

Entre Diagonales - Descubriendo la historia y el encanto de la Ciudad de La Plata a través de desafíos interactivos con IA

Tesina de Licenciatura en Sistemas

Ignacio Cafiero Torrubia y José Ignacio Borrajo

Directoras Académicas: Claudia Banchoff y Laura Fava



Facultad de Informática
Universidad Nacional de La Plata

Agradecimientos

Queremos expresar nuestro más sincero agradecimiento a todas las personas que hicieron posible la realización de esta tesina.

En primer lugar, agradecemos a nuestras directoras por su acompañamiento, orientación y compromiso a lo largo de todo el proceso. Sus aportes y sugerencias fueron fundamentales para el desarrollo de este trabajo.

A nuestras familias, por el apoyo incondicional, la paciencia y el aliento constante durante esta etapa.

A nuestros compañeros y amigos, por el intercambio de ideas, la ayuda brindada y por acompañarnos en este recorrido.

Asimismo, queremos destacar el trabajo en conjunto, la dedicación y el compromiso compartido que nos permitió llevar adelante este proyecto.

Finalmente, agradecemos a todas aquellas personas que, de una u otra manera, contribuyeron a la concreción de esta tesina.

Índice general

Resumen	1
1. Capítulo 1. Introducción	2
1.1. Motivación	2
1.2. Objetivos y metodología	3
1.3. Organización del documento	3
2. Capítulo 2. Turismo Gamificado, la Temática en Entre Dia- gonales	4
2.1. Introducción	4
2.2. Turismo	4
2.3. Gamificación	5
2.3.1. Origen	5
2.3.2. Concepto	5
2.3.3. Propósito y Beneficios	5
2.3.4. Elementos Estructurales de la Gamificación	6
2.4. Turismo Gamificado	7
2.4.1. Combinación Estratégica en Turismo	7
2.4.2. Implementación a lo Largo del Viaje	8
2.4.3. Tecnologías de Soporte (Turismo 4.0)	8
2.4.4. Conclusiones	9
2.5. Aplicaciones turísticas	10
2.5.1. Tendencias Tecnológicas en el Sector Turístico	10
2.5.2. Mercado de Aplicaciones de Viajes en el 2025	13
2.5.3. Análisis del Mercado y Brechas	13
2.5.4. Implementación de la IA Visual en Entre Diagonales	18
2.5.5. Estrategia de Gamificación de Entre Diagonales	20
2.5.6. Propuesta Diferencial de Entre Diagonales	20
3. Capítulo 3. Diseño de Entre Diagonales	22
3.1. Entre Diagonales VS mercado actual	22
3.2. Descripción general	22
3.3. Recorridos	23
3.3.1. Diseño general de los recorridos	23
3.3.2. Progreso del recorrido	25
3.3.3. Mapa del recorrido	26
3.3.4. Escaneo de una parada u objeto secreto	27
3.3.5. Interacción con cada parada	28
3.3.6. Finalización del recorrido	29
3.3.7. Perfil del usuario	30

3.3.8.	Recorridos implementados	31
3.4.	Objetos Secretos	32
3.4.1.	Funcionamiento general	32
3.4.2.	Progreso dentro del recorrido	33
3.4.3.	Visualización de un objeto secreto encontrado	33
3.4.4.	Colección de objetos secretos	34
3.5.	Personaje Principal: Lúmino	35
3.6.	Personajes	36
3.7.	Gamificación	39
3.7.1.	Niveles	39
3.7.2.	Monedas de plata y gemas	41
3.7.3.	Retos y logros	42
3.7.4.	Trivias	45
3.7.5.	Tabla de clasificación	47
3.8.	Identidad visual: tipografía, colores y logotipo	49
3.8.1.	Tipografía	49
3.8.2.	Logotipo	50
3.8.3.	Paleta cromática	50
4.	Capítulo 4. Desarrollo de Entre Diagonales	52
4.1.	Resumen del desarrollo	52
4.2.	Arquitectura y visión general del sistema	52
4.3.	Tecnologías seleccionadas y justificación	53
4.3.1.	Tecnologías de inteligencia artificial	53
4.3.2.	Tecnologías backend	54
4.3.3.	Tecnologías frontend móvil	54
4.3.4.	Infraestructura y despliegue	55
4.4.	Flujo de interacción entre los componentes	55
4.5.	Alcance y limitaciones de la primera versión	56
4.6.	Construcción de un modelo de reconocimiento visual confiable	56
4.6.1.	Naturaleza del problema y criterio de selección de ar- quitecturas	56
4.6.2.	Selección del modelo base	57
4.6.3.	Construcción y etiquetado del dataset	58
4.6.4.	Metodología de evaluación	58
4.6.5.	Entrenamiento, validación y métricas	59
4.6.6.	Umbral de confianza y manejo de errores	60
4.6.7.	Integración del modelo en el flujo de la aplicación . . .	60
4.6.8.	Modelos descartados	62
4.6.9.	Decisión final del modelo	62
4.7.	Desarrollo del servidor	63
4.7.1.	Arquitectura y diseño del servidor	63
4.7.2.	API, autenticación y seguridad	63
4.7.3.	Integración con el reconocimiento visual	63

4.7.4. Despliegue y monitoreo	63
4.8. Implementación de la aplicación móvil	64
4.9. Selección del framework	64
4.10. Arquitectura interna	65
4.11. Flujo de navegación	65
4.12. Integración con el servidor	66
4.13. Potenciando la experiencia con efectos hápticos y sonoros . .	66
4.14. Cálculo de rutas y tiempos	66
4.14.1. Sistema de gamificación	67
4.14.2. Mapa interactivo	68
4.14.3. Escáner visual y validación	68
4.14.4. Notificaciones push	69
4.14.5. Accesibilidad y personalización de la interfaz	70
5. Capítulo 5. Desafíos Tecnológicos	72
5.1. Desafíos tecnológicos	72
5.1.1. Control de integridad por localización	72
5.1.2. Integración de servicios de rutas y mapas	72
5.1.3. Diseño de mecánicas sin repetición	73
5.1.4. Optimización de las imágenes	73
5.1.5. Mejora del rendimiento y fluidez de la aplicación . . .	74
5.1.6. Fluidez en los efectos visuales	75
5.1.7. Adaptabilidad a distintas pantallas	75
6. Capítulo 6. Evaluación de Entre Diagonales	76
6.1. Introducción	76
6.2. Pruebas	76
6.2.1. Pruebas técnicas de la aplicación	76
6.2.2. Pruebas funcionales	76
6.3. Encuestas	76
6.3.1. Metodología	76
6.3.2. Planificación	77
6.4. Resultados	79
6.4.1. Perfil de los participantes	79
6.4.2. Experiencia general de uso	80
6.4.3. Interfaz y usabilidad	80
6.4.4. Rutas y contenido cultural	81
6.4.5. Experiencia con el mapa	83
6.4.6. Reconocimiento visual	84
6.4.7. Gamificación	85
6.4.8. Evaluación final	85

7. Capítulo 7. Conclusiones y trabajos futuros	87
7.1. Conclusiones	87
7.2. Trabajos Futuros	88
7.2.1. Ampliación de la Cantidad de Recorridos	88
7.2.2. Creación de Equipos y Competencia Colaborativa	88
7.2.3. Optimización de la Navegación y Experiencia de Usuario (UX)	89
7.2.4. Expansión del Sistema de Monedas y Gemas	89
7.2.5. Incorporación de Sistema de Audio-Guía Interactiva	90

Índice de figuras

2.1. Interfaz principal de Layla	11
2.2. Búsqueda sobre aplicaciones de turismo en la Ciudad De La Plata.	14
2.3. Ejemplos disponibles en Google Play al momento del relevamiento (2026). Estas aplicaciones funcionan principalmente como guías digitales informativas.	15
2.4. Búsqueda sobre la Ciudad de La Plata en la aplicación Get Your Guide.	16
2.5. Diferencia de itinerarios entre TripAdvisor y <i>Entre Diagonales</i> . Primer pantalla Entre Diagonales y segunda TripAdvisor	17
3.1. Pantalla principal de la aplicación	23
3.2. Pantalla de recorridos	24
3.3. Pantalla de progreso de recorrido	25
3.4. Pantalla del mapa del recorrido	26
3.5. Pantalla de captura fotográfica de una parada.	27
3.6. Pantalla una parada del recorrido	29
3.7. Pantalla de fin del recorrido	30
3.8. Pantalla de perfil del usuario.	31
3.9. Pantalla correspondiente al objeto secreto <i>Euterpe</i> , presente en el recorrido <i>Plaza Moreno</i>	34
3.10. Pantalla de Objetos Secretos	35
3.11. Lúmino	36
3.12. Pantalla de selección de personaje	38
3.13. Ilustraciones de los niveles disponibles en <i>Entre Diagonales</i>	40
3.14. Ilustración de la moneda de plata.	41
3.15. Ilustración de la gema.	42
3.16. Pantalla de visualización de logros en <i>Entre Diagonales</i>	45
3.17. Pantalla de resolución de trivía en <i>Entre Diagonales</i>	46
3.18. Pantalla de la tabla de clasificación en <i>Entre Diagonales</i>	48
3.19. Logotipo de <i>Entre Diagonales</i>	50
4.1. Arquitectura general del sistema: interacción entre capas.	53
4.2. Comparación de <i>accuracy</i> y <i>loss</i> durante el entrenamiento de MobileNetV2 (izquierda) y ConvNeXt Tiny (derecha).	59
4.3. Comparación de las matrices de confusión de MobileNetV2 (izquierda) y ConvNeXt Tiny (derecha).	60
4.4. Flujo principal del sistema. Interacción entre componentes	62
4.5. Visual de notificación recibida al teléfono móvil.	70

4.6. Opciones de accesibilidad y personalización visual de la interfaz, incluyendo el ajuste del tamaño de texto.	71
6.1. Conocimiento previo de las personas en la Ciudad de La Plata.	79
6.2. Uso habitual de aplicaciones turísticas por parte de los usuarios.	80
6.3. Resultados sobre la lectura de textos e indicaciones.	81
6.4. Resultados sobre el interés sobre el contenido.	82
6.5. Resultados sobre el seguimiento de las indicaciones	82
6.6. Resultados sobre la precisión de la ubicación.	83
6.7. Resultados sobre la carga del mapa.	84
6.8. Resultados del funcionamiento del reconocimiento de objetos.	84
6.9. Resultados sobre la motivación de los desafíos.	85
6.10. Resultados sobre la calificación de la aplicación.	86

Resumen

Entre Diagonales es un prototipo de aplicación turística gamificada, diseñada para que el público en general se interese más por la historia y los secretos que tiene la ciudad De La Plata. La aplicación promueve el turismo cultural de la ciudad a través del uso de reconocimiento de objetos, rutas temáticas y desafíos. Utilizando la cámara del dispositivo móvil, la aplicación permite al usuario enfocar edificios, monumentos emblemáticos y objetos secretos para completar un punto o desbloquear recompensas, además de realizar recorridos guiados paso a paso.

Se integra una arquitectura, donde a través de una API-REST que utiliza la aplicación en conjunto con un modelo de inteligencia artificial que se encarga del reconocimiento visual de los objetos y monumentos.

Capítulo 1. Introducción

1.1. Motivación

En los últimos años, diversos estudios han demostrado que los videojuegos y las dinámicas lúdicas se han convertido en una forma de entretenimiento ampliamente adoptada tanto por niños como por personas adultas. Ofrecen un medio atractivo y efectivo para fomentar procesos de aprendizaje significativo, entendido como un proceso en el que la persona conecta la información nueva con los conocimientos que ya posee en su estructura cognitiva de manera sustancial y no arbitraria. A diferencia de la memorización mecánica, este enfoque permite otorgar un sentido personal a lo aprendido, facilitando su retención a largo plazo y su aplicación en diversos contextos[1]. Su potencial no solo radica en el componente lúdico, sino también en la capacidad de transformar contenidos complejos en experiencias accesibles, motivadoras y memorables [2].

Este enfoque ha impulsado múltiples iniciativas orientadas a integrar mecánicas de juego en ámbitos educativos, culturales y sociales, evidenciando que la interacción, el desafío y la recompensa pueden convertirse en motores de participación activa y construcción de conocimiento. En este marco, la gamificación y los juegos serios se presentan como herramientas particularmente valiosas para promover la curiosidad, el descubrimiento y la apropiación del entorno.

Trasladar estas estrategias al turismo urbano, especialmente en ciudades con un fuerte valor histórico y cultural como La Plata, abre la posibilidad de generar experiencias que no solo entretengan, sino que también eduquen y fortalezcan el vínculo emocional con el patrimonio local. Así como existen proyectos que utilizan videojuegos para introducir a jóvenes en prácticas ecologistas o científicas [3], también es posible aplicarlos a la exploración cultural, invitando a residentes y visitantes a descubrir la ciudad de manera mas activa, dinámica y significativa.

Desde esta perspectiva nace *Entre Diagonales*, una propuesta que busca aprovechar la capacidad motivadora del juego para despertar el interés por la historia, la arquitectura y los relatos que conforman la identidad platense. Mediante desafíos interactivos, rutas temáticas y reconocimiento visual de monumentos, la aplicación aspira a transformar la experiencia turística tradicional en una aventura urbana que inspire a las personas a explorar, aprender y valorar su ciudad con una mirada renovada.

1.2. Objetivos y metodología

El objetivo de esta tesina es desarrollar un prototipo funcional de una aplicación móvil de turismo gamificado que utilice reconocimiento visual para identificar monumentos y edificios emblemáticos de la ciudad de La Plata, ofreciendo al usuario una experiencia interactiva basada en desafíos, recompensas e itinerarios culturales. La propuesta busca fomentar el aprendizaje histórico y arquitectónico mediante dinámicas de gamificación, incentivando a residentes y turistas a explorar el patrimonio local de manera activa, entretenida y significativa.

1.3. Organización del documento

Este documento describe el proceso creativo y técnico detrás de *Entre Diagonales*, narrando cada etapa, desde la concepción inicial hasta las iteraciones de desarrollo y las pruebas del prototipo.

- **Capítulo 1:** presentamos la motivación del proyecto y describimos el objetivo de la tesis.
- **Capítulo 2:** presentamos conceptos de turismo y como se puede integrar la gamificación.
- **Capítulo 3:** describimos algunas definiciones de videojuegos educativos y aplicaciones turísticas. Presentamos antecedentes de aplicaciones sobre turismo y realizamos comparaciones con algunas de ellas.
- **Capítulo 4:** describimos y analizamos el diseño de las mecánicas e interfaz de usuario en *Entre Diagonales*, explicando como integramos la información turística en la aplicación.
- **Capítulo 5:** compartimos el proceso de desarrollo de *Entre Diagonales*, desde la creación del modelo de reconocimiento visual, siguiendo con la implementación del servidor y culminando en el desarrollo de la aplicación móvil. Asimismo, se describen y justifican las tecnologías seleccionadas para llevar adelante el proyecto.
- **Capítulo 6:** enumeramos varios desafíos técnicos y de diseño que enfrentamos en la construcción del prototipo y cómo los resolvimos.
- **Capítulo 7:** describimos cómo diseñamos y llevamos a cabo las pruebas con jugadores y cuáles fueron los resultados obtenidos.
- **Capítulo 8:** compartimos conclusiones y posibles trabajos futuros.

Capítulo 2. Turismo Gamificado, la Temática en Entre Diagonales

2.1. Introducción

Si bien la gamificación, entendida como el uso de mecánicas de juego en contextos ajenos a este, posee raíces históricas previas a la digitalización, la reciente y constante evolución de las Tecnologías de la Información y las Comunicaciones (TIC) ha actuado como un catalizador sin precedentes. Según Zichermann y Cunningham [4], aunque el concepto no depende intrínsecamente de lo digital, las herramientas tecnológicas y los dispositivos móviles han potenciado su alcance, interactividad y escalabilidad. En este sentido, el sector turístico ha comenzado a integrar la gamificación mediada por tecnología como una herramienta estratégica clave para la gestión y promoción de destinos, permitiendo una conexión más profunda y ubicua con el usuario.

Esta situación ha dado lugar a significativas novedades en la gestión y el marketing turístico. Los destinos turísticos, a través de sus organismos de gestión y promoción, están comenzando a incorporar esta tendencia en sus actividades, ya que la gamificación se presenta como una poderosa estrategia para potenciar la motivación, el esfuerzo y la fidelización de los turistas. Esto responde a la práctica empresarial de innovar y adaptarse a la experiencia del consumidor en línea, quien ahora también participa en dinámicas de juego [5].

2.2. Turismo

El turismo, como fenómeno global, constituye un pilar económico y social de relevancia crítica para el desarrollo de las comunicaciones.

La Organización de las Naciones Unidas establece que el turismo “*comprende las actividades que realizan las personas durante sus viajes y estancias en lugares distintos a su entorno habitual, por un periodo de tiempo consecutivo inferior a un año, con fines de ocio, por negocios y por otros motivos turísticos, siempre y cuando no sea desarrollar una actividad remunerada en el lugar visitado*” [6].

Sabemos que el crecimiento progresivo del turismo tiene impactos importantes en el ámbito cultural, social y económico de un país [6]. La medición económica del fenómeno turístico se aborda desde la perspectiva de la demanda y la oferta:

- **Oferta:** se define en función de las ramas de actividad que satisfacen el consumo turístico, incluyendo hoteles y establecimientos similares,

inmuebles en alquiler, restaurantes, cafés, transporte y agencias de viaje u operadores turísticos.

- **Impacto económico:** el gasto turístico motoriza cadenas de pago y repercute en el empleo y el bienestar de la comunidad. La medición a través de encuestas como la Encuesta de Ocupación Hotelera (EOH) es crucial para el cálculo del Producto Bruto Geográfico (PBG), dado que el turismo desplaza geográficamente el nivel de actividad, generando impactos significativos a nivel local.

2.3. Gamificación

2.3.1. Origen

Si bien el término fue acuñado inicialmente por Nick Pelling en 2002, la gamificación (o ludificación) comenzó a cobrar relevancia institucional y académica a partir de 2008, cuando se popularizó dentro de la industria de los medios digitales y el desarrollo de software [7]. Este periodo marcó la transición de la gamificación como una práctica dispersa hacia una estrategia formalmente reconocida y potenciada por las nuevas tecnologías.

2.3.2. Concepto

Podemos plantear las siguientes definiciones:

- El uso de elementos del diseño de juegos en contextos no relacionados con el juego [8].
- La aplicación del pensamiento y de la mecánica de los juegos para involucrar a los usuarios en la resolución de problemas o en practicar actividades colectivamente, estableciendo un sentido de compromiso [9].
- La aplicación de elementos de juego y técnicas de diseño de juegos digitales a problemas ajenos al juego, como los retos empresariales y de impacto social.

2.3.3. Propósito y Beneficios

El propósito fundamental de la gamificación es convertir tareas o procesos que podrían ser percibidos como mundanos o difíciles en desafíos lúdicos y motivadores. La idea principal es envolver a la audiencia. La gamificación busca:

- Potenciar la motivación y el compromiso de los usuarios.

- Aumentar la retención de usuarios y el *engagement* [10].
- Fomentar la creatividad, la actitud emprendedora y el trabajo en equipo en entornos [11].
- Generar un sentido de compromiso o una relación leal entre las personas y las empresas o marcas.

2.3.4. Elementos Estructurales de la Gamificación

Según Werbach [12] la complejidad de la gamificación va más allá de los elementos superficiales. Para diseñar experiencias significativas, no alcanza con incorporar puntos o recompensas de manera aislada, sino que es necesario comprender cómo se relacionan los distintos niveles que estructuran la experiencia lúdica.

En este sentido, Werbach y Hunter [12] retoman el modelo MDA (Mechanics, Dynamics, Aesthetics), un marco conceptual originalmente utilizado en el diseño de videojuegos [13], que permite analizar y construir sistemas gamificados a partir de tres niveles interrelacionados. En el contexto de la gamificación, este modelo suele adaptarse para centrarse en Mecánicas, Dinámicas y Componentes, facilitando la traducción de conceptos del diseño de juegos hacia entornos no lúdicos. Los fundamentos del diseño del juego basados en este enfoque se muestran en el Cuadro 2.1.

Elemento	Descripción	Ejemplos
Dinámicas	El concepto o la estructura implícita del juego, es decir, aquello que da sentido y continuidad a la experiencia. Se asocia a conceptos como curiosidad, estatus, competencia o altruismo.	Narrativas, progresión, emociones, interacción social.
Mecánicas	Los procesos que impulsan el desarrollo del juego; son las reglas, acciones y sistemas que orientan el comportamiento del jugador.	La tríada PBL (Puntos, Insignias/Medallas y Tablas de Clasificación).
Componentes	Las implementaciones concretas y visibles de las dinámicas y mecánicas dentro del sistema gamificado.	Avatares, niveles, equipos, desafíos, barras de progreso, recompensas.

Cuadro 2.1: Elementos del modelo MDA aplicados a la gamificación según Werbach y Hunter [12]

Para lograr un compromiso sostenido en el tiempo, las estrategias de gamificación deben enfocarse especialmente en la **motivación intrínseca**, es

decir, en aquellos factores internos que impulsan la participación del usuario más allá de las recompensas externas. Entre estos factores se destacan la **autonomía**, la **competencia** (o dominio) y el **propósito** (o finalidad), considerados pilares fundamentales para promover una experiencia significativa y duradera [14].

2.4. Turismo Gamificado

La gamificación se ha convertido en una estrategia de marketing efectiva para llamar la atención de los viajeros. Al aplicar los elementos de diseño de juegos, se transforma al viajero de un espectador pasivo a un jugador activo y protagonista de su experiencia [15].

2.4.1. Combinación Estratégica en Turismo

La aplicación de la gamificación en el turismo se divide en dos grandes grupos de posibilidades: aumentar la motivación de turistas y empleados para cambiar comportamientos, y permitir a turistas y empleados una co-creación de valor que genere motivación intrínseca.

Fidelización

Las tarjetas de fidelidad o programas de vuelo frecuente (millas) son ejemplos de enfoques similares a la gamificación utilizados en el sector. Los programas de lealtad, que consisten en un conjunto de reglas con incentivos, son inherentemente gamificables y buscan mejorar el comportamiento de consumo [9].

Experiencia Enriquecida

La gamificación permite traducir la complejidad histórica o cultural de un destino en formatos accesibles, atractivos y memorables, aumentando el interés y la comprensión del público. El aprendizaje se asocia a una experiencia divertida y participativa, lo que lo hace más ameno y duradero [16][17].

Por ejemplo, en lugar de presentar información histórica de forma exclusivamente expositiva, una aplicación turística gamificada puede proponer al usuario recorrer un circuito temático, identificar un monumento mediante la cámara del dispositivo y desbloquear una recompensa digital al responder correctamente una pregunta sobre su contexto histórico. De este modo, el contenido cultural deja de percibirse como información estática y pasa a integrarse en una experiencia interactiva, favoreciendo tanto la participación como la retención del conocimiento.

Gestión del Destino

La gamificación puede ser utilizada para mejorar los procesos internos de las empresas turísticas, buscando aumentar la productividad y la satisfacción de los trabajadores. También es una herramienta para fomentar el turismo sostenible, promoviendo la descentralización de turistas desde zonas saturadas hacia puntos geográficos menos congestionados, a través de rutas gamificadas que ofrecen bonificaciones por exploración [18].

2.4.2. Implementación a lo Largo del Viaje

La gamificación tiene la capacidad de intervenir en todas las etapas del ciclo de viaje:

1. **Fase Pre-viaje:** puede aparecer como elemento incitador al deseo de visitar un destino (por ejemplo, al realizar una experiencia de Realidad Virtual) o para ayudar en la compra o reserva del viaje mediante programas de puntos. Asimismo, plataformas lúdicas basadas en la exploración geográfica, como *GeoGuessr* [19], pueden despertar el interés previo por determinados lugares al invitar al usuario a reconocer ciudades, paisajes o puntos de referencia a partir de imágenes inmersivas, generando curiosidad y familiaridad con el destino incluso antes de concretar la visita [20].
2. **Fase Durante la Estancia:** es el punto de máxima interacción, transformando la visita en una aventura lúdica. Se utilizan experiencias inmersivas, juegos con realidad aumentada y geolocalización, rutas gamificadas para entornos urbanos y rurales, y juegos temáticos en museos o parques. Un ejemplo de este tipo de estrategias puede observarse en aplicaciones inspiradas en dinámicas de exploración urbana, similares a *Pokémon GO* [21], donde el usuario debe desplazarse físicamente por el espacio para descubrir puntos de interés y desbloquear interacciones. En el contexto turístico, estas mecánicas pueden aplicarse a circuitos patrimoniales en los que cada ubicación activa desafíos, pistas o recompensas vinculadas con la historia y la identidad cultural del lugar.
3. **Fase Post-viaje:** se mantiene la fidelización al compartir la experiencia y opinar en redes sociales que valoran esa participación.

2.4.3. Tecnologías de Soporte (Turismo 4.0)

El desarrollo de experiencias turísticas gamificadas y de turismo inteligente requiere el uso de diversas tecnologías digitales que permiten enriquecer la interacción entre el usuario y el entorno. En el contexto del denominado Turismo 4.0, estas tecnologías facilitan la creación de experiencias más

inmersivas, personalizadas e interactivas para los visitantes [22]. Entre las principales tecnologías de soporte se destacan:

- **Realidad Aumentada (RA):** permite superponer contenido digital (narrativas, información o elementos interactivos) sobre escenarios reales, enriqueciendo la experiencia turística y favoreciendo la gamificación de recorridos. Por ejemplo, distintas experiencias de RA aplicadas a entornos urbanos, como las desarrolladas en ciudades turísticas de alta densidad patrimonial como París, demuestran cómo el reconocimiento espacial y la superposición de capas digitales pueden utilizarse para destacar puntos de interés, activar contenidos contextuales y transformar el recorrido en una experiencia más inmersiva e interactiva. En este sentido, la RA puede emplearse para que el visitante reciba información histórica, pistas o desafíos al enfocar monumentos, edificios emblemáticos o espacios públicos con la cámara de su dispositivo [23].
- **Realidad Virtual (RV):** ofrece inmersión para la previsualización de destinos o la recreación virtual de patrimonio cultural, permitiendo al usuario explorar espacios y contextos históricos que podrían no estar físicamente accesibles. Un ejemplo significativo es la experiencia inmersiva vinculada a la Catedral de Notre-Dame de París, en la que los visitantes pueden recorrer virtualmente distintas etapas de la historia del monumento, desde su construcción en la Edad Media hasta procesos contemporáneos de restauración. Este tipo de propuestas permite no solo anticipar una visita, sino también preservar y difundir el patrimonio cultural mediante reconstrucciones digitales de alto valor educativo y experiencial [24].
- **Geolocalización:** es fundamental para los juegos apoyados en tecnología, permitiendo realizar un seguimiento de los flujos de turistas y circunscribir los juegos a un punto, zona o la totalidad de un destino, facilitando la deslocalización.
- **Inteligencia Artificial (IA):** la IA generativa se utiliza para la personalización extrema de itinerarios y desafíos, lo cual es crucial para mantener la motivación intrínseca del viajero y asegurar una gestión turística más inteligente. La IA aprovecha la gran cantidad de datos que genera la gamificación [25].

2.4.4. Conclusiones

La gamificación, al integrarse con tecnologías de vanguardia, permite a los destinos turísticos ofrecer un producto diferenciador y novedoso en un mercado altamente competitivo.

Metafóricamente, la combinación del turismo y la gamificación es como transformar un mapa estático y unidimensional en un tesoro interactivo. El mapa (el destino turístico) ya contiene la riqueza (la cultura, la historia y la naturaleza), pero al aplicar la gamificación (el diseño de juego), se le añade una brújula digital, pistas ocultas y recompensas, representadas por la tríada PBL (puntos, insignias y tablas de clasificación) y las narrativas, que motivan al explorador (el turista) a desviarse del camino trillado, a invertir tiempo en descifrar los secretos de la tierra, y a sentirse recompensado no solo por llegar a un punto, sino por el placer activo del descubrimiento mismo.

2.5. Aplicaciones turísticas

La tecnología ha transformado irrevocablemente el sector turístico y la experiencia del viajero, consolidando el teléfono móvil como herramienta fundamental para la navegación, el descubrimiento y la planificación de viajes.

2.5.1. Tendencias Tecnológicas en el Sector Turístico

El sector turístico está siendo revolucionado por diversas innovaciones tecnológicas:

- **Inteligencia Artificial (IA) y Chatbots:** proporcionan asistencia instantánea, recomendaciones personalizadas, ayuda en la planificación o reserva y actualizaciones en tiempo real. Por ejemplo, plataformas turísticas y agencias de viaje online han comenzado a incorporar asistentes conversacionales basados en IA, como *SOFIA* de Despegar, orientados a ayudar al usuario a explorar destinos, resolver dudas y personalizar su experiencia de viaje. Asimismo, soluciones especializadas como *Layla* Figura 2.1 o planificadores inteligentes de viajes permiten generar itinerarios adaptados a preferencias, presupuesto e intereses del turista, demostrando el potencial de la IA para acompañar tanto la etapa de planificación como la experiencia general del viaje [26][27].



Figura 2.1: Interfaz principal de Layla

- **Internet de las cosas (IoT):** permite la creación de ciudades inteligentes y hoteles inteligentes, brindando una experiencia conectada y personalizada. En el ámbito hotelero, un ejemplo representativo es el uso de asistentes de voz integrados en habitaciones, como *Alexa Smart Properties for Hospitality*, que permite al huésped controlar iluminación, climatización, entretenimiento y solicitar servicios mediante comandos de voz. En el entorno urbano, el IoT también se aplica en sistemas de movilidad y señalización inteligente, así como en mapas interactivos con información en tiempo real sobre transporte, ocupación o eventos, mejorando la experiencia del visitante [28].
- **Plataformas de reserva online:** han simplificado la planificación de viajes al permitir a los usuarios comparar precios y reservar distintos servicios desde un mismo entorno digital. Un ejemplo ampliamente difundido es *Booking.com*, que centraliza reservas de alojamiento y otros servicios turísticos, mientras que plataformas como *Skyscanner* facilitan la comparación de vuelos, hoteles y alquileres de vehículos, optimizando el proceso de búsqueda y toma de decisiones del viajero [29][30].

- **Pagos móviles y tecnología NFC:** han hecho las transacciones más cómodas y seguras para los turistas, reduciendo la necesidad de efectivo y agilizando el acceso a servicios. Un ejemplo de ello es el uso de pagos sin contacto en sistemas de transporte urbano, como ocurre en distintas redes europeas donde los usuarios pueden acceder mediante tarjetas o dispositivos móviles con tecnología NFC. Este tipo de soluciones mejora la movilidad del turista al simplificar el pago y reducir barreras operativas durante el desplazamiento.
- **Big Data y análisis de datos:** permiten a las empresas comprender mejor el comportamiento y las preferencias de los clientes para ofrecer experiencias más personalizadas y anticipar necesidades. En el sector turístico, el análisis de grandes volúmenes de datos provenientes de búsquedas, reservas, valoraciones, movilidad y hábitos de consumo permite detectar patrones de comportamiento, segmentar usuarios y optimizar recomendaciones. De esta manera, las empresas pueden ajustar ofertas, priorizar destinos o servicios de mayor interés y mejorar la experiencia general del viajero [31][32].
- **Inteligencia de localización y geolocalización:** las aplicaciones basadas en la inteligencia de localización y geolocalización han mejorado la experiencia del turista al proporcionar recomendaciones y ofertas personalizadas según su ubicación. Un ejemplo cotidiano es *Google Maps*, cuya función *Explore* permite descubrir restaurantes, atracciones, museos y otros puntos de interés cercanos en función de la localización del usuario, combinando relevancia, distancia y popularidad para ofrecer sugerencias contextualizadas. Este tipo de herramientas facilita la exploración espontánea del destino y favorece una experiencia más dinámica y adaptada al contexto [33][34].

En conjunto, estas tecnologías están redefiniendo la forma en que los turistas planifican, experimentan y recuerdan sus viajes. La integración de herramientas digitales, sistemas inteligentes y dispositivos móviles permite generar experiencias más personalizadas, interactivas y dinámicas. En este contexto, las aplicaciones móviles adquieren un rol central al actuar como intermediarias entre el turista y el entorno urbano, facilitando el acceso a información contextual, la exploración de destinos y la participación en experiencias innovadoras. Estas transformaciones tecnológicas sientan las bases para el desarrollo de propuestas turísticas interactivas, como las que se exploran en esta investigación, donde la combinación de geolocalización, reconocimiento visual y elementos de gamificación permite enriquecer la forma en que los visitantes descubren y recorren la ciudad.

2.5.2. Mercado de Aplicaciones de Viajes en el 2025

Según un estudio del Nimble AppGenie el mercado de plataformas de aplicaciones de viajes se espera que crezca a una TCAC¹ del **9.2 %**, pasando de USD 134.4 mil millones en 2023 a **USD 242.4 mil millones en 2030** [35].

El crecimiento está impulsado por la penetración de los teléfonos inteligentes, el auge del turismo y los avances tecnológicos como la IA y la AR/VR.

Las aplicaciones de viajes alcanzaron nuevos récords en 2024 en descargas (4.2 mil millones) y tiempo de uso (más de 20 mil millones de horas), lo que indica una adopción creciente de herramientas móviles para la planificación y reserva de viajes [36].

Las aplicaciones turísticas pueden clasificarse según diversas funciones; si bien existe una amplia variedad de categorías, las más frecuentes incluyen: **Reserva de vuelo** (Skyscanner, Expedia), **Reserva de hotel** (Booking.com, Airbnb), **Planificación de viajes** (TripIt) y **Transporte** (Uber, Lyft).

La economía colaborativa ha permitido la existencia de plataformas como **TouristLike**, una aplicación móvil orientada al turismo que conecta a guías locales con turistas para ofrecer actividades, operando mediante el intercambio de servicios a cambio de una retribución monetaria [37].

2.5.3. Análisis del Mercado y Brechas

Previo al desarrollo de *Entre Diagonales*, se realizó un relevamiento de aplicaciones móviles vinculadas al turismo cultural, tanto a nivel nacional como internacional, con especial atención a propuestas centradas en ciudades intermedias y en el uso de tecnologías emergentes como Inteligencia Artificial (IA), gamificación y Realidad Aumentada (RA).

El análisis permitió identificar una serie de tendencias del mercado y, al mismo tiempo, brechas específicas que justifican la propuesta desarrollada para la ciudad de **La Plata**.

1. Enfoque Geográfico y Especialización Local

Al buscar aplicaciones en Google Play bajo términos como “La Plata” o “La Plata Turismo”, no se identificó una aplicación oficial o tecnológicamente actual dedicada exclusivamente al patrimonio cultural de la ciudad como se muestra la Figura 2.2.

¹Tasa de crecimiento anual compuesto o CAGR por sus siglas en inglés

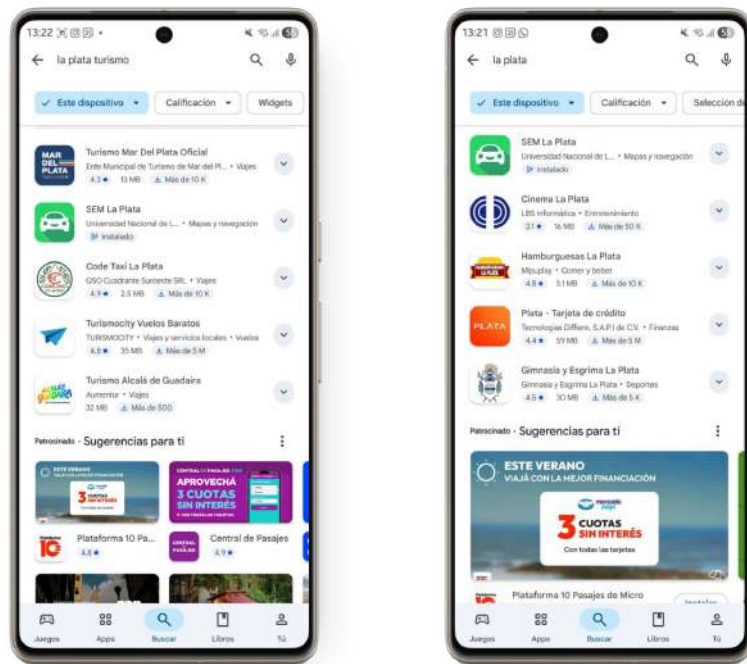


Figura 2.2: Búsqueda sobre aplicaciones de turismo en la Ciudad De La Plata.

Si bien existen aplicaciones turísticas de ciudades argentinas como *Turismo Mina Clavero*, *Jujuy Turismo* o *Buenos Aires Turismo* 2.3, estas suelen operar como guías informativas digitales, replicando el formato de un folleto tradicional en versión móvil, sin incorporar dinámicas interactivas avanzadas. La Figura 2.3 muestra unas capturas sobre ciertas aplicaciones con este formato.

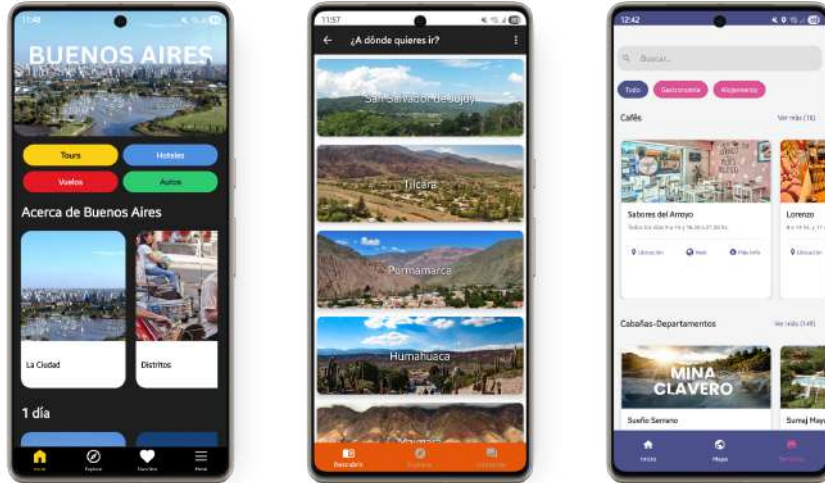


Figura 2.3: Ejemplos disponibles en Google Play al momento del relevamiento (2026). Estas aplicaciones funcionan principalmente como guías digitales informativas.

Por otra parte, plataformas globales como *TripAdvisor* o *GetYourGuide* no están diseñadas con un enfoque identitario local, sino como agregadores internacionales de servicios y experiencias como vemos en la Figura 2.4. Esto implica que la experiencia no se percibe como propia de la ciudad, sino como parte de un ecosistema global de reservas o planificación de viajes.

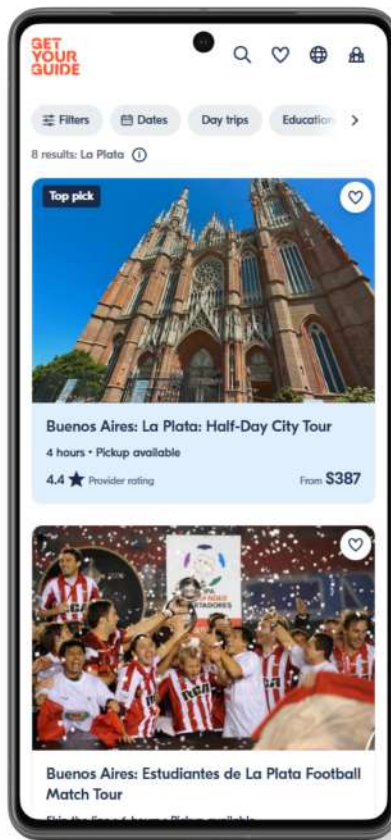


Figura 2.4: Búsqueda sobre la Ciudad de La Plata en la aplicación Get Your Guide.

2. Uso de Inteligencia Artificial

La IA se encuentra presente en diversas aplicaciones turísticas actuales, principalmente en sistemas de recomendación, generación de itinerarios personalizados o asistentes conversacionales basados en modelos de lenguaje.

Existen aplicaciones como *TourAI* o planificadores con IA que generan rutas automáticas en base a fechas y preferencias del usuario. Asimismo, algunas ciudades del mundo han incorporado experiencias de Realidad Aumentada para superponer información sobre puntos específicos.

Sin embargo, en el relevamiento realizado no se identificó una aplicación que utilice **IA visual aplicada al reconocimiento de monumentos como mecanismo central de interacción lúdica y desbloqueo de contenido**. La mayoría de las soluciones basadas en IA se enfocan en texto o planificación, no en reconocimiento visual

como disparador de progreso dentro de una experiencia gamificada.

3. Tipo de Interacción y Experiencia de Usuario

Muchas aplicaciones turísticas ofrecen una experiencia predominantemente pasiva: lectura de información, consulta de mapas o reserva de actividades. Incluso cuando permiten generar itinerarios, el usuario suele ser quien debe organizarlos manualmente.

En el caso de *TripAdvisor*, por ejemplo, el itinerario es una herramienta de planificación flexible, pero no está estructurado como una narrativa temática guiada ni como un sistema progresivo de desafíos.

En contraste, *Entre Diagonales* propone rutas temáticas predefinidas que integran historia, cultura y exploración urbana bajo una lógica narrativa. La Figura 2.5 muestra capturas de ambas aplicaciones en donde se pueden notar los enfoques diferentes de cada una sobre la misma ciudad.

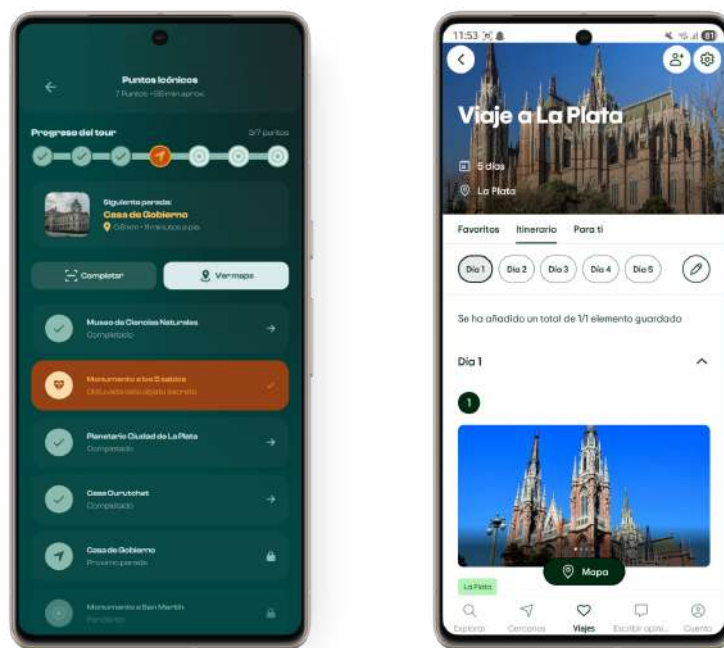


Figura 2.5: Diferencia de itinerarios entre TripAdvisor y *Entre Diagonales*. Primer pantalla Entre Diagonales y segunda TripAdvisor

4. Gamificación y Profundidad Cultural

Existen aplicaciones que incorporan gamificación, como *TripRanger* o plataformas de búsquedas del tesoro digitales (por ejemplo, *ScavengerHunt.com*). Estas suelen incluir sistemas de puntos, niveles e

insignias canjeables.

No obstante, estas dinámicas operan generalmente de forma independiente del contexto cultural profundo o del reconocimiento visual del entorno. En muchos casos, los desafíos consisten en visitar puntos marcados previamente por otros usuarios o completar tareas genéricas, sin una integración educativa o narrativa específica vinculada al patrimonio local.

La brecha identificada no radica en la ausencia de gamificación en el mercado, sino en la **falta de integración entre identidad local, narrativa histórica, reconocimiento visual y progresión lúdica coherente**.

En este sentido, *Entre Diagonales* se centra específicamente en la ciudad de La Plata, articulando:

- Rutas temáticas basadas en su trazado urbano singular y patrimonio arquitectónico.
- Reconocimiento visual de monumentos como mecanismo de validación.
- Desbloqueo de puntos u objetos virtuales mediante la captura de imágenes.
- Una progresión gamificada con sentido educativo y contextual.

En síntesis, si bien el mercado ofrece múltiples soluciones turísticas con distintos grados de digitalización, planificación asistida y gamificación, no se identificó una propuesta que combine, de forma integrada y centrada exclusivamente en La Plata, reconocimiento visual mediante IA, narrativa patrimonial y mecánicas lúdicas progresivas orientadas a la exploración urbana.

2.5.4. Implementación de la IA Visual en Entre Diagonales

El sistema de visión por computadora de *Entre Diagonales* se centra en la **identificación de monumentos e ítems secretos** al apuntar la cámara del dispositivo móvil.

La aplicación utiliza una **API en la nube desarrollada en Python** para procesar la imagen con un modelo de IA y retornar el resultado del reconocimiento. Este flujo evita la necesidad de ejecutar el modelo en el dispositivo móvil, relajando las restricciones de hardware.

El proyecto se enfoca en arquitecturas de **clasificación de imágenes** (ya que el escaneo se centra en un único objeto de interés) y se evaluaron modelos optimizados para rendimiento y eficiencia como **MobileNet, ResNet, EfficientNet y ConvNeXt**.

¿Por qué IA y no un simple código QR?

El planteamiento de utilizar Inteligencia Artificial (IA) visual en lugar de códigos QR responde a una necesidad doble dentro del proyecto *Entre Diagonales*: **garantizar la autenticidad de la interacción y potenciar la experiencia de usuario** al transformar la cámara del dispositivo en el mecanismo central de juego y descubrimiento.

La decisión estratégica de basar el *core* de la aplicación *Entre Diagonales* en la Inteligencia Artificial visual (reconocimiento de monumentos en tiempo real con la cámara), complementada con mecanismos de geolocalización, en lugar de soluciones basadas en marcadores físicos como los códigos QR, aborda directamente el problema de la **adulteración** y la **integridad de la experiencia**.

La **IA visual es inherentemente más segura y auténtica** que un código QR, el cual puede ser:

1. **Duplicado o Falsificado (Adulterado):** Un código QR es una imagen estática que puede ser fácilmente impresa, duplicada o pegada en un lugar incorrecto por cualquier persona. Esto permitiría a un usuario engañar al sistema sin estar realmente en el punto de interés, comprometiendo la integridad de la gamificación y los logros.
2. **Dañado o Removido:** Los marcadores físicos están sujetos al vandalismo, el clima o el deterioro, lo que podría inutilizar el punto de interés para el turista o requerir un mantenimiento constante por parte de la administración del sistema.

En contraste, el sistema de **reconocimiento visual de *Entre Diagonales*** se centra en la **identificación del monumento o ítem secreto en sí mismo**. El modelo de IA está entrenado con la arquitectura, forma y características visuales únicas del objeto (por ejemplo, la Catedral o un elemento esotérico específico), lo que asegura que:

- **Se valida la realidad física:** El usuario debe enfocar la cámara al objeto real e inamovible (el monumento) para que se active la recompensa, no a una etiqueta temporal o removible.
- **Se potencia la interacción activa y la curiosidad:** El usuario no está buscando una pegatina (código QR), sino que utiliza la cámara como una herramienta de exploración activa y lúdica que actúa como el disparador central para acceder al contenido y la gamificación.
- **Se ofrece un valor de innovación superior:** La incorporación de Inteligencia Artificial visual representa una innovación tecnológica en términos conceptuales, ya que modifica la forma en que el usuario

interactúa con la información turística. En lugar de depender exclusivamente de búsquedas, menús o recomendaciones textuales, la propuesta sitúa al entorno urbano y al patrimonio arquitectónico como disparadores directos de la experiencia digital. De este modo, el reconocimiento visual en tiempo real transforma al monumento o edificio en una interfaz de acceso al contenido, articulando percepción, desplazamiento físico e interpretación cultural en una misma acción. Esta lógica no solo introduce una capa tecnológica novedosa, sino que también redefine la relación entre usuario, espacio y aprendizaje dentro de la experiencia turística.

El uso de la IA, en este contexto, transforma la experiencia pasiva (como leer un folleto digitalizado o escanear un código QR) en una **interacción en campo**, donde el descubrimiento visual genuino es recompensado, asegurando que el contenido histórico y cultural se desbloquea en el momento exacto y lugar correcto.

2.5.5. Estrategia de Gamificación de Entre Diagonales

El proyecto utiliza mecánicas de gamificación para aumentar el *engagement*, promover la exploración activa y reforzar el aprendizaje. Las implementaciones incluyen:

- **Sistema de Insignias y Logros Visuales:** Recompensas gráficas por completar hitos.
- **Sistema de Puntos y Niveles:** Generar progresión visible.
- **Trivias Interactivas y Reforzadoras:** Preguntas sobre el punto de interés luego del escaneo para fomentar el **aprendizaje significativo y la retención de conocimiento**.
- **Avatares:** Proporcionan un sentido de identidad y pertenencia en la aplicación.
- **Sistema de Progreso Narrativo:** Desbloquear la historia de la ciudad por capítulos o zonas para mantener el interés.
- **Tabla de clasificación:** Crear una competencia activa y sana entre los usuarios de la aplicación.

2.5.6. Propuesta Diferencial de Entre Diagonales

Entre Diagonales está diseñado para incluir algunas de las características que en el relevamiento aparecieron, que no aparecen en las aplicaciones relevadas. Esto se hace mediante la integración de cuatro pilares tecnológicos y conceptuales que se muestran en la tabla 2.2:

Elemento Clave	Estado Actual del Mercado	Propuesta de <i>Entre Diagonales</i>
Enfoque Geográfico	Global o centrado en grandes ciudades.	La Plata como protagonista , destacando su patrimonio y urbanismo.
Inteligencia Artificial	Planeamiento basado en texto.	Reconocimiento de monumentos en tiempo real con la cámara (IA visual).
Tipo de Interacción	Pasiva o rutas rígidas.	Activa y en campo , recompensando la curiosidad y exploración progresiva del usuario.
Gamificación	Existente, pero desvinculada del entorno y la IA.	Sistema integral : logros y trivias activados por el reconocimiento visual o diario , integrada con contexto cultural profundo.

Cuadro 2.2: Pilares tecnológicos y conceptuales que diferencian a *Entre Diagonales*.

En resumen, *Entre Diagonales* transforma la experiencia turística pasiva del mercado actual en una **exploración activa y lúdica** al utilizar la IA visual como el disparador central para la gamificación y el acceso a contenido cultural e histórico sobre la ciudad de La Plata. Es como usar la cámara del teléfono como una llave maestra que, al escanear un monumento, no solo te dice qué es, sino que desbloquea un nivel, te premia con una insignia y te desafía con una pregunta sobre su historia.

Capítulo 3. Diseño de Entre Diagonales

3.1. Entre Diagonales VS mercado actual

La aplicación *Entre Diagonales* se posiciona para explotar las brechas identificadas en el mercado actual de aplicaciones turísticas, el cual, a pesar de la abundancia de herramientas, carece de una **integración sinérgica de tecnologías avanzadas, gamificación y enfoque local**.

En este capítulo se presentan las decisiones de diseño que dieron forma a *Entre Diagonales*, abarcando sus mecánicas principales, las interfaces desarrolladas y los elementos de gamificación implementados.

3.2. Descripción general

Entre Diagonales es una aplicación de turismo gamificado desarrollada para la Ciudad de La Plata. Su premisa central consiste en transformar el espacio urbano en un tablero de juego, donde el usuario adopta el rol de explorador y debe completar retos, recorrer itinerarios temáticos y descubrir objetos secretos distribuidos por la ciudad.

En la Figura 3.1 se presenta la pantalla principal de la aplicación. Desde esta interfaz el usuario accede de forma inmediata a la información esencial para comenzar la experiencia. En la parte superior (A) se muestra el personaje seleccionado durante el registro, acompañado por el estado climático y la temperatura actual en la ciudad utilizando la API Open Source de Open-Meteo [38]. Más abajo (B) se visualiza el progreso del nivel del usuario, incluyendo sus puntos de experiencia, el nivel alcanzado y el próximo objetivo de progreso, si corresponde.

La sección de rutas recomendadas (C) permite al jugador identificar rápidamente distintos recorridos posibles, indicando su nombre y la cantidad de paradas asociadas. Finalmente, en la parte inferior (D) se encuentra el menú principal de navegación, compuesto por las secciones: *Inicio* (pantalla actual), *Rutas* (listado completo de recorridos disponibles), *Escaneo* (interfaz destinada al reconocimiento visual y futuros usos de lectura de códigos), *Ranking* (acceso a la tabla de clasificación) y *Perfil* (información detallada sobre el progreso del usuario y su desempeño dentro del juego).



Figura 3.1: Pantalla principal de la aplicación

3.3. Recorridos

Los recorridos constituyen una de las mecánicas centrales de *Entre Diagonales*, ya que representan experiencias temáticas dentro de la ciudad que guían al usuario a través de puntos de interés histórico, arquitectónico o cultural. Cada recorrido está diseñado como una secuencia ordenada de paradas que el jugador debe visitar para completar la experiencia. Esta estructura refuerza la narrativa del explorador, promoviendo la curiosidad y el descubrimiento del entorno urbano.

3.3.1. Diseño general de los recorridos

En la pantalla principal de recorridos (Figura 3.2) se presenta una lista de rutas disponibles, cada una acompañada por su nombre, una descripción

breve, la cantidad total de paradas que la componen y el progreso actual. Este diseño permite identificar rápidamente la temática del recorrido y evaluar su complejidad o duración antes de iniciarlo.

Además, la aplicación permite filtrar los recorridos por temática como turísticos, culturales, familiares o gastronómicos, entre otros facilitando al usuario la selección de experiencias acordes a sus intereses. También es posible aplicar un filtro por dificultad: los recorridos catalogados como *Fácil* incluyen hasta 5 paradas, los de nivel *Intermedio* cuentan con más de 10 paradas y los recorridos *Difíciles* presentan más de 15 paradas.



Figura 3.2: Pantalla de recorridos

Cada recorrido está formado por una serie de paradas geolocalizadas. Cada parada contiene un título, imágenes de referencia, una descripción contextual, las coordenadas que permiten calcular la distancia desde la parada anterior, su dirección, los horarios, el valor de la entrada y una descripción adicional. La aplicación ordena automáticamente estas paradas según

el recorrido definido y presenta al usuario la secuencia correcta a seguir.

3.3.2. Progreso del recorrido

Durante el recorrido, el usuario accede a la pantalla de progreso (Figura 3.3), donde puede visualizar cuántas paradas ha completado y cuántas aún restan para finalizar la experiencia. La interfaz incluye una barra de avance, además del nombre, la distancia y el tiempo estimado hasta la siguiente parada caminando. También se presenta un listado completo de todas las paradas del recorrido, indicando cuáles de ellas contienen objetos secretos por descubrir.

Asimismo, la pantalla incorpora dos botones principales: uno destinado a completar una parada mediante reconocimiento visual (A) y otro para acceder al mapa general del recorrido (B). Ambos elementos serán detallados en secciones posteriores.

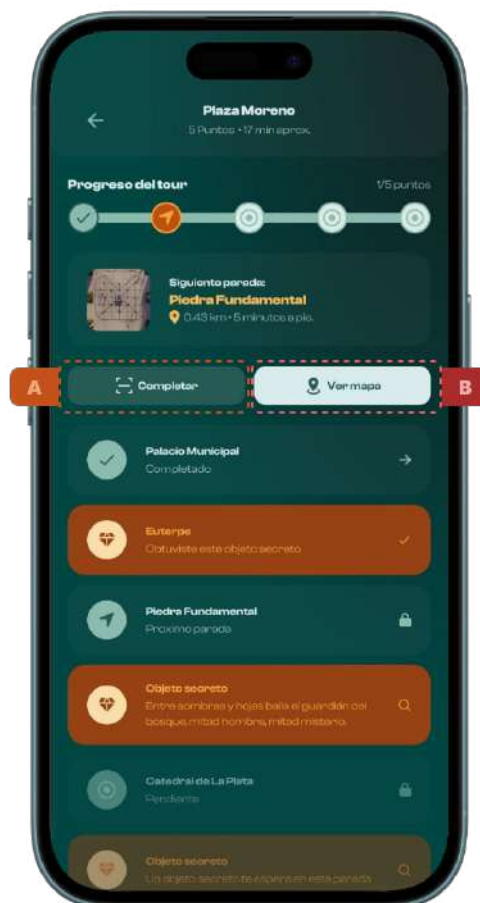


Figura 3.3: Pantalla de progreso de recorrido

3.3.3. Mapa del recorrido

La visual del mapa (Figura 3.4) muestra el trazado completo a la siguiente parada mediante una polilínea y marcadores para cada parada. El usuario puede observar la parada actual, las futuras y su propia ubicación en tiempo real. Este diseño permite orientarse fácilmente y tomar decisiones sobre cómo llegar al siguiente punto de interés.



Figura 3.4: Pantalla del mapa del recorrido

En la parte superior (A) se visualizan la cantidad total de paradas del recorrido y el tiempo estimado para completarlo. Este tiempo se calcula a partir de la ubicación actual del usuario hacia la primera parada, empleando la API de rutas de código abierto OpenRouteService [39].

Entre cada parada del listado (B), el jugador puede observar tanto la distancia como el tiempo estimado para desplazarse desde su ubicación actual hasta la siguiente parada, considerando el modo de desplazamiento a

pie.

En la segunda imagen, en la sección (C), se destaca de manera clara y diferenciada el punto próximo a alcanzar, junto con la ruta trazada en el mapa desde la ubicación del usuario hasta dicha parada.

3.3.4. Escaneo de una parada u objeto secreto

La funcionalidad de escaneo permite al usuario capturar una imagen de una parada o de un objeto secreto asociado al recorrido como se ve en al Figura 3.5, con el objetivo de validar su descubrimiento dentro del sistema de gamificación. Para ello, la aplicación presenta una interfaz de cámara integrada, acompañada de indicaciones visuales que orientan al usuario sobre cómo realizar la captura, por ejemplo sugiriendo apuntar de frente al lugar para mejorar las condiciones de reconocimiento.



Figura 3.5: Pantalla de captura fotográfica de una parada.

3.3.5. Interacción con cada parada

Al llegar a una parada, el usuario accede a la vista de lugar (Figura 3.6), donde se presenta la imagen del mismo, un texto informativo, un dato curioso e información sobre las visitas. Los datos curiosos fueron seleccionados a partir de diversas fuentes de divulgación, en particular de blogs especializados en la ciudad de La Plata, que recopilan hechos, anécdotas y aspectos poco conocidos de sus espacios urbanos [40]. Se priorizó aquella información que resulta llamativa o no evidente para el visitante promedio, con el objetivo de enriquecer la experiencia y fomentar un mayor interés por el entorno. Para completar efectivamente la parada, el usuario debe presionar el botón *Completar* de la pantalla (Figura 3.3), lo que activa la cámara del dispositivo e inicia el proceso de reconocimiento visual (ver Sección 4.6).

Si el reconocimiento es exitoso, la parada se valida como visitada, se desbloquea la información completa del lugar y en caso de existir para dicha parada una trivia para responder (ver Sección 3.7.4). Esta acción actualiza el progreso general del recorrido y habilita la transición hacia la siguiente parada.

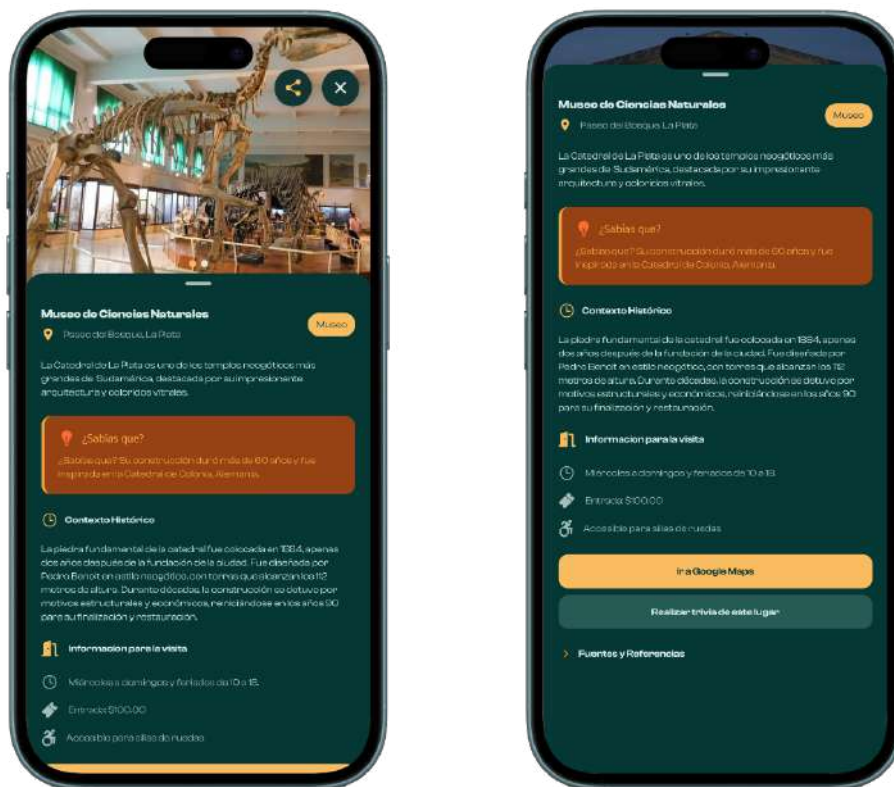


Figura 3.6: Pantalla una parada del recorrido

3.3.6. Finalización del recorrido

Una vez completadas todas las paradas del recorrido, la aplicación presenta la pantalla de finalización (Figura 3.7), donde el usuario puede visualizar un resumen de la experiencia realizada. Esta vista incluye la cantidad de objetos secretos descubiertos sobre el total disponible, las trivias respondidas correctamente, y las recompensas obtenidas, tales como monedas de plata y puntos de experiencia, que contribuyen al avance dentro del sistema general de gamificación (ver Sección 3.7).



Figura 3.7: Pantalla de fin del recorrido

Esta instancia tiene como propósito reforzar el logro alcanzado e incentivar al jugador a continuar explorando nuevos recorridos dentro de la aplicación.

3.3.7. Perfil del usuario

La sección de perfil del usuario centraliza la información asociada al progreso individual dentro de la aplicación. En esta pantalla (Figura 3.8) se visualizan datos básicos de la cuenta, junto con indicadores de avance vinculados al sistema de gamificación, como la experiencia acumulada (XP), el nivel alcanzado, la moneda virtual obtenida y el progreso hacia el siguiente rango.

Además, se presentan resúmenes de la actividad realizada, incluyendo las rutas completadas o en curso, la cantidad de paradas recorridas, los objetos secretos descubiertos y los logros desbloqueados. Esta vista cumple una doble

función: por un lado, permite al usuario consultar de forma clara su historial y estado actual dentro del sistema; por otro, refuerza la motivación para continuar explorando mediante elementos visuales de progreso, colección y recompensa.

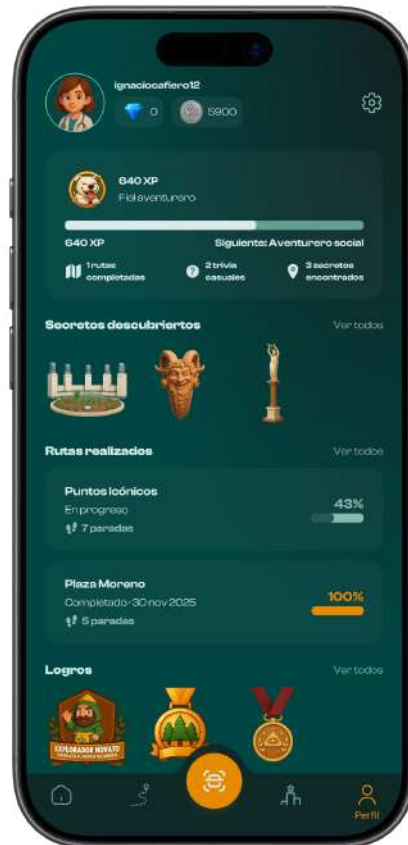


Figura 3.8: Pantalla de perfil del usuario.

3.3.8. Recorridos implementados

Actualmente, *Entre Diagonales* cuenta con dos recorridos desarrollados y plenamente funcionales, cada uno con su propia temática y conjunto de paradas:

- **Recorrido 1: Puntos icónicos.** Incluye las siguientes paradas: Museo de Ciencias Naturales, Planetario Ciudad de La Plata, Casa Curchet, Casa de Gobierno, Monumento a San Martín, Pasaje Dardo Rocha y Teatro Argentino.

- **Recorrido 2: Plaza Moreno.** Compuesto por las paradas: Palacio Municipal, Piedra Fundacional, Catedral, Museo Dardo Rocha y Palacio de Justicia.

La selección de estos puntos se realizó considerando distintos criterios. Por un lado, se priorizaron lugares de relevancia histórica, cultural y turística dentro de la ciudad de La Plata. Por otro, se tuvo en cuenta la proximidad geográfica entre las paradas, con el objetivo de favorecer recorridos caminables y coherentes desde el punto de vista espacial.

En particular, el recorrido centrado en Plaza Moreno aprovecha la alta concentración de puntos de interés en esa zona, donde es posible observar en un mismo entorno la Catedral, el Palacio Municipal y el Museo Dardo Rocha, además de elementos destacados de la propia plaza, como sus fuentes, esculturas y otros detalles urbanos mencionados en fuentes de divulgación [40].

Por su parte, el recorrido de Puntos Icónicos responde a un criterio diferente, orientado a la selección de sitios emblemáticos de la ciudad. Si bien presenta una mayor dispersión geográfica, se procuró mantener distancias razonables entre paradas, buscando un equilibrio entre representatividad y accesibilidad para el usuario.

Estos recorridos sirven como base inicial del sistema y ejemplifican la estructura general utilizada para futuras expansiones.

3.4. Objetos Secretos

Los objetos secretos constituyen una de las mecánicas más distintivas de *Entre Diagonales*, ya que introducen una capa adicional de exploración y descubrimiento dentro de la ciudad. Cada recorrido puede incluir uno o varios objetos ocultos asociados a determinadas paradas, los cuales deben ser identificados por el jugador mediante una pista y posteriormente confirmados a través del reconocimiento visual. Esta dinámica refuerza la atención al entorno, fomenta la curiosidad y aporta un componente lúdico complementario a la experiencia general del recorrido.

3.4.1. Funcionamiento general

Cada parada puede contener uno o más objetos secretos vinculados a un punto físico del entorno urbano. El jugador recibe una pista breve que describe o sugiere características del objeto a encontrar, sin revelar su ubicación exacta. Una vez interpretada la pista, el usuario debe presionar el botón correspondiente para activar la cámara del dispositivo e iniciar el proceso de reconocimiento visual.

Si el modelo identifica correctamente el objeto secreto, la aplicación registra el hallazgo, lo marca como desbloqueado y otorga puntos de experien-

cia adicionales. En caso contrario, se permite reintentar más adelante sin penalización, manteniendo el flujo natural de exploración.

3.4.2. Progreso dentro del recorrido

En la pantalla de progreso del recorrido, previamente presentada en la Figura 3.3, se incluye una referencia al estado de los objetos secretos asociados a cada parada. El usuario puede visualizar cuántos objetos lleva descubiertos y cuántos restan para completar la experiencia del recorrido en curso. Esta vista ofrece un seguimiento claro del avance global, combinando la progresión estándar del recorrido con los hallazgos opcionales.

3.4.3. Visualización de un objeto secreto encontrado

Cuando un objeto secreto es identificado exitosamente, la aplicación muestra una pantalla dedicada (Figura 3.9) con la imagen animada generada por inteligencia artificial, acompañada del nombre del objeto y una breve descripción contextual. Este diseño busca reforzar el impacto del hallazgo, otorgando un momento de recompensa visual y narrativa al jugador.

Además de registrar el descubrimiento, esta pantalla otorga los puntos de experiencia correspondientes e informa si el hallazgo contribuye al progreso de una insignia o logro específico dentro del sistema de gamificación.



Figura 3.9: Pantalla correspondiente al objeto secreto *Euterpe*, presente en el recorrido *Plaza Moreno*.

3.4.4. Colección de objetos secretos

La aplicación cuenta también con una sección donde reúne todos los objetos secretos del recorrido o de la aplicación en general (Figura 3.10). En esta vista, el usuario puede observar tanto los objetos ya descubiertos como aquellos que aún permanecen ocultos, los cuales se muestran de manera bloqueada para sugerir su existencia sin revelar su contenido.

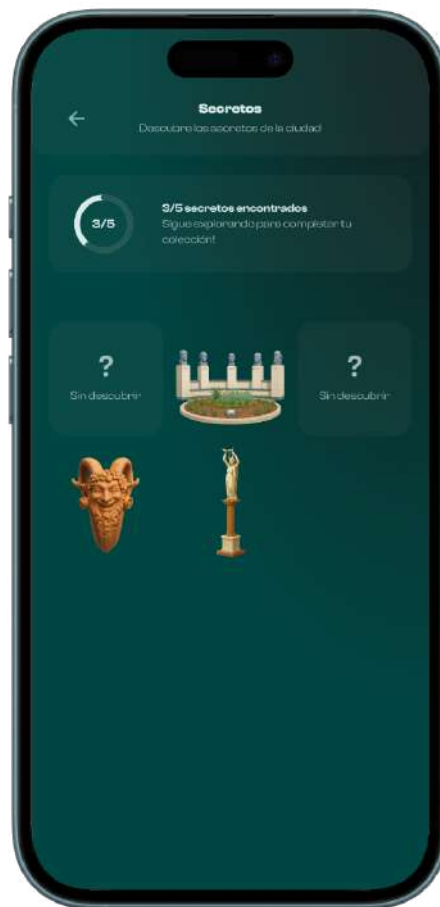


Figura 3.10: Pantalla de Objetos Secretos

Esta colección funciona como un incentivo continuo que promueve seguir jugando, invita a explorar nuevos recorridos y motiva a completar el conjunto total de objetos disponibles.

3.5. Personaje Principal: Lúmino

Con el objetivo de reforzar la cercanía con el usuario y aportar una capa adicional de narrativa a la experiencia, *Entre Diagonales* incorpora un compañero de aventuras denominado **Lúmino**. Este personaje, simpático y enigmático, está inspirado en la estética amigable y expresiva de asistentes virtuales contemporáneos como Duolingo, pero adaptado al imaginario simbólico característico de la ciudad de La Plata. El nombre del personaje surge de la idea de “luz” como símbolo de descubrimiento y conocimiento, en relación con la exploración de la ciudad y sus secretos.

Lúmino ¹ se presenta como un entusiasta explorador de secretos urbanos, profundamente fascinado por la historia, los símbolos y la tradición masónica que forman parte del patrimonio cultural de la ciudad. Su rol dentro de la aplicación es acompañar al jugador durante los recorridos, brindar pistas relacionadas con objetos secretos, celebrar los logros obtenidos y ofrecer motivación continua para seguir avanzando.

Desde una perspectiva de diseño, Lúmino cumple una doble función. Por un lado, actúa como guía narrativo que otorga coherencia temática al sistema de descubrimiento y gamificación. Por otro, funciona como un elemento emocional que genera identidad y continuidad en la experiencia del usuario. Gracias a su presencia constante en distintas pantallas, como el inicio, las pistas y las recompensas, Lúmino fortalece el vínculo entre jugador y aplicación, haciendo que la exploración urbana resulte más inmersiva, divertida y accesible.



Figura 3.11: Lúmino

3.6. Personajes

Además de Lúmino, *Entre Diagonales* incorpora un conjunto de seis personajes que el usuario puede elegir al momento de crear su cuenta. Estos personajes acompañan al jugador durante toda la experiencia y funcionan como una representación visual de su perfil dentro de la aplicación. Su diseño busca transmitir diversidad, cercanía y afinidad con distintos estilos de ex-

¹El diseño de Lumino fue generado mediante herramientas de inteligencia artificial desarrolladas por OpenAI, como parte del proceso de prototipado de la aplicación.

ploración, de manera que cada jugador pueda identificarse con el personaje que mejor refleje su personalidad o motivación.²

Dado que *Entre Diagonales* se presenta como un prototipo funcional, la implementación actual de los personajes se limita a su dimensión representativa y estética. No obstante, como línea de trabajo futuro, se proyecta dotar a cada personaje de habilidades, ventajas o características diferenciales que permitan personalizar aún más la experiencia del usuario, reforzando así la identificación y el estilo de juego individual.

La selección del personaje se realiza en la última etapa del proceso de registro, donde la aplicación presenta una cuadrícula de seis opciones distribuidas en tres columnas y dos filas (Figura 3.12). Cada personaje puede ser pulsado para visualizarlo resaltado y seleccionarlo antes de comenzar la aventura. Esta pantalla utiliza un diseño claro y accesible que facilita la toma de decisiones y refuerza la sensación de personalización inicial.

²Los diseños visuales de los personajes fueron generados mediante herramientas de inteligencia artificial desarrolladas por OpenAI, como parte del proceso de prototipado de la aplicación.



Figura 3.12: Pantalla de selección de personaje

A continuación, se describen los seis personajes disponibles:

- **Emilia – La médica curiosa.** Emilia es una joven doctora apasionada por la historia de la ciudad. Siempre está dispuesta a compartir datos curiosos sobre monumentos y a acompañar a otros usuarios durante los recorridos. Cree firmemente que la salud y el conocimiento van de la mano.
- **Roberta – La exploradora entusiasta.** Roberta es la integrante más joven del grupo, pero también la más valiente. Le encanta descubrir secretos escondidos en los rincones menos esperados. Su energía, su sonrisa constante y su espíritu competitivo la convierten en una compañera ideal para quienes buscan una experiencia dinámica y divertida.
- **Julio – El guía urbano.** Julio creció recorriendo las diagonales de La

Plata. Conoce cada calle, edificio y anécdota de la ciudad. Es amigable, sereno y siempre tiene una recomendación especial para vivir cada recorrido como un verdadero local.

- **Juan – El creativo hipster.** Amante de la fotografía, el arte callejero y los cafés ocultos, Juan ofrece una mirada distinta de la ciudad. Es el compañero ideal para quienes buscan rutas alternativas, puntos de vista únicos o simplemente una experiencia más relajada y artística.
- **José – El historiador sabio.** José ha dedicado su vida a investigar los misterios y la evolución urbana de La Plata. Conserva datos, fechas y nombres que pocos recuerdan. Es perfecto para los jugadores interesados en comprender la historia en profundidad y conectar cada recorrido con el pasado de la ciudad.
- **Noelia – La estrategia profesional.** Noelia es organizada, decidida y siempre va un paso adelante. Disfruta resolver trivias, completar logros y desbloquear todo el contenido posible. Es ideal para quienes buscan una compañera competitiva, analítica y altamente comprometida con cada desafío.

3.7. Gamificación

3.7.1. Niveles

El sistema de niveles de *Entre Diagonales* está diseñado para reflejar el progreso del jugador a través de la acumulación de *Puntos de Sabiduría*. Estos puntos se obtienen al completar recorridos, descubrir objetos secretos, resolver trivias y avanzar dentro de las distintas mecánicas de exploración urbana. Cada nivel está asociado a un animal representativo, perteneciente a la fauna actual, histórica o prehistórica de la región, lo que permite incorporar identidad, simbolismo y cohesión temática dentro de la experiencia³.

El **Nivel 1** corresponde al *Hornero*, un ave emblemática argentina que simboliza perseverancia y esfuerzo. Con cero puntos requeridos, este nivel inicial recibe el nombre de *Explorador urbano* y representa el primer paso del jugador dentro de la aplicación.

El **Nivel 2**, alcanzado con 500 puntos, está representado por el *Dogo argentino*, un animal asociado a la lealtad y la determinación. Su nombre, *Fiel aventurero*, destaca el compromiso del jugador con sus primeras misiones.

El **Nivel 3** requiere 1000 puntos y está simbolizado por el *Carpincho*, conocido por su carácter social y tranquilo. Bajo el título de *Aventurero*

³Los diseños visuales de los diferentes niveles fueron generados mediante herramientas de inteligencia artificial desarrolladas por OpenAI, como parte del proceso de prototipado de la aplicación.

social, este nivel reconoce al jugador que ya ha explorado distintos espacios y comienza a integrarse con los desafíos más complejos.

Al llegar al **Nivel 4**, con 1500 puntos, el jugador obtiene el título de *Rastreador elegante*. Este nivel está representado por el *Yaguareté*, un felino histórico de los bosques sudamericanos, que transmite habilidad, precisión y conocimiento del entorno.

El **Nivel 5**, alcanzado con 2000 puntos, está inspirado en el *Gliptodonte*, un animal prehistórico caracterizado por su imponente protección natural. Su denominación, *Guardián acorazado*, simboliza solidez y constancia en la exploración.

Finalmente, el **Nivel 6** requiere 3000 puntos y está asociado al *Diente de sable*, uno de los depredadores prehistóricos más icónicos. Su título, *Sabio formidable*, representa el dominio total del jugador sobre las dinámicas del juego, alcanzando el grado más alto de experiencia y conocimiento dentro de la aplicación.



Figura 3.13: Ilustraciones de los niveles disponibles en *Entre Diagonales*.

En conjunto, estos niveles (Figura 3.13) no solo reflejan el progreso cuantitativo del usuario, sino que también incorporan elementos culturales y na-

turales de la región, contribuyendo a una experiencia de gamificación coherente, motivadora y vinculada con la identidad local.

3.7.2. Monedas de plata y gemas

El sistema de economía interna de *Entre Diagonales* está compuesto por dos tipos de recursos: las **monedas de plata** y las **gemas**. Ambos elementos forman parte del sistema de gamificación y permiten extender la interacción, ofrecer recompensas adicionales y habilitar mecánicas estratégicas dentro de la aplicación.

Monedas de plata

Las monedas de plata representan la moneda principal del juego y se obtienen a través de distintas acciones que reflejan el avance del usuario dentro de la experiencia. Se entregan como recompensa por descubrir objetos secretos, responder correctamente trivias y completar recorridos. Las cantidades otorgadas son las siguientes:

- **Respuesta correcta en una trivia:** 100 monedas.
- **Punto completado dentro de un recorrido:** 200 monedas.
- **Objeto secreto descubierto:** 150 monedas.

Las monedas (Figura 3.14) pueden utilizarse para adquirir ayudas en ciertos desafíos, principalmente dentro de las trivias. Actualmente, la aplicación ofrece la ayuda *50/50*, que elimina la mitad de las opciones incorrectas a cambio de un costo de 400 monedas de plata. El uso de esta asistencia implica una reducción directa en la obtención de monedas, ya que su propósito es brindar una ventaja estratégica al usuario sin otorgar recompensas adicionales.



Figura 3.14: Ilustración de la moneda de plata.

Aunque su implementación completa excede el alcance actual de esta tesis, las monedas también están pensadas para ser utilizadas en un sistema futuro de personalización del avatar, donde el usuario podrá adquirir accesorios, elementos cosméticos o mejoras visuales que acompañen su experiencia de exploración.

Gemas

Las gemas (Figura 3.15) constituyen un recurso premium dentro de *Entre Diagonales*. A diferencia de las monedas de plata, no se obtienen por acciones cotidianas del recorrido, sino que requieren la consecución de logros específicos. Esta característica las convierte en un recurso más valioso y de obtención limitada.



Figura 3.15: Ilustración de la gema.

Si bien en la versión actual de la aplicación las gemas no poseen aún una funcionalidad activa, se proyectan varios usos futuros que reforzarán su valor dentro del sistema de gamificación. Entre las posibles funcionalidades previstas se encuentran:

- Desbloqueo de personalizaciones exclusivas del avatar.
- Acceso anticipado a rutas temáticas especiales o de mayor dificultad.
- Obtención de pistas avanzadas para objetos secretos particularmente difíciles.
- Activación de bonificaciones temporales, como multiplicadores de experiencia.

La incorporación de ambos recursos permite diseñar una economía mixta que equilibra recompensas accesibles (monedas de plata) con un recurso de mayor rareza y prestigio (gemas), ampliando las posibilidades de progresión y motivación del jugador a largo plazo.

3.7.3. Retos y logros

El sistema de retos y logros de *Entre Diagonales* tiene como objetivo reforzar la motivación del jugador mediante metas claras y progresivas, asociadas a diferentes dimensiones de la experiencia: exploración de recorridos, resolución de trivias y descubrimiento de objetos secretos. Cada logro se estructura en tres niveles de dificultad; al completar un nivel, se desbloquea automáticamente el siguiente, favoreciendo la continuidad en el uso de la aplicación.

Las recompensas por cada nivel de logro combinan experiencia y recursos premium, de acuerdo con el siguiente esquema:

- **Nivel 1:** 5 gemas + 20 puntos de experiencia (XP).
- **Nivel 2:** 10 gemas + 20 puntos de experiencia (XP).
- **Nivel 3:** 15 gemas + 20 puntos de experiencia (XP).

De este modo, los logros no solo reconocen acciones específicas del jugador, sino que también contribuyen de manera directa a su progresión general dentro del sistema de gamificación.

Logro *Explorador*

El logro *Explorador* está asociado a la finalización de recorridos dentro de la ciudad. Su propósito es incentivar al usuario a completar las rutas propuestas y a avanzar de manera sostenida en la exploración urbana.

Este logro se compone de los siguientes niveles:

- **Nivel 1 – Explorador Novato:** se obtiene al completar el primer recorrido.
- **Nivel 2 – Perito Moreno:** se obtiene al completar todos los recorridos disponibles en la aplicación.

Logro *Maestro de Trivias*

El logro *Maestro de Trivias* premia la correcta resolución de preguntas de trivia, reforzando el componente de aprendizaje y la atención a los contenidos históricos y culturales presentados.

Sus niveles se definen de la siguiente manera:

- **Nivel 1 – Aprendiz de Trivia:** se obtiene al responder correctamente 1 trivia.
- **Nivel 2 – Maestro de Trivia:** se obtiene al responder correctamente 3 trivias.
- **Nivel 3 – Sabio de Trivias:** se obtiene al responder correctamente 5 trivias.

Logro *Cazador de Secretos*

El logro *Cazador de Secretos* está vinculado al descubrimiento de objetos secretos durante los recorridos. Este logro refuerza la dimensión de exploración detallada del entorno y el uso del reconocimiento visual.

Sus niveles son los siguientes:

- **Nivel 1 – Iniciado del Bosque:** se obtiene al descubrir el primer secreto.

- **Nivel 2 – Rastreador de Símbolos:** se obtiene al descubrir 3 secretos.
- **Nivel 3 – Guardián del Plano Perfecto:** se obtiene al descubrir 5 secretos.

Visualización de logros en la aplicación

La aplicación cuenta con una pantalla específica para la visualización de logros (Figura 3.16), donde el usuario puede consultar su progreso en cada uno de ellos, identificar qué niveles ha completado y cuáles permanecen bloqueados. Cada logro se presenta con su nombre, su iconografía asociada⁴ y un indicador de estado, lo que facilita comprender de un vistazo qué objetivos se han alcanzado y cuáles constituyen los próximos desafíos a completar.

⁴Los diseños visuales de los diferentes logros fueron generados mediante herramientas de inteligencia artificial desarrolladas por OpenAI, como parte del proceso de prototipado de la aplicación.



Figura 3.16: Pantalla de visualización de logros en *Entre Diagonales*.

3.7.4. Trivias

Las trivias constituyen una de las mecánicas educativas y lúdicas más importantes de *Entre Diagonales*, ya que permiten reforzar el conocimiento adquirido durante los recorridos mediante preguntas relacionadas con la historia, los monumentos y la cultura de la ciudad de La Plata. Esta funcionalidad cumple un doble propósito: promover el aprendizaje significativo y ofrecer una instancia de juego dinámico que complementa la exploración urbana.

Cada trivia presenta una pregunta con cuatro opciones de respuesta, de las cuales solo una es correcta. La interfaz está diseñada para ser clara, accesible y adecuada para su uso en contextos de movilidad, permitiendo responder con rapidez y continuidad durante la experiencia del recorrido. En la Figura 3.17 se muestra la visualización principal de una trivia en la

aplicación.

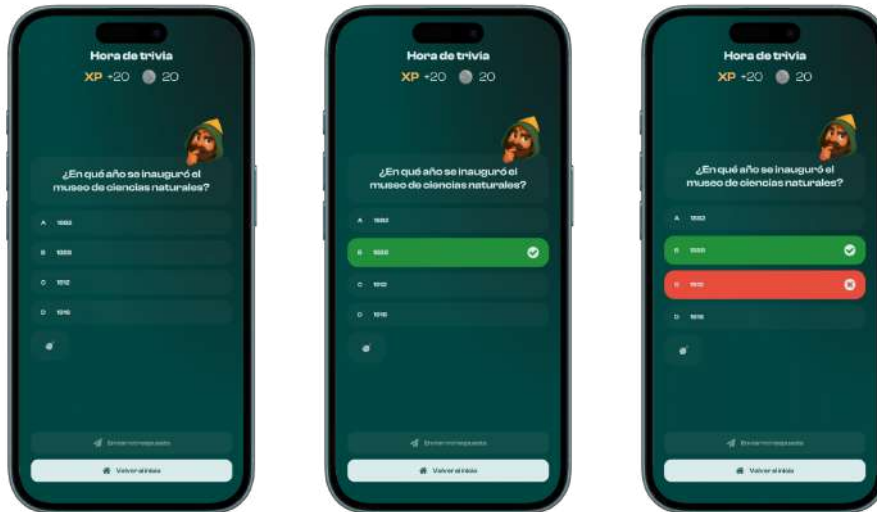


Figura 3.17: Pantalla de resolución de trivia en *Entre Diagonales*.

Recompensas y progresión

Responder correctamente a una trivia otorga **100 monedas de plata** y suma puntos de experiencia que contribuyen al progreso general del usuario dentro del sistema de gamificación. Además, las trivias están directamente vinculadas con los logros de trivia, cuyos niveles se desbloquean según la cantidad de respuestas correctas que alcanza el usuario.

De este modo, las trivias no solo constituyen una prueba de conocimiento, sino también un mecanismo fundamental para avanzar en logros, obtener monedas y acceder a futuras funcionalidades dentro de la aplicación.

Ayudas y estrategias

Para brindar una experiencia más accesible y dinámica, la aplicación ofrece la ayuda *50/50*, que elimina la mitad de las opciones incorrectas y aumenta las probabilidades de seleccionar la respuesta correcta. Esta ayuda tiene un costo de **400 monedas de plata** y es opcional; su uso implica un sacrificio económico, ya que no otorga recompensas adicionales ni multiplica la ganancia de monedas en caso de acierto.

El propósito de esta mecánica es incorporar una dimensión estratégica: el usuario debe decidir si vale la pena gastar sus monedas para asegurar una respuesta o si prefiere arriesgarse y conservarlas para futuras compras o beneficios.

Integración con el recorrido y la narrativa

Las trivias se pueden activar al finalizar ciertas paradas, reforzando la relación entre lo observado en el entorno y el conocimiento adquirido. Esta integración contribuye al carácter educativo de la aplicación, promoviendo la comprensión histórica y cultural de los monumentos y espacios visitados.

Asimismo, la presencia de Lúmino como guía ayuda a contextualizar las trivias mediante breves mensajes o celebraciones al responder correctamente, reforzando la narrativa del compañero de aventura y generando una experiencia más cercana y motivadora.

En conjunto, las trivias aportan una dinámica de aprendizaje activo que combina exploración, memoria y estrategia, posicionándose como una mecánica central dentro del ecosistema de juego de *Entre Diagonales*.

3.7.5. Tabla de clasificación

La tabla de clasificación es una de las mecánicas sociales más relevantes de *Entre Diagonales*, ya que permite comparar el progreso de los jugadores y fomentar la motivación mediante competencia amistosa. Su función principal es mostrar el desempeño relativo de cada usuario en términos de puntos de sabiduría (XP), incentivando a continuar explorando, resolviendo trivias y descubriendo secretos para ascender posiciones.

La aplicación ofrece dos vistas diferenciadas para consultar el ranking: **Global** y **Nivel Actual**. En la vista *Global*, se muestran todos los usuarios ordenados según la cantidad total de XP acumulado desde sus primeras actividades. Esta vista permite observar el progreso histórico y la constancia a largo plazo. Por otro lado, la vista *Nivel Actual* presenta únicamente a los usuarios que se encuentran dentro del mismo nivel de progresión, lo que favorece comparaciones más equilibradas y una competencia justa entre jugadores con experiencia similar.

En la Figura 3.18 se muestra la visualización de la tabla de clasificación dentro de la aplicación, donde es posible identificar el nombre del jugador, su nivel, su posición relativa y la cantidad total de puntos de sabiduría acumulados.



Figura 3.18: Pantalla de la tabla de clasificación en *Entre Diagonales*.

Puntos de sabiduría (XP)

Los puntos de sabiduría constituyen la métrica central para determinar la posición del usuario dentro del ranking. Su distribución está diseñada para reflejar fielmente el esfuerzo, la exploración y el aprendizaje del jugador. Las fuentes de experiencia son las siguientes:

- **Experiencia por parada completada:** 100 XP.
- **Experiencia por logro obtenido:** 20 XP.
- **Experiencia por trivia respondida correctamente:** 20 XP.
- **Experiencia por objeto secreto descubierto:** 30 XP.

Estos valores permiten equilibrar la progresión entre distintos estilos de juego: quienes prefieren completar recorridos avanzan mediante las paradas; quienes disfrutan aprender y responder trivias también progresan de manera constante; y quienes se enfocan en los secretos reciben una recompensa

proporcional al desafío que implica cada descubrimiento. Además, los logros funcionan como un complemento que otorga experiencia adicional por hitos relevantes.

En conjunto, la tabla de clasificación y el sistema de puntos de sabiduría conforman una mecánica social y progresiva que refuerza la motivación del jugador, promoviendo la constancia, el aprendizaje activo y la exploración continua de la ciudad.

3.8. Identidad visual: tipografía, colores y logotipo

La identidad visual de *Entre Diagonales* se desarrolló con el objetivo de transmitir una estética moderna, cercana y vinculada con la ciudad de La Plata. Esta identidad se compone de tres elementos principales: la tipografía, la paleta cromática y el logotipo de la aplicación.

3.8.1. Tipografía

Para el diseño visual de la aplicación, se seleccionó la tipografía **Clash Display**, distribuida por Indian Type Foundry a través de su plataforma Fontshare. Esta fuente se encuentra bajo la licencia *Free Font End User License Agreement* (FF EULA), que permite su uso sin coste para proyectos personales y comerciales, incluyendo aplicaciones móviles.

La elección de Clash Display responde a su legibilidad, estilo contemporáneo y coherencia con la identidad gráfica general de la aplicación. Su licencia abierta facilita su integración directa en el paquete de la app, sin restricciones en cuanto a dispositivos, número de instalaciones o escalabilidad. De acuerdo con los términos de la licencia:

- El uso de la fuente en aplicaciones móviles está permitido sin costo adicional.
- No se requiere atribución obligatoria al autor, aunque puede incluirse de manera voluntaria.
- No se permite la modificación, redistribución o extracción del archivo original de la fuente.
- Su uso en logotipos, material gráfico y elementos visuales está autorizado.

La incorporación de Clash Display cumple plenamente con los requisitos legales y garantiza una implementación correcta, profesional y visualmente consistente dentro del proyecto.

3.8.2. Logotipo

El logotipo de *Entre Diagonales* (Figura 3.19) fue diseñado con un enfoque minimalista, manteniendo una estética limpia y reconocible⁵. Para su generación se utilizaron herramientas de inteligencia artificial, mediante un proceso iterativo de prueba y refinamiento a partir de distintas descripciones textuales. En este proceso se buscó integrar elementos representativos de la ciudad y de la funcionalidad de la aplicación.

En particular, el diseño se estructura a partir de una forma que combina una ventana en forma de ícono de marcador de ubicación utilizado en aplicaciones de mapas, en alusión al uso del posicionamiento dentro del sistema. En su interior, se presentan distintas siluetas representativas, entre ellas una versión estilizada de la Catedral de La Plata (uno de los íconos arquitectónicos más relevantes de la ciudad), junto con una rosa de los vientos y un camino, elementos que remiten tanto a la exploración como al trazado característico de diagonales y avenidas.

El objetivo fue transmitir identidad local sin sobrecargar la composición, combinando simplicidad con un detalle visual distintivo que refuerza la temática de exploración urbana.



Figura 3.19: Logotipo de *Entre Diagonales*.

3.8.3. Paleta cromática

La paleta de colores utilizada en *Entre Diagonales* toma como referencia los numerosos parques, plazas y áreas verdes que caracterizan a la ciudad de La Plata. El color verde predomina en la interfaz como símbolo de naturaleza, serenidad y exploración. Se complementa con tonos cálidos y contrastes

⁵El diseño visual del logotipo fue generado mediante herramientas de inteligencia artificial desarrolladas por OpenAI, como parte del proceso de prototipado de la aplicación.

suaves que contribuyen a una estética equilibrada y accesible. Los principales colores empleados son los siguientes:

- #31544F — Verde profundo, color principal de la interfaz.
- #FAF9F8 — Blanco cálido, utilizado como color base de textos.
- #7A330F — Naranja, destacado para llamadas a la acción.
- #dbf0e6 — Verde pastel, color para algunos textos.

A esta paleta de colores se le realizaron pruebas de contraste con la herramienta web Colour Contrast Checker [41]. Las combinaciones que se probaron fueron el Blanco cálido sobre el Verde profundo y el Verde pastel sobre el Naranja ya que son las combinaciones presentes en el prototipo.

Esta combinación cromática aporta cohesión, claridad y una identidad visual propia, reforzando la relación entre la aplicación y el paisaje urbano-ambiental de la ciudad.

Capítulo 4. Desarrollo de Entre Diagonales

4.1. Resumen del desarrollo

Este capítulo describe el proceso de diseño y construcción de *Entre Diagonales*, detallando de manera integral la arquitectura general del sistema, el desarrollo del modelo de reconocimiento visual y la implementación de la aplicación móvil. El desarrollo se apoyó en decisiones técnicas fundamentadas en experimentación controlada, mediciones de rendimiento reales y pruebas realizadas en dispositivos físicos, con el objetivo de validar la viabilidad del sistema en un contexto urbano real.

4.2. Arquitectura y visión general del sistema

El sistema se diseñó con una arquitectura modular, como se ve en la Figura 4.1 de tres capas claramente diferenciadas, que interactúan entre sí de forma desacoplada para facilitar el mantenimiento y la evolución independiente de cada componente.

Capa de reconocimiento visual: modelos de clasificación de imágenes basados en arquitecturas convolucionales, entrenados específicamente para identificar puntos de interés arquitectónicos y objetos urbanos relevantes de La Plata. Los modelos se ejecutan en el servidor, procesando las imágenes enviadas desde la aplicación móvil y devolviendo los resultados de clasificación a través de la API.

Capa backend: una API REST desarrollada con Django [42] y Django REST Framework, desplegada en Railway. Esta capa centraliza la lógica de negocio del sistema: administración de usuarios, recorridos, puntos del recorrido, objetos secretos, el sistema de gamificación y la validación de las predicciones generadas por los modelos de visión por computadora.

Capa de aplicación móvil: aplicación desarrollada con Expo [43] y React Native [44], que actúa como la interfaz principal con el usuario. Integra el uso de la cámara y la geolocalización para que los usuarios exploren recorridos e interactúen con el entorno urbano, y consume los servicios proporcionados por la API para registrar avances y sincronizar el estado del recorrido.

Las decisiones tecnológicas adoptadas estuvieron guiadas por criterios de rendimiento, portabilidad, simplicidad operativa y escalabilidad, priorizando especialmente la correcta ejecución en dispositivos móviles de gama media y la sostenibilidad del proyecto a largo plazo.

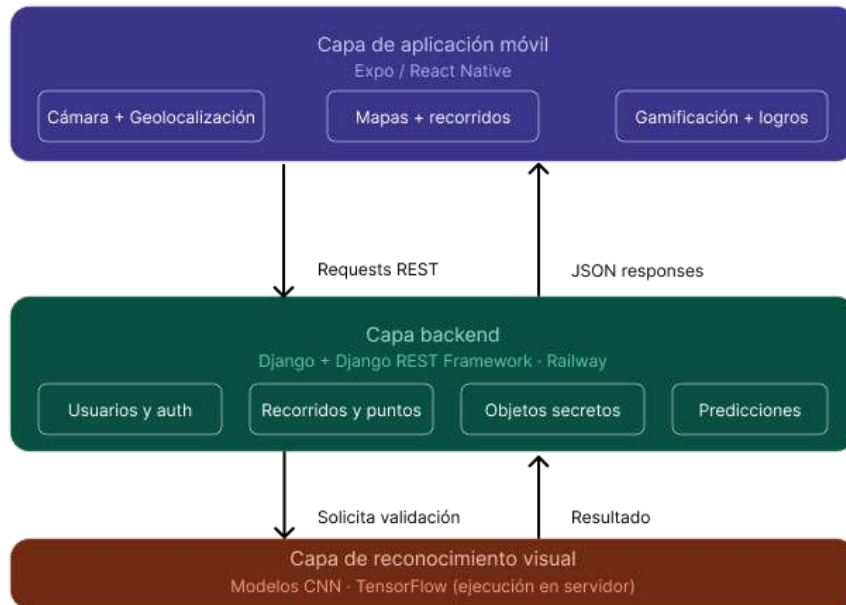


Figura 4.1: Arquitectura general del sistema: interacción entre capas.

4.3. Tecnologías seleccionadas y justificación

La selección tecnológica del prototipo respondió a criterios de rendimiento, facilidad de integración, costos operativos y adecuación al problema planteado. Dado que el proyecto combina reconocimiento visual, servicios backend y una aplicación móvil multiplataforma, fue necesario elegir herramientas que permitieran construir una solución funcional, escalable y viable dentro del alcance de una tesina.

4.3.1. Tecnologías de inteligencia artificial

Para el componente de reconocimiento visual se utilizaron **TensorFlow** y **Keras**, dos bibliotecas ampliamente adoptadas en el ámbito del aprendizaje automático. TensorFlow se empleó como framework principal para el entrenamiento y exportación de modelos, mientras que Keras permitió definir, entrenar y validar arquitecturas de manera más sencilla mediante una API de alto nivel. La elección de estas herramientas se justificó por su madurez, abundante documentación, compatibilidad con flujos de trabajo de clasificación de imágenes y facilidad para integrar modelos posteriormente en entornos productivos.

Dentro de las arquitecturas evaluadas, se seleccionó **MobileNetV2** como modelo final de clasificación de imágenes. MobileNetV2 es una red neu-

ronal convolucional diseñada específicamente para dispositivos con recursos limitados, como teléfonos móviles, priorizando un equilibrio entre precisión y eficiencia computacional. Su elección respondió a que el problema abordado en el prototipo consistía en **clasificar imágenes de puntos de interés o elementos visuales previamente definidos**, y no en detectar múltiples objetos simultáneamente dentro de una escena compleja.

En este sentido, se consideró que una arquitectura de **clasificación** resultaba suficiente para el objetivo planteado. Alternativas como **YOLO (You Only Look Once)**, orientadas a **la detección de objetos**, ofrecen la capacidad de localizar y clasificar varios objetos en una misma imagen, pero implican una mayor complejidad de entrenamiento, anotación y procesamiento. Dado que el prototipo no requería delimitar objetos mediante bounding boxes ni identificar múltiples instancias en una misma captura, el uso de un modelo de detección habría introducido un costo computacional y de implementación innecesario para esta etapa del desarrollo.

Por lo tanto, MobileNetV2 resultó ser una elección adecuada al permitir:

- Un entrenamiento más accesible con datasets de clasificación.
- Una inferencia más liviana para no saturar el servidor.
- Tiempos de respuesta compatibles con una experiencia interactiva.
- Una integración más simple con el flujo general de la aplicación.

4.3.2. Tecnologías backend

Para el backend se utilizó **Django** junto con **Django REST Framework (DRF)**. Django fue seleccionado por tratarse de un framework maduro, robusto y ampliamente adoptado para el desarrollo de aplicaciones web con Python, especialmente adecuado cuando se requiere una estructura clara, administración integrada y rapidez de desarrollo. DRF complementó esta elección al facilitar la construcción de una API RESTful, necesaria para exponer datos de usuarios, desafíos, rutas, progreso y recompensas hacia la aplicación móvil.

Además, esta decisión permitió mantener coherencia tecnológica con el componente de inteligencia artificial, ya que tanto el backend como el entrenamiento de modelos se desarrollaron en el ecosistema Python. Esto simplificó la integración futura entre servicios de inferencia, lógica de negocio y persistencia de datos.

4.3.3. Tecnologías frontend móvil

En el desarrollo de la aplicación móvil, se optó por **React Native** con **Expo**. React Native permitió construir una única base de código capaz de

ejecutarse en múltiples plataformas móviles, reduciendo los tiempos de desarrollo y mantenimiento. Por su parte, Expo facilitó el acceso a capacidades nativas del dispositivo, como cámara, geolocalización y sensores, mediante librerías ya integradas. Además, simplificó el ciclo de pruebas y despliegue durante la etapa de prototipado.

Esta elección fue especialmente relevante en un proyecto como *Entre Diagonales*, donde la experiencia del usuario depende de la interacción con funcionalidades móviles específicas, tales como la captura de imágenes, la ubicación en tiempo real y la navegación contextual.

4.3.4. Infraestructura y despliegue

Finalmente, para el despliegue del backend se utilizó **Railway**. Esta plataforma fue elegida por su simplicidad operativa, su integración directa con repositorios de código y la posibilidad de realizar despliegues rápidos sin requerir una configuración compleja de infraestructura. Dado que el objetivo del proyecto era validar un prototipo funcional y no desplegar una arquitectura productiva de gran escala, Railway ofreció un equilibrio adecuado entre facilidad de uso, disponibilidad y soporte para escalado básico.

En conjunto, las tecnologías seleccionadas permitieron construir una solución coherente con los objetivos del proyecto, priorizando la viabilidad técnica, la rapidez de implementación y la compatibilidad entre sus distintos componentes.

4.4. Flujo de interacción entre los componentes

El flujo general de interacción del sistema se diseñó para minimizar la latencia y asegurar la validez de las acciones del usuario:

1. El usuario captura una imagen de un edificio u objeto desde la aplicación móvil.
2. La aplicación envía al servidor la imagen junto con el identificador, el tipo de objeto a validar y las coordenadas actuales del usuario.
3. El servidor verifica que el usuario se encuentre dentro del rango mínimo permitido (200 metros a la redonda) respecto al objeto.
4. En caso afirmativo, la imagen es procesada por el modelo de reconocimiento visual correspondiente.
5. Si la predicción supera el umbral de confianza definido, el servidor registra la interacción y devuelve una respuesta válida a la aplicación.

4.5. Alcance y limitaciones de la primera versión

La primera versión de *Entre Diagonales* fue concebida como un **prototipo funcional** orientado a validar la viabilidad técnica y conceptual de una aplicación turística gamificada para la ciudad de La Plata. En esta etapa inicial, el sistema permitió implementar y poner a prueba los componentes centrales de la propuesta, sin buscar todavía una cobertura completa de todos los puntos de interés ni un nivel de precisión equiparable al de un producto final en producción.

En términos de alcance, el prototipo incorporó un **módulo de reconocimiento visual** capaz de identificar un conjunto acotado de edificios y objetos urbanos previamente seleccionados. Esta funcionalidad permitió verificar que era posible integrar un modelo de clasificación de imágenes dentro del flujo general de la aplicación y utilizar sus resultados como disparadores de experiencias turísticas interactivas.

Es importante destacar que el sistema no fue diseñado para reconocer de manera abierta cualquier edificio o elemento urbano de la ciudad, sino para clasificar un conjunto previamente definido de puntos de interés incluidos en el dataset de entrenamiento. En consecuencia, el reconocimiento depende directamente de las clases contempladas durante la etapa de entrenamiento y no constituye, en esta primera versión, un sistema de identificación general de entornos urbanos.

Asimismo, se implementó un **sistema básico de gamificación**, incluyendo la asignación de puntos, el desbloqueo de desafíos y la lógica de progreso del usuario. Estos elementos permitieron validar la dinámica principal de interacción propuesta por el proyecto, en la cual el reconocimiento de lugares y la exploración de la ciudad se combinan con mecánicas lúdicas orientadas a incentivar la participación y el descubrimiento.

4.6. Construcción de un modelo de reconocimiento visual confiable

4.6.1. Naturaleza del problema y criterio de selección de arquitecturas

El problema abordado en este trabajo corresponde a una tarea de **clasificación de imágenes**. En este tipo de problemas, el objetivo del modelo es determinar a qué categoría pertenece una imagen completa, asignándola a una de las clases previamente definidas durante el entrenamiento. En el contexto de *Entre Diagonales*, esto implica que, a partir de una fotografía capturada por el usuario, el sistema debe identificar si la imagen corresponde a un punto de interés específico, por ejemplo, un edificio emblemático, una estatua o un objeto urbano relevante, siempre que dicho elemento forme

parte del conjunto de clases contempladas en el dataset.

Es importante destacar que esta formulación difiere de una tarea de **detección de objetos**. Mientras que en clasificación se analiza la imagen de manera global para determinar su categoría principal, en detección el modelo debe además localizar uno o varios objetos dentro de la escena, generalmente mediante regiones o *bounding boxes*. Modelos como **YOLO (You Only Look Once)** pertenecen a esta segunda categoría, ya que resuelven simultáneamente la localización y la clasificación de múltiples objetos.

Para el caso de este prototipo, se consideró que una estrategia de clasificación era suficiente para los objetivos planteados. La aplicación no requería, en esta primera versión, identificar múltiples elementos en una misma escena ni señalar visualmente su ubicación exacta dentro de la imagen. En cambio, el objetivo principal era reconocer si la captura realizada por el usuario correspondía a alguno de los puntos de interés previamente definidos para activar contenido, desafíos o recompensas dentro de la experiencia gamificada. Por este motivo, el uso de modelos de detección hubiera incorporado una complejidad adicional en términos de anotación de datos, entrenamiento e inferencia, sin aportar un beneficio proporcional para el alcance funcional del prototipo.

A partir de esta definición, la comparación de modelos se centró en **arquitecturas diseñadas para la clasificación de imágenes**. Se seleccionaron **MobileNetV2**, **ConvNeXt Tiny**, **EfficientNetB3** y **ResNeSt50** por representar enfoques distintos dentro del diseño de redes convolucionales y permitir una evaluación comparativa equilibrada entre precisión, costo computacional y factibilidad de integración.

En particular, **MobileNetV2** fue considerada por su orientación a entornos con recursos limitados y su eficiencia; **EfficientNetB3** por su buen desempeño en clasificación con una estrategia de escalado balanceado entre profundidad, ancho y resolución; **ConvNeXt Tiny** por representar una arquitectura más moderna, optimizada a partir de principios contemporáneos de diseño para visión por computadora; y **ResNeSt50** por aportar una variante robusta de redes residuales con mecanismos de atención por canales, útil como referencia de un modelo más expresivo.

De esta manera, la selección no buscó cubrir exhaustivamente todas las arquitecturas existentes, sino comparar un conjunto representativo de modelos con perfiles diferentes: algunos orientados a eficiencia, otros a mayor capacidad de representación y otros a enfoques más recientes. Esto permitió evaluar cuál ofrecía el mejor equilibrio entre desempeño y viabilidad técnica para las necesidades concretas del prototipo.

4.6.2. Selección del modelo base

Como mencionamos anteriormente se valoraron cuatro arquitecturas: **MobileNetV2**, **ConvNeXt Tiny**, **EfficientNetB3** y **ResNeSt50**. Todas ellas fueron

entrenadas bajo las mismas condiciones y utilizando idénticos conjuntos de datos, con el objetivo de realizar una comparación justa y controlada de su rendimiento.

4.6.3. Construcción y etiquetado del dataset

Se planificó una sesión de captura fotográfica destinada a recolectar imágenes reales de los ítems seleccionados. Para cada objeto se capturaron aproximadamente 130 imágenes, contemplando distintas perspectivas, escalas, condiciones de iluminación y contextos ambientales, con el objetivo de incorporar variabilidad visual y mejorar la capacidad de generalización del modelo.

El proceso de etiquetado consistió en la asignación manual de una clase a cada imagen, correspondiente al objeto o punto de interés representado. De esta manera, se construyó un **dataset supervisado**, en el cual cada ejemplo de entrenamiento cuenta con su categoría asociada, condición necesaria para el entrenamiento de modelos de clasificación. Además, las clases definidas en el dataset no solo cumplieron un rol en el entrenamiento supervisado, sino que también funcionaron como identificadores lógicos dentro de la aplicación. Por ejemplo, una etiqueta como *catedral* podía ser utilizada posteriormente para vincular la predicción del modelo con el punto de interés esperado por el sistema y, de este modo, validar si el usuario había identificado correctamente el objeto requerido para completar un desafío u objetivo. Este procedimiento fue realizado de forma manual, garantizando la consistencia entre clases y reduciendo la introducción de ruido en los datos.

Del total de imágenes obtenidas, el 92 % fue destinado al conjunto de entrenamiento, mientras que el 8 % restante se reservó para el conjunto de prueba.

Adicionalmente, con el fin de mitigar el sobreajuste y mejorar la capacidad de generalización de los modelos, se aplicaron técnicas de *data augmentation* sobre las imágenes del conjunto de entrenamiento. Entre las transformaciones utilizadas se incluyeron rotaciones aleatorias, ajustes de brillo y contraste, zoom, *flips* horizontales y recortes adaptativos. Estas operaciones permiten simular variaciones frecuentes en las condiciones reales de captura, como cambios de iluminación, leves diferencias de perspectiva o encuadres parciales del objeto, favoreciendo así un aprendizaje más robusto frente a escenarios no idénticos a los observados durante la recolección inicial.

4.6.4. Metodología de evaluación

Cada modelo fue entrenado utilizando técnicas de *transfer learning*, que consisten en reutilizar modelos previamente entrenados sobre grandes conjuntos de datos generales y adaptarlos a un problema específico. Esta estrategia permite aprovechar características visuales ya aprendidas en etapas

previas como bordes, formas y patrones básicos y ajustarlas al conjunto de clases definido para este trabajo, reduciendo el tiempo de entrenamiento y mejorando la eficiencia del proceso.

Con el objetivo de realizar una comparación justa entre arquitecturas, todas fueron entrenadas bajo las mismas condiciones, manteniendo la misma cantidad de épocas, iteraciones y configuraciones generales de entrenamiento. De esta manera, las diferencias observadas en el rendimiento pudieron atribuirse principalmente a las características propias de cada modelo y no a variaciones en el procedimiento experimental.

4.6.5. Entrenamiento, validación y métricas

El entrenamiento se realizó durante 15 épocas, donde cada época corresponde a una iteración completa sobre el conjunto de datos de entrenamiento. Para monitorear el aprendizaje del modelo y detectar posibles problemas de sobreajuste, se utilizó una división interna de 80 % de los datos para entrenamiento y 20 % para validación.

Durante este proceso se analizaron distintas métricas. La *precisión* (*accuracy*) se utilizó para medir la proporción de predicciones correctas realizadas por el modelo sobre el total de ejemplos evaluados. La *pérdida* (*loss*) permitió cuantificar el error cometido durante el entrenamiento, indicando en qué medida las predicciones se alejaban de las etiquetas reales. Finalmente, se emplearon **matrices de confusión** para observar con mayor detalle el comportamiento del clasificador, identificando no solo cuántos aciertos se obtenían, sino también entre qué clases se producían las principales confusiones.

Comparación de métricas de entrenamiento

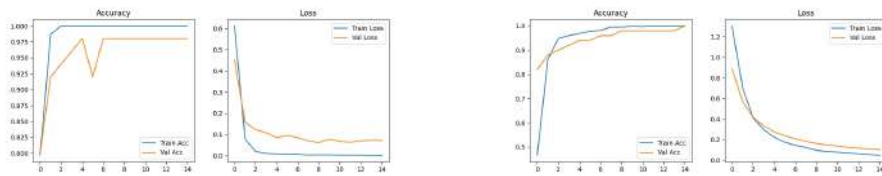


Figura 4.2: Comparación de *accuracy* y *loss* durante el entrenamiento de MobileNetV2 (izquierda) y ConvNeXt Tiny (derecha).

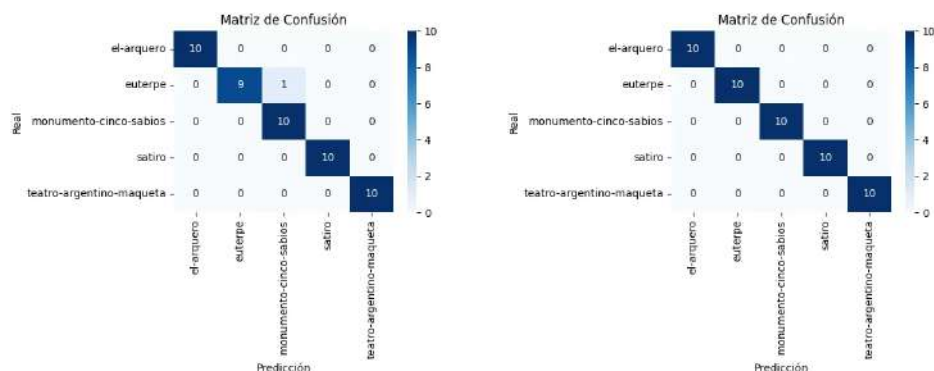


Figura 4.3: Comparación de las matrices de confusión de MobileNetV2 (izquierda) y ConvNeXt Tiny (derecha).

Si bien ConvNeXt Tiny alcanzó una exactitud del 100 %, su tamaño final (aproximadamente 112 MB) y sus mayores tiempos de inferencia lo volvieron poco prácticos para su integración en una aplicación móvil en tiempo real. En cambio, MobileNetV2 obtuvo una exactitud del 98 %, con un tamaño cercano a 9 MB, menor latencia y un comportamiento estable entre entrenamiento y validación, lo que lo convirtió en la alternativa más adecuada para esta primera versión del sistema.

4.6.6. Umbral de confianza y manejo de errores

Se definió un umbral de confianza del 60 % a partir de pruebas empíricas realizadas durante la validación del prototipo. Este valor representó un equilibrio adecuado entre sensibilidad y precisión: permitió aceptar predicciones suficientemente confiables sin volver excesivamente restrictiva la interacción para el usuario. Las predicciones que no superan dicho valor requieren que el usuario repita la captura, reduciendo la probabilidad de falsos positivos dentro del sistema de gamificación.

4.6.7. Integración del modelo en el flujo de la aplicación

Como vemos en la Figura 4.4, la integración del modelo de reconocimiento visual se realizó a través de la API desarrollada en Django, que permite centralizar la validación lógica y mantener el control del sistema de gamificación desde el *servidor*.

Desde la aplicación móvil, el usuario captura una imagen del monumento u objeto secreto mediante la cámara del dispositivo. Junto con la imagen se envían a la API el identificador del elemento esperado, su tipo (punto del recorrido u objeto secreto) y las coordenadas actuales. Esta información es recibida por un endpoint específico del servidor, encargado de procesar la solicitud.

En primer lugar, el servidor valida la autenticación del usuario y verifica su ubicación geográfica, comprobando que se encuentre dentro del radio permitido respecto al punto correspondiente. Esta verificación evita el uso indebido del sistema y reduce ejecuciones innecesarias del modelo de reconocimiento visual.

Una vez superada la validación de ubicación, el servidor selecciona dinámicamente el modelo de clasificación correspondiente según el tipo de objeto recibido. A continuación, la imagen es preprocesada y enviada al modelo de reconocimiento visual para ejecutar la inferencia.

El resultado obtenido incluye la clase predicha y el nivel de confianza asociado. En caso de que la confianza supere el umbral definido (60 %), el reconocimiento se considera válido y el servidor registra el evento en la base de datos, actualizando el progreso del usuario dentro del sistema de gamificación. Si la confianza es inferior, la API devuelve una respuesta negativa, solicitando al usuario repetir la captura.

Finalmente, el *servidor* retorna a la aplicación móvil el estado de la validación, permitiendo mostrar al usuario si el reconocimiento fue exitoso o no. Este enfoque centralizado garantiza consistencia en las reglas del sistema, seguridad en la validación y trazabilidad de las interacciones realizadas.

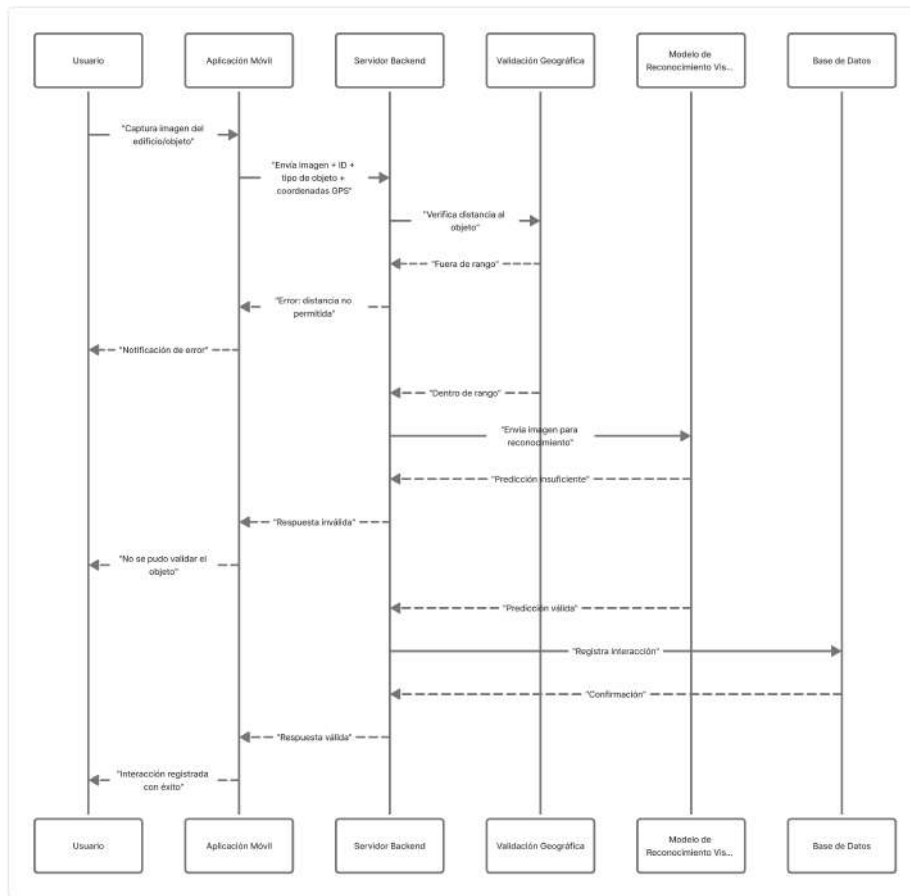


Figura 4.4: Flujo principal del sistema. Interacción entre componentes

4.6.8. Modelos descartados

EfficientNetB3 y ResNeSt50 presentaron menor estabilidad y dificultades de generalización. Sus métricas completas se incluyen en el Anexo 1.

4.6.9. Decisión final del modelo

El proceso de evaluación permitió identificar con claridad las arquitecturas más adecuadas para el contexto real de *Entre Diagonales*. ConvNeXt Tiny y MobileNetV2 demostraron un desempeño sobresaliente incluso con un dataset limitado. Sin embargo, considerando las restricciones propias de los dispositivos móviles, se priorizaron la eficiencia, el tamaño del modelo y la velocidad de inferencia. En este contexto, se aceptó resignar aproximadamente un 2 % de exactitud y se seleccionó MobileNetV2 como modelo principal.

4.7. Desarrollo del servidor

4.7.1. Arquitectura y diseño del servidor

El backend fue diseñado como un servicio central encargado de administrar usuarios, recorridos, puntos del recorrido, objetos secretos y el sistema de gamificación, incluyendo logros, ranking, trivia y niveles. Para el almacenamiento persistente de la información se utilizó PostgreSQL como sistema gestor de base de datos, con un esquema compuesto por aproximadamente 20 tablas principales. Este diseño permitió estructurar de forma modular los distintos dominios funcionales de la aplicación y facilitar su escalabilidad. El esquema lógico completo se presenta en el Anexo 2.

4.7.2. API, autenticación y seguridad

La API expone endpoints organizados por dominio funcional, entre ellos `/auth`, `/profile`, `/tours`, `/spots`, `/achievements`, `/secret_items`, `/ranking` y `/quizzes`. La documentación detallada se incluye en el Anexo 3.

Para la autenticación se implementó JWT (*JSON Web Token*), un mecanismo basado en tokens firmados digitalmente que permite identificar y validar usuarios de forma segura sin mantener sesiones persistentes en el servidor. Se configuró una expiración de 30 días, permitiendo que los usuarios se registren tanto mediante correo electrónico y contraseña como a través de su cuenta de Google.

En cuanto a la seguridad, se aplicó validación de tokens en los endpoints sensibles y controles adicionales sobre la lógica del juego, como la verificación de distancia previa a la ejecución del modelo de reconocimiento. Esta última medida permitió reducir intentos de validación fuera de contexto y reforzar la coherencia entre la ubicación física del usuario y los objetivos del recorrido.

4.7.3. Integración con el reconocimiento visual

El servidor actuó como intermediario entre la aplicación móvil y los modelos de reconocimiento visual, centralizando las validaciones y el registro de resultados. Esta arquitectura permitió desacoplar la lógica de inferencia de la interfaz cliente, unificar el tratamiento de errores y controlar de forma consistente la validación de objetivos dentro del sistema de gamificación.

4.7.4. Despliegue y monitoreo

El servidor fue desplegado utilizando Railway, integrando un flujo de despliegue continuo desde GitHub. Adicionalmente, se implementó un sistema de registro de logs para capturar errores y eventos relevantes durante la ejecución. Si bien estas herramientas resultaron suficientes para el prototipo

funcional, la incorporación de monitoreo avanzado y alertas automáticas se plantea como una línea de mejora para futuras versiones.

4.8. Implementación de la aplicación móvil

En este capítulo se describe el proceso de desarrollo de la aplicación móvil *Entre Diagonales*, detallando las decisiones técnicas adoptadas, la arquitectura interna del proyecto, la integración con servicios externos y las optimizaciones implementadas para mejorar la experiencia del usuario.

4.9. Selección del framework

Para el desarrollo de la aplicación móvil se optó por utilizar el framework **Expo**, basado en React Native. Esta elección se fundamentó tanto en el auge actual de las aplicaciones desarrolladas con Expo como en su madurez tecnológica y su amplio ecosistema de herramientas.

Otro factor determinante fue la experiencia previa que ambos desarrolladores tenían con **React**, lo que permitió acelerar el desarrollo y mantener un alto grado de productividad. Expo proporciona un entorno de desarrollo cómodo y completo, con integración directa de múltiples funcionalidades nativas sin necesidad de configuración compleja.

Entre las principales bibliotecas utilizadas se destacan:

- `expo-camera`, para el uso de la cámara y el reconocimiento visual.
- `expo-maps`, para la visualización de mapas y recorridos.
- `expo-notifications`, para el envío de notificaciones al usuario.
- `expo-updates`, que permite realizar actualizaciones OTA (Over-The-Air).
- `expo-haptics`, para la incorporación de efectos hápticos.
- `expo-audio`, para la reproducción de sonidos.
- `expo-location`, para la obtención de la ubicación del usuario.
- `@react-native-google-signin/google-signin`, para la autenticación mediante Google.

La combinación de estas herramientas permitió desarrollar una aplicación robusta, multiplataforma y fácilmente escalable.

4.10. Arquitectura interna

La arquitectura interna de la aplicación se organizó siguiendo un enfoque modular, con el objetivo de facilitar el mantenimiento, la escalabilidad y la reutilización de código. La estructura principal se divide en los siguientes componentes:

- **Pantallas**, que representan las distintas vistas de la aplicación.
- **Servicios**, encargados de la comunicación con la API y la gestión de datos externos.
- **Contexto global**, utilizado para manejar el estado compartido de la aplicación, como la información del usuario, el progreso y la autenticación.
- **Componentes reutilizables**, que encapsulan elementos visuales y funcionales comunes, promoviendo consistencia en la interfaz.

Esta separación de responsabilidades permitió mantener un código más claro, ordenado y fácil de extender.

4.11. Flujo de navegación

El flujo de navegación de la aplicación se diseñó para ser simple e intuitivo, priorizando el acceso rápido a las funcionalidades principales. La estructura de navegación incluye las siguientes pantallas principales:

- **Inicio**, donde el usuario visualiza su progreso general y los recorridos recomendados. Figura 3.1
- **Rutas**, que presentan el listado completo de recorridos disponibles. Figura 3.2
- **Escáner**, destinado al uso de la cámara para el reconocimiento visual y futuras funcionalidades. 3.5
- **Ranking**, donde el usuario podrá ver la tabla de clasificación global o de su nivel actual. 3.18
- **Perfil**, donde se muestra la información del usuario, sus logros y configuraciones. 3.8

Este esquema permitió una experiencia de uso clara y coherente, alineada con los objetivos de la aplicación.

4.12. Integración con el servidor

Todas las operaciones que requieren persistencia de datos o lógica de negocio se realizan a través de una API remota. La aplicación consume estos servicios mediante solicitudes HTTPS autenticadas, garantizando la seguridad de la información y el correcto acceso a los recursos.

Entre las operaciones principales se incluyen la obtención de recorridos, paradas, trivias, logros, progreso del usuario y el registro de acciones relevantes dentro del sistema de gamificación.

4.13. Potenciando la experiencia con efectos hápticos y sonoros

Con el objetivo de mejorar la experiencia de uso y reforzar la respuesta emocional del jugador, se incorporaron efectos hápticos y sonoros en acciones clave de la aplicación. Entre los eventos que generan este tipo de feedback se encuentran:

- Vibraciones al interactuar con ciertos botones.
- Sonidos diferenciados al responder correctamente o incorrectamente una trivia.
- Efectos sonoros al descubrir un objeto secreto.
- Sonidos de cierre y recompensa al completar un recorrido.

Los recursos sonoros utilizados fueron obtenidos de la plataforma *Mixkit* [45], bajo licencia gratuita. Todas estas opciones pueden ser activadas o desactivadas por el usuario desde la configuración de la aplicación, garantizando una experiencia adaptable a diferentes preferencias.

4.14. Cálculo de rutas y tiempos

Inicialmente, el cálculo de distancias y tiempos se realizaba exclusivamente mediante la API de Google Maps. Sin embargo, con el objetivo de reducir costos y optimizar el rendimiento, se implementó una solución híbrida.

Para el cálculo de distancias aproximadas entre dos puntos geográficos, se utilizó una implementación de la fórmula de *Haversine* [46] adaptada a la lógica del desarrollo de la aplicación. En primer lugar, las coordenadas de latitud y longitud, expresadas en grados, se convierten a radianes, dado que las funciones trigonométricas utilizadas operan sobre dicho sistema.

A partir de esta conversión, se calculan las diferencias de latitud y longitud entre ambos puntos. Estas diferencias se emplean para obtener el valor

intermedio h , que representa la distancia angular entre los puntos sobre la superficie esférica de la Tierra. Dicho valor se calcula mediante la siguiente expresión:

$$h = \sin^2\left(\frac{\Delta\varphi}{2}\right) + \cos(\varphi_1) \cdot \cos(\varphi_2) \cdot \sin^2\left(\frac{\Delta\lambda}{2}\right) \quad (4.1)$$

Luego, se obtiene el ángulo central c utilizando la función arco tangente, lo que mejora la estabilidad numérica del cálculo:

$$c = 2 \cdot \arctan 2\left(\sqrt{h}, \sqrt{1-h}\right) \quad (4.2)$$

Finalmente, la distancia lineal entre los puntos se determina multiplicando este ángulo por el radio medio de la Tierra $R = 6.371.000$ metros:

$$d = R \cdot c \quad (4.3)$$

Este valor corresponde a la distancia aproximada en línea recta entre dos ubicaciones geográficas cercanas y resulta adecuado para el contexto urbano de la ciudad de La Plata, donde las distancias entre paradas consecutivas son relativamente cortas.

Para el cálculo del tiempo estimado de desplazamiento a pie se asumió una velocidad promedio constante de caminata de 4,8 km/h, un valor comúnmente utilizado como referencia para desplazamientos urbanos. A partir de esta velocidad, se derivó una equivalencia de aproximadamente 80 metros por minuto, la cual se emplea para convertir la distancia calculada en metros a una duración estimada en minutos.

Por otro lado, para el trazado visual de las rutas en el mapa y el cálculo detallado de los trayectos se empleó la API de *OpenRouteService* (ORS), la cual se basa en datos geospaciales provistos por *OpenStreetMap* (OSM). Con el fin de optimizar el rendimiento de la aplicación y reducir el consumo de recursos externos, las solicitudes a este servicio se realizan únicamente en situaciones específicas: cuando el usuario cambia de recorrido, cuando se actualiza la parada objetivo dentro del recorrido o cuando el usuario se desplaza una distancia mínima de 100 metros respecto a la última ubicación registrada. Este enfoque permite mantener un equilibrio entre precisión en el ruteo, eficiencia en el uso de la API y sostenibilidad económica del proyecto.

4.14.1. Sistema de gamificación

El sistema de gamificación de *Entre Diagonales* fue implementado como un conjunto de reglas y mecanismos que permiten registrar, calcular y persistir el progreso del usuario dentro de la aplicación. Este sistema integra puntos de sabiduría (experiencia), logros, desafíos y el avance en los recorridos, funcionando de manera transversal a las distintas funcionalidades.

Desde el punto de vista técnico, cada acción relevante (como completar una parada, responder una trivia, descubrir un objeto secreto o desbloquear un logro) genera un evento que actualiza el estado del usuario tanto a nivel local como en el servidor. De esta manera, se garantiza la consistencia del progreso, incluso ante cierres inesperados de la aplicación o cambios de dispositivo.

La implementación modular del sistema de gamificación permitió desacoplar la lógica de progreso de las pantallas visuales, facilitando su mantenimiento y la incorporación futura de nuevas mecánicas sin afectar la estructura general de la aplicación.

4.14.2. Mapa interactivo

Para la visualización de los recorridos y la orientación del usuario se integró la biblioteca **expo-maps**, que permitió representar el mapa de la ciudad junto con distintos elementos interactivos. Sobre este mapa se incorporaron marcadores dinámicos correspondientes a las paradas del recorrido, la ubicación actual del usuario y el punto objetivo a alcanzar.

El mapa se actualiza en tiempo real según la posición del usuario, permitiendo recalcular distancias y ofrecer una referencia visual clara del trayecto a seguir. Asimismo, se trazan rutas mediante polilíneas, combinando cálculos locales de distancia con servicios externos de ruteo, lo que contribuye a una experiencia de navegación fluida y precisa.

Esta integración resulta fundamental para reforzar la relación entre el espacio físico y la experiencia digital propuesta por la aplicación.

4.14.3. Escáner visual y validación

El escáner visual constituye uno de los componentes técnicos más relevantes de *Entre Diagonales*. A través del uso de la cámara del dispositivo, la aplicación captura imágenes del entorno que luego son procesadas por un modelo de reconocimiento visual previamente entrenado.

El flujo de funcionamiento consiste en la captura de la imagen, la inferencia de la clase correspondiente por parte del modelo y la validación del resultado mediante criterios definidos en el servidor. Este proceso permite confirmar si el objeto identificado coincide con el esperado para una parada u objeto secreto determinado.

La validación del lado del servidor añade una capa adicional de seguridad y control, evitando falsos positivos y garantizando que los avances registrados en el sistema de gamificación sean legítimos. Esta arquitectura híbrida combina procesamiento local con verificación remota, logrando un equilibrio entre rendimiento y confiabilidad.

4.14.4. Notificaciones push

Las notificaciones push se incorporaron como un mecanismo para reforzar la comunicación con el usuario y fomentar la continuidad en el uso de la aplicación. A través de **expo-notifications**, se implementó un sistema que permite enviar mensajes informativos y recordatorios directamente al dispositivo móvil.

Entre los principales usos de las notificaciones se incluyen avisos de bienvenida, mensajes motivacionales, recordatorios para continuar recorridos incompletos y notificaciones relacionadas con eventos relevantes dentro de la experiencia de juego. Este sistema contribuye a mejorar la retención del usuario y a mantener un vínculo activo con la aplicación, incluso cuando no se encuentra en uso.

El diseño de las notificaciones prioriza mensajes breves, claros y coherentes con la identidad visual y narrativa de *Entre Diagonales* como vemos en la Figura 4.5.

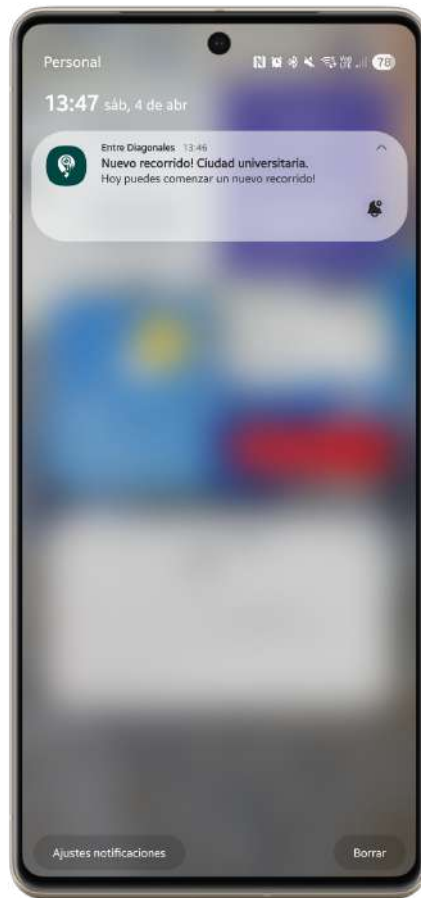


Figura 4.5: Visual de notificación recibida al teléfono móvil.

4.14.5. Accesibilidad y personalización de la interfaz

Como parte del diseño centrado en el usuario, se incorporaron mecanismos básicos de accesibilidad destinados a mejorar la legibilidad y la adaptabilidad de la interfaz (Figura 4.6). Entre ellos, se incluyó la posibilidad de modificar el tamaño de los textos y de determinados elementos visuales de la aplicación, permitiendo que cada usuario ajuste la presentación de la información según sus necesidades o preferencias.

Esta decisión de diseño adquiere especial relevancia en un contexto de uso móvil y en exteriores, donde factores como la iluminación, la distancia de observación, el movimiento durante el recorrido o las diferencias en la capacidad visual pueden afectar la experiencia de uso. En consecuencia, la incorporación de opciones de personalización visual no solo mejora la comodidad de interacción, sino que también refuerza criterios básicos de

accesibilidad dentro del prototipo desarrollado.



Figura 4.6: Opciones de accesibilidad y personalización visual de la interfaz, incluyendo el ajuste del tamaño de texto.

Capítulo 5. Desafíos Tecnológicos

El desarrollo de *Entre Diagonales* implicó enfrentar diversos desafíos tecnológicos derivados tanto de las características del entorno urbano como de los requerimientos propios de una aplicación móvil gamificada. En este capítulo se describen los principales problemas identificados durante el desarrollo y las estrategias adoptadas para resolverlos, priorizando la precisión, el rendimiento y la experiencia del usuario.

5.1. Desafíos tecnológicos

5.1.1. Control de integridad por localización

Uno de los principales desafíos del proyecto fue garantizar que las acciones realizadas por el usuario, como completar una parada o descubrir un objeto secreto, se correspondieran efectivamente con su presencia física en el lugar indicado. Dado que la aplicación se basa en recorridos geolocalizados, era fundamental evitar que los usuarios pudieran avanzar sin desplazarse realmente por la ciudad.

Para abordar este problema, se implementó un control de integridad basado en la ubicación del dispositivo, verificando que el usuario se encontrara dentro de un radio de proximidad definido respecto al punto objetivo. Este control se complementó con el uso del reconocimiento visual, lo que permitió validar no solo la cercanía geográfica, sino también la presencia efectiva frente al objeto o monumento correspondiente. De esta manera, se redujo significativamente la posibilidad de avances fraudulentos y uso innecesario del modelo de reconocimiento visual dentro del sistema de gamificación.

5.1.2. Integración de servicios de rutas y mapas

La integración de servicios de mapas y cálculo de rutas representó otro desafío relevante. Inicialmente, el uso de APIs comerciales, como Google Directions u ORS ofrecía una solución directa, pero implicaba costos elevados y una dependencia constante de servicios externos.

Para resolver esta problemática, se adoptó una solución híbrida que combina cálculos locales de distancia mediante la fórmula de Haversine con el uso de servicios de ruteo open source para el trazado visual de las rutas. Además, se implementaron mecanismos de optimización que limitan las solicitudes a los servicios externos únicamente a eventos relevantes, como el cambio de recorrido, la actualización del punto objetivo o desplazamientos significativos (100 metros a la redonda) del usuario. Esta estrategia permitió equilibrar precisión, rendimiento y sostenibilidad económica.

5.1.3. Diseño de mecánicas sin repetición

Uno de los desafíos tecnológicos y de diseño más relevantes fue garantizar la integridad del sistema de gamificación, evitando que los usuarios pudieran explotar mecánicas repetibles para acumular puntos o recompensas sin haber adquirido el conocimiento correspondiente.

En particular, el sistema de trivias fue diseñado bajo el criterio de resolución única. Cada trivia puede ser respondida una sola vez por el usuario y su resultado queda registrado de forma persistente. En caso de que la respuesta sea incorrecta, la trivia no vuelve a otorgar puntos de experiencia ni monedas en intentos posteriores, aun cuando el usuario conozca la respuesta correcta en una interacción futura.

Este enfoque busca asegurar que las recompensas obtenidas reflejen el aprendizaje real del usuario en el momento de la interacción y no el resultado de la repetición o el ensayo y error. De esta manera, se evita la acumulación artificial de puntos basada en intentos reiterados y se refuerza el valor educativo de la mecánica.

Desde el punto de vista técnico, esta decisión implicó la necesidad de registrar el estado de cada trivia a nivel de usuario, permitiendo identificar si una pregunta ya fue respondida y cuál fue su resultado. Esta información se utiliza tanto para bloquear la obtención de recompensas adicionales como para mantener la coherencia del progreso general dentro del sistema de gamificación.

La implementación de mecánicas no repetibles contribuye a un sistema más justo, equilibrado y consistente, alineado con los objetivos de aprendizaje y exploración propuestos por *Entre Diagonales*.

5.1.4. Optimización de las imágenes

El uso intensivo de recursos gráficos dentro de la aplicación, incluyendo imágenes de recorridos, paradas, objetos secretos y elementos visuales de la interfaz, representó un desafío en términos de consumo de memoria, tiempos de carga y rendimiento general.

Para mitigar estos problemas, se realizó un proceso de optimización de imágenes mediante la conversión de archivos en formatos tradicionales como JPG y PNG al formato WebP, un formato moderno de compresión de imágenes que permite reducir el tamaño de los archivos manteniendo una buena calidad visual. Esta conversión permitió disminuir significativamente el peso de los recursos sin comprometer de manera perceptible su apariencia, mejorando así los tiempos de descarga y renderizado dentro de la aplicación.

El proceso de optimización se llevó a cabo utilizando el servicio web *Squoosh* [47], una herramienta especializada en la compresión y conversión de imágenes. Esta decisión resultó especialmente relevante dado que la aplicación está pensada para ser utilizada principalmente en exteriores y en

movimiento, donde es frecuente depender de conexiones móviles y planes de datos limitados. Gracias a esta optimización, se logró reducir el consumo de datos, acelerar la carga de contenidos y mejorar la experiencia de usuario, especialmente en dispositivos con recursos limitados o en condiciones de conectividad inestable.

5.1.5. Mejora del rendimiento y fluidez de la aplicación

Mantener un rendimiento estable y una experiencia fluida fue un objetivo central durante todo el desarrollo. La combinación de geolocalización, mapas, reconocimiento visual y gamificación implica una carga considerable sobre el dispositivo móvil, especialmente considerando que la aplicación fue pensada para ser utilizada en movimiento y, en muchos casos, bajo redes móviles.

Para abordar este desafío, se aplicaron estrategias de optimización orientadas tanto al procesamiento como al renderizado de la interfaz. En primer lugar, se buscó limitar renderizados innecesarios dentro de la aplicación, evitando que componentes visuales complejos se actualizaran cuando no existían cambios relevantes en los datos. Por ejemplo, se procuró que elementos como el mapa, la lista de paradas del recorrido o los indicadores visuales de progreso no se volvieran a renderizar ante cambios menores en estados no relacionados, como modificaciones en textos auxiliares, temporizadores o valores transitorios de la interfaz. Esta separación entre lógica y presentación permitió reducir el trabajo innecesario del motor de renderizado y mejorar la respuesta general de la aplicación.

Asimismo, se implementó un uso más eficiente del estado global, procurando almacenar únicamente la información verdaderamente compartida entre pantallas o componentes, mientras que los estados locales de interacción se mantuvieron aislados. Esto evitó propagaciones de cambios que podían generar actualizaciones en cascada sobre múltiples componentes.

Otro aspecto importante fue el control de la frecuencia de actualización de la ubicación del usuario. Si bien el GPS puede entregar nuevas coordenadas constantemente, estas mediciones suelen presentar pequeñas variaciones incluso cuando la persona permanece prácticamente quieta o se desplaza muy poco, fenómeno conocido como *ruido del GPS*. Si cada una de estas variaciones activara de inmediato un recálculo de distancia, una actualización del marcador en el mapa o una verificación de proximidad con la siguiente parada, se generarían operaciones redundantes que afectarían negativamente el rendimiento y el consumo de batería.

Por este motivo, se estableció una política de actualización basada en distancia mínima recorrida, procesando cambios significativos de ubicación únicamente cuando el usuario se desplaza al menos 50 metros respecto de la última posición considerada. Este criterio permitió filtrar variaciones irrelevantes producidas por el ruido del GPS, reducir la cantidad de cálculos

geográficos y evitar renderizados excesivos del mapa y de los elementos asociados al recorrido activo. Como resultado, se logró una navegación más estable, un comportamiento visual más fluido y un uso más eficiente de los recursos del dispositivo.

5.1.6. Fluidez en los efectos visuales

La incorporación de efectos visuales, como animaciones y confeti al completar logros o recorridos, aportó un valor significativo a la experiencia del usuario, pero también introdujo desafíos en términos de rendimiento gráfico.

Para garantizar una ejecución fluida, se optimizaron las animaciones mediante el uso de librerías especializadas dentro del ecosistema de React Native, particularmente `react-native-reanimated`, `react-native-worklets` y `typegpu-confetti`. Estas herramientas permitieron delegar parte de la ejecución de animaciones y efectos visuales a mecanismos más eficientes que los provistos por actualizaciones tradicionales de estado en JavaScript, reduciendo bloqueos en la interfaz y mejorando la suavidad de las transiciones.

Asimismo, los efectos fueron limitados a momentos específicos de alto valor dentro de la experiencia, como la validación de una parada, el descubrimiento de un objeto secreto o la finalización de un recorrido, evitando una sobrecarga visual o computacional innecesaria. De esta manera, se logró un equilibrio entre impacto estético y estabilidad general, asegurando que los efectos refuerzan la experiencia de gamificación sin afectar negativamente la usabilidad.

5.1.7. Adaptabilidad a distintas pantallas

Finalmente, otro desafío clave fue asegurar que la aplicación se adaptara correctamente a dispositivos con distintos tamaños de pantalla y resoluciones. Dado el amplio abanico de teléfonos móviles disponibles, era fundamental garantizar una experiencia visual consistente y una correcta legibilidad en todos los casos.

Para abordar este problema, se implementaron diseños responsivos y escalables, utilizando unidades relativas y ajustes dinámicos en los componentes visuales. Además, se incorporó el uso de factores de escala para los textos, calculados en función de las dimensiones de la pantalla del dispositivo. Este enfoque permitió ajustar automáticamente el tamaño de la tipografía según el ancho y la densidad de la pantalla, evitando textos excesivamente grandes o pequeños en determinados dispositivos.

La combinación de layouts flexibles y escalado dinámico de textos permitió que la aplicación mantuviera su estética, legibilidad y funcionalidad tanto en dispositivos de pantalla reducida como en equipos con pantallas de mayor tamaño, mejorando de forma significativa la experiencia de uso general.

Capítulo 6. Evaluación de Entre Diagonales

6.1. Introducción

El proceso de evaluación de *Entre Diagonales* tuvo como finalidad determinar el grado en que el prototipo cumple los objetivos planteados en este trabajo, tanto desde una perspectiva técnica como desde la experiencia de los usuarios. De acuerdo con lo establecido en la propuesta inicial del proyecto, se llevó a cabo una evaluación interna exhaustiva del funcionamiento, rendimiento y estabilidad de la aplicación, seguida de una instancia de pruebas con usuarios reales, con el fin de obtener retroalimentación sobre la usabilidad, la interacción con las rutas, la mecánica de reconocimiento visual y los elementos de gamificación.

6.2. Pruebas

6.2.1. Pruebas técnicas de la aplicación

Se realizaron pruebas internas para analizar el rendimiento general del sistema, incluyendo el comportamiento del modelo de reconocimiento visual, la precisión de la ubicación en el mapa, la carga de rutas y la estabilidad en dispositivos de gama media. Se midieron aspectos como tiempos de inferencia del modelo, latencia en la renderización del mapa, manejo de errores y consumo de memoria.

6.2.2. Pruebas funcionales

Además, se evaluaron los principales flujos de uso de la aplicación: inicio de un recorrido, detección de objetos secretos, avance entre paradas, visualización de pistas y canje de logros. Estas pruebas permitieron identificar problemas relacionados con la claridad de la interfaz, la consistencia de la navegación y la experiencia general del usuario, implementándose mejoras antes de las pruebas con usuarios.

6.3. Encuestas

6.3.1. Metodología

Para evaluar la percepción de los usuarios se realizaron encuestas y sesiones de prueba con participantes pertenecientes al público objetivo. Se registraron datos sobre su experiencia previa con aplicaciones turísticas, su

familiaridad con la ciudad y su interacción con el prototipo durante un recorrido guiado.

6.3.2. Planificación

A continuación se presenta el cuestionario utilizado para evaluar la experiencia de los usuarios durante el uso del prototipo de *Entre Diagonales*. Las preguntas se organizaron por categorías con el fin de obtener información relacionada con la usabilidad, el contenido, el reconocimiento visual y los elementos de gamificación.

1. Datos generales del participante

- ¿Conocías previamente la ciudad de La Plata? (Sí / No / Parcialmente)
- ¿Usás habitualmente aplicaciones de turismo? (Sí / No)

2. Experiencia general de uso

- ¿Qué tan fácil te resultó entender cómo funciona la aplicación? (1–5)
- ¿La aplicación te pareció intuitiva? (Sí / No / Parcialmente)
- ¿Disfrutaste la experiencia general de usar la aplicación? (1–5)
- ¿Volverías a utilizar la aplicación en otro recorrido? (Sí / No / Tal vez)

3. Interfaz y usabilidad

- ¿Qué tan clara te pareció la interfaz? (1–5)
- ¿Encontraste fácilmente las opciones principales (mapa, rutas, perfil)? (Sí / No / Con dificultad)
- ¿La lectura de textos e indicaciones resultó accesible? (Sí / No / Parcialmente)
- ¿La fluidez y rendimiento de la app fueron adecuados en tu dispositivo? (1–5)

4. Rutas y contenido cultural

- ¿Qué tan interesante te resultó el contenido cultural presentado? (1–5)
- ¿Las explicaciones sobre los lugares fueron claras y útiles? (Sí / No / Parcialmente)

- ¿El recorrido fue fácil de seguir con las indicaciones provistas? (Sí / No / Parcialmente)
- ¿Sentiste que aprendiste algo nuevo durante el recorrido? (Sí / No)

5. Experiencia con el mapa

- ¿Qué tan precisa fue la ubicación en el mapa durante el recorrido? (1–5)
- ¿Fue fácil interpretar las rutas y paradas en el mapa? (Sí / No / Parcialmente)
- ¿El tiempo de carga del mapa fue adecuado? (1–5)

6. Reconocimiento visual

- ¿El reconocimiento del objeto funcionó correctamente la mayoría de las veces? (Sí / No / A veces)
- ¿Te resultó fácil tomar la foto para completar la parada? (Sí / No)
- ¿Qué tan confiable te pareció el reconocimiento visual? (1–5)

7. Gamificación

- ¿Los desafíos te resultaron motivadores? (Sí / No / Parcialmente)
- ¿Los logros e insignias agregaron valor a la experiencia? (Sí / No)
- ¿Te gustaría que hubiera más tipos de desafíos? (Sí / No / No estoy seguro)
- ¿Consideras que el sistema de puntos y recompensas mejora la experiencia? (1–5)

8. Evaluación final

- ¿Qué fue lo que más te gustó de la aplicación? (Respuesta abierta)
- ¿Qué mejorarías de la aplicación? (Respuesta abierta)
- ¿Recomendarías esta aplicación a otras personas que visiten La Plata? (Sí / No / Tal vez)
- En general, ¿qué calificación le darías a la aplicación? (1–10)

6.4. Resultados

Las encuestas realizadas permitieron obtener información relevante sobre la experiencia de los usuarios al interactuar con el prototipo de la aplicación *Entre Diagonales*. En total participaron 12 personas, quienes utilizaron la aplicación y posteriormente respondieron el cuestionario presentado en la sección anterior. Los resultados obtenidos permiten analizar distintos aspectos del sistema, incluyendo la experiencia general de uso, la usabilidad de la interfaz, la comprensión del contenido cultural, el funcionamiento del mapa, el sistema de reconocimiento visual y los elementos de gamificación.

6.4.1. Perfil de los participantes

Los participantes de la encuesta presentan edades comprendidas entre los 19 y 45 años, lo que permitió obtener una muestra heterogénea en términos etarios.

En relación con el conocimiento previo de la ciudad de La Plata, como se ve en el Gráfico 6.1, el 91,7 % de los participantes indicó conocer previamente la ciudad, mientras que el 8,3 % manifestó conocerla parcialmente. Este resultado es relevante, ya que demuestra que incluso personas familiarizadas con la ciudad pueden encontrar valor en una experiencia turística que propone una exploración diferente del entorno urbano.

¿Conocías previamente la ciudad de La Plata?

12 respuestas

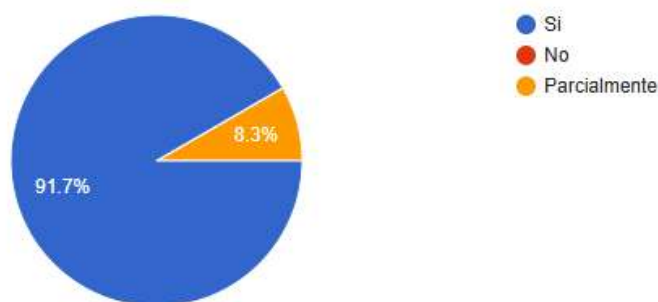


Figura 6.1: Conocimiento previo de las personas en la Ciudad de La Plata.

Por otra parte, al consultar si los usuarios utilizan habitualmente aplicaciones de turismo, vemos en el Gráfico 6.2 que solo el 33,3 % respondió afirmativamente, mientras que el 66,7 % indicó no utilizarlas. Este dato sugiere que la propuesta de la aplicación logra captar el interés incluso de personas que no suelen interactuar con herramientas digitales de turismo.

¿Usás habitualmente aplicaciones de turismo?

12 respuestas

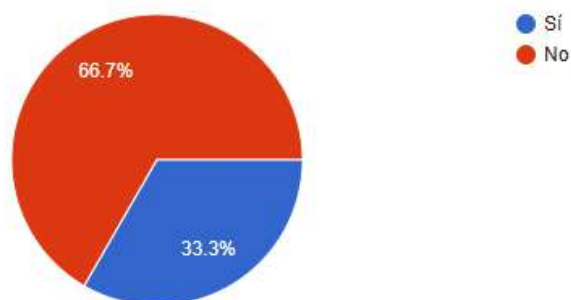


Figura 6.2: Uso habitual de aplicaciones turísticas por parte de los usuarios.

6.4.2. Experiencia general de uso

Los resultados obtenidos en relación con la experiencia general de uso fueron altamente positivos. Ante la pregunta sobre qué tan fácil resultó entender el funcionamiento de la aplicación, el 50 % de los participantes otorgó la máxima calificación (5), mientras que el 50 % restante asignó una calificación de 4. Esto indica que la mayoría de los usuarios pudo comprender rápidamente el funcionamiento del sistema.

Asimismo, el 100 % de los encuestados consideró que la aplicación resulta intuitiva. Este resultado sugiere que la estructura de navegación y la organización de las funcionalidades fueron adecuadamente diseñadas para facilitar la interacción de los usuarios.

En cuanto al disfrute general de la experiencia, el 58,3 % de los participantes calificó la experiencia con 5, mientras que el 41,7 % la calificó con 4. Además, el 100 % de los usuarios indicó que volvería a utilizar la aplicación en otro recorrido, lo cual evidencia un alto nivel de satisfacción con el prototipo.

6.4.3. Interfaz y usabilidad

La claridad de la interfaz también obtuvo valoraciones positivas. El 66,7 % de los participantes calificó la interfaz con 4 sobre 5, mientras que el 33,3 % restante le otorgó la máxima puntuación.

Además, el 100 % de los usuarios indicó haber encontrado fácilmente las opciones principales de la aplicación, como el mapa, las rutas y el perfil. Este resultado demuestra que la organización de los elementos de navegación es clara para los usuarios.

Respecto a la lectura de textos e indicaciones vemos en el Gráfico 6.3, que el 91,7 % consideró que resultaron accesibles, mientras que un 8,3 % indicó que fueron parcialmente accesibles. Esto sugiere que el contenido textual es comprensible en la mayoría de los casos, aunque podrían realizarse ajustes menores para mejorar su claridad.

¿La lectura de textos e indicaciones resultó accesible?

12 respuestas

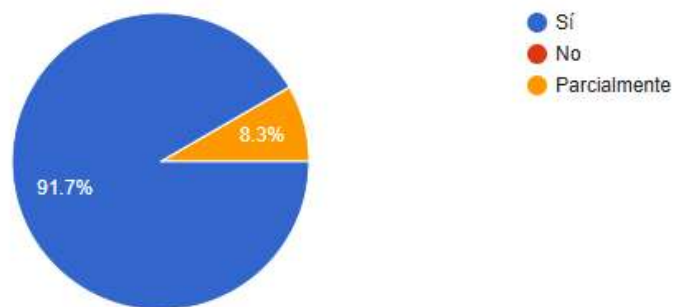


Figura 6.3: Resultados sobre la lectura de textos e indicaciones.

En relación con el rendimiento del sistema, el 50 % de los usuarios calificó la fluidez de la aplicación con 5, mientras que el 50 % restante la calificó con 4, lo que indica un buen desempeño en los dispositivos utilizados durante las pruebas.

6.4.4. Rutas y contenido cultural

El contenido cultural presentado durante los recorridos también recibió evaluaciones favorables. Observando el Gráfico 6.4 vemos que el 58,3 % de los participantes consideró que el contenido fue muy interesante (calificación 5), mientras que el 41,7 % lo calificó con 4.

¿Qué tan interesante te resultó el contenido cultural presentado?

12 respuestas



Figura 6.4: Resultados sobre el interés sobre el contenido.

Además, observamos en el Gráfico 6.5 el 91,7 % de los usuarios indicó que las explicaciones sobre los lugares fueron claras y útiles.

¿El recorrido fue fácil de seguir con las indicaciones provistas?

12 respuestas

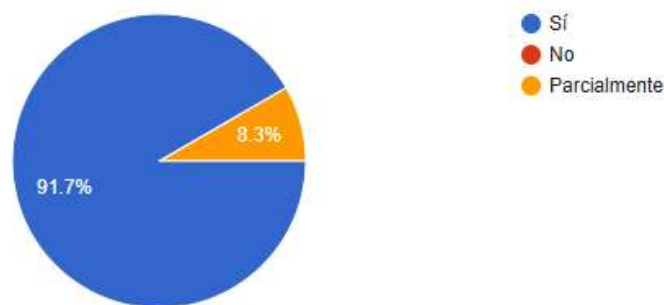


Figura 6.5: Resultados sobre el seguimiento de las indicaciones

Un resultado particularmente destacable es que el 100 % de los participantes manifestó haber aprendido algo nuevo durante el recorrido. Este dato evidencia el potencial educativo de la aplicación, combinando información cultural con una experiencia interactiva y lúdica.

6.4.5. Experiencia con el mapa

El sistema de geolocalización y visualización de rutas como vemos en el Gráfico 6.6 fue evaluado de forma positiva por los participantes. En relación con la precisión de la ubicación en el mapa, el 75 % de los usuarios otorgó una calificación de 4, mientras que dos participantes asignaron la puntuación máxima y uno calificó con 3.



Figura 6.6: Resultados sobre la precisión de la ubicación.

Asimismo, el 91,7 % de los encuestados indicó que las rutas y paradas fueron fáciles de interpretar en el mapa, lo que demuestra que la representación visual del recorrido resulta comprensible.

Respecto al tiempo de carga del mapa, el 66,7 % de los participantes lo calificó con 4, mientras que el 33,3 % le otorgó la máxima calificación. Como se puede observar en el Gráfico 6.7.

¿Qué tan interesante te resultó el contenido cultural presentado?

12 respuestas



Figura 6.7: Resultados sobre la carga del mapa.

6.4.6. Reconocimiento visual

Uno de los componentes más innovadores del sistema es el reconocimiento visual de objetos o monumentos durante el recorrido. En este aspecto, vemos en el Gráfico 6.8 que el 83,3 % de los participantes indicó que el reconocimiento funcionó correctamente la mayoría de las veces, mientras que el 16,7 % señaló que funcionó solo en algunas ocasiones.

¿El reconocimiento del objeto funcionó correctamente la mayoría de las veces?

12 respuestas

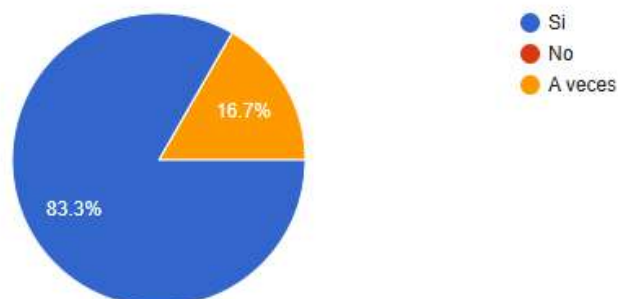


Figura 6.8: Resultados del funcionamiento del reconocimiento de objetos.

Además, el 100 % de los usuarios afirmó que fue fácil tomar la fotografía para completar la parada correspondiente.

En términos de confiabilidad, el 83,3 % calificó el reconocimiento visual con 4, mientras que el 16,7 % lo calificó con 5. Estos resultados indican un buen funcionamiento general del sistema, aunque existe margen para futuras mejoras en la precisión del modelo.

6.4.7. Gamificación

Los elementos de gamificación fueron uno de los aspectos mejor valorados por los usuarios. En el Gráfico 6.9 vemos que el 83,3 % de los participantes indicó que los desafíos resultaron motivadores, mientras que el 16,7 % señaló que lo fueron parcialmente.

¿Los desafíos te resultaron motivadores?

12 respuestas

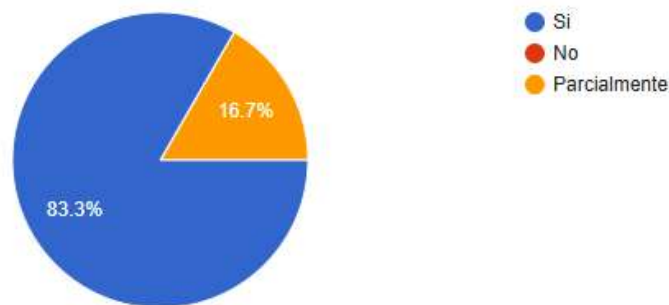


Figura 6.9: Resultados sobre la motivación de los desafíos.

Asimismo, el 100 % de los encuestados afirmó que los logros e insignias agregan valor a la experiencia de uso.

Otro resultado relevante es que todos los participantes manifestaron su interés en que la aplicación incorpore más tipos de desafíos, lo que sugiere que ampliar las mecánicas de juego podría incrementar aún más el nivel de participación y motivación de los usuarios.

En relación con el sistema de puntos y recompensas, las valoraciones se concentraron entre 4 y 5, evidenciando una percepción positiva de este mecanismo.

6.4.8. Evaluación final

Finalmente, la aplicación obtuvo una calificación promedio de 8,9 sobre 10 en la evaluación general realizada por los participantes como se puede observar en el Gráfico 6.10.

En general, ¿qué calificación le darías a la aplicación?

12 respuestas

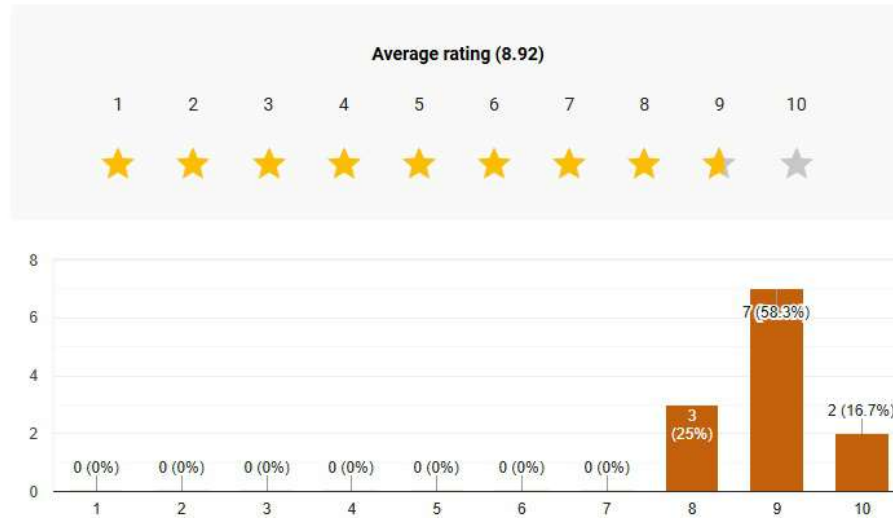


Figura 6.10: Resultados sobre la calificación de la aplicación.

Además, el 100 % de los encuestados indicó que recomendaría la aplicación a otras personas que visiten la ciudad de La Plata.

Las respuestas abiertas permitieron identificar algunos de los aspectos más valorados de la aplicación, entre los que se destacan la gamificación del recorrido, la búsqueda de objetos secretos, la exploración del mapa interactivo y la posibilidad de descubrir información histórica sobre distintos puntos de la ciudad.

Entre las sugerencias de mejora mencionadas por los participantes se encuentran la incorporación de más monumentos y contenido cultural, la inclusión de nuevos desafíos, mejoras en el reconocimiento visual y algunos ajustes en el mapa y la navegación. Estas observaciones resultan valiosas para futuras iteraciones del sistema.

Capítulo 7. Conclusiones y trabajos futuros

7.1. Conclusiones

A lo largo de esta tesina se propuso y desarrolló *Entre Diagonales*, un prototipo que nace de una convicción central: la tecnología no debe reemplazar la experiencia urbana, sino amplificarla. En un contexto donde muchas aplicaciones turísticas se limitan a digitalizar información o actuar como simples planificadores de viaje, esta tesina buscó transformar la ciudad de La Plata en un escenario interactivo, donde el patrimonio cultural se convierte en el eje de una experiencia activa, lúdica y significativa.

El análisis del mercado permitió identificar una brecha concreta: la ausencia de propuestas que integren identidad local, narrativa histórica, gamificación estructurada y reconocimiento visual mediante inteligencia artificial como núcleo de interacción. Frente a esta situación, *Entre Diagonales* no solo plantea una alternativa tecnológica, sino un cambio de paradigma: el usuario deja de ser un lector pasivo de datos y pasa a convertirse en explorador, jugador y protagonista de su propio recorrido urbano.

Desde el punto de vista técnico, el proyecto demostró la viabilidad de integrar una arquitectura con un modelo de clasificación visual entrenado específicamente para reconocer monumentos y objetos emblemáticos de la ciudad. La implementación de un servidor robusto, el despliegue del modelo en la nube y la integración con la aplicación móvil evidencian que es posible llevar la inteligencia artificial al terreno real, utilizándola no como un recurso accesorio, sino como el disparador central de la experiencia.

En el plano conceptual, la aplicación articula los fundamentos de la gamificación: dinámicas, mecánicas y componentes con objetivos educativos concretos, promoviendo el aprendizaje significativo a través del descubrimiento, la validación en campo y la retroalimentación inmediata. De este modo, el conocimiento histórico y arquitectónico no se presenta como información estática, sino como recompensa, desafío y narrativa progresiva.

Si bien las pruebas se realizaron con pocas personas, estas primeras experiencias indican que la propuesta resulta atractiva, innovadora y motivadora para los usuarios, validando la hipótesis inicial: la combinación de inteligencia artificial visual y diseño lúdico puede fortalecer el vínculo emocional con el entorno urbano.

En definitiva, *Entre Diagonales* demuestra que una ciudad puede ser más que un conjunto de calles y edificios: puede convertirse en un tablero de juego, en un espacio de aprendizaje y en una experiencia inmersiva mediada por tecnología inteligente. Este trabajo no solo aporta un prototipo funcional, sino que abre una línea de investigación y desarrollo donde el turismo,

la educación y la inteligencia artificial convergen para resignificar la forma en que habitamos y descubrimos nuestros propios espacios.

Cerrar esta tesis implica también reafirmar una idea: cuando la tecnología se diseña con identidad, propósito y narrativa, deja de ser una herramienta invisible y se transforma en una llave que permite redescubrir lo que siempre estuvo ahí. *Entre diagonales*, entre monumentos y entre historias, la ciudad se revela como un juego que espera ser explorado.

7.2. Trabajos Futuros

Si bien *Entre Diagonales* cumple con los objetivos planteados en esta etapa de desarrollo, el proyecto presenta múltiples posibilidades de expansión tanto a nivel funcional como experiencial. A continuación, se describen las principales líneas de trabajo futuras identificadas.

7.2.1. Ampliación de la Cantidad de Recorridos

Una de las principales líneas de crecimiento consiste en aumentar la cantidad y diversidad de recorridos temáticos disponibles dentro de la aplicación.

Actualmente, las rutas desarrolladas permiten explorar determinados ejes patrimoniales de la ciudad. No obstante, La Plata presenta una riqueza histórica, arquitectónica y cultural que habilita la creación de nuevos circuitos, tales como:

- Recorridos arquitectónicos específicos (neogótico, ecléctico, modernismo).
- Circuitos sobre las Facultades de La Plata.
- Rutas sobre los museos.
- Recorridos nocturnos o narrativas alternativas.

La ampliación del contenido fortalecería la rejugabilidad de la aplicación y permitiría atraer tanto a turistas como a residentes interesados en redescubrir la ciudad desde nuevas perspectivas.

7.2.2. Creación de Equipos y Competencia Colaborativa

Otra línea de evolución consiste en habilitar la creación de equipos para realizar recorridos de manera colaborativa o competitiva.

La incorporación de dinámicas grupales permitiría:

- Competencias por tiempo o puntaje.

- Clasificaciones semanales o mensuales.
- Desafíos entre grupos.
- Eventos especiales organizados por la plataforma.

Este enfoque potenciaría el componente social de la experiencia y aumentaría el nivel de compromiso de los usuarios.

7.2.3. Optimización de la Navegación y Experiencia de Usuario (UX)

Si bien la interfaz actual cumple con criterios de funcionalidad y claridad, se identifican oportunidades de mejora en la experiencia de navegación.

Entre los posibles desarrollos se incluyen:

- Refinamiento del flujo entre mapa, cámara y desafíos.
- Mejora en la claridad visual de los objetivos activos.
- Optimización de los tiempos de carga.
- Incorporación de microinteracciones que refuercen la sensación lúdica.

Estas mejoras permitirían reducir fricciones en el uso cotidiano y consolidar una experiencia más intuitiva e inmersiva.

7.2.4. Expansión del Sistema de Monedas y Gemas

Actualmente, las monedas y gemas cumplen una función de recompensa dentro del sistema gamificado. Como trabajo futuro, se propone ampliar su utilidad para otorgar mayor profundidad estratégica al sistema.

Algunas posibles implementaciones incluyen:

- Canje de monedas por pistas adicionales en la búsqueda de objetos secretos.
- Desbloqueo de rutas especiales.
- Acceso anticipado a nuevos recorridos.
- Personalización de perfiles o insignias exclusivas.

La expansión del sistema económico interno permitiría generar una mayor motivación intrínseca y extender el ciclo de vida del usuario dentro de la aplicación.

7.2.5. Incorporación de Sistema de Audio-Guía Interactiva

Como línea de desarrollo futuro, se propone integrar un sistema de audio-guía que complemente la experiencia visual y gamificada de la aplicación.

La incorporación de narrativas sonoras permitiría:

- Profundizar la contextualización histórica de los monumentos.
- Mejorar la accesibilidad para personas con dificultades visuales.
- Enriquecer la inmersión durante los recorridos.
- Ofrecer distintos niveles de profundidad (breve, extendido o especializado).

A diferencia de una audio-guía tradicional lineal, esta funcionalidad podría integrarse dinámicamente al sistema existente, activándose automáticamente tras el reconocimiento visual de un monumento o al completar un desafío.

Asimismo, podrían incorporarse variantes futuras como:

- Voces narrativas con distintos enfoques (histórico, anecdótico o arquitectónico).
- Recreaciones dramatizadas de acontecimientos históricos.
- Integración con efectos sonoros ambientales.
- Soporte multilingüe para ampliar el alcance a turistas internacionales.

La implementación de audio-guías fortalecería el carácter multisensorial de la experiencia y ampliaría el alcance educativo del proyecto, consolidando a *Entre Diagonales* como una plataforma integral de exploración patrimonial.

Bibliografía

- [1] Ausubel, D. P., Novak J. D. y Hanesian H.: *Psicología educativa: un punto de vista cognoscitivo*. <https://docs.google.com/file/d/0B71eLBF7dL2vQUt1T3ZNWjdmTlk/edit?resourcekey=0-7rZQYX1VeCQaBs1MHiCVCg>, 1983. Accedido: 2026.
- [2] Rojas-García[®]: *Rojas-García, P., Sáez-Delgado, F., Badilla-Quintana, M., Jiménez-Pérez, L. (2022). Análisis de intervenciones educativas con videojuegos en educación secundaria: una revisión sistemática*. <https://periodicos.ufmg.br/index.php/textolivre/article/view/37810/30373>, 2022. Accedido: 2025.
- [3] Moresco, Nazareno: *PUMA: diseño y desarrollo de un videojuego educativo sobre agroecología*. <https://sedici.unlp.edu.ar/handle/10915/175200>, 2024. Accedido: 2026.
- [4] Zicherman, Gabe y Christopher Cunningham: *Gamification by design: Game mechanics in web mobile apps*. <https://www.studocu.com/cl/document/universidad-nacional-andres-bello/fisica/gamification-by-design/69607099>, 2011. Accedido: 2026.
- [5] Pablo-Diaz-Luque[®]: *Gamificación y destinos turísticos. Clasificación de los juegos desde el punto de vista de las organizaciones de destinos turísticos*. https://www.researchgate.net/publication/322970878_Gamificacion_y_destinos_turisticos_Clasificacion_de_los_juegos_desde_el_punto_de_vista_de_las_organizaciones_de_destinos_turisticos, 2017. Accedido: 2025.
- [6] INDEC[®]: *Turismo, conceptos y definiciones*. https://www.indec.gob.ar/ftp/cuadros/economia/turismo_cyd.pdf. Accedido: 2025.
- [7] Deterding, S., Dixon D. Khaled R. y Nacke L.: *Proceedings of the 15th international academic mindtrek conference: Envisioning future media environments*. <https://dl.acm.org/doi/epdf/10.1145/2181037.2181040>, 2011. Accedido: 2026.
- [8] Castro-Rivera[®]: *La gamificación aplicada en el sector turístico*. https://www.researchgate.net/publication/322970878_Gamificacion_y_destinos_turisticos_Clasificacion_de_los_juegos_desde_el_punto_de_vista_de_las_organizaciones_de_destinos_turisticos, 2021. Accedido: 2025.
- [9] Cynthia-Abraham[®]: *Gamificación y programas de fidelización en empresas de servicios*. , 2023. Accedido: 2025.

- [10] Gema-Serrano[®]: *Gamificación: transformar la experiencia para generar impacto real.* <https://www.open-ideas.es/gamificacion-transformar-experiencia-impacto/>, 2025. Accedido: 2025.
- [11] Silvia-Casamitjana[®]: *Gamificación para innovar en las empresas turísticas.* <https://www.segittur.es/blog/firmas-invitas/gamificacion-innovar-empresas/>, 2025. Accedido: 2025.
- [12] Werbach, Kevin y Dan Hunter: *For the Win: How Game Thinking Can Revolutionize Your Business*, 2012.
- [13] Hunicke, Robin, Marc LeBlanc y Robert Zubek: *MDA: A Formal Approach to Game Design and Game Research*, 2004.
- [14] Playmotiv[®]: *¿Cómo actúa la gamificación sobre la motivación? Motivación intrínseca y extrínseca en dinámicas de juego.* <https://playmotiv.com/gamificacion-y-motivacion/>, 2019. Accedido: 2025.
- [15] guevara[®] blanca-correa: *Gamificación en turismo cultural: cómo las experiencias lúdicas están transformando la forma de viajar.* <https://www.anahuac.mx/mexico/noticias/gamificacion-en-turismo-cultural#:~:text=En%20este%20contexto%2C%20la%20gamificaci%C3%B3n,con%20valor%20cultural%20o%20patrimonial>. Accedido: 2025.
- [16] Daremapp[®]: *¿Cómo usar la gamificación para aprender sobre un lugar mientras lo visito?* <https://daremapp.com/es/preguntas-frecuentes/gamificacion-para-aprender-viajando/>. Accedido: 2025.
- [17] Smarttravel[®]: *6 beneficios de aplicar la gamificación al sector turístico.* <https://www.smarttravel.news/6-beneficios-de-aplicar-la-gamificacion-al-sector-turistico/>. Accedido: 2025.
- [18] Destinos-Inteligentes[®]: *Identify CultureQuiz - Rutas Gamificadas.* <https://www.destinosinteligentes.es/soluciones/identify-culturequiz-rutas-gamificadas/>, 2024. Accedido: 2025.
- [19] Geoguessr: *Geoguessr.* <https://www.geoguessr.com/es>. Accedido: 2026.
- [20] Bancocooperativo[®]: *El turismo del futuro: tendencias del sector para los próximos años.* <https://www.bancocooperativo.es/es/digitalizacion/>

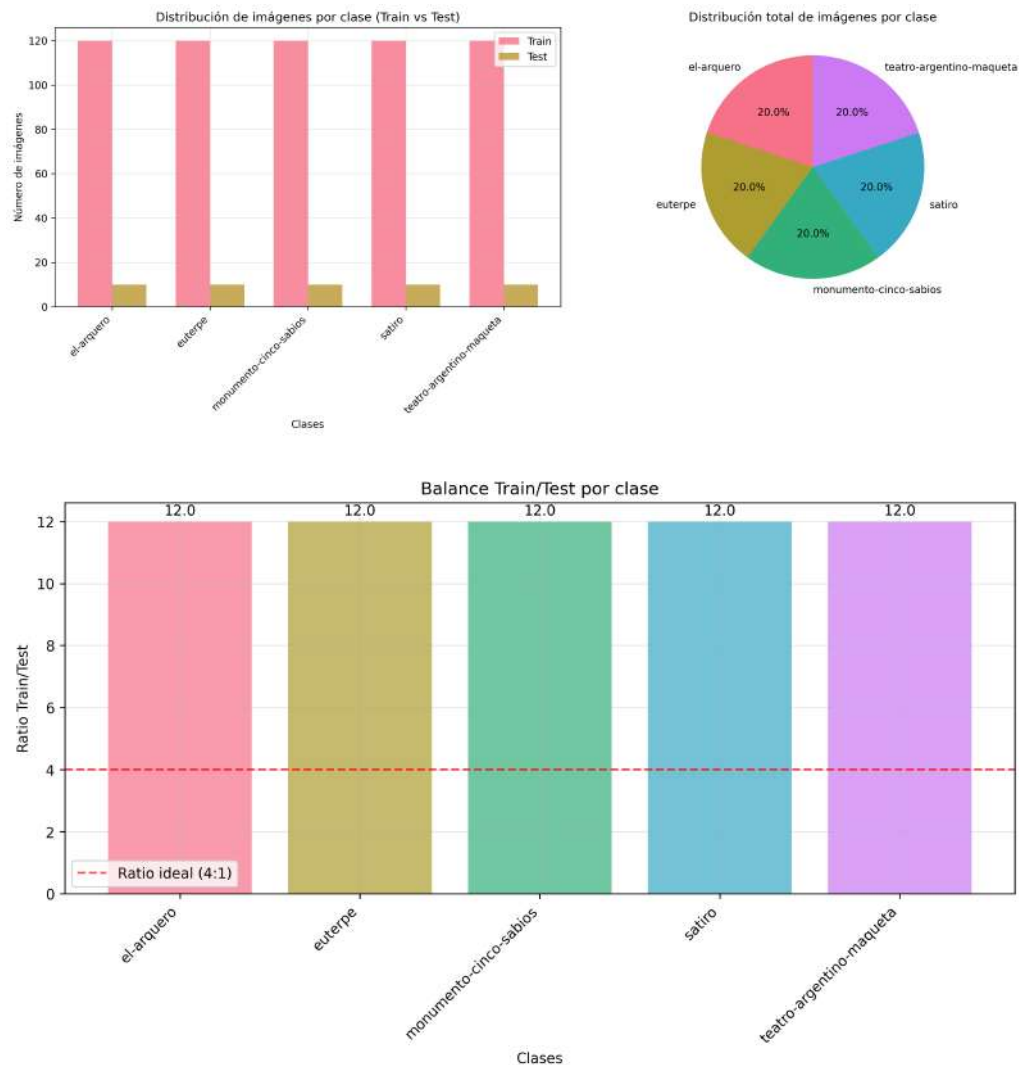
- turismo-del-futuro-tendencias-del-sector-proximos-anos, 2024. Accedido: 2025.
- [21] Pokemon: *Pokemon go*. <https://pokemongo.com/es>. Accedido: 2026.
- [22] Impulso06[®]: *Aplicaciones de Realidad Virtual y Aumentada en el sector turístico 4.0: desarrollo con Unity3D*. <https://impulso06.com/aplicaciones-de-realidad-virtual-y-aumentada-en-el-sector-turistico-4-0-desarrollo>. 2024. Accedido: 2025.
- [23] Kivisense: *Augmented reality paris*. <https://tryon.kivisense.com/blog/augmented-reality-paris/>. Accedido: 2026.
- [24] Myprivateparis: *Virtual reality of Notre-Dame cathedral*. <https://myprivateparis.com/virtual-reality-of-notre-dame-cathedral/>. Accedido: 2026.
- [25] Iebschool[®]: *La IA Generativa aplicada al Turismo*. <https://www.iebschool.com/hub/la-ia-generativa-aplicada-al-turismo-inteligencia-artificial/>, 2024. Accedido: 2025.
- [26] Despegar: *SOFIA, your new travel assistant with Artificial Intelligence*, 2024. <https://www.youtube.com/watch?v=GHAoS3IvGXE>, Consultado el 3 de abril de 2026.
- [27] Layla: *Layla AI Travel Planner*, 2026. <https://layla.ai/es>, Consultado el 3 de abril de 2026.
- [28] Amazon: *Alexa Smart Properties for Hospitality*, 2026. <https://www.amazon.com/b?node=21208571011>, Consultado el 3 de abril de 2026.
- [29] Booking.com: *Booking.com*, 2026. <https://www.booking.com/>, Consultado el 3 de abril de 2026.
- [30] Skyscanner: *Skyscanner*, 2026. <https://www.skyscanner.net/>, Consultado el 3 de abril de 2026.
- [31] Amadeus: *Amadeus Insights Blog*, 2026. <https://amadeus.com/en/insights/blog>, Consultado el 3 de abril de 2026.
- [32] Chen, Aili, Xuyang Ge, Ziquan Fu, Yanghua Xiao y Jiangjie Chen: *TravelAgent: An AI Assistant for Personalized Travel Planning*, 2024. <https://arxiv.org/abs/2409.08069>, Consultado el 3 de abril de 2026.
- [33] Google: *Find nearby inspiration on where to go in the Explore tab*, 2026. <https://support.google.com/maps/answer/10014587?hl=en>, Consultado el 3 de abril de 2026.

- [34] Google: *Search for nearby places and explore the area*, 2026. <https://support.google.com/maps/answer/4610185>, Consultado el 3 de abril de 2026.
- [35] AppGenie, Nimble: *Travel app statistics: Market share and size*. <https://www.nimbleappgenie.com/blogs/travel-app-statistics/>, 2025. Accedido: 2026.
- [36] sensortower[®]: *2025 state of mobile: Travel apps sustained gains after huge resurgence*. <https://sensortower.com/blog/2025-state-of-mobile-travel-apps>, 2025. Accedido: 2025.
- [37] cristina[®] violeta: *Desarrollo de modelo de negocio para una aplicación móvil de guías de turismo*. <https://repositorio.uchile.cl/bitstream/handle/2250/170917/Desarrollo-de-modelo-de-negocio-para-una-aplicacion-movil-de-guias-de-turismo.pdf?sequence=1>, 2019. Accedido: 2025.
- [38] meteo[®] open: *Api open source para obtener el clima de la ciudad de la plata*. <https://open-meteo.com/en/docs>. Accedido: 2025.
- [39] openrouteservice[®]: *Api open source para calculo de rutas*. <https://openrouteservice.org/>. Accedido: 2025.
- [40] La Plata, Misterios de la Ciudad de: *Misterios de la Ciudad de La Plata*. <https://misteriosdelaplata.blogspot.com/>. Accedido: 2026.
- [41] colourcontrastcc[®]: *Colour contrast cc*. <https://colourcontrast.cc/>. Accedido: 2026.
- [42] Django[®]: *Django framework*. <https://www.djangoproject.com/>. Accedido: 2026.
- [43] expo[®]: *Expo framework*. <https://docs.expo.dev/>. Accedido: 2025.
- [44] Native[®], React: *React native*. <https://reactnative.dev/>. Accedido: 2026.
- [45] kit[®] mix: *Mix-kit*. <https://mixkit.co/>. Accedido: 2025.
- [46] type[®] movable: *Movable type scripts, calculate distance, bearing and more between latitude/longitude points*. <https://www.movable-type.co.uk/scripts/latlong.html>. Accedido: 2025.
- [47] squoosh[®]: *Squoosh*. <http://squoosh.app>. Accedido: 2025.

Anexo 1

Puesta en practica con los objetos secretos

Distribución del dataset



Entrenamientos

Análisis del Entrenamiento y Rendimiento del Modelo ConvNeXt

El modelo ConvNeXt Tiny fue entrenado para reconocer cinco objetos escultóricos de la ciudad de La Plata: *el-arquero*, *euterpe*, *monumento-cinco-sabios*, *satiro* y *teatro-argentino-maqueta*. El entrenamiento se realizó durante 15 épocas, utilizando el dataset previamente normalizado y aumentado artificialmente para compensar la baja cantidad de imágenes disponibles.

A continuación se presenta un análisis comparativo entre la matriz de confusión obtenida, las curvas de accuracy y loss, y las métricas globales exportadas durante la evaluación del modelo.

Matriz de Confusión: desempeño perfecto clase por clase

La matriz de confusión evidencia un comportamiento ideal: el modelo clasificó correctamente los 50 ejemplos del conjunto de prueba, sin registrar ninguna confusión entre clases.

En la matriz (ver Figura) puede observarse que cada clase presenta 10 predicciones correctas sobre 10 muestras. No se observan falsos positivos ni falsos negativos.

Este resultado se alinea con las métricas detalladas en el archivo de clasificación, donde todas las clases presentan:

- **Precision = 1.0**
- **Recall = 1.0**
- **F1-Score = 1.0**

El rendimiento perfecto sugiere que el modelo logró aprender representaciones características muy consistentes para cada objeto, probablemente debido a que:

1. Las esculturas poseen **geometrías y estilos bien diferenciados**.
2. El pipeline de aumento de datos aplicado sobre el set de entrenamiento fue efectivo para generalizar variaciones de iluminación y color.
3. La arquitectura ConvNeXt, diseñada para robustez en visión por computadora, logra extraer características de alto nivel incluso con datasets pequeños.

Curvas de Accuracy: convergencia rápida y estable

En la gráfica de entrenamiento puede observarse que:

- La accuracy de entrenamiento se eleva desde **0.46 en la primera época** hasta **1.00 desde la época 12 en adelante**.
- La accuracy de validación muestra un patrón similar, alcanzando **1.00 en la última época**.

Este comportamiento indica una **convergencia muy estable**, y no presenta oscilaciones ni divergencias típicas de sobreajuste temprano. Las líneas de entrenamiento y validación se mantienen cercanas en todo momento, lo cual refuerza la estabilidad del modelo y la consistencia entre ambos conjuntos.

Curvas de Loss: disminución progresiva sin señales de overfitting

En el gráfico de pérdida:

- El *train loss* disminuye de **1.30 a 0.045**.
- El *validation loss* desciende de **0.88 a 0.099**.

A pesar de la diferencia inicial entre ambas curvas, la distancia se reduce progresivamente, indicando que el modelo mejora simultáneamente en ambos dominios. El hecho de que la pérdida de validación también disminuya a lo largo del entrenamiento confirma que el modelo **no memoriza**, sino que efectivamente generaliza.

Métricas globales del modelo

El log de entrenamiento muestra:

- **Accuracy final: 1.0000**
- **Validation accuracy: 1.0000**
- **Validation loss: 0.0999**
- **F1-score macro: 1.0000**
- Distribución de confianza:

- Alta (>90%): 80%
- Media (70–90%): 14%
- Baja (<70%): 6%**

El predominio de predicciones con alta confianza indica que el modelo no solo clasifica bien, sino que **lo hace con seguridad**, lo cual es crítico para la experiencia del usuario dentro de *Entre Diagonales* (por ejemplo, para la detección correcta de objetos secretos en tiempo real).

Interpretación global para la tesina

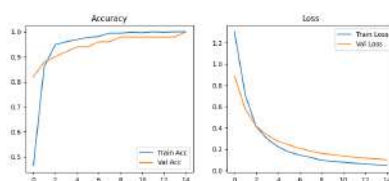
Los resultados obtenidos muestran que el modelo ConvNeXt Tiny fue capaz de aprender patrones visuales claros y discriminativos incluso con un dataset reducido, apoyándose en estrategias de normalización y aumento. La arquitectura moderna ConvNeXt, combinada con la calidad visual distintiva de los objetos, permitió alcanzar un rendimiento perfecto tanto en accuracy como en las métricas por clase.

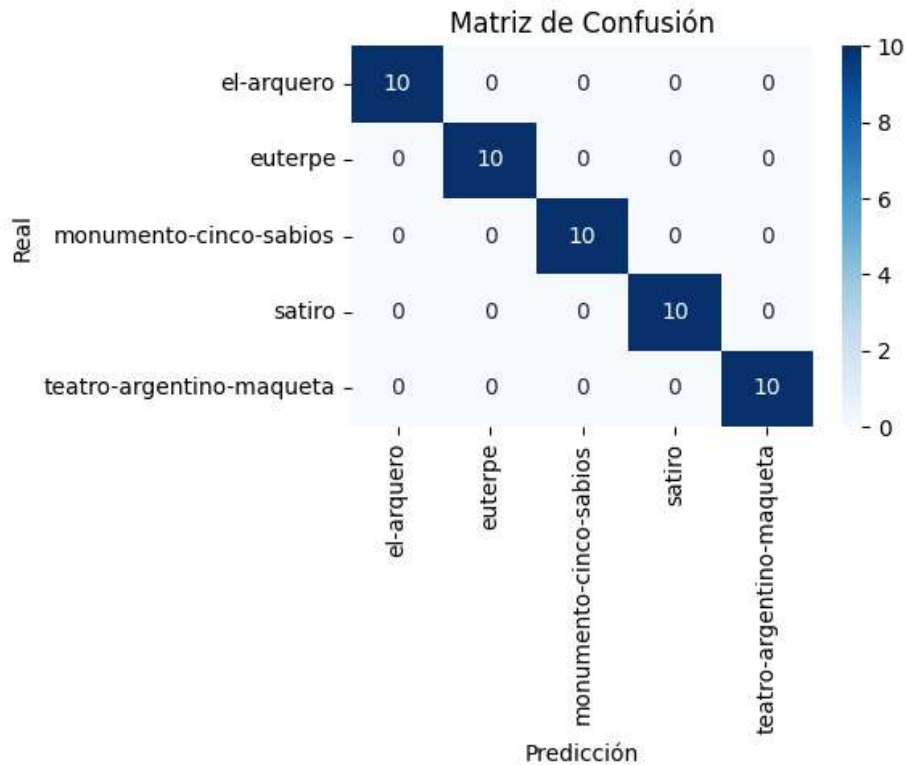
Este rendimiento es especialmente relevante para la implementación en la aplicación móvil, ya que garantiza:

- **Reconocimiento confiable de esculturas y elementos urbanos**, incluso en condiciones variables.
- **Rápida ejecución**, dado que ConvNeXt Tiny es eficiente para dispositivos móviles.
- **Consistencia crítica para la mecánica de juego** ligada a descubrimiento de objetos secretos.

Conclusión

En conclusión, el modelo entrenado cumple plenamente con los requisitos del proyecto y demuestra que es adecuado para ser integrado en el flujo de reconocimiento visual de *Entre Diagonales*.





Análisis del Entrenamiento y Rendimiento del Modelo MobileNetV2

El modelo MobileNetV2 fue entrenado sobre las mismas cinco clases de esculturas y objetos urbanos de La Plata, utilizando el mismo dataset normalizado y expandido mediante aumentos de datos. MobileNetV2 es una arquitectura optimizada para dispositivos móviles, diseñada para ser liviana y eficiente en tareas de visión por computadora, por lo que es especialmente relevante para aplicaciones móviles como *Entre Diagonales*.

A continuación se presenta una evaluación detallada del comportamiento del modelo, comparando la matriz de confusión, las curvas de entrenamiento y las métricas finales registradas.

Matriz de Confusión: desempeño excelente con una única confusión menor

La matriz de confusión muestra un rendimiento muy alto, con 49 predicciones correctas sobre 50 ejemplos del conjunto de prueba.

El único error se registra en la clase **euterpe**, donde 1 de sus 10 imágenes fue clasificada como *monumento-cinco-sabios*. Este tipo de error es esperable debido a que ambas esculturas poseen elementos verticales y detalles similares a la distancia, especialmente bajo iluminación variable.

En las métricas exportadas, esto se refleja en:

- **euterpe:**
 - precision = 1.0
 - recall = **0.9**
 - f1-score ≈ 0.947
- **monumento-cinco-sabios:**
 - precision ≈ **0.91**
 - recall = 1.0

El resto de las clases (*el-arquero*, *satiro* y *teatro-argentino-maqueta*) fueron clasificadas perfectamente, con valores de precision/recall/F1 iguales a 1.0.

En conjunto, el modelo alcanzó un **98% de accuracy**, lo cual es notablemente alto para un dataset reducido.

Curvas de Accuracy: rápida convergencia y comportamiento estable

En la gráfica de accuracy se observa:

- Un inicio sólido con **0.79 de accuracy de entrenamiento** en la primera época.
- Una subida abrupta hasta **1.0** ya en la tercera época.
- Accuracy de validación oscilando entre **0.92 y 0.98**, estabilizándose en **0.98**.

Esta convergencia rápida es característica de MobileNetV2 cuando se utiliza como *feature extractor*: las capas congeladas ya contienen representaciones universales que permiten aprender con pocos datos.

La pequeña oscilación en la validación entre las épocas 4 y 6 es normal y sugiere sensibilidad a casos específicos del conjunto de prueba, algo también esperable con datasets pequeños.

Curvas de Loss: descenso progresivo y sin señales de sobreajuste severo

En la gráfica de pérdida:

- El *train loss* cae de **0.61 a 0.0014**, mostrando un ajuste extremadamente fino.
- El *validation loss* desciende rápidamente hasta estabilizarse alrededor de **0.07** en las últimas épocas.

Aunque la pérdida de entrenamiento llega casi a cero, la validación no se dispara, lo que indica que **no hay un sobreajuste crítico**. El patrón es típico de MobileNetV2: aprende rápido y se estabiliza gracias a su arquitectura compacta.

Métricas globales del modelo

Según los logs de entrenamiento:

- **Accuracy final: 100%**
- **Validation accuracy: 98%**
- **Validation loss: 0.0714**
- **F1-score macro: 0.9799**
- **Distribución de confianza:**
 - Alta (>90%): **92% de las predicciones**
 - Media (70–90%): 6%
 - Baja (<70%): solo 2%

train-mobilenet

Este patrón demuestra que el modelo no sólo clasifica muy bien, sino que además lo hace con un alto nivel de seguridad, un factor crucial para la experiencia del usuario en la app.

Interpretación global y relevancia para la aplicación móvil

Los resultados muestran que MobileNetV2 logra un rendimiento sobresaliente, incluso ligeramente inferior al de ConvNeXt Tiny pero con ventajas importantes:

Ventajas específicas de MobileNetV2 para la app

- Es **extremadamente liviano** (≈9 MB de parámetros).
- Su tiempo de inferencia es muy rápido, ideal para **procesamiento en tiempo real dentro de Expo/React Native**.
- Su rendimiento se mantiene estable incluso con datasets pequeños gracias a sus bloques *inverted residuals*.

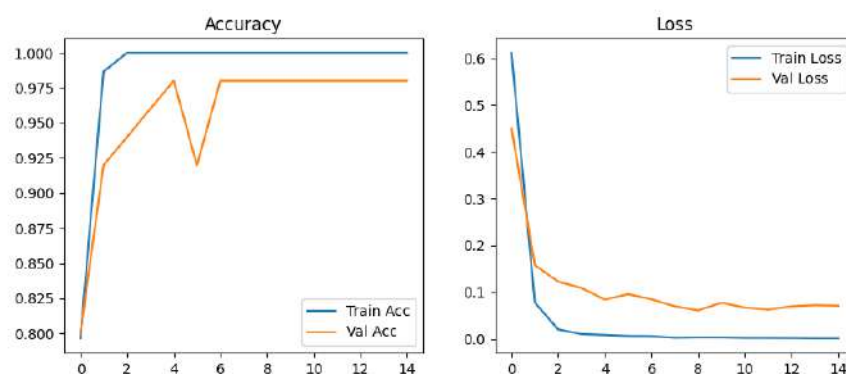
Sobre el único error registrado

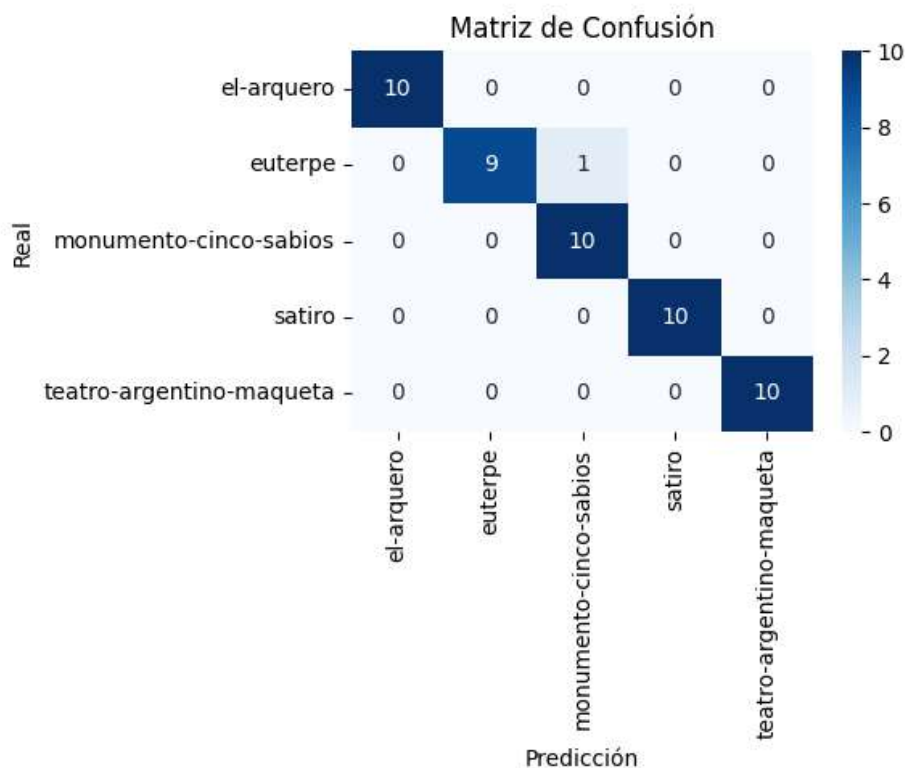
La única confusión entre *euterpe* y *monumento-cinco-sabios* indica que ciertas esculturas con formas esbeltas y tonos similares pueden resultar más difíciles bajo cambios fuertes de luz. Esto no afecta significativamente la experiencia final, y puede reducirse aún más incorporando:

- Más ejemplos angulares de ambas esculturas.
- Aumento específico basado en contraste y siluetas.
- Un pequeño ajuste fino (**fine-tuning**) de las últimas capas convolucionales.

Conclusión

MobileNetV2 demuestra ser una opción altamente efectiva y eficiente para integrar reconocimiento de objetos en *Entre Diagonales*, ofreciendo un equilibrio ideal entre precisión, velocidad y tamaño del modelo. Su rendimiento del 98% de accuracy lo convierte en un candidato sólido para la versión móvil en producción.





Análisis del Entrenamiento y Rendimiento del Modelo EfficientNetB3

El modelo EfficientNetB3 fue evaluado como una tercera alternativa para el reconocimiento de esculturas en *Entre Diagonales*. A diferencia de ConvNeXt y MobileNetV2, EfficientNetB3 es una arquitectura sustancialmente más pesada, diseñada para obtener alta precisión en escenarios con grandes cantidades de datos. Debido a su complejidad, su desempeño en datasets pequeños es menos predecible.

En esta sección se analizan la matriz de confusión, las curvas de accuracy y pérdida, y las métricas finales extraídas del proceso de entrenamiento.

Matriz de Confusión: resultados inconsistentes y poca discriminación entre clases

La matriz de confusión refleja un rendimiento significativamente inferior comparado con los otros modelos evaluados. El modelo solo logra predecir correctamente algunas muestras de *monumento-cinco-sabios* (9 de 10), pero falla en la mayoría de las demás clases:

- **el-arquero**: 0 predicciones correctas
- **euterpe**: solo 1 correcta
- **satiro**: solo 1 correcta
- **teatro-argentino-maqueta**: solo 2 correctas
- Una gran cantidad de imágenes fueron clasificadas erróneamente como *sátiro* o *monumento-cinco-sabios*

Esto indica que el modelo **no logró aprender representaciones discriminativas suficientes** para separar las clases de manera adecuada.

Las métricas oficiales lo confirman:

- F1-score para algunas clases cercano a **0.0**
- Accuracy global: **0.32**

- F1-score macro: **0.2656**
-

Curvas de Accuracy: ausencia de convergencia y bajo desempeño sostenido

A diferencia de los modelos anteriores:

- La accuracy de entrenamiento inicia en **0.22** y solo sube hasta **0.34** tras 15 épocas.
- La accuracy de validación oscila entre **0.20 y 0.44**, sin un patrón claro de mejora.

Este comportamiento indica que el modelo **no logró aprender las características esenciales de las imágenes**, y que los aumentos de datos no fueron suficientes para compensar su alta complejidad.

Aquí no hay señales de sobreajuste clásico; simplemente **no hay aprendizaje significativo**.

Curvas de Pérdida: descenso lento y que no se traduce en mejor performance

La pérdida de entrenamiento disminuye de **1.63 a 1.45**, mientras que la pérdida de validación desciende de **1.61 a 1.42**.

Si bien hay una leve mejora, esta no se refleja en accuracy ni en f1-score. Esta disociación entre disminución de pérdida y mala precisión es típica cuando:

- El modelo reduce su error, pero **no aprende los límites de decisión correctos**.
 - El dataset es insuficiente para soportar una arquitectura tan grande.
 - El preprocesamiento o tamaño de imagen esperados por EfficientNet no son óptimos para este dominio.
-

Métricas globales del entrenamiento

Según el log oficial:

- **Accuracy final:** 0.3433
- **Validation accuracy:** 0.3200
- **Validation loss:** 1.4265
- **F1-score macro:** 0.2656
- **Confianza de predicciones:**
 - Alta (>90%): **0%**
 - Media (70–90%): **0%**
 - Baja (<70%): **100%**

Esto indica que el modelo **estuvo inseguro en el 100% de los casos**, lo que confirma que no logró establecer patrones visuales robustos.

Interpretación general: por qué EfficientNet no funcionó en este proyecto

Los resultados sugieren que EfficientNetB3 **no es adecuado** para el dataset y la tarea específica por varias razones:

- **Modelo demasiado grande para pocos datos**

EfficientNetB3 posee más de **10 millones de parámetros**, lo que requiere:

- Miles de imágenes por clase para generalizar correctamente
- Entrenamientos prolongados
- Mayor diversidad de escenarios de iluminación, fondo y perspectiva

- **El dataset, aun aumentado, no tiene suficiente variabilidad**

Aunque se aplicaron aumentos de color, iluminación y normalización, el modelo sigue necesitando más ejemplos reales.

- **Las esculturas compartían patrones que EfficientNet no logró separar**

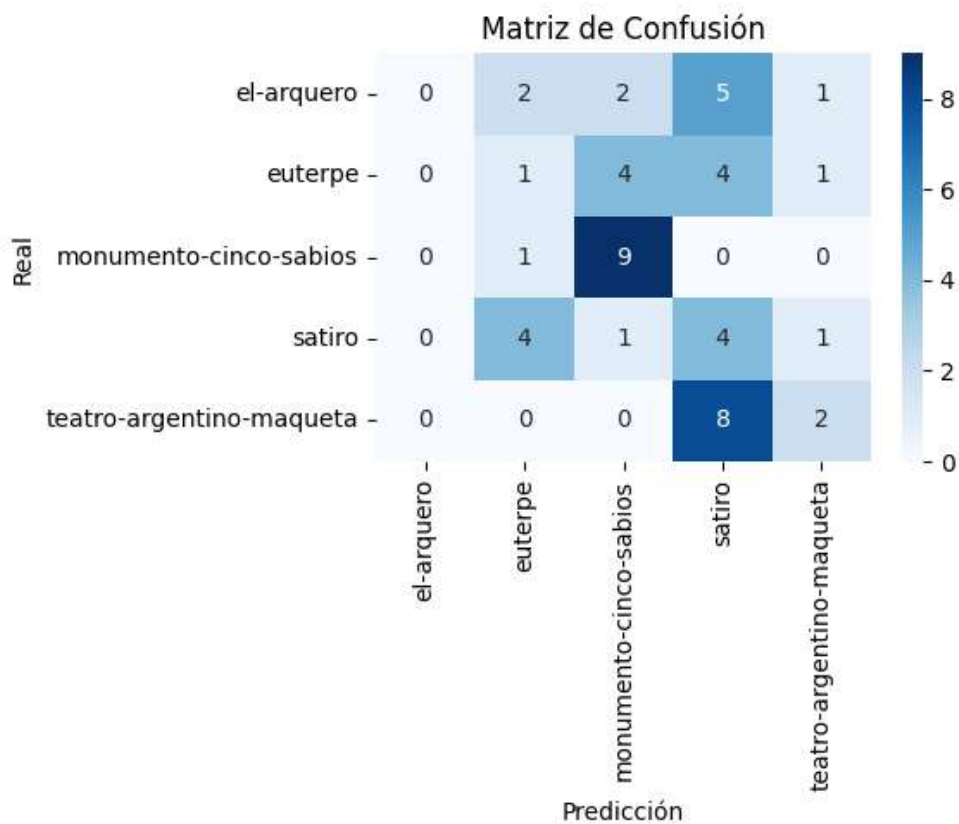
Debido a que EfficientNet trabaja con una resolución más grande (300×300), pequeños detalles pueden diluirse o mezclarse con ruido visual.

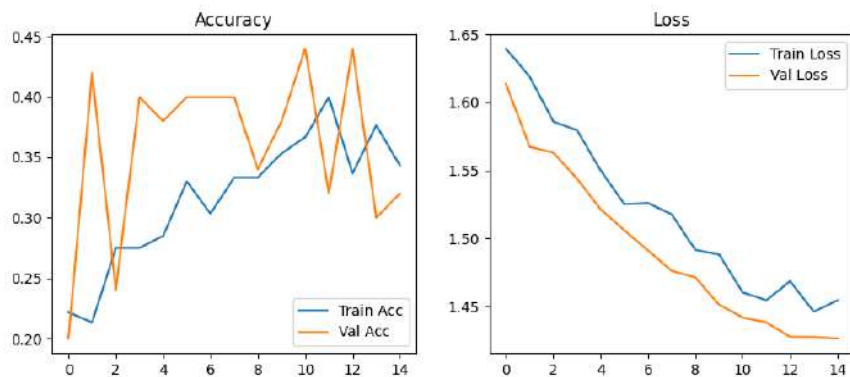
- **El entrenamiento fue demasiado corto para el tamaño del modelo**

15 épocas suelen ser suficientes para MobileNet o ConvNeXt Tiny, pero **insuficientes para EfficientNet**, que escala más lentamente.

Conclusión

EfficientNetB3 **no logra un aprendizaje significativo** en este contexto, y por lo tanto **no es recomendable para integrar en la aplicación**. ConvNeXt y MobileNet siguen siendo los modelos más adecuados para el reconocimiento en tiempo real dentro de *Entre Diagonales*.





Análisis del Entrenamiento y Rendimiento del Modelo ResNeSt50

El modelo ResNeSt50 fue evaluado como una arquitectura adicional de alta capacidad y enfoque jerárquico, combinando principios de *split-attention* y convoluciones profundas. Este tipo de arquitectura suele rendir bien en datasets grandes y variados, pero su desempeño puede ser irregular en escenarios con pocas muestras por clase, como en este caso.

Este análisis examina la matriz de confusión, las curvas de accuracy y loss, y las métricas globales registradas para comprender cómo se comportó el modelo frente a las cinco esculturas utilizadas en *Entre Diagonales*.

Matriz de Confusión: desempeño intermedio con patrones de confusión claros

La matriz de confusión revela un rendimiento **moderado**, superior al de EfficientNet pero muy por debajo de ConvNeXt y MobileNet:

- **el-arquero**: 4/10 predicciones correctas
- **euterpe**: 5/10 correctas (leve predominio, pero aún con confusiones)
- **monumento-cinco-sabios**: 9/10 correctas (la clase mejor reconocida)
- **sátiro**: 6/10 correctas
- **teatro-argentino-maqueta**: 8/10 correctas

Los patrones de error indican:

- Frecuentes confusiones entre **el-arquero** y **sátiro**, probablemente debido a similitudes en su silueta o textura en las fotos tomadas.
- Algunas imágenes de **euterpe** se confunden con el-arquero o teatro-argentino-maqueta.
- La clase **monumento-cinco-sabios** es la más distintiva visualmente y por eso fue clasificada casi siempre de manera correcta.

Las métricas oficiales reflejan este rendimiento intermedio:

- Accuracy global: **0.64**
- F1-score macro: **0.636**
- Precision y recall por clase variables (entre 0.44 y 0.88)

Curvas de Accuracy: mejora progresiva pero con fuerte inestabilidad

Las curvas de accuracy muestran:

- Una accuracy de entrenamiento que inicia en **0.40** y sube gradualmente hasta **0.77**.

- La accuracy de validación oscila entre **0.52 y 0.74**, sin un patrón estable de convergencia.

Este comportamiento indica que el modelo **sí aprende**, pero lo hace de forma irregular, probablemente debido a:

- Alta sensibilidad a pequeñas variaciones del dataset.
- Sobreajuste parcial a ciertas clases (ej.: monumento-cinco-sabios).
- La complejidad del modelo frente a un dataset reducido.

La variabilidad repetida en la validación sugiere que el modelo depende demasiado de características específicas no generalizables.

Curvas de Loss: decrecimiento correcto pero desconectado de la accuracy

Las curvas de pérdida muestran un descenso pronunciado en las primeras épocas:

- *Train loss*: de ~153 a menos de 1.0
- *Validation loss*: de ~53 a ~1.58

Este descenso es típico de modelos grandes cuando ajustan parámetros generales, pero la falta de relación directa entre una pérdida menor y mejor precisión indica:

- Dificultad del modelo para establecer **fronteras de decisión claras** entre las clases.
- Un aprendizaje que prioriza minimizar error a nivel global, pero sin distinguir adecuadamente características específicas de cada escultura.

Este fenómeno es común en modelos grandes entrenados con datasets pequeños.

Métricas globales del modelo

Los logs del entrenamiento muestran:

- **Accuracy final (train)**: 0.7767
- **Validation accuracy**: 0.6400
- **Validation loss**: 1.5810
- **F1-score macro**: 0.6365
- **Confianza de predicciones**:
 - Alta (>90%): **62%**
 - Media (70–90%): **10%**
 - Baja (<70%): **28%**

A diferencia de EfficientNet (que tuvo 0% predicciones seguras), ResNeSt sí logra predicciones de alta confianza, pero aún así se observan confusiones notorias en cuatro de las cinco clases.

Interpretación general: un modelo potente, pero no adecuado para este dataset

ResNeSt demuestra que logró aprender **rasgos generales** del dataset, pero no lo suficiente como para superar el rendimiento de MobileNetV2 ni acercarse al de ConvNeXt Tiny. Las razones principales son:

- **Modelo demasiado grande para el tamaño del dataset**
ResNeSt50 posee más de **24 millones de parámetros**, lo cual exige mucha más variabilidad visual para aprender patrones robustos.
- **Mayor tendencia a sobreajuste parcial**

Si bien la pérdida disminuye, la accuracy no acompaña de la misma forma, lo que sugiere que el modelo retuvo detalles innecesarios del set de entrenamiento.

- **Dependencia de características de alto nivel difíciles de separar**

Las esculturas más similares visualmente resultaron más difíciles para este modelo que para MobileNet o ConvNeXt.

- **Mejor que EfficientNet, pero insuficiente para la app**

Aunque ResNeSt supera ampliamente a EfficientNetB3 en todas las métricas, su precisión del 64% sigue siendo insuficiente para una aplicación de turismo gamificado donde las detecciones deben ser inmediatas y confiables.

Conclusión

ResNeSt50 demuestra un aprendizaje parcial, pero insuficiente para el reconocimiento preciso y confiable de esculturas dentro de *Entre Diagonales*. El modelo no será utilizado en la versión final de la aplicación, siendo superado ampliamente por ConvNeXt y MobileNet.

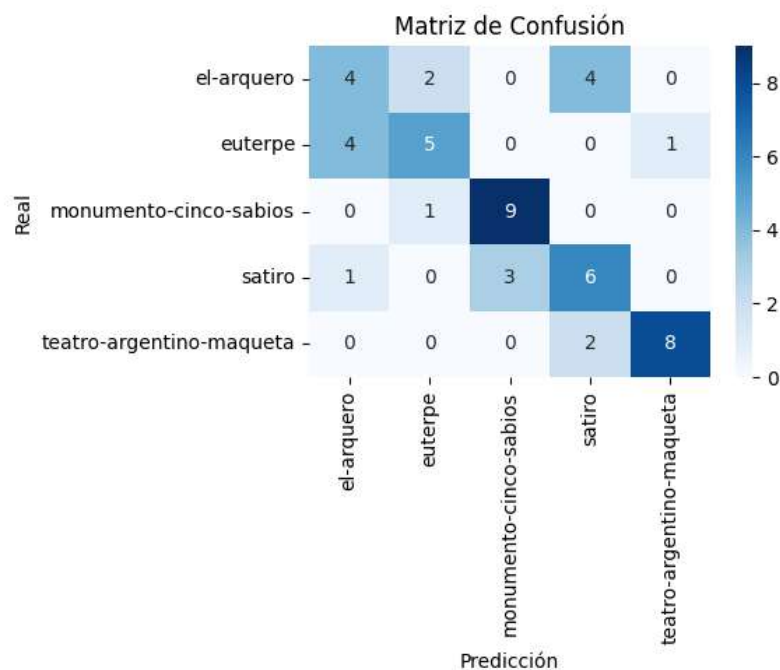
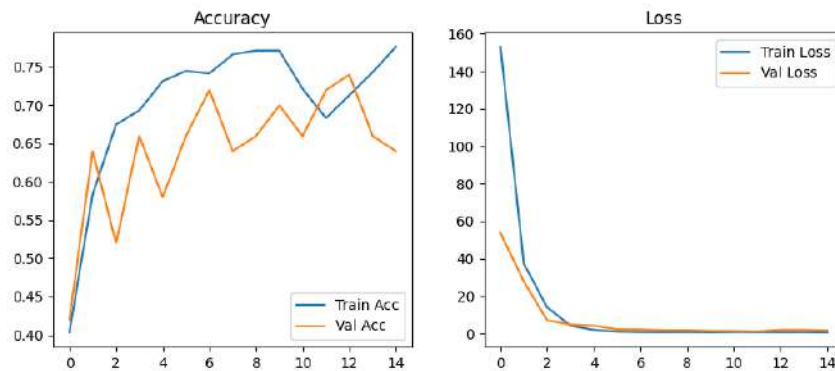


Tabla Comparativa de Modelos

Modelo	Accuracy Validación	F1-Score Macro	Confianza Alta	Parámetros	Desempeño General	Adecuado para Producción
ConvNeXt Tiny	100%	1.00	80%	~28M	Excelente (líder absoluto)	Si
MobileNetV2	98%	0.979	92%	~2.3M	Muy alto, rápido y liviano	Si
ResNeSt50	64%	0.636	62%	~24M	Intermedio, inconsistente	No
EfficientNetB3	32%	0.265	0%	~10.7M	Bajo, no aprende correctamente	No

Tabla Comparativa Final de Modelos

Modelo	Accuracy (%)	F1-Score	Confianza Alta (%)	Precisión Promedio	Pérdida (Loss)
ConvNeXt Tiny	100%	1.00	80%	1.00	0.0999
MobileNetV2	98%	0.9799	92%	0.9818	0.0714
ResNeSt50	64%	0.6364	62%	0.6417	1.5810
EfficientNetB3	32%	0.2656	0%	0.2556	1.4265

Anexo 2

Estructura de la Base de Datos — Entre Diagonales

Diagrama de relaciones



Modelos

User

Tabla: **users**

Extiende **AbstractUser** de Django. Es el modelo central de la aplicación.

Campo	Tipo	Restricciones	Descripción
id	Integer (PK)	Auto, único	Clave primaria
email	EmailField	Único, requerido	Email del usuario, usado como campo de login

Campo	Tipo	Restricciones	Descripción
<code>username</code>	CharField	Requerido	Nombre de usuario (heredado de <code>AbstractUser</code>)
<code>password</code>	CharField	—	Contraseña hasheada
<code>provider</code>	CharField(20)	Default: <code>"email"</code>	Método de registro: <code>"email"</code> o <code>"google"</code>
<code>display_name</code>	CharField(100)	Nullable	Nombre a mostrar en la interfaz
<code>coins</code>	BigIntegerField	Min: 0, Default: 0	Monedas del usuario (moneda premium)
<code>gems</code>	BigIntegerField	Min: 0, Default: 0	Gemas del usuario (moneda secundaria)
<code>experience</code>	BigIntegerField	Min: 0, Default: 0	Puntos de experiencia acumulados
<code>character</code>	FK → Character	Nullable, SET_NULL on delete	Personaje seleccionado por el usuario
<code>level</code>	FK → Level	Nullable, SET_NULL on delete	Nivel actual del usuario

Comportamiento especial:

- Al crear un usuario, se le asigna automáticamente el nivel con menor `xp_required`.
- Al guardar por primera vez, se crea automáticamente un registro `UserSettings` asociado.

UserSettings

Tabla: `user_settings`

Configuración de cuenta de cada usuario. Se crea automáticamente al registrar un `User`.

Campo	Tipo	Restricciones	Descripción
<code>id</code>	Integer (PK)	Auto	Clave primaria
<code>user</code>	FK → User	OneToOne, CASCADE	Usuario al que pertenece esta configuración
<code>notifications</code>	BooleanField	Default: False	Si el usuario tiene notificaciones activadas
<code>notification_token</code>	CharField(255)	Nullable	Token del dispositivo para push notifications (Expo)
<code>on_boarding_completed_at</code>	DateTimeField	Nullable	Fecha en que completó el proceso de onboarding

Level

Tabla: `levels`

Define los niveles de progresión de la aplicación.

Campo	Tipo	Restricciones	Descripción
<code>id</code>	Integer (PK)	Auto	Clave primaria
<code>name</code>	CharField(100)	Requerido	Nombre del nivel (ej: "Explorador", "Aventurero")
<code>description</code>	TextField	Requerido	Descripción del nivel
<code>xp_required</code>	BigIntegerField	Min: 0, Default: 0	Experiencia mínima necesaria para alcanzar este nivel
<code>image_url</code>	URLField(500)	Nullable	URL de la imagen representativa del nivel

Relaciones: Referenciado por `User.level` .

Character

Tabla: `characters`

Personajes que los usuarios pueden elegir como avatar dentro de la aplicación.

Campo	Tipo	Restricciones	Descripción
<code>id</code>	Integer (PK)	Auto	Clave primaria
<code>name</code>	CharField(100)	Requerido	Nombre del personaje
<code>description</code>	TextField	Requerido	Descripción del personaje
<code>image_url</code>	URLField(500)	Nullable	URL de la imagen del personaje

Relaciones: Referenciado por `User.character` .

Tour

Tabla: `tours`

Recorridos turísticos que agrupan una secuencia de spots (monumentos/lugares).

Campo	Tipo	Restricciones	Descripción
<code>id</code>	Integer (PK)	Auto	Clave primaria
<code>name</code>	CharField(100)	Único, requerido	Nombre del tour
<code>description</code>	TextField	Nullable	Descripción del tour
<code>tag</code>	CharField(50)	Nullable	Etiqueta o categoría del tour
<code>active</code>	BooleanField	Default: True	Si el tour está disponible para los usuarios

Relaciones: Un `Tour` tiene muchos `TourSpot` (spots ordenados) y muchos `UserTour` (relaciones con usuarios).

UserTour

Tabla: `user_tours`

Tabla intermedia que registra la relación entre un usuario y un tour (progreso, estado y feedback).

Campo	Tipo	Restricciones	Descripción
<code>id</code>	Integer (PK)	Auto	Clave primaria
<code>user</code>	FK → User	CASCADE on delete	Usuario que realiza el tour
<code>tour</code>	FK → Tour	CASCADE on delete	Tour relacionado
<code>feedback</code>	FK → Feedback	OneToOne, Nullable	Feedback dejado por el usuario al finalizar
<code>progress</code>	IntegerField	Min: 0, Max: 100, Default: 0	Cantidad de spots completados en el tour
<code>completed_at</code>	DateTimeField	Nullable	Fecha/hora en que el usuario completó el tour

Spot

Tabla: `spots`

Representa un lugar físico (monumento, punto de interés) que puede formar parte de un tour.

Campo	Tipo	Restricciones	Descripción
<code>id</code>	Integer (PK)	Auto	Clave primaria
<code>name</code>	CharField(100)	Requerido	Nombre del lugar
<code>slug</code>	SlugField(100)	Único, requerido	Identificador URL-friendly (usado en verificación IA)
<code>address</code>	CharField(255)	Nullable	Dirección física del lugar
<code>latitude</code>	FloatField	Nullable	Latitud geográfica
<code>longitude</code>	FloatField	Nullable	Longitud geográfica
<code>description</code>	TextField	Requerido	Descripción general del lugar
<code>historical_information</code>	TextField	Nullable	Información histórica del lugar
<code>fun_facts</code>	TextField	Nullable	Datos curiosos del lugar
<code>schedule</code>	TextField	Nullable	Horario de apertura/funcionamiento
<code>ticket_price</code>	DecimalField(10,2)	Nullable	Precio de entrada (si aplica)
<code>wheelchair_accessible</code>	BooleanField	Default: False	Accesibilidad para sillas de ruedas
<code>tag</code>	CharField(50)	Nullable	Etiqueta o categoría del spot
<code>activated</code>	BooleanField	Default: True	Si el spot está activo y visible
<code>image_urls</code>	JSONField	Default: []	Lista de URLs de imágenes del lugar
<code>ranking</code>	DecimalField(2,1)	Default: 0.0	Puntuación/ranking del spot

Relaciones: Un `Spot` puede estar en varios tours (a través de `TourSpot`), tener varios `Quiz` y varios `SpotSecretItem`.

TourSpot

Tabla: `tour_spots`

Tabla intermedia que define qué spots pertenecen a un tour y en qué orden deben visitarse.

Campo	Tipo	Restricciones	Descripción
<code>id</code>	Integer (PK)	Auto	Clave primaria
<code>spot</code>	FK → Spot	CASCADE on delete	Spot incluido en el tour
<code>tour</code>	FK → Tour	CASCADE on delete	Tour al que pertenece el spot
<code>order</code>	PositiveIntegerField	Min: 1, Default: 1	Posición del spot dentro del tour

Restricción única: La combinación `(spot, tour, order)` debe ser única.

SecretItem

Tabla: `secret_items`

Objetos secretos escondidos en los spots que los usuarios pueden descubrir subiendo una foto verificada por IA.

Campo	Tipo	Restricciones	Descripción
<code>id</code>	Integer (PK)	Auto	Clave primaria
<code>name</code>	CharField(100)	Requerido	Nombre del objeto secreto

Campo	Tipo	Restricciones	Descripción
<code>description</code>	TextField	Requerido	Descripción del objeto
<code>image_url</code>	URLField(500)	Nullable	URL de la imagen del objeto
<code>is_active</code>	BooleanField	Default: True	Si el objeto está disponible para ser encontrado
<code>hint</code>	TextField	Nullable	Pista para ayudar al usuario a encontrarlo
<code>slug</code>	SlugField(100)	Único	Identificador URL-friendly (usado en verificación IA)

UserSecretItem

Tabla: `user_secret_items`

Registra qué objetos secretos obtuvo cada usuario.

Campo	Tipo	Restricciones	Descripción
<code>id</code>	Integer (PK)	Auto	Clave primaria
<code>user</code>	FK → User	CASCADE on delete	Usuario que obtuvo el objeto
<code>secret_item</code>	FK → SecretItem	CASCADE on delete	Objeto secreto obtenido
<code>obtained_at</code>	DateTimeField	Auto al crear	Fecha y hora en que fue obtenido

Restricción única: La combinación `(user, secret_item)` debe ser única — un usuario no puede obtener el mismo objeto dos veces.

SpotSecretItem

Tabla: `spot_secret_items`

Define qué objetos secretos están asociados a qué spots (dónde se puede encontrar cada objeto).

Campo	Tipo	Restricciones	Descripción
<code>id</code>	Integer (PK)	Auto	Clave primaria
<code>spot</code>	FK → Spot	CASCADE on delete	Spot donde se puede encontrar el objeto
<code>secret_item</code>	FK → SecretItem	CASCADE on delete	Objeto secreto ubicado en el spot

Restricción única: La combinación `(spot, secret_item)` debe ser única.

Quiz

Tabla: `quizzes`

Preguntas de trivia asociadas a un spot específico.

Campo	Tipo	Restricciones	Descripción
<code>id</code>	Integer (PK)	Auto	Clave primaria
<code>name</code>	CharField(255)	Requerido	Texto de la pregunta
<code>explanation</code>	TextField	Requerido	Explicación que se muestra después de responder
<code>activated</code>	BooleanField	Default: True	Si el quiz está activo y disponible
<code>spot</code>	FK → Spot	CASCADE on delete	Spot al que pertenece este quiz

Answer

Tabla: `answers`

Opciones de respuesta para un quiz. Cada quiz tiene múltiples respuestas, una de las cuales es correcta.

Campo	Tipo	Restricciones	Descripción
<code>id</code>	Integer (PK)	Auto	Clave primaria
<code>quiz</code>	FK → Quiz	CASCADE on delete	Quiz al que pertenece esta respuesta
<code>name</code>	CharField(255)	Requerido	Texto de la respuesta
<code>is_correct</code>	BooleanField	Default: False	Si esta respuesta es la correcta

UserQuiz

Tabla: `user_quizzes`

Registra los intentos de un usuario en los quizzes (si acertó o no).

Campo	Tipo	Restricciones	Descripción
<code>id</code>	Integer (PK)	Auto	Clave primaria
<code>user</code>	FK → User	CASCADE on delete	Usuario que respondió el quiz
<code>quiz</code>	FK → Quiz	CASCADE on delete	Quiz respondido
<code>success</code>	BooleanField	Default: False	Si el usuario respondió correctamente

Achievement

Tabla: `achievements`

Logros que los usuarios pueden desbloquear y canjear por recompensas.

Campo	Tipo	Restricciones	Descripción
<code>id</code>	Integer (PK)	Auto	Clave primaria
<code>name</code>	CharField(100)	Único, requerido	Nombre del logro
<code>description</code>	TextField	Nullable	Descripción del logro
<code>multiplier</code>	FloatField	Default: 1.0	Multiplicador de gemas al canjear
<code>expired_at</code>	DateTimeField	Nullable	Fecha de expiración del logro (si aplica)
<code>goal</code>	IntegerField	Min: 0, Default: 0	Cantidad objetivo a alcanzar para completar el logro
<code>goal_type</code>	CharField(50)	Default: <code>"tours"</code>	Tipo de acción que mide el logro: <code>"tours"</code> , <code>"trivia"</code> , <code>"secret_item"</code>
<code>is_active</code>	BooleanField	Default: True	Si el logro está activo y disponible
<code>image_url</code>	URLField(500)	Nullable	URL de la imagen del logro

UserAchievement

Tabla: `user_achievements`

Registra el progreso de cada usuario en cada achievement.

Campo	Tipo	Restricciones	Descripción
<code>id</code>	Integer (PK)	Auto	Clave primaria
<code>user</code>	FK → User	CASCADE on delete	Usuario al que pertenece el progreso
<code>achievement</code>	FK → Achievement	CASCADE on delete	Achievement relacionado
<code>progress</code>	IntegerField	Min: 0, Max: 1000, Default: 0	Progreso actual hacia el objetivo
<code>completed_at</code>	DateTimeField	Nullable	Fecha en que el usuario completó y canjeó el logro

Restricción única: La combinación `(user, achievement)` debe ser única.

Feedback

Tabla: `feedbacks`

Evaluación que un usuario deja al finalizar un tour.

Campo	Tipo	Restricciones	Descripción
<code>id</code>	Integer (PK)	Auto	Clave primaria
<code>name</code>	CharField(100)	Requerido	Nombre/título del feedback
<code>qualification</code>	PositiveIntegerField	Default: 0	Puntuación numérica otorgada al tour
<code>trivia_liked</code>	BooleanField	Default: False	Si al usuario le gustó la sección de trivia
<code>spots_liked</code>	BooleanField	Default: False	Si al usuario le gustaron los spots
<code>secrets_liked</code>	BooleanField	Default: False	Si al usuario le gustaron los objetos secretos
<code>route_liked</code>	BooleanField	Default: False	Si al usuario le gustó la ruta
<code>rewards_liked</code>	BooleanField	Default: False	Si al usuario le gustaron las recompensas
<code>comment</code>	TextField	Nullable	Comentario libre del usuario

Relaciones: Referenciado por `UserTour.feedback` (OneToOne). Un tour completado puede tener un único feedback.

Notification

Tabla: `notifications`

Notificaciones push enviadas desde el sistema (solo por superusuarios).

Campo	Tipo	Restricciones	Descripción
<code>id</code>	Integer (PK)	Auto	Clave primaria
<code>title</code>	CharField(255)	Requerido	Título de la notificación
<code>description</code>	TextField	Requerido	Cuerpo/descripción de la notificación
<code>category</code>	CharField(50)	Nullable	Categoría de la notificación (ej: <code>"promocion"</code> , <code>"actualizacion"</code> , <code>"recordatorio"</code>)
<code>created_at</code>	DateTimeField	Auto al crear	Fecha de creación del registro
<code>sent_at</code>	DateTimeField	Nullable	Fecha en que fue enviada efectivamente

UserNotification

Tabla: `user_notifications`

Tabla intermedia que registra qué notificaciones recibió cada usuario y cuándo las leyó.

Campo	Tipo	Restricciones	Descripción
<code>id</code>	Integer (PK)	Auto	Clave primaria
<code>notification</code>	FK → Notification	CASCADE on delete	Notificación enviada
<code>user</code>	FK → User	CASCADE on delete	Usuario receptor
<code>sent_at</code>	DateTimeField	Auto al crear	Fecha en que fue enviada a este usuario
<code>read_at</code>	DateTimeField	Nullable	Fecha en que el usuario la leyó

Restricción única: La combinación `(notification, user)` debe ser única.

Waitlist

Tabla: `waitlist`

Registro de emails de personas interesadas en la aplicación antes de su lanzamiento.

Campo	Tipo	Restricciones	Descripción
<code>id</code>	Integer (PK)	Auto	Clave primaria
<code>email</code>	EmailField	Único, requerido	Email del interesado
<code>created_at</code>	DateTimeField	Auto al crear	Fecha de registro en la waitlist
<code>notified</code>	BooleanField	Default: False	Si ya fue notificado sobre el lanzamiento
<code>notified_at</code>	DateTimeField	Nullable	Fecha en que fue notificado

Nota: Este modelo no tiene relación con `User`. Cualquier persona puede registrarse (endpoint público).

Resumen de relaciones

Relación	Tipo	Descripción
<code>User</code> → <code>Level</code>	ManyToOne	Un usuario tiene un nivel; un nivel puede tener muchos usuarios
<code>User</code> → <code>Character</code>	ManyToOne	Un usuario tiene un personaje elegido
<code>User</code> ↔ <code>UserSettings</code>	OneToOne	Cada usuario tiene exactamente una configuración
<code>User</code> ↔ <code>Tour</code> (via <code>UserTour</code>)	ManyToMany	Un usuario puede hacer muchos tours; cada tour puede ser hecho por muchos usuarios
<code>Tour</code> ↔ <code>Spot</code> (via <code>TourSpot</code>)	ManyToMany	Un tour tiene varios spots; un spot puede estar en varios tours
<code>UserTour</code> → <code>Feedback</code>	OneToOne	Un tour completado puede tener un feedback
<code>Spot</code> → <code>Quiz</code>	OneToMany	Un spot puede tener varios quizzes
<code>Quiz</code> → <code>Answer</code>	OneToMany	Un quiz tiene varias respuestas posibles
<code>User</code> ↔ <code>Quiz</code> (via <code>UserQuiz</code>)	ManyToMany	Un usuario puede responder varios quizzes
<code>Spot</code> ↔ <code>SecretItem</code> (via <code>SpotSecretItem</code>)	ManyToMany	Un spot puede tener varios objetos secretos

Relación	Tipo	Descripción
<code>User</code> ↔ <code>SecretItem</code> (via <code>UserSecretItem</code>)	ManyToMany	Un usuario puede obtener varios objetos secretos
<code>User</code> ↔ <code>Achievement</code> (via <code>UserAchievement</code>)	ManyToMany	Un usuario puede progresar en varios achievements
<code>User</code> ↔ <code>Notification</code> (via <code>UserNotification</code>)	ManyToMany	Un usuario puede recibir varias notificaciones

Anexo 3

Documentación de Endpoints — Entre Diagonales API

Base URL: `/api/`

Autenticación: JWT Bearer Token (`Authorization: Bearer <access_token>`) salvo que se indique lo contrario.

Autenticación

POST `/api/auth/login/`

Inicia sesión con email y contraseña.

Auth requerida: No

Body:

```
{
  "email": "usuario@ejemplo.com",
  "password": "mi_contraseña"
}
```

Respuesta 200:

```
{
  "access": "<jwt_access_token>",
  "refresh": "<jwt_refresh_token>"
}
```

Errores: `400` datos inválidos · `401` credenciales incorrectas

POST `/api/auth/google/`

Inicia sesión con un token de Google OAuth2. Si el usuario no existe, lo crea automáticamente.

Auth requerida: No

Body:

```
{
  "token": "<google_id_token>"
}
```

Respuesta 200:

```
{
  "access": "<jwt_access_token>",
}
```

```
"refresh": "<jwt_refresh_token>"
}
```

Errores: 400 token de Google inválido o email faltante

POST /api/auth/register/

Registra un nuevo usuario con email y contraseña.

Auth requerida: No

Body:

```
{
  "email": "usuario@ejemplo.com",
  "username": "mi_usuario",
  "password": "mi_contraseña"
}
```

Respuesta 201:

```
{
  "access": "<jwt_access_token>",
  "refresh": "<jwt_refresh_token>"
}
```

Errores: 400 datos inválidos (email duplicado, contraseña débil, etc.)

POST /api/auth/change-password/

Cambia la contraseña del usuario autenticado.

Auth requerida: Sí

Body:

```
{
  "current_password": "contraseña_actual",
  "new_password": "nueva_contraseña",
  "new_password_confirm": "nueva_contraseña"
}
```

Respuesta 200:

```
{
  "message": "Contraseña cambiada exitosamente"
}
```

Errores: 400 datos inválidos o contraseña actual incorrecta · 401 no autenticado

POST /api/auth/password-reset/

Solicita un email de recuperación de contraseña.

Auth requerida: No

Body:

```
{
  "email": "usuario@ejemplo.com"
}
```

Respuesta 200:

```
{
  "message": "Se ha enviado un email con las instrucciones para restablece  
r tu contraseña"
}
```

Nota: Por seguridad, siempre devuelve 200 aunque el email no exista en el sistema.

Errores: 400 email inválido · 500 error de configuración SMTP

GET · POST `/api/password-reset/<token>/`

Vista HTML (no JSON) para completar el restablecimiento de contraseña. El usuario accede a esta URL desde el link del email.

- `GET` : Muestra el formulario para ingresar la nueva contraseña.
- `POST` : Procesa el formulario y actualiza la contraseña.

Auth requerida: No (usa el token como autenticación temporal)

Waitlist

POST `/api/waitlist/`

Registra un email en la lista de espera. Endpoint público con rate limiting de **5 peticiones por hora por IP**.

Auth requerida: No

Body:

```
{
  "email": "interesado@ejemplo.com"
}
```

Respuesta 201:

```
{
  "email": "interesado@ejemplo.com",
  "created_at": "2024-01-15T10:30:00Z",
  "notified": false,
  "notified_at": null,
}
```

```
{
  "message": "Email registrado exitosamente en la waitlist"
}
```

Errores: 400 email inválido o ya registrado · 429 demasiadas peticiones

Perfil

GET /api/profile/

Obtiene la información completa del perfil del usuario autenticado.

Auth requerida: Sí

Respuesta 200:

```
{
  "id": 1,
  "email": "usuario@ejemplo.com",
  "username": "mi_usuario",
  "display_name": "Mi Nombre",
  "coins": 500,
  "gems": 10,
  "experience": 1200,
  "level": { "id": 2, "name": "Aventurero", "xp_required": 1000, "image_url": "..."},
  "character": { "id": 1, "name": "Explorador", "image_url": "..."},
  "settings": {
    "notifications": true,
    "on_boarding_completed_at": "2024-01-10T08:00:00Z"
  }
}
```

Errores: 401 no autenticado

PUT /api/profile/

Edita la información del perfil del usuario autenticado. Todos los campos son opcionales (PATCH semántico).

Auth requerida: Sí

Body (campos opcionales):

```
{
  "username": "nuevo_usuario",
  "email": "nuevo@ejemplo.com",
  "display_name": "Nuevo Nombre",
  "character": 2
}
```

Respuesta 200: Perfil completo actualizado (mismo formato que GET /api/profile/)

Errores: 400 email o username ya en uso · 401 no autenticado

POST `/api/profile/notifications/`

Actualiza la configuración de notificaciones push del usuario. Se puede enviar uno o ambos campos.

Auth requerida: Sí

Body:

```
{
  "notification_token": "ExponentPushToken[xxxxxx]",
  "notifications": true
}
```

Respuesta 200:

```
{
  "message": "Configuración de notificaciones actualizada exitosamente",
  "notification_token": "ExponentPushToken[xxxxxx]",
  "notifications": true
}
```

Errores: 400 se debe enviar al menos uno de los dos campos · 401 no autenticado

Onboarding

POST `/api/onboarding/`

Marca el onboarding del usuario como completado. Opcionalmente registra el token de notificación push.

Auth requerida: Sí

Body:

```
{
  "notification_token": "ExponentPushToken[xxxxxx]"
}
```

(el campo `notification_token` es opcional)

Respuesta 200:

```
{
  "message": "Onboarding completado exitosamente"
}
```

Errores: 400 datos inválidos · 401 no autenticado

Notificaciones

POST `/api/notifications/send/`

Envía una notificación push a usuarios. **Solo disponible para superusuarios.**

- Si `notification_tokens` está vacío: envía a todos los usuarios con notificaciones activadas.
- Si `notification_tokens` tiene valores: envía solo a esos usuarios.

Auth requerida: Sí (superusuario)

Body:

```
{
  "title": "¡Nuevo tour disponible!",
  "description": "Explorá el nuevo recorrido por el centro histórico.",
  "category": "actualizacion",
  "notification_tokens": []
}
```

Respuesta 201:

```
{
  "message": "Notificación enviada exitosamente",
  "notification_id": 5,
  "users_notified": 42,
  "tokens_sent": 42,
  "push_responses": [...]
}
```

Errores: `400` datos inválidos o sin usuarios con token válido · `401` no autenticado · `403` no es superusuario

Respuesta 207: Si la notificación se guardó en BD pero hubo error al enviar las push.

Personajes

GET `/api/characters/`

Lista todos los personajes disponibles para elegir como avatar.

Auth requerida: Sí

Respuesta 200:

```
[
  {
    "id": 1,
    "name": "Explorador",
    "description": "Un aventurero curioso...",
    "image_url": "https://..."
  }
]
```

Errores: 401 no autenticado

Niveles

GET `/api/levels/`

Lista todos los niveles disponibles, ordenados de menor a mayor XP requerida.

Auth requerida: Sí

Respuesta 200:

```
[
  {
    "id": 1,
    "name": "Principiante",
    "description": "Estás comenzando tu aventura.",
    "xp_required": 0,
    "image_url": "https://..."
  }
]
```

Errores: 401 no autenticado

Tours

GET `/api/tours/`

Lista todos los tours activos con el estado del usuario autenticado en cada uno (progreso, si lo inició, si lo completó) e incluye los spots de cada tour.

Auth requerida: Sí

Respuesta 200:

```
[
  {
    "id": 1,
    "name": "Tour Centro Histórico",
    "description": "Recorrido por los principales monumentos...",
    "tag": "historia",
    "active": true,
    "user_tours_count": 150,
    "user_tour": {
      "progress": 2,
      "completed_at": null
    },
    "spots": [...]
  }
]
```

Errores: 401 no autenticado

GET /api/tours/<tour_id>/

Obtiene el detalle de un tour específico con los spots y el progreso del usuario.

Auth requerida: Sí

Parámetros de ruta: tour_id (integer)

Respuesta 200: Mismo formato que un elemento de GET /api/tours/

Errores: 404 tour no encontrado · 401 no autenticado

POST /api/tours/start/

Marca un tour como iniciado por el usuario (crea el registro UserTour).

Auth requerida: Sí

Body:

```
{
  "tour_id": 1
}
```

Respuesta 200:

```
{
  "message": "Tour iniciado correctamente"
}
```

Errores: 400 datos inválidos · 401 no autenticado

POST /api/tours/<tour_id>/spots/<spot_id>/complete

Marca un spot como completado dentro de un tour. Requiere enviar una imagen del lugar que es verificada con IA y las coordenadas GPS del usuario.

Auth requerida: Sí

Content-Type: multipart/form-data

Parámetros de ruta: tour_id (integer) · spot_id (integer)

Body (form-data):

Campo	Tipo	Requerido	Descripción
file	File	Sí	Imagen del spot tomada por el usuario
latitude	String	Sí	Latitud actual del usuario
longitude	String	Sí	Longitud actual del usuario

Respuesta 200 (spot intermedio):

```
{
  "detail": "Spot completado correctamente.",
}
```

```

    "progress": 2,
    "tour_completed": false
  }

```

Respuesta 200 (último spot — tour completado):

```

{
  "detail": "¡Tour completado! Has recibido recompensas.",
  "progress": 4,
  "tour_completed": true,
  "rewards": {
    "experience": 2000,
    "coins": 800
  },
  "quizzes_completed": 3,
  "secret_items_completed": 1,
  "total_quizzes": 4,
  "total_secret_items": 2,
  "tour_name": "Tour Centro Histórico"
}

```

Errores: 400 no se envió archivo / imagen no corresponde al lugar / sin coordenadas / fuera de rango · 404 tour o spot no encontrado · 401 no autenticado

Nota: La verificación por IA y por ubicación puede estar desactivada según configuración del entorno (`AI_VERIFICATION_ENABLED` , `LOCATION_VERIFICATION_ENABLED`).

POST `/api/tours/<tour_id>/feedback/`

Permite al usuario dejar un feedback sobre un tour.

Auth requerida: Sí

Parámetros de ruta: `tour_id` (integer)

Body:

```

{
  "name": "Mi experiencia",
  "qualification": 5,
  "trivia_liked": true,
  "spots_liked": true,
  "secrets_liked": false,
  "route_liked": true,
  "rewards_liked": true,
  "comment": "Excelente recorrido, muy entretenido."
}

```

Respuesta 200:

```
{
  "message": "Feedback creado correctamente"
}
```

Errores: 400 datos inválidos · 404 tour no encontrado · 401 no autenticado

Spots

GET `/api/spots/<spot_id>/`

Obtiene la información completa de un lugar específico, incluyendo sus quizzes y objetos secretos disponibles.

Auth requerida: Sí

Parámetros de ruta: `spot_id` (integer)

Respuesta 200:

```
{
  "id": 1,
  "name": "Plaza Mayor",
  "slug": "plaza-mayor",
  "address": "Av. Principal 100",
  "latitude": -34.6037,
  "longitude": -58.3816,
  "description": "La plaza central de la ciudad...",
  "historical_information": "Fundada en 1580...",
  "fun_facts": "Aquí ocurrió...",
  "schedule": "Abierto las 24hs",
  "ticket_price": null,
  "wheelchair_accessible": true,
  "tag": "historia",
  "activated": true,
  "image_urls": ["https://...", "https://..."],
  "ranking": "4.5",
  "quizzes": [...],
  "secret_items": [...]
}
```

Errores: 404 lugar no encontrado · 401 no autenticado

Objetos Secretos

GET `/api/secret_items/`

Lista todos los objetos secretos activos e indica si el usuario autenticado ya los obtuvo.

Auth requerida: Sí

Respuesta 200:

```
[
  {
    "id": 1,
    "name": "Medalla Antigua",
    "description": "Una medalla del siglo XVIII...",
    "image_url": "https://...",
    "hint": "Buscá en la base de la fuente...",
    "obtained": false
  }
]
```

Errores: 401 no autenticado

POST /api/spots/<spot_id>/secret_items/<secret_item_id>/

Sube una imagen para descubrir un objeto secreto en un spot. La imagen es verificada por IA.

Auth requerida: Sí

Content-Type: multipart/form-data

Parámetros de ruta: spot_id (integer) · secret_item_id (integer)

Body (form-data):

Campo	Tipo	Requerido	Descripción
image	File	Sí	Imagen del objeto secreto tomada por el usuario

Respuesta 201 (objeto obtenido por primera vez):

```
{
  "image": "https://...",
  "name": "Medalla Antigua",
  "description": "Una medalla del siglo XVIII...",
  "success": true,
  "already_obtained": false,
  "coins": 100,
  "xp": 200
}
```

Respuesta 200 (ya obtenido anteriormente):

```
{
  "image": "https://...",
  "name": "Medalla Antigua",
  "description": "Una medalla del siglo XVIII...",
  "success": true,
  "already_obtained": true,
  "coins": 0,
}
```

```
"xp": 0
}
```

Errores: 400 sin imagen / imagen no coincide con el objeto · 401 no autenticado · 404 spot o secret item no encontrado

Trivia (Quizzes)

GET `/api/quizzes/<quiz_id>/`

Obtiene la información de un quiz con todas sus opciones de respuesta.

Auth requerida: Sí

Parámetros de ruta: `quiz_id` (integer)

Respuesta 200:

```
{
  "id": 1,
  "name": "¿En qué año fue fundada la plaza?",
  "explanation": "La plaza fue fundada en 1580 por...",
  "activated": true,
  "spot": 1,
  "answers": [
    { "id": 1, "name": "1580", "is_correct": true },
    { "id": 2, "name": "1810", "is_correct": false },
    { "id": 3, "name": "1750", "is_correct": false },
    { "id": 4, "name": "1900", "is_correct": false }
  ]
}
```

Errores: 404 quiz no encontrado · 401 no autenticado

POST `/api/quizzes/<quiz_id>/answer/<answer_id>/`

Verifica si la respuesta elegida es correcta, registra el intento y otorga recompensas si acertó.

Auth requerida: Sí

Parámetros de ruta: `quiz_id` (integer) · `answer_id` (integer)

Body: No requiere body.

Respuesta 200:

```
{
  "id": 2,
  "quiz": 1,
  "name": "1810",
  "is_correct": false,
  "success": false,
  "correct_answer_id": 1,
}
```

```

    "explanation": "La plaza fue fundada en 1580 por...",
    "rewards": {
      "experience": 50,
      "coins": 100
    }
  }
}

```

Nota: Las recompensas (`experience` y `coins`) solo se suman al usuario si `success` es `true` . También se actualiza el progreso en achievements de tipo `"trivia"` .

Errores: `404` quiz o respuesta no encontrada · `401` no autenticado

POST `/api/quizzes/<quiz_id>/powerup/5050/`

Usa el powerup 50/50: elimina 2 respuestas incorrectas aleatorias del quiz. Cuesta 400 monedas.

Auth requerida: Sí

Parámetros de ruta: `quiz_id` (integer)

Body: No requiere body.

Respuesta 200:

```

{
  "remaining_answers": [
    { "id": 1, "name": "1580", "is_correct": true },
    { "id": 3, "name": "1750", "is_correct": false }
  ],
  "coins_remaining": 600,
  "coins_spent": 400
}

```

Errores: `400` monedas insuficientes / quiz sin suficientes respuestas incorrectas · `404` quiz no encontrado · `401` no autenticado

Achievements

GET `/api/achievements/`

Lista todos los achievements activos con el progreso actual del usuario autenticado.

Auth requerida: Sí

Respuesta 200:

```

[
  {
    "id": 1,
    "achievement": {
      "id": 1,
      "name": "Primer Tour",
      "description": "Completá tu primer tour.",

```

```

    "goal": 1,
    "goal_type": "tours",
    "multiplier": 1.0,
    "image_url": "https://...",
    "is_active": true
  },
  "progress": 1,
  "completed_at": "2024-01-15T12:00:00Z"
}
]

```

Errores: 401 no autenticado

POST `/api/achievements/redeem/<achievement_id>/`

Canjea un achievement completado por recompensas (XP y gemas).

Auth requerida: Sí

Parámetros de ruta: `achievement_id` (integer)

Body: No requiere body.

Respuesta 200:

```

{
  "completed": true,
  "rewards": {
    "experience": 300,
    "gems": 5.0
  }
}

```

Respuesta 400:

```

{
  "completed": false
}

```

Errores: 400 achievement no disponible para canjear · 401 no autenticado

Ranking

GET `/api/ranking/`

Obtiene la lista de usuarios ordenados por experiencia (de mayor a menor). Soporta filtrado por nivel.

Auth requerida: Sí

Query params:

Parámetro	Tipo	Descripción
<code>level</code>	integer	(Opcional) Filtra el ranking por ID de nivel

Respuesta 200:

```
[
  {
    "id": 3,
    "username": "aventurero99",
    "display_name": "El Aventurero",
    "experience": 5000,
    "level": { "id": 3, "name": "Experto", "xp_required": 4000 }
  }
]
```

Errores: `401` no autenticado

Health Checks

Endpoints de estado del sistema disponibles en la raíz (sin prefijo `/api/`). No requieren autenticación.

GET `/health/`

Verifica el estado general del sistema (base de datos, configuración, etc.).

Auth requerida: No

Respuesta 200: Estado general del sistema.

GET `/health/ready/`

Verifica si la aplicación está lista para recibir tráfico (readiness check). Usado por load balancers y orquestadores.

Auth requerida: No

GET `/health/live/`

Verifica si la aplicación está viva (liveness check). Confirma que el proceso está corriendo correctamente.

Auth requerida: No

Documentación interactiva

La API expone documentación automática generada con Swagger/OpenAPI:

URL	Descripción
<code>/swagger/</code>	UI interactiva de Swagger
<code>/redoc/</code>	UI de ReDoc (solo lectura)
<code>/swagger.json</code>	Schema en formato JSON

URL	Descripción
<code>/swagger.yaml</code>	Schema en formato YAML