Acta de constitución

Geo Transporte

**Índice**

Este informe explica a detalle los siguientes contenidos:

1. Problemática del caso .
2. Propuesta de Solución del caso.
3. Metodología a aplicar en el proyecto.
4. Sprint Backlog.
5. Roles del proyecto.
6. Categorías de Riesgos.
7. Áreas y/o Fuentes de Riesgo.
8. RBS.

.

1. Recursos.
2. Riesgos .
3. Cronograma.
4. Presupuesto.
5. Conclusión.

**Problemática**

En las zonas rurales de Chile, los tiempos de espera del transporte público son prolongados y los paraderos se encuentran a mucha distancia de los hogares de gran parte de sus usuarios. Esta situación genera inconvenientes tanto para los pasajeros como a quienes proveen el servicio de transporte público. Los pasajeros se ven enfrentados a dificultades a la hora de acceder de manera eficiente al transporte público a su vez que los proveedores de este carecen de mecanismos efectivos para monitorear sus vehículos y brindar un mejor servicio.

La falta de una solución que permita a quienes proveen el servicio de transporte público monitorear en tiempo real la ubicación de sus conductores al mismo tiempo que ofrecen a sus usuarios información sobre la llegada del próximo transporte, afecta la calidad del servicio. Las zonas rurales de Chile son donde más relevancia toma esta problemática debido a la baja frecuencia del transporte público y la logística de las rutas es mucho más compleja.

La ausencia de herramientas de monitoreo y seguimiento impacta negativamente en la seguridad de sus usuarios, quienes se ven obligados a salir de sus hogares considerando largos tiempos de espera, en horas de madrugada, y de sus conductores los cuales carecen de información sobre su ubicación constante y sus rutas.

Toda esta situación contribuye a la insatisfacción de sus usuarios y a una percepción negativa del servicio, la falta de herramientas adecuadas para mejorar la eficiencia del servicio no solo incrementa los tiempos de espera y la incertidumbre de los pasajeros, sino también limita la capacidad de los proveedores para optimizar su servicio, generar confianza y garantizar un servicio más seguro.

En base a todo este contexto, es fundamental desarrollar un sistema que integre tanto tecnologías de software como bases de datos para optimizar el servicio de transporte público, entregando a sus usuarios la ubicación del próximo transporte y a sus proveedores la información relevante para la optimización y seguridad del servicio.

**Detalles del Proyecto**

Geo Transporte es una aplicación móvil que busca ser de uso público, idealmente para esto es necesario contar con socios ya sean municipales o dueños de todo tipo de vehículos que viajan por el país de uso público. Esto para que los usuarios puedan ir a tomar locomoción cerca de su entorno y zona, sin esperar tiempo de más y cumpliendo una hora establecida de llegada.

Como profesionales nuestro objetivo principal es entregar una aplicación móvil consistente con un buen desarrollo y rendimiento, para crear los distintos tipos de usuarios existentes y sus módulos correspondientes según sus credenciales.

El funcionamiento de la aplicación se basará en el uso de sistemas de señal gps con los cuales se puedan localizar los vehículos que están por pasar a través de las rutas y reducir en gran medida la probabilidad de largos tiempos de espera en la intemperie, a su vez reduciendo el riesgo de peligrosidad en diversas zonas poco concurridas y mejorando la calidad del servicio para el usuario.

También los conductores se verán beneficiados en la seguridad para ellos y los socios podrán comprobar el uso correcto de sus vehículos.

Recapitulando los puntos más importantes del proyecto y sus beneficios son :

* Reducción en tiempos de espera
* Aumento de la seguridad de trabajadores y cliente
* Mejora en el monitoreo de vehículos de transporte público

**Metodología**

La metodología a aplicar para el desarrollo de esta solución tecnológica será la metodología ágil Scrum, esto ya que podremos abordar problemas complejos adaptativos de forma eficiente y creativa con el máximo valor, ayudando así a estructurar y gestionar el trabajo de una manera más simple e intuitiva.

Para esto la estructura del proyecto en base al análisis del desafío a abordar consideraremos herramientas de arquitectura de sistema, mockups, gestión de riesgos, definición de épicas e historias de usuario, asignación de roles y tareas, para luego estructurar en jira los sprints, reuniones, cronogramas, backlog y el burdon charter:

1. Planificación : Product Backlog
2. Ejecución : Sprint
3. Control y monitorización : Daily Scrum y Burn Down Chart
4. Revisión y adaptación : Sprint Review & Retrospective

**Product Owner (Matias Sandoval)**: Responsable de definir las historias de usuario y priorizar el backlog. Representa los intereses del cliente.

**Scrum Master (Miguel Angel Hernàndez)**: Facilita las reuniones y se asegura de que el equipo siga las prácticas ágiles.

**Developer ( Michael Encina / Miguel Angel Hernàndez / Matias Sandoval )**: Desarrolladores que trabajan en la implementación del sistema. Incluye roles específicos como desarrollador front-end, back-end, y QA.

**Roles del Proyecto**

| Cargo | Responsabilidades |
| --- | --- |
| Director de proyecto | Planificar, ejecutar y cerrar proyectos, gestionar recursos, asegurar la entrega a tiempo y dentro del presupuesto. |
| Director de riesgos | Identificar, evaluar y mitigar riesgos, desarrollar planes de contingencia, supervisar el cumplimiento de las políticas de gestión de riesgos. |
| Gerente del proyecto | Coordinar las actividades diarias del proyecto, comunicar con los equipos, asegurar el cumplimiento de los plazos. |
| Coordinador del riesgo | Apoyar en la identificación y evaluación de riesgos, monitorear y reportar el estado de los riesgos, ayudar en la implementación de planes de mitigación. |
| Jefe de proyectos | Supervisar varios proyectos, proporcionar dirección y apoyo a los gerentes de proyecto, asegurar la alineación con los objetivos estratégicos de la empresa. |
| Técnico | Proveer soporte técnico, realizar instalaciones y mantenimiento, solucionar problemas técnicos. |
| Programador | Escribir, probar y mantener el código fuente de programas de software, colaborar con otros desarrolladores y equipos para asegurar la funcionalidad del software. |
| Tester | Realizar pruebas de software para identificar errores y problemas, documentar los resultados de las pruebas; trabajar con los desarrolladores para corregir los problemas. |
| Interesados | Involucrarse en el proyecto para asegurar que sus necesidades y expectativas se cumplan, proporcionar retroalimentación y apoyo; participar en las decisiones clave del proyecto. |

**Categorías de Riesgo**

**Técnico:** Se refiere a los riesgos asociados con las herramientas, plataformas, tecnologías y metodologías utilizadas en el desarrollo del proyecto. Incluye problemas de integración, compatibilidad, rendimiento, seguridad y cambios tecnológicos.

**Externo:** Se enfoca en los riesgos fuera del control directo del proyecto, como la dependencia de terceros (servicios de APIs, socios o patrocinadores), cambios en políticas regulatorias o legislativas, y factores económicos o sociales que puedan afectar el proyecto.

**Organizacional**: Involucra riesgos relacionados con la estructura interna de la organización, incluyendo problemas financieros, falta de recursos, problemas de comunicación interna, y conflictos entre diferentes áreas o departamentos.

**Dirección de Proyectos:** Esta categoría abarca los riesgos que surgen de la gestión del proyecto, incluyendo planificación deficiente, control ineficaz del alcance, falta de seguimiento de riesgos, y problemas de coordinación y comunicación entre los miembros del equipo.

**Operativo:** Se refiere a los riesgos relacionados con la operación diaria del proyecto, incluyendo mantenimiento, escalabilidad, monitoreo y respuesta a incidentes, y problemas de disponibilidad y conectividad.

**Áreas y/o Fuentes de Riesgo**

**Desarrollo :** Esta área se enfoca en los riesgos relacionados con el proceso de desarrollo de la aplicación, incluyendo problemas en la integración de APIs, compatibilidad con diferentes plataformas, y desafíos en la optimización de la aplicación.

**Tecnología :** Esta área busca analizar los riesgos posibles al implementar una infraestructura tecnológica, con las herramientas de desarrollo y red. Para así obtener una evaluación y alcance de la la plataforma de desarrollo

**Requisitos:**Esta área busca analizar todos los requisitos funcionales y no funcionales del proyecto para así encontrar posibles riesgos que no se han descubierto, no están bien evaluados o están mal desarrollados

**Seguridad :**  Incluye riesgos relacionados con la protección de datos, vulnerabilidades en la comunicación, gestión de permisos y privacidad del usuario. Es crucial prevenir la exposición de datos sensibles y asegurar que la aplicación cumpla con las normativas de seguridad.

**Dependencias de Terceros :** Esta área se centra en los riesgos derivados de la dependencia de servicios externos, como APIs de mapas, servicios de GPS, y patrocinadores. Los cambios en las políticas de uso, fallas en los servicios externos, y la dependencia crítica de socios son algunos de los riesgos clave.

**Legales y Regulatorios :** Involucra el cumplimiento de leyes y regulaciones, especialmente en cuanto a la protección de datos y los requisitos legales específicos de cada región o país. El incumplimiento de estas normativas puede resultar en sanciones y limitar el alcance del proyecto.

**Usuarios :** Esta área se enfoca en los riesgos asociados con la gestión de usuarios, incluyendo la exposición de datos sensibles, la falta de validación de sesión, y la privacidad de los datos del usuario. Es esencial para asegurar una experiencia de usuario segura y confiable.

**Financiamiento :** Esta área se encarga de analizar y prevenir todos los posibles casos de financiamiento del proyecto, tanto para la falta de capital o ingreso insuficiente para cubrir los costos de operación del proyecto.

**Recursos :** Esta área busca analizar prevenir los riegos posible en el ámbito de ganancia a ser un proyecto de uso público y aun sin socios

**Comunicación :** Esta área se encarga de evaluar y prevenir los posibles riesgos por falta de comunicación entre colaboradores, empleadores y desarrolladores pertenecientes al proyecto. Para así mitigar y evitar una mala comunicación e iteración del desarrollo de las actividades.

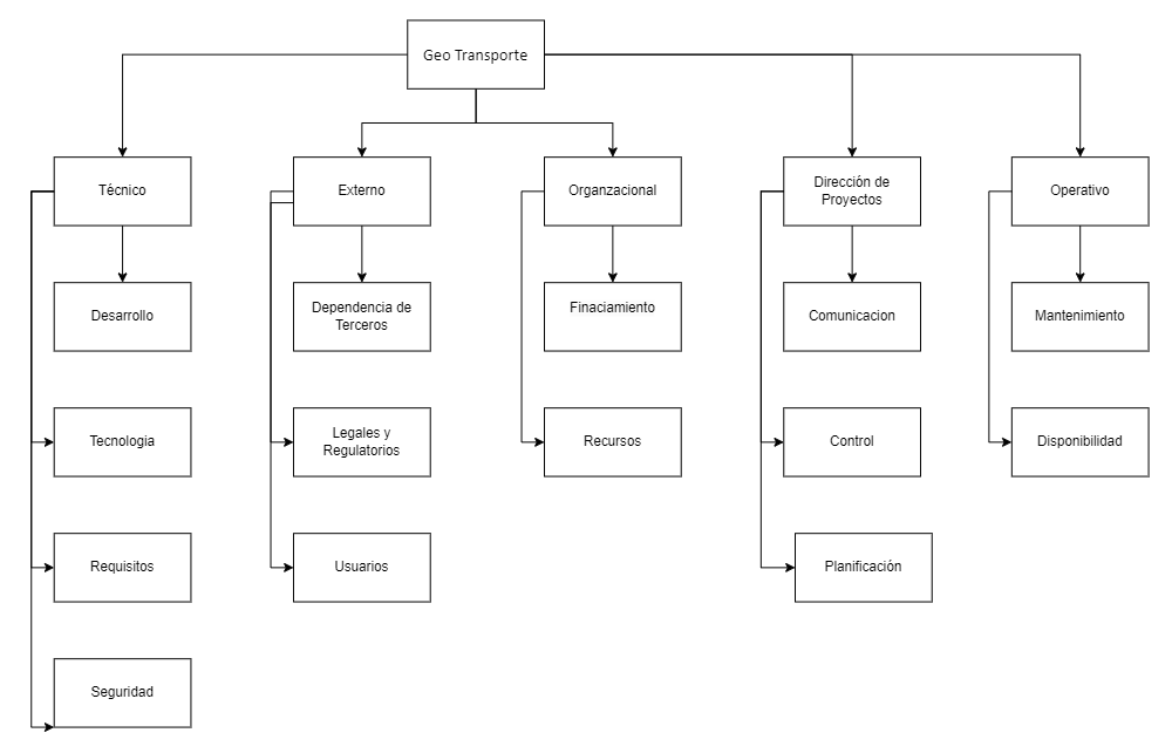
**Control :** Esta área se encarga de prevenir una mala gestión y control de parte del Encargado/a Institucional de Riesgo y del Coordinador de Riesgos para entregar una retroalimentación y control continuo y eficiente de estos actores para un buen desarrollo del proyecto.

**Planificación :** Esta área se encarga de analizar cuál sería la forma más eficiente de planificar y organizar todas las actividades en un paso a paso en el desarrollo del proyecto. Para así evitar riesgos de mala gestión de las actividades, desarrollos fuera de tiempo o incumplimientos por parte del equipo de trabajo.

**Mantenimiento :** Esta área se enfoca en los riesgos relacionados con la capacidad de la aplicación para mantenerse operativa y actualizada a lo largo del tiempo. Los principales riesgos incluyen problemas en la implementación de actualizaciones, mantenimiento inadecuado que podría afectar el rendimiento y la seguridad, y la falta de recursos o planificación para el mantenimiento a largo plazo.

**Disponibilidad :** Se centra en los riesgos relacionados con la disponibilidad de la aplicación para los usuarios finales. Incluye problemas de conectividad, fallos en la infraestructura que soporta la aplicación, y desafíos para escalar la infraestructura para manejar un aumento en la demanda. Garantizar la disponibilidad continua es crucial para mantener la confianza del usuario y la reputación del servicio.

**RBS**



**Recursos**

Los recursos necesarios para el desarrollo e implementación del proyecto GeoTransporte, son los necesarios para generar el ambiente de desarrollo del proyecto, para esto se estima tanto el área de hardware como también la de software. Para esto se hará una lista de las herramientas necesarias

**1. Recursos de software**

**Framework y Herramientas de Desarrollo**

* **Ionic Framework (v7):** El framework para desarrollar la aplicación híbrida.
* **Angular :** Framework frontend para gestionar la lógica de la aplicación.
* **Capacitor/Cordova :** Para la integración de las API nativas necesarias en dispositivos móviles.
* **Firebase y Firestore :** Para el almacenamiento en tiempo real y base de datos.
* **Node.js + npm :** Para gestionar dependencias y scripts de construcción.
* **Visual Studio Code o cualquier editor de texto robusto para escribir y mantener el código.**
* **Git + GitHub/GitLab :** Para control de versiones y colaboración en el equipo.

**Servicios en la nube**

* **Firebase :** Almacenamiento en tiempo real, hosting y otras funcionalidades de backend.

#### **Pruebas y control de calidad**

1. **Simuladores y emuladores**
   * **Emulador de Android Studio** : Para probar la aplicación en diferentes versiones de Android.
2. **Pruebas Funcionales**
   * **Selenium** : Herramientas para pruebas funcionales.
3. **CI/CD (Integración Continua/Despliegue)**
   * **GitHub Actions/GitLab CI** : Para automatizar las pruebas y el despliegue de la aplicación.

#### **Api Keys y Dependencias de Software**

* **Api keys(Google Cloud** : Apis relacionadas a mapas, rutas, localizadores para el desarrollo e implementación en la aplicación .
* **Plugins o dependencias de terceros** : Plugins de Ionic/Capacitor/Cordova, si los usas para funcionalidades específicas como geolocalización, notificaciones push, etc.

**2. Recursos de hardware**

**Desarrollo**

1. **Estación de trabajo del desarrollador :**
   * **PC o Mac :** Para el desarrollo de aplicaciones móviles. Recomendación**:**
     + **Mínimo:**
       - **Procesador: Intel Core i5 o AMD Ryzen 5**
       - **RAM: 8 GB**
       - **Almacenamiento: SSD de 256 GB**
     + **Recomendado:**
       - **CPU: Intel Core i7 o AMD Ryzen 7 (o superior)**
       - **RAM: 16 GB o superior**
       - **Almacenamiento: 512GB SSD o superior**
2. **Dispositivos móviles para pruebas :**
   * **Smartphones Android : Prueba en una gama de dispositivos con versiones de Android 10 o superior.**
   * **iPhone : Al menos uno o dos dispositivos que soporten iOS 14 o superior.**
3. **Tableta u otros dispositivos :**
   * **iPad o tableta Android : Para pruebas en pantallas más grandes y asegurar la compatibilidad.**

**Pruebas y control de calidad**

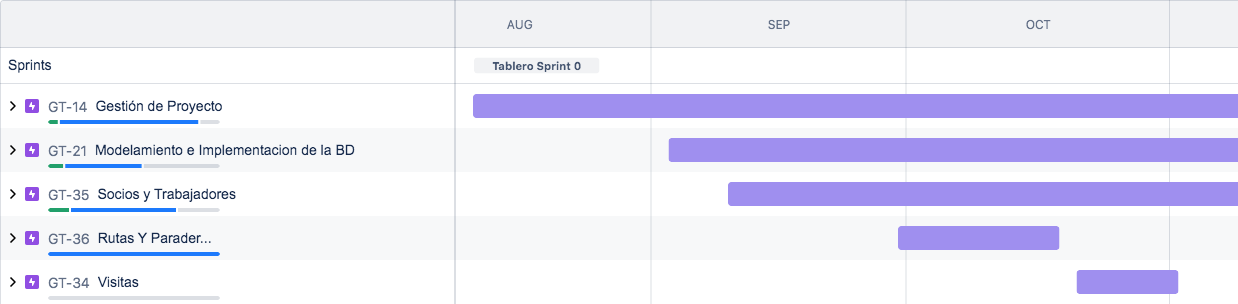
1. **Redes y servidores :**
   * **Servidor de pruebas : VPS o servidor en la nube con Linux para pruebas de integración, si no estás usando servicios como Firebase o Supabase exclusivamente.**
   * **Ancho de banda de internet confiable :** Para sincronización y pruebas de backend, idealmente con un buen ancho de banda, especialmente al trabajar con servicios en la nube.

**Infraestructura para el Despliegue y Producción** (**Acceso a tiendas de aplicaciones) :**

* **Cuenta de desarrollador en Google Play :** $25 (pago único).

1. Riesgos .
2. Cronograma.

Es el siguiente cronograma del Proyecto [GeoTransporte](https://duocuc-team-x0zo4fwj.atlassian.net/jira/software/projects/GT/boards/5/timeline?selectedIssue=GT-14&shared=&atlOrigin=eyJpIjoiZTNmOTViZmU0MWRjNDZiNWI0ZDBiNmEzN2FhN2JjMTAiLCJwIjoiaiJ9) en jira se presenta la cronología del desarrollo según las épicas del proyecto, el print 0 aparece en blanco ya que este ya se termino y finalizó, la épica de gestión de proyecto es una de las épicas que duran durante todo el proyecto ya que en ella se encuentran actividades de relación a la implementación análisis del proyecto como también la puesta en marcha blanca y pruebas de funcionalidad y rendimiento en la fase 5



1. Presupuesto.
2. Conclusión.