

**Metro Cultural**

**Integrantes: Matias Millaqueo - Felipe Gallardo**

**Docente: Fernando Gonzalo Herrera Francesconi**

**Sección: 706v**

**Índice**

[**Introducción 2**](#_heading=h.2p3anhd830f)

[**Alcance del proyecto 3**](#_heading=h.4u0bndd7yin1)

[**Justificación y relevancia 4**](#_heading=h.ddo7t75hru84)

[**Metodología 5**](#_heading=h.5n17id3tnukc)

[**Desarrollo 6**](#_heading=h.l302pxy1qwr7)

[**Evidencias 7**](#_heading=h.og3bl65gwxg3)

[Sprint Backlog (Trello) 8](#_heading=h.kpgqyy65jhyp)

[Mockups (Figma) 8](#_heading=h.3mx0qluolyz6)

[Vistas del Front end 9](#_heading=h.69f9jnv84dab)

[Ajustes de requisitos funcionales 9](#_heading=h.y8cambi4rg86)

[Diagrama BD: 10](#_heading=h.4i2c2oimvwmg)

[Diagramas 4+1: 11](#_heading=h.618mha9ztt78)

[Diagrama caso de uso 11](#_heading=h.49o6745v37gx)

[Diagrama de clases: 12](#_heading=h.q8izc7ghz5cf)

[Diagrama de comunicación 13](#_heading=h.9mex94vmmtgf)

[Diagrama de secuencias 13](#_heading=h.yim3nmlduumt)

[Diagrama de componentes 14](#_heading=h.wv1r29rwz4nw)

[Diagrama de paquetes 15](#_heading=h.emosx2iz8060)

[Diagrama de actividad 16](#_heading=h.tht518b1a83o)

[Diagrama de despliegue 17](#_heading=h.h33owblaqzi0)

[Repositorio del código de front end en GitHub: 18](#_heading=h.qj9k00o2z0vn)

[**Resultados 18**](#_heading=h.heyaoc8qtvip)

# Introducción

El proyecto Metro Cultural surge como una propuesta tecnológica orientada a mejorar el acceso a la información cultural de la ciudad de Santiago. Actualmente, los eventos, museos, ferias y actividades culturales se encuentran dispersos en múltiples fuentes, lo que dificulta que usuarios y turistas puedan conocer, comparar y asistir a estas iniciativas dentro de su rutina diaria.

Considerando que el Metro de Santiago es uno de los principales medios de transporte urbano y conecta gran parte de la ciudad, se identifica una oportunidad para centralizar y acercar la oferta cultural a las personas mediante una aplicación móvil que relacione cada estación con los puntos de interés cultural más cercanos.

El presente informe describe el desarrollo del proyecto,abordando su propósito, alcance, metodología aplicada, arquitectura propuesta, proceso de diseño, y los resultados obtenidos durante el desarrollo

# Alcance del proyecto

El proyecto contempla el diseño y desarrollo de una aplicación móvil que permita a los usuarios consultar información cultural cercana a las estaciones del Metro de Santiago, integrando geolocalización, visualización de eventos, y acceso a datos actualizados mediante APIs externas o base de datos propia.

Funcionalidades Incluidas:

Visualización de actividades culturales cercanas a estaciones del metro.

Geolocalización básica asociada a estaciones.

Diseño UI/UX en Figma de las principales pantallas.

Desarrollo inicial del Front-End en HTML, CSS y JavaScript.

Integración de base de datos en Supabase para almacenamiento de eventos.

Fuera de Alcance:

Integración total con APIs gubernamentales de cultura.

Multilenguaje.

# Justificación y relevancia

El proyecto Metro Cultural se justifica por la falta de una plataforma centralizada que entregue información cultural cercana a las estaciones del Metro de Santiago. Actualmente la información se encuentra dispersa, lo que dificulta que los usuarios conozcan actividades, eventos y espacios culturales durante sus trayectos.

Su relevancia radica en que facilita el acceso a la cultura, mejora la experiencia del usuario en el transporte público y promueve la participación ciudadana.

# Metodología

La metodología utilizada para este proyecto será **Scrum**, un marco ágil para la gestión de proyectos de desarrollo de software. Scrum es adecuado para proyectos de software con entregas iterativas e incrementales, donde los requisitos pueden cambiar durante el desarrollo, y permite una mejor adaptación a las necesidades del usuario. Dado que se trata de un proyecto de desarrollo de una aplicación móvil y web, que involucra varias fases y depende de la retroalimentación del usuario, Scrum es ideal para este tipo de entorno.

# Desarrollo

El proyecto APT se ejecutó utilizando la **metodología ágil Scrum**, organizando el trabajo en sprints y realizando seguimiento mediante Trello. El objetivo principal fue desarrollar una aplicación móvil que permitiera a los usuarios visualizar actividades culturales cercanas a estaciones del Metro de Santiago. Las principales etapas y actividades del desarrollo fueron las siguientes:

Planificación y organización del Sprint 1: Se definieron tareas, responsables, criterios de éxito y tiempos de ejecución, lo que permitió estructurar el avance inicial.

Diseño UI/UX en Figma: Se elaboraron mockups de las vistas principales (inicio, búsqueda por estación, listado de eventos y detalle de evento), priorizando la usabilidad, accesibilidad y claridad visual.

Desarrollo Front-End: Se inició el desarrollo de la interfaz utilizando HTML5, CSS y JavaScript, adaptada a formato móvil. Las pantallas fueron implementadas en base a los prototipos aprobados en Figma.

Implementación del Back-End: Se definió la base para la lógica del sistema utilizando Node.js, permitiendo estructurar la gestión de servicios y flujos de información.

Configuración de Base de Datos: Se integró Supabase como motor de base de datos para el almacenamiento de información cultural asociada a estaciones.

Modelado y documentación técnica: Se desarrollaron los artefactos del modelo 4+1, incluyendo diagramas de casos de uso, actividades, componentes, despliegue y vista lógica, permitiendo alinear la arquitectura con los objetivos funcionales del sistema.

# Evidencias

| **Evidencias** |
| --- |
| Sprint Backlog (Trello) Se organizó el trabajo del proyecto siguiendo la metodología ágil Scrum. En él se pueden observar las tareas priorizadas, su estado de avance y la columna correspondiente al flujo de trabajo: Por hacer, En progreso, En revisión y Completado.   Mockups (Figma) Se presentan los mockups desarrollados en Figma, los cuales representan la interfaz gráfica propuesta para la aplicación móvil Metro Cultural. Estas pantallas muestran el flujo principal del usuario, desde la visualización del logo y pantalla inicial, hasta el acceso a líneas del metro, listado de estaciones y detalle de actividades culturales.   Vistas del Front end Se evidencia el avance en el desarrollo del Front-End de la aplicación implementado, utilizando HTML5, CSS y JavaScript. Se visualizan las pantallas principales, tales como el formulario de registro y login, junto a la vista de navegación por líneas del Metro.  Además, se incluye la captura del entorno de desarrollo Visual Studio Code, donde se aprecia la estructura del código fuente implementado, validando el proceso de construcción de la interfaz definida previamente en los prototipos de Figma.   Ajustes de requisitos funcionales La siguiente evidencia corresponde a la matriz de ajustes de Requerimientos Funcionales, la cual recopila los cambios aplicados durante el sprint. La tabla muestra el detalle de cada requerimiento, incluyendo su código, descripción, tipo, prioridad, estado y responsables.  Este registro permitió mantener el control sobre el alcance del proyecto y tomar decisiones informadas frente a las restricciones del sprint.   Diagrama BD: La evidencia corresponde al Diagrama entidad–relación utilizado para modelar los datos de la aplicación Metro Cultural. En él se definen las entidades principales del sistema, cómo Estación, Evento Cultural, Cliente y Favoritos, junto con sus atributos y relaciones.   Diagramas 4+1:Diagrama caso de uso El siguiente diagrama muestra las funcionalidades principales que cada tipo de usuario puede realizar dentro del sistema. Diagrama de clases: Este diagrama muestra la estructura de los datos que usa la aplicación. Incluye las clases Cliente, Estación, Lugar Cultural, Evento, Favorito y Notificación, con sus principales atributos y relaciones.  Sirve para organizar la información y definir cómo se conecta cada parte del sistema, ayudando al diseño de la base de datos.   Diagrama de comunicación Este diagrama muestra la interacción y el flujo de mensajes entre un actor (el Cliente) y los diferentes componentes del sistema. Incluye al actor Cliente y los objetos Página Principal, Iniciar sesión o creación de cuenta, Visualizar mapa, Ver lugares culturales y Ver eventos, junto con la secuencia numerada de acciones que se envían entre ellos. Diagrama de secuencias Este diagrama muestra el orden cronológico de las interacciones y mensajes que se intercambian entre los participantes Cliente, Sistema y Base de datos. Incluye cuatro escenarios distintos: "Creación cuenta", "Iniciar sesión", "Seleccionar líneas/estación" y "Ver/Editar perfil", detallando el flujo de acciones y respuestas para cada uno. Diagrama de componentes Este diagrama muestra la arquitectura del sistema, dividiéndola en módulos lógicos y físicos. Incluye las capas principales: Capa de negocio (con las vistas de administración), Lógica de negocio (con los componentes para gestionar usuarios, líneas, estaciones y eventos), Bddd (PostgreSQL) y Servicios (React, Supabase, Google maps, Postman). Diagrama de paquetes Este diagrama muestra la organización de alto nivel del sistema, agrupando componentes y clases relacionadas en "paquetes" o módulos. Incluye los paquetes principales: Autenticación, Infraestructura, Estaciones y lugares, Perfil, Eventos, Favoritos y Administrador, mostrando las dependencias (flechas) entre ellos. Diagrama de actividad General:  Este diagrama muestra el flujo de trabajo (workflow) del sistema, detallando la secuencia de acciones desde un punto de "Inicio" hasta un "Fin". El flujo está organizado en "calles" (swimlanes) que separan las responsabilidades de los participantes: el Cliente, la App web y la Base de datos.    Registro de cliente:  Este diagrama muestra el flujo de trabajo (workflow) específico para el proceso de registro e inicio de sesión del usuario. El flujo está organizado en "calles" (swimlanes) que separan las responsabilidades del Cliente y del sistema Metro Cultural Sys.   Diagrama de despliegue Este diagrama muestra la arquitectura física del sistema, ilustrando cómo los componentes de hardware (nodos) y software (artefactos) se distribuyen y conectan. Incluye los nodos como el Pc Cliente, Internet, el Sitio web (Frontend), el Sitio web (Backend), Supabase y un Servidor adquirido.   Repositorio del código de front end en GitHub: |

# 

# 

# 

# Resultados

El desarrollo del proyecto Metro Cultural permitió obtener un Producto Mínimo Viable (MVP) funcional, que integra los elementos esenciales de la solución propuesta: el diseño de la experiencia de usuario, la estructura técnica base y la visualización inicial de información cultural asociada a estaciones del Metro de Santiago. A pesar de ajustes en el alcance, se logró cumplir con los objetivos principales del sprint, evidenciando avances concretos tanto a nivel técnico como metodológico.