

# Prueba de habilidades Bosch: AR Tools & Products

Gustavo Adolfo Bonilla González

Desarrollador de Software

Centro de Enseñanza Técnica Industrial

[gabg\\_97@outlook.com](mailto:gabg_97@outlook.com)

**Resumen-** Este documento presenta la documentación del desarrollo de la prueba AR Tools & Products, teniendo como objetivo el desarrollar una aplicación móvil que permita mostrar modelos 3D de productos de la tienda en línea de Bosch mediante el uso de realidad aumentada. Para cumplir dicho objetivo se deberán cumplir 4 objetivos principales: utilizar un endpoint previamente dado para realizar peticiones web a la tienda en línea de Bosch, mantener los datos obtenidos del endpoint en un archivo para su posterior uso, mostrar un menú con los productos disponibles y emplear realidad aumentada para mostrar el modelo 3D del producto.

## I. INTRODUCCIÓN

Para el desarrollo de este proyecto se optó por realizarlo en Android, por lo que este documento estará orientado a una aplicación móvil Android, para poder realizar este proyecto serán necesarias algunas librerías y herramientas: para trabajar con el endpoint e interactuar con el servidor de la tienda en línea de Bosch será necesario utilizar Retrofit (un cliente HTTP para Android y Java), para implementar la persistencia de datos se utilizará SQLite (un sistema de gestión de bases de datos mediante archivos locales), como menú principal de la aplicación se creará un Navigation Drawer (consiste en un menú lateral deslizable muy fácil de implementar) y finalmente se empleará AR Core (es la plataforma de Google para crear experiencias de realidad aumentada) con ayuda del framework Sceneform (este permite el fácil manejo de la creación de sesiones y de los elementos comunes de “User Experience” de AR Core) para trabajar la realidad aumentada.

## II. DESARROLLO

Primeramente, para desarrollar este proyecto es necesario crear un nuevo proyecto, en el cual se colocarán todas las clases, recursos y layouts necesarios. Es preciso asegurar que en el manifest de la aplicación se provean los permisos necesarios para el correcto funcionamiento de la aplicación, en este caso serán requeridos los permisos de uso de la cámara del dispositivo y el uso de internet.

### II-I. MENÚ Y VISTAS DE LA APLICACIÓN

Lo primero que debe ser visto a la hora de abrir la aplicación es la splash screen de la aplicación, esta tiene el logo de la app, se despliega en pantalla completa y tiene una duración de 2.5 segundos.

A continuación, deberá aparecer la vista principal del Navigation Drawer, el cual contará con un menú desplegable con acceso a las vistas Products (vista principal), Bosch Web Store, About Us y Help como los cuales serán fragments.

La vista Products tendrá botones con los nombres de los productos principales y al momento de interactuar con alguno de estos botones se mostrará un panel con detalles del producto seleccionado (ID, Price, Name, Description y una foto del producto) y un botón que permitirá acceder al modo de realidad aumentada al interactuar con él.

Dicho botón mostrará las imágenes captadas por la cámara del dispositivo y al tocar la pantalla en un área válida mostrará el modelo 3D del producto seleccionado.

Las imágenes de las capturas de pantalla de las vistas mencionadas se encuentran en la carpeta “Capturas” de esta documentación.

### II-II. ACCESO A LOS SERVICIOS WEB

Lamentablemente este objetivo del proyecto no fue conseguido, debido a que no llegué a entender del todo el uso de Retrofit y de los datos proporcionados para el endpoint, se efectuaron pruebas e intentos fallidos para realizar una autenticación básica y para obtener los datos productos de la página (de los cuales no se tenían las key de los campos en el archivo json), se adjunta parte del código de pruebas con Retrofit en la carpeta de este documento.

### II-III. PERSISTENCIA DE LOS DATOS

La persistencia de los datos se efectuó con el uso de SQLite, se implementó un método que se encarga de proveer y guardar información de los productos (para sustituir la etapa previa), este método se encuentra en la clase “NavigationMenu.java” y no recibe ni retorna parámetros.

```
private void toolsRecord();
```

El método anterior utiliza la clase “SQLiteConectionHelper.java”, la cual se encarga de manejar los archivos generados para la base de datos con dos métodos que reciben parámetros de características generales de la base de datos y no retornan parámetros.

```
public void onCreate(SQLiteDatabase db);  
public void onUpgrade(SQLiteDatabase db, int  
oldVersion, int newVersion);
```

También es necesario crear una clase que se encargue de guardar los atributos de cada elemento en la base de datos con sus respectivos setters y getters, esta clase es: “Tool.java” y se encuentra en el directorio “sqliteEntities”.

El método que se encarga de recuperar información del archivo generado por SQLite se encuentra en la clase “NavigationMenu.java”, recibe el parámetro ID (el cual es el ID del producto a buscar), guarda los atributos del producto encontrado en la clase “Utilities.java” (se encuentra en el directorio “utilities”), no retorna parámetros y es el método:

```
public void consultSQLite(String id);
```

## II-IV. IMPLEMENTACION DE AR

Para desarrollar esta etapa se utilizaron AR Core y el framework Sceneform, todos los métodos necesarios para trabajar esta etapa se encuentran en la clase “AR\_SceneformActivity.java”.

El método:

```
public static boolean  
checkIsSupportedDeviceOrFinish(final Activity  
activity);
```

Se encarga de revisar si el dispositivo en el que está instalada la aplicación es compatible para trabajar con AR Core, retorna un valor booleano (dependiendo de si el dispositivo es compatible o no) y recibe el activity actual.

El método:

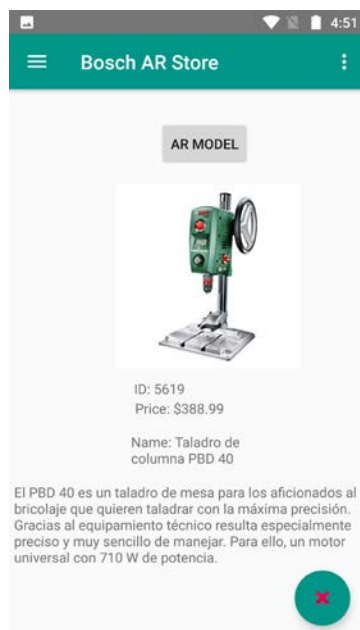
```
public void goBack(View v);
```

Simplemente se encarga de salir del modo de realidad aumentada y regresar a la vista principal de “Products”, recibe un parámetro tipo “View” y no retorna ningún parámetro.

El resto de configuraciones importantes para AR Core se realizan en el “onCreate” de la clase, primeramente manda a llamar al método que evalúa si el dispositivo es compatible, después muestra la vista “activity\_ux” que muestra lo que captura la cámara y al botón “Go Back”, crea el modelo renderizable del modelo 3D del producto, cuando el usuario toque la pantalla en una parte que se detecte como plana se crearan los nodos para colocar e interactuar (mediante gestos) con el modelo del producto.

## III. RESULTADOS

Los resultados finales obtenidos son bastante satisfactorios (con la excepción de la implementación de Retrofit), la aplicación final recupera los datos del producto de un archivo SQLite y te los muestra dependiendo del producto seleccionado, ya en esa sección de detalles del producto se puede acceder al modo de realidad aumentada que muestra un modelo 3D del producto seleccionado al momento de tocar la pantalla en un área detectada como un plano, aparte se tiene la posibilidad de interactuar con el modelo 3D mediante gestos (mover de lugar, cambiar su tamaño y rotar).



## REFERENCIAS

- [1] Android Developers. (2019). *Room Persistence Library* / *Android Developers*. [online] Available at: <https://developer.android.com/topic/libraries/architecture/room> [Accessed 16 Apr. 2019].
- [2] Google. (2019). Quickstart for Android [online] Available at: <https://developers.google.com/ar/develop/java/quickstart> [Accessed 16 Apr. 2019].