



Interacción Humano Computador

AR Finder: Sistema de Búsqueda de Objetos con Realidad Aumentada

Estudiantes:
Luciano Aguirre
Mitsuo Murakami

19 de mayo de 2025

Índice

1. Antecedentes	3
2. Requisitos de Diseños y Personas	3
2.1. Audiencia	3
2.2. Requisitos funcionales y no funcionales	3
3. Framework de Diseño	4
3.1. Restricciones y oportunidades del usuario	4
3.1.1. Funciones para el usuario	4
3.1.2. Limitaciones	4
4. Prototipo	4
5. Implementación	5
6. Conclusiones	5
7. Referencias	5

1. Antecedentes

En la vida cotidiana es común olvidarse dónde se dejó algún objeto. Por ejemplo, en un estacionamiento grande, usar la llave para escuchar la alarma del auto puede tener complicaciones si hay mucho ruido o el rango de la señal no es muy lejano. En la actualidad existen dispositivos de IoT, como AirTag, SmartTag, que sirven para rastrear objetos; sin embargo, se debe comprar uno para cada objeto que se quiera ubicar. Como propone Moya (2019) [1] en su investigación sobre un algoritmo de localización en interiores mediante Bluetooth Low Energy, estos dispositivos requieren una fuente de energía (batería o pilas) para su funcionamiento.

Una ayuda actual para buscar objetos mediante la cámara del celular es Google Lens [2], el cual analiza la imagen que capta la cámara y lo compara con data de internet por similitud. Esto es útil para encontrar objetos dentro del campo de visión.

Otras soluciones actuales son las aplicaciones de Apple y Google, Find Devices [3] y Find My Device [4]. Los cuales solo son para dispositivos inteligentes con conexión a internet o al alcance del Bluetooth como smartphones, laptops, smartwatches, audífonos, entre otros. Estas aplicaciones muestran en sus respectivas aplicaciones de mapa una aproximación en dónde se encuentra cada dispositivo vinculado.

2. Requisitos de Diseños y Personas

2.1. Audiencia

Esta app está diseñada para todo tipo de personas (con conocimientos básicos para instalar una app desde Play Store o App Store) que requieran localizar objetos. Esta app tiene como objetivo ayudar a las personas, que usualmente se olvidan dónde dejaron un objeto, a ubicarlo con facilidad. Por ejemplo, un estudiante que usa pocas veces su calculadora, no sabe en dónde lo dejó y lo necesita para un examen. Con esta app podría añadir un pin con el tag de calculadora y encontrarlo cuando necesite la calculadora.

2.2. Requisitos funcionales y no funcionales

Funcionales:

- Almacena las coordenadas de un pin
- Almacena y etiqueta cada pin
- Ubica gráficamente los pines almacenados en la pantalla usando la cámara (AR)
- Almacena una lista de pines
- Muestra la lista de pines

- Muestra la distancia en el que se encuentra el pin

No funcionales:

- Compatible en Android y iOS
- Ubicación rápida (la data se almacena localmente)
- Interfaz simple y amigable al usuario

3. Framework de Diseño

3.1. Restricciones y oportunidades del usuario

3.1.1. Funciones para el usuario

Los usuarios pueden crear pines, leer la lista de pines guardados, modificar los nombres de cada pin y eliminarlos. Además, en la vista principal, se puede visualizar mediante la cámara del dispositivo la ubicación de los pines en la realidad.

3.1.2. Limitaciones

Como mejora se podría guardar una ubicación más precisa de los pins, haciendo uso de la latitud y longitud a la hora de guardar los pins, evitando que cambien de posición cuando un usuario inicia la aplicación. Asimismo, se podría agregar un botón a la interfaz para guardar manualmente la lista de pins, evitando que se pierdan en el caso ocurra un error y no logren guardarse cuando se cierra la aplicación.

4. Prototipo

Para desarrollar esta aplicación, primero se realizó un prototipo en Figma, que nos permitió planear su diseño y funcionalidad. Este prototipo se creó considerando las cuatro funcionalidades principales de la aplicación: Mostrar todos los pins, editar y eliminar los pins, y visualizar una lista completa de todos los pins, para los cuales se crearon cuatro botones que se mostrarían en todo momento en la pantalla.

Además, el diseño de la aplicación debía ser minimalista, fácil de usar y tener elementos no obstructivos, debido a que se necesitaba interactuar directamente con el entorno para crear los pins.

5. Implementación

Se inició la implementación de la aplicación utilizando Unity 2022.3.50f1. Para implementar el componente principal de la aplicación (Agregar pins en el entorno virtual al interactuar) se hizo uso del paquete Lightship de Niantic, el cual nos permitía colocar objetos tridimensionales en el espacio mediante una profundidad simulada desde la cámara del dispositivo. Para cada Pin creado al interactuar, se autogeneraba un ID y un nombre que el usuario podía modificar para tener un mejor reconocimiento de este. Cada vez que se creaban pins nuevos, se almacenaban en memoria, para luego ser almacenados en un archivo a la hora de cerrar la aplicación.

En cuanto a la interfaz, se utilizó un Game Object del tipo Canvas, en donde se crearon distintos paneles, tanto para el panel principal que se muestra en todo momento, como los paneles pop-up que pedían una interacción por parte del usuario.

6. Conclusiones

En conclusión, se desarrolló una aplicación funcional que permite guardar ubicaciones específicas en el entorno físico mediante el uso de realidad aumentada, ofreciendo una forma más precisa, visual e intuitiva de marcar puntos de interés, superando las limitaciones de las herramientas tradicionales basadas únicamente en mapas 2D o GPS.

7. Referencias

- 1 Moya Velasco, W. P. (2019). Implementación de un algoritmo de localización en interiores para resolver el problema de ubicación de objetos usando tecnología Bluetooth Low Energy (Bachelor's thesis, Quito, 2019.).
- 2 Google Lens. (2025). ¿Qué es Google Lens? <https://lens.google/intl/es-419/howlensworks/>. Visitado el 19 de mayo de 2025.
- 3 Find Devices. (2025). Buscar Dispositivos. <https://www.icloud.com/find> Visitado el 19 de mayo de 2025.
- 4 Find My Device. (2025). Encontrar mi dispositivo. <https://www.google.com/android/find/about>. Visitado el 19 de mayo de 2025.