en *sqlite* la cual contiene además del valor tomado la hora y fecha en la cual se registró ese valor. Para esto utilizan métodos que permiten abrir esta para luego ingresar los valores ordenadamente.

En la Fig. 14 se observa el proceso que se realizó para mostrar y graficar los datos. Cuando se consulta el *Layout* de historial, este visualiza los últimos 10 valores tomados, los



Figura 7. Actividad Escaneo.

cuales se ven a través de un *Listview* que indica la hora y la fecha en la cual se tomó el valor. Dentro de este *Layout* se encuentran tres *Button* como se puede evidenciar en la Fig. 13.

El primero *Button* es el que posibilita enviar el histórico completo mediante correo electrónico. Esta base de datos se enviará al correo que ya ha sido previamente registrado. Se puede acceder al *Layout* gráfica, en este se grafican los últimos 20 valores tomados de la base de datos. A esta se le puede dar ampliación (*zoom*), correr a la izquierda o a la derecha. Cuenta también con un *Button* de reiniciar el cual



Figura 8. Layout Principal.



Figura 9. Layout Registro.

permite dejar la gráfica en el estado inicial. Este se muestra en la Fig. 15. El tercero hace la comparación con respecto a los valores normalizados teniendo en la edad registrada con la cual se hace un cálculo, y el último valor tomado (Fig. 16).

En la Fig. 17 se muestra el procedimiento en el cual se solicitan los datos ya registrados, los cuales son visualizados

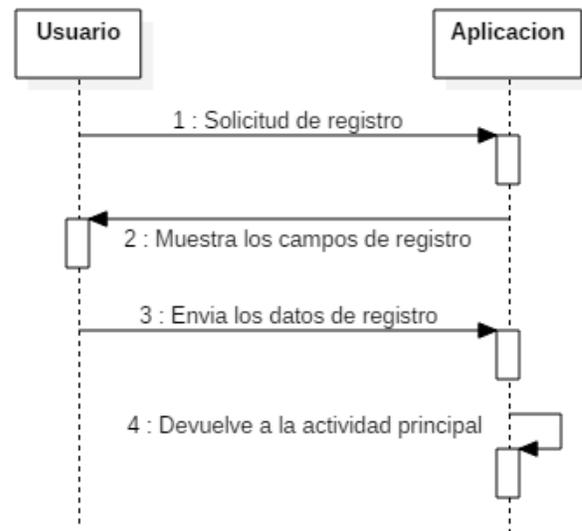


Figura 10. Diagrama Registro.

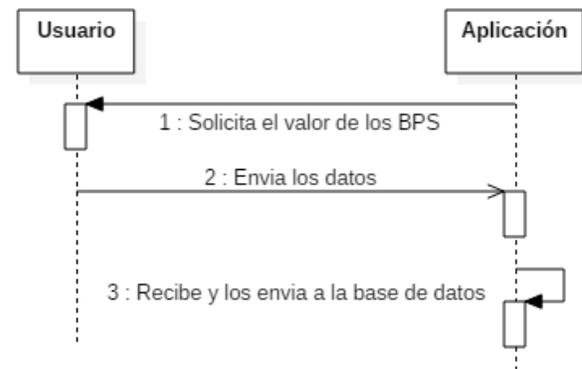


Figura 11. Diagrama lectura de datos.

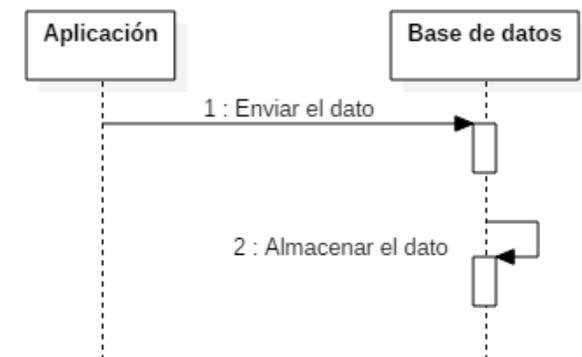


Figura 12. Diagrama almacenamiento de datos.

gracias a *textView* Fig. 18. En la Fig. 19 se explica la manera de ajuste si se sale de los valores normalizados por encima o por debajo de estos; automáticamente la aplicación enviará el



Figura 13. Layout Historial.

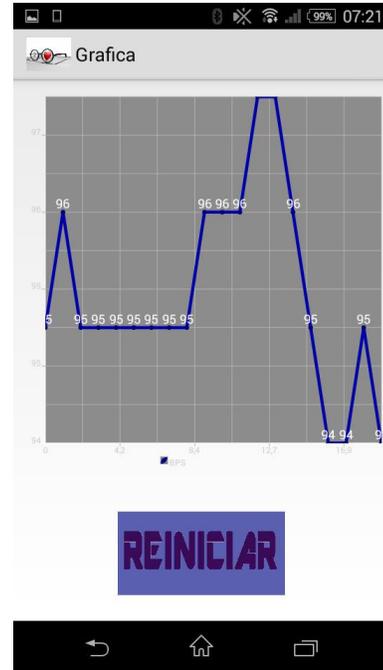


Figura 15. Layout Gráfica.

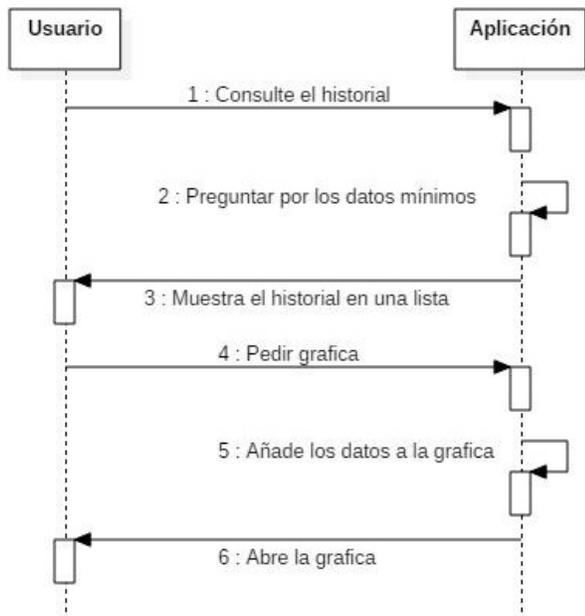


Figura 14. Diagrama Historial y Gráfica.

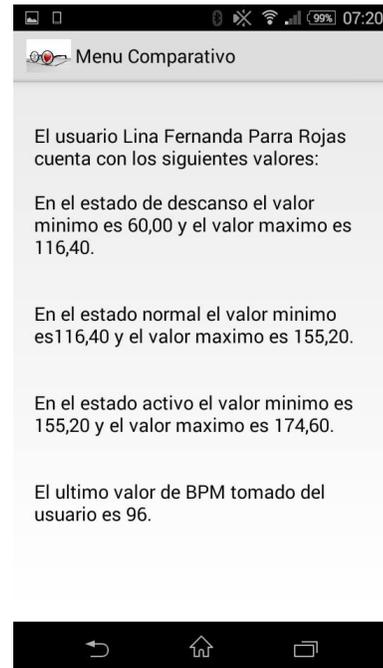


Figura 16. Layout Comparación.

mensaje de texto dando aviso a esto. Este mensaje se enviará al número de teléfono que se solicitó a la hora de registrarse.

La aplicación cuenta actualmente con 11 clases implementadas. A continuación se dará una breve explicación de la función que cumplen:

- Basebps: esta clase se encarga de declarar la base de datos principal en la cual se almacenan los datos tomados, incluye todos los métodos de lectura y escritura necesarios.
- Base Registro: implementa una base de datos que almacenara todos los valores de registro para evitar que sean eliminados al cerrar la aplicación o por agentes externos.

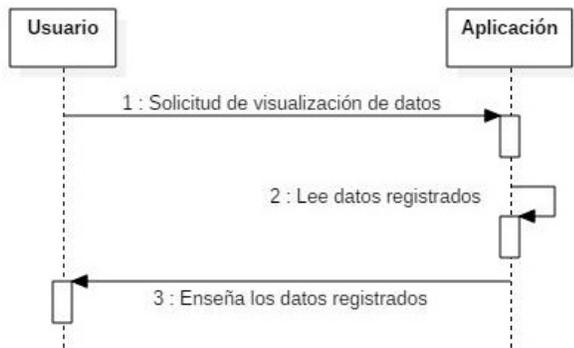


Figura 17. Diagrama de Datos.

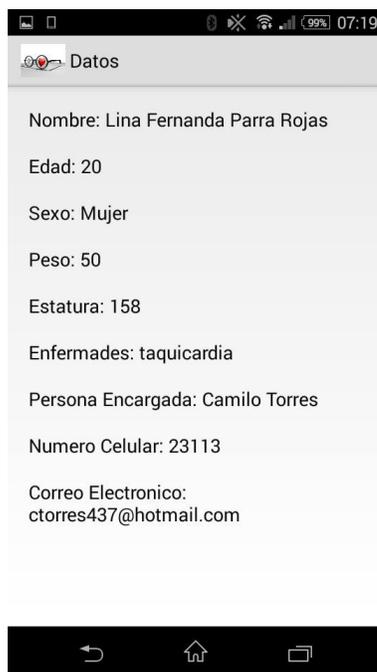


Figura 18. Layout de Datos.

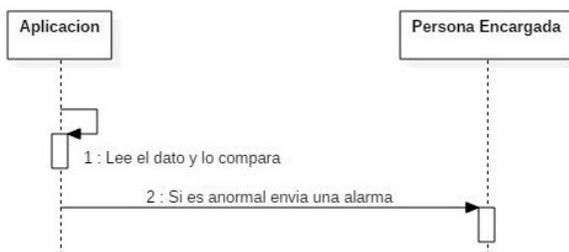


Figura 19. Diagrama de Alarma.

- **BluetoothLeService:** este es un servicio que lleva a cabo los procesos necesarios para tomar los valores enviados por la pulsera, comunicándose con la clase Principal y enviándole los datos de la frecuencia cardiaca solo en el

caso de ser este el dato tomado, tiene métodos de conexión, desconexión, lectura y envío entre otros.

- **Datos:** esta toma los valores ingresados en registro y los visualiza en una lista.
 - **Escaneo:** esta es la clase que se ejecuta en primer lugar, se encarga de escanear los dispositivos cercanos y mostrarlos en una lista para poder seleccionar el adecuado para la aplicación.
 - **Gráfica:** la clase gráfica contiene los comandos necesarios para hacer que la librería *Android Plot* realice la gráfica de los datos almacenados en el archivo generado por *Basebps*.
 - **Historial:** esta clase va asociada al Layout del mismo nombre aquí se manejan todos los procesos relacionados con los botones y la visualización de la base de datos de historial en una lista.
 - **Principal:** esta es la clase que permite visualizar la pantalla principal de la aplicación, aquí se encuentra el acceso a las demás partes, y el manejo de la alarma. Además de poder observar el valor tomado actualmente.
 - **Registro:** aquí se ingresan los datos del paciente que se desea monitorear para poder almacenarlos y que la aplicación realice su trabajo.
 - **UUID_HRM:** esta clase contiene los UUID que se utilizan para reconocer el tipo de dato que envía la pulsera.
 - **Menu_comparativo:** en esta clase se puede observar los rango a los cuales el usuario debe encontrarse dependiendo del estado en el que se encuentre.
- Es de aclarar que sin haberse registrado el usuario no podrá acceder a las demás partes de la aplicación, esto como medida de seguridad y de control interno de la aplicación.

Comparación con el mercado actual

Actualmente en el mercado se encuentran variedad de aplicaciones que permiten tomar el valor de la frecuencia cardíaca, pero estas son básicamente de dos tipos. En uno de los casos el valor que se toma proviene del dedo índice del usuario (Azumio, 2015; Runtastic, 2015) y este debe ser colocado en la cámara del dispositivo móvil para que sea detectado, lo cual resulta complicado además de no permitir un constante monitoreo debido a que se debe estar en permanente contacto. Otra forma de tomarlo es con una banda pectoral, lo cual resulta incómodo y estorboso en ciertas actividades. Aplicaciones que usan esta funcionalidad se encuentran múltiples en el mercado, pero no cumplen con uno de los requerimientos pedidos, que es la libertad de movimiento mientras se es tomada.

El segundo tipo de aplicaciones que predominan es aquel que de igual manera usa algún tipo de pulsera que permite obtener los datos sin mucha molestia, pero su monitoreo no está enfocado en ningún tipo de problema de salud (Endomodo, 2015; Wahoo, 2015). Entonces, solo se encargan de tomar los datos, y un objetivo primordial es que

una persona aparte pueda obtener información del estado del usuario para poder atender cualquier emergencia.

El mercado que se plantea como objetivo no presenta mucha competencia debido a que para lo que se diseña este proyecto no se han desarrollado una inmensa cantidad de aplicaciones comerciales. A pesar de esto, se han desarrollado diferentes proyectos investigativos que pueden realizar esta tarea para realizar un comparativo se buscaron resultados similares en fuentes de datos como la IEEE, y estos suelen utilizar sistemas más complejos para tomar esta medida o simplemente llevar un control sin necesidad de tomarlo, además de todo ser directamente interacción con el usuario.

Conclusiones

A pesar de los avances y la cantidad de dispositivos tecnológicos disponibles hoy en día no es sencillo encontrar fácilmente sistemas que permitan un monitoreo constante de la frecuencia cardíaca por lo cual el proyecto cuenta como una alternativa viable, más teniendo en cuenta que funciona con diferentes dispositivos que tomen valores de bps con protocolo Ble.

Para el desarrollo de la aplicación se encontró que a pesar de la variedad de sistemas disponibles para dispositivos móviles es más común encontrar usuarios de dispositivos Android debido a factores tales como la amplia variedad de aplicaciones y opciones, además de la popularidad que llega a tener.

La comunicación por Bluetooth de baja energía cuenta con varias ventajas que aportan beneficios al proyecto, tal como su bajo consumo y el permitir transmitir diferentes datos al tiempo.

Con la construcción de la base de datos se pudo determinar que las facilidades ofrecidas para desarrollar en Sqlite para Android son amplias gracias a las clases predeterminadas que facilitan este proceso.

El sistema de alarma es una de las partes más novedosas de la aplicación a pesar de encontrarse en otros dispositivos, son extremadamente escasos y no cumplen con las facilidades de este caso.

Una de las principales utilidades de esta aplicación se encuentra en poder comparar los valores tomados con los valores estándar, permitiendo dar una aproximación de la gravedad del problema.

El proyecto logra dar una opción útil para las personas que desean mantener monitoreado su valor de la frecuencia

cardíaca por motivos de salud, debido a que la mayoría de aplicaciones presentes en el mercado se encuentran enfocadas en el ámbito deportivo y no se encuentran opciones viables si el objetivo es hacer uso de funciones enfocadas en el ámbito médico.

Trabajo futuro

Lo que se busca en esta aplicación es que las personas que sufren de enfermedades del corazón o que estén relacionadas a él, lleven una vida lo más normal posible. Las mejoras que se ven a futuro puede llevar un monitoreo más adecuado y constante. Con esto se podría graficar en tiempo real. Otra cosa que se busca es que la base de datos sea envía a una plataforma como Google Fit u otro servicio online para poder estar más al pendiente del estado del usuario. Además, asegurar totalmente la compatibilidad con otro tipo de dispositivos similares.

Referencias

- Azumio. (2015). *Instant heart rate*. online. Descargado de <http://www.azumio.com/s/instantheartrate/index.html>
- Bluetooth. (2015). *Bluetooth technology basics*. online. Descargado de www.bluetooth.com/what-is-bluetooth-technology/bluetooth-technology-basics
- Cortes, C., Leguizamon, J., y Jacinto, E. (2015). Sistema de procesamiento de imágenes vasculares infrarrojas para aplicación en dispositivos biométricos de control de acceso. *Tekhnê*, 12(1), 13-22. (ISSN 1692-8407)
- Endomodo. (2015). *Endomodo*. online. Descargado de <https://www.endomodo.com/>
- Google Inc. (2015). *Android studio the official ide for android*. online. Descargado de <https://developer.android.com/studio/index.html>
- MIO. (2015). *Mio link heart rate wristband*. online. Descargado de www.mioglobal.com/Product.aspx?ProductID=15&DeptID=1
- OMS. (2015). *Enfermedades cardiovasculares*. online. Descargado de www.who.int/mediacentre/factsheets/fs317/es/
- Runtastic. (2015). *Runtastic heart rate*. online. Descargado de <https://www.runtastic.com/es/apps/hearttrate>
- Wahoo. (2015). *Wahoo fitness*. online. Descargado de <http://www.wahoofitness.com/>

