



**Universidad Nacional Autónoma de
México**

Facultad de Ingeniería

División de Ingeniería Eléctrica



Cómputo Móvil

Semestre: 2023-1

Grupo: 03

Profesor: *Ing. Marduk Pérez de Lara Domínguez*

Propuesta app: Metrofy. Análisis técnico

Equipo 2

Martínez Juárez, Abigail Meztli

Mateos Flores, Erik Esteban

Fecha de Entrega: 6 de enero de 2023



Contenido

Introducción.....	2
Wireframe	3
<i>Mapa de sitio</i>	<i>3</i>
<i>Wireframes.....</i>	<i>3</i>
Análisis por pantallas.....	7
Lenguajes de programación y servicios	17
Metodología y equipo de trabajo	17
Android Developer	18
Backend Developer	18
UI/UX Designer	18
Quality Assurance Engineer	18
Project Manager	18
Estimación de costos	18
Demo de la aplicación	20
Conclusiones y reflexiones.....	22
Bibliografía.....	23



Introducción

El presente trabajo es una continuación del análisis para el desarrollo de la aplicación móvil llamada Metrofy, esta aplicación es una aplicación móvil nativa para dispositivos Android y tiene como objetivo servir como herramienta para facilitar la forma en la que los usuarios viajan a través del sistema de transporte colectivo Metro.

Como anteriormente se había mencionado, la razón por la cual se decidió trabajar con el sistema de transporte colectivo Metro es debido a que este medio de transporte es utilizado diariamente por todos los integrantes del equipo y al discutir y analizar cómo funciona este medio de transporte, así como los problemas principales que como usuarios teníamos en común, encontramos un área de oportunidad al querer modernizar y optimizar la forma en la que miles de mexicanos se trasladan diariamente, pues si bien, es cierto que existen muchas aplicaciones con objetivos y funcionalidades similares a lo que proponemos, las cuales fueron analizadas en la primera parte del trabajo, también es cierto que hay muchas cosas que aún se pueden mejorar y con nuestra propuesta pretendemos es lo que pretendemos lograr con, por ejemplo, nuestra funcionalidad principal la cual consiste en el monitoreo en tiempo real de los trenes lo cual mejoraría funcionalidades implementadas en otras aplicaciones como lo es el cálculo de rutas pues este cálculo ahora nosotros lo proponemos dinámico en base a la información del monitoreo; además de la funcionalidad anterior, y aunque el tema de las funcionalidades fue tratado en la primera parte del trabajo, es importante destacar que a uno de los problemas a los que también pretendemos darle solución es la desinformación pues a pesar de que existen cuentas oficiales del metro y que ahí se publica información muchas veces esta información o avisos llegan tarde o no llegan, lo que por ende da lugar a que los mismos usuarios se manifiesten en redes sociales dando avisos, sin embargo, esto puede dar lugar a la desinformación y es por ello que planteamos las opciones de comunidad y avisos oficiales, en donde, en la primera los usuarios de la aplicación podrán dar sus reportes de problemas, retrasos o cualquier inconveniente o aviso que sea de ayuda para los demás usuarios, los cuales se pueden consultar por línea del metro para poder corroborar que esta información sea verídica, y en la segunda opción, se mostrarán avisos oficiales del metro para que los usuarios puedan consultarlos de manera fácil y además puedan corroborar los datos publicados por los usuarios.

Si bien en la primera parte del trabajo hay algunos puntos que ya habíamos desarrollado esta segunda parte pretende hacer un análisis más profundo acerca de las herramientas, lenguajes de programación y servicios a utilizar para el desarrollo de la aplicación con un análisis por pantalla, así también, volvemos a analizar cuál es el equipo de trabajo con el que debemos de contar, el tiempo de elaboración y la estimación de los costos del proyecto.

Por último, en este nuevo análisis nos adentramos en la parte de diseño al realizar los wireframes y el demo de la aplicación.

Wireframe

Mapa de sitio

Para comenzar, tenemos a continuación (Imagen 1) el mapa de sitio de nuestra aplicación el cuál nos ayuda a entender de forma más clara cómo se dan las interacciones dentro de la aplicación y cuál es el flujo de esta.

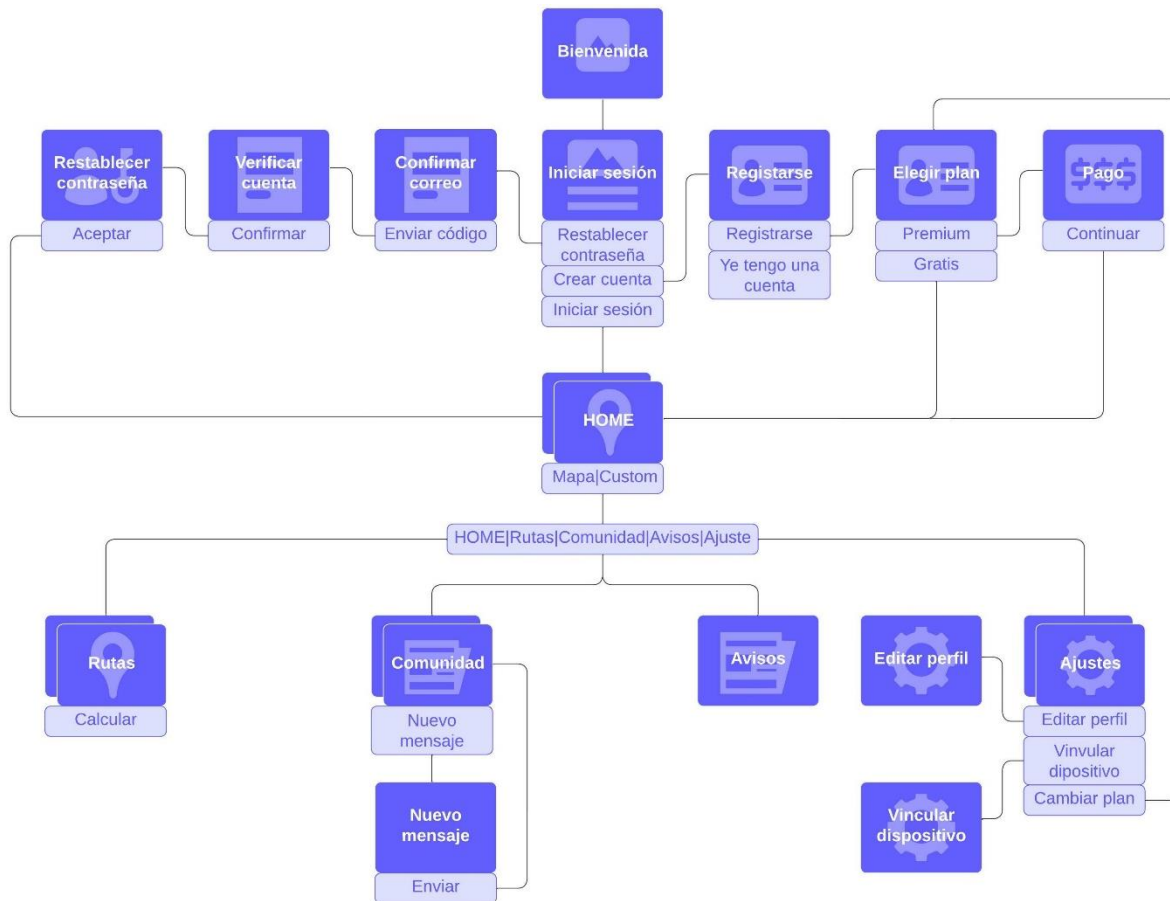


Imagen 1. Mapa de sitio Metrofy

Wireframes

La herramienta que utilizamos para el diseño de los wireframes fue *Balsamiq Cloud* ya que es una herramienta gratuita, nos ofrece varios componentes y su uso es bastante intuitivo.

A continuación, se agregan las pantallas de los wireframes y mediante el siguiente link se puede consultar el proyecto para una mejor visualización del diseño: <https://balsamiq.cloud/snscho5/pf531w>

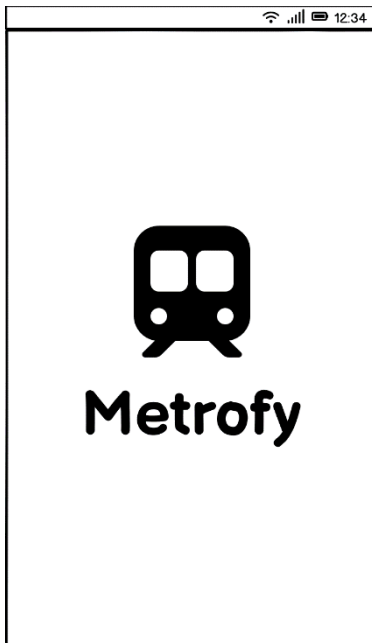


Imagen 2. Bienvenida



Imagen 3. Inicio de sesión



Imagen 4. Confirmar correo



Imagen 5. Verificar cuenta



Imagen 6. Restablecer contraseña

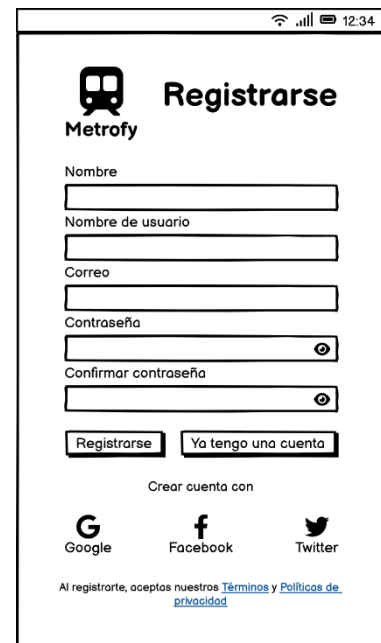


Imagen 7. Registro

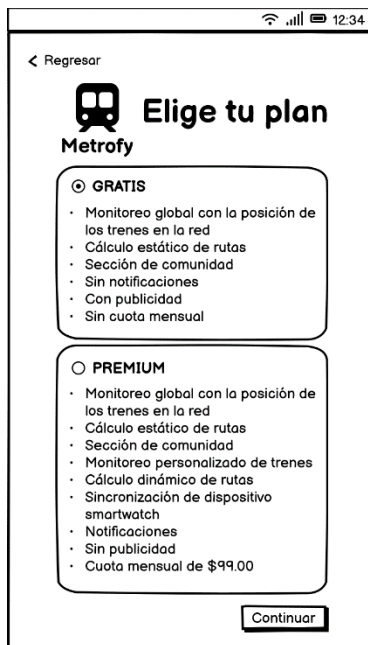


Imagen 8. Elección de plan

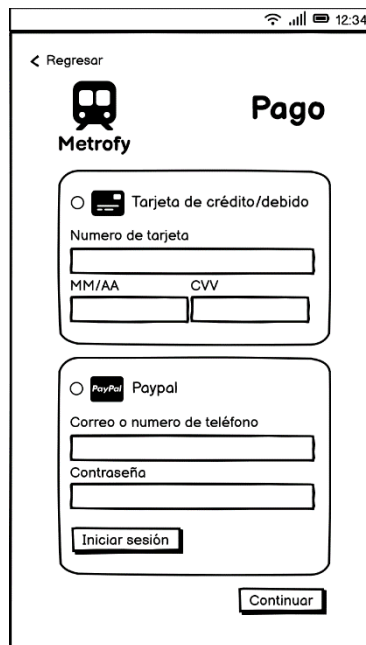


Imagen 9. Pago

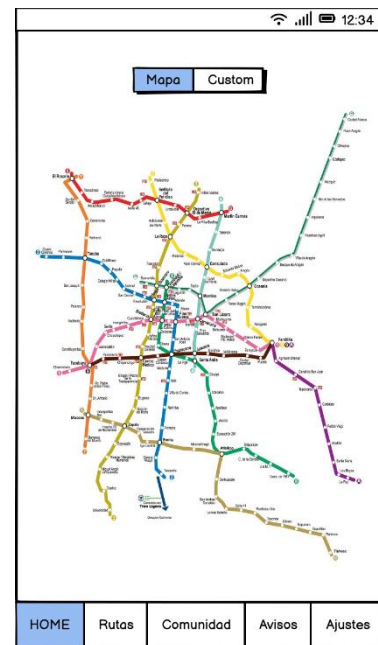


Imagen 10. HOME (mapa)

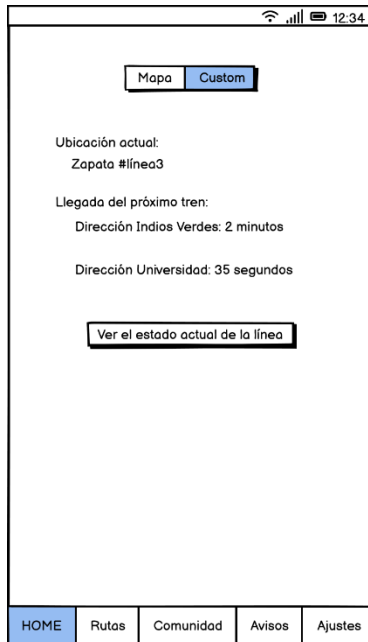


Imagen 11. HOME (Custom)



Imagen 12. HOME (Estado línea)

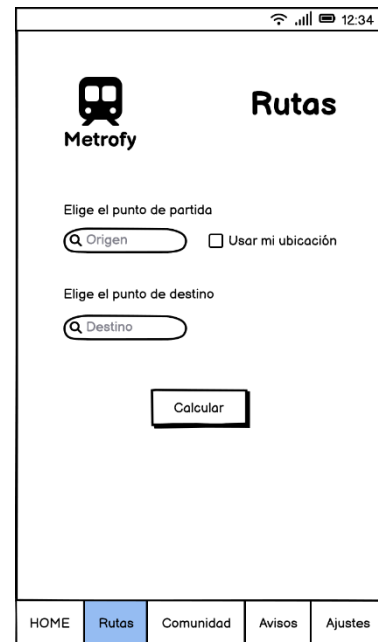


Imagen 13. Rutas

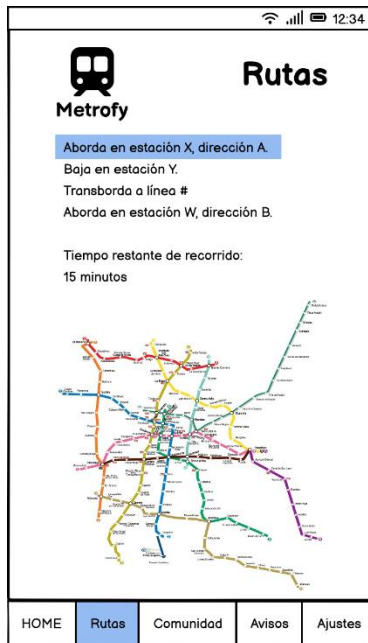


Imagen 14. Rutas (indicaciones)



Imagen 15. Comunidad

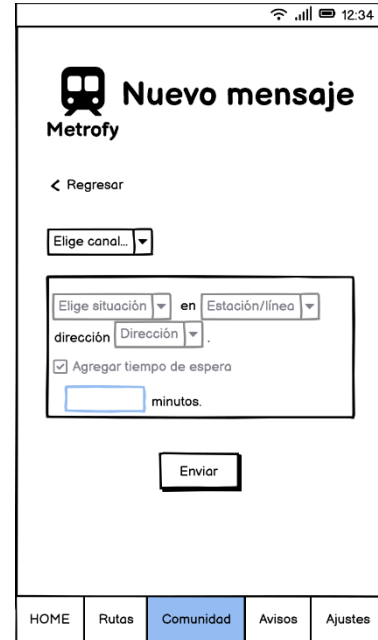


Imagen 16. Comunidad (nuevo mensaje)

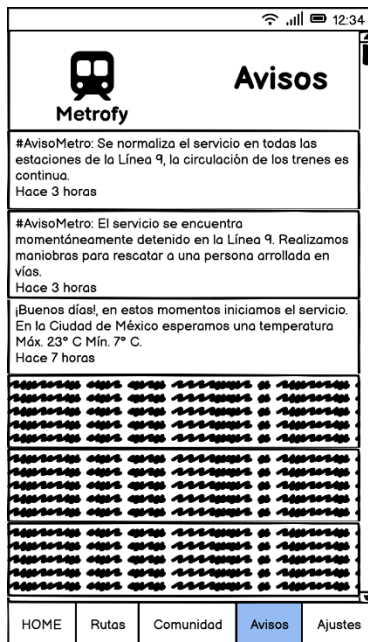


Imagen 17. Avisos

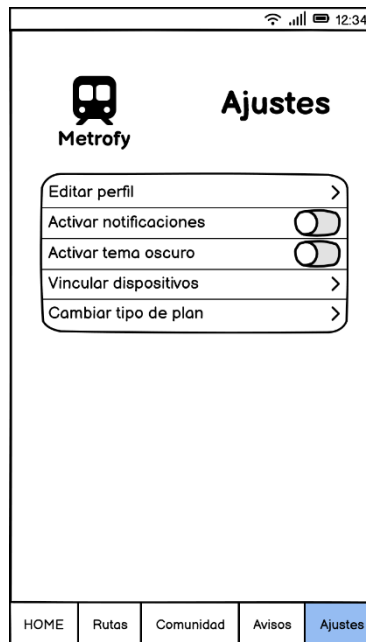


Imagen 18. Ajustes

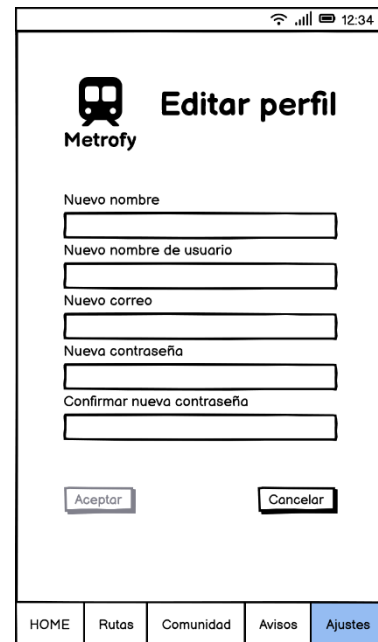


Imagen 19. Editar perfil

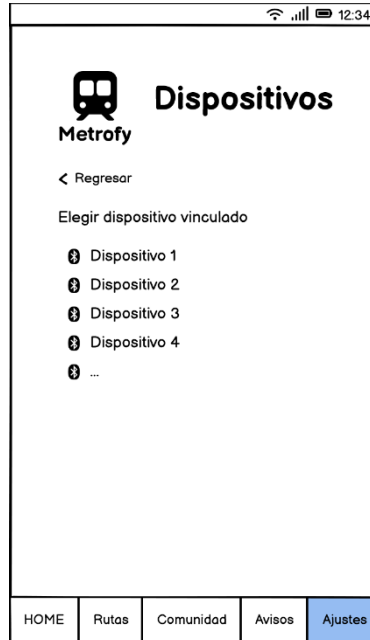


Imagen 20. Vincular dispositivo

Análisis por pantallas

A continuación, se hace el análisis para cada una de las pantallas de la aplicación.

BIENVENIDA			
Funcionalidades			
Nombre		Descripción	
Bienvenida		Despliega el logo de la aplicación al iniciarla	
Análisis de datos			
Nombre		Tipo de dato	CRUD tipo
logo		image	read
Servicios/API			
Nombre		Característica	Requerimientos
Ninguno			
Tipo de almacenamiento			
Tipo	Dato	Descripción	
Local	logo	Requerimiento para la carga de la aplicación	
INICIO DE SESIÓN			
Funcionalidades			
Nombre		Descripción	
Inicio de sesión		Le permite al usuario iniciar sesión con una cuenta previamente existente en la base de datos	
Análisis de datos			
Nombre		Tipo de dato	CRUD tipo
logo		Image	read
iniciar_sesion_label		String	read
usuario_label		string	read



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO
FACULTAD DE INGENIERÍA
DIVISIÓN DE INGENIERÍA ELÉCTRICA



usuario_input	string	read
psw_label	string	read
psw_input	string	read
usuario_recordar_label	string	read
usuario_recordar_input	boolean	update
restore_psw_label	string	read
google_icon	image	read
facebook_icon	image	read
twitter_icon	image	read

Servicios/API

Nombre	Característica	Requerimientos
Google Cloud Platform	Almacena los datos de los usuarios registrados	Usuario y contraseña
API de Google	Iniciar sesión con Google	OAuth 2.0 y contar con una cuenta de Google
SDK de inicio de sesión de Facebook	Iniciar sesión con Facebook	Tener instalada la aplicación de Facebook y contar con una cuenta
API de Twitter	Iniciar sesión con Twitter	OAuth 2.0 y contar con una cuenta de Twitter

Tipo de almacenamiento

Tipo	Dato	Descripción
Local	Imágenes y etiquetas	Requerimiento para el correcto despliegue y visualización de la aplicación
Nube	Usuario	Consulta en la base de datos que el usuario exista

CONFIRMAR CORREO

Funcionalidades

Nombre	Descripción
Verificar correo electrónico	Envía por correo un código de verificación para poder comprobar su identidad siempre y cuando el usuario exista en la base de datos.

Análisis de datos

Nombre	Tipo de dato	CRUD tipo
logo	Image	read
confirmar_correo_label	string	read
descripcion_label	string	read
correo_label	string	read
correo_input	string	read
confirmar_correo_label	string	read
confirmar_correo_input	string	read

Servicios/API

Nombre	Característica	Requerimientos
Google Cloud Platform	Consulta los datos (que el correo exista en la base de datos)	Correo y cuenta existente

Tipo de almacenamiento

Tipo	Dato	Descripción
Local	Imágenes y etiquetas	Requerimiento para el correcto despliegue y visualización de la aplicación
Nube	Usuarios	Consulta de los datos de los usuarios



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO
FACULTAD DE INGENIERÍA
DIVISIÓN DE INGENIERÍA ELÉCTRICA



VERIFICAR CUENTA		
Funcionalidades		
Nombre	Descripción	
Verificar correo electrónico	Envía al correo del usuario un código de verificación para poder comprobar su identidad siempre y cuando el usuario exista en la base de datos.	
Análisis de datos		
Nombre	Tipo de dato	CRUD tipo
logo	Image	read
verificar_cuenta_label	string	read
descripcion_verificar_label	string	read
ingresar_codigo_label	string	read
ingresar_codigo_input	string	read
Servicios/API		
Nombre	Característica	Requerimientos
Google Cloud Platform	Verifica que el código exista en la base de datos	Código de verificación
Tipo de almacenamiento		
Tipo	Dato	Descripción
Local	Imágenes y etiquetas	Requerimiento para el correcto despliegue y visualización de la aplicación
Nube	Código de verificación	Almacenamiento en la base de datos del código de verificación

RESTABLECER CONTRASEÑA		
Funcionalidades		
Nombre	Descripción	
Restablecer contraseña	Le permite al usuario restablecer la contraseña si olvidó la suya	
Análisis de datos		
Nombre	Tipo de dato	CRUD tipo
logo	Image	read
restablecer_psw_label	string	read
nueva_psw_label	string	read
nueva_psw_input	string	update
confirmar_psw_label	string	read
confirmar_psw_input	string	update
Servicios/API		
Nombre	Característica	Requerimientos
Google Cloud Platform	Actualiza el dato contraseña del usuario en la base de datos	Usuario, nueva contraseña
Tipo de almacenamiento		
Tipo	Dato	Descripción
Local	Imágenes y etiquetas	Requerimiento para el correcto despliegue y visualización de la aplicación
Nube	Usuario	Almacenamiento del nuevo valor del dato contraseña

REGISTRARSE		
Funcionalidades		



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO
FACULTAD DE INGENIERÍA
DIVISIÓN DE INGENIERÍA ELÉCTRICA



Nombre	Descripción	
Registrar usuario	Almacena en la base de datos un nuevo usuario	
Análisis de datos		
Nombre	Tipo de dato	CRUD tipo
logo	Image	read
registrarse_label	string	read
nombre_label	string	read
nombre_input	string	create
nombre_usuario_label	string	read
nombre_usuario_input	string	create
correo_label	string	read
correo_input	string	create
psw_label	string	read
psw_input	string	create
confirmar_psw_label	string	read
confirmar_psw_input	string	create
crear_con_label	string	read
google_icon	image	create
facebook_icon	image	create
twitter_icon	image	create
terminos_condiciones_label	string	read
Servicios/API		
Nombre	Característica	Requerimientos
Google Cloud Platform	Creación de un nuevo usuario en la base de datos	Correo electrónico, cuenta de google, facebook o twitter
API de Google	Iniciar sesión con Google	OAuth 2.0 y contar con una cuenta de Google
SDK de inicio de sesión de Facebook	Iniciar sesión con Facebook	Tener instalada la aplicación de Facebook y contar con una cuenta
API de Twitter	Iniciar sesión con Twitter	OAuth 2.0 y contar con una cuenta de Twitter
Tipo de almacenamiento		
Tipo	Dato	Descripción
Local	Imágenes y etiquetas	Requerimiento para el correcto despliegue y visualización de la aplicación
Nube	Nuevo usuario	Almacena los datos del nuevo usuario en la base de datos
ELIGE TU PLAN		
Funcionalidades		
Nombre	Descripción	
Elegir plan	Crea el usuario con el plan que elija, para proporcionarle las funcionalidades dependiendo del tipo de cuenta	
Análisis de datos		
Nombre	Tipo de dato	CRUD tipo
logo	Image	read
elige_plan_label	string	read
gratis_input	boolean	update
gratis_descripcion_label	string	read



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO
FACULTAD DE INGENIERÍA
DIVISIÓN DE INGENIERÍA ELÉCTRICA



premium_input	boolean	update
premium_descripcion_label	string	read
Servicios/API		
Nombre	Característica	Requerimientos
Google Cloud Platform	Actualiza el dato tipo de cuenta del usuario	
Tipo de almacenamiento		
Tipo	Dato	Descripción
Local	Imágenes y etiquetas	Requerimiento para el correcto despliegue y visualización de la aplicación
Nube	Usuario	Actualiza el dato tipo de cuenta para el usuario previamente creado

PAGO	
Funcionalidades	
Nombre	Descripción
Realizar pago	Realiza el pago de acuerdo con la forma que seleccione el usuario.

Análisis de datos		
Nombre	Tipo de dato	CRUD tipo
logo	image	read
pago_label	string	read
pago_tarjeta_input	boolean	create
numero_tarjeta_label	string	read
numero_tarjeta_input	string	read
mm_aa_label	string	read
mm_aa_input	string	read
cvv_label	string	read
cvv_input	string	read
paypal_input	boolean	create
correo_telefono_label	string	read
correo_telefono_input	string	read
psw_label	string	read
psw_input	string	read

Servicios/API		
Nombre	Característica	Requerimientos
Google Cloud Platform	Actualiza el dato pago para el usuario	
Braintree Android SDK	integra las formas de pago con tarjeta y paypal	
Tipo de almacenamiento		
Tipo	Dato	Descripción
Local	Imágenes y etiquetas	Requerimiento para el correcto despliegue y visualización de la aplicación
Nube	Dato del pago para el usuario	Almacena el dato de realización del pago para el usuario

HOME (MAPA)	
Funcionalidades	
Nombre	Descripción
Monitoreo de todos los trenes	Muestra en el mapa completo de la red del metro la ubicación de todos los trenes monitoreados en tiempo real.



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO
FACULTAD DE INGENIERÍA
DIVISIÓN DE INGENIERÍA ELÉCTRICA



Gestos especiales		
Gesto	Descripción	
zoom	Para hacer zoom en el mapa se debe de dar doble clic	
desplazamiento	Para desplazar el mapa se arrastra el mapa con un sólo dedo	
Análisis de datos		
Nombre	Tipo de dato	CRUD tipo
mapa	image	read
Servicios/API		
Nombre	Característica	Requerimientos
Google Maps Platform	Geolocalización	API Geolocation y Places SDK for Android
Tipo de almacenamiento		
Tipo	Dato	Descripción
Local	Imágenes	Requerimiento para el correcto despliegue y visualización de la aplicación
Nube	Datos de geolocalización	Almacena y actualiza los datos relacionados a la geolocalización del usuario y los trenes
HOME (CUSTOM)		
Funcionalidades		
Nombre	Descripción	
Próximo arribo	Despliega la información de los próximos trenes a arribar dependiendo de la línea en la que nos encontremos (premium)	
Análisis de datos		
Nombre	Tipo de dato	CRUD tipo
ubicacion_actual_label	string	read
ubicacion_actual_output	string	read
llegada_proxima_label	string	read
primera_direccion_output	string	read
segunda_direccion_output	string	read
tiempo_primera_output	string	read
tiempo_segunda_output	string	read
Servicios/API		
Nombre	Característica	Requerimientos
Google Maps Platform	Geolocalización	API Geolocation y Places SDK for Android
Tipo de almacenamiento		
Tipo	Dato	Descripción
Local	Etiquetas	Requerimiento para el correcto despliegue y visualización de la aplicación
Nube	Datos de geolocalización y cálculo de tiempo	Almacena y actualiza los datos relacionados a la geolocalización del usuario y los trenes, además del despliegue del tiempo de arribo
HOME (ESTADO DE LA LÍNEA)		
Funcionalidades		
Nombre	Descripción	
Visualizar línea	Permite que al usuario premium visualizar el monitoreo de los trenes para la línea del metro en donde se encuentra ubicado	
Análisis de datos		



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO
FACULTAD DE INGENIERÍA
DIVISIÓN DE INGENIERÍA ELÉCTRICA



Nombre		Tipo de dato	CRUD tipo
mapa_linea		image	read
Servicios/API			
Nombre		Característica	Requerimientos
Google Maps Platform		Geolocalización	API Geolocation y Places SDK for Android
Tipo de almacenamiento			
Tipo	Dato		Descripción
Local	Imágenes		Requerimiento para el correcto despliegue y visualización de la aplicación
Nube	Datos de geolocalización		Almacena y actualiza los datos relacionados a la geolocalización del usuario y los trenes

RUTAS	
Funcionalidades	
Nombre	Descripción
Cálculo de ruta	Permite al usuario ingresar un origen y destino dentro de la red para calcular una ruta de trayecto.

Análisis de datos			
Nombre		Tipo de dato	CRUD tipo
logo		image	read
rutas_logo		image	read
partida_label		string	read
destino_label		string	read
usar_location_label		string	read
partida_input		string	read
destino_input		string	read

Servicios/API			
Nombre		Característica	Requerimientos
Google Maps Platform		Realiza el cálculo de la mejor ruta	Origen y destino proporcionado por usuario
Tipo de almacenamiento			
Tipo	Dato		Descripción
Local	imágenes y etiquetas		Muestran las indicaciones para que el usuario sepa cómo ingresar los datos para el cálculo de ruta.

RUTAS (INDICACIONES)	
Funcionalidades	
Nombre	Descripción
Indicaciones	Muestra cada paso en las indicaciones para seguir una ruta calculada.
Tiempo de llegada	Muestra el tiempo aproximado de llegada al destino actualizado en tiempo real.
Mapa de ruta	Resalta en el mapa de la red la ruta a seguir, así como la ubicación actual.

Análisis de datos			
Nombre		Tipo de dato	CRUD tipo
logo		image	read
rutas_logo		image	read
ruta_indicacion		string	read



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO
FACULTAD DE INGENIERÍA
DIVISIÓN DE INGENIERÍA ELÉCTRICA



nombre_estacion	string	read
nombre_linea	string	read
nombre_direccion	string	read
tiempo_restante_label	string	read
tiempo_restante	int	read
Servicios/API		
Nombre	Característica	Requerimientos
Google Maps Platform	Muestra el estado actual del recorrido de la ruta.	Ubicación actual, destino y mapa de la red.
Tipo de almacenamiento		
Tipo	Dato	Descripción
Local	imágenes y etiquetas	Despliegan información de la app.
COMUNIDAD		
Funcionalidades		
Nombre	Descripción	
Selección de reportes	A partir de una lista desplegable, se pueden seleccionar la lista de novedades por línea o de la red completa del metro.	
Reportes	Muestra un listado de los reportes que realiza la comunidad sobre el estado (averías, accidentes, retrasos, buen avance) de cada línea.	
Análisis de datos		
Nombre	Tipo de dato	CRUD tipo
logo	image	read
comunidad_logo	image	read
filtrar_label	string	read
linea_nombre	string	read
usuario_nombre	string	read
mensaje_usuario	string	read
Servicios/API		
Nombre	Característica	Requerimientos
Google Cloud Platform	Obtiene los mensajes más recientes del estado de la red, así como el nombre del usuario que lo publica.	Nombre de usuario, mensaje enviado.
Tipo de almacenamiento		
Tipo	Dato	Descripción
Local	imágenes y etiquetas	Despliegan información de la app.
Nube	Mensajes y usuarios	Recupera la información necesaria para desplegar los mensajes del estado de cada línea.
COMUNIDAD (NUEVO MENSAJE)		
Funcionalidades		
Nombre	Descripción	
Selección de canal	De una lista desplegable, el usuario puede elegir la línea en que desea ingresar un mensaje sobre su estado.	
Nuevo mensaje	Usando una estructura predeterminada, el usuario puede redactar un mensaje que detalle el estado de una línea o estación en la red, eligiendo de una lista el estado de la línea.	
Enviar mensaje	Tras redactar el mensaje, el mensaje se puede subir con el botón Enviar	
Análisis de datos		



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO
FACULTAD DE INGENIERÍA
DIVISIÓN DE INGENIERÍA ELÉCTRICA



Nombre	Tipo de dato	CRUD tipo
logo	image	read
mensaje_logo	image	read
nombre_canal	string	read
situacion	string	read
nombre_estacion	string	read
nombre_linea	string	read
direccion	string	read
tiempo_espera_label	string	read
tiempo_espera_input	number	create

Servicios/API

Nombre	Característica	Requerimientos
Google Cloud Platform	Guarda los mensajes creados	Campos completos en mensaje predeterminado

Tipo de almacenamiento

Tipo	Dato	Descripción
Local	imágenes y etiquetas	Muestra una estructura que ayude al usuario a crear un mensaje de forma sencilla
Nube	Mensajes (string)	Tras crearse un nuevo mensaje, este se sube a la nube para consultas posteriores

AVISOS OFICIALES

Funcionalidades

Nombre	Descripción
Despliegue de avisos	Muestra avisos publicados desde fuentes oficiales del metro

Análisis de datos

Nombre	Tipo de dato	CRUD tipo
logo	image	read
avisos_logo	image	read
aviso_oficial	string	read

Servicios/API

Nombre	Característica	Requerimientos
Twitter API	Obtiene tweets desde la cuenta oficial del metro	Acceso a alguna cuenta oficial (twitter, por ejemplo)

Tipo de almacenamiento

Tipo	Dato	Descripción
Local	imágenes y etiquetas	Muestra información de la app
Nube	mensajes de avisos (string)	Recupera mensajes de cuentas oficiales

AJUSTES

Funcionalidades

Nombre	Descripción
Ajustes de app	Permite modificar el estado de la app (apariciencia y notificaciones) así como la información del usuario.

Análisis de datos

Nombre	Tipo de dato	CRUD tipo
logo	image	read



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO
FACULTAD DE INGENIERÍA
DIVISIÓN DE INGENIERÍA ELÉCTRICA



ajustes_log	image	read
editar_perfil_label	string	read
notificaciones_label	string	read
tema_label	string	read
vincular_label	string	read
plan_label	string	read

Servicios/API

Nombre	Característica	Requerimientos
--------	----------------	----------------

Ninguno

Tipo de almacenamiento

Tipo	Dato	Descripción
Local	imágenes y etiquetas	Despliega los detalles de las opciones a elegir en el menú de ajustes

EDITAR PERFIL

Funcionalidades

Nombre	Descripción
Editar usuario	Permite modificar el nombre, nombre de usuario, email y contraseña de la cuenta ya creada.

Análisis de datos

Nombre	Tipo de dato	CRUD tipo
logo	image	read
perfil_logo	image	read
nuevo_nombre_label	string	read
nuevo_usuario_label	string	read
nuevo_psw_label	string	read
confirmar_psw_label	string	read
nuevo_nombre_input	string	update
nuevo_usuario_input	string	update
nuevo_psw_input	string	update
confirmar_psw_input	string	update

Servicios/API

Nombre	Característica	Requerimientos
Google Cloud Platform	Ingresa y actualiza la información del usuario modificada	Llenar y confirmar al menos un campo editable

Tipo de almacenamiento

Tipo	Dato	Descripción
Local	imágenes y etiquetas	Muestra detalles de la app y campos para cambiar información de usuario.
Nube	nueva información de usuario (string)	Los campos modificados deben verse reflejados en la nube con una actualización directa.

VINCULAR DISPOSITIVO

Funcionalidades

Nombre	Descripción
Vincular	Permite vincular un dispositivo smartwatch vía bluetooth

Análisis de datos

Nombre	Tipo de dato	CRUD tipo
--------	--------------	-----------



logo	image	read
vincular_logo	image	read
elegir_dispositivo_label	string	read
dispositivo	string	read
Servicios/API		
Nombre	Característica	Requerimientos
Ninguno		
Tipo de almacenamiento		
Tipo	Dato	Descripción
Local	imágenes y etiquetas	Muestra detalles de la app, así como la lista de dispositivos bluetooth detectados por el sistema.

Lenguajes de programación y servicios

Metrofy inicialmente será una app que será lanzada exclusivamente para smartphones con Android, por lo que el lenguaje de programación principal a utilizar será Kotlin. Este lenguaje ofrece la facilidad de que su implementación en Java lo hace fácil de aprender y permite una codificación más clara y organizada.

La IDE por excelencia para desarrollar apps para este Sistema Operativo es Android Studio, la cual tiene integrada la operabilidad con Kotlin. Esta IDE, además de tener una interfaz gráfica amigable para el programador, ofrece los siguientes beneficios principales:

- Emulador con la vista previa de la(s) pantalla(s) que se están desarrollando. Estas vistas pueden manipularse tal y como se haría desde un dispositivo con Android, esto para hacer pruebas y depuraciones.
- Compatibilidad para el manejo de versiones con GitHub
- Compatibilidad con Google Cloud Services
- Selección de diversas plantillas según el modelo de negocio que se emplee.

Google Cloud Platform es el servicio principal para la implementación de nuestra app. Esta herramienta interactúa de forma directa con Android Studio, lo cual facilita en gran medida la implementación de tareas y actividades medulares para el correcto funcionamiento de la app. Entre sus principales tareas están:

- Acceso a servidores de almacenamiento y bases de datos.
- Buen desempeño para cargas de trabajo elevadas en tiempo real empleando la red.
- Seguridad para los datos y microservicios empleados.
- Sistemas para la prevención de pérdida de datos.
- Análisis de macrodatos y manejo optimizado de BLOBs empleando MySQL.

Metodología y equipo de trabajo

Tras realizar un análisis más detallado de las funciones principales que tendrá la app, se reitera que la metodología Scrum es la mejor opción para nuestro proyecto, además



de que actualmente esta metodología es una de las más empleadas y conocidas entre los desarrolladores. De igual forma, se confirma que el tiempo de desarrollo será de 6 a 8 meses, empleando un equipo de trabajo de 5 personas, cuyos perfiles se describen a continuación:

Android Developer

Especializado en el uso de Kotlin empleando Android Studio, además de tener experiencia previa en el desarrollo de apps en Android. Su tarea fundamental es crear la estructura principal del código del sistema y el diseño de la base de datos empleada, así como el manejo y corrección de errores.

Backend Developer

Encargado de la administración de un servidor funcional, de la lógica operativa y del almacenamiento en la nube. Deberá tener experiencia en el uso de Google Cloud Platform, así como del correcto manejo de versiones en Git.

UI/UX Designer

Metrofy se plantea como una app altamente amigable e intuitiva para el usuario, esto debido a que uno de sus objetivos es mejorar la experiencia de los usuarios del metro en aspectos como tiempos de traslado, rutas y seguridad. Por ello, este perfil es fundamental para lograr esto. Deberá tener una experiencia considerable en el manejo de usuarios, cuentas e interfaces de comunicación como mensajería o redes sociales.

Quality Assurance Engineer

Su tarea es explorar los posibles problemas que puedan presentarse en el software desarrollado. Para esto es vital que el encargado de esta área cuente con experiencia utilizando herramientas de prueba de QA (Robotium, TestComplete, etc.).

Project Manager

Encargado de asignar, supervisar y coordinar las tareas de cada uno de los miembros del equipo. Además, tendrá como misión principal establecer y hacer cumplir los tiempos de entrega adhiriéndose a los costos y plazos establecidos en un inicio. El encargado de estas tareas; deberá estar especializado en el manejo de Scrum, así como en el manejo de equipos pequeños de trabajo.

Estimación de costos

A pesar de que el desarrollo de esta app no requiere un equipo muy grande de trabajo o de un tiempo prolongado de implementación y creación del software, es importante mantener una correcta estructura para el manejo de costos. Para nuestro caso, se ha optado por clasificar los costos en las siguientes categorías:

Trabajo. Incluye salarios, prestaciones y demás beneficios del personal que participe en el proyecto.

Hardware. Herramientas y equipo físicos requeridas por el personal tales como equipos de cómputo, accesorios diversos y medios de comunicación.



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO
FACULTAD DE INGENIERÍA
DIVISIÓN DE INGENIERÍA ELÉCTRICA



Servicios. Herramientas adicionales que complementan los materiales. Entre estos están los servicios de Internet y proveedores externos.

Software. Incluye el pago para el uso de herramientas de desarrollo, así como las herramientas lógicas vitales para el proyecto.

Otros. Incluyen el costo de instalaciones o actividades extra para el desarrollo del proyecto.

A continuación, se enlistan los posibles elementos a considerar para cada una de las categorías anteriores.

Categoría	Elementos
Trabajo	<ul style="list-style-type: none">• Salario bruto para cada miembro del equipo de trabajo• Prestaciones extra para cada miembro del equipo de trabajo
Hardware	<ul style="list-style-type: none">• Equipo de cómputo para los miembros del equipo que así lo requieran• Dispositivos de almacenamiento
Servicios	<ul style="list-style-type: none">• Apoyo al colaborador para el pago de Internet• Apoyo al colaborador para el pago de electricidad
Software	<ul style="list-style-type: none">• Licencias para desarrollador Android• Cloud Storage en Google Cloud
Otros	<ul style="list-style-type: none">• Renta de oficinas de trabajo para reuniones presenciales semanales

En la siguiente tabla se muestra una estimación de los costos que tendrá cada uno de los elementos ya mencionados:

Concepto	Costo unitario	Cantidad	Costo Total
Salario mensual para 1 Project Manager	\$35,500	8	\$284,000
Salario mensual para 1 UI/UX Designer	\$24,500	8	\$196,000
Salario mensual para 1 Android Developer	\$30,200	8	\$241,600
Salario mensual para 1 Backend Developer	\$28,000	8	\$224,000
Salario mensual para 1 QA Engineer	\$29,500	8	\$236,000
Equipo de cómputo (opcional)	\$21,000	5	\$105,000
USB 64GB	\$125	5	\$625
Disco duro 1TB	\$1,300	1	\$1,300
Apoyo económico mensual para el pago de Internet	\$500	8	\$4,000
Apoyo económico mensual para el pago de electricidad	\$200	8	\$800
Licencia para desarrollador Android	\$478	5	\$2,390

Renta mensual de sala de juntas en CDMX	\$1,913	8	\$15,304
		Total	\$1,311,019

Siguiendo el criterio de estimación de magnitud, se tiene un intervalo de +- 25% del costo calculado del proyecto. Con esto, el costo para realizar esta app va de \$983,265.00 a \$1,683,773.75.

Demo de la aplicación

El demo de la aplicación fue realizado en la herramienta *Proto.io editor* ya que la versión de prueba ofrece varias características que nos llamaron la atención, además de que la interfaz es bastante intuitiva lo que nos facilitó el desarrollo del demo. A la par de este documento pdf en el repositorio también se encontrará una carpeta llamada *Metrofy_demo* con el demo de la aplicación en html para su consulta o en su defecto esta también se puede consultar a través de la siguiente liga <https://pr.to/58PT4G/>

Para el desarrollo de la demo se tomó en cuenta lo siguiente:

Logo de la aplicación

Se realizaron dos versiones para el logo de la aplicación, uno en donde el texto se encuentra al lado derecho del logo y otro en donde el texto se encuentra debajo del logo.

Además de lo anterior el tipo de letra usado para el texto es Caveat Brush



Imagen 21. Logo versión 1



Imagen 22. Logo versión 2

Paleta de colores

La paleta de colores usada para la aplicación es la siguiente (Imagen 23) está paleta de colores la sacamos extrayendo los colores de la imagen del logo.

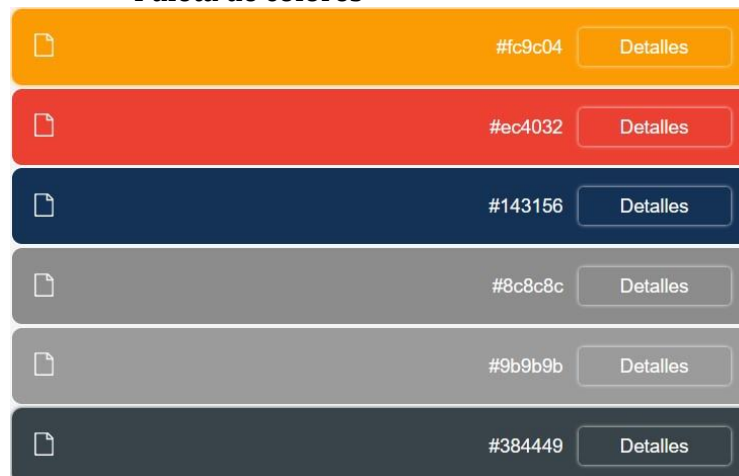


Imagen 23. Paleta de colores de la aplicación

Tipografía de la aplicación

Nombre de la pantalla: Caveat Brush

Texto: Roboto

Por último, las imágenes que se ocuparon para simular la vista del mapa fueron las siguientes, primero la de la red completa del Metro (Imagen 24) y posteriormente la vista de una sola línea (Imagen 25) en este caso fue la línea 2. Estas imágenes también se encuentran en el repositorio, en la carpeta Metrofy_demo/src/assets en caso de querer visualizarlas mejor.

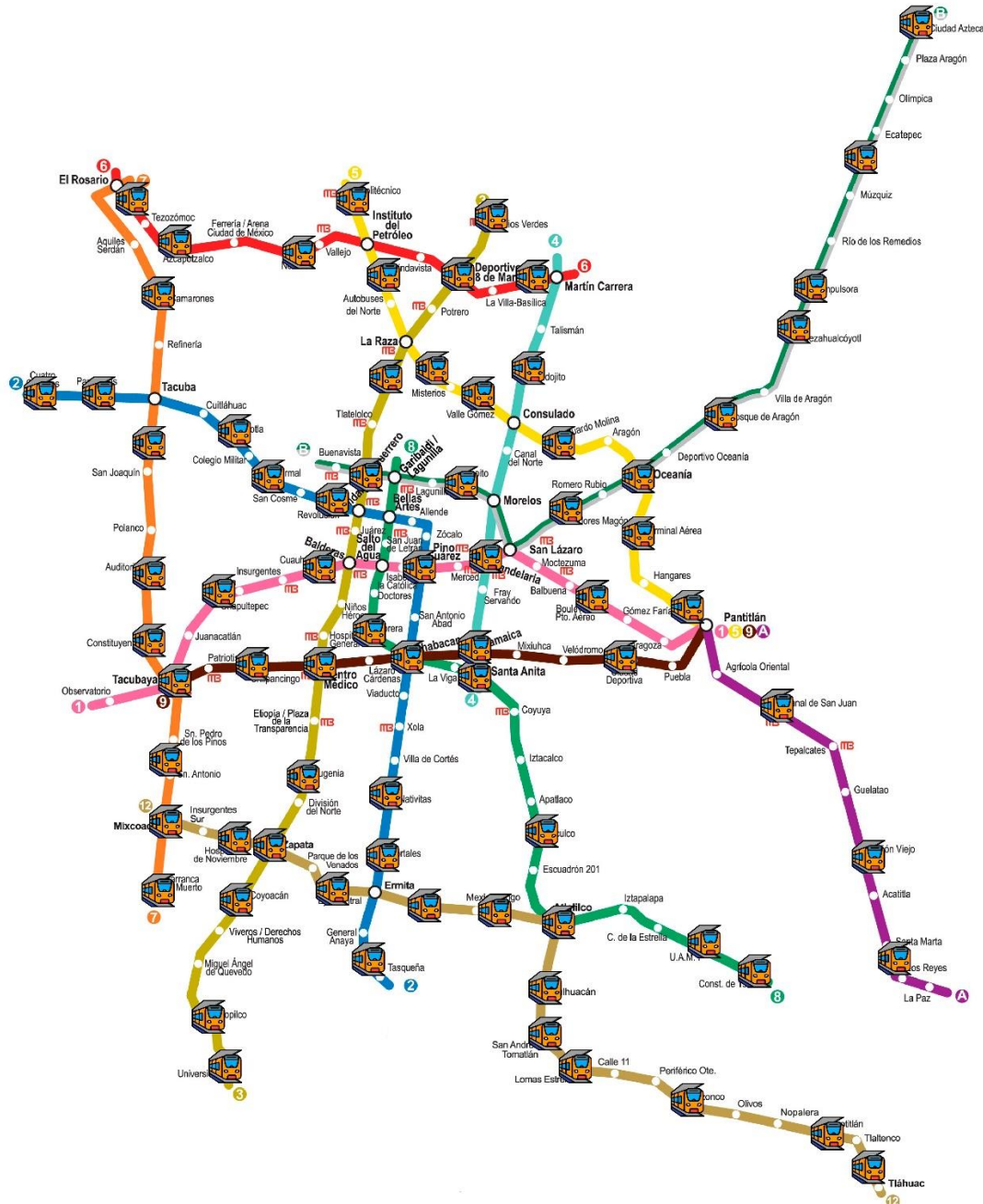


Imagen 24. Mapa red del metro

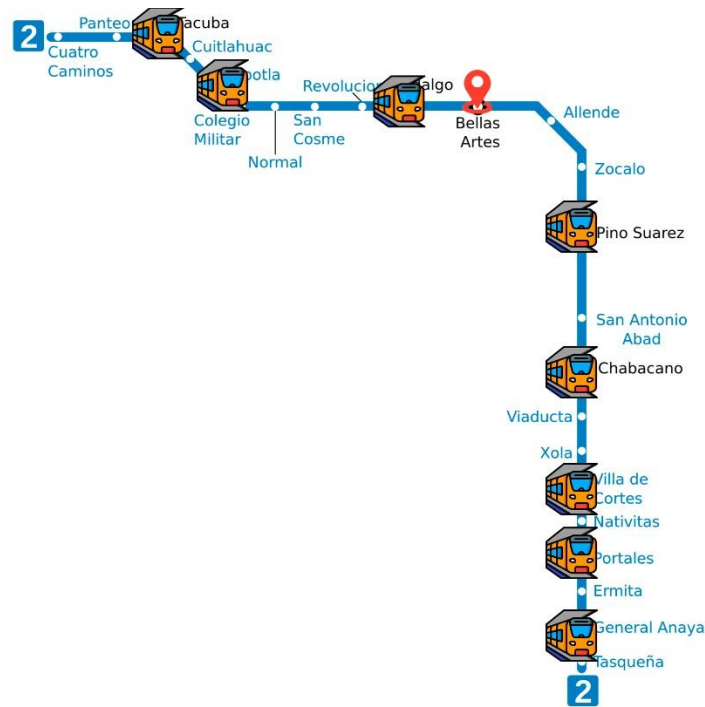


Imagen 25. Mapa línea 2

Conclusiones y reflexiones

Para esta segunda parte del análisis del desarrollo de la aplicación nos enfocamos más en la parte técnica, por lo cual fue necesario que profundizáramos en la investigación de algunos temas que previamente ya habíamos tocado en el primer análisis como tal es el caso de los servicios a ocupar para el desarrollo de la aplicación y es que al realizar el análisis por pantalla fue mucho más fácil darnos cuenta de qué herramientas/servicios sería necesario utilizar en función de las funciones que se realizaría en dicha pantalla. Otra parte en la que tuvimos que investigar más a fondo fue en la parte de estimación de costos que si bien habíamos revisado un poco de ello en la primera entrega y habíamos hecho una estimación en base a proyectos similares en esta nos enfocamos en realizar la estimación para nuestro proyecto en particular y nos dimos cuenta de qué es muy diferente hacerlo de esta manera pues la estimación obtenida fue más precisa.

Además de lo anterior en esta entrega también fue necesario hacer énfasis en el diseño de la aplicación lo que nos permitió trabajar y probar varias herramientas de diseño, pues, ya que, si bien al final sólo trabajamos con dos, *Balsamiq Cloud* para los Wireframe y *Proto.io* editor para el demo, al momento de elegir con qué herramienta trabajar tuvimos la oportunidad de probar más herramientas y darnos cuenta de las ventajas y desventajas que cada una de estas nos ofrecían.

Como bien se mencionó anteriormente, tuvimos que enfocarnos en el diseño gráfico de la aplicación y para ello se partió de un mapa de pantallas para poder realizar sus respectivos wireframes. Esta parte resultó la más interesante, ya que había que



considerar la disposición preliminar de botones y demás objetos para darle una apariencia sencilla e intuitiva además de proporcionar un flujo coherente para la aplicación.

En el caso del diseño para el demo tuvimos especial cuidado en los elementos visuales como la fuente y tamaño de letra, así como los colores, esto para que fuera atractiva pero también intuitiva y de fácil uso; siempre pensando en lo que le resultaría más cómodo al usuario.

Bibliografía

- Dore, E. (10 de Junio de 2022). *Entiende cómo desarrollar una app con geolocalización*.
Obtenido de Maplink Blog: <https://maplink.global/blog/es/crear-app-con-google-maps/>
- glassdoor. (4 de Enero de 2023). *Sueldos para Android Developer*. Obtenido de glassdoor:
https://www.glassdoor.com.mx/Sueldos/android-developer-sueldo-SRCH_K00,17.htm
- glassdoor. (4 de Enero de 2023). *Sueldos para Backend Developer*. Obtenido de glassdoor:
https://www.glassdoor.com.mx/Sueldos/backend-developer-sueldo-SRCH_K00,17.htm
- glassdoor. (5 de Enero de 2023). *Sueldos para Project Manager*. Obtenido de glassdoor:
https://www.glassdoor.com.mx/Sueldos/project-manager-sueldo-SRCH_K00,15.htm
- glassdoor. (5 de Enero de 2023). *Sueldos para Qa Engineer*. Obtenido de glassdoor:
https://www.glassdoor.com.mx/Sueldos/qa-engineer-sueldo-SRCH_K00,11.htm
- glassdoor. (3 de Enero de 2023). *Sueldos para Ui Ux Designer*. Obtenido de glassdoor:
https://www.glassdoor.com.mx/Sueldos/ui-ux-designer-sueldo-SRCH_K00,14.htm
- Juice Bot. (10 de Marzo de 2021). *¿Cómo integrar una pasarela de pago en una aplicación móvil?* Obtenido de Juice Studio: <https://juice-studio.com/como-integrar-una-pasarela-de-pago-en-una-aplicacion-movil>
- Kate Eby. (27 de Marzo de 2017). *Guía definitiva para estimar los costos del proyecto*. Obtenido de smartsheet: <https://es.smartsheet.com/ultimate-guide-project-cost-estimating>
- Regus. (s.f.). *Obtener cotización*. Obtenido de Regus: <https://www.regus.com/es-mx/get-a-quote>
- Twitter, Inc. (s.f.). *Twitter, Inc.* Obtenido de Twitter:
<https://developer.twitter.com/en/docs/twitter-api>