

APP MOVIL

ESTADO INICIAL

La aplicación móvil tendrá que modificarse para adecuarse a las nuevas exigencias del proyecto, como base de inicio se hará uso de la versión original del proyecto denominado “DIMApp” el cual se encuentra desarrollado en el entorno de desarrollo Android Studio con Java como lenguaje de programación se espera que al proyecto debido al tiempo fuera de mantenimiento requiera de actualizaciones de versión, estabilidad, funcionamiento y afines.

El estado inicial de la aplicación es el siguiente:

- Gradle Versión: 3.6.1
 - Fecha de lanzamiento: Feb 28, 2020
 - Vulnerabilidades del sistema: requiere actualización
 - Documentación:
<https://mvnrepository.com/artifact/com.android.tools.build/gradle/3.6.1>
- Java Versión: 1.8

El proyecto requiere actualización de gradle por vulnerabilidades de la versión actual a la última versión a la fecha y continuar actualizando según las versiones sean lanzadas oficialmente y se consideren estables.

- Versión 8.0.0: <https://mvnrepository.com/artifact/com.android.tools.build/gradle/8.0.0>
 - Fecha de lanzamiento: Apr 13, 2023
- Versión 8.0.1: <https://mvnrepository.com/artifact/com.android.tools.build/gradle/8.0.1>
 - Fecha de lanzamiento: May 01, 2023
- Versión 8.0.2: <https://mvnrepository.com/artifact/com.android.tools.build/gradle/8.0.2>
 - Fecha de lanzamiento: May 24, 2023

PROBLEMAS ENCONTRADOS

1. El nombre original del proyecto “com.example.dimapp” no es un nombre válido para publicar la aplicación en la playstore en un futuro por lo que es necesario actualizar a un nombre válido.
2. Se debe cambiar el min SDK de la aplicación actual API 23 a una versión que no se encuentre descontinuada
3. Actualizar el Gradle a una versión actual estable que permita compilar.

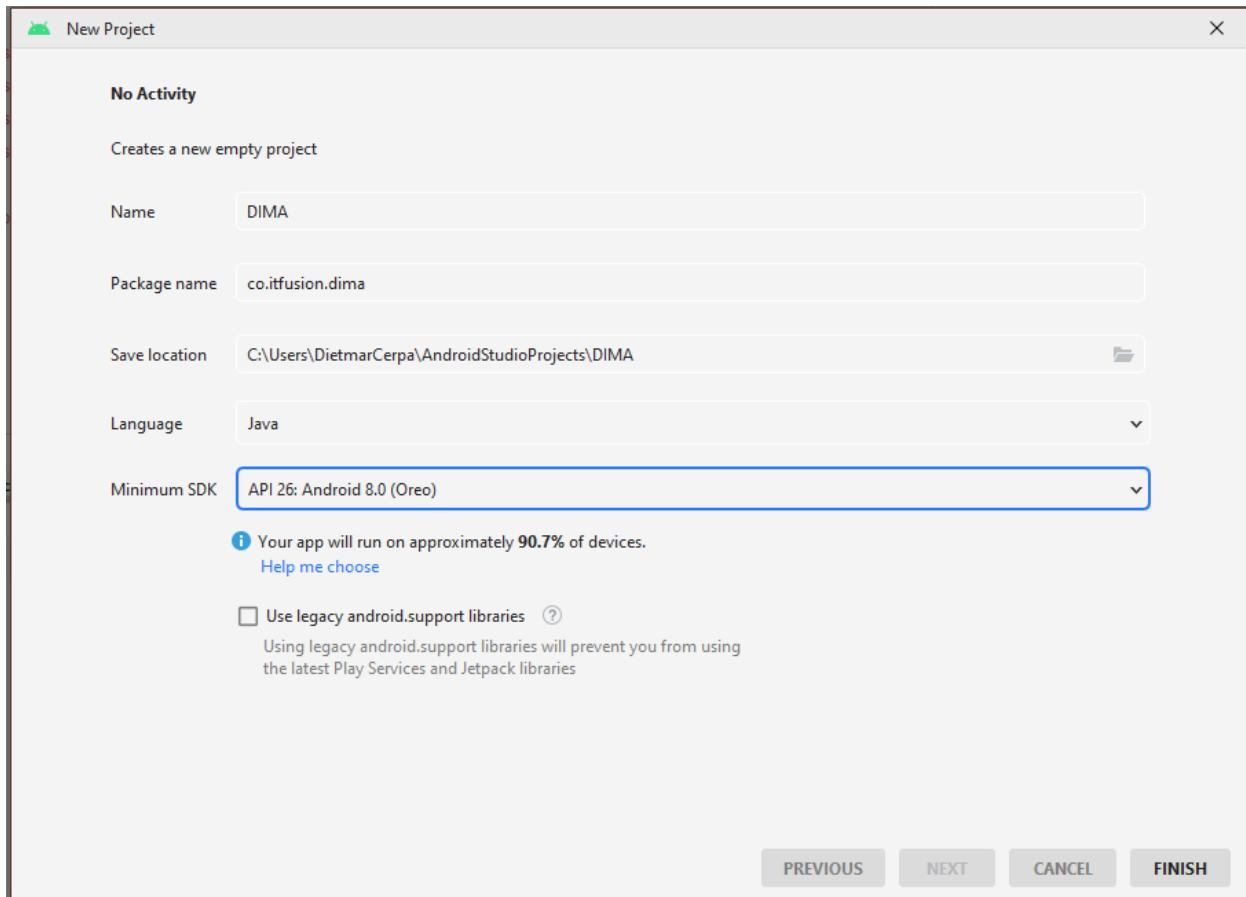
SOLUCIONES APLICADAS

1. Cambiar el nombre de la solución a un nombre válido: “No permite ya que requiere actualizar Gradle”
2. Actualizar la versión de Gradle de 3.6.1 a 8.0.2
 1. ESTADO: FALLIDO

2. REPORTES GENERADOS:

1. https://drive.google.com/file/d/11XuThzk0FWT2RaWodR_USgiClg73v-Mn/view?usp=drive_link
2. https://drive.google.com/file/d/11N1qOgj9Zn6twaXC_4Tps-5BCDAQF_Iu/view?usp=drive_link
3. https://drive.google.com/file/d/11OEpfAaNBFhMJpOiZxykJtG1n5Run6n/view?usp=drive_link

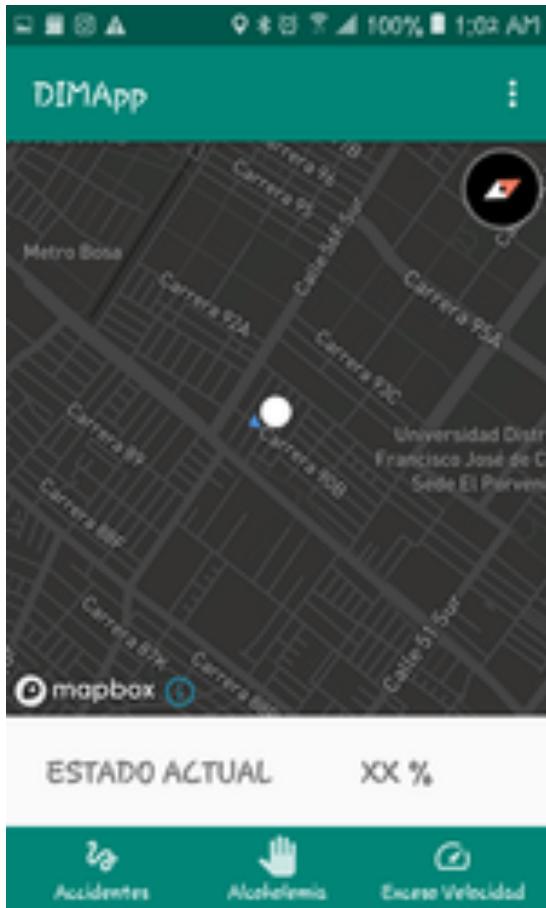
Al resolver los problemas listados en los documentos anteriormente adjuntos el proyecto perdió funcionalidades y lógicas anteriormente desarrolladas debido a que el código se encuentra descontinuado (algunas librerías), se tomó la decisión de crear un nuevo proyecto, que apunte a una API válida, con un nombre que permita su publicación y con el Gradle actualizado, migrando solo la lógica que se llegase a necesitar dentro del proyecto original, la configuración del proyecto ahora es:



- NOMBRE: “DIMAX”
- Nombre del paquete: “co.itfusion.dima”
- Lenguaje: JAVA
- Min SDK: API 26 Android 8.

DESARROLLO

El estilo grafico anterior de la aplicación no contaba con Material Design V2 por lo que su diseño no era el más adecuado, a continuación se muestra como la aplicación era anteriormente.



El proyecto actual permite hacer uso de Material Design v2 mejorando su estilo grafico y su rendimiento, adicionalmente permite el uso de la nueva tecnologia Bluetooth LE (Low Energy), a continuación se adjuntan imagenes respectivas al desarrollo actual de la app.

12:29 29.3kB/s



12:29



>Loading

12:29 70.2kB/s



12:30

Acá se mostrarán los accidentes detectados

El diseño minimalista y responsive permitirá a la app mantenerse actualizada graficamente de manera sencilla y confiable, no obstante se solicita eliminar el mapa ya que para efectos prácticos sera poco útil, al contrario se solicita agregar una menu estilo dashboard que muestre un resumen de la información detectada al usuario a través de la diadema.

BLUETOOTH LOW ENERGY

Bluetooth Low Energy o Bluetooth LE es una tecnología de comunicación para uso personal inalámbrica, la cual es ampliamente usada en aplicaciones de salud, seguridad, bienestar, hogar, entre otras.

Para mayor información referirse al siguiente enlace: [WIKI: Bluetooth BLE](#)

BLE Y ANDROID

La aplicación se encuentra desarrollada como anteriormente se detalló en Android Studio y se considera una APP nativa, lo que significa que se tiene acceso al hardware del smartphone, lo cual es importante para el uso del Bluetooth como medio de comunicación Serial.

La comunicación serial mediante BLE será implementada por efectos de bajo consumo energético (eléctrico) lo que es beneficioso no solo para el smartphone si no que proporciona las mismas ventajas eléctricas para el dispositivo o diadema en construcción.

Cabe destacar que la mayoría de dispositivos móviles o smartphones hacen uso de la tecnología BLE sin embargo mencionada tecnología funciona con retrocompatibilidad lo que permite que funcione con versiones anteriores de Bluetooth Classic y Bluetooth LE, cada dispositivo posee información importante que puede ser accedida mediante código en JAVA o KOTLIN pudiendo determinar la lógica o metodología de conexión con estos dispositivos desde el smartphone.

La versión anterior del prototipo denominado DIMA V1 contaba con un módulo Bluetooth Classic V2.0 para mayor información puede revisar el siguiente documento [Bluetooth HC05](#) posteriormente para efectos del desarrollo de la versión actual de la diadema se consideró el uso de un módulo Bluetooth BLE que posee mejores prestaciones eléctricas, puede revisar el siguiente documento para más detalles [Bluetooth BLE HM10](#).

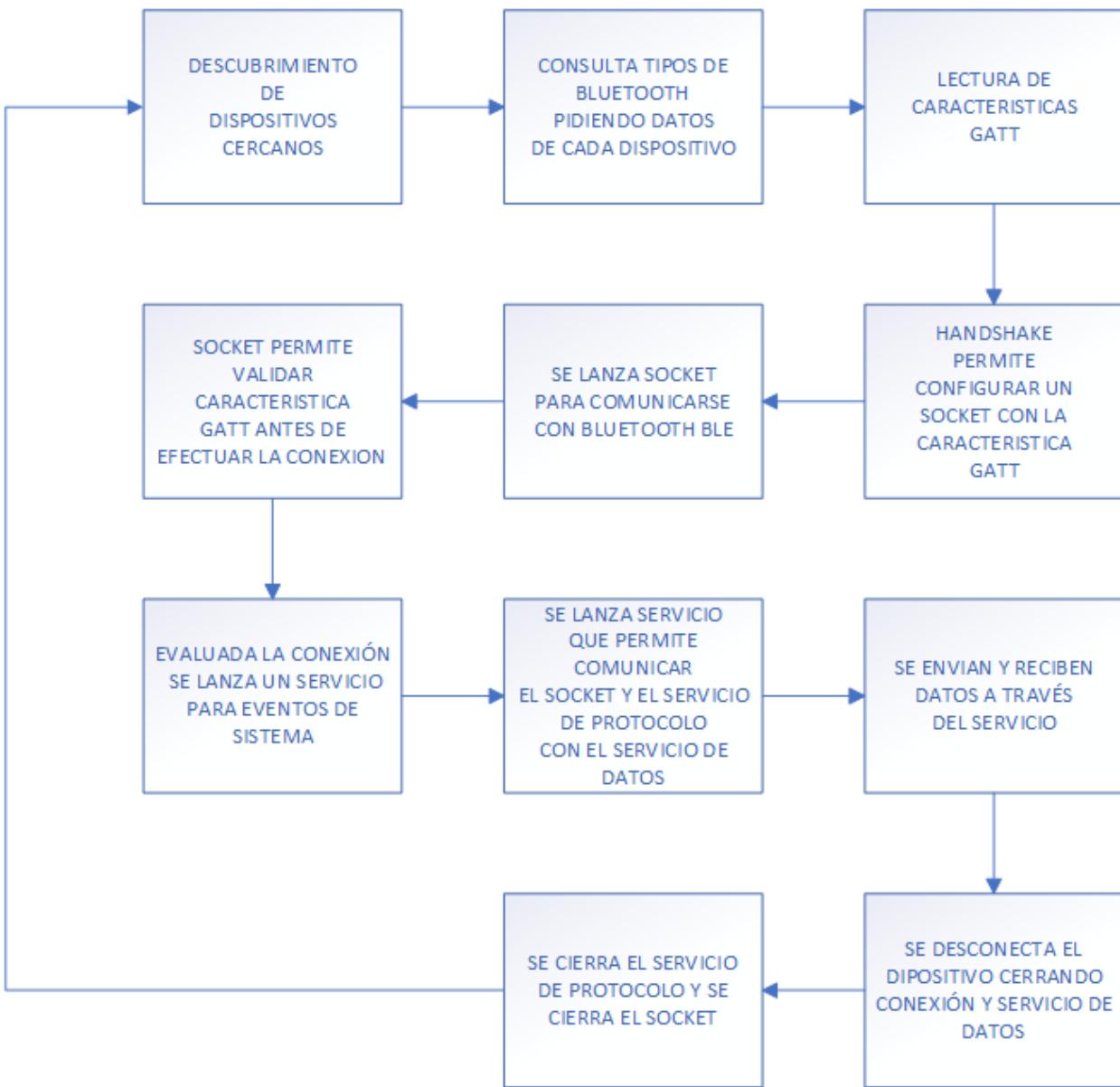
Las mejoras derivadas de este cambio requieren un código que funciona con un método de conexión diferente al que se tenía con el módulo Bluetooth Clásico.

- Esquema de conexión Bluetooth Clásico



El proceso de conexión es relativamente sencillo permitiendo que se implemente abiertamente en diferentes aplicaciones, sin embargo el uso de UUID genérico supone un riesgo de seguridad que se encuentra implícito en versiones anteriores de Bluetooth.

- Esquema de conexión Bluetooth BLE



A diferencia de su antecesor el código que controla el protocolo de comunicación es mucho más complejo y agrega capas de seguridad que no se consideraban anteriormente en versiones anteriores, los servicios que corren en segundo plano permiten verificar la comunicación y la compatibilidad del dispositivo al que se quiere conectar.

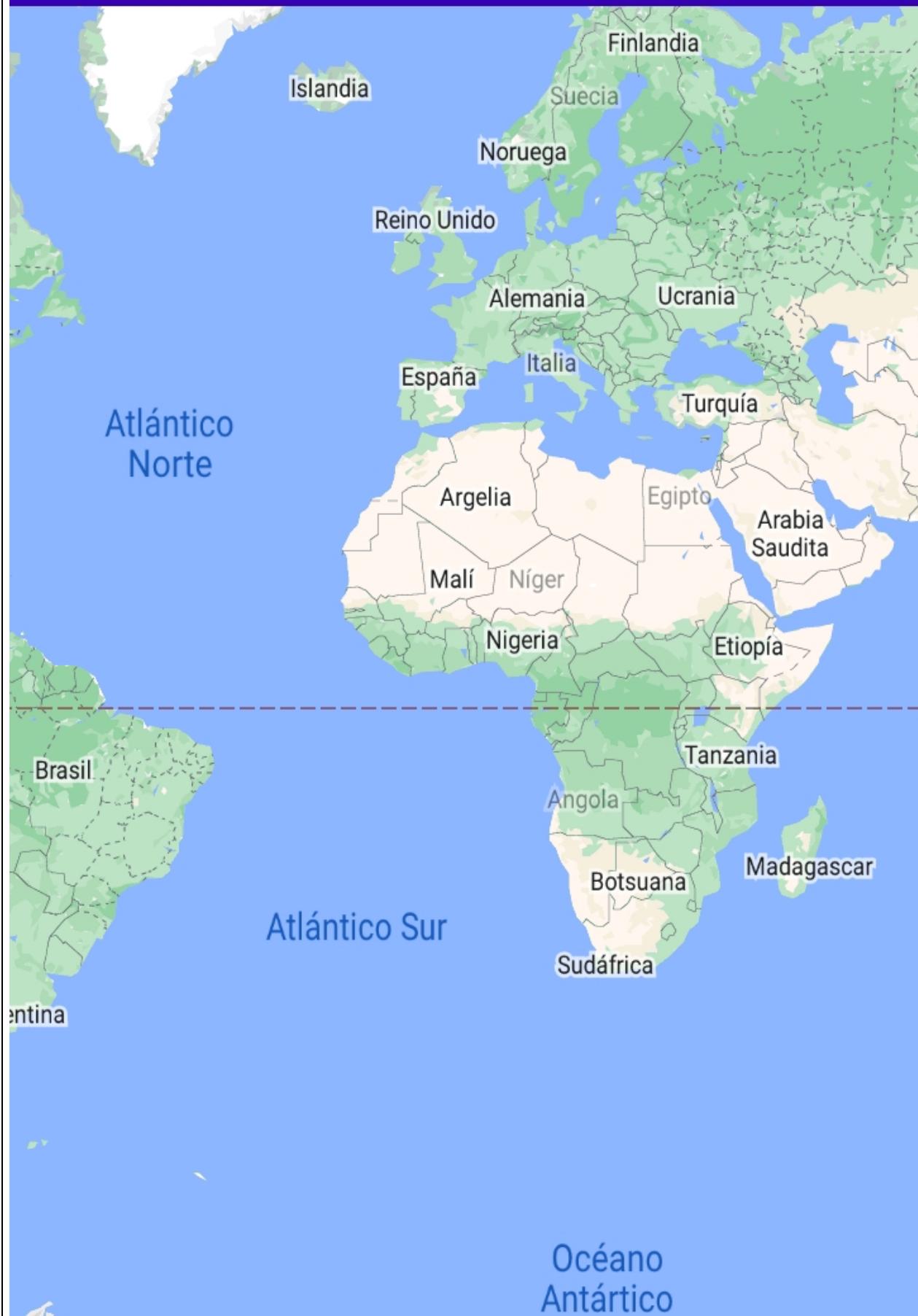
CAMBIOS EN MENUS SOLICITADOS

En reuniones con el equipo de trabajo se determinó que para efectos de usabilidad y uso de la app el mapa ubicado en el menú principal será removido y ahora se mostrará en su lugar un dashboard (resumen) de la información del dispositivo y afines.

12:29 24.4kB/s



17:58



CONEXIÓN BLUETOOTH

Para que la conexión bluetooth se pueda realizar, es necesario esario crear un menu adicional donde se pueda buscar, listar, escoger y conectar con el dispositivo bluetooth correspondiente, aunado a esto es necesario agregar permisos en el manifesto de android y establecer enlace directo sobre el servicio que se va a utilizar para conectar.

- AndroidManifest - Permisos

```
!-- NOTE: Following the android documentation stands that, for API 30+ is not required BLUETOOTH and BLUETOOTH_ADMIN permissions, regardless of this, removing the attribute "maxSdk" it allows to connect with BT Classic Devices.-->

<uses-permission
    android:name="android.permission.BLUETOOTH" />
<uses-permission
    android:name="android.permission.BLUETOOTH_ADMIN" />
<!-- endregion -->
<!-- region Bluetooth Permissions for New devices -->
<uses-permission android:name="android.permission.BLUETOOTH_SCAN" />
<uses-permission android:name="android.permission.BLUETOOTH_CONNECT" />
<uses-permission android:name="android.permission.BLUETOOTH_ADVERTISE" /> <!-- endregion -->
<!-- region Location Permissions -->
<uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION" />
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" /> <!-- endregion -->
```

- AndroidManifest - Servicio

```
<service
    android:name="co.itfusion.services.SerialServiceDiadem"
    android:enabled="true"/>
```

Lo anteriormente mostrado son requerimientos minimos para solicitar permisos y lanzar el servicio de conexion bluetooth, no obstante, se debe solicitar de manera dinamica los permisos al usuario siempre que se vaya a realizar un intento de conexión,as condiciones para que el bluetooth BLE funcione requiere no solo los permisos de Bluetooth si no ademas los permisos de Ubicación, se muestra entonces a continuación el menú agregado para la busqueda de dispositivos bluetooth y el proceso de busqueda de dispositivos.

19:14

Pro

19:14 21 ● ● -

00:00

Protégez-moi

DISPOSITIVOS BLUETOOTH
ENCONTRADOS



Protégeme

DISPOSITÍVOS BLUETOOTH ENCONTRADOS

Para continuar, activa la ubicación del dispositivo, que usa el servicio de ubicación de Google

Tu dispositivo necesitará lo siguiente:



Usar GPS, sensores y redes Wi-Fi y móviles



Usar el servicio de ubicación de Google. Como parte de este servicio, es posible que Google recopile datos anónimos periódicamente y los use para mejorar la precisión de la ubicación y los servicios basados en la ubicación.

Para obtener más información, ve a la configuración de la ubicación.

Una vez los dispositivos se encuentran listados solo es necesario conectar con el dispositivo en cuestión y si el mismo cumple con las características necesarias de conexión entonces se procede a lanzar el servicio, abrir el socket de conexión y permitir la comunicación de dos vías (envío y recepción)

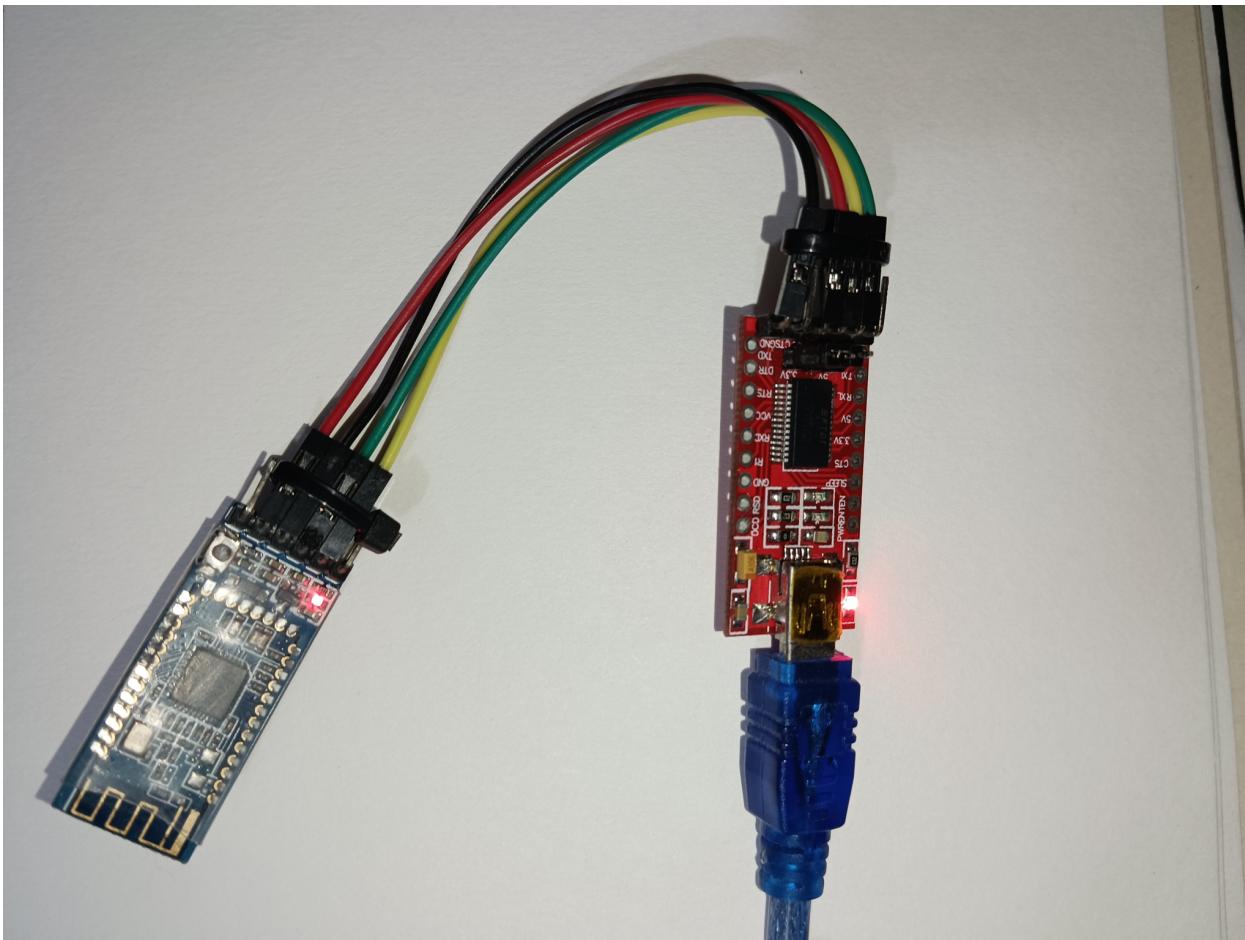


PRUEBAS DE CONEXIÓN

Para las pruebas de conexión se hará uso de:

- Bluetooth BLE HM-10: Dispositivo en modulo que se encargara de enviar los datos a través del protocolo Bluetooth y que es el modelo escogido como primera opcion para envío y recepción de datos para la diadema.
- Modulo FT232 a USB: Es un dispositivo que permite convertir los datos del modulo Bluetooth HM-10 de USART a USB Serial para permitir la lectura mediante la apertura de un puerto serial desde un computador.
- XCTU: Programa que posee un modulo de Consola Serial muy útil para permitir la apertura del puerto serial, recepcion de datos y envio.
- Android Studio Logcat: Herramienta incluida internamente en el entorno de desarrollo que permite obtener datos directamente de la aplicacion conectada mediante un log de informacion ordenada.
- Redmi 7: Dispositivo Android API 29 (Android 10) para instalación y pruebas de la app.
- DIMA (App): Aplicación desarrollada.

A continuación imagenes referentes a la lista anterior:



XCTU Working Modes Tools Help

Radio Modules

Click on Add devices or Discover devices to add radio modules to the list.

Serial Console

This tool allows you to communicate with your XBee devices without having to add them to the list of radio modules.

Serial Console

COM1 - 9600/8/N/1/N

Console log

```
$data_testing_for_d1
ma#
24 64 61 74 61 5F 74 65 73 74 69 6E 67 5F 66 6F 72 5F 64 69
6D 61 23
```

Send packets

Name	Data
packet_0	\$data_testing_for_d1#

Send a single packet

Send sequence

Transmit interval (ms): 500

Repeat times: 1

Loop infinitely

Start sequence

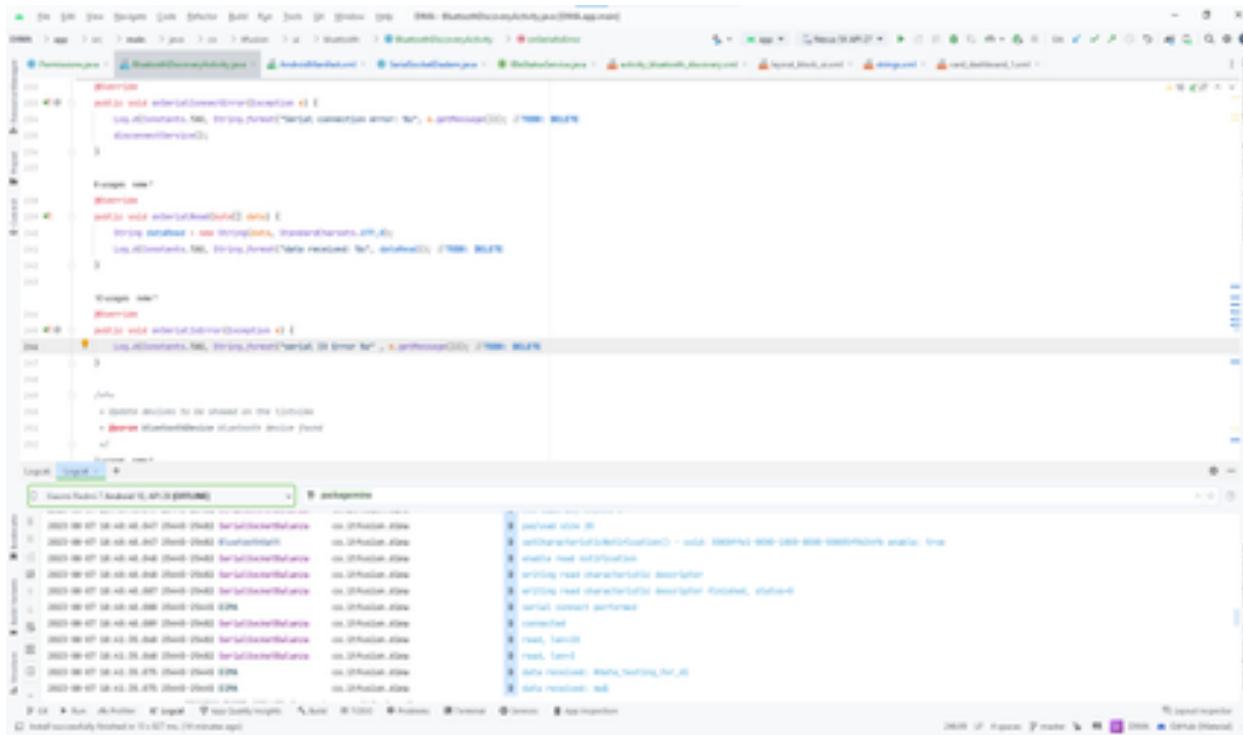
Close

Change between Configuration, Consoles and Network working modes to display their functionality in the working area.

Updates Available

Updates are available for your software. Click to review and install updates. Set up Reminder options

Checking for Radio Firmware updates: (4%)



CONCLUSIONES

El proceso de conexión con el dispositivo bluetooth seleccionado para el diseño inicial de la nueva versión de la diadema haciendo uso del bluetooth de bajo consumo resultó en pruebas exitosas mostradas en las imágenes anteriormente expuestas.

La app está diseñada para conectarse con dispositivos bluetooth que cuenten con las características de envío y recepción de datos mediante un perfil Gatt determinado, la app estará recibiendo datos continuamente mediante el socket en segundo plano siempre que la app se encuentre abierta.

Es importante destacar que si la app se cierra, no es posible que la app se cierra, no es posible obtener datos de la conexión, la misma debe estar en segundo plano como mínimo para poder mantener la comunicación con la diadema.

MEJORAS A BLUETOOTH DUAL MODELO BTM0608C2X

El dispositivo con el modelo anteriormente descrito permite la comunicación serial y el envío de audio en un solo bluetooth lo que no solo permite establecer una vía directa y sencilla de conexión para los usuarios si no que también se tendría que usar un módulo bluetooth menos.

Originalmente la vela versión original de DIMA contaba con:

- Bluetooth Classic V2.0 Modelo HC-05
- Bluetooth BLE V4.0 Audio BK-8000L

La versión mejorada propuesta de DIMA cuenta con:

- Bluetooth BLE V4.0 Modelo HM-10
- Bluetooth BLE V4.0 Audio BK-8000L

De lo anterior se puede apreciar que el módulo de audio siempre estuvo presente, con el mismo modelo BK-8000L sin embargo en el proceso de investigación surge un módulo bluetooth que según sus características técnicas es capaz de realizar ambos procesos al mismo tiempo, envío y recepción de audio y aunado a esto envío y recepción de datos mediante USART.

Realizada la adquisición y pruebas en la fase PRE ALFA se determinó que es posible usar el dispositivo para tal propósito, no obstante no hay mucha información sobre el mismo y se requiere comunicación directa con el fabricante para ajustar el código agregando los perfiles GATT necesarios del fabricante para conectar.

al momento no se ha logrado conectar satisfactoriamente con mencionado módulo: para mayor información puede visitar el siguiente documento: [LOG CAT BLE CONNECTION](#)