



**UNIVERSIDAD DE CONCEPCIÓN
FACULTAD DE INGENIERÍA
DEPARTAMENTO DE INGENIERÍA ELÉCTRICA**



**SISTEMA DE NAVEGACIÓN INTRAHOSPITALARIA CON DISPOSITIVO MÓVIL
COMO GUÍA PARA PACIENTES.**

POR

Valentina Teresa León Molina

Memoria de Título presentada a la Facultad de Ingeniería de la Universidad de Concepción para
optar al título profesional de Ingeniero Civil Biomédico

Profesor(es) Guía:
D. Sc. Esteban Pino Quiroga

Julio 2024
Concepción (Chile)

© 2024 Valentina Teresa León Molina

© 2024 Valentina León Molina

Se autoriza la reproducción total o parcial, con fines académicos, por cualquier medio o procedimiento, incluyendo la cita bibliográfica del documento.



Resumen

La falta de orientación en entornos hospitalarios dificulta que los pacientes logren ubicar lugares dentro de las instalaciones, lo cual resulta en una pérdida de tiempo para los pacientes y el personal de salud.

En el presente informe se propone un sistema de posicionamiento intrahospitalario mediante una aplicación móvil para la Unidad de Imagenología del Hospital Clínico Regional de Concepción (HGGB) que busca dar una solución a esta problemática.

Se realiza una búsqueda bibliográfica con el fin de establecer un marco teórico que contenga la información conceptual necesaria para idear y proponer un sistema de localización intrahospitalario integrado con una aplicación móvil y un mapa interactivo mediante AR que utiliza códigos QR posicionados en los puntos de acceso del establecimiento. La solución permite a pacientes llegar a salas de manera eficiente, utilizando realidad aumentada para guiarlos en tiempo real a través de la cámara del celular.

La aplicación móvil fue desarrollada en Unity utilizando el SDK de Vuforia para AR y el *Package* AI Navigation como componentes principales. Luego, fue evaluada mediante pruebas de funcionamiento dentro del hospital y en el edificio de Sistemas de la Universidad de Concepción comprobando el desempeño de la interfaz de usuario y la correcta ejecución del sistema de navegación.

Como resultado, se logró desarrollar un sistema accesible, seguro y de bajo costo que otorga una alternativa viable para guiar al paciente hacia algún lugar de interés dentro del hospital sin la necesidad de tener acceso a datos personales o a señal telefónica. Sin embargo, aún se pueden seguir haciendo mejoras para otorgarle a los usuarios una aplicación móvil más completa especialmente al considerar no siempre tiene un funcionamiento correcto.

Abstract

Lack of orientation in the hospital environment makes it difficult for patients to find their way around the facility, resulting in wasted time for patients and staff.

This report proposes an intrahospital positioning system through a mobile application for the Imaging department of the Hospital Clínico Regional de Concepción (HGGB), which seeks to provide a solution to this problem.

A bibliographical search is carried out in order to establish a theoretical framework with the necessary conceptual information to devise and propose an intrahospital positioning system integrated with a mobile application and an interactive map using AR and QR codes placed at the facility's access points. The solution will allow patients to reach wards efficiently, using augmented reality to guide them in real time through the mobile phone camera.

The mobile application was developed in Unity using the Vuforia SDK for AR and the AI Navigation package as main components. Then, it was evaluated through functional tests in the hospital and in the Systems Building of the Universidad de Concepción, checking the performance of the user interface and the correct execution of the navigation system.

As a result, a safe, accessible, and low-cost system was developed, providing a viable alternative to guide patients to a point of interest within the hospital without the need for access to personal data or a phone signal. However, further improvements can still be made to provide users with a more complete mobile application, especially considering that it does not always function correctly.

Tabla de Contenidos

Capítulo 1. Introducción	1
1.1. Introducción General	1
1.2. Objetivos	2
1.2.1 Objetivo General	2
1.2.2 Objetivos Específicos	2
1.3. Alcances y Limitaciones	2
1.4. Metodología	3
1.5. Temario	3
Capítulo 2. Marco Teórico	4
2.1 Introducción	4
2.2 Los Servicios Basados en la Localización (LBS)	4
2.3 Principios de posicionamiento	4
2.3.1 Indicador de Fuerza de Señal Recibida (RSSI)	4
2.3.2 Trilateración	5
2.3.3 Triangulación	6
2.3.3.1 Tiempo de vuelo (ToF/ToA)	6
2.3.3.2 Diferencia de Tiempo de Llegada (TDoA)	7
2.3.3.3 Tiempo de Vuelo de Retorno (RToF)	8
2.3.3.4 Ángulo de Llegada AoA	9
2.3.4 Proximidad	9
2.3.5 Análisis de escena	9
2.4 Sistemas de Posicionamiento Interno (IPS)	10
2.4.1 Wi-Fi	10
2.4.2 Dispositivo de Identificación por Radiofrecuencia (RFID)	10
2.4.3 Bluetooth	11
2.4.4 Ultrasonido	11
2.4.5 Banda Ultraancha (UWB)	11
2.4.6 Pedestrian Dead Reckoning (PDR)	12
2.4.7 Otros	12
2.5 Seguridad y privacidad en IPS	13

	II
2.6 IPS y la infraestructura hospitalaria	14
2.7 Códigos QR en un entorno hospitalario	14
2.8 Sensores móviles	14
2.8.1 Acelerómetro	15
2.8.2 Giroscopio	16
2.9 Unidad de Medición Inercial (IMU)	16
2.10 Sensor Manager Android	16
2.11 Sistema de coordenadas	16
2.12 Realidad Aumentada (AR)	17
2.13 Unity para el desarrollo de aplicaciones con AR	18
2.13.1 AI Navigation	19
2.13.2 Vuforia Engine	20
2.13.2.1 ARCore	20
2.13.2.2 SLAM	21
2.13.2.3 VISLAM	21
2.14 Discusión	21
Capítulo 3. Diseño de la aplicación móvil	22
3.1 Introducción	22
3.2 Implementación de tecnología IPS con Códigos QR para la localización dentro de un entorno hospitalario	22
3.3 Especificaciones de Software	22
3.4 Requerimientos	23
3.5 Funcionamiento de la aplicación móvil	23
3.5.1 Funciones de la Aplicación móvil	23
3.5.2 Configuración de Vuforia en Unity para el desarrollo de la aplicación móvil	24
3.5.3 Configuración de componentes AI Navigation para el desarrollo de la aplicación móvil	25
3.5.4 Interfaz de usuario	28
3.5.5 Integración del sistema	28
3.6 Planos y Rutas	30
3.6.1 Infraestructura Unidad de Imagenología	31



	III
3.6.1.1 Zocalo	31
3.6.1.2 Piso 1	31
3.6.2 Rutas	31
3.6.2.1 Selección de inicio del trayecto	31
3.6.2.2 Selección del destino de trayecto	33
3.7 Seguridad y costos	33
3.8 Discusión	33
Capítulo 4. Pruebas de la Aplicación Móvil y Resultados	35
4.1 Introducción	35
4.2 Escenas, funcionamiento de botones y cambios de escena	35
4.2.1 Resultados	35
4.3 Lectura e identificación de códigos QR	38
4.3.1 Resultados	38
4.4 Identificación del destino seleccionado	39
4.4.1 Resultados	39
4.5 Seguimiento de rutas dentro de la Unidad de Imagenología	40
4.5.1 Resultados	40
4.6 Pruebas de funcionamiento dentro de otro establecimiento	40
4.6.1 Resultados	40
4.7 Discusión	41
Capítulo 5. Conclusión	43
5.1 Conclusiones	43
5.2 Trabajo a Futuro	43
Capítulo 6. Bibliografía	45
Anexo A FUNCIONAMIENTO SCRIPT CENTRAL	52
Anexo B. GUÍA PARA LA IMPLEMENTACIÓN DEL SISTEMA	54
Anexo C. PLANO ZOCALO C.A.A	57
Anexo D. PLANO PRIMER PISO C.A.A	58



Anexo E. PLANO PRIMER PISO MONOBLOCK

Anexo F. PLANO EDIFICIO DE SISTEMAS



Índice de Tablas

1.	Comparación entre Tecnologías IPS	13
2.	Activación de la cámara AR y del <i>NavMeshAgent</i> a partir del inicio del recorrido . . .	53
3.	Selección del <i>targetObject</i> a partir del final del recorrido	53



Índice de Figuras

1.	Esquema de sistema de localización basado en RSSI	5
2.	Esquema común de localización con uso de ToF	7
3.	Esquema común de localización con uso de TDoA	8
4.	Sistema de coordenadas para un dispositivo Android	17
5.	Configuración de parámetros para un <i>NavMesh Agent</i> , Unity.	19
6.	Funcionamiento del <i>NavMesh Path</i> , Unity.	20
7.	Configuración de Vuforia, Unity	25
8.	Perspectiva Tridimensional de los Planos, Unity	26
9.	Configuraciones para crear un <i>NavMesh Agent</i> , Unity	26
10.	Configuraciones para crear una <i>NavMesh Surface</i> , Unity	27
11.	Configuraciones para crear una <i>Line Renderer</i> , Unity	27
12.	Esquema de funcionamiento para el sistema de localización	29
13.	Diagrama de flujo para la interacción entre escenas.	29
14.	Esquema de <i>scripts</i> en Unity.	30
15.	Ubicación entrada C.A.A en Google Maps.	32
16.	Ubicación entrada visitas en Google Maps.	32
17.	Escena de Bienvenida para la Aplicación Móvil	35
18.	Escena de Informaciones para la Aplicación Móvil	36
19.	Escena de Selección de Inicio para la Aplicación Móvil	37
20.	Escena de Selección de Destino para la Aplicación Móvil	37
21.	Escena de Visualización de la Ruta para la Aplicación Móvil	38
22.	Identificación de códigos QR en Unity.	38
23.	Identificación del destino seleccionado en Unity.	39
24.	Error en consola para la lectura de códigos QR en Unity.	39
25.	Prueba de recorrido dentro del edificio de Sistemas.	41

26.	Esquema del <i>script CreaRuta.cs</i>	52
27.	Creación de una ruta con y sin <i>path</i>	53
28.	Configuración de cámaras AR en Unity.	54
29.	Configuración de los <i>Empty Objects</i> para el inicio del trayecto en Unity.	54
30.	Integración de elementos para el <i>script CreaRuta.cs</i> en Unity.	55
31.	Integración de elementos para el <i>script ApagaAR.cs</i> en Unity.	56
32.	Configuración del botón de retroceso en Unity.	56
33.	Plano Zocalo C.A.A, HGGB	57
34.	Plano Primer Piso C.A.A, HGGB	58
35.	Plano Primer Piso Monoblock, HGGB	59
36.	Plano del segundo piso del edificio de Sistemas.	60



Siglas

- ANN** Redes Neuronales Artificiales (Artificial Neural Networks)
- AoA** Ángulo de Llegada (Angle of Arrival)
- AR** Realidad Aumentada (Augmented Reality)
- ATG** Generador de Objetivos de Área (Area Target Generator)
- BLE** Bluetooth de Baja Energía (Bluetooth Low Energy)
- GNSS** Sistema Global de Navegación por Satélite (Global Navigation Satellite System)
- GPS** Sistema de Posicionamiento Global (Global Positioning System)
- IMU** Unidades de Medición Inercial (Inertial Measurement Units)
- IPS** Sistemas de Posicionamiento Interior (Indoor Positioning Systems)
- kNN** k-Vecinos más Cercanos (k-Nearest Neighbor)
- LBS** Servicios Basados en la Localización (Location-Based Services)
- MAC** Control de Acceso al Medio (Medium Access Control)
- MEMS** Sistemas Microelectromecánicos (Microelectromechanical Systems)
- PDR** Navegación por Estima Peatonal (Pedestrian Dead Reckoning)
- QR** Quick Response
- RF** Radio Frecuencia (Radio Frequency)
- RFID** Dispositivo de Identificación por Radiofrecuencia (Radio Frequency Identification Device)
- RSS** Fuerza de Señal Recibida (Received Signal Strength)
- RSSI** Indicador de Fuerza de Señal Recibida (Received Signal Strength Indicator)
- RTof** Tiempo de Vuelo de Retorno (Return Time of Flight)
- SDK** Kit de Desarrollo de Software (Software Development Kit)
- SLAM** Localización y Mapeo Simultáneo (Simultaneous Localization And Mapping)
- SVM** Máquina de Vectores de Soporte (Support Vector Machine)
- TDoA** Diferencia de Tiempo de Llegada (Time Difference of Arrival)
- ToA** Tiempo de Llegada (Time of Arrival)
- ToF** Tiempo de Vuelo (Time of Flight)
- UHF** Frecuencia Ultra Alta (Ultra High Frequency)
- UWB** Banda Ultraancha (Ultra WideBand)
- VIO** Odometría Visual-Inercial (Visual-Inertial Odometry)
- VISLAM** Localización y Mapeo Simultáneo Visual-Inercial (Visual-Inertial Simultaneous Localiza-

tion And Mapping)

VLC Comunicación por Luz Visible (Visible Light Communication)

VR Realidad Virtual (Virtual Reality)



Capítulo 1. Introducción

1.1. Introducción General

El ingreso constante de avances tecnológicos dentro de los hospitales a lo largo de los años ha sido primordial para contribuir a la mejora de los procesos en el área de la salud y a la entrega de servicios más personalizados. De esta forma es que los recintos hospitalarios buscan aumentar la eficiencia y promover su interoperabilidad con la ayuda de nuevas herramientas tecnológicas.

A pesar de los grandes avances logrados en los últimos años, aún no se ha profundizado en sistemas que logren guiar a una persona en movimiento hacia un lugar en específico dentro de un hospital. Este hecho ha sido una desventaja en la atención hospitalaria ya que generalmente para los usuarios suele ser complicado ubicar lugares dentro de grandes instalaciones ya sea por problemas de movilidad o por el desconocimiento que podrían tener los nuevos pacientes sobre el recinto hospitalario al cual se van a atender por alguna necesidad. A esto se le añade el hecho de que el Sistema Global de Navegación por Satélite (Global Navigation Satellite System, GNSS) es inefectivo en una infraestructura hospitalaria por lo que no es adecuado para lograr estimar la posición de un objeto dentro de una construcción o edificio [1], lo que conlleva una pérdida de tiempo tanto para la persona que se va a atender como para los funcionarios de la salud.

Actualmente, existen sistemas que logran hacer seguimiento de objetos o dispositivos dentro de edificaciones para estimar su posición en tiempo real, lo cual puede llegar a ser costoso por la necesidad de infraestructura y el desarrollo de softwares sofisticados [2].

Con el fin de darle una solución a esta problemática, se hace necesaria la exploración de nuevas alternativas para el desarrollo de una propuesta que utilice tecnologías de Sistemas de Posicionamiento Interior (Indoor Positioning Systems, IPS). Por lo cual se busca desarrollar una herramienta de posicionamiento interior dentro de una infraestructura hospitalaria mediante una aplicación móvil que indique la trayectoria o ruta hacia el destino solicitado con Realidad Aumentada (Augmented Reality, AR). También, se propone que el sistema trabaje en conjunto y a través de códigos QR que enlacen las rutas, otorgando una solución de fácil uso, segura y asequible para la mayoría de los usuarios del Hospital Clínico Regional de Concepción (HGGB) en la Unidad de Imagenología.

El centro de responsabilidad (CR) de Imagenología está a cargo de 3 unidades: Imagenología, Neurovascular y Medicina Nuclear. A su vez cada una de estas unidades ofrece los distintos servicios relacionados con exámenes médicos. La unidad de Medicina Nuclear utiliza equipos como un SPECT-CT. Luego, esta la sección Neurovascular, la cual está a cargo del equipo de angiografía. Por último,

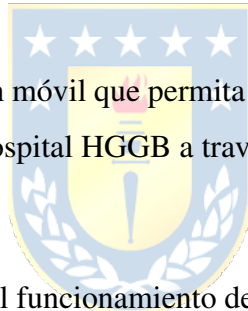
Imageneología utiliza ecógrafos, equipos de rayos X, TAC y resonadores magnéticos.

Las unidades anteriormente mencionadas dan atención a pacientes en situación de urgencia, hospitalizados y ambulatorios. La atención ambulatoria es para aquellos pacientes que se tratan por alguna patología y deben solicitar una hora para la realización de algún el examen que es pedido por su médico tratante. Por lo general, es en la atención ambulatoria donde se presentan problemas desde el proceso de la solicitud de una hora para el examen hasta la momento de realizarlo. La primera razón radica en que los pacientes podrían desconocer el horario y el lugar donde se realizan las solicitudes de horas. La segunda razón se debe a las dificultades que se presentan al querer encontrar las salas donde se realizan los exámenes, en parte debido a la infraestructura del hospital. Como consecuencia se generan retrasos, pérdida de horas, un incremento en la carga administrativa y descontento entre los pacientes lo que afecta directamente a la eficiencia del hospital.

1.2. Objetivos

1.2.1 Objetivo General

Crear e implementar una aplicación móvil que permita al paciente, mediante códigos QR, navegar a las ubicaciones de salas dentro del hospital HGGB a través de planos interactivos.



1.2.2 Objetivos Específicos

- i. Recopilar información respecto al funcionamiento de la Unidad de Imageneología.
- ii. Diseñar el plano y las rutas.
- iii. Crear el algoritmo para implementar la app móvil.
- iv. Validar la aplicación móvil.

1.3. Alcances y Limitaciones

El desarrollo del sistema de localización interior solo contemplará la Unidad de Imageneología del hospital HGGB mediante el diseño de un mapa con las salas correspondientes.

En cuanto a las características del sistema, se trabajará con un mecanismo de posicionamiento donde el entorno no tenga acceso a la ubicación del dispositivo para la seguridad y privacidad del usuario. El dispositivo móvil tendrá acceso al mapa de la unidad y las rutas preestablecidas hacia las salas.

1.4. Metodología

La metodología a seguir para llevar a cabo el proyecto incluye una serie de pasos para desarrollar eficientemente el sistema de localización intrahospitalario de la Unidad de Imageneología. En primer lugar, se realizará una búsqueda bibliográfica atinente al proyecto y enfocada en sistemas de localización en interiores. Luego, se establecerá el marco teórico para seleccionar la tecnología más adecuada que será utilizada en el desarrollo del sistema de localización. El siguiente paso consiste en seleccionar un IDE adecuado para desarrollar los algoritmos necesarios en la implementación del IPS donde también se incluye el diseño de la aplicación móvil y de las rutas hacia las salas enlazadas a códigos QR. Finalmente, se buscará verificar el correcto funcionamiento del sistema y comprobar su precisión.

1.5. Temario

- i. Capítulo 1: Se presenta la introducción general del informe, incluye objetivos, alcances y metodología a realizar.
- ii. Capítulo 2: Se presenta el marco teórico, se describen las bases para el desarrollo de tecnologías IPS, las diferencias entre cada una, avances tecnológicos atinentes al tema y otras definiciones importantes.
- iii. Capítulo 3: Se presenta una propuesta de aplicación móvil, se abarcan sus características y el proceso de su desarrollo.
- iv. Capítulo 4: Se presentan las pruebas realizadas para comprobar el funcionamiento de la aplicación móvil y los resultados principales.
- v. Capítulo 5: Se presentan la conclusión del informe y el trabajo a futuro.

Capítulo 2. Marco Teórico

2.1 Introducción

En este capítulo se abordan las bases teóricas que sustentan las tecnologías de posicionamiento en interiores, se hace necesario analizar los principios que rigen a las señales de localización, describir los distintos sistemas que han sido implementados y analizar las alternativas disponibles para desarrollar la aplicación móvil comparando sus ventajas y desventajas. Además, es importante destacar aspectos y herramientas que facilitan la comprensión del funcionamiento de los dispositivos móviles para que puedan ser utilizados como dispositivos de localización.

2.2 Los Servicios Basados en la Localización (LBS)

Los Servicios Basados en la Localización (Location-Based Services, LBS) son servicios que proveen las ubicaciones geográficas tanto de usuarios como de entidades dentro de entornos exteriores e interiores [3]. Uno de los LBS más utilizado es el Sistema de Posicionamiento Global (Global Positioning System, GPS), es muy eficiente para localizar personas o dirigirse a algún lugar en específico de entornos exteriores pero, al enfrentarse en un entorno interior, la señal GPS pierde su eficacia debido a la estructura de la edificación [4]. Como consecuencia, se ha ido expandiendo el campo de desarrollo de los IPS. Esto comenzó a tener mayor relevancia debido al gran desarrollo tecnológico de dispositivos móviles ya que actualmente permiten implementar sistemas de localización interior de mayor precisión [2].

2.3 Principios de posicionamiento

Las tecnologías IPS hacen uso de técnicas métricas que entregan la información necesaria para lograr localizar y posicionar acertadamente objetos o dispositivos que se encuentren en ambientes interiores basándose en principios de posicionamiento para señales. Las técnicas y/o parámetros más relevantes se describen a continuación.

2.3.1 Indicador de Fuerza de Señal Recibida (RSSI)

La Fuerza de Señal Recibida (Received Signal Strength, RSS) es la potencia real de la señal recibida por el receptor medida en decibelios-milivoltios (dBm) o milivoltios (mV). Mediante el valor de RSS se estima la distancia entre el transmisor y el receptor a través de modelos de propagación de señales (ver Fig. 1). Esta metodología es ampliamente utilizada porque su implementación es de baja complejidad y de bajo costo.

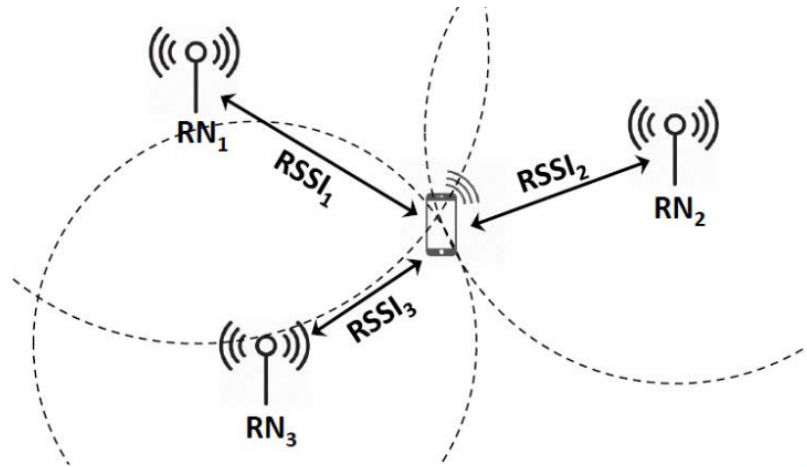


Fig. 1: Esquema de sistemas de localización basados en RSSI [2].

El Indicador de Fuerza de Señal Recibida (Received Signal Strength Indicator, RSSI) es una medida relativa de RSS con variables arbitrarias. Además, el rango de los valores que toma es definido por cada fabricante.

La distancia d entre un transmisor y un receptor se calcula a través de la siguiente fórmula:

$$RSSI = A - 10n \log_{10} \quad (1)$$

donde n es el exponente de pérdida de la señal y A es el valor que adquiere el RSSI a la distancia de referencia desde el receptor [2].

La RSS puede sufrir interferencias debido a la presencia de otros objetos y, en consecuencia, hay una disminución en la precisión de los sistemas basados en RSSI. Para contrarrestar este efecto, se suelen usar metodologías combinadas con RSSI para mejorar la efectividad de la localización [5].

2.3.2 Trilateración

La Trilateración es un método de fácil implementación que consiste en encontrar las coordenadas (x, y) de un objetivo mediante las coordenadas (x_n, y_n) de los nodos de referencia:

$$d_n^2 = (x_n - x)^2 + (y_n - y)^2 \quad (2)$$

Esta técnica se acompaña con los cálculos de RSSI [5]. De esta forma obtenemos las distancias d_n despejando en la fórmula (RSSI) de la siguiente manera:

$$d_n = 10^{\frac{RSSI - A}{10 * n}} \quad (3)$$

2.3.3 Triangulación

La Triangulación hace uso de las propiedades geométricas de los triángulos para encontrar la distancia entre un transmisor y un receptor o la orientación de llegada de la señal al receptor [6]. A continuación, se mencionan las técnicas de cálculo basadas en Triangulación.

2.3.3.1 Tiempo de vuelo (ToF/ToA)

El Tiempo de Vuelo (Time of Flight, ToF) o Tiempo de Llegada (Time of Arrival, ToA), es el tiempo que tarda una señal en viajar desde el emisor hasta el receptor. Por lo tanto, la distancia d entre el transmisor y el receptor corresponde a la siguiente ecuación:

$$d = c * t \quad (4)$$

donde c es la velocidad de la luz y t es el tiempo que demora la señal en llegar desde el emisor al receptor [6].

Sin embargo, en condiciones no ideales, el camino entre el transmisor y el receptor no es directo. Por lo tanto existe un retraso en la llegada de la señal afectando directamente en el calculo de la distancia d real [2]. Suponiendo que la señal se transmite en un tiempo t_1 pero la señal se recibe en el tiempo t_2 , la distancia real se calcula de la siguiente forma:

$$d_{Real} = c * (t_2 - t_1) \quad (5)$$

Al encontrarnos con espacios bidimensionales, es necesario que se haga uso de al menos 3 puntos de referencia R_i ya que de esta manera se pueden calcular las coordenadas del objeto que quiere ser localizado (ver Fig. 2). Siguiendo la ecuación de una circunferencia:

$$d_i = \sqrt{(x_i - x)^2 + (y_i - y)^2} \quad (6)$$

Es importante que estos nodos o puntos de referencia R_x estén correctamente sincronizados para la transmisión de la señal [6].

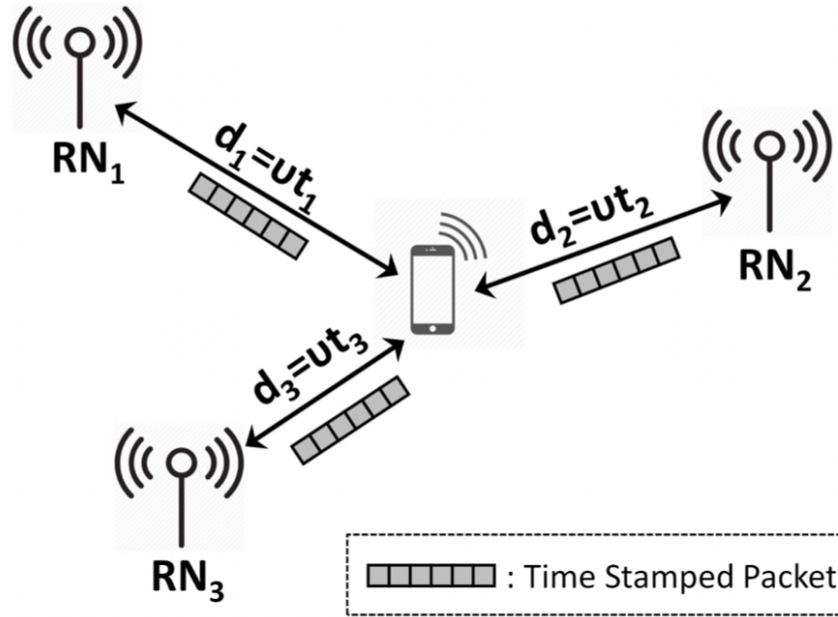


Fig. 2: Esquema común de localización con uso de ToF [2].

Otro aspecto a mencionar es que los cálculos de ToF también pueden verse afectados por el ancho de banda y la frecuencia de muestreo. A mayor ancho de banda, mayor es la resolución del ToF. Lo mismo ocurre en el caso de aumentar la frecuencia de muestreo [2].

2.3.3.2 Diferencia de Tiempo de Llegada (TDoA)

La Diferencia de Tiempo de Llegada (Time Difference of Arrival, TDoA) consiste en una medida de localización que permite obtener la ubicación de un objetivo a través de la diferencia de tiempo de llegada entre las propagaciones de señales que envían varios transmisores a un receptor [2]. A diferencia de ToF, la técnica de medición TDoA no necesita una sincronización en el tiempo entre transmisores y es más fácil de implementar [7, 6]. La ecuación que describe esta situación es la siguiente:

$$\Delta d = c * \Delta t. \quad (7)$$

donde c es la velocidad de la luz y Δt es la diferencia en el tiempo de llegada [6]. En espacios bidimensionales es posible obtener las coordenadas del objetivo (x, y) siguiendo la ecuación [6]:

$$\Delta d = \sqrt{(x_i - x)^2 - (y_i - y)^2} - \sqrt{(x_j - x)^2 - (y_j - y)^2} \quad (8)$$

En espacios tridimensionales:

$$\Delta d = \sqrt{(x_i - x)^2 + (y_i - y)^2 + (z_i - z)^2} - \sqrt{(x_j - x)^2 + (y_j - y)^2 + (z_j - z)^2} \quad (9)$$

En estos casos los nodos de referencia quedan representados con las coordenadas $(x_{i,j}, y_{i,j}, z_{i,j},)$ y, al igual que en el calculo de ToA, se necesitan de al menos 3 nodos para conocer la ubicación exacta del objetivo la cual estará en el punto de intersección de hipérbolas [2] (ver Fig. 3).

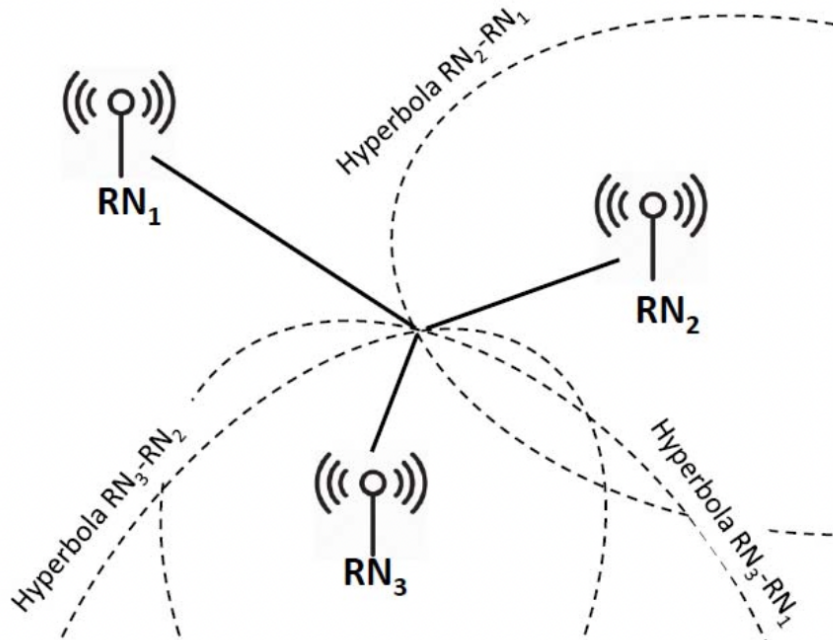


Fig. 3: Esquema común de localización con uso de TDoA [2].

2.3.3.3 Tiempo de Vuelo de Retorno (RToF)

El Tiempo de Vuelo de Retorno (Return Time of Flight, RToF) corresponde a una medida del tiempo que suma lo que tarda la señal en ser enviada por el transmisor hasta el receptor y lo que se tarda la señal en ser devuelta desde el receptor al mismo transmisor. La distancia se calcula de la siguiente forma:

$$d = c * \frac{(t_4 - t_1) - (t_3 - t_2)}{2} \quad (10)$$

donde c es la velocidad de la luz, t_1 corresponde al tiempo en que la señal fue transmitida por el transmisor, t_2 el tiempo en que la señal llega al receptor, t_3 el tiempo en que el transmisor envía una señal de respuesta y t_4 es el tiempo en el cual la señal llega al transmisor [2].

2.3.3.4 Ángulo de llegada AoA

El Ángulo de Llegada (Angle of Arrival, AoA) corresponde al ángulo en el cual la señal emitida llega al receptor. Para un sistema basado en AoA, es común que se haga uso de una matriz de antenas en el receptor con el fin de estimar el ángulo mediante la intersección de las señales. De esta forma, es posible calcular la distancia a través del ángulo estimado. Una desventaja es que la precisión disminuye a medida que la distancia entre el receptor y el transmisor de la señal aumenta [2].

2.3.4 Proximidad

La proximidad es una técnica que entrega la información de un dispositivo u objeto con respecto a una posición ya conocida. Por lo tanto, la localización del dispositivo se considera como la localización del punto de referencia o nodo. La precisión de este método depende de la sensibilidad de los detectores [6]. Las siguientes técnicas corresponden a métodos de proximidad:

- **Landmarking**: los nodos no tienen que transmitir una señal para saber la posición de un objeto. Un ejemplo es el uso de Landmarking con códigos QR, en este método se utilizan los códigos para conocer las coordenadas del lugar donde ha sido escaneado. [8].
- **Closest Neighbour Method**: los nodos son utilizados como transmisores. Existen dos sistemas basados en este método. El primer método tiene enfoque en el usuario, el objeto o dispositivo recibe las señales transmitidas y en base a eso estima su propia ubicación de acuerdo con el nodo más cercano, es decir, toma la ubicación del nodo que le transmita la señal más fuerte. El segundo método se basa en el servidor, el servidor se encarga de ubicar el dispositivo y consulta una base de datos para ubicar a los nodos. En consecuencia, el método basado en el usuario es el más seguro ya que no necesita transmitir su ubicación al exterior. En cambio, el método basado en el servidor involucra más procesos externos por lo cual es menos seguro.[8].

2.3.5 Análisis de escena

El análisis de escena es un método de localización que consiste en ir mapeando la señal transmitida para recopilar sus propiedades en distintas ubicaciones. Luego, las propiedades, también llamadas huellas digitales, se almacenan en una base de datos y, al momento de estimar la ubicación de un objetivo, se comparan las mediciones realizadas sobre el objetivo con las mediciones almacenadas. Finalmente, la ubicación del dispositivo se estima con la huella digital más similar [6].

Al momento de implementar el sistema, es necesario utilizar algoritmos para comparar las mediciones en tiempo real con las huellas digitales. Los métodos más comunes implementados en los

algoritmos son las Redes Neuronales Artificiales (Artificial Neural Networks, ANN), los k-Vecinos más Cercanos (k-Nearest Neighbor, kNN), la Máquina de Vectores de Soporte (Support Vector Machine, SVM) y los métodos probabilísticos [2].

La ventaja de utilizar análisis de escena es que no se necesita conocer la ubicación de los nodos ya que la localización es en base a las huellas digitales. Las desventajas radican en que pueden aumentar los costos al ir comparando las mediciones obtenidas con la base de datos y que podría surgir la necesidad de volver a recopilar los datos de las huellas digitales [9].

2.4 Sistemas de Posicionamiento Interno (IPS)

Un IPS es sistema de posicionamiento local que estima la posición dentro de edificios o infraestructuras. El funcionamiento y la precisión de estos sistemas varia según el rango de cobertura y la estructura del edificio. Su desarrollo varia notablemente ya que se ven influenciados por los materiales de construcción, objetos dentro de la edificación y el transito de personas. Esto afecta especialmente a aquellos sistemas de posicionamiento interior que hacen uso de las señales para las estimaciones de posicionamiento. Es por esta razón que se han creado e implementado nuevas tecnologías para ir mejorando la precisión de los sistemas [10].

Hasta el momento, podemos encontrar numerosas tecnologías IPS, cada una con sus propias ventajas y desventajas. A continuación, se describe cada una de ellas.

2.4.1 Wi-Fi

Aunque es principalmente utilizado para acceder a internet desde algún dispositivo, Wi-Fi es uno de los sistemas más estudiados para la localización en interiores. Opera bajo los estándares de IEEE 802.11 y posee mayores ventajas otorgadas por su infraestructura y su disponibilidad [11]. Convenientemente, los puntos o zonas WiFi cumplen una doble funcionalidad ya que también son usados como puntos de referencia para recolectar las señales emitidas por los dispositivos, es por esta razón que no necesita mayor infraestructura [2]. Sin embargo, el método se ve afectado por interferencias causadas por los servicios en el área industrial, científica y médica [12].

2.4.2 Dispositivo de Identificación por Radiofrecuencia (RFID)

La tecnología de Radio Frecuencia (Radio Frequency, RF) es una de las más utilizadas por su alta cobertura [4]. Un Dispositivo de Identificación por Radiofrecuencia (Radio Frequency Identification Device, RFID) tiene la capacidad de transmitir y almacenar datos mediante los campos electromagnéticos. El sistema consta de lectores RFID, etiquetas RFID y la red de comunicación que conecta a

los dos componentes anteriormente mencionados. En su conjunto, los lectores RFID se encargan de leer los datos emitidos por las etiquetas RFID usando una frecuencia predeterminada [13]. Existen 2 tipos de RFID, el RFID activo y el RFID pasivo. Por un lado, el RFID activo utiliza bandas de frecuencias de microondas y bandas de Frecuencia Ultra Alta (Ultra High Frequency, UHF), esto permite su utilización para IPS. Por otro lado, el RFID pasivo tiene un rango de comunicación muy acotado haciéndolo inviable para ser usado en sistemas de posicionamiento [12].

2.4.3 Bluetooth

Es un sistema que logra conectar diferentes dispositivos dentro de un espacio común mediante especificaciones de capas físicas y de Control de Acceso al Medio (Medium Access Control, MAC) [2]. La evolución y desarrollo de los servicios otorgados por Bluetooth resultaron en la creación de Bluetooth de Baja Energía (Bluetooth Low Energy, BLE), este nuevo sistema se utiliza en conjunto con balizas para lograr establecer un IPS. El objetivo es determinar la distancia entre la baliza y el dispositivo compatible con BLE. Cuando el dispositivo detecta el identificador transmitido por la baliza se desarrolla una respuesta y el mismo dispositivo proporciona un valor de RSSI. Por lo tanto, la ubicación se determina por el poder del RSSI. [14]

2.4.4 Ultrasonido

Un IPS basado en ultrasonido consiste en detectar y determinar la posición de frecuencias de sonidos entre nodos y transmisores que no son perceptibles por el oído humano (30kHz a 130kHz) [15]. De esta forma, la distancia entre un transmisor y un nodo se calcula a través del ToF y de la velocidad del sonido [2]. Las formas de utilizar un sistema de posicionamiento con ultrasonido pueden ser activas o pasivas. El IPS activo requiere una sincronización entre los nodos de la red por lo cual es acompañado por un impulso RF para lograr la sincronización e incluye estaciones receptoras y emisores de ultrasonido móviles. El IPS pasivo utiliza transmisores de baliza y un dispositivo con micrófono que funciona como el receptor [16]. Estos sistemas son de bajo costo y precisos pero presentan la desventaja de ser sensibles a los cambios de temperatura y humedad afectando directamente a la velocidad del sonido [2].

2.4.5 Banda Ultraancha (UWB)

El funcionamiento de un IPS basado en Banda Ultraancha (Ultra WideBand, UWB) se caracteriza por enviar pulsos ultra cortos mediante un gran ancho de banda que va de 3.1 GHz a 10.6 GHz y ciclos de trabajo cortos, esto ayuda a una mayor exactitud y seguridad al momento de que el sistema

sea implementado. Por los aspectos anteriormente mencionados, hace que sean superiores a IPS que se basan en bandas estrechas como el Wi-fi o Bluetooth [17]. Las ventajas radican en su inmunidad ante la interferencia de otras señales y, además, puede penetrar distintos tipos de materiales [2].

2.4.6 Pedestrian Dead Reckoning (PDR)

La Navegación por Estima Peatonal (Pedestrian Dead Reckoning, PDR) es un método basado en Unidades de Medición Inercial (Inertial Measurement Units, IMU) que nos entrega la estimación del movimiento y la dirección de un dispositivo utilizado por el usuario a partir de una posición inicial conocida. El uso de PDR es de bajo costo y muy versátil ya que para su implementación se necesitan dispositivos que contengan sensores como un acelerómetro, magnetometro, barometro y giroscopio [4, 18]. En la actualidad la mayoría de los *smartphones* vienen integrados con estos sensores por lo que es conveniente que sean aplicados dentro del desarrollo de un IPS basado en PDR [19].

A diferencias de las técnicas descritas anteriormente, los sistemas PDR no necesitan del uso de nodos que transmitan alguna señal. La posición inicial comúnmente se considera como las coordenadas $(0, 0, 0)$ y una aceleración de $0g$ para los ejes de coordenadas (x, y, z) [20]. Luego, la distancia recorrida se calcula a través de los valores entregados por el acelerómetro y la orientación se estima con los valores dados por el giroscopio. Para lograr una buena precisión, es necesario que el usuario sostenga el dispositivo móvil hacia adelante con su mano y que se haga uso de la base de datos de mapas detallados del entorno de la infraestructura para ir corrigiendo la trayectoria del dispositivo [19]. También, para realizar los cálculos de localización con los parámetros entregados por el acelerómetro y el giroscopio es usual que el sistema venga implementado con una app móvil que, mediante algoritmos, logre estimar la ubicación del dispositivo dentro del mapa.

La precisión de la localización depende del modelo desarrollado, las formulas utilizadas para la estimación de la marcha recorrida, la posición del dispositivo al momento de recolectar los datos para los cálculos, la velocidad de la marcha, la altura del usuario y la longitud de las zancadas. Algunos problemas que podrían presentarse en el sistema vienen dados por la acumulación de errores en los datos medidos o que el usuario utilice el dispositivo con una orientación incorrecta [20].

2.4.7 Otros

En la actualidad existen otras tecnologías para desarrollar un IPS a parte de las que se mencionaron anteriormente. Algunos ejemplos son sistemas tales como ZigBee, Comunicación por Luz Visible (Visible Light Communication, VLC), el uso de señales acústicas, Posicionamiento Óptico en Interio-

res y Sistemas de Posicionamiento Infrarrojo [2, 4]. Últimamente, se han implementado tecnologías híbridas que nacen de la combinación de los distintos tipo de IPS con resultados más prometedores ya que disminuyen las desventajas que cada sistema posee individualmente [6].

Cada tecnología IPS descrita cuenta con sus propios rangos y precisiones al momento de localizar un objetivo. La comparación entre los sistemas se ve reflejada en la tabla 1.

Tabla 1: Comparación entre Tecnologías IPS

Tecnología	Rango [m]	Precisión [m]	Costo
Wi-Fi	250	10	Bajo
RFID	200	1-5	Alto
Bluetooth	100	2-5	Bajo /Medio
Ultrasonido	2-10	1	Medio /Alto
UWB	1-50	0.01-1	Alto
PDR	-	1-2	Bajo

2.5 Seguridad y privacidad en IPS

La seguridad y privacidad de un IPS depende de su infraestructura y sus mecanismos de posicionamiento. Al momento de modelar el sistema, el factor más importante es la precisión de localización. Como consecuencia, la protección del usuario pasa a tomar un papel de menor relevancia debido a que desarrollar un IPS requiere una mayor demanda de recursos computacionales [8].

En primera instancia, para analizar la seguridad y privacidad del sistema es necesario determinar el enfoque que podría tener. Por un lado, podemos encontrarnos con sistemas de localización en que el entorno conozca la ubicación del dispositivo, esto conlleva a la realización de seguimiento en tiempo real sobre algún objeto, es decir, el dispositivo comparte su ubicación con el sistema. Por otro lado, es posible que el dispositivo conozca las ubicaciones de objetos o lugares en su entorno realizando una acción de mapeo sin la necesidad de que el entorno sepa la ubicación del dispositivo. Cada enfoque mencionado tendrá distintas consecuencias en la seguridad, privacidad e integridad del usuario.

Existen varios problemas de seguridad que podrían aparecer en el uso de un sistema de localización interior. Los ataques de seguridad pueden surgir tanto en el servidor como en la comunicación entre el servidor y el dispositivo usuario [9]. Esto también depende de que si existe o no el uso de transmisores de señal dentro del sistema o si la localización se basa en el usuario o en el servidor.

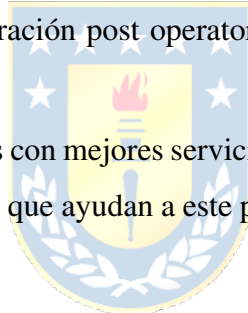
2.6 IPS y la infraestructura hospitalaria

Para el diseño de un IPS es necesario conocer al entorno al cual nos enfrentamos y sus características por lo que se debe considerar de que cada recinto hospitalario tiene una topología interior única. Estos espacios poseen distintas distribuciones interiores que involucran puertas, habitaciones, pasillos, salas de exámenes, etc [3]. Es un requerimiento representar las áreas físicas que delimitan la zona externa de la interna y también garantizar que las características del interior queden definidas de forma precisa [4].

2.7 Códigos QR en un entorno hospitalario

En un entorno hospitalario, es habitual el desarrollo y la implementación de nuevas tecnologías en pro de ir mejorando y personalizando los servicios que otorgan los centros salud. Es por esta razón que la implementación de la innovación dentro de los hospitales ha jugado un rol importante en distintos aspectos en el sector de la salud, por ejemplo, la instauración de telemedicina a nivel mundial, la disminución en los tiempos de recuperación post operatorios, la renovación de la infraestructura de establecimientos, etc [21].

Con el fin de proveer a los pacientes con mejores servicios, el sector de la salud ha sido el foco para la introducción de nuevas herramientas que ayudan a este propósito, dentro de las cuales encontramos los Códigos Quick Response (QR).



Los códigos QR son códigos de barra bidimensionales, es decir, incluye información tanto verticalmente como horizontalmente permitiendo contener una gran cantidad de dígitos [22]. La información codificada se representa en un patrón formado de píxeles blancos y negros. Esto posibilita enlazar páginas web, mensajes de textos, imágenes, aplicaciones móviles o servicios afines con dispositivos inteligentes al momento de ser escaneado [23].

La gran ventaja de los códigos QR es que la mayoría de las cámaras de los teléfonos móviles están habilitadas para la lectura del código por lo que es asequible a una gran cantidad de usuarios. Otra ventaja es la disponibilidad de páginas web gratuitas que ofrecen crear los códigos QR.

2.8 Sensores móviles

En la actualidad, los *smartphones* están integrados con una serie de sensores muy versátiles a la hora de analizar las actividades que realizamos día a día. En definitiva, son muchos los estudios que dan cuenta de esto. Existen diversas funcionalidades que se les puede dar, por ejemplo, para la estimación de la aceleración de un vehículo [24], para sistemas de posicionamiento y localización basados

en PDR [25, 26], para monitorear las actividades físicas realizadas por pacientes diagnosticados con COVID-19 [27] y han sido ampliamente utilizados en la salud para el estudio de condiciones médicas como Parkinson y esquizofrenia [28].

Cuando se habla de querer analizar o monitorear actividades físicas o de localización, los *smartphones* poseen sensores específicos que ayudan a detectar cuando una persona u objeto realiza algún tipo de movimiento. Cada vez es más común su uso para el diseño de sistemas de posicionamiento.

La gran ventaja de los sensores integrados en los teléfonos móviles radica en que su diseño es fundamentado en Sistemas Microelectromecánicos (Microelectromechanical Systems, MEMS). Estos sensores destacan por poseer un tamaño compacto y ser de bajo costo [28], otorgando una integración beneficiosa y optimizada lo que permite acceder a funciones avanzadas como las mencionadas anteriormente.

2.8.1 Acelerómetro

Los acelerómetros son dispositivos que se utilizan para medir la aceleración de un objeto [29]. Los datos monitorizados se utilizan para realizar cálculos tales como número de pasos y distancia de zancada [30]. Esto permite estimar la distancia recorrida teniendo como base las ecuaciones adecuadas para la obtención de la distancia recorrida a través de la aceleración. La posición se puede obtener integrando 2 veces la aceleración lineal [31].

El primer paso es calcular la velocidad $v(t)$:

$$v(t) = \sum_{i=0}^t \int_i^{i+\delta} a(t) dt \quad (11)$$

donde $a(t)$ corresponde a la aceleración y t es el tiempo. Luego, la distancia ($s(t)$) recorrida corresponde a:

$$s(t) = \sum_{i=0}^t \int_i^{i+\delta} v(t) dt \quad (12)$$

Las mediciones realizadas por este dispositivo depende principalmente del ancho de banda, en este aspecto se considera que el ser humano no genera movimientos con rangos de frecuencia superiores a 10[Hz]. Otros aspectos que influyen en las mediciones son la sensibilidad, la frecuencia de respuesta y el rango dinámico [29].

2.8.2 Giroscopio

Los giroscopios son sensores de velocidad angular que tienen la capacidad de detectar cuando el dispositivo está girando mediante la vibración mecánica. Con el giro, se genera una aceleración angular llamada aceleración de Coriolis [32]. Ahora bien, si consideramos que el acelerómetro mide la aceleración en cada eje de coordenadas (x, y, z) e integramos los valores obtenidos en el giroscopio para cada eje es posible obtener el desplazamiento angular O_z en el eje z:

$$O_z \approx \frac{\overline{a_x}}{g} \cdot \theta_x + \frac{\overline{a_y}}{g} \cdot \theta_y + \frac{\overline{a_z}}{g} \cdot \theta_z \quad (13)$$

donde θ_x, θ_y y θ_z corresponde al desplazamiento angular en los ejes x, y, z, respectivamente [30].

2.9 Unidad de Medición Inercial (IMU)

Un sistema IMU generalmente nace de la combinación entre giroscopios y acelerómetros para lograr estimar velocidad, posición y orientación de forma tridimensional. Al hacer uso de este sistema se debe tener en consideración que luego de un periodo de uso, los sensores comienzan a acumular errores causando que se produzca fallos en la estimación de distancia recorrida y orientación [29].

2.10 Sensor Manager Android

La plataforma Android dispone de la herramienta Sensor Manager Android para acceder a los valores medidos por los sensores de movimiento integrados a los dispositivos. Además, da una serie de indicaciones con respecto al manejo de los datos obtenidos. En primer lugar, los acelerómetros y los giroscopios dentro del dispositivo están basados en Hardware y de los cuales se desarrollan sensores basados en Software relacionados con el movimiento, tales como la aceleración lineal. En segundo lugar, gran parte de los dispositivos ya vienen integrados con el acelerómetro y el giroscopio. Por último, todos los valores adquiridos por los sensores se almacenan dentro de una matriz multidimensional nombrada *SensorEvent* y las unidades de medida correspondientes a la aceleración y a la rotación son m/s^2 y rad/s , respectivamente [33].

2.11 Sistema de coordenadas

Los dispositivos Android tienen un sistema de coordenadas de referencia específico. El eje X es tangencial al suelo y apunta hacia el este, el eje Z es tangencial al suelo y apunta hacia el norte. Finalmente, el eje Y es perpendicular a la tierra y apunta hacia el cielo [33] (ver Fig. 4).

Esto adquiere relevancia al momento de la adquisición de datos a través de la herramienta Sensor Manager Android ya que al momento de medir la distancia total recorrida hacia adelante o hacia atrás,

la posición del dispositivo deberá ser siempre la misma para no inducir errores de calculo derivados por la medición de aceleración en los 3 ejes. La razón detrás de esto se da porque al rotar el dispositivo mientras se esta en movimiento, el eje medido ya no capturará el movimiento hacia adelante. Por esto es importante mantener en el tiempo de adquisición de datos la misma orientación y posición del celular para obtener mediciones más precisas y coherentes.

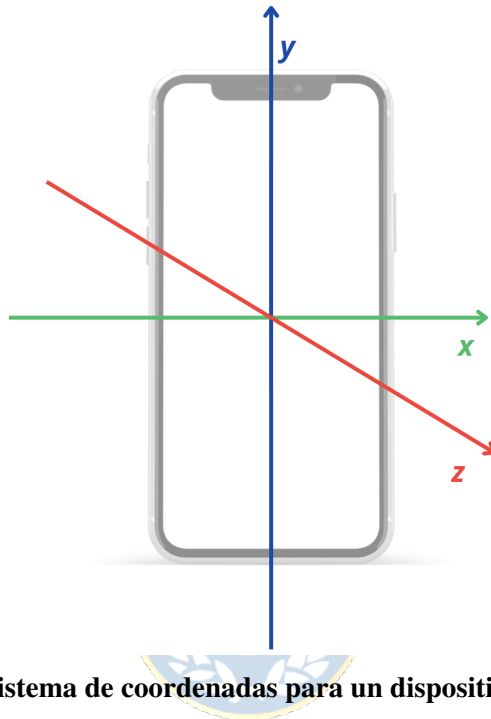


Fig. 4: Sistema de coordenadas para un dispositivo Android.

2.12 Realidad Aumentada (AR)

La AR es un concepto tecnológico que se caracteriza por hacer uso de la superposición de objetos virtuales en la vida real de forma complementaria al entorno para mejorar y enriquecer la experiencia de las personas en su interacción y percepción con la realidad [34].

La AR se diferencia de la Realidad Virtual (Virtual Reality, VR) básicamente porque la VR da una experiencia inmersiva en un entorno virtual donde es posible percibir e interactuar con él. En otras palabras AR trabaja en el mundo real y VR en el mundo virtual [35].

En el ámbito de la investigación, se han desarrollado varios avances los cuales contribuyen a potenciar las distintas funciones que otorga el uso de la AR dando nuevas perspectivas y posibilidades para la innovación de esta tecnología. En el ámbito de la medicina existen investigaciones que se enfocan en el desarrollo de estrategias para la implementación de la AR entre las cuales se pueden encontrar la realización de operaciones en conjunto con sistemas de AR [36] y el desarrollo de un simulador

de ultrasonido de Realidad Aumentada (AR) [37]. Esta tecnología se puede aplicar a más áreas. En la industria para dar soporte a trabajadores que dan servicios de mantenimiento complejo [38] y en la educación mediante aplicaciones móviles didácticas para entretenimiento [39] o para promover una buena alimentación [40].

2.13 Unity para el desarrollo de aplicaciones con AR

Unity es un motor de juegos multiplataforma reconocido por su accesibilidad, flexibilidad y eficiencia al momento de desarrollar videojuegos 2D y 3D. También, se destaca por permitirle a los usuarios generar proyectos en AR y VR [41]. El editor de Unity proporciona herramientas para diseñar y testear las aplicaciones en tiempo real. Además, se pueden añadir paquetes (*Packages*) y componentes que le dan funcionalidades a la aplicación creando un ambiente óptimo para potenciarla. A continuación, se describen *Packages* y componentes útiles en desarrollo de videojuegos y aplicaciones móviles.

- **Scenes:** Las escenas en Unity son la base del proyecto, en ellas se integran los diseños y los elementos que el usuario ve en pantalla. Un proyecto puede estar compuesto por más de una escena [42]. Esto permite que el usuario pueda interactuar con una aplicación más completa y dinámica, por ejemplo, pasando de una escena que contenga el inicio de un videojuego a otra escena que contenga el menú de configuraciones del mismo videojuego.
- **GameObjects:** Son los objetos que están dentro de las escenas y cada uno adquiere distintas funcionalidades y propiedades según los componentes que se le adjunten [43]. Un *GameObject* puede ser un sonido, una imagen, un personaje, un objeto vacío (*Empty Object*), etc.
- **Scripts:** Son códigos en lenguaje C# y se utilizan como componentes dentro de Unity. Esto permite otorgarle propiedades y características a los elementos que componen las escenas [44].
- **Button:** Son elementos dentro de la escena que tienen la capacidad de generar un evento al ser presionados [45].
- **PlayerPrefs:** Es una herramienta que permite almacenar información de forma local [46]. Es útil para pasar datos que no sean confidenciales entre las escenas.
- **Line Renderer:** Componente que dibuja líneas rectas entre puntos cuyas coordenadas se encuentran en una matriz [47]. Al ser un componente visible en escena, Unity permite realizar ajustes de color, efectos, etc.
- **Position Constraint:** Componente que permite mover un objeto de la escena a través del movimiento de uno o más objetos fuentes. Funciona para desplazamientos en los ejes X, Y y Z.

[48].

- **Sprites**: Son objetos 2D y permiten agregar imágenes o texturas a la escena [49].

2.13.1 AI Navigation

Es un *Package* de Unity que permite el movimiento de personajes u objetos dentro de una escena de forma inteligente. El sistema analiza el entorno para definir las zonas transitables a través de la geometría de la escena y considera elementos tales como paredes, obstáculos, escaleras o superficies [50, 51]. En el uso de AI Navigation para el desarrollo de videojuegos se destacan componentes que son muy utilizados para su funcionamiento:

- **NavMesh Agent**: Es el componente que puede desplazarse por la escena. Contiene propiedades ajustables para definir el movimiento del agente (ver Fig. 4).

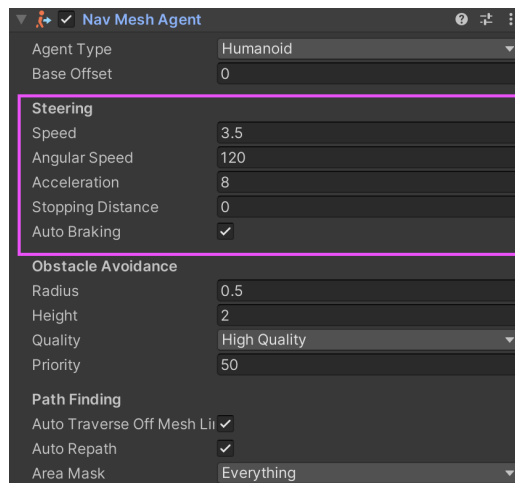


Fig. 5: Configuración de parámetros para un *NavMesh Agent*, Unity.

- **NavMesh Surface**: Es el componente que representa la malla de navegación, es decir, el área por donde puede moverse el *NavMesh Agent* [52].
- **NavMesh Path**: Es una lista de coordenadas que representan las esquinas de una ruta para llegar de un punto a otro dentro de la *NavMesh Surface*. AI Navigation permite calcular el *NavMesh Path* entre 2 puntos, para esto es importante contar con la posición del *NavMesh Agent*, un destino o *target* y una *NavMesh Surface*. El resultado es una lista con las esquinas de la ruta más corta entre el *NavMesh Agent* y el *target* evitando, por ejemplo, chocar con obstáculos o atravesar paredes [53, 54].

En la figura 6 se ejemplifica una situación que muestra la ruta entre dos objetos. La ruta se forma al conectar las coordenadas del *NavMesh Path* indicadas por las flechas sobre una *NavMesh*

Surface representada en color celeste. Es posible observar que AI Navigation permite obtener la menor cantidad de esquinas posibles para crear la trayectoria más corta entre los dos objetos evitando todas las zonas que no están habilitadas.

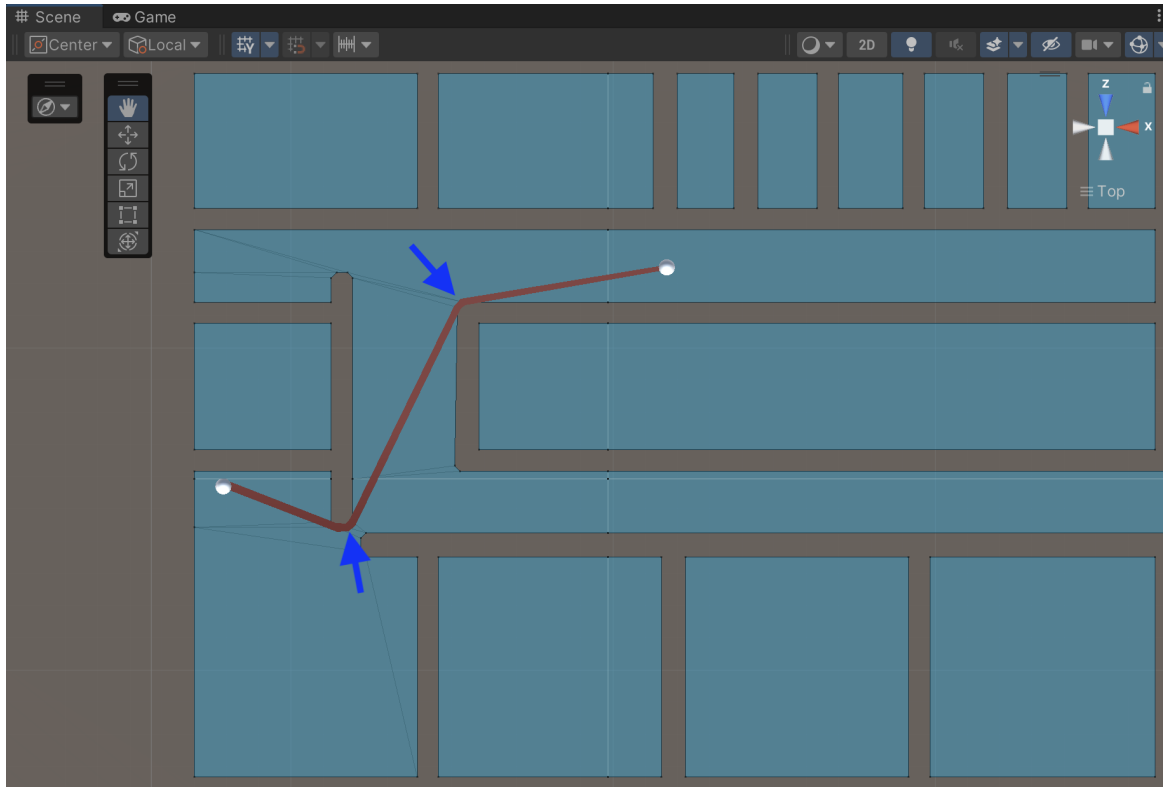


Fig. 6: Ejemplo del Funcionamiento del *NavMesh Path*, Unity.

2.13.2 Vuforia Engine

Vuforia Engine es un Kit de Desarrollo de Software (Software Development Kit, SDK) gratuito y multiplataforma enfocado en implementar la Realidad Aumentada (AR) en aplicaciones o videojuegos a través del uso de herramientas tales como el reconocimiento de imágenes, objetos y espacios para conectar el mundo virtual con el mundo real. El SDK es compatible con Unity y se instala como un *Unity Asset Package* [55]. Al momento de ser implementado en un sistema Android mediante una aplicación móvil, Vuforia utiliza 3 tecnologías dependiendo de las características del teléfono móvil.

2.13.2.1 ARCore

Es un SDK de Realidad Aumentada (AR) desarrollado por Google [56]. Cuando el dispositivo móvil está habilitado para su uso, Vuforia empaqueta la librería ARCore dentro de la instalación y ejecución de la aplicación móvil. La librería es utilizada por Vuforia siempre y cuando el dispositivo

sea compatible con ARCore [57].

2.13.2.2 SLAM

La Localización y Mapeo Simultáneo (Simultaneous Localization And Mapping, SLAM) busca saber la posición dentro un espacio sin poseer previa información del entorno. La técnica tiene por objetivo generar un mapa del lugar mediante su exploración (*Mapping*) y luego estimar la posición dentro del mapa utilizando las trayectoria y un punto de referencia en el mapa (*Localization*) [58, 59].

2.13.2.3 VISLAM

Localización y Mapeo Simultáneo Visual-Inercial (Visual-Inertial Simultaneous Localization And Mapping, VISLAM) es un algoritmo utilizado por Vuforia que fusiona la Odometría Visual-Inercial (Visual-Inertial Odometry, VIO) con SLAM [60]. VIO es la utilización de los sensores IMU y la cámara disponibles en los dispositivos móviles para medir desplazamientos y giros [61].

2.14 Discusión

Son variados los aspectos a considerar para la selección de un sistema de posicionamiento interior que pueda guiar a los pacientes dentro del hospital.

Entre los elementos más destacados se encuentra la infraestructura necesaria para desarrollar un IPS, los costos asociados y la precisión que se desea alcanzar. Para el desarrollo de una aplicación móvil que integre una tecnología IPS dentro de un entorno hospitalario se añaden otros aspectos importantes a considerar: mejorar la experiencia de los pacientes en la búsqueda de lugares o salas dentro del hospital y contribuir a una atención más eficiente y efectiva.

Con la tecnología disponible hasta el momento y el uso de herramientas adecuadas nace la posibilidad de desarrollar una aplicación móvil que cumpla con los objetivos deseados y que minimice costos sin sacrificar la precisión del sistema.

Capítulo 3. Diseño de la aplicación móvil

3.1 Introducción

En este capítulo se presenta una propuesta de aplicación móvil que busca abordar el desafío de guiar a pacientes o usuarios a distintos lugares dentro de un entorno hospitalario. Para esto se hace necesario especificar y analizar todos los aspectos vinculados con el desarrollo y la implementación de la solución propuesta.

3.2 Implementación de tecnología IPS con Códigos QR para la localización dentro de un entorno hospitalario

El uso de códigos QR en los centros de salud ha adquirido una gran relevancia e importancia en cuanto a los nuevos servicios desarrollados para mejorar la atención hospitalaria con enfoques en la tecnología y la innovación. Estas novedosas herramientas abarcan soluciones en distintos aspectos de forma simple y eficaz dentro de los establecimientos. Actualmente, se han implementado para educación en salud [23], mantenimiento de equipos médicos [62], reemplazar folletos de papel en hospitalarios por folletos digitales [63], acceso rápido a la información de un paciente [64], registro de los medicamentos adquiridos por los pacientes [65] y para crear un ID tanto para pacientes como funcionarios dentro del hospital [66]. Sin embargo, aún son pocas las investigaciones de los usos de los códigos QR en el ámbito de localización intrahospitalaria. En consecuencia, se busca implementar un sistema que unifique la tecnología IPS con los códigos QR para otorgar una herramienta de posicionamiento interior. Se propone crear un sistema de localización dentro de un hospital que logre enlazar las rutas hacia las salas del recinto mediante códigos QR desde distintos puntos de acceso. El sistema planteado busca integrar un mapa interactivo en una aplicación móvil que este disponible para aquellos pacientes o usuarios que deseen llegar a alguna sala del hospital sin tener que recurrir a recorridos más largos.

3.3 Especificaciones de Software

La aplicación móvil será desarrollada en Unity 2022.3.9f1 en conjunto con distintos paquetes añadidos dentro del entorno de la plataforma de desarrollo:

- Vuforia Engine
- AI Navigation
- TextMeshPro

El algoritmo es desarrollado en lenguaje de programación C# en conjunto con la plataforma Unity

y está encargado de definir la posición del usuario dentro del hospital. La aplicación busca obtener todas las funcionalidades necesarias para el correcto manejo del sistema, esto incluye todo lo relacionado con la apertura de la cámara móvil, la lectura de códigos QR, la selección de rutas, la visualización de la ruta y la detección de la llegada al lugar seleccionado por el usuario.

3.4 Requerimientos

La aplicación móvil deberá cumplir con un conjunto de requisitos que garanticen su correcta utilidad y, al mismo tiempo, que proporcionen una experiencia agradable para los posibles usuarios.

- **Diseño:** La aplicación móvil contará con un diseño no complejo y lo más intuitivo posible, mostrará los textos de forma clara y concisa. También, contará con botones para iniciar el escaneo de códigos QR, para retroceder y para seleccionar destinos. Cualquier objeto añadido con fines de hacer más agradable la visualización de la aplicación móvil no entorpecerá sus funcionalidades.
- **Funcionalidad sin conexión a internet:** La aplicación móvil deberá ser funcional en cualquier zona del hospital, inclusive aquellas donde haya una señal telefónica deficiente.
- **Optimización:** La aplicación móvil será capaz de generar y mostrar de forma visual la ruta de menor distancia hacia el destino previamente seleccionado por el usuario.
- **Seguridad:** La aplicación móvil solo tendrá acceso a la cámara trasera del dispositivo móvil siempre y cuando la aplicación este en uso.
- **Compatibilidad:** La aplicación móvil será compatible con sistemas Android 8.0 en adelante.

3.5 Funcionamiento de la aplicación móvil

Los aspectos de la aplicación móvil y su funcionamiento interno son abordados a detalle con la perspectiva de mejorar la experiencia de los usuarios y cumplir con los objetivos principales de la aplicación.

3.5.1 Funciones de la Aplicación móvil

A continuación, se describen las funciones que realiza la aplicación móvil.

- **Accesibilidad a la cámara móvil:** La cámara es la herramienta que se usa para 2 de las funciones más significativas de la aplicación, es decir, realizar la lectura de códigos QR y mostrar en pantalla lo que ve el usuario al inicializar el recorrido en tiempo real.
- **Lectura de códigos QR:** Los códigos QR tienen como objetivo marcar el inicio del recorrido una vez escaneado, identificando cuál de los códigos QR disponibles ha sido leído. Esta información se almacena mediante el método *PlayerPrefs* de Unity hasta que el usuario seleccione

otro código QR.

- **Selección de destino:** El usuario puede elegir entre varios destinos mediante un menú tipo *scroll*. Esta información se almacena mediante el método *PlayerPrefs* de Unity hasta que el usuario seleccione otro destino.
- **Generación de rutas:** Se identifica el inicio y el fin de la ruta para generar el camino de menor distancia por caminos habilitados hacia el destino seleccionado.
- **Indicación de la ruta a seguir:** Se hace uso de AR para poder visualizar la ruta en pantalla al mismo tiempo que se muestra la zona por donde va caminando el usuario.

3.5.2 Configuración de Vuforia en Unity para el desarrollo de la aplicación móvil

Con Vuforia Engine importado en Unity, se habilita la cámara AR dentro de las escenas deseadas. El SDK de Vuforia otorga dos funcionalidades importantes a la aplicación. Primero, permite la lectura e identificación de códigos QR. Segundo, permite visualizar objetos virtuales a través de la cámara móvil.

En el desarrollo de la app móvil, es necesario activar la cámara AR en las dos situaciones:

- Al leer los códigos QR en el inicio, se establece el punto de partida dentro del modelo 3D del recinto hospitalario.
- Con la información obtenida en la lectura de códigos QR, se posiciona la cámara AR donde se desea comenzar el recorrido y se genera una ruta desde el inicio del recorrido hasta el destino seleccionado.

Vuforia permite habilitar y configurar las opciones de seguimiento del dispositivo. El seguimiento de dispositivos permite identificar la posición del dispositivo con respecto a objetos virtuales que estén dentro de la escena . Al integrar una cámara AR, la habilitación del seguimiento es predeterminada pero modificable. Se selecciona si se requiere o no el uso de la librería ARCore dentro de la aplicación (ver Fig.7).

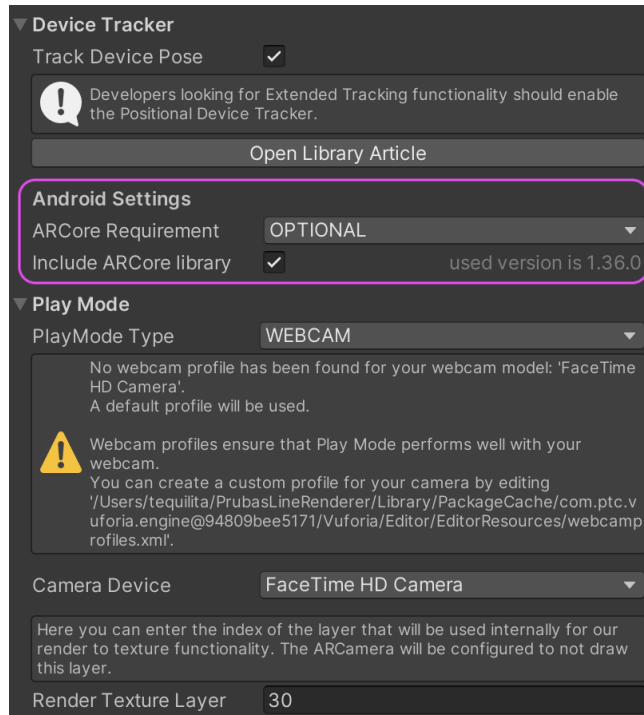


Fig. 7: Configuración de Vuforia, Unity.

El uso de ARCore es opcional para la aplicación desarrollada. Esto implica que se presentan dos situaciones al momento de que la aplicación móvil es instalada:

- El dispositivo móvil está habilitado para el uso de ARCore: Vuforia Engine incluye la librería ARCore dentro de la instalación de la aplicación móvil en dispositivos compatibles.
- El dispositivo móvil no está habilitado para el uso de ARCore: Vuforia Engine utiliza VISLAM.

El SDK se encarga de acceder a los sensores del dispositivo automáticamente y de hacer las estimaciones correspondientes al desplazamiento del usuario para actualizar la posición continuamente mientras esté siguiendo uno de los recorridos dentro de la aplicación móvil.

3.5.3 Configuración de componentes AI Navigation para el desarrollo de la aplicación móvil

Para generar el recorrido dentro de Unity es necesario contar con:

- **Estructura tridimensional:** Se compone de objetos 3D (*3D object*) que simulan las paredes del recinto hospitalario (ver Fig. 8). Las dimensiones de los objetos son trabajadas en metros.

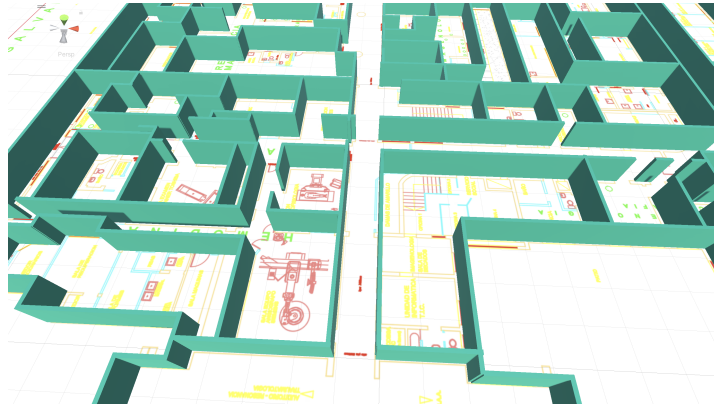


Fig. 8: Perspectiva Tridimensional de los Planos, Unity.

- **NavMesh Agent:** Componente añadido a un *empty object*, está adjunto a la cámara AR mediante *Position Constraint* y representa al usuario dentro de la escena desde que inicia el recorrido hasta que lo finaliza. Por cada lugar en donde se quiera iniciar el trayecto, existe un *NavMesh Agent* con su respectiva cámara AR (ver Fig. 9).

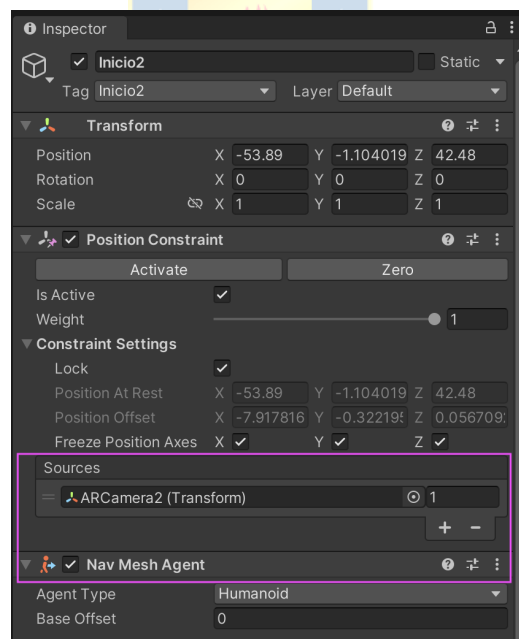


Fig. 9: Configuraciones para crear un NavMesh Agent, Unity.

- **NavMesh Surface:** Componente añadido a un *3D object* plano para simular y definir la superficie por la cual el *NavMesh Agent* esta habilitado para desplazarse (ver Fig. 10).

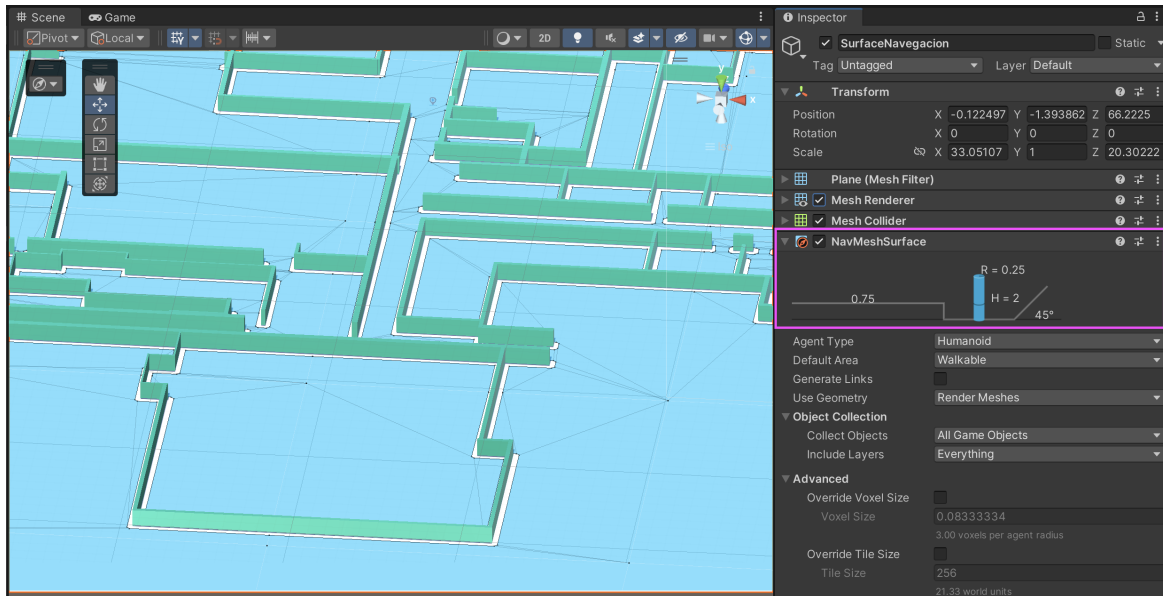


Fig. 10: Configuraciones para crear una *NavMesh Surface*, Unity.

- **Line Renderer:** Componente añadido al *NavMesh Agent* y tiene por objetivo indicar el camino al usuario (ver Fig. 11).

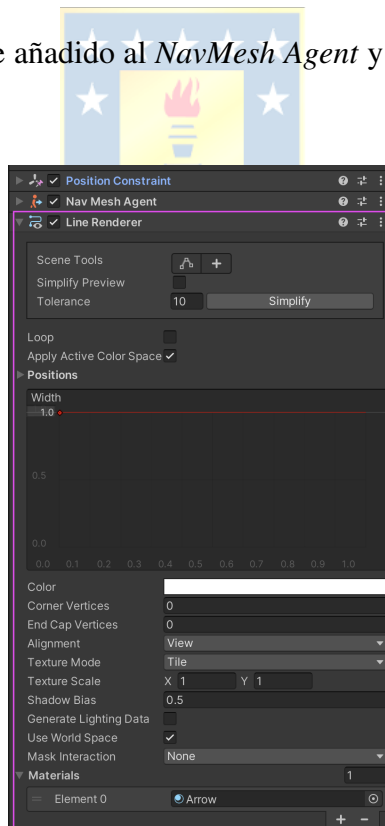


Fig. 11: Configuraciones para crear una *Line Renderer*, Unity.

- **Destinos:** Son *objects* que se posicionan en los lugares de destino dentro de la estructura tridimensional.

3.5.4 Interfaz de usuario

El usuario tiene acceso a distintas escenas diseñadas en Unity al momento de interactuar con la aplicación móvil.

- **Bienvenida:** La escena de bienvenida permite que el usuario pueda abrir la cámara para leer un código QR o acceder a una ventana de informaciones presionando un botón.
- **Informaciones:** La escena de informaciones contiene la información de los lugares para la toma de horas y la recepción de pacientes para exámenes realizados en la Unidad de Imagenología.
- **Selección de inicio de la ruta:** La escena de selección de inicio de la ruta mantiene la cámara activa permitiendo al usuario escanear uno de los códigos QR disponibles en los lugares donde se quiera iniciar el recorrido.
- **Selección de destino:** La escena de selección de destino permite al usuario seleccionar entre varios botones que representan los destinos preestablecidos.
- **Visualización de la ruta:** La escena de visualización de la ruta permite ver las indicaciones que debe seguir el usuario dentro del hospital para llegar al destino deseado en conjunto con la Realidad Aumentada (AR).



3.5.5 Integración del sistema

En la Figura 12, se detalla el procedimiento realizado por el usuario desde el momento que llega al hospital. El primer paso es abrir la aplicación móvil para escanear un código QR, momento en el cual se hace un registro de la identificación del código que ha sido escaneado. Posteriormente, se espera que el usuario seleccione un destino deseado, al igual que en el paso anterior, esta información es almacenada. Ya establecida la información del punto de partida y del destino, es posible generar una ruta que se visualiza en el dispositivo móvil por lo que el usuario procederá a seguirla hasta que la ruta termine y llegue a su destino.

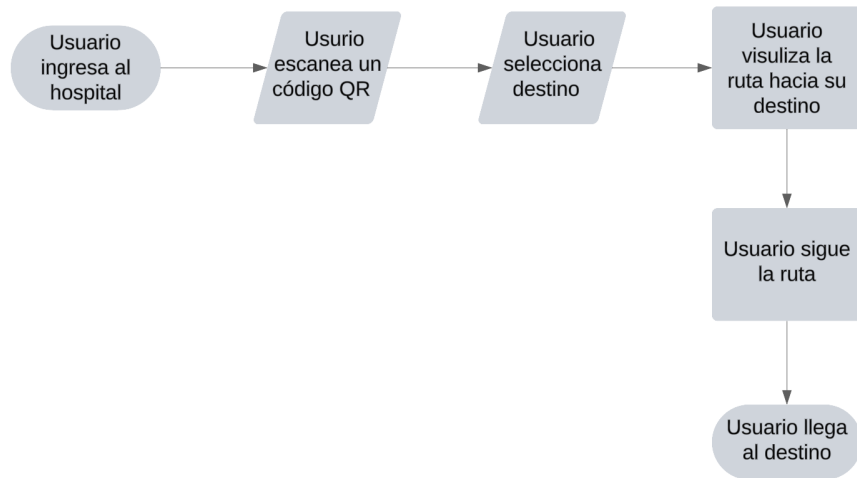


Fig. 12: Esquema de Funcionamiento para el Sistema de Localización.

En la Figura 13, se detalla la integración del conjunto de escenas enfocando el acceso a cada una de estas.

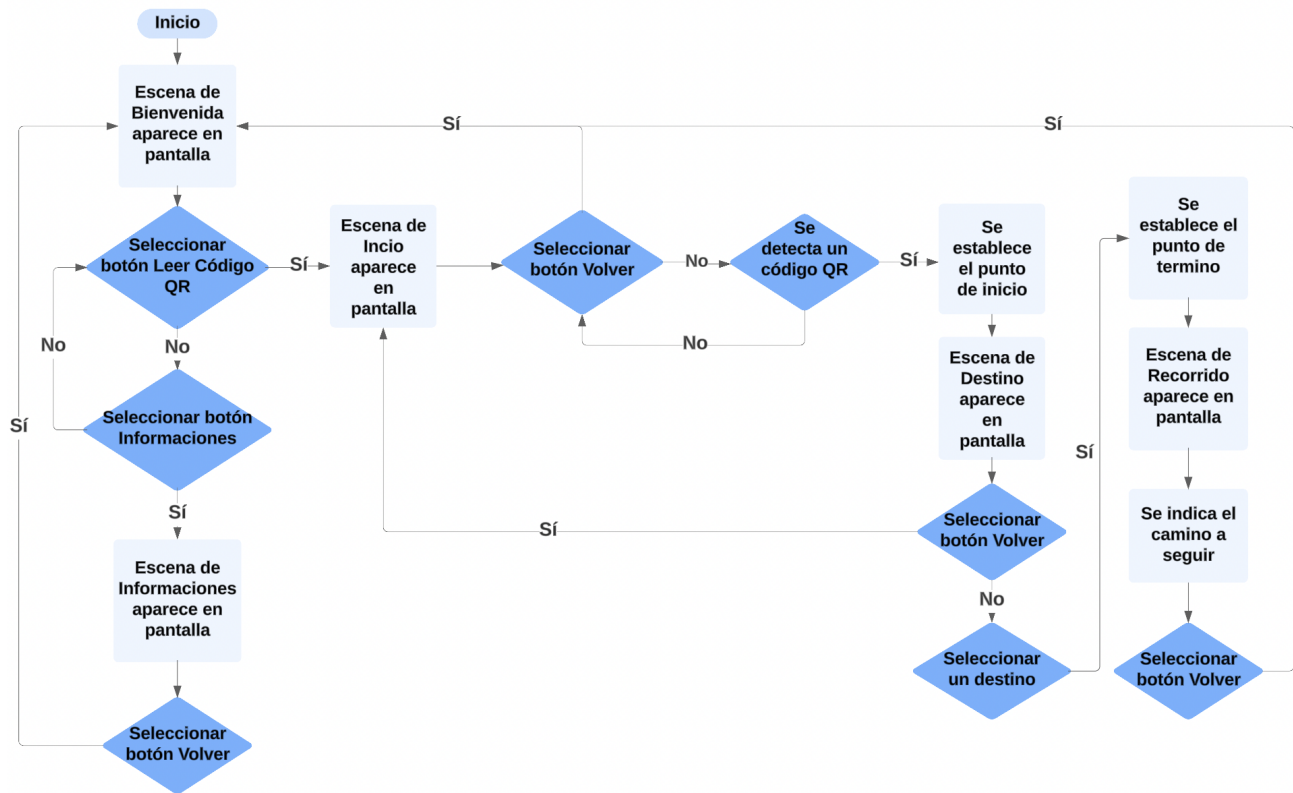


Fig. 13: Diagrama de flujo para la interacción entre escenas.

Cada escena funciona a través de *Scripts* o códigos que se mencionan a continuación.

- **CambioEscena.cs:** Permite hacer un cambio de escena al presionar un botón.
- **BarcoderScanner.cs:** Permite escanear un código QR, establecer un índice según el código identificado, almacenar ese índice de inicio de la ruta con *PlayerPrefs* y cambiar a la escena de "selección de destino" automáticamente.
- **CambioDestino.cs:** Permite identificar el destino seleccionado al presionar un botón en la escena, establece un índice según el destino identificado y lo almacena con *PlayerPrefs*.
- **CreaRuta.cs:** Es el *Script* central de la aplicación móvil y se encarga de generar la ruta entre el inicio del recorrido y el destino seleccionado según se explica en el Anexo A. En el Anexo B se presenta una guía para utilizar el código dentro de Unity.
- **IndicadorDestino.cs:** Permite indicar el final del recorrido mediante un *sprite*.
- **ApagaAR.cs:** Permite deshabilitar la cámara AR activada y desactivar el inicio seleccionado al momento de presionar el botón *Volver* en la escena de *Visualización de la ruta*. Este paso es importante para que nunca esté más de una cámara activada al utilizar la aplicación móvil.

En el esquema de la Figura 14 se presenta cada *script* y dónde son utilizados.



Fig. 14: Esquema de *scripts* en Unity.

3.6 Planos y Rutas

Dada la complejidad de los planos, se ha optado por considerar y focalizar solo los puntos de interés con el fin de simplificar la visualización y la comprensión en el trazado de rutas para que los usuarios puedan llegar a los destinos deseados. De esta forma, se modela el diseño del plano en Unity, aprovechando la simplificación de los puntos de interés.

3.6.1 Infraestructura Unidad de Imagenología

3.6.1.1 Zocalo

En esta zona del hospital hay anexos del CR de Imagenología, como bodegas y oficinas. Estos lugares no son de destino para los pacientes. Sin embargo, se encuentra el SOME de Imagenología. En consecuencia, solo se consideran como rutas habilitadas los accesos a ascensores y escaleras que enlazan con el SOME indicadas en el Anexo C.

3.6.1.2 Piso 1

En esta zona del hospital podemos encontrar las salas de exámenes. Se consideran como lugares de destino las recepciones y las entradas a cada sala de atención del área de Imagenología indicadas en el Anexo D y Anexo E. Desde el C.A.A se puede acceder a un pasillo que tiene conexión con las recepciones y las entradas a las salas de atención.

3.6.2 Rutas

El trazado de las rutas que guiarán a los usuarios a distintos lugares dentro del hospital es un paso esencial a realizar para desarrollar el sistema de posicionamiento. Esto define los límites que tendrá la aplicación móvil en cuanto al desplazamiento del usuario y su accesibilidad hacia las salas de adquisición de exámenes médicos en la Unidad de Imagenología. Se prioriza que los pacientes puedan efectuar la realización del examen sin inconvenientes y sin contratiempos. Este paso contribuirá significativamente a la eficiencia y efectividad general del sistema de posicionamiento intrahospitalario. Para esto, es necesario ir definiendo qué puntos de partidas comunes se tendrán en cuenta dentro del hospital para llegar a los distintos destinos de las 3 unidades: Imagenología, Medicina Nuclear y Neurovascular. La perspectiva tridimensional del hospital permite configurar el área de navegación en Unity donde se genera cada ruta automáticamente entre 2 puntos dentro del plano los cuales pueden estar en diferentes pisos dentro del recinto. A continuación, se mencionan los puntos de interés para las rutas generadas entre los puntos de inicio y de destino.

3.6.2.1 Selección de inicio del trayecto

En el análisis de planos, se identifican lugares estratégicos que funcionan como puntos desde los cuales los pacientes pueden iniciar su trayecto para dirigirse convenientemente hacia distintas secciones especializadas que están a cargo del CR de Imagenología. Esto no solo facilita la orientación de

los usuarios, sino que proporciona puntos de conexión directa con las instalaciones haciéndolas más accesibles a los pacientes.

- **Entrada C.A.A:** La entrada se ubica en la intersección San Martín-Janequeo (ver Fig.15) y cuenta con un alto flujo de personas.

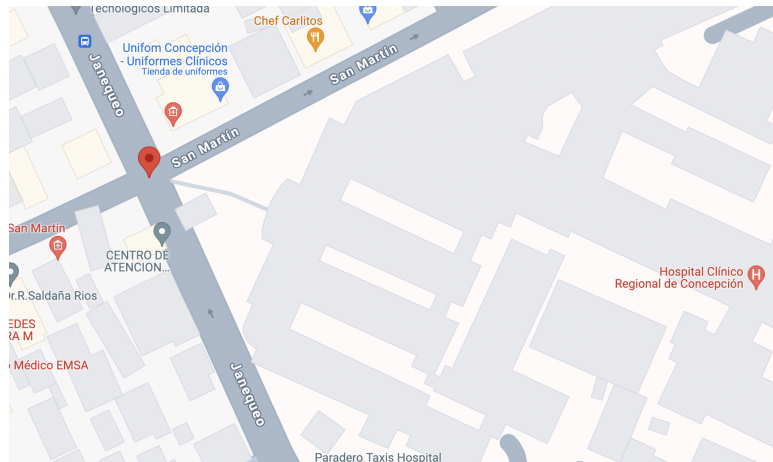


Fig. 15: Ubicación entrada C.A.A en Google Maps.

- **Entrada visitas:** La entrada se ubica en la intersección San Martín-Lautaro (ver Fig.15).

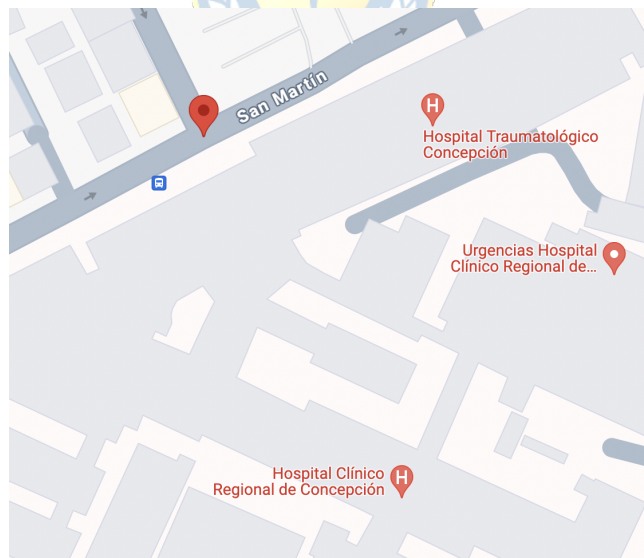


Fig. 16: Ubicación entrada visitas en Google Maps.

- **SOME Imagenología:** Se encuentra ubicado en el zócalo. Desde este punto, los pacientes deben comenzar otro recorrido en caso de tener una cita de exámenes en las áreas de Ecografía, Rayos X, Endoscopia o Mamografía.

3.6.2.2 Selección del destino de trayecto

Los destinos de interés considerados en el desarrollo de la aplicación móvil para la realización de exámenes y toma de horas dentro de la Unidad de Imagenología son:

- Resonancia Magnética
- TAC/Scanner
- Hemodinamia
- Neurovascular/Angiografía
- Mamografía
- Medicina Nuclear
- Ecografía
- Rayos X
- Endoscopia
- TAC/Scanner
- SOME Imagenología



3.7 Seguridad y costos

Otros aspectos a destacar tienen relación con la privacidad del usuario. La mayoría de las tecnologías IPS anteriormente mencionadas necesitan de una infraestructura que involucra que el sistema y el entorno sepa cuál es la posición del usuario lo cual conlleva altos costos de infraestructura. El sistema a desarrollar no requiere almacenar la ubicación del usuario, solo contempla obtener el acceso a la cámara trasera del dispositivo móvil cuando la aplicación este en uso.

En cuanto a los costos asociados, solo se necesita la plataforma Unity para el desarrollo de la app móvil y la plataforma online ME-QR [67] para la creación de códigos QR, ambos se pueden utilizar de forma gratuita.

3.8 Discusión

El desarrollo de la aplicación en Unity ha permitido realizar un avance significativo en una solución que permita orientar a pacientes o usuarios a distintos lugares dentro del hospital. Si bien aún faltan realizar ajustes para que las rutas dispongan de una alineación más precisa con las instalaciones, uno de los aspectos más importantes se ha podido abarcar: generar las indicaciones para seguir la ruta con la menor distancia posible entre un inicio y un destino. Esta funcionalidad es la base de la aplicación móvil y se hizo posible con Vuforia para la utilización de Realidad Aumentada (AR) y AI Navigation

para crear planos y agentes de navegación.

Para garantizar la efectividad de la aplicación móvil es necesario llevar a cabo pruebas en el entorno real del hospital. Esto permite la identificación de posibles falencias y el perfeccionamiento de la precisión de la ruta haciendo las modificaciones que sean necesarias de los diseño de planos en Unity para garantizar una experiencia de usuario más fluida, confiable y sin contratiempos.

Por último, se propone realizar pruebas en otros entornos para que la aplicación sea funcional en todo lugar que se quiera disponer, no solo en un edificio específico.



Capítulo 4. Pruebas de la Aplicación Móvil y Resultados

4.1 Introducción

La realización de pruebas es el siguiente paso luego de haber desarrollado la aplicación móvil y busca asegurar que el sistema funcione correctamente. Esto implica ejecutar una serie de evaluaciones dentro de la aplicación tanto en el simulador de Unity como en el dispositivo móvil con el propósito de corregir posibles errores, para las pruebas en el simulador de Unity se utiliza un MacBook Air M1 (2020) y para las pruebas en dispositivo móvil se utiliza un Samsung Galaxy A51 (2019).

4.2 Escenas, funcionamiento de botones y cambios de escena

Las pruebas para el funcionamiento de botones y cambios de escena se realizaron en el simulador y en el dispositivo móvil con la finalidad de asegurar de que el usuario tenga acceso a todas las funciones de la aplicación.

4.2.1 Resultados

Las escenas disponibles en la aplicación móvil se muestran a continuación.

- **Bienvenida:** Desde la escena, el usuario tiene acceso a la escena de *Informaciones* y a la escena de *Selección de inicio de la ruta* presionando el botón *Informaciones* y el botón *Leer Código QR*, respectivamente (Fig.17).



Fig. 17: Escena de Bienvenida para la Aplicación Móvil.

- **Informaciones:** La escena tiene un botón de retroceso *Volver* para acceder a la escena *Bienvenida*. Además, Se tiene acceso a toda la información disponible deslizando el texto en pantalla (Fig.18).

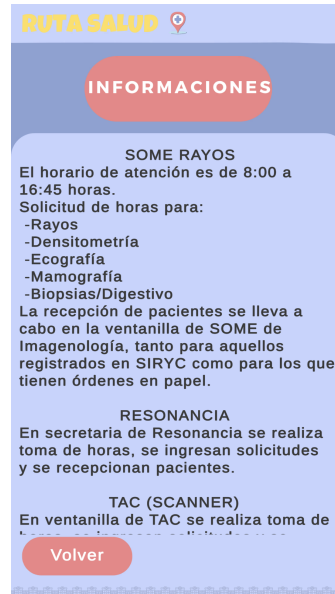


Fig. 18: Escena de Informaciones para la Aplicación Móvil.

- **Selección de inicio de la ruta:** En esta escena se accede a la cámara del dispositivo para escanear un código QR, el usuario puede volver a la escena de *Bienvenida* presionando el botón de retroceso (Fig.19). La escena *Selección de destino* se carga automáticamente cuando uno de los códigos QR registrados es escaneado e identificado.



Fig. 19: Escena de Selección de Inicio para la Aplicación Móvil.

- **Selección de destino:** El menú de destinos está compuesto por una serie de botones que al ser presionados por el usuario se accede automáticamente a la escena de *Visualización de la ruta*. Además, el usuario puede volver a la escena de *Selección de inicio de la ruta* presionando el botón de retroceso (Fig.20).



Fig. 20: Escena de Selección de Destino para la Aplicación Móvil.

- **Visualización de la ruta:** En esta escena se activa la cámara AR y el usuario retrocede a la escena de *Bienvenida* mediante el botón de retroceso (Fig.21).



Fig. 21: Escena de Visualización de la Ruta para la Aplicación Móvil.

4.3 Lectura e identificación de códigos QR

Las pruebas para la lectura e identificación de códigos QR se realizaron en el simulador de Unity, para esto se muestra en consola si algún código QR ha sido escaneado y a cuál corresponde. Ya cargada la escena de *Selección de destino*, se presiona el botón de retroceso, se vuelve a escanear otro código QR y así sucesivamente hasta que se hayan escaneado correctamente todos los códigos disponibles.

4.3.1 Resultados

En la Figura 21 se muestra en la consola de Unity los códigos QR que han sido escaneados y sus respectivos índices.



Fig. 22: Identificación de códigos QR en Unity.

4.4 Identificación del destino seleccionado

Las pruebas para la identificación del destino seleccionado se realizaron en el simulador de Unity mostrando en consola el destino seleccionado. Ya cargada la escena de *Visualización de la ruta*, se presiona el botón de retroceso, se llega a la escena de *Bienvenida*, luego se escanea un código QR y finalmente se selecciona otro destino, esta acción se repite hasta que todos los destinos seleccionados se muestren correctamente en la consola.

4.4.1 Resultados

En la Figura 23 se muestra en consola los índices de los destinos seleccionados. En la aplicación móvil el usuario puede decidir cambiar de destino siempre y cuando se identifique nuevamente un código QR. Sin embargo, al salir de la escena visualización de la ruta existe la posibilidad de que se presente un error proveniente de Vuforia al retroceder y volver a escanear un código QR de forma repetitiva lo que puede llegar a dificultar el correcto funcionamiento de la aplicación (ver Fig.24).

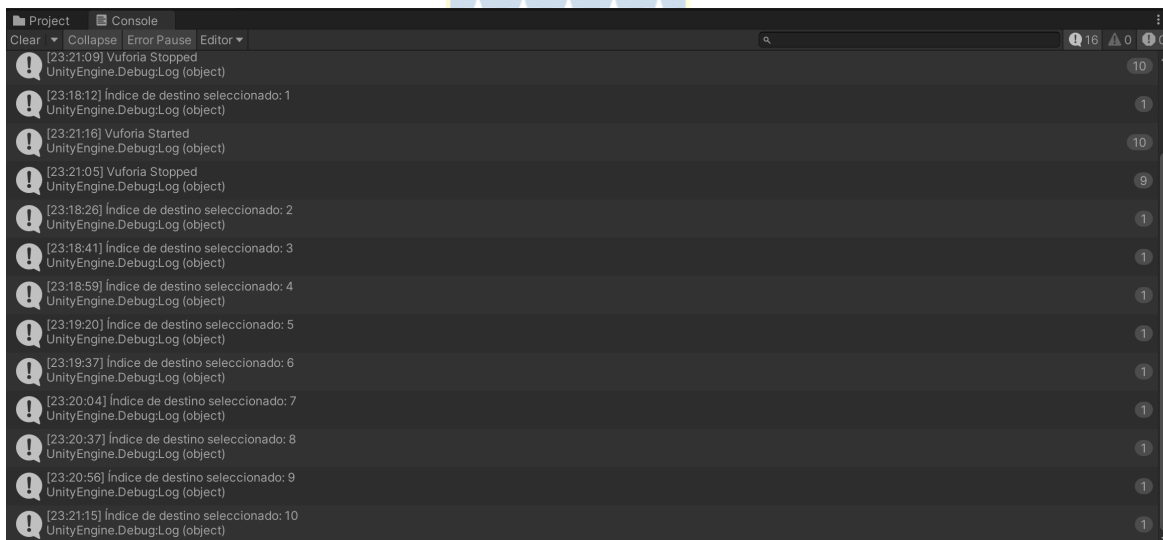


Fig. 23: Identificación del destino seleccionado en Unity.

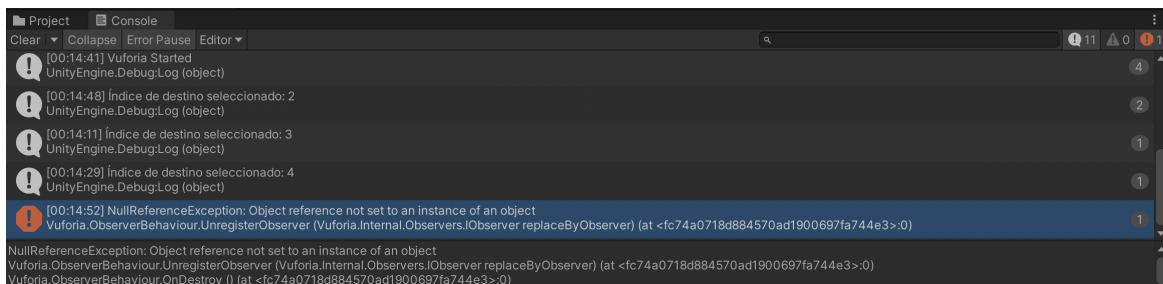


Fig. 24: Error en consola para la lectura de códigos QR en Unity.

4.5 Seguimiento de rutas dentro de la Unidad de Imagenología

Las pruebas son realizadas dentro del recinto hospitalario utilizando el dispositivo móvil. Para esto se evalúan todas las combinaciones posibles entre los inicios y los destinos y de esta forma abarcar todos los trayectos que el usuario tiene disponible dentro de la aplicación móvil.

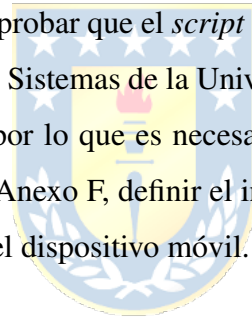
4.5.1 Resultados

Si bien la aplicación es funcional dentro del entorno de Unity, al comenzar las pruebas de los recorridos dentro del hospital se presentan dificultades:

- El indicador de trayecto tarda en aparecer en pantalla.
- Desaparición repentina del indicador de trayecto.
- El indicador no aparece luego de haber cambiado el lugar de destino anteriormente establecido.

4.6 Pruebas de funcionamiento dentro de otro establecimiento

Esta prueba tiene por objetivo comprobar que el *script* central *CreaRuta.cs* es funcional para otros lugares, en este caso para el edificio de Sistemas de la Universidad de Concepción. Para esto se utiliza la escena de *Seguimiento de la ruta*, por lo que es necesario diseñar la estructura tridimensional de los planos del lugar presentados en el Anexo F, definir el inicio de la ruta, definir el destino dentro de Unity y luego probar la aplicación en el dispositivo móvil.



4.6.1 Resultados

En la Figura 25 se observa cómo es posible llegar a un destino deseado siguiendo el indicador de la ruta dentro del Edificio de Sistemas. El final del recorrido queda señalado por el *sprite* que se ve en la puerta de entrada, dando a conocer que el recorrido ha concluido.

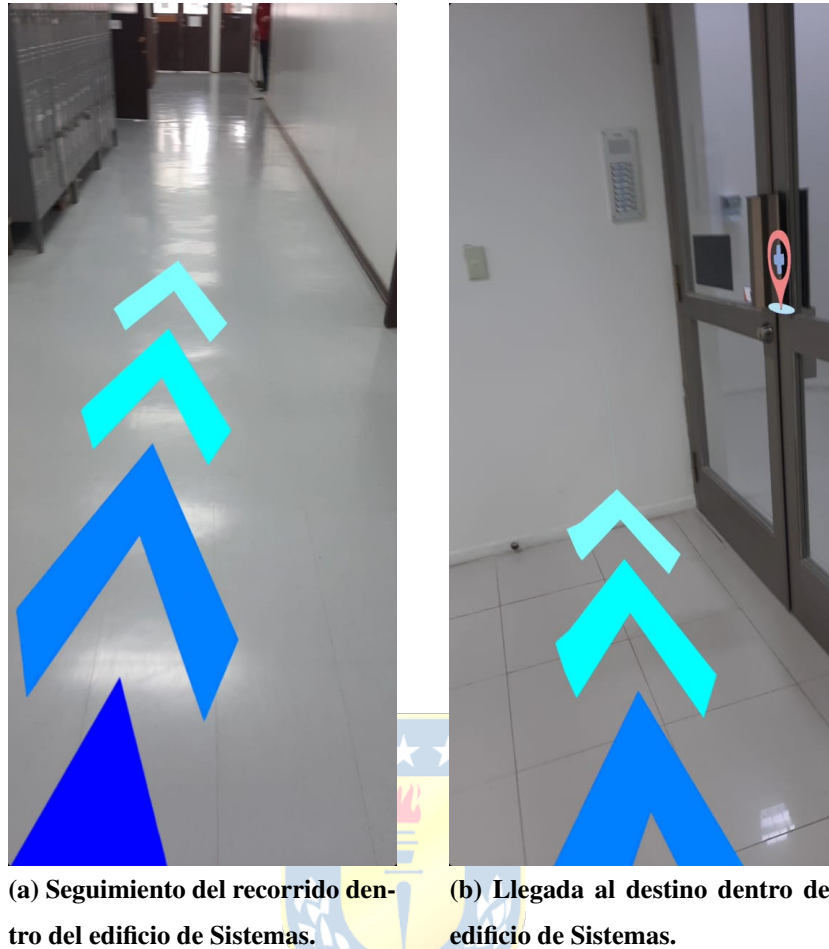


Fig. 25: Prueba de recorrido dentro del edificio de Sistemas.

4.7 Discusión

En cuanto a los errores presentados al realizar la prueba de los cambios de destino y el cómo abordarlos, una solución puede ser modificar la forma de establecer el punto de inicio cambiando la lectura de códigos QR por un menú de botones de la ruta teniendo en cuenta que el usuario puede seleccionar el inicio de la ruta y el final de la ruta mediante distintos métodos. En este caso la aplicación móvil hace uso de códigos QR y botones pero al querer volver a escanear otro código QR existe un error proveniente desde Vuforia.

Otra solución consistiría en que por cada inicio de la ruta sea creada una escena con la estructura del hospital para que solo exista una cámara AR en las escenas. Esto considerando que para el caso pruebas de funcionamiento dentro de otro establecimiento se utilizó una sola cámara y no existieron dificultades al hacer el recorrido.

Fuera de los problemas mencionados, se demuestra que la aplicación es correctamente funcional y cumple con los requerimientos de interfaz de usuario.



Capítulo 5. Conclusión

5.1 Conclusiones

El uso de un sistema de posicionamiento móvil en entornos hospitalarios representa un avance significativo para mejorar la atención médica. Mientras que las herramientas de posicionamiento han sido aplicadas en diversos contextos de la salud, la integración de la tecnología IPS con códigos QR para guiar a pacientes dentro de los recintos hospitalarios es una propuesta con poca exploración y desarrollo.

El sistema de localización desarrollado tiene por objetivo otorgar una forma efectiva y de fácil uso para ubicar distintas áreas dentro del hospital. El hecho de proponer códigos QR como puntos estratégicos de partida mejora el seguimiento de las rutas hacia el destino del usuario. Además, la integración de la realidad aumentada mediante la cámara del dispositivo móvil otorga una orientación en tiempo real dando una mejor experiencia a los pacientes. Otra ventaja importante es que el sistema propuesto considera la privacidad del usuario, diferenciándolo de otras tecnologías basadas en IPS.

Como resultado, se logró desarrollar un sistema accesible, seguro y de bajo costo que otorga una alternativa viable para guiar al paciente hacia algún lugar de interés dentro del hospital sin la necesidad de tener acceso a datos personales o a señal telefónica. Sin embargo, aún se pueden seguir haciendo mejoras para otorgarle a los usuarios una aplicación móvil más completa especialmente al considerar no siempre tiene un funcionamiento correcto.

5.2 Trabajo a Futuro

Con respecto a la aplicación desarrollada, es posible integrar mejoras tanto en el sistema como en el diseño. En el sistema se podría identificar la fuente del error que surge de las cámaras AR con Vuforia y optimizar la aplicación. En cuanto al diseño, se podría cambiar el indicador de la ruta, hacer cambios en la estructura visual de la aplicación móvil o incluir un minimapa en pro de mejorar la experiencia de los pacientes.

Otro aspecto a considerar, sería integrar más herramientas tales como una guía auditiva para que la aplicación sea más inclusiva. También, es posible ampliar el alcance de la aplicación móvil dentro del hospital añadiendo todos los destinos que sean requeridos.

Por último, se plantea utilizar un Generador de Objetivos de Área (Area Target Generator, ATG) de Vuforia [68] en conjunto con Unity y Matterport™ [69]. A grandes rasgos consiste en escanear el entorno para generar un modelo digital utilizando Matterport™, luego crear el ATG e importarlo a

Unity. Esta metodología es muy utilizada para obtener sistemas de navegación interior confiables con AR pero que precisan de una mayor inversión de tiempo para su desarrollo [70].



Bibliografía

- [1] S. Grimes, “Clinical engineers: stewards of healthcare technologies,” *IEEE Engineering in Medicine and Biology Magazine*, vol. 23, no. 3, pp. 56–58, 2004.
- [2] F. Zafari, A. Gkelias, and K. K. Leung, “A survey of indoor localization systems and technologies,” *IEEE Communications Surveys Tutorials*, vol. 21, no. 3, pp. 2568–2599, 2019.
- [3] M. A. Cheema, “Indoor location-based services: Challenges and opportunities,” *Association for Computing Machinery*, vol. 10, no. 2, 2018.
- [4] K. Al Nuaimi and H. Kamel, “A survey of indoor positioning systems and algorithms,” in *2011 International Conference on Innovations in Information Technology*, Abu Dhabi, United Arab Emirates, 2011, pp. 185–190.
- [5] Y. Shi, W. Shi, X. Liu, and X. Xiao, “An RSSI classification and tracing algorithm to improve trilateration-based positioning,” *Sensors*, vol. 20, no. 15, 2020.
- [6] L. Batistić and M. Tomic, “Overview of indoor positioning system technologies,” in *2018 41st International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, Opatija, Croatia, 2018, pp. 473–478.
- [7] J. Dong, Z. Lian, J. Xu, and Z. Yue, “An improved adaptive sparrow search algorithm for tdoa-based localization,” *ISPRS International Journal of Geo-Information*, vol. 12, no. 8, 2023.
- [8] Y. Sartayeva and H. C. Chan, “A survey on indoor positioning security and privacy,” *Computers Security*, vol. 131, 2023.
- [9] S. Kim, S. Ha, A. Saad, and J. Kim, “Indoor positioning system techniques and security,” in *2015 Forth International Conference on e-Technologies and Networks for Development (ICeND)*, Lodz, Polonia, 2015, pp. 1–4.
- [10] G. M. Mendoza, J. Torres, and J. Huerta, “A meta-review of indoor positioning systems,” *Sensors*, vol. 19, no. 20, 2019.
- [11] H. Zou, M. Jin, H. Jiang, L. Xie, and C. J. Spanos, “WinIPS: WiFi-based non-intrusive indoor positioning system with online radio map construction and adaptation,” *IEEE Transactions on Wireless Communications*, vol. 16, no. 12, pp. 8118–8130, 2017.

- [12] W. Ali, A. Kareem, and M. Jasim, "Survey on wireless indoor positioning systems," *Cihan University-Erbil Scientific Journal*, vol. 3, no. 2, pp. 42–47, 2019.
- [13] H. Zou, H. Wang, L. Xie, and Q.-S. Jia, "An RFID indoor positioning system by using weighted path loss and extreme learning machine," in *2013 IEEE 1st International Conference on Cyber-Physical Systems, Networks, and Applications (CPSNA)*, Taipei, Taiwan, 2013, pp. 66–71.
- [14] G. Shipkovenski, T. Kalushkov, E. Petkov, and V. Angelov, "A beacon-based indoor positioning system for location tracking of patients in a hospital," in *2020 International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA)*, Ankara, Turkey, 2020, pp. 1–6.
- [15] D. Bragin, E. Kostyuchenko, V. Faerman, A. Kobzev, and I. Sidorov, "Comparison of technologies of local patient positioning," *International Journal of Advanced Research in Engineering and Technology (IJARET)*, vol. 12, no. 1, pp. 362–386, 2021.
- [16] C. Medina, J. C. Segura, and De la Torre, "Ultrasound indoor positioning system based on a low-power wireless sensor network providing sub-centimeter accuracy," *Sensors*, vol. 13, no. 3, pp. 3501–3526, 2013.
- [17] F. Che, Q. Z. Ahmed, P. I. Lazaridis, P. Sureephong, and T. Alade, "Indoor positioning system (IPS) using ultra-wide bandwidth (UWB) for industrial internet of things (IIoT)," *Sensors*, vol. 23, no. 12, 2023.
- [18] J. Yoon and S. Kim, "Practical and accurate indoor localization system using deep learning," *Sensors*, vol. 22, no. 18, 2022.
- [19] H.-H. Hsu, W.-J. Peng, T. K. Shih, T.-W. Pai, and K. L. Man, "Smartphone indoor localization with accelerometer and gyroscope," in *2014 17th International Conference on Network-Based Information Systems*, Salerno, Italy, 2014, pp. 465–469.
- [20] C.-H. Hsu and C.-H. Yu, "An accelerometer based approach for indoor localization," in *2009 Symposia and Workshops on Ubiquitous, Autonomic and Trusted Computing*, Brisbane, QLD, Australia, 2009, pp. 223–227.
- [21] E. N. Nwobodo-Anyadiiegwu, M. N. Ditend, and A. K. Lumbwe, "The benefits and challenges of implementing smart hospital projects: A systematic review," in *2022 IEEE 28th International*

Conference on Engineering, Technology and Innovation (ICE/ITMC) and 31st International Association For Management of Technology (IAMOT) Joint Conference, Nancy, Francia, 2022, pp. 1–7.

- [22] L.-C. Chu, C.-L. Lee, and C.-J. Wu, “Applying QR code technology to facilitate hospital medical equipment repair management,” in *2012 International Conference on Control Engineering and Communication Technology*, 2012, pp. 856–859.
- [23] C. T. Karia, A. Hughes, and S. Carr, “Uses of quick response codes in healthcare education: a scoping review,” *BMC Medical Education*, vol. 19, 2019.
- [24] F. Bruwer and M. T. Booysen, “Vehicle acceleration estimation using smartphone-based sensors,” 2015.
- [25] V. Renaudin and C. Combettes, “Magnetic, acceleration fields and gyroscope quaternion (magnyq)-based attitude estimation with smartphone sensors for indoor pedestrian navigation,” *Sensors*, vol. 14, no. 12, pp. 22 864–22 890, 2014.
- [26] A. Manos, T. Hazan, and I. Klein, “Walking direction estimation using smartphone sensors: A deep network-based framework,” *IEEE Transactions on Instrumentation and Measurement*, vol. 71, pp. 1–12, 2022.
- [27] T.-H. Tan, J.-Y. Wu, S.-H. Liu, and M. Gochoo, “Human activity recognition using an ensemble learning algorithm with smartphone sensor data,” *Electronics*, vol. 11, no. 3, 2022.
- [28] P. Kulkarni, R. Kirkham, and R. McNaney, “Opportunities for smartphone sensing in e-health research: A narrative review,” *Sensors*, vol. 22, no. 10, 2022.
- [29] I. Faisal, T. Purboyo, and A. Ansori, “A review of accelerometer sensor and gyroscope sensor in imu sensors on motion capture,” *Journal of Engineering and Applied Sciences*, vol. 15, pp. 826–829, 2019.
- [30] C. Huang, S. He, Z. Jiang, C. Li, Y. Wang, and X. Wang, “Indoor positioning system based on improved pdr and magnetic calibration using smartphone,” in *2014 IEEE 25th Annual International Symposium on Personal, Indoor, and Mobile Radio Communication (PIMRC)*, 2014, pp. 2099–2103.

- [31] C. K. Schindhelm, F. Gschwandtner, and M. Banholzer, “Usability of apple iphones for inertial navigation systems,” in *2011 IEEE 22nd International Symposium on Personal, Indoor and Mobile Radio Communications*, 2011, pp. 1254–1258.
- [32] V. M. N. Passaro, A. Cuccovillo, L. Vaiani, M. De Carlo, and C. E. Campanella, “Gyroscope technology and applications: A review in the industrial perspective,” *Sensors*, vol. 17, no. 10, 2017.
- [33] A. Developers, “Sensores de movimiento,” Android Developers. [Online]. Available: https://developer.android.com/guide/topics/sensors/sensors_motion?hl=es-419
- [34] N. M. Alzahrani, “Augmented reality: A systematic review of its benefits and challenges in e-learning contexts,” *Applied Sciences*, vol. 10, no. 16, 2020.
- [35] P. Cipresso, I. A. C. Giglioli, M. A. Raya, and G. Riva, “The past, present, and future of virtual and augmented reality research: a network and cluster analysis of the literature,” *Frontiers in Psychology*, vol. 9, 2018.
- [36] C. Bichlmeier, B. Ockert, S. M. Heining, A. Ahmadi, and N. Navab, “Stepping into the operating theater: Arav — augmented reality aided vertebroplasty,” in *2008 7th IEEE/ACM International Symposium on Mixed and Augmented Reality*, 2008, pp. 165–166.
- [37] T. Blum, S. M. Heining, O. Kutter, and N. Navab, “Advanced training methods using an augmented reality ultrasound simulator,” in *2009 8th IEEE International Symposium on Mixed and Augmented Reality*, 2009, pp. 177–178.
- [38] M. Lorenz, S. Knopp, and P. Klimant, “Industrial augmented reality: Requirements for an augmented reality maintenance worker support system,” in *2018 IEEE International Symposium on Mixed and Augmented Reality Adjunct (ISMAR-Adjunct)*, 2018, pp. 151–153.
- [39] J. Zhou, Z. Xu, H. Yan, B. Gao, O. Yang, and Z. Zhao, “Ar creator: A mobile application of logic education based on ar,” in *2020 International Conference on Virtual Reality and Visualization (ICVRV)*, 2020, pp. 379–380.
- [40] Y.-T. Lin, F.-N. Wu, Z.-Y. Tsai, and Y.-S. Huang, “Development of an ar food education system to support elementary school nutrition education,” in *2023 IEEE International Conference on Advanced Learning Technologies (ICALT)*, 2023, pp. 242–244.

- [41] A. Hussain, H. Shakeel, F. Hussain, N. Uddin, and T. Ghouri, "Unity game development engine: A technical survey," *University of Sindh Journal of Information and Communication Technology*, vol. 4, no. 2, 2020.
- [42] Unity Technologies, "Scenes," 2024, [Online]. Available: <https://docs.unity3d.com/Manual/CreatingScenes.html> [Accessed: 10-06-2024].
- [43] Unity Technologies, "GameObjects," 2024, [Online]. Available: <https://docs.unity3d.com/es/530/Manual/GameObjects.html> [Accessed: 10-06-2024].
- [44] Unity Technologies, "Creando y usando scripts," 2024, [Online]. Available: <https://docs.unity3d.com/es/530/Manual/CreatingAndUsingScripts.html> [Accessed: 10-06-2024].
- [45] Unity Technologies, "Button," 2024, [Online]. Available: <https://docs.unity3d.com/es/2017.4/ScriptReference/UI.Button.html> [Accessed: 10-06-2024].
- [46] Unity Technologies, "PlayerPrefs," 2024, [Online]. Available: <https://docs.unity3d.com/ScriptReference/PlayerPrefs.html> [Accessed: 10-06-2024].
- [47] Unity Technologies, "Line Renderer component," 2024, [Online]. Available: <https://docs.unity3d.com/Packages/com.unity.ai.navigation@1.1/manual/NavMeshSurface.html> [Accessed: 12-06-2024].
- [48] Unity Technologies, "Position Constraints," 2024, [Online]. Available: <https://docs.unity3d.com/Manual/class-PositionConstraint.html> [Accessed: 12-06-2024].
- [49] Unity Technologies, "Sprites," 2024, [Online]. Available: <https://docs.unity3d.com/es/530/Manual/Sprites.html> [Accessed: 12-06-2024].
- [50] Unity Technologies, "AI Navigation," 2024, [Online]. Available: <https://docs.unity3d.com/Packages/com.unity.ai.navigation@2.0/manual/index.html> [Accessed: 12-06-2024].
- [51] Unity Technologies, "Inner Workings of the Navigation System," 2024, [Online]. Available: <https://docs.unity3d.com/Packages/com.unity.ai.navigation@2.0/manual/NavInnerWorkings.html> [Accessed: 12-06-2024].
- [52] Unity Technologies, "NavMesh Surface," 2024, [Online]. Available: <https://docs.unity3d.com/Packages/com.unity.ai.navigation@1.1/manual/NavMeshSurface.html> [Accessed: 12-06-2024].

- [53] Unity Technologies, “NavMeshPath,” 2024, [Online]. Available: <https://docs.unity3d.com/ScriptReference/AI.NavMeshPath.html> [Accessed: 14-06-2024].
- [54] Unity Technologies, “NavMesh.CalculatePath,” 2024, [Online]. Available: <https://docs.unity3d.com/ScriptReference/AI.NavMesh.CalculatePath.html> [Accessed: 14-06-2024].
- [55] Vuforia Engine, “Getting Started with Vuforia Engine in Unity,” 2024, [Online]. Available: <https://developer.vuforia.com/library/getting-started/getting-started-vuforia-engine-unity> [Accessed: 09-06-2024].
- [56] Google, “ARCore,” [Software Development Kit], Google LLC, 2018, [Online]. Available: <https://developers.google.com/ar/develop>.
- [57] Vuforia Engine, “Enable Device Tracking in Unity,” 2024, [Online]. Available: <https://developer.vuforia.com/library/device-tracking/enable-device-tracking-unity> [Accessed: 09-06-2024].
- [58] A. R. Khairuddin, M. S. Talib, and H. Haron, “Review on simultaneous localization and mapping (slam),” in *2015 IEEE International Conference on Control System, Computing and Engineering (ICCSCCE)*, 2015, pp. 85–90.
- [59] H. Durrant-Whyte and T. Bailey, “Simultaneous localization and mapping: part i,” *IEEE Robotics Automation Magazine*, vol. 13, no. 2, pp. 99–110, 2006.
- [60] Vuforia Engine, “Vuforia Fusion,” 2024, [Online]. Available: <https://developer.vuforia.com/library/environments/vuforia-fusion> [Accessed: 09-06-2024].
- [61] E. M. Lee, I. Wee, T. Kim, and D. H. Shim, “Comparison of visual inertial odometry using flightgoggles simulator for uav,” in *2019 19th International Conference on Control, Automation and Systems (ICCAS)*, 2019, pp. 1166–1169.
- [62] L.-C. Chu, C.-L. Lee, and C.-J. Wu, “Applying QR code technology to facilitate hospital medical equipment repair management,” in *2012 International Conference on Control Engineering and Communication Technology*, Shenyang, China, 2012, pp. 856–859.
- [63] S. Sharara and S. Radia, “Quick response (QR) codes for patient information delivery: A digital innovation during the coronavirus pandemic,” *Journal of Orthodontics*, vol. 49, pp. 89–97, 2021.

- [64] S. Dube, S. Sihwa, T. Nyathi, and K. Sibanda, "QR code based patient medical health records transmission: Zimbabwean case," in *2015 Informing Science IT Education Conference (InSITE)*, Tampar, USA, 2015, pp. 521–530.
- [65] Y.-W. Ti, S.-K. Chen, and W.-C. Wu, "A new visual cryptography-based QR code system for medication administration," *Mobile Information Systems*, vol. 2020, pp. 1–10, 2020.
- [66] V. Uzun and S. Bilgin, "Evaluation and implementation of QR code identity tag system for healthcare in turkey," *SpringerPlus*, vol. 5, 2016.
- [67] Me Team LTD, "ME-QR," 2021, [Online]. Availale: <https://me-qr.com>.
- [68] Vuforia Engine, "Area Target Generator User Guide," 2024, [Online]. Availale: <https://developer.vuforia.com/library/creating-area-targets/area-target-generator-user-guide> [Accessed: 09-06-2024].
- [69] Matterport, "Using Matterport for Android," 2023, [Online]. Availale: https://support.matterport.com/s/article/Matterport-for-Android?language=en_US&categfilter=Matterport_for_Mobile&parentCategoryLabel=Matterport_for_Mobile [Accessed: 09-06-2024].
- [70] R. Bhattacharyya and U. Riedlinger, "Minar: Mobile indoor navigation using augmented reality," in *GI VR/AR Workshop 2023*, Cologne, Germany, 2023.

Anexo A FUNCIONAMIENTO SCRIPT CENTRAL

El esquema presentado en la Figura 26 describe las funciones que componen el *script CreaRuta.cs*. La función *Start()* se encarga de acceder al índice obtenido en *BarcodeScanner.cs* para habilitar una de las cámaras AR de la escena y almacenar el *LineRenderer* en la variable *Ruta*. La función *Update()* llama a la función *DrawPath(NavMeshAgent NV)* en cada *frame*. Finalmente, la función *DrawPath(NavMeshAgent NV)* toma como argumento el *NavMeshAgent*, accede al índice del destino seleccionado obtenido en *CambioDestino.cs* y define el destino como el *targetObject* de la ruta. Luego, calcula el *path* entre el *NavMeshAgent* y el *targetObject* para obtener las esquinas del *path* en forma de coordenadas y que la *Ruta* tenga las mismas esquinas. El procedimiento de pasar las esquinas del *path* a la *Ruta* es importante ya que el *path* solo contiene una lista de puntos pero el *LineRenderer* tiene el atributo de conectar estos puntos mediante rectas visibles en escena. En la Figura 27 se hace una comparación con respecto a la forma que adquiere el *LineRenderer* con y sin el *path*. En la recta azul se observa que la *Ruta* solo esta compuesta por 2 puntos que conectan los 2 objetos dentro de la escena y atraviesa lugares que no están dentro del *NavMesh Surface*.

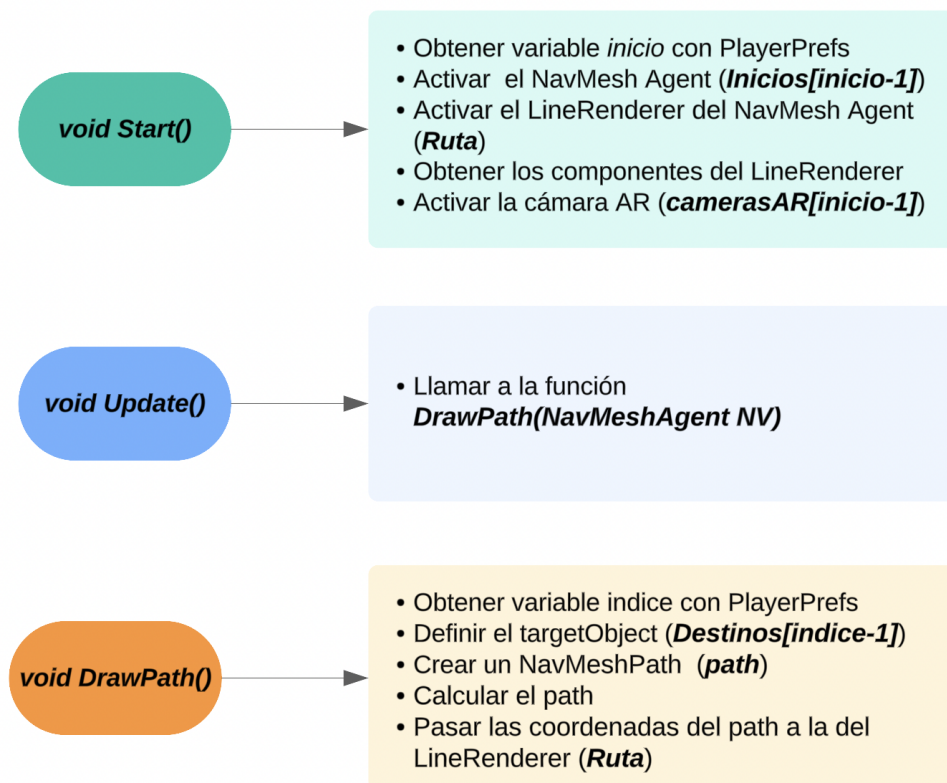
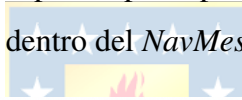


Fig. 26: Esquema del *script CreaRuta.cs*.

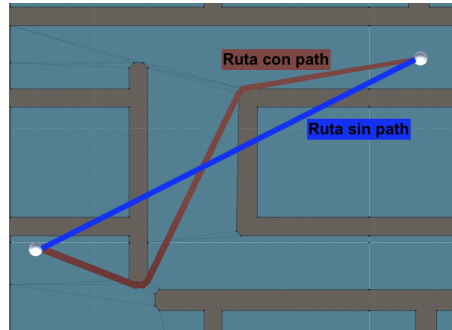


Fig. 27: Creación de una ruta con y sin *path*.

Para el *script CreaRuta.cs* los índices y los elementos se seleccionan en función del inicio del recorrido y en el final del recorrido según se indica en las tablas 2 y 3, respectivamente.

Tabla 2: Activación de la cámara AR y del *NavMeshAgent* a partir del inicio del recorrido

Inicio del recorrido	Índice del código QR	Cámara AR activada <i>camerasAR[inicio-1]</i>	<i>NavMeshAgent</i> activado <i>Inicios[inicio-1]</i>
Entrada C.A.A	<i>inicio = 1</i>	<i>camerasAR[0] = ARCamera1</i>	<i>Inicios[0] = Inicio1</i>
Entrada Visitas	<i>inicio = 2</i>	<i>camerasAR[1] = ARCamera2</i>	<i>Inicios[1] = Inicio2</i>
SOME Imagenología	<i>inicio = 3</i>	<i>camerasAR[2] = ARCamera3</i>	<i>Inicios[2] = Inicio2</i>

Tabla 3: Selección del *targetObject* a partir del final del recorrido

Final del recorrido	Índice del destino	<i>targetObject</i> <i>Destinos[indice-1]</i>
Resonancia Magnética	<i>indice = 1</i>	<i>Indices[0] = ResonanciaMagnetica</i>
SOME Imagenología	<i>indice = 2</i>	<i>Indices[1] = SomeImagenologia</i>
Scanner	<i>indice = 3</i>	<i>Indices[2] = Scanner</i>
Hemodinamia	<i>indice = 4</i>	<i>Indices[3] = Hemodinamia</i>
Endoscopia	<i>indice = 5</i>	<i>Indices[4] = Endoscopia</i>
RayosX	<i>indice = 6</i>	<i>Indices[5] = Rayos</i>
Mamografía	<i>indice = 7</i>	<i>Indices[6] = Mamografia</i>
Neurovascular	<i>indice = 8</i>	<i>Indices[7] = Neurovascular</i>
Medicina Nuclear	<i>indice = 9</i>	<i>Indices[8] = MedicinaNuclear</i>
Ecografía	<i>indice = 10</i>	<i>Indices[9] = Ecografia</i>

Anexo B. GUÍA PARA LA IMPLEMENTACIÓN DEL SISTEMA

La implementación dentro de Unity de los *scripts CreaRuta.cs* y *ApagaAR.cs* se puede realizar de acuerdo a las siguientes indicaciones para utilizar el sistema de navegación en una edificación definida.

1. Crear una escena y construir el modelo tridimensional de los planos.
2. Crear una *NavMesh Surface*.
3. Añadir *AR Cameras* y posicionar una en cada inicio de los recorridos. Desactivar cada cámara (Fig.28).

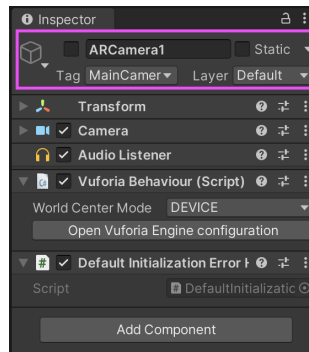


Fig. 28: Configuración cámaras AR en Unity.

4. Añadir *Empty Objects*, posicionar uno en cada inicio de los recorridos. Configurar cada uno tal como se muestra en la figura 29.

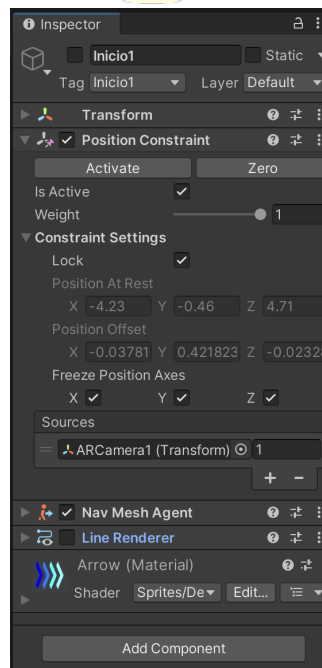


Fig. 29: Configuración de los *Empty Objects* para el inicio del trayecto en Unity.

5. Añadir *Objects* y posicionar cada uno al final de cada recorrido.
6. Crear un *Empty Object* y agregar el *script CreaRuta.cs* como componente. Llenar los elementos solicitados en el *script* con *NavMesh Agents*, *Objects* de destino y *AR Cameras* (Fig.30).



Fig. 30: Integración de elementos para el *script CreaRuta.cs* en Unity.

7. Crear un *Empty Object* y agregar el *script ApagaAR.cs* como componente. Llenar los elementos solicitados en el *script* con *NavMesh Agents* y *AR Cameras*.
8. Crear un *Empty Object* y agregar el *script CambioEscena.cs* como componente (Fig.31).

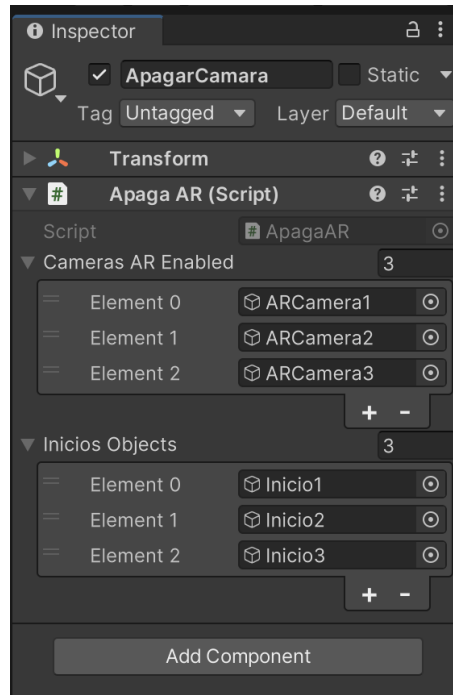


Fig. 31: Integración de elementos para el script *ApagaAR.cs* en Unity.

9. Crear un *Button* para salir de la escena y adjuntar en el componente *On Click()* los *Empty Objects* de los scripts *ApagaAR.cs* y *CambioEscena.cs* (Fig.32).

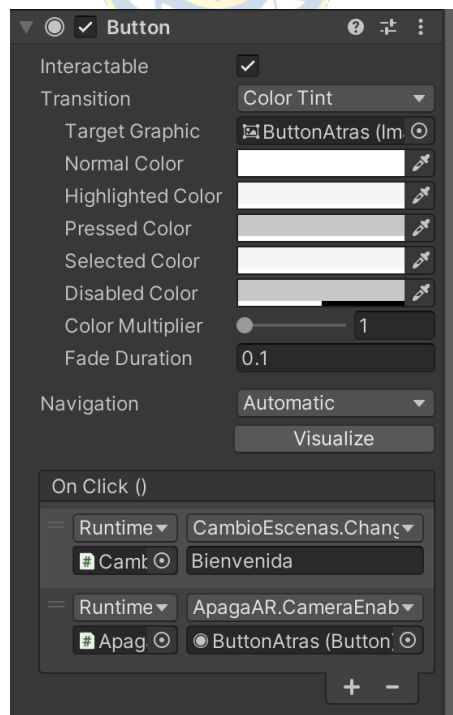


Fig. 32: Configuración del botón de retroceso en Unity.

Anexo C. PLANO ZOCALO C.A.A

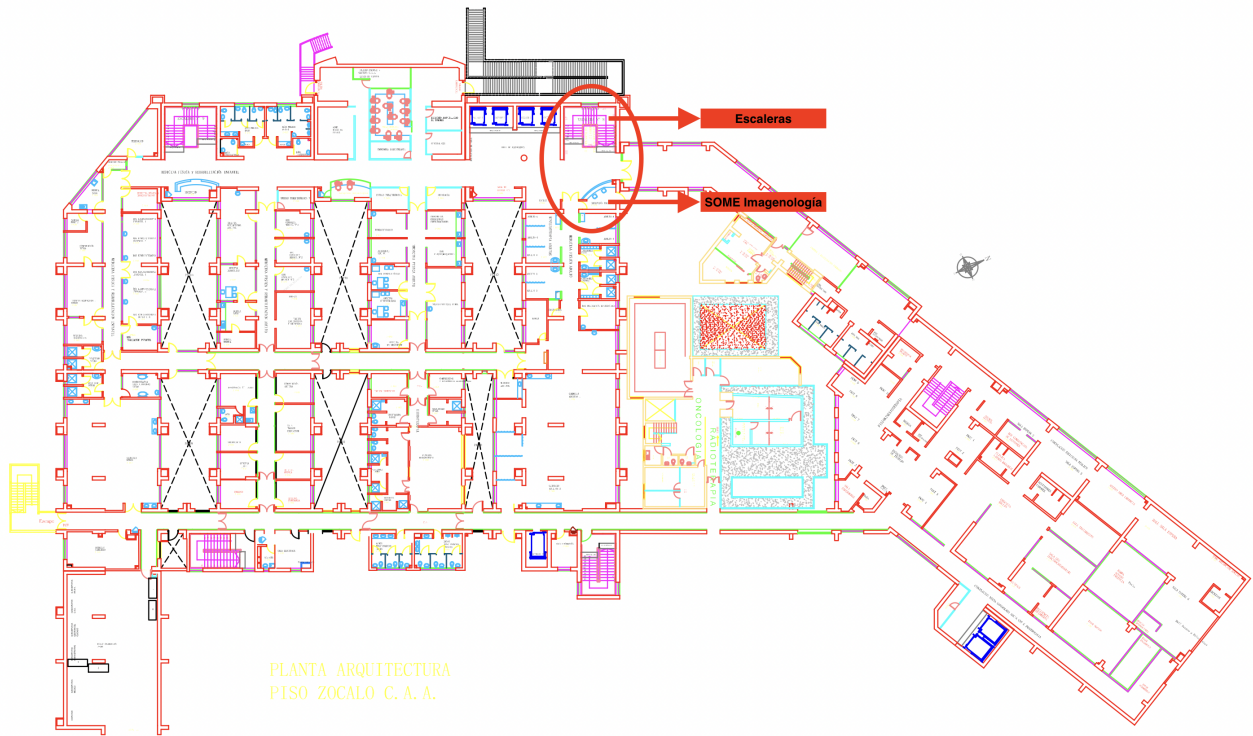


Fig. 33: Plano Zocalo C.A.A, HGGB.



Anexo D. PLANO PRIMER PISO C.A.A

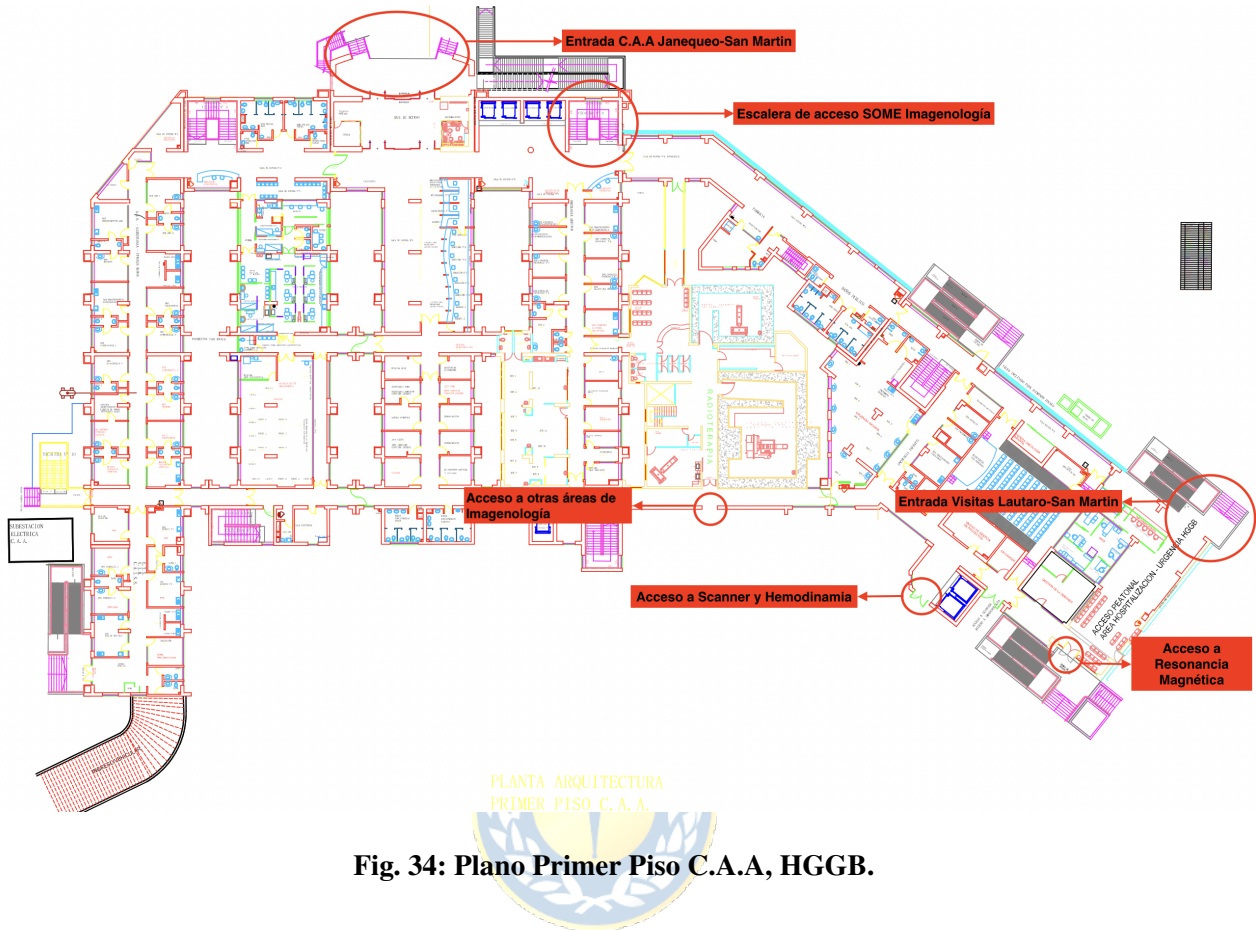


Fig. 34: Plano Primer Piso C.A.A, HGGB.

Anexo E. PLANO PRIMER PISO MONOBLOCK

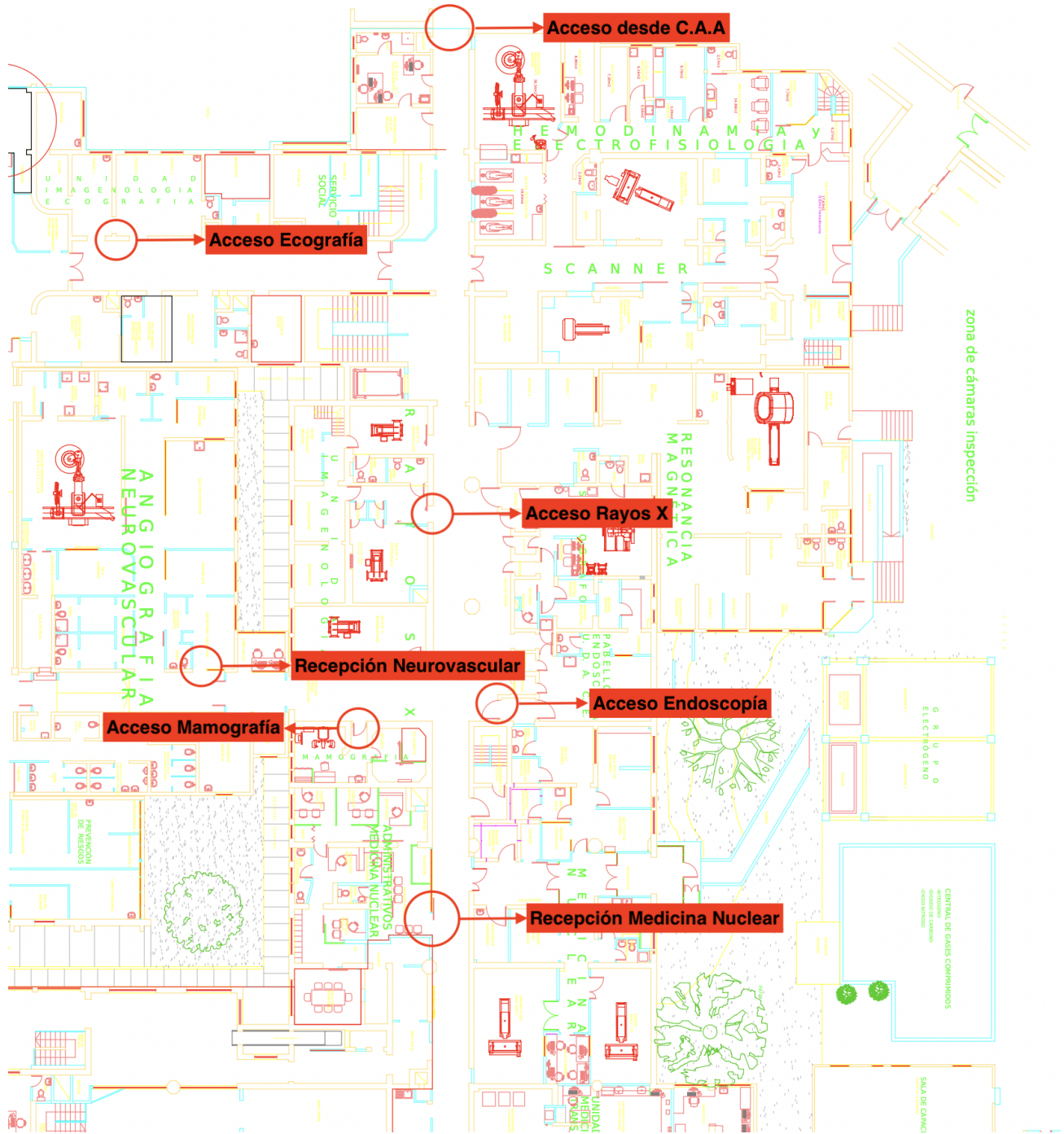


Fig. 35: Plano Primer Piso Monoblock, HGGB.

Anexo F. PLANO EDIFICIO DE SISTEMAS

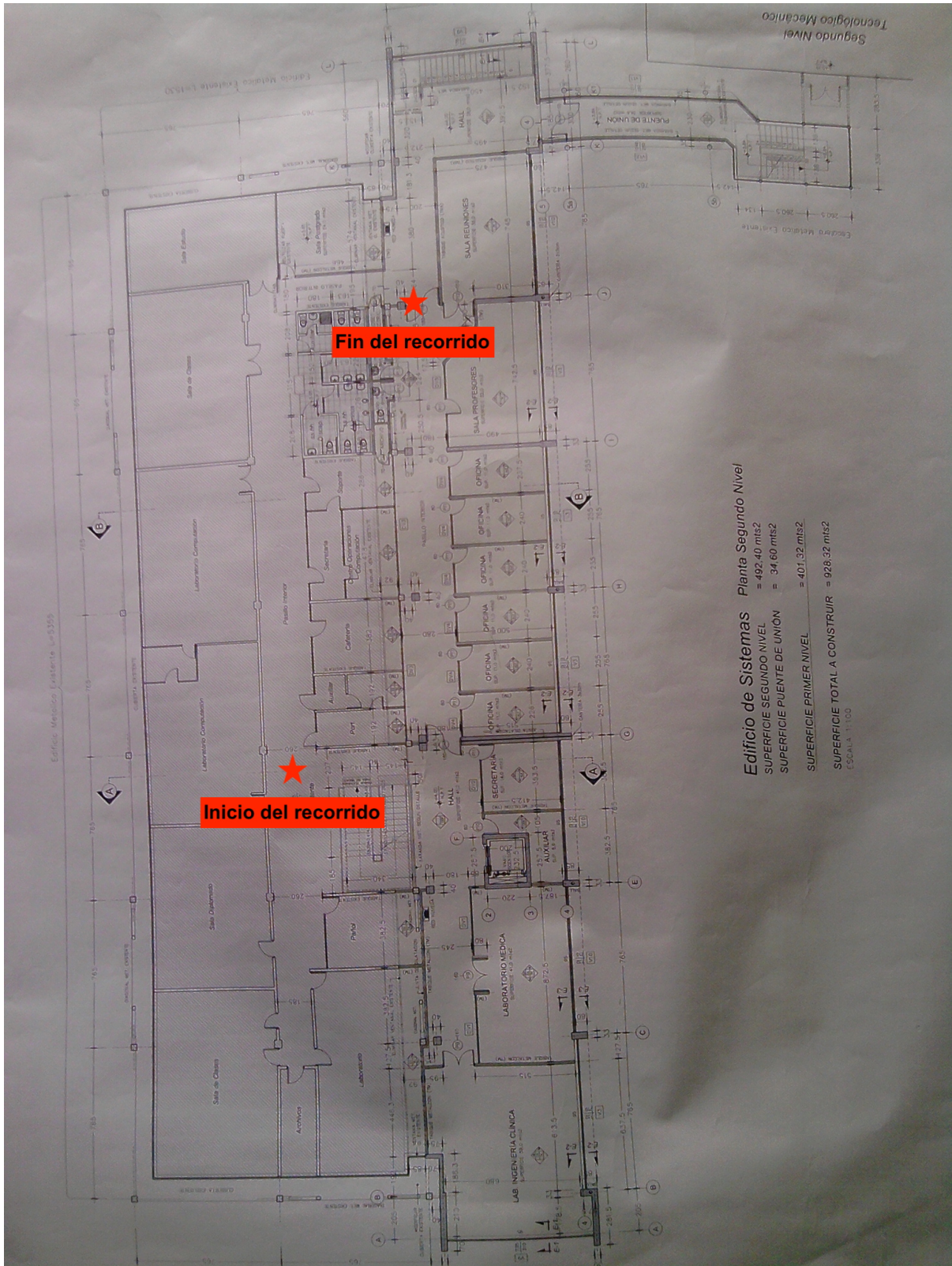


Fig. 36: Plano del segundo piso del edificio de Sistemas.

**UNIVERSIDAD DE CONCEPCION – FACULTAD DE INGENIERIA
RESUMEN DE MEMORIA DE TITULO**

Departamento : Departamento de Ingeniería Eléctrica
Carrera : Ingeniería Civil Biomédica
Nombre del memorista : **Valentina Teresa León Molina**
Título de la memoria : Sistema de navegación intrahospitalaria con dispositivo móvil como guía para pacientes.
Fecha de la presentación oral : 29 de Agosto, 2024.

Profesor(es) Guía : **Esteban Pino Quiroga**
Profesor(es) Revisor(es) : Sergio Sobarzo Guzmán.
Supervisor(es) Externo(s) : Macarena Araneda Herrera.
Concepto :
Calificación :

Resumen

En el presente informe se propone un sistema de posicionamiento intrahospitalario mediante una aplicación móvil para la Unidad de Imagenología del Hospital Clínico Regional de Concepción (HGGB) que busca guiar a pacientes hacia distintos lugares dentro del establecimiento.

En primer lugar, se realiza una búsqueda bibliográfica sobre sistemas de localización interior, así como su funcionamiento, sus ventajas, desventajas y el tipo de tecnología que utilizan. Posteriormente, se establece un marco teórico que contenga la información conceptual necesaria con el fin de idear una propuesta para el sistema de posicionamiento, incluyendo conceptos bases tales como códigos QR, seguridad y privacidad, sensores móviles y se da una introducción a Unity para el desarrollo de aplicaciones móviles.

En segundo lugar, se propone un sistema de localización integrado a una aplicación móvil que utiliza Realidad Aumentada (AR) para guiar a pacientes en tiempo real a través de la cámara del celular.

Por último, se hace una descripción de la aplicación móvil que se está desarrollando para luego ser evaluada mediante pruebas de funcionamiento dentro del hospital y en el edificio de Sistemas de la Universidad de Concepción comprobando el desempeño de la interfaz de usuario y la correcta ejecución del sistema de navegación