

Integration of a 3D scanner with ultrasonic sensors and a mobile application for visualization of three-dimensional models

Integración de un escáner 3D con sensores ultrasónicos y una aplicación móvil para visualización de modelos tridimensionales

Axel Issai Alemán-Delgado¹, Isaac De Jesús Becerra-Martínez¹, Karen Lizeth García-Rivera¹, Daniela Estefanía Huerta-Vázquez¹, Didre Dayan Mota-González¹, Carlos Alejandro Reyes-Puga¹, and Marco Aurelio Nuño-Maganda^{*1}

¹Universidad Politécnica de Victoria (UPV), Ingeniería en Tecnologías de la Información,
Av. Nuevas Tecnologías 5902, Parque Científico y Tecnológico de Tamaulipas, 87138 Cd Victoria, Tamps.
{1930204, 1930062, 1930468, 1930474, 1830419, 1930033, mnunom}@upv.edu.mx

Abstract

3D printing is a recent process that has revolutionized different areas of everyday life. Tasks traditionally performed by computers are available on smartphones. In this article, we propose a mobile application with a Bluetooth interface that allows obtaining a three-dimensional model of a 3D scanner made with common elements, such as an AT-Mega 2560 and the HC-SR04 ultrasonic sensor, in addition to the motors required for the movement of the rotating base and the ultrasonic sensor. The application allows to the user to visualize the model through the OpenGL ES library, and to control the scanning process. We report a working prototype that allows to the user to generate and visualize 3D models of simple objects.

Keywords— Mobile App, OpenGL ES, Tridimensional Scanner

Resumen

La impresión 3D es un proceso reciente que ha revolucionado diferentes ámbitos de la vida cotidiana. El teléfono inteligente es una herramienta común en estos días, y muchas de las tareas que se llevaban a cabo en una computadora ahora es posible llevarla a cabo en este dispositivo. En este artículo, proponemos una aplicación móvil que mediante comunicación serial utilizando una interfaz Bluetooth, permita

obtener el modelo tridimensional de un escáner 3D hecho con elementos comunes, como son un microcontrolador AT-Mega 2560 integrado en Arduino UNO y el sensor ultrasónico HC-SR04, además de los motores requeridos para el movimiento de la base giratoria y del sensor ultrasónico. La aplicación permite al usuario visualizar el modelo, mediante el uso de la librería OpenGL ES, además de controlar los detalles del proceso de escaneo. Se reporta un prototipo funcional que permite al usuario generar y visualizar modelos 3D de objetos simples.

Palabras clave— Aplicación Móvil, OpenGL ES, Escáner tridimensional

I. Introducción

La digitalización de objetos es posible gracias al escaneo 3D debido a su gran utilidad y precisión, ha revolucionado diversos sectores industriales, educativos, de construcción y de manufactura.

Este proyecto surgió de la necesidad de replicar un objeto complejo en dimensiones más grandes a las originales. Debido a que obtenerlas físicamente tenía una gran complejidad, se realizó este escáner 3D.

Un escáner 3D [1] es un dispositivo de imagen que recoge medidas de puntos de distancia de un objeto del mundo real y las traduce en un objeto 3D virtual. Los escáneres 3D son utilizados para crear imágenes y animaciones realistas, e incluso llegar a recrear objetos físicos

* Autor de correspondencia

mediante los datos obtenidos.

En el presente artículo se muestran los resultados de implementar un escáner 3D utilizando componentes de bajo costo, el cual tiene la capacidad de escanear cualquier objeto y proporcionar sus coordenadas cartesianas en los ejes X, Y y Z.

El artículo se organiza de la siguiente manera: en la Sección II se realiza una comparación de aplicaciones existentes que tienen funciones similares a la que se hace referencia en este artículo. En la Sección III se explica a detalle lo que es el sistema propuesto, se describe el hardware y software utilizados en la implementación del proyecto así como el proceso de creación de este. Finalmente, en las Secciones IV y V se muestran los resultados obtenidos y las conclusiones de las pruebas realizadas.

II. Revisión de la literatura

En esta sección revisaremos algunos de los sistemas de escaneo de objetos 3D que emplean componentes de bajo costo, de código abierto y que generalmente incluyen instrucciones de armado relativamente sencillas para que cualquier usuario pueda poner a funcionar dicha aplicación.

II.1. AAScan

El proyecto AAScan [2] es una propuesta de escaner que tiene como plataforma de captura de imágenes de entrada a un teléfono inteligente, como plataforma de procesamiento a una computadora que ejecuta un script en Python utilizando QPython, además del uso de un Arduino para el control de los motores de la plataforma. También requiere de una computadora con un GPU Nvidia. El proyecto incluye una lista de componentes en donde todos son de bajo costo si se excluye la computadora con tarjeta gráfica Nvidia. En la Figura 1 se muestran los componentes de este sistema.

II.2. FabScan

El proyecto FabScan [3] es un escáner láser 3D que requiere de una computadora Raspberry Pi y un módulo de cámara Raspberry Pi para la adquisición de las imágenes requeridas para el escaneo 3D. Otros componentes de este sistema son los motores a paso, un cubo de montaje y otros componentes electrónicos requeridos para la base de escaneo. En la Figura 2, se muestran los componentes del proyecto.

II.3. DIY Standalone 3D Scanner

El DIY Standalone 3D Scanner permite que al combinar una placa Mbed con una cámara y bibliotecas OpenCV, el proceso de escaneo se automatiza en gran medida

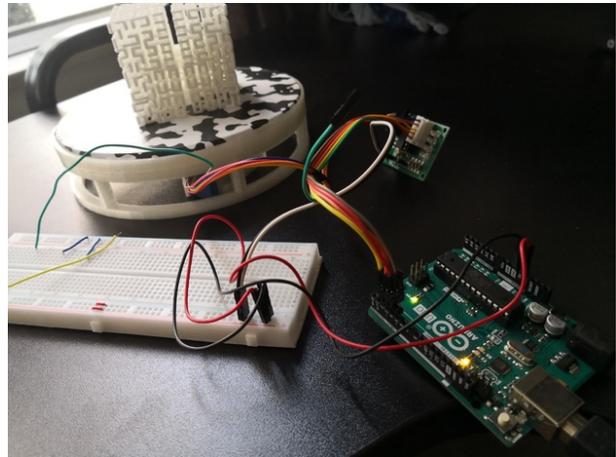


Figura 1: Componentes utilizados para el proyecto AAScan [2]



Figura 2: Material necesario para FabScan [3]

con solo presionar un botón. El escáner captura múltiples imágenes de un objeto para crear un modelo 3D que luego se envía como un archivo STL escrito en una tarjeta SD. Su unidad de control está basada en un dispositivo GR-LYCHEE, además de piezas electrónicas más pequeñas, láminas de plástico para crear la carcasa y varias tuercas y cableado para ensamblarlo todo [4].

II.4. OpenScan

El proyecto OpenScan [5] es un escáner que permite a partir de fotografías desde diferentes ángulos o de un video capturado en vivo, generar un modelo 3D que puede ser editado y modificado para propósitos de impresión en 3D. El proyecto tiene la flexibilidad en cuanto a la unidad de control, que puede ser un dispositivo Arduino o Raspberry Pi, la cual puede ser controlada a distancia mediante comunicación Bluetooth usando un teléfono inteligente.

II.5. Silhouette standalone 3D scanner

Este escáner 3D se basa en la placa y la cámara de prototipos GR-Lychee y utiliza un algoritmo que analiza siluetas para obtener un modelo 3D. La idea detrás de este algoritmo es tomar fotografías de la silueta de un objeto desde varios ángulos. Luego, las fotos se convierten en datos de nubes de puntos que representan la forma 3D escaneada. Finalmente, se crea una malla poligonal a partir de esos datos y se guarda en una tarjeta SD como un archivo STL [4]. El sistema arroja como resultado final un modelo 3D con una resolución de escaneo de un milímetro, listo para ser utilizado con cualquier dispositivo de impresión 3D. Para su construcción se requiere un tablero GR-Lychee y algunas láminas de plástico.

La principal desventaja que se desea atacar es substituir con un sensor ultrasónico aquellos sistemas que hagan uso de una cámara, debido a que este sensor podría proveer información mucho más precisa de las dimensiones del objeto en comparación con una cámara. La aportación de este proyecto consiste en agregar una aplicación móvil que permita al usuario llevar a cabo el escaneo tridimensional de un objeto simple sin requerir de una computadora.

III. Sistema Propuesto

III.1. Aplicación móvil

En este proyecto se siguió el paradigma de prototipado propuesto por Pressmann [6]. La creación de prototipos es un enfoque de desarrollo de software en el que la parte interesada (en este caso, específicamente los usuarios, que son los estudiantes usuarios de dicha aplicación) define los objetivos generales para el proyecto, pero no clasifica a fondo los requisitos para los detalles. Esto llevará a los desarrolladores (el equipo de estudiantes que implementaron tanto la aplicación móvil como el prototipo del escáner) a utilizar el paradigma de la creación de prototipos, en el que el prototipo inicial se presenta a las partes interesadas. Así las partes interesadas pueden comentar si el proyecto se ajusta a sus necesidades o no.

Android es un sistema operativo móvil diseñado para su uso en teléfonos y tabletas inteligentes [7]. De manera que dentro de este sistema operativo pueden ejecutarse aplicaciones móviles nativas, permitiendo un alto rendimiento y una experiencia de interfaz de usuario del dispositivo [8].

Para el desarrollo de aplicaciones específicas para el entorno Android, existen diferentes lenguajes de programación que son empleados para la construcción de dichas aplicaciones, como lo son Java y Kotlin [9, 10], así como entornos de desarrollo (IDE, por sus siglas en inglés) que proveen herramientas para la generación de aplicaciones, realizando un compilado rápido y unificado para Android

[11].

Debido a ello, la configuración del ambiente de desarrollo de la aplicación móvil propuesta fue considerando los siguientes aspectos técnicos:

- Codificación en el lenguaje de programación Java.
- Versión de Android SDK mínima: 21 Lollipop (Android 5.0).
- IDE Android Studio versión Bumblebee 2021.1.1.
- Dispositivos para la ejecución de pruebas técnicas Huawei P30 Pro, Oppo A53 y Xiaomi Poco F2 Pro.
- Librería OpenGL ES 2.0 para gráficos de alto rendimiento.

Diseño de aplicación

En la elaboración de la aplicación móvil se contemplan dos módulos que permiten el funcionamiento de la aplicación para la conexión y recepción de los datos del escáner desarrollado, así como el modelado de los datos recibidos. Los módulos se describen a continuación:

- **Módulo de conexión con escáner 3D:** En este apartado se realiza la configuración previa a la recepción de los datos por parte del escáner propuesto, de manera que se realiza la visualización de los dispositivos cercanos disponibles para su conexión por medio de Bluetooth. Permitiendo la selección de un dispositivo que funcionará como receptor de datos periódicamente.
- **Módulo de visualización de datos:** Se posibilita la generación de un modelo 3D con los datos recibidos del dispositivo remoto previamente configurado, para ello se puede autorizar el inicio del modelado para iniciar el proceso de recepción de los datos y su muestreo en tiempo real de manera gráfica.

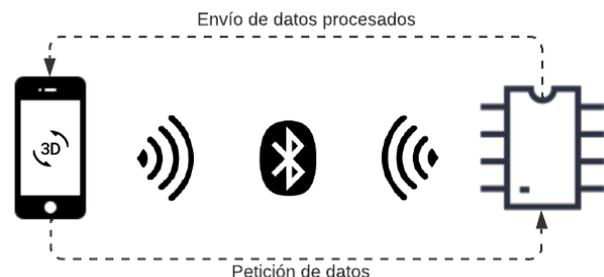


Figura 3: Proceso de inicio de conexión P2P Bluetooth con dispositivo y recepción de los datos

Transferencia de información por Bluetooth entre placa Arduino y dispositivo Android

Para la conexión de manera remota con el escáner 3D, se implementa la API de Bluetooth de Android [12].

Estableciendo la búsqueda de dispositivos para realizar la conexión P2P (siglas en inglés de *Peer to Peer*) bajo el protocolo RFCOMM y el recibimiento de los datos entre los dispositivos [13].

Del mismo modo, se emplea el siguiente diagrama de flujo en la Figura 3 donde se muestra el proceso de comunicación para la conexión de la aplicación móvil con el escáner 3D y la recepción de los datos.

Generación de modelo 3D

Para la implementación de un modelo 3D de datos en Android se utilizó el motor de gráficos integrados OpenGL ES, con el objetivo de realizar la visualización de los datos de manera nativa en el dispositivo, empleando la API proporcionada por Android [14].

En la elaboración del modelo 3D se contemplan las coordenadas de plano cartesiano en tres dimensiones, como lo son los ejes X, Y, Z [15]. Permitiendo la representación de puntos mediante un conjunto de valores que posibilitan la construcción de una figura tridimensional y los puntos que la conforman.

Para ello, se realiza la recepción activa con el dispositivo conectado por medio de Bluetooth de los datos que representan las coordenadas del punto a graficar considerando los ejes de coordenadas mencionados, realizando la actualización de la figura en tiempo real conforme los datos sean recibidos.

Eliminación de ruido del modelo generado

Conforme los datos son recibidos por el controlador AT-Mega 2650 y el sensor ultrasónico RS04, se observó que las coordenadas del objeto escaneado necesitan ser escaladas al rango permitido por OpenGL para la construcción de modelos, los cuales son mostrados en la Tabla 1 con su respectivo eje.

Con respecto a la reducción de ruido de los datos provenientes del sensor ultrasónico, se utilizó una técnica básica que consiste estimar un valor promedio de las lecturas para minimizar el error de falsos valores entre las lecturas obtenidas.

Tabla 1: Rango de valores soportados en el mapa cartesiano de OpenGL ES

Eje Cartesiano	Rango mínimo	Rango máximo
X	-1	1
Y	-1	1
Z	-1	1

Fue necesario realizar la conversión de los datos utilizando la Ecuación (1).

$$v = \left(x * \frac{(b_2 - b_1)}{(a_2 - a_1)} \right) + \frac{(b_1 a_2 - b_2 a_1)}{(a_2 - a_1)} \quad (1)$$

donde :

- x es el valor original a transformar.
- $[a_1, a_2]$ es el rango original mínimo y máximo de x .
- $[b_1, b_2]$: es el nuevo rango máximo y mínimo hacia escalar.
- v es el valor obtenido del dato original transformado al nuevo rango objetivo.

De esta manera, los puntos escaneados por el sensor ultrasónico son transformados a la nueva escala, con el fin de implementarlos en la graficación del modelado en 3D.

III.2. Materiales

Los principales componentes de hardware empleados para el desarrollo del escáner 3D son los siguientes:

- Placa controladora AT-Mega 2560.
- Sensor ultrasónico - HC-SR04.
- Micro-servo motor - SG90.
- Motor 3-6V.
- Texas Instruments Dual H-Bridge motor drivers L293D.
- Módulo de alimentación.
- 9V 1A Switching Wall Power Supply.
- Módulo Bluetooth HC-06.

Los principales componentes de software empleados para la programación tanto de la aplicación móvil como el Arduino para la adquisición de datos de los sensores son:

- Arduino IDE.
- Android Studio.

Circuito para control de los motores de la plataforma

En la Figura 4 se muestra el circuito utilizado, así como la conexión de los componentes al controlador AT-Mega 2560.

Para la realización del escáner 3D se realizó el siguiente orden de construcción:

1. Selección de componentes a utilizar.
2. Conexión de placa AT-Mega 2560 con protoboard de 830 puntos.
3. Conexión de sensor ultrasónico HC-SR04.
4. Conexión de servomotor SG90.
5. Conexión de motor 3-6V.
6. Adición del módulo Bluetooth HC-06.
7. Implementación del módulo de energía.

De manera que el escáner resultante se puede observar en la Figura 5.

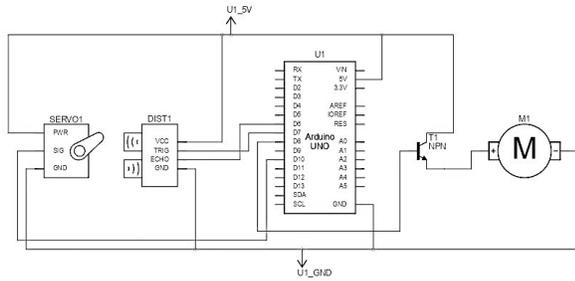


Figura 4: Esquema del circuito para controlar los motores de la plataforma del escáner 3D, incluyendo la conexión al dispositivo Arduino

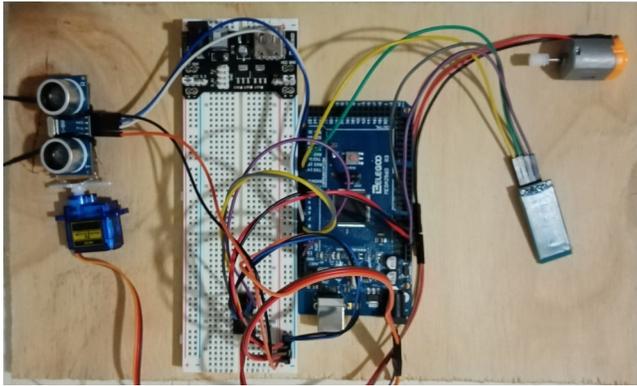


Figura 5: Circuito Arduino

IV. Resultados

Al acceder a la aplicación, en caso de ser necesario, se solicita activar la funcionalidad de Bluetooth en el dispositivo móvil (Figura 6), esto con el fin de poder establecer una comunicación con el módulo Bluetooth conectado a la placa Arduino. Adicionalmente, es necesario mantener activa la funcionalidad de Ubicación, ya que es un requisito para poder listar los dispositivos Bluetooth vinculados.

Una vez activado el adaptador Bluetooth, se muestra un listado de los dispositivos que se encuentran disponibles en ese momento (Figura 7), es decir, que se encuentran al alcance de nuestro dispositivo y con los cuales puede establecer una comunicación.

A continuación, identificamos el SSID del módulo Bluetooth conectado a la placa Arduino, damos clic en el botón, esto arrojará una ventana de confirmación (Figura 8) en la cual daremos clic en “Conectar” para establecer la comunicación entre los dispositivos.

Hecho esto, la aplicación enviará una señal a la placa Arduino para iniciar el escaneo del objeto. Las coordenadas obtenidas por el sensor ultrasónico serán dibujadas en tiempo real en la aplicación, ver Figura 9.

Las pruebas de escaneo se realizaron utilizando un

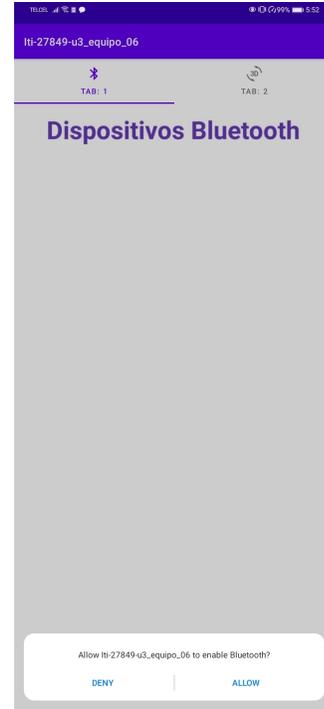


Figura 6: Solicitud de encendido de Bluetooth



Figura 7: Listado de dispositivos Bluetooth disponibles

vaso como modelo (Figura 10). Dicho objeto tiene una forma de cilindro.

Posteriormente, se agregó la funcionalidad de cambiar el color de los puntos a graficar de la figura escaneada, esto se realiza a través de la clasificación de sus valores en cada punto en el eje coordenado Z con base en las siguientes condiciones:

- El punto se grafica de color rojo si el valor del eje Z es negativo $Z < 0$.
- El punto se grafica de color negro en caso que el valor del eje coordenado Z sea positivo $Z \geq 0$.

En la Figura 11 se muestra la salida de la aplicación después de aplicar los cambios mencionados anteriormente. Como se logra observar, los puntos superiores sufren un cambio en su tonalidad de color, mostrándolos de un color rojizo.

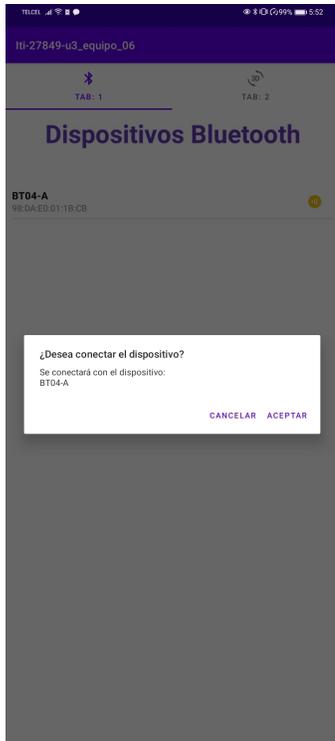


Figura 8: Confirmación de conexión



Figura 10: Objeto escaneado



Figura 9: Generación de puntos a partir de coordenadas obtenidas por el sensor

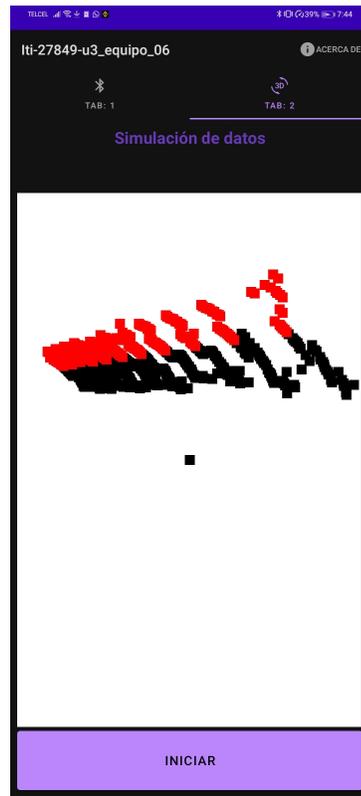


Figura 11: Generación de puntos a partir de coordenadas obtenidas por el sensor con cambio de color



Figura 12: Segundo objeto de prueba

El objeto escaneado para esta prueba fue un cilindro de cartón, mismo que puede observarse en la Figura 12.

V. Conclusiones

Conforme al sistema propuesto para la realización del escáner 3D, construido mediante el controlador AT-Mega 2560 y el sensor ultrasónico HC SR04, se permitió llevar a cabo la digitalización de distintos objetos, determinando una alta capacidad en la ejecución del dispositivo. Así mismo, en el desarrollo de la aplicación móvil planteada para Android se observa la comunicación P2P Bluetooth con el dispositivo correspondiente para el inicio y recepción de los datos de manera constante, de la misma forma, se implementó de manera correcta los gráficos de alto rendimiento OpenGL ES 2.0, permitiendo obtener un modelo 3D graficando los datos recibidos y visualizando un esquema similar conforme a la figura escaneada.

Algunas limitaciones que presenta el trabajo desarrollado consiste en que son mejorables tanto la aplicación móvil en cuanto a la construcción y despliegado del modelo 3D, así como la fase de adquisición de los datos por parte del sensor ultrasónico, en donde hay algunos problemas debido a la velocidad de operación y de manejo de ruido.

Como trabajo futuro, se propone realizar una fusión de los datos obtenidos mediante el sensor ultrasónico con imágenes adquiridas por la cámara del teléfono inteligente, con la finalidad de mejorar el desempeño del sistema.

Además, se propone mejorar el resultado de escaneo utilizando un sensor ultrasónico de mayor fiabilidad, y sustituir los motores que controlan el movimiento de la base por aquellos que permitan un escaneo mucho más preciso del objeto de interés a ser escaneado.

Otro punto importante a desarrollar para mejorar el trabajo es efectuar una comparativa de la calidad de la imagen vs distancia, ya que los resultados preliminares sugieren que esto es mejorable en base a lo obtenido para las pruebas realizadas.

Referencias

- [1] 3-D scanner. <https://www.techtarget.com/whatis/definition/3-D-scanner>. Consultado el 07-02-2023.
- [2] AAScan: Open source, minimalist, fully automated 3D scanner based on Arduino and Android! <https://www.thingiverse.com/thing:4167615>. Consultado el 07-02-2023.
- [3] FabScanPi's documentation. <https://fabscanpi-server.readthedocs.io/en/latest/>. Consultado el 07-02-2023.
- [4] DIY Standalone 3D Scanner. <https://www.instructables.com/DIY-Standalone-3D-Scanner/>. Consultado el 07-02-2023.
- [5] OpenScan. <https://en.openscan.eu/>. Consultado el 07-02-2023.
- [6] Roger Pressman. *Software Engineering: A Practitioner's Approach*. 7.^a ed. USA: McGraw-Hill, Inc., 2009. ISBN: 0073375977.
- [7] Qué es Android. https://www.android.com/intl/es-419_mx/what-is-android/. Consultado el 07-02-2023.
- [8] What is a native app? <https://magenest.com/en/native-app/>. Consultado el 07-02-2023.
- [9] Kotlin Programming Language. <https://kotlinlang.org/>. Consultado el 07-02-2023.
- [10] Java Language Basics. <https://dev.java/learn/java-language-basics/>. Consultado el 07-02-2023.
- [11] android. https://www.android.com/intl/es-419_mx/. Consultado el 07-02-2023.
- [12] Introducción general a Bluetooth. <https://developer.android.com/guide/topics/connectivity/bluetooth?hl=es-419>. Consultado el 07-02-2023.
- [13] Bluetooth RFCOMM. <https://docs.microsoft.com/en-us/windows/uwp/devices-sensors/send-or-receive-files-with-rfcomm>. Consultado el 07-02-2023.
- [14] OpenGL. <https://www.opengl.org/>. Consultado el 07-02-2023.
- [15] 3D Coordinate System – Definition, Graphing Techniques, and Examples. <https://www.storyofmathematics.com/3d-coordinate-system/>. Consultado el 07-02-2023.