

Proyecto\_BioM\_Android

Generado por Doxygen 1.12.0



<b>1 Índice de espacios de nombres</b>	<b>1</b>
1.1 Lista de espacios de nombres	1
<b>2 Índice jerárquico</b>	<b>3</b>
2.1 Jerarquía de clases	3
<b>3 Índice de clases</b>	<b>5</b>
3.1 Lista de clases	5
<b>4 Índice de archivos</b>	<b>7</b>
4.1 Lista de archivos	7
<b>5 Documentación de espacios de nombres</b>	<b>9</b>
5.1 Paquete com.example.aplicacion_3ro	9
<b>6 Documentación de clases</b>	<b>11</b>
6.1 Referencia de la interface com.example.aplicacion_3ro.ApiService	11
6.1.1 Descripción detallada	11
6.1.2 Documentación de funciones miembro	11
6.1.2.1 enviarMediciones()	11
6.2 Referencia de la clase com.example.aplicacion_3ro.MainActivity	12
6.2.1 Descripción detallada	14
6.2.2 Documentación de funciones miembro	14
6.2.2.1 botonBuscarDispositivosBTLEPulsado()	14
6.2.2.2 botonBuscarNuestroDispositivoBTLEPulsado()	14
6.2.2.3 botonDetenerBusquedaDispositivosBTLEPulsado()	15
6.2.2.4 buscarEsteDispositivoBTLE()	16
6.2.2.5 buscarTodosLosDispositivosBTLE()	17
6.2.2.6 detenerBusquedaDispositivosBTLE()	18
6.2.2.7 enviarDatosAlServidor()	19
6.2.2.8 inicializarBlueTooth()	20
6.2.2.9 mostrarInformacionDispositivoBTLE()	20
6.2.2.10 obtenerTipoDeSensor()	22
6.2.2.11 onCreate()	22
6.2.2.12 onRequestPermissionsResult()	22
6.2.3 Documentación de datos miembro	23
6.2.3.1 apiService	23
6.2.3.2 callbackDelEscaneo	23
6.2.3.3 CODIGO_PETICION_PERMISOS	23
6.2.3.4 elEscanner	23
6.2.3.5 ETIQUETA_LOG	23
6.2.3.6 valor1	23
6.2.3.7 valor2	24
6.3 Referencia de la clase com.example.aplicacion_3ro.Medicion	24

6.3.1 Descripción detallada	24
6.3.2 Documentación de constructores y destructores	25
6.3.2.1 Medicion()	25
6.3.3 Documentación de funciones miembro	25
6.3.3.1 getDato()	25
6.3.3.2 getId_tipo()	25
6.3.3.3 getId_usuario()	25
6.3.3.4 setDato()	25
6.3.3.5 setId_tipo()	26
6.3.3.6 setId_usuario()	26
6.3.4 Documentación de datos miembro	26
6.3.4.1 dato	26
6.3.4.2 id_tipo	26
6.3.4.3 id_usuario	27
6.4 Referencia de la clase com.example.aplicacion_3ro.TramalBeacon	27
6.4.1 Descripción detallada	28
6.4.2 Documentación de constructores y destructores	28
6.4.2.1 TramalBeacon()	28
6.4.3 Documentación de funciones miembro	29
6.4.3.1 getAdvFlags()	29
6.4.3.2 getAdvHeader()	29
6.4.3.3 getCompanyID()	30
6.4.3.4 getiBeaconLength()	30
6.4.3.5 getiBeaconType()	30
6.4.3.6 getLosBytes()	31
6.4.3.7 getMajor()	31
6.4.3.8 getMinor()	31
6.4.3.9 getPrefijo()	32
6.4.3.10 getTxPower()	32
6.4.3.11 getUUID()	32
6.4.4 Documentación de datos miembro	33
6.4.4.1 advFlags	33
6.4.4.2 advHeader	33
6.4.4.3 companyID	33
6.4.4.4 iBeaconLength	33
6.4.4.5 iBeaconType	33
6.4.4.6 losBytes	33
6.4.4.7 major	33
6.4.4.8 minor	34
6.4.4.9 noadvFlags	34
6.4.4.10 prefijo	34
6.4.4.11 txPower	34

6.4.4.12 uuid	34
6.5 Referencia de la clase <code>com.example.aplicacion_3ro.Utilidades</code>	34
6.5.1 Descripción detallada	35
6.5.2 Documentación de funciones miembro	35
6.5.2.1 <code>bytesToHexString()</code>	35
6.5.2.2 <code>bytesToInt()</code>	36
6.5.2.3 <code>bytesToIntOK()</code>	36
6.5.2.4 <code>bytesToLong()</code>	37
6.5.2.5 <code>bytesToString()</code>	37
6.5.2.6 <code>dosLongToBytes()</code>	38
6.5.2.7 <code>stringToBytes()</code>	38
6.5.2.8 <code>stringToUUID()</code>	39
6.5.2.9 <code>uuidToHexString()</code>	40
6.5.2.10 <code>uuidToString()</code>	41
<b>7 Documentación de archivos</b>	<b>43</b>
7.1 Referencia del archivo <code>BTLEAlumnos2021/Aplicacion_3ro/app/src/main/java/com/example/aplicacion_3ro/ApiService.java</code>	43
7.2 Referencia del archivo <code>BTLEAlumnos2021/Aplicacion_3ro/app/src/main/java/com/example/aplicacion_3ro/MainActivity.java</code>	44
7.3 Referencia del archivo <code>BTLEAlumnos2021/Aplicacion_3ro/app/src/main/java/com/example/aplicacion_3ro/Medicion.java</code>	44
7.4 Referencia del archivo <code>BTLEAlumnos2021/Aplicacion_3ro/app/src/main/java/com/example/aplicacion_3ro/TramalBeacon.java</code>	45
7.5 Referencia del archivo <code>BTLEAlumnos2021/Aplicacion_3ro/app/src/main/java/com/example/aplicacion_3ro/Utilidades.java</code>	45
<b>Índice alfabético</b>	<b>47</b>



# Capítulo 1

## Índice de espacios de nombres

### 1.1. Lista de espacios de nombres

Lista de los espacios de nombres documentados, con breves descripciones:

<a href="#">com.example.aplicacion_3ro</a> . . . . .	9
--	---





## Capítulo 2

# Índice jerárquico

### 2.1. Jerarquía de clases

Este listado de herencia está ordenado de forma general pero no está en orden alfabético estricto:

com.example.aplicacion_3ro.ApiService . . . . .	11
com.example.aplicacion_3ro.Medicion . . . . .	24
com.example.aplicacion_3ro.TramalBeacon . . . . .	27
com.example.aplicacion_3ro.Utilidades . . . . .	34
AppCompatActivity	
com.example.aplicacion_3ro.MainActivity . . . . .	12



## Capítulo 3

# Índice de clases

### 3.1. Lista de clases

Lista de clases, estructuras, uniones e interfaces con breves descripciones:

<a href="#">com.example.aplicacion_3ro.ApiService</a>	
Interfaz para definir las llamadas HTTP de la API . . . . .	11
<a href="#">com.example.aplicacion_3ro.MainActivity</a>	
Clase principal de la aplicación que maneja la interfaz de usuario y la interacción con Bluetooth LE . . . . .	12
<a href="#">com.example.aplicacion_3ro.Medicion</a>	
Clase que representa una medición enviada al servidor . . . . .	24
<a href="#">com.example.aplicacion_3ro.TramaIBeacon</a>	
Clase que representa y procesa una trama iBeacon . . . . .	27
<a href="#">com.example.aplicacion_3ro.Utilidades</a>	
Clase de utilidades para conversiones de datos y manipulación de UUIDs . . . . .	34



## Capítulo 4

# Índice de archivos

### 4.1. Lista de archivos

Lista de todos los archivos con breves descripciones:

BTLEAlumnos2021/Aplicacion_3ro/app/src/main/java/com/example/aplicacion_3ro/ <a href="#">ApiService.java</a> . . .	43
BTLEAlumnos2021/Aplicacion_3ro/app/src/main/java/com/example/aplicacion_3ro/ <a href="#">MainActivity.java</a> . .	44
BTLEAlumnos2021/Aplicacion_3ro/app/src/main/java/com/example/aplicacion_3ro/ <a href="#">Medicion.java</a> . . .	44
BTLEAlumnos2021/Aplicacion_3ro/app/src/main/java/com/example/aplicacion_3ro/ <a href="#">TramalBeacon.java</a> .	45
BTLEAlumnos2021/Aplicacion_3ro/app/src/main/java/com/example/aplicacion_3ro/ <a href="#">Utilidades.java</a> . . .	45



## Capítulo 5

# Documentación de espacios de nombres

### 5.1. Paquete com.example.aplicacion\_3ro

#### Clases

- interface [ApiService](#)  
*Interfaz para definir las llamadas HTTP de la API.*
- class [MainActivity](#)  
*Clase principal de la aplicación que maneja la interfaz de usuario y la interacción con Bluetooth LE.*
- class [Medicion](#)  
*Clase que representa una medición enviada al servidor.*
- class [TramaBeacon](#)  
*Clase que representa y procesa una trama iBeacon.*
- class [Utilidades](#)  
*Clase de utilidades para conversiones de datos y manipulación de UUIDs.*





## Capítulo 6

# Documentación de clases

### 6.1. Referencia de la interface com.example.aplicacion\_3ro.ApiService

Interfaz para definir las llamadas HTTP de la API.

#### Métodos públicos

- Call< Void > [enviarMediciones](#) (@Body [Medicion](#) medicion)  
*Envía una medición al servidor.*

#### 6.1.1. Descripción detallada

Interfaz para definir las llamadas HTTP de la API.

Esta interfaz define el método para enviar las mediciones al servidor utilizando una solicitud HTTP POST. Retrofit se encargará de crear la implementación de esta interfaz en tiempo de ejecución.

#### 6.1.2. Documentación de funciones miembro

##### 6.1.2.1. [enviarMediciones\(\)](#)

```
Call< Void > com.example.aplicacion_3ro.ApiService.enviarMediciones (  
    @Body Medicion medicion)
```

Envía una medición al servidor.

Este método realiza una solicitud POST al endpoint "/mediciones" para enviar los datos de una medición al servidor.

#### Parámetros

<i>medicion</i>	El objeto de tipo <a href="#">Medicion</a> que contiene los datos a enviar.
-----------------	---

**Devuelve**

Call<Void> Objeto Call de Retrofit que representa la solicitud HTTP.

Gráfico de llamadas a esta función:



La documentación de esta interface está generada del siguiente archivo:

- BTLEAlumnos2021/Aplicacion\_3ro/app/src/main/java/com/example/aplicacion\_3ro/[ApiService.java](#)

## 6.2. Referencia de la clase com.example.aplicacion\_3ro.MainActivity

Clase principal de la aplicación que maneja la interfaz de usuario y la interacción con Bluetooth LE.

Diagrama de herencia de com.example.aplicacion\_3ro.MainActivity

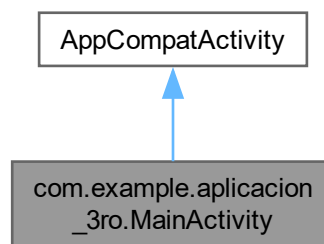
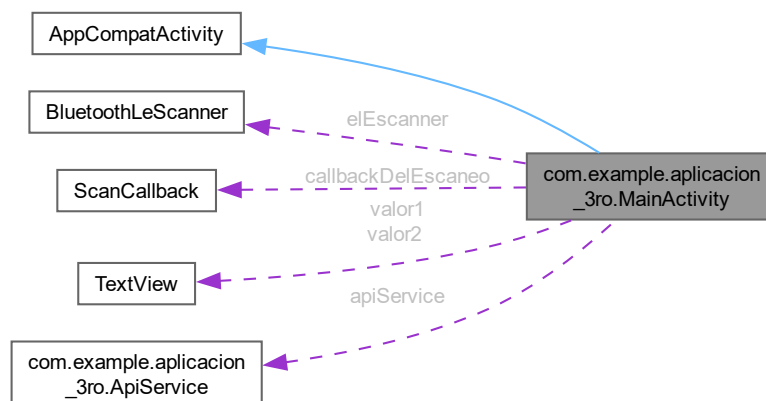


Diagrama de colaboración de com.example.aplicacion\_3ro.MainActivity:



### Métodos públicos

- void [botonBuscarNuestroDispositivoBTLEPulsado](#) (View v)  
*Método invocado al pulsar el botón para buscar un dispositivo BTLE específico.*
- void [botonBuscarDispositivosBTLEPulsado](#) (View v)  
*Método invocado al pulsar el botón para buscar todos los dispositivos BTLE.*
- void [botonDetenerBusquedaDispositivosBTLEPulsado](#) (View v)  
*Método invocado al pulsar el botón para detener la búsqueda de dispositivos BTLE.*
- void [onRequestPermissionsResult](#) (int requestCode, String[] permissions, int[] grantResults)  
*Método que gestiona los resultados de la solicitud de permisos.*

### Atributos públicos

- TextView [valor1](#)
- TextView [valor2](#)

### Métodos protegidos

- void [onCreate](#) (Bundle savedInstanceState)  
*Método llamado al crear la actividad. Inicializa la vista y configura el escáner Bluetooth.*

### Métodos privados

- void [enviarDatosAlServidor](#) (final int idTipo, final int valor, final int idUsuario)  
*Envía los datos al servidor utilizando Retrofit.*
- void [buscarTodosLosDispositivosBTLE](#) ()  
*Inicia la búsqueda de todos los dispositivos Bluetooth LE.*
- void [mostrarInformacionDispositivoBTLE](#) (ScanResult resultado)  
*Muestra la información detallada del dispositivo Bluetooth LE detectado.*
- int [obtenerTipoDeSensor](#) (int majorValue)  
*Determina el tipo de sensor basándose en el valor de 'major'.*
- void [buscarEsteDispositivoBTLE](#) (final UUID dispositivoBuscado)  
*Inicia un escaneo Bluetooth LE para buscar un dispositivo con un UUID específico.*
- void [detenerBusquedaDispositivosBTLE](#) ()  
*Detiene la búsqueda de dispositivos BTLE en curso.*
- void [inicializarBlueTooth](#) ()  
*Inicializa el adaptador Bluetooth y solicita permisos necesarios.*

### Atributos privados

- BluetoothLeScanner [elEscanner](#)
- ScanCallback [callbackDelEscaneo](#) = null
- ApiService [apiService](#)

### Atributos estáticos privados

- static final String [ETIQUETA\\_LOG](#) = ">>>>"
- static final int [CODIGO\\_PETICION\\_PERMISOS](#) = 11223344

### 6.2.1. Descripción detallada

Clase principal de la aplicación que maneja la interfaz de usuario y la interacción con Bluetooth LE.

### 6.2.2. Documentación de funciones miembro

#### 6.2.2.1. botonBuscarDispositivosBTLEPulsado()

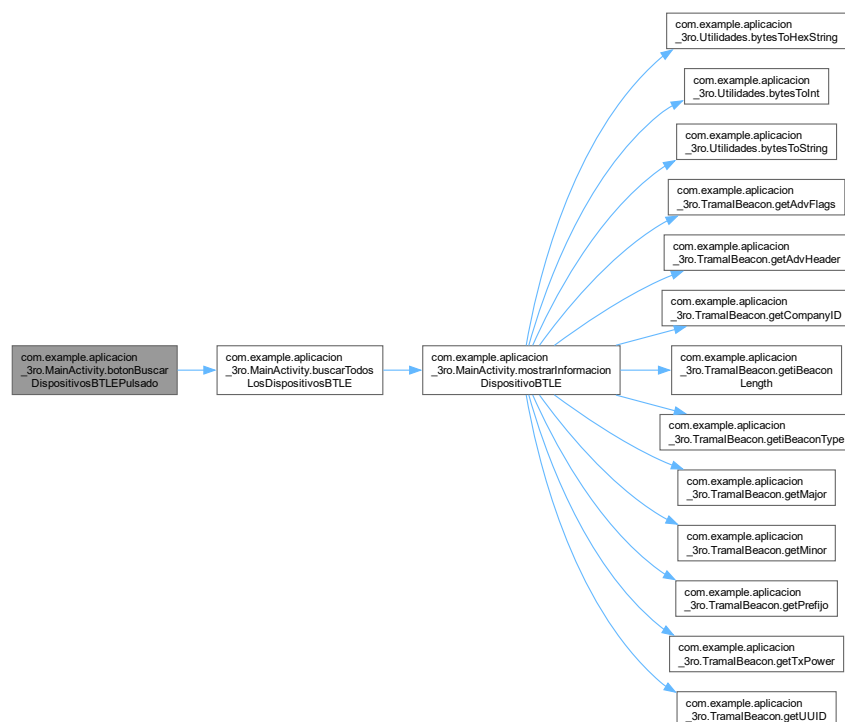
```
void com.example.aplicacion_3ro.MainActivity.botonBuscarDispositivosBTLEPulsado (
    View v) [inline]
```

Método invocado al pulsar el botón para buscar todos los dispositivos BTLE.

##### Parámetros

v	Vista que representa el botón pulsado.
---	--

Este método inicia una búsqueda para todos los dispositivos Bluetooth LE cercanos llamando a la función correspondiente. Gráfico de llamadas de esta función:



#### 6.2.2.2. botonBuscarNuestroDispositivoBTLEPulsado()

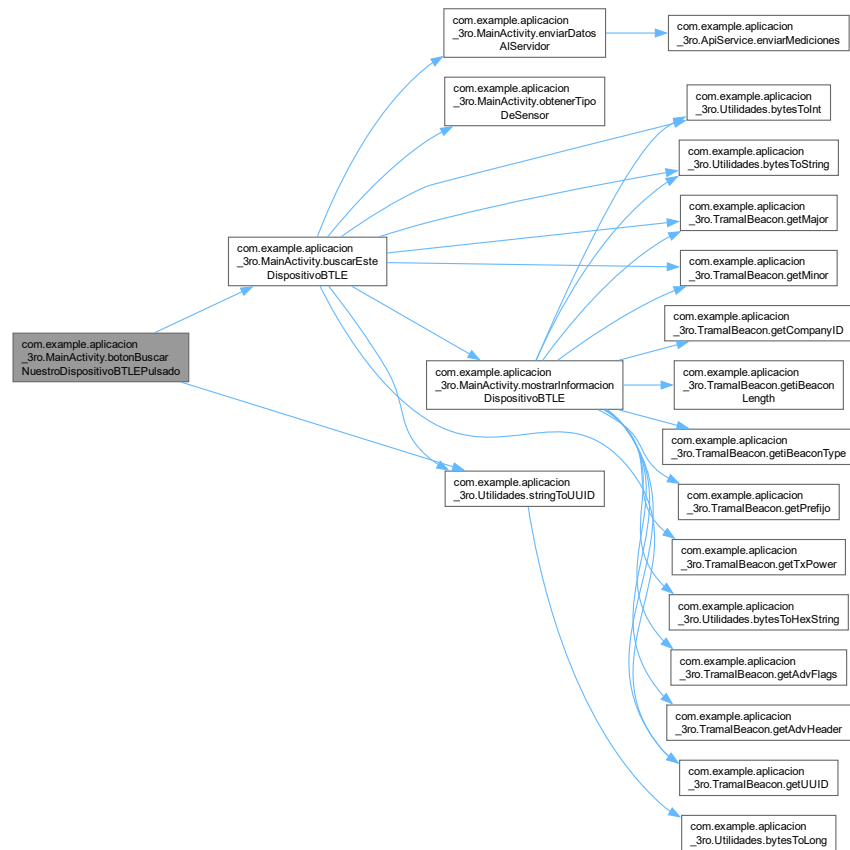
```
void com.example.aplicacion_3ro.MainActivity.botonBuscarNuestroDispositivoBTLEPulsado (
    View v) [inline]
```

Método invocado al pulsar el botón para buscar un dispositivo BTLE específico.

## Parámetros

v	Vista que representa el botón pulsado.
---	--

Este método utiliza un UUID predefinido y llama a la función para iniciar la búsqueda de dicho dispositivo mediante Bluetooth LE. Gráfico de llamadas de esta función:



## 6.2.2.3. botonDetenerBusquedaDispositivosBTLEPulsado()

```
void com.example.aplicacion_3ro.MainActivity.botonDetenerBusquedaDispositivosBTLEPulsado (
    View v) [inline]
```

Método invocado al pulsar el botón para detener la búsqueda de dispositivos BTLE.

## Parámetros

v	Vista que representa el botón pulsado.
---	--

---

Detiene la búsqueda de dispositivos Bluetooth LE en curso. Gráfico de llamadas de esta función:



#### 6.2.2.4. buscarEsteDispositivoBTLE()

```
void com.example.aplicacion_3ro.MainActivity.buscarEsteDispositivoBTLE (  
    final UUID dispositivoBuscado) [inline], [private]
```

Inicia un escaneo Bluetooth LE para buscar un dispositivo con un UUID específico.

##### Parámetros

<i>dispositivoBuscado</i>	UUID del dispositivo Bluetooth LE que se está buscando.
---------------------------	---

Esta función busca un dispositivo Bluetooth LE cuyo UUID coincida con el proporcionado. Si se encuentra el dispositivo, se extrae información adicional como el valor mayor y menor, se determina el tipo de sensor y se actualizan

los valores de temperatura u ozono en la interfaz. Gráfico de llamadas de esta función:

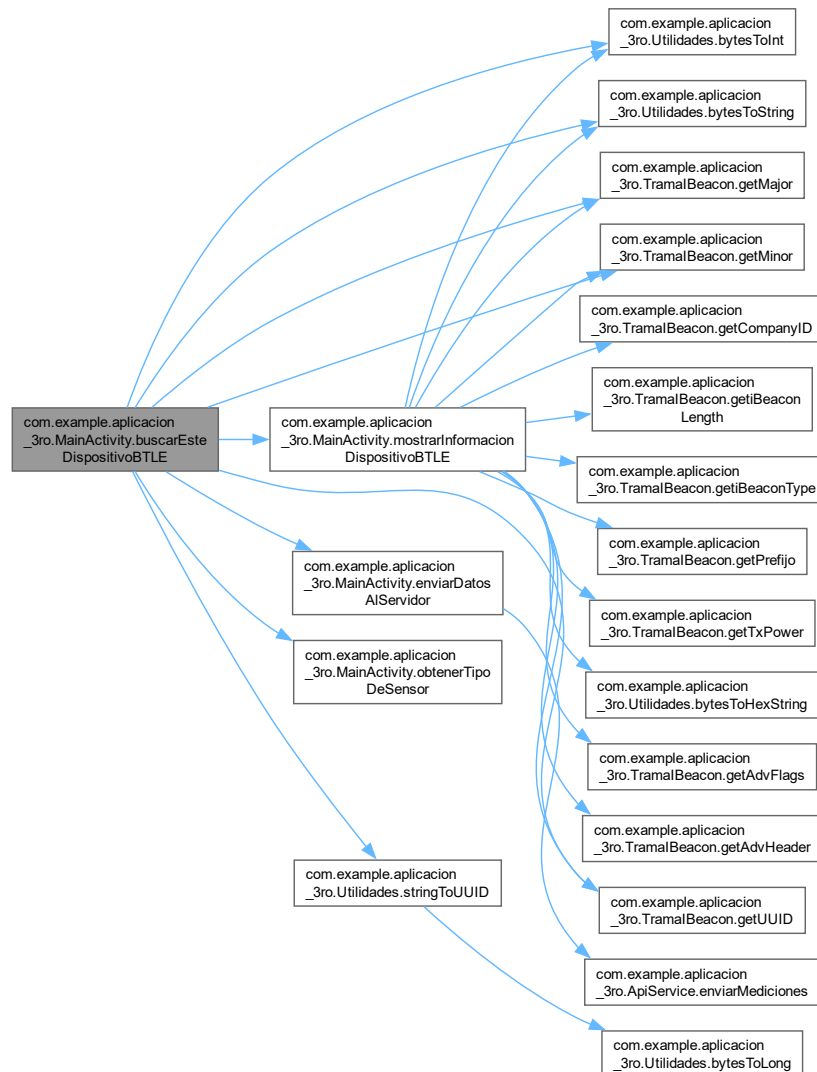


Gráfico de llamadas a esta función:



#### 6.2.2.5. buscarTodosLosDispositivosBTLE()

```
void com.example.aplicacion_3ro.MainActivity.buscarTodosLosDispositivosBTLE () [inline],
```

```
[private]
```

Inicia la búsqueda de todos los dispositivos Bluetooth LE.

Gráfico de llamadas de esta función:

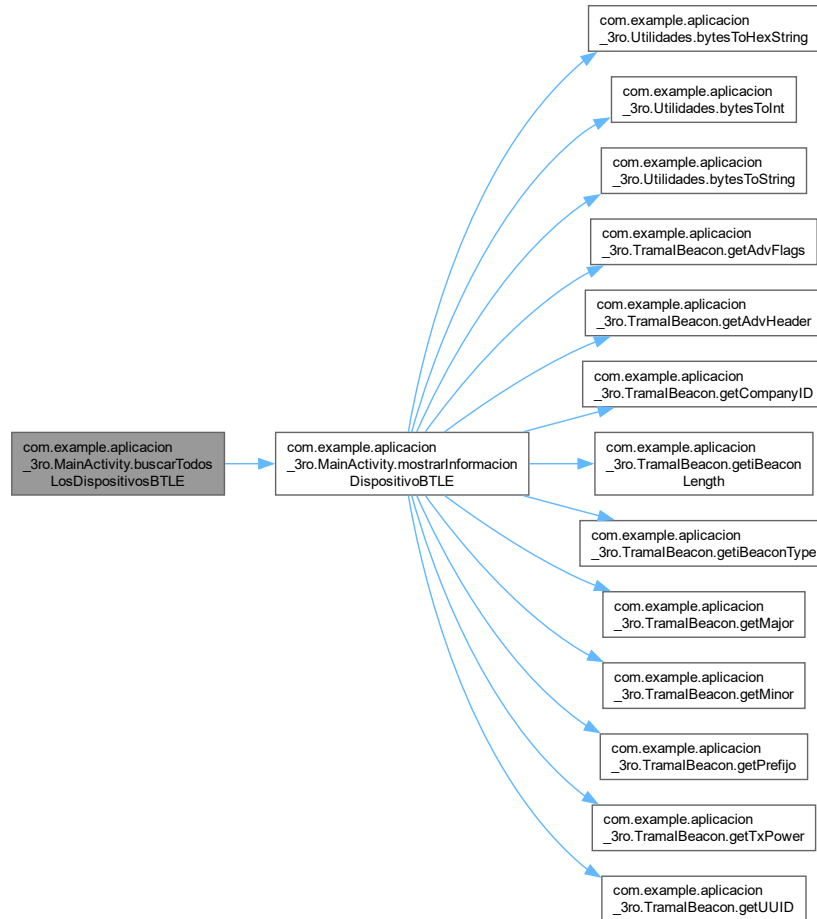
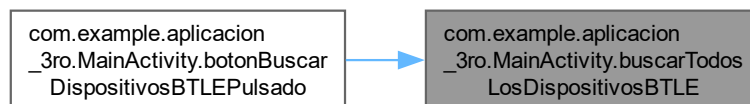


Gráfico de llamadas a esta función:



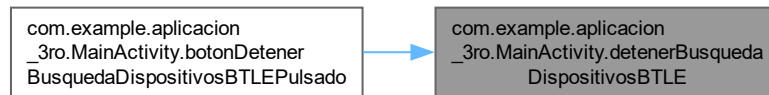
#### 6.2.2.6. detenerBusquedaDispositivosBTLE()

```
void com.example.aplicacion_3ro.MainActivity.detenerBusquedaDispositivosBTLE () [inline],
[private]
```



Detiene la búsqueda de dispositivos BTLE en curso.

Si la búsqueda está activa, detiene el escaneo y desactiva el callback asociado. Gráfico de llamadas a esta función:



#### 6.2.2.7. enviarDatosAlServidor()

```
void com.example.aplicacion_3ro.MainActivity.enviarDatosAlServidor (
    final int idTipo,
    final int valor,
    final int idUsuario) [inline], [private]
```

Envía los datos al servidor utilizando Retrofit.

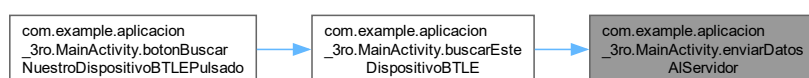
##### Parámetros

<i>idTipo</i>	Tipo de medición.
<i>valor</i>	Valor de la medición.
<i>idUsuario</i>	ID del usuario.

Gráfico de llamadas de esta función:



Gráfico de llamadas a esta función:



#### 6.2.2.8. inicializarBlueTooth()

```
void com.example.aplicacion_3ro.MainActivity.inicializarBlueTooth () [inline], [private]
```

Inicializa el adaptador Bluetooth y solicita permisos necesarios.

Este método obtiene el adaptador Bluetooth del dispositivo, habilita el Bluetooth si no está habilitado, y solicita los permisos necesarios dependiendo de la versión de Android para realizar escaneos de dispositivos Bluetooth LE.

En versiones de Android 12 y superiores, se solicitan permisos para escanear y conectar dispositivos Bluetooth cercanos. En versiones anteriores, se solicitan permisos para Bluetooth y localización.

#### 6.2.2.9. mostrarInformacionDispositivoBTLE()

```
void com.example.aplicacion_3ro.MainActivity.mostrarInformacionDispositivoBTLE (  
    ScanResult resultado) [inline], [private]
```

Muestra la información detallada del dispositivo Bluetooth LE detectado.

##### Parámetros

<i>resultado</i>	Resultado del escaneo del dispositivo Bluetooth LE.
------------------	---

Gráfico de llamadas de esta función:

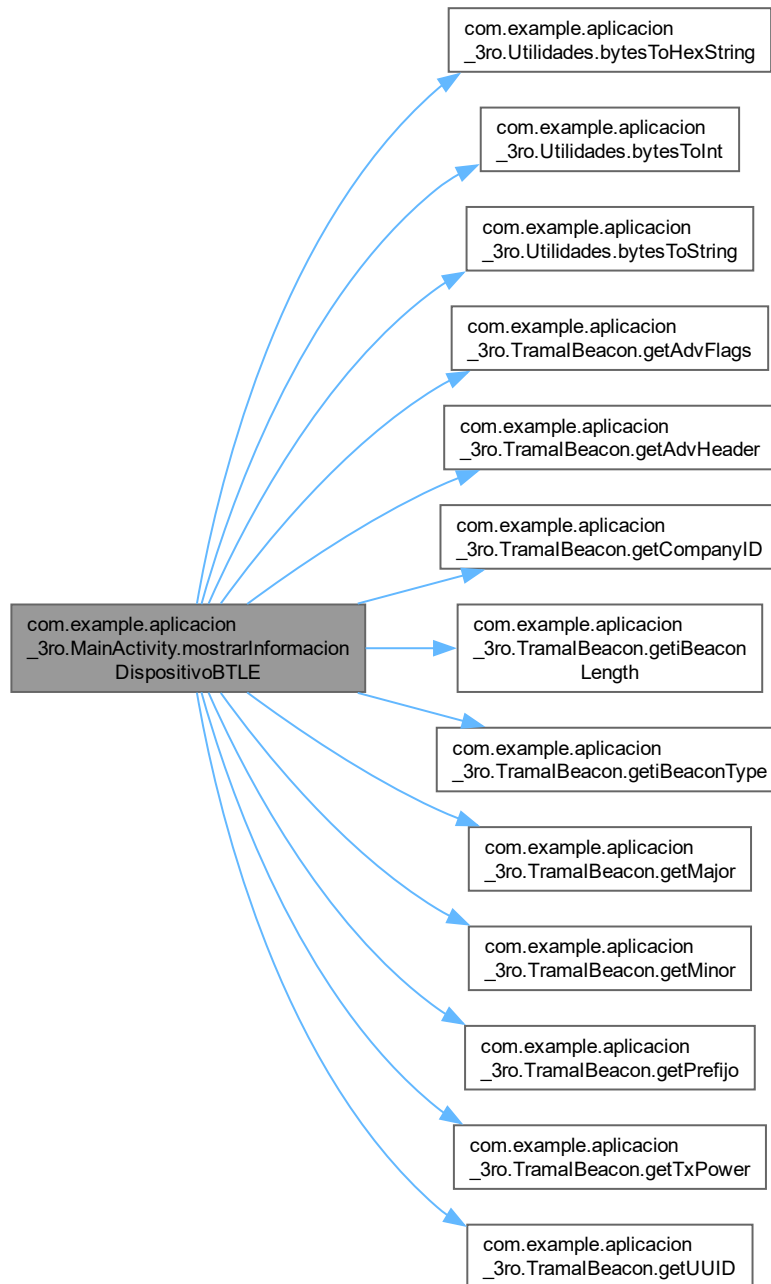
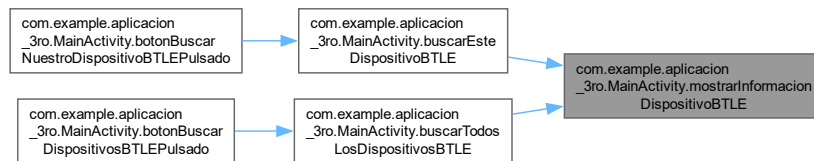


Gráfico de llamadas a esta función:



#### 6.2.2.10. obtenerTipoDeSensor()

```
int com.example.aplicacion_3ro.MainActivity.obtenerTipoDeSensor (
    int majorValue) [inline], [private]
```

Determina el tipo de sensor basándose en el valor de 'major'.

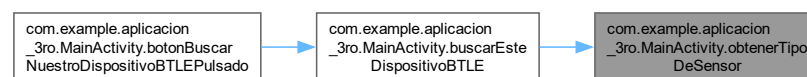
##### Parámetros

<i>majorValue</i>	Valor 'major' extraído del iBeacon.
-------------------	-------------------------------------

##### Devuelve

2 si el sensor es de ozono, 1 si es de temperatura, -1 si es desconocido.

Gráfico de llamadas a esta función:



#### 6.2.2.11. onCreate()

```
void com.example.aplicacion_3ro.MainActivity.onCreate (
    Bundle savedInstanceState) [inline], [protected]
```

Método llamado al crear la actividad. Inicializa la vista y configura el escáner Bluetooth.

##### Parámetros

<i>savedInstanceState</i>	Estado previamente guardado de la actividad.
---------------------------	--

#### 6.2.2.12. onRequestPermissionsResult()

```
void com.example.aplicacion_3ro.MainActivity.onRequestPermissionsResult (
    int requestCode,
    String[] permissions,
    int[] grantResults) [inline]
```

Método que gestiona los resultados de la solicitud de permisos.

**Parámetros**

<i>requestCode</i>	Código que identifica la solicitud de permisos.
<i>permissions</i>	Arreglo con los nombres de los permisos solicitados.
<i>grantResults</i>	Arreglo con los resultados de los permisos (concedidos o denegados).

Este método es llamado cuando el usuario concede o deniega los permisos solicitados. Si los permisos son concedidos, permite continuar con la inicialización o el escaneo.

**6.2.3. Documentación de datos miembro****6.2.3.1. apiService**

```
ApiService com.example.aplicacion_3ro.MainActivity.apiService [private]
```

Servicio API para enviar datos al servidor.

**6.2.3.2. callbackDelEscaneo**

```
ScanCallback com.example.aplicacion_3ro.MainActivity.callbackDelEscaneo = null [private]
```

Callback utilizado para manejar los resultados del escaneo BLE.

**6.2.3.3. CODIGO\_PETICION\_PERMISOS**

```
final int com.example.aplicacion_3ro.MainActivity.CODIGO_PETICION_PERMISOS = 11223344 [static],  
[private]
```

Código de petición para los permisos requeridos.

**6.2.3.4. elEscanner**

```
BluetoothLeScanner com.example.aplicacion_3ro.MainActivity.elEscanner [private]
```

Escáner Bluetooth Low Energy (BLE) utilizado para buscar dispositivos.

**6.2.3.5. ETIQUETA\_LOG**

```
final String com.example.aplicacion_3ro.MainActivity.ETIQUETA_LOG = ">>>>" [static], [private]
```

Etiqueta de log para el seguimiento de eventos.

**6.2.3.6. valor1**

```
TextView com.example.aplicacion_3ro.MainActivity.valor1
```

TextView para mostrar el valor de la temperatura.

#### 6.2.3.7. valor2

```
TextView com.example.aplicacion_3ro.MainActivity.valor2
```

TextView para mostrar el valor del ozono.

La documentación de esta clase está generada del siguiente archivo:

- [BTLEAlumnos2021/Aplicacion\\_3ro/app/src/main/java/com/example/aplicacion\\_3ro/MainActivity.java](#)

### 6.3. Referencia de la clase com.example.aplicacion\_3ro.Medicion

Clase que representa una medición enviada al servidor.

#### Métodos públicos

- [Medicion](#) (int [dato](#), int [id\\_tipo](#), int [id\\_usuario](#))  
*Constructor de la clase [Medicion](#).*
- int [getDato](#) ()  
*Obtiene el valor de la medición.*
- void [setDato](#) (int [dato](#))  
*Establece el valor de la medición.*
- int [getId\\_tipo](#) ()  
*Obtiene el identificador del tipo de medición.*
- void [setId\\_tipo](#) (int [id\\_tipo](#))  
*Establece el identificador del tipo de medición.*
- int [getId\\_usuario](#) ()  
*Obtiene el identificador del usuario.*
- void [setId\\_usuario](#) (int [id\\_usuario](#))  
*Establece el identificador del usuario.*

#### Atributos privados

- int [dato](#)  
*Valor de la medición.*
- int [id\\_tipo](#)  
*Identificador del tipo de medición.*
- int [id\\_usuario](#)  
*Identificador del usuario.*

#### 6.3.1. Descripción detallada

Clase que representa una medición enviada al servidor.

Esta clase contiene los atributos de una medición: el valor del dato, el tipo de medición y el identificador del usuario que realiza la medición.

## 6.3.2. Documentación de constructores y destructores

### 6.3.2.1. Medicion()

```
com.example.aplicacion_3ro.Medicion.Medicion (
    int dato,
    int id_tipo,
    int id_usuario) [inline]
```

Constructor de la clase [Medicion](#).

Inicializa una nueva instancia de la clase [Medicion](#) con los valores especificados.

#### Parámetros

<i>dato</i>	El valor de la medición.
<i>id_tipo</i>	El identificador del tipo de medición.
<i>id_usuario</i>	El identificador del usuario que realiza la medición.

## 6.3.3. Documentación de funciones miembro

### 6.3.3.1. getDato()

```
int com.example.aplicacion_3ro.Medicion.getDato () [inline]
```

Obtiene el valor de la medición.

#### Devuelve

El valor de la medición.

### 6.3.3.2. getId\_tipo()

```
int com.example.aplicacion_3ro.Medicion.getId_tipo () [inline]
```

Obtiene el identificador del tipo de medición.

#### Devuelve

El identificador del tipo de medición.

### 6.3.3.3. getId\_usuario()

```
int com.example.aplicacion_3ro.Medicion.getId_usuario () [inline]
```

Obtiene el identificador del usuario.

#### Devuelve

El identificador del usuario.

### 6.3.3.4. setDato()

```
void com.example.aplicacion_3ro.Medicion.setDato (
    int dato) [inline]
```

Establece el valor de la medición.

**Parámetros**

<i>dato</i>	El valor a establecer.
-------------	------------------------

**6.3.3.5. setId\_tipo()**

```
void com.example.aplicacion_3ro.Medicion.setId_tipo (  
    int id_tipo) [inline]
```

Establece el identificador del tipo de medición.

**Parámetros**

<i>id_tipo</i>	El identificador del tipo de medición a establecer.
----------------	---

**6.3.3.6. setId\_usuario()**

```
void com.example.aplicacion_3ro.Medicion.setId_usuario (  
    int id_usuario) [inline]
```

Establece el identificador del usuario.

**Parámetros**

<i>id_usuario</i>	El identificador del usuario a establecer.
-------------------	--

**6.3.4. Documentación de datos miembro****6.3.4.1. dato**

```
int com.example.aplicacion_3ro.Medicion.dato [private]
```

Valor de la medición.

Representa el valor de la medición que se quiere enviar al servidor.

**6.3.4.2. id\_tipo**

```
int com.example.aplicacion_3ro.Medicion.id_tipo [private]
```

Identificador del tipo de medición.

Este valor indica el tipo de medición que se está realizando, por ejemplo, si es una medición de temperatura o de ozono.



#### 6.3.4.3. id\_usuario

```
int com.example.aplicacion_3ro.Medicion.id_usuario [private]
```

Identificador del usuario.

El ID del usuario que está realizando la medición.

La documentación de esta clase está generada del siguiente archivo:

- [BTLEAlumnos2021/Aplicacion\\_3ro/app/src/main/java/com/example/aplicacion\\_3ro/Medicion.java](#)

## 6.4. Referencia de la clase com.example.aplicacion\_3ro.TramalBeacon

Clase que representa y procesa una trama iBeacon.

### Métodos públicos

- [TramalBeacon](#) (byte[] bytes)  
*Constructor que recibe un arreglo de bytes y procesa la trama iBeacon.*
- byte[] [getPrefijo](#) ()  
*Obtiene el prefijo de la trama.*
- byte[] [getUUID](#) ()  
*Obtiene el UUID de la trama.*
- byte[] [getMajor](#) ()  
*Obtiene el valor del campo mayor.*
- byte[] [getMinor](#) ()  
*Obtiene el valor del campo menor.*
- byte [getTxPower](#) ()  
*Obtiene el valor de la potencia de transmisión (TxPower).*
- byte[] [getLosBytes](#) ()  
*Obtiene el arreglo de bytes de la trama completa.*
- byte[] [getAdvFlags](#) ()  
*Obtiene el valor de las banderas de publicidad (advFlags).*
- byte[] [getAdvHeader](#) ()  
*Obtiene el encabezado de publicidad (advHeader).*
- byte[] [getCompanyID](#) ()  
*Obtiene el identificador de la compañía.*
- byte [getiBeaconType](#) ()  
*Obtiene el tipo de iBeacon.*
- byte [getiBeaconLength](#) ()  
*Obtiene la longitud de iBeacon.*

## Atributos privados

- `byte[] prefijo` = null  
*Prefijo de 9 bytes (o 6 bytes si no incluye advFlags).*
- `byte[] uuid` = null  
*UUID de 16 bytes del iBeacon.*
- `byte[] major` = null  
*Campo mayor de 2 bytes.*
- `byte[] minor` = null  
*Campo menor de 2 bytes.*
- `byte txPower` = 0  
*Valor de la potencia de transmisión (TxPower) de 1 byte.*
- `byte[] losBytes`  
*Arreglo de bytes que representa la trama completa recibida.*
- `byte[] advFlags` = null  
*Bandera de publicidad de 3 bytes (solo si está presente).*
- `byte[] advHeader` = null  
*Cabecera de la publicidad (advertisement header) de 2 bytes.*
- `byte[] companyID` = new byte[2]  
*Identificador de la compañía de 2 bytes.*
- `byte iBeaconType` = 0  
*Tipo de iBeacon de 1 byte.*
- `byte iBeaconLength` = 0  
*Longitud del iBeacon de 1 byte.*
- `boolean noadvFlags`  
*Indica si los advFlags están presentes en la trama.*

### 6.4.1. Descripción detallada

Clase que representa y procesa una trama iBeacon.

Esta clase se encarga de interpretar los datos de una trama iBeacon, extrayendo información como el UUID, el mayor, el menor, y la potencia de transmisión (TxPower), entre otros campos.

### 6.4.2. Documentación de constructores y destructores

#### 6.4.2.1. TramaIBeacon()

```
com.example.aplicacion_3ro.TramaIBeacon.TramaIBeacon (
    byte[] bytes) [inline]
```

Constructor que recibe un arreglo de bytes y procesa la trama iBeacon.

#### Parámetros

<code>bytes</code>	Arreglo de bytes que representa la trama recibida del iBeacon.
--------------------	--

### 6.4.3. Documentación de funciones miembro

#### 6.4.3.1. getAdvFlags()

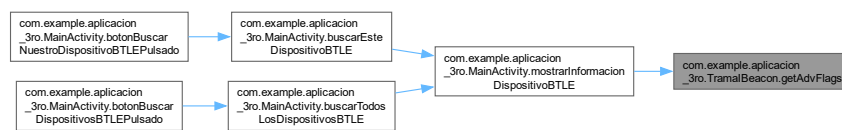
```
byte[] com.example.aplicacion_3ro.TramaIBeacon.getAdvFlags () [inline]
```

Obtiene el valor de las banderas de publicidad (advFlags).

##### Devuelve

El valor de advFlags en un arreglo de bytes.

Gráfico de llamadas a esta función:



#### 6.4.3.2. getAdvHeader()

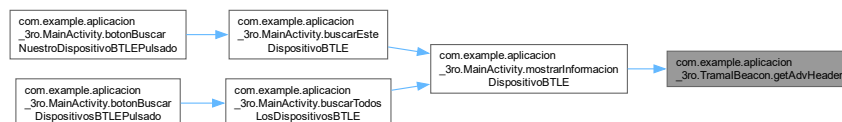
```
byte[] com.example.aplicacion_3ro.TramaIBeacon.getAdvHeader () [inline]
```

Obtiene el encabezado de publicidad (advHeader).

##### Devuelve

El encabezado de publicidad en un arreglo de bytes.

Gráfico de llamadas a esta función:



#### 6.4.3.3. getCompanyID()

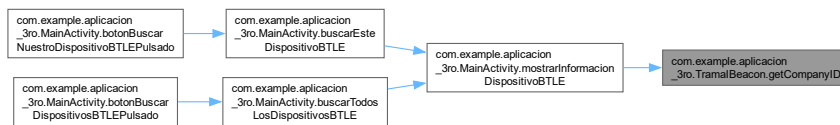
```
byte[] com.example.aplicacion_3ro.TramaIBeacon.getCompanyID () [inline]
```

Obtiene el identificador de la compañía.

**Devuelve**

El identificador de la compañía en un arreglo de bytes.

Gráfico de llamadas a esta función:



#### 6.4.3.4. getiBeaconLength()

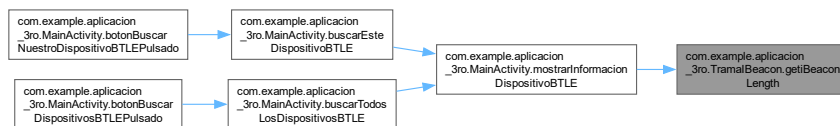
```
byte com.example.aplicacion_3ro.TramaIBeacon.getiBeaconLength() [inline]
```

Obtiene la longitud de iBeacon.

**Devuelve**

La longitud de iBeacon como un byte.

Gráfico de llamadas a esta función:



#### 6.4.3.5. getiBeaconType()

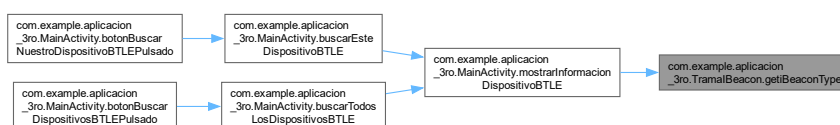
```
byte com.example.aplicacion_3ro.TramaIBeacon.getiBeaconType () [inline]
```

Obtiene el tipo de iBeacon.

**Devuelve**

El tipo de iBeacon como un byte.

Gráfico de llamadas a esta función:



#### 6.4.3.6. getLosBytes()

```
byte[] com.example.aplicacion_3ro.TramaIBeacon.getLosBytes () [inline]
```

Obtiene el arreglo de bytes de la trama completa.

**Devuelve**

El arreglo de bytes de la trama completa.

#### 6.4.3.7. getMayor()

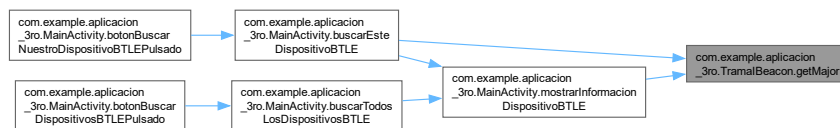
```
byte[] com.example.aplicacion_3ro.TramaIBeacon.getMayor () [inline]
```

Obtiene el valor del campo mayor.

**Devuelve**

El valor mayor en un arreglo de bytes.

Gráfico de llamadas a esta función:



#### 6.4.3.8. getMinor()

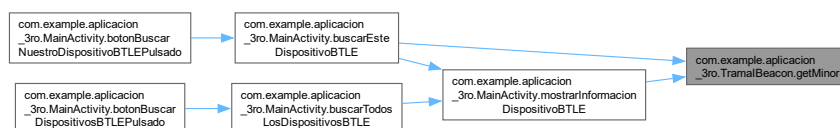
```
byte[] com.example.aplicacion_3ro.TramaIBeacon.getMinor () [inline]
```

Obtiene el valor del campo menor.

**Devuelve**

El valor menor en un arreglo de bytes.

Gráfico de llamadas a esta función:



#### 6.4.3.9. getPrefijo()

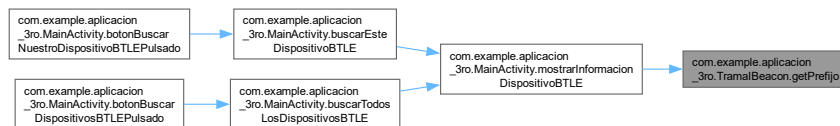
```
byte[] com.example.aplicacion_3ro.TramaIBeacon.getPrefijo () [inline]
```

Obtiene el prefijo de la trama.

**Devuelve**

El prefijo en un arreglo de bytes.

Gráfico de llamadas a esta función:



#### 6.4.3.10. getTxPower()

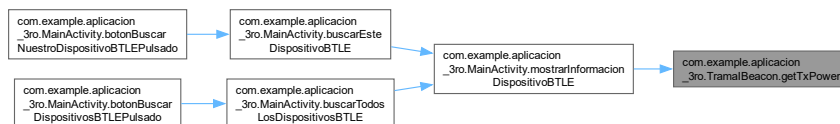
```
byte com.example.aplicacion_3ro.TramaIBeacon.getTxPower () [inline]
```

Obtiene el valor de la potencia de transmisión (TxPower).

**Devuelve**

El valor de TxPower como un byte.

Gráfico de llamadas a esta función:



#### 6.4.3.11. getUUID()

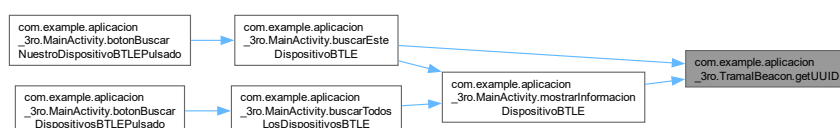
```
byte[] com.example.aplicacion_3ro.TramaIBeacon.getUUID () [inline]
```

Obtiene el UUID de la trama.

**Devuelve**

El UUID en un arreglo de bytes.

Gráfico de llamadas a esta función:



## 6.4.4. Documentación de datos miembro

### 6.4.4.1. `advFlags`

```
byte [] com.example.aplicacion_3ro.TramaIBeacon.advFlags = null [private]
```

Bandera de publicidad de 3 bytes (solo si está presente).

### 6.4.4.2. `advHeader`

```
byte [] com.example.aplicacion_3ro.TramaIBeacon.advHeader = null [private]
```

Cabecera de la publicidad (advertisement header) de 2 bytes.

### 6.4.4.3. `companyID`

```
byte [] com.example.aplicacion_3ro.TramaIBeacon.companyID = new byte[2] [private]
```

Identificador de la compañía de 2 bytes.

### 6.4.4.4. `iBeaconLength`

```
byte com.example.aplicacion_3ro.TramaIBeacon.iBeaconLength = 0 [private]
```

Longitud del iBeacon de 1 byte.

### 6.4.4.5. `iBeaconType`

```
byte com.example.aplicacion_3ro.TramaIBeacon.iBeaconType = 0 [private]
```

Tipo de iBeacon de 1 byte.

### 6.4.4.6. `losBytes`

```
byte [] com.example.aplicacion_3ro.TramaIBeacon.losBytes [private]
```

Arreglo de bytes que representa la trama completa recibida.

### 6.4.4.7. `major`

```
byte [] com.example.aplicacion_3ro.TramaIBeacon.major = null [private]
```

Campo major de 2 bytes.

#### 6.4.4.8. minor

```
byte [] com.example.aplicacion_3ro.TramaIBeacon.minor = null [private]
```

Campo minor de 2 bytes.

#### 6.4.4.9. noadvFlags

```
boolean com.example.aplicacion_3ro.TramaIBeacon.noadvFlags [private]
```

Indica si los advFlags están presentes en la trama.

#### 6.4.4.10. prefijo

```
byte [] com.example.aplicacion_3ro.TramaIBeacon.prefijo = null [private]
```

Prefijo de 9 bytes (o 6 bytes si no incluye advFlags).

#### 6.4.4.11. txPower

```
byte com.example.aplicacion_3ro.TramaIBeacon.txPower = 0 [private]
```

Valor de la potencia de transmisión (TxPower) de 1 byte.

#### 6.4.4.12. uuid

```
byte [] com.example.aplicacion_3ro.TramaIBeacon.uuid = null [private]
```

UUID de 16 bytes del iBeacon.

La documentación de esta clase está generada del siguiente archivo:

- [BTLEAlumnos2021/Aplicacion\\_3ro/app/src/main/java/com/example/aplicacion\\_3ro/TramaIBeacon.java](#)

## 6.5. Referencia de la clase com.example.aplicacion\_3ro.Utilidades

Clase de utilidades para conversiones de datos y manipulación de UUIDs.



## Métodos públicos estáticos

- static byte[] [stringToBytes](#) (String texto)  
*Convierte un String a un arreglo de bytes.*
- static UUID [stringToUUID](#) (String uuid)  
*Convierte un string UUID a un objeto UUID.*
- static String [uuidToString](#) (UUID uuid)  
*Convierte un objeto UUID a String.*
- static String [uuidToHexString](#) (UUID uuid)  
*Convierte un UUID en su representación hexadecimal.*
- static String [bytesToString](#) (byte[] bytes)  
*Convierte un arreglo de bytes en String.*
- static byte[] [dosLongToBytes](#) (long masSignificativos, long menosSignificativos)  
*Convierte dos valores long en un arreglo de bytes.*
- static int [bytesToInt](#) (byte[] bytes)  
*Convierte un arreglo de bytes a un entero.*
- static long [bytesToLong](#) (byte[] bytes)  
*Convierte un arreglo de bytes a un long.*
- static int [bytesToIntOK](#) (byte[] bytes)  
*Convierte un arreglo de bytes a entero (int) manejando los casos de signo.*
- static String [bytesToHexString](#) (byte[] bytes)  
*Convierte un arreglo de bytes a un String en formato hexadecimal.*

### 6.5.1. Descripción detallada

Clase de utilidades para conversiones de datos y manipulación de UUIDs.

Esta clase incluye métodos para convertir cadenas a bytes, UUIDs a cadenas, conversión de bytes a tipos primitivos como int y long, y otros tipos de conversiones útiles.

### 6.5.2. Documentación de funciones miembro

#### 6.5.2.1. bytesToHexString()

```
static String com.example.aplicacion_3ro.Utilidades.bytesToHexString (
    byte[] bytes) [inline], [static]
```

Convierte un arreglo de bytes a un String en formato hexadecimal.

Cada byte se convierte a dos caracteres hexadecimales.

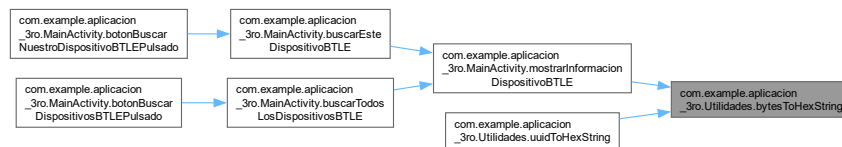
#### Parámetros

<i>bytes</i>	Arreglo de bytes.
--------------	-------------------

**Devuelve**

String en formato hexadecimal con los bytes separados por ":".

Gráfico de llamadas a esta función:

**6.5.2.2. bytesToInt()**

```
static int com.example.aplicacion_3ro.Utilidades.bytesToInt (
    byte[] bytes) [inline], [static]
```

Convierte un arreglo de bytes a un entero.

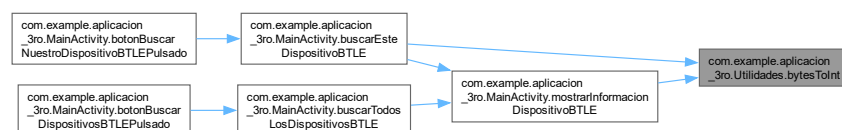
**Parámetros**

<i>bytes</i>	Arreglo de bytes.
--------------	-------------------

**Devuelve**

Entero equivalente al arreglo de bytes.

Gráfico de llamadas a esta función:

**6.5.2.3. bytesToIntOK()**

```
static int com.example.aplicacion_3ro.Utilidades.bytesToIntOK (
    byte[] bytes) [inline], [static]
```

Convierte un arreglo de bytes a entero (int) manejando los casos de signo.

Este método convierte bytes a entero utilizando el concepto de "complemento a 2" para manejar signos negativos.

**Parámetros**

<i>bytes</i>	Arreglo de bytes a convertir.
--------------	-------------------------------

**Devuelve**

El valor entero resultante.

**Excepciones**

<i>Error</i>	Si el arreglo de bytes es mayor a 4 bytes (tamaño de int).
--------------	--

**6.5.2.4. bytesToLong()**

```
static long com.example.aplicacion_3ro.Utilidades.bytesToLong (  
    byte[] bytes) [inline], [static]
```

Convierte un arreglo de bytes a un long.

**Parámetros**

<i>bytes</i>	Arreglo de bytes.
--------------	-------------------

**Devuelve**

Long equivalente al arreglo de bytes.

Gráfico de llamadas a esta función:

**6.5.2.5. bytesToString()**

```
static String com.example.aplicacion_3ro.Utilidades.bytesToString (  
    byte[] bytes) [inline], [static]
```

Convierte un arreglo de bytes en String.

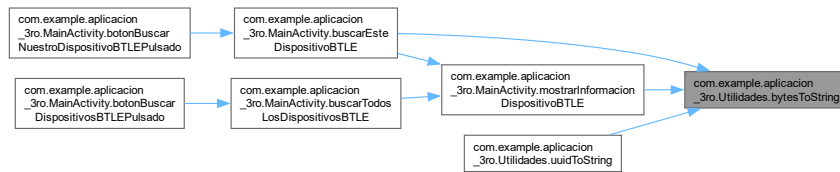
**Parámetros**

<i>bytes</i>	Arreglo de bytes.
--------------	-------------------

**Devuelve**

String equivalente al arreglo de bytes.

Gráfico de llamadas a esta función:

**6.5.2.6. dosLongToBytes()**

```
static byte[] com.example.aplicacion_3ro.Utilidades.dosLongToBytes (
    long masSignificativos,
    long menosSignificativos) [inline], [static]
```

Convierte dos valores long en un arreglo de bytes.

Este método combina dos valores long (más y menos significativos) en un arreglo de bytes.

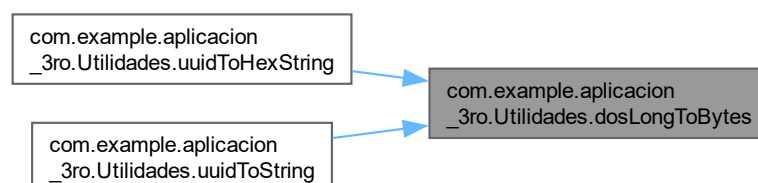
**Parámetros**

<i>masSignificativos</i>	Bits más significativos del UUID.
<i>menosSignificativos</i>	Bits menos significativos del UUID.

**Devuelve**

Un arreglo de bytes que combina los dos long.

Gráfico de llamadas a esta función:

**6.5.2.7. stringToBytes()**

```
static byte[] com.example.aplicacion_3ro.Utilidades.stringToBytes (
    String texto) [inline], [static]
```

Convierte un String a un arreglo de bytes.

**Parámetros**

<i>texto</i>	El texto a convertir.
--------------	-----------------------

**Devuelve**

Un arreglo de bytes que representa el texto.

**6.5.2.8. stringToUUID()**

```
static UUID com.example.aplicacion_3ro.Utilidades.stringToUUID (  
    String uuid) [inline], [static]
```

Convierte un string UUID a un objeto UUID.

El string debe tener una longitud de 16 caracteres.

**Parámetros**

<i>uuid</i>	String con la representación del UUID.
-------------	--

**Devuelve**

El objeto UUID correspondiente.

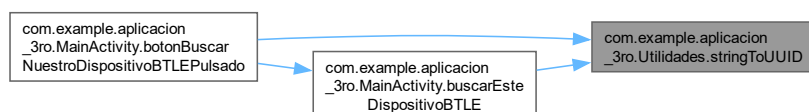
**Excepciones**

<i>Error</i>	Si el string no tiene 16 caracteres.
--------------	--------------------------------------

Gráfico de llamadas de esta función:



Gráfico de llamadas a esta función:



#### 6.5.2.9. uuidToHexString()

```
static String com.example.aplicacion_3ro.Utilidades.uuidToHexString (  
    UUID uuid)    [inline], [static]
```

Convierte un UUID en su representación hexadecimal.

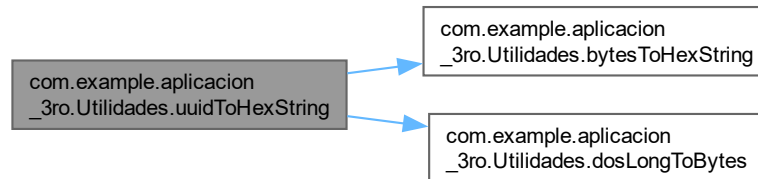
**Parámetros**

<i>uuid</i>	El objeto UUID a convertir.
-------------	-----------------------------

**Devuelve**

Una representación en hexadecimal del UUID como String.

Gráfico de llamadas de esta función:

**6.5.2.10. uuidToString()**

```
static String com.example.aplicacion_3ro.Utilidades.uuidToString (
    UUID uuid) [inline], [static]
```

Convierte un objeto UUID a String.

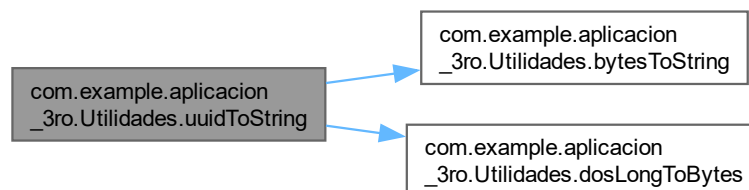
**Parámetros**

<i>uuid</i>	El objeto UUID a convertir.
-------------	-----------------------------

**Devuelve**

Una representación en String del UUID.

Gráfico de llamadas de esta función:



La documentación de esta clase está generada del siguiente archivo:

- [BTLEAlumnos2021/Aplicacion\\_3ro/app/src/main/java/com/example/aplicacion\\_3ro/Utilidades.java](#)





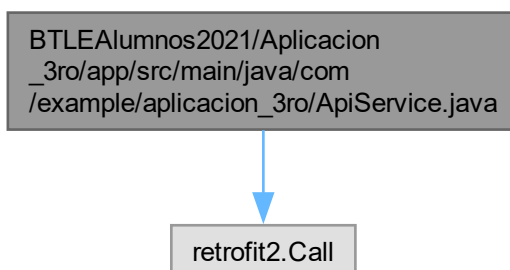
## Capítulo 7

# Documentación de archivos

### 7.1. Referencia del archivo BTLEAlumnos2021/Aplicacion\_3ro/app/src/main/java/com/example/aplicacion\_3ro/ApiService.java

```
import retrofit2.Call;
```

Gráfico de dependencias incluidas en ApiService.java:



#### Clases

- interface `com.example.aplicacion_3ro.ApiService`  
*Interfaz para definir las llamadas HTTP de la API.*

#### Paquetes

- package `com.example.aplicacion_3ro`

## 7.2. Referencia del archivo BTLEAlumnos2021/Aplicacion\_3ro/app/src/main/java/com/example/aplicacion\_3ro/MainActivity.java

```
import android.Manifest;
```

Gráfico de dependencias incluidas en MainActivity.java:



### Clases

- class [com.example.aplicacion\\_3ro.MainActivity](#)

*Clase principal de la aplicación que maneja la interfaz de usuario y la interacción con Bluetooth LE.*

### Paquetes

- package [com.example.aplicacion\\_3ro](#)

## 7.3. Referencia del archivo BTLEAlumnos2021/Aplicacion\_3ro/app/src/main/java/com/example/aplicacion\_3ro/Medicion.java

### Clases

- class [com.example.aplicacion\\_3ro.Medicion](#)

*Clase que representa una medición enviada al servidor.*

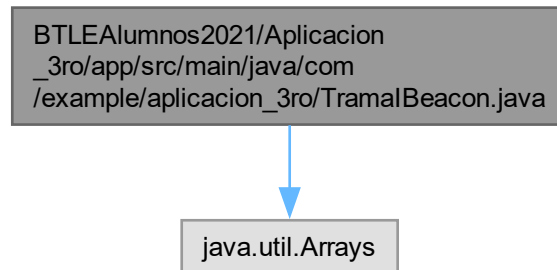
### Paquetes

- package [com.example.aplicacion\\_3ro](#)

## 7.4. Referencia del archivo BTLEAlumnos2021/Aplicacion\_3ro/app/src/main/java/com/example/aplicacion\_3ro/TramalBeacon.java

```
import java.util.Arrays;
```

Gráfico de dependencias incluidas en TramalBeacon.java:



### Clases

- class [com.example.aplicacion\\_3ro.TramalBeacon](#)  
*Clase que representa y procesa una trama iBeacon.*

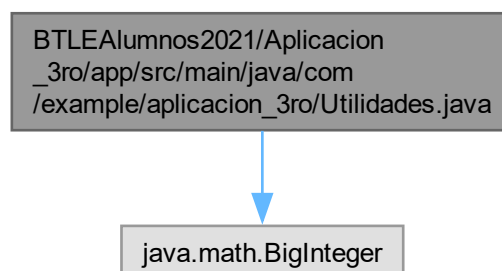
### Paquetes

- package [com.example.aplicacion\\_3ro](#)

## 7.5. Referencia del archivo BTLEAlumnos2021/Aplicacion\_3ro/app/src/main/java/com/example/aplicacion\_3ro/Utilidades.java

```
import java.math.BigInteger;
```

Gráfico de dependencias incluidas en Utilidades.java:



## Clases

- class [com.example.aplicacion\\_3ro.Utilidades](#)

*Clase de utilidades para conversiones de datos y manipulación de UUIDs.*

## Paquetes

- package [com.example.aplicacion\\_3ro](#)

# Índice alfabético

advFlags  
    com.example.aplicacion\_3ro.TramalBeacon, 33  
advHeader  
    com.example.aplicacion\_3ro.TramalBeacon, 33  
apiService  
    com.example.aplicacion\_3ro.MainActivity, 23  
  
botonBuscarDispositivosBTLEPulsado  
    com.example.aplicacion\_3ro.MainActivity, 14  
botonBuscarNuestroDispositivoBTLEPulsado  
    com.example.aplicacion\_3ro.MainActivity, 14  
botonDetenerBusquedaDispositivosBTLEPulsado  
    com.example.aplicacion\_3ro.MainActivity, 15  
BTLEAlumnos2021/Aplicacion\_3ro/app/src/main/java/com/example/aplicacion\_3ro/ApiService.java, 43  
BTLEAlumnos2021/Aplicacion\_3ro/app/src/main/java/com/example/aplicacion\_3ro/MainActivity.java, 44  
BTLEAlumnos2021/Aplicacion\_3ro/app/src/main/java/com/example/aplicacion\_3ro/Medicion.java, 44  
BTLEAlumnos2021/Aplicacion\_3ro/app/src/main/java/com/example/aplicacion\_3ro/TramalBeacon.java, 45  
BTLEAlumnos2021/Aplicacion\_3ro/app/src/main/java/com/example/aplicacion\_3ro/Utilidades.java, 45  
buscarEsteDispositivoBTLE  
    com.example.aplicacion\_3ro.MainActivity, 16  
buscarTodosLosDispositivosBTLE  
    com.example.aplicacion\_3ro.MainActivity, 17  
bytesToHexString  
    com.example.aplicacion\_3ro.Utilidades, 35  
bytesToInt  
    com.example.aplicacion\_3ro.Utilidades, 36  
bytesToIntOK  
    com.example.aplicacion\_3ro.Utilidades, 36  
bytesToLong  
    com.example.aplicacion\_3ro.Utilidades, 37  
bytesToString  
    com.example.aplicacion\_3ro.Utilidades, 37  
  
callbackDelEscaneo  
    com.example.aplicacion\_3ro.MainActivity, 23  
CODIGO\_PETICION\_PERMISOS  
    com.example.aplicacion\_3ro.MainActivity, 23  
com.example.aplicacion\_3ro, 9  
com.example.aplicacion\_3ro.ApiService, 11  
    enviarMediciones, 11  
com.example.aplicacion\_3ro.MainActivity, 12  
    apiService, 23  
    botonBuscarDispositivosBTLEPulsado, 14  
    botonBuscarNuestroDispositivoBTLEPulsado, 14  
    botonDetenerBusquedaDispositivosBTLEPulsado, 15  
    buscarEsteDispositivoBTLE, 16  
    buscarTodosLosDispositivosBTLE, 17  
    callbackDelEscaneo, 23  
    CODIGO\_PETICION\_PERMISOS, 23  
    detenerBusquedaDispositivosBTLE, 18  
    elEscanner, 23  
    enviarDatosAlServidor, 19  
    ETIQUETA\_LOG, 23  
    inicializarBlueTooth, 19  
    mostrarInformacionDispositivoBTLE, 20  
    obtenerTipoDeSensor, 22  
    onCreate, 22  
    onRequestPermissionsResult, 22  
    valor1, 23  
    valor2, 23  
    com.example.aplicacion\_3ro/Medicion.java, 24  
    dato, 26  
    getDato, 29  
    getId\_tipo, 25  
    getId\_usuario, 28  
    id\_tipo, 26  
    id\_usuario, 26  
    Medicion, 25  
    setDato, 25  
    setId\_tipo, 26  
    setId\_usuario, 26  
com.example.aplicacion\_3ro.TramalBeacon, 27  
    advFlags, 33  
    advHeader, 33  
    companyID, 33  
    getAdvFlags, 29  
    getAdvHeader, 29  
    getCompanyID, 29  
    getiBeaconLength, 30  
    getiBeaconType, 30  
    getLosBytes, 30  
    getMajor, 31  
    getMinor, 31  
    getPrefijo, 31  
    getTxPower, 32  
    getUUID, 32  
    iBeaconLength, 33  
    iBeaconType, 33  
    losBytes, 33  
    major, 33  
    minor, 33  
    noadvFlags, 34

- prefijo, 34
- TramalBeacon, 28
- txPower, 34
- uuid, 34
- com.example.aplicacion\_3ro.Utilidades, 34
  - bytesToHexString, 35
  - bytesToInt, 36
  - bytesToIntOK, 36
  - bytesToLong, 37
  - bytesToString, 37
  - dosLongToBytes, 38
  - stringToBytes, 38
  - stringToUUID, 39
  - uuidToHexString, 39
  - uuidToString, 41
- companyId
  - com.example.aplicacion\_3ro.TramalBeacon, 33
- dato
  - com.example.aplicacion\_3ro.Medicion, 26
- detenerBusquedaDispositivosBTLE
  - com.example.aplicacion\_3ro.MainActivity, 18
- dosLongToBytes
  - com.example.aplicacion\_3ro.Utilidades, 38
- elEscanner
  - com.example.aplicacion\_3ro.MainActivity, 23
- enviarDatosAlServidor
  - com.example.aplicacion\_3ro.MainActivity, 19
- enviarMediciones
  - com.example.aplicacion\_3ro.ApiService, 11
- ETIQUETA\_LOG
  - com.example.aplicacion\_3ro.MainActivity, 23
- getAdvFlags
  - com.example.aplicacion\_3ro.TramalBeacon, 29
- getAdvHeader
  - com.example.aplicacion\_3ro.TramalBeacon, 29
- getCompanyId
  - com.example.aplicacion\_3ro.TramalBeacon, 29
- getDato
  - com.example.aplicacion\_3ro.Medicion, 25
- getiBeaconLength
  - com.example.aplicacion\_3ro.TramalBeacon, 30
- getiBeaconType
  - com.example.aplicacion\_3ro.TramalBeacon, 30
- getId\_tipo
  - com.example.aplicacion\_3ro.Medicion, 25
- getId\_usuario
  - com.example.aplicacion\_3ro.Medicion, 25
- getLosBytes
  - com.example.aplicacion\_3ro.TramalBeacon, 30
- getMajor
  - com.example.aplicacion\_3ro.TramalBeacon, 31
- getMinor
  - com.example.aplicacion\_3ro.TramalBeacon, 31
- getPrefijo
  - com.example.aplicacion\_3ro.TramalBeacon, 31
- getTxPower
  - com.example.aplicacion\_3ro.TramalBeacon, 32
  - com.example.aplicacion\_3ro.TramalBeacon, 32
- iBeaconLength
  - com.example.aplicacion\_3ro.TramalBeacon, 33
- iBeaconType
  - com.example.aplicacion\_3ro.TramalBeacon, 33
- id\_tipo
  - com.example.aplicacion\_3ro.Medicion, 26
- id\_usuario
  - com.example.aplicacion\_3ro.Medicion, 26
- inicializarBlueTooth
  - com.example.aplicacion\_3ro.MainActivity, 19
- losBytes
  - com.example.aplicacion\_3ro.TramalBeacon, 33
- major
  - com.example.aplicacion\_3ro.TramalBeacon, 33
- Medicion
  - com.example.aplicacion\_3ro.Medicion, 25
- minor
  - com.example.aplicacion\_3ro.TramalBeacon, 33
- mostrarInformacionDispositivoBTLE
  - com.example.aplicacion\_3ro.MainActivity, 20
- noadvFlags
  - com.example.aplicacion\_3ro.TramalBeacon, 34
- obtenerTipoDeSensor
  - com.example.aplicacion\_3ro.MainActivity, 22
- onCreate
  - com.example.aplicacion\_3ro.MainActivity, 22
- onRequestPermissionsResult
  - com.example.aplicacion\_3ro.MainActivity, 22
- prefijo
  - com.example.aplicacion\_3ro.TramalBeacon, 34
- setData
  - com.example.aplicacion\_3ro.Medicion, 25
- setId\_tipo
  - com.example.aplicacion\_3ro.Medicion, 26
- setId\_usuario
  - com.example.aplicacion\_3ro.Medicion, 26
- stringToBytes
  - com.example.aplicacion\_3ro.Utilidades, 38
- stringToUUID
  - com.example.aplicacion\_3ro.Utilidades, 39
- TramalBeacon
  - com.example.aplicacion\_3ro.TramalBeacon, 28
- txPower
  - com.example.aplicacion\_3ro.TramalBeacon, 34
- uuid
  - com.example.aplicacion\_3ro.TramalBeacon, 34
- uuidToHexString
  - com.example.aplicacion\_3ro.Utilidades, 39

uuidToString

com.example.aplicacion\_3ro.Utilidades, [41](#)

valor1

com.example.aplicacion\_3ro.MainActivity, [23](#)

valor2

com.example.aplicacion\_3ro.MainActivity, [23](#)