



Universidad Peruana de Ciencias Aplicadas

Carrera de Ingeniería de Software

Ciclo 202520

1ASI0572 - Desarrollo de Soluciones IOT

NRC: 3443

Profesor: Velásquez Núñez, Angel Augusto

Informe de Trabajo Final

Startup: Los Parkers

Producto: CargaSafe

Relación de integrantes

Código	Nombre
u20201c410	Garro Vega, Marcelo Fabian
u202113324	Sanchez Ignacio, Jefrey Martin
u20211c273	Aliaga Pimentel, George Arturo
u202113640	Bernardo Eusebio Alessandro Joaquin
u202019577	Chirinos Arevalo Daniel Rodrigo
u202214869	Vera Nuñez Nicolas Alejandro

Diciembre 2025

Registro de Versiones del Informe

Versión	Fecha	Autor	Descripción de modificación
1.0	20/09/2025	George	Se entregó una primera entrega del informe con los siguientes apartados: Carátula, Registro de Versiones, Project Report Collaboration Insights, Contenido, Student Outcome, Capítulo I: Introducción, Capítulo II: Requirements Elicitation & Analysis, Capítulo III: Requirements Specification, Capítulo IV: Solution Software Design y Bibliografía.
1.0.1	12/09/2025	Daniel y Marcelo	Se añadió el capítulo 3 que es en base al product backlog que se trabajará en el proyecto
1.0.2	13/09/2025	Jefrey y Nicolas	Se añadió el capítulo 4 que es en base a las soluciones en diseño del software que trabajaremos en el proyecto
2.0	09/10/2025	Jefrey	Se añadió el capítulo 6.2.1.4 que es en base evidencias del trabajo durante el proyecto
2.0.1	09/10/2025	Daniel	Se realizaron las correcciones y se añadió el capítulo 6 que es la configuración y el uso de las herramientas que se utilizaran durante el ciclo.
2.0.2	09/10/2025	Marcelo	Se agrego proceso de event storming con sus resultados finales
3.0	14/11/2025	Daniel	Se añadió un nuevo Sprint sobre el desarrollo de los productos
3.0.1	14/11/2025	Daniel	Se añadieron las entrevistas de las validaciones de los usuarios
3.0.2	14/11/2025	George	Se añadió las heurísticas de validación
4.0	02/12/2025	Daniel	Se añadieron las conclusiones y recomendaciones
4.0.1	03/12/2025	Daniel	Se añadió el Sprint 3
4.0.2	03/12/2025	Daniel	Se añadieron los anexos
4.0.3	03/12/2025	Marcelo	Se actualizo el contenido del Sprint Goal 3 alineado al formato de Scrum.org

Project Report Collaboration Insights

En esta sección, se registra las colaboraciones realizadas por los miembros del equipo durante el desarrollo del informe del proyecto. En primer lugar, se brinda el enlace del repositorio del reporte del proyecto en la plataforma GitHub.

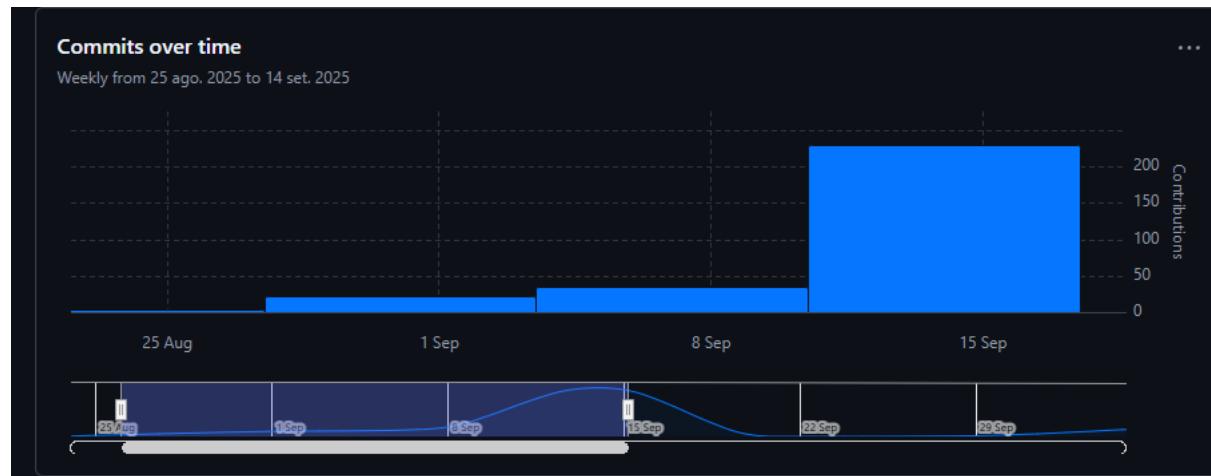
-**Project Report:** <https://github.com/Los-Parkers-IoT/LosParkers-report>

A continuación, se explicará todo a cerca del desarrollo de actividades para la elaboración del informe. Adicionalmente, se presentan las métricas de las acciones del equipo del Project Report de cada entrega correspondiente en forma de contribuciones, commits y network graph registrados en GitHub.

TB1 Project Report Collaboration Insights

Para la entrega del TB1 se realizaron las actividades necesarias para completar los capítulos I, II, III y IV, resaltando la importancia de la constancia en el trabajo. Como equipo, mantuvimos una frecuencia adecuada de commits y actualizaciones, proyectada como óptima para el desarrollo futuro, y se incluyen en el informe las evidencias de los cambios efectuados.

Vista general de las contribuciones del equipo



Contribuciones de cada miembro del equipo para la TB1





De las contribuciones del equipo y commits se concluye lo siguiente:

- Para esta entrega, se han realizado un total de **262 commits** aproximadamente por parte del equipo.
- La mayor concentración de commits se dio en la semana del 15 de septiembre, alcanzando aproximadamente 200 contribuciones.
- El equipo considera que hubo falta de organización durante la realización de la TB1.

Para el control de versiones, decidimos emplear **6 ramas principales**:

main : Esta rama se mantiene como un reflejo del documento en su última versión estable. El estado del documento será listo para entregar.

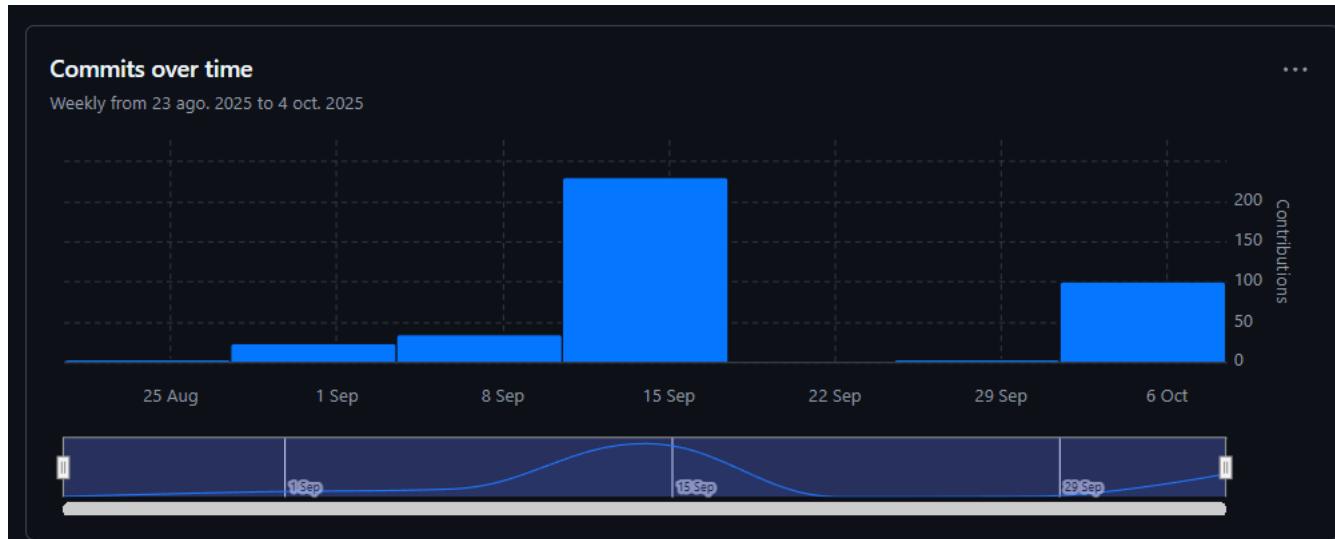
chapter-1, chapter-2, chapter-3 y chapter-4 : Estas ramas permiten integrar los artefactos solicitados por capítulo. Se trabaja de esta manera para asegurar una integración controlada, una colaboración enfocada y un aislamiento de cambios.

chapter-1-2-3-4 : Esta rama funciona como zona de integración pre-producción donde se consolidan y revisan todos los capítulos antes de fusionarlos a **main**. Permite realizar pruebas de integración completas, verificar la coherencia entre capítulos y realizar ajustes finales sin afectar la rama principal. Actúa como un paso intermedio de validación antes del merge final.

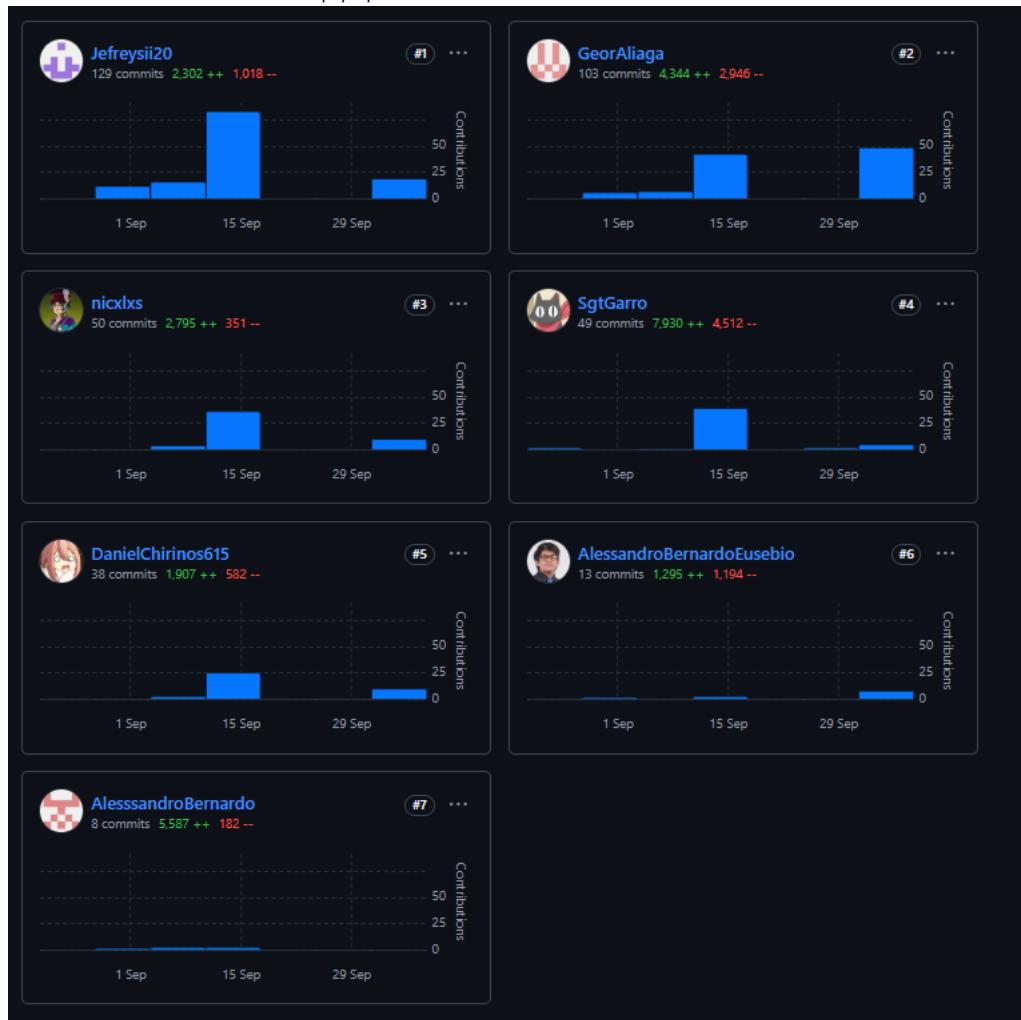
TP Project Report Collaboration Insights

Para la entrega del TP se realizaron las actividades necesarias para corregir y completar los capítulos I, II, III, IV, V y VI resaltando la importancia de la constancia en el trabajo. Como equipo, mantuvimos una frecuencia adecuada de commits y actualizaciones, proyectada como óptima para el desarrollo futuro, y se incluyen en el informe las evidencias de los cambios efectuados.

Vista general de las contribuciones del equipo



Contribuciones de cada miembro del equipo para la TP



Commits

De las contribuciones del equipo y commits se concluye lo siguiente:

- Para esta entrega, se han realizado un total de **390 commits** aproximadamente por parte del equipo.
- El equipo considera que hubo una ligera mejor organización durante la realización de la TP1.

Para el control de versiones, decidimos emplear **6 ramas principales**:

main: Esta rama se mantiene como un reflejo del documento en su última versión estable. El estado del documento será listo para entregar.

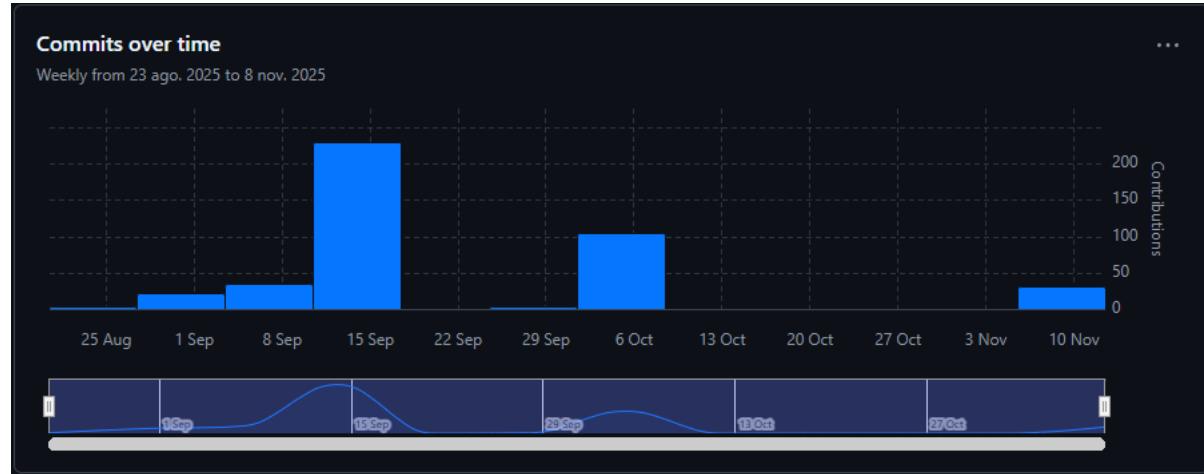
chapter-1, chapter-2, chapter-3, chapter-4, chapter-5, chapter-6: Estas ramas permiten integrar los artefactos solicitados por capítulo. Se trabaja de esta manera para asegurar una integración controlada, una colaboración enfocada y un aislamiento de cambios.

TB2 Project Report Collaboration Insights

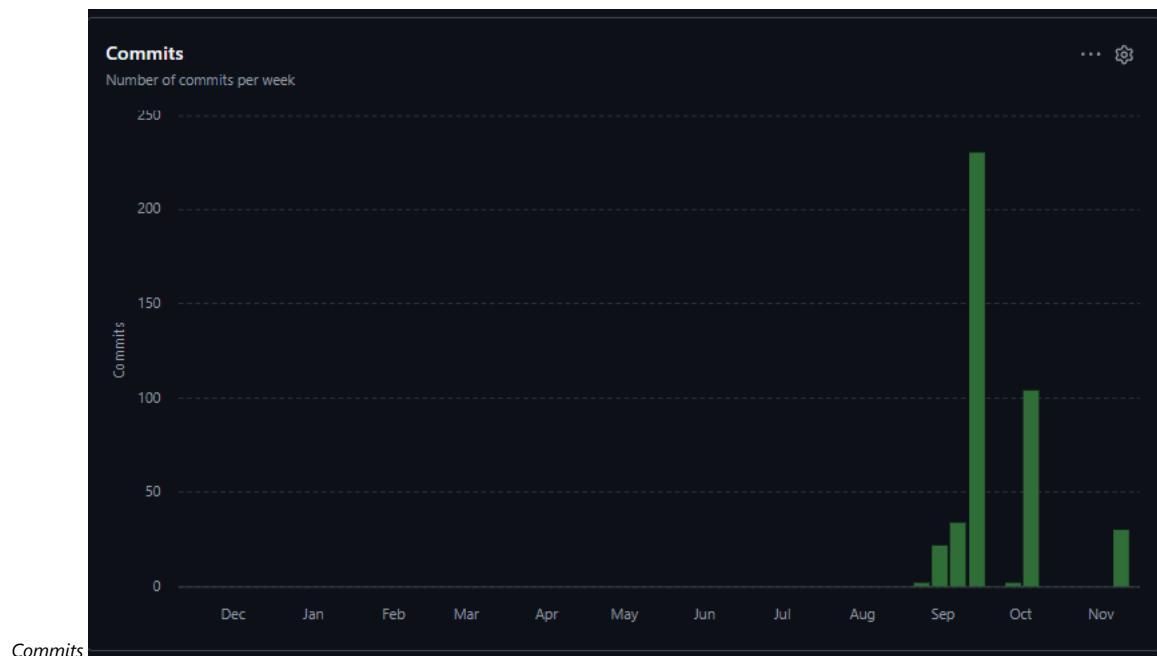
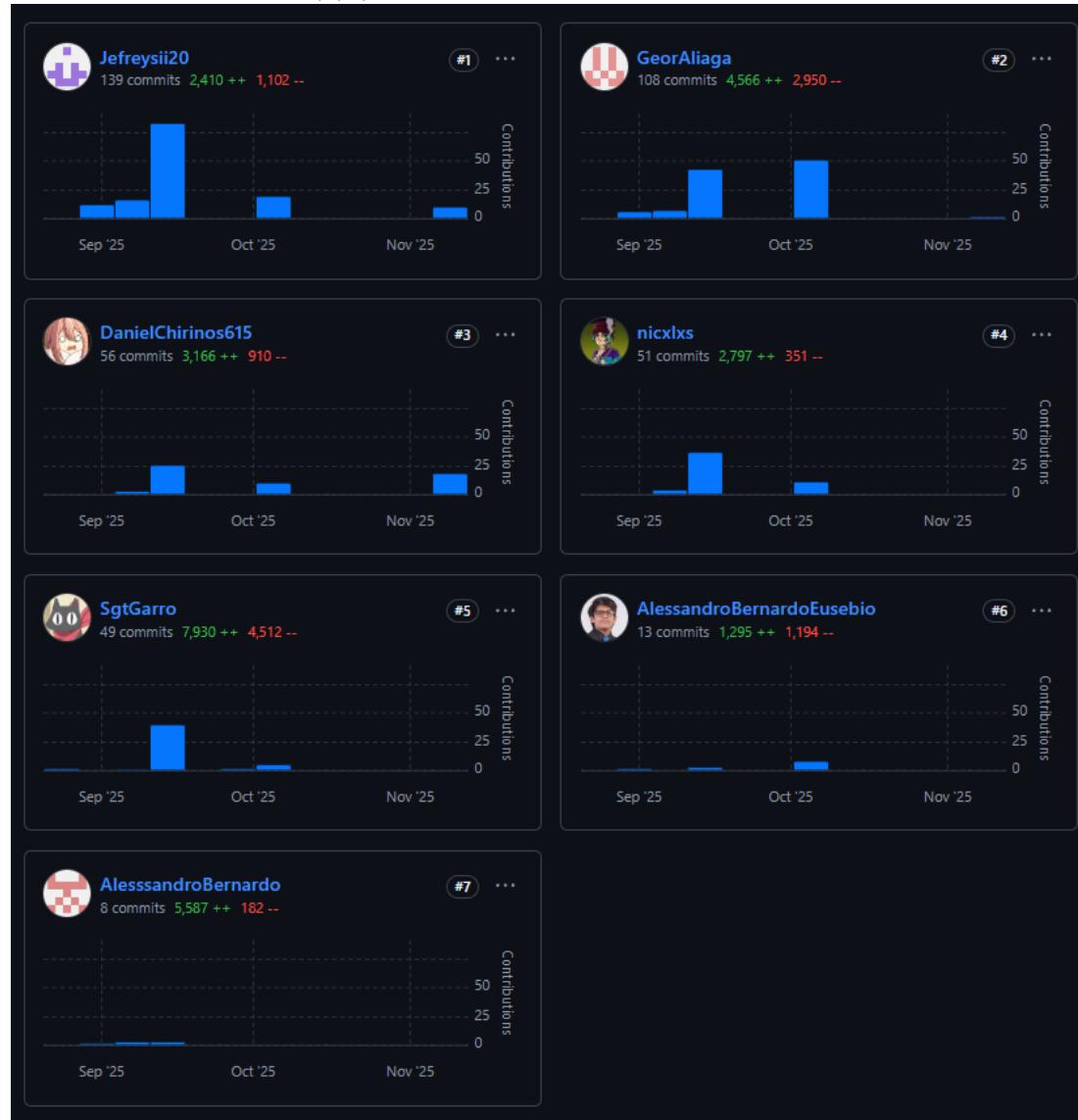
Para la entrega del TB2 se realizaron las actividades necesarias para completar el ultimo capítulo resaltando la importancia de la constancia en el trabajo. Como equipo, mantuvimos una frecuencia adecuada de commits y actualizaciones, proyectada como óptima para el desarrollo futuro, y se incluyen en el informe las evidencias de los cambios efectuados.

De las contribuciones del equipo y commits se concluye lo siguiente:

Vista general de las contribuciones del equipo



Contribuciones de cada miembro del equipo para la TB2



- Para esta entrega, se han realizado un total de **424 commits** aproximadamente por parte del equipo.
- El equipo considera que hubo una mejor organización durante la realización de la TF.

TF Project Report Collaboration Insights

Para la entrega del Trabajo Final (TF), se realizaron las actividades necesarias para completar los capítulos finales y consolidar el proyecto, resaltando la importancia de la constancia en el trabajo. Como equipo, hemos mantenido una frecuencia adecuada de commits y actualizaciones, proyectada como óptima para el desarrollo futuro, y se incluyen en el informe las evidencias de los cambios efectuados.

De las contribuciones del equipo y commits se concluye lo siguiente:

Vista general de las contribuciones del equipo



Contribuciones de cada miembro del equipo para la TF





Commits

- Para esta entrega, se han realizado un total de **440 commits** aproximadamente por parte del equipo.
 - El equipo considera que hubo una ligera mejor organización durante la realización de la TF.
-

Contenido

Tabla de contenidos

- Student Outcome
- Capítulo I: Introducción
 - 1.1. Startup Profile
 - 1.1.1. Descripción de la Startup
 - 1.1.2. Perfiles de integrantes del equipo
 - 1.2. Solution Profile
 - 1.2.1. Antecedentes y problemática
 - What (¿Qué?)
 - Who (¿Quién?)
 - Where (¿Dónde?)
 - When (¿Cuándo?)
 - Why (¿Por qué?)
 - How (¿Cómo?)
 - How much (¿Cuánto?)
 - 1.2.2. Lean UX Process
 - 1.2.2.1. Lean UX Problem Statements
 - 1.2.2.2. Lean UX Assumptions
 - Business Assumptions
 - Business Outcome Assumptions
 - User Assumptions
 - User Outcomes & Benefits Assumptions
 - Feature Assumptions
 - 1.2.2.3. Lean UX Hypothesis Statements
 - Hipótesis 1
 - Hipótesis 2
 - Hipótesis 3
 - Hipótesis 4
 - 1.2.2.4. Lean UX Canvas
 - 1.3. Segmentos objetivo
 - 1.3.1 Segmento 1: Empresas Clientes
 - 1.3.2 Segmento 2: Clientes Finales
- Capítulo II: Requirements Elicitation & Analysis
 - 2.1. Competidores
 - 2.1.1. Análisis competitivo
 - 2.1.2. Estrategias y tácticas frente a competidores
 - 2.2. Entrevistas
 - 2.2.1. Diseño de entrevistas
 - 1. Preguntas generales
 - 2. Preguntas — **Segmento: Empresa (Gestores de transporte)**
 - 3. Preguntas — **Segmento: Clientes Finales (Consumidores finales)**
 - 2.2.2. Registro de entrevistas
 - Segmento 1: Empresa
 - Segmento 2: Clientes Finales (Consumidores finales)
 - 2.2.3. Análisis de entrevistas
 - 2.3. Needfinding
 - 2.3.1. User Personas
 - 2.3.2. User Task Matrix
 - 2.3.3. User Journey Mapping
 - Journey Map: Carlos Mendoza (Gestor de transporte)
 - Journey Map: María González (Gerente de Compras de Restaurante)
 - 2.3.4. Empathy Mapping
 - Segmento: Empresa (Gestores de transporte) - Carlos Mendoza
 - Segmento: Clientes Finales (Consumidores finales) - María González
 - 2.4. Big Picture EventStorming
 - 1. Preparing the Room
 - 2. Energizing the audience
 - 3. Briefing and Presenting the Agenda
 - 4. Generating Domain Events
 - 5. Sorting Domain Events
 - 6. Adding Actors and External Systems
 - Actores:
 - Sistemas Externos:
 - 7. Storytelling
 - 8. Reverse Storytelling

- 9. Closing
- 2.5. Ubiquitous Language
 - **Términos del Dominio Central**
 - **Términos de Actores**
 - **Términos de Procesos de Negocio**
 - **Contexto de Métricas y KPIs**
- Capítulo III: Requirements Specification
 - 3.1. User Stories
 - 3.2. Impact Mapping
 - Empresa (Gestores de transporte)
 - Clientes Finales (Consumidores finales)
 - 3.3. Product Backlog
- Capítulo IV: Solution Software Design
 - 4.1. Strategic-Level Domain-Driven Design
 - 4.1.1. Design-Level EventStorming
 - 4.1.1.1 Candidate Context Discovery
 - 4.1.1.2. Domain Message Flows Modeling
 - Historias de dominio (Domain Stories)
 - Resultados
 - 4.1.1.3. Bounded Context Canvases
 - 4.1.2. Context Mapping
 - Resultado
 - 4.1.3. Software Architecture
 - 4.1.3.1. Software Architecture System Landscape Diagram
 - Propósito
 - Elementos incluidos
 - Relaciones principales
 - Resultado
 - 4.1.3.2. Software Architecture Context Level Diagrams
 - 4.1.3.2. Software Architecture Container Level Diagrams
 - 4.1.3.3. Software Architecture Deployment Diagrams
- 4.2. Tactical-Level Domain-Driven Design
 - 4.2.1. Bounded Context: Identity and Access Management
 - 4.2.1.1. Domain Layer
 - 4.2.1.2. Interface Layer
 - 4.2.1.3. Application Layer
 - 4.2.1.4. Infrastructure Layer
 - 4.2.1.5. Bounded Context Software Architecture Component Level Diagrams
 - 4.2.1.6. Bounded Context Software Architecture Code Level Diagrams
 - 4.2.1.6.1. Bounded Context Domain Layer Class Diagrams
 - 4.2.1.6.2. Bounded Context Database Design Diagram
 - 4.2.2. Bounded Context: *Subscriptions and Billing*
 - 4.2.2.1. Domain Layer
 - 4.2.2.2. Interface Layer
 - 4.2.2.3. Application Layer
 - 4.2.2.4. Infrastructure Layer
 - 4.2.2.5. Bounded Context Software Architecture Component Level Diagrams
 - 4.2.2.6. Bounded Context Software Architecture Code Level Diagrams
 - 4.2.2.6.1. Bounded Context Domain Layer Class Diagrams
 - Explicación del diagrama
 - 4.2.2.6.2. Bounded Context Database Design Diagram
 - Explicación del diagrama
 - 4.2.3. Bounded Context: *Alerts & Resolution*
 - 4.2.3.1. Domain Layer
 - 4.2.3.2. Interface Layer
 - 4.2.3.3. Application Layer
 - 4.2.3.4. Infrastructure Layer
 - 4.2.3.5. Bounded Context Software Architecture Component Level Diagrams
 - 4.2.3.6. Bounded Context Software Architecture Code Level Diagrams
 - 4.2.3.6.1. Bounded Context Domain Layer Class Diagrams
 - 4.2.3.6.2. Bounded Context Database Design Diagram
 - 4.2.4. Bounded Context: *Real-Time Monitoring*
 - 4.2.4.1. Domain Layer.
 - 4.2.4.2. Interface Layer.
 - 4.2.4.3. Application Layer.
 - 4.2.4.4. Infrastructure Layer.
 - 4.2.4.5. Bounded Context Software Architecture Component Level Diagrams
 - 4.2.4.6. Bounded Context Software Architecture Code Level Diagrams
 - 4.2.4.6.1. Bounded Context Domain Layer Class Diagrams

- 4.2.4.6.2. Bounded Context Database Design Diagram
 - 4.2.5. Bounded Context: *Trip management*
 - 4.2.5.1. Domain Layer.
 - 4.2.5.2. Interface Layer.
 - 4.2.5.3. Application Layer.
 - 4.2.5.4. Infrastructure Layer.
 - 4.2.5.5. Bounded Context Software Architecture Component Level Diagrams.
 - 4.2.5.6. Bounded Context Software Architecture Code Level Diagrams.
 - 4.2.5.6.1. Bounded Context Domain Layer Class Diagrams.
 - 4.2.5.6.2. Bounded Context Database Design Diagram.
 - 4.2.6. Bounded Context: Fleet Management
 - 4.2.6.1. Domain Layer
 - 4.2.6.2. Interface Layer
 - Controllers principales (HTTP REST)
 - 4.2.6.3. Application Layer
 - 4.2.6.4. Infrastructure Layer
 - 4.2.6.5. Bounded Context Software Architecture Component Level Diagrams.
 - 4.2.6.6. Bounded Context Software Architecture Code Level Diagrams.
 - 4.2.6.6.1. Bounded Context Domain Layer Class Diagrams.
 - 4.2.6.6.2. Bounded Context Database Design Diagram.
 - 4.2.7. Bounded Context: Profile and Preferences Management
 - 4.2.7.1. Domain Layer.
 - 4.2.7.2. Interface Layer.
 - 4.2.7.3. Application Layer.
 - 4.2.7.4. Infrastructure Layer.
 - 4.2.7.5. Bounded Context Software Architecture Component Level Diagrams.
 - 4.2.7.6. Bounded Context Software Architecture Code Level Diagrams.
 - 4.2.7.6.1. Bounded Context Domain Layer Class Diagrams.
 - 4.2.7.6.2. Bounded Context Database Design Diagram
 - 4.2.8. Bounded Context: Visualization Analytics
 - 4.2.8.1. Domain Layer
 - 4.2.8.2. Interface Layer
 - 4.2.8.3. Application Layer
 - 4.2.8.4. Infrastructure Layer
 - 4.2.8.5. Bounded Context Software Architecture Component Level Diagrams
 - 4.2.8.6. Bounded Context Software Architecture Code Level Diagrams
 - 4.2.8.6.1. Bounded Context Domain Layer Class Diagrams
 - 4.2.8.6.2. Bounded Context Database Design Diagram
 - 4.2.9. Bounded Context: Merchant
 - 4.2.9.1. Domain Layer
 - 4.2.9.2. Interface Layer
 - 4.2.9.3. Application Layer
 - 4.2.9.4. Infrastructure Layer
 - 4.2.9.5. Bounded Context Software Architecture Component Level Diagrams
 - 4.2.9.6. Bounded Context Software Architecture Code Level Diagrams
 - 4.2.9.6.1. Bounded Context Domain Layer Class Diagrams
 - 4.2.9.6.2. Bounded Context Database Design Diagram
- Capítulo V: Solution UI/UX Design
 - 5.1. Style Guidelines.
 - 5.1.1. General Style Guidelines.
 - 5.1.2. Web, Mobile and IoT Style Guidelines.
 - 5.2. Information Architecture.
 - 5.2.1. Organization Systems.
 - 5.2.2. Labeling Systems.
 - 5.2.3. SEO Tags and Meta Tags
 - 5.2.4. Searching Systems.
 - 5.2.5. Navigation Systems.
 - 5.3. Landing Page UI Design.
 - 5.3.1. Landing Page Wireframe.
 - 5.3.2. Landing Page Mock-up.
 - 5.4. Applications UX/UI Design.
 - 5.4.1. Applications Wireframes.
 - 5.4.2. Applications Wireflow Diagrams.
 - 5.4.2. Applications Mock-ups.
 - 5.4.3. Applications User Flow Diagrams.
 - 5.5. Applications Prototyping.
 - Capítulo VI: Product Implementation, Validation & Deployment
 - 6.1. Software Configuration Management.
 - 6.1.1. Software Development Environment Configuration.
 - 6.1.2. Source Code Management.

- 6.1.3. Source Code Style Guide & Conventions.
- 6.1.4. Software Deployment Configuration.
- 6.2. Landing Page, Services & Applications Implementation.
 - 6.2.1. Sprint 1
 - 6.2.1.1. Sprint Planning 1.
 - 6.2.1.2. Aspect Leaders and Collaborators.
 - 6.2.1.3. Sprint Backlog 1.
 - 6.2.1.4. Development Evidence for Sprint Review
 - 6.2.1.5. Testing Suite Evidence for Sprint Review.
 - 6.2.1.6. Execution Evidence for Sprint Review.
 - 6.2.1.7. Services Documentation Evidence for Sprint Review
 - 6.2.1.8. Software Deployment Evidence for Sprint Review
 - 6.2.1.9. Team Collaboration Insights during Sprint.
 - 6.2.2. Sprint 2
 - 6.2.2.1. Sprint Planning 2.
 - 6.2.2.2. Aspect Leaders and Collaborators.
 - 6.2.2.3. Sprint Backlog 2.
 - 6.2.2.4. Development Evidence for Sprint Review
 - 6.2.2.5. Testing Suite Evidence for Sprint Review.
 - Alcance del Testing
 - Unit Tests
 - Integration Tests
 - Acceptance Tests (BDD)
 - Repositorio y Commits de Testing
 - Resumen de Cobertura
 - 6.2.2.6. Execution Evidence for Sprint Review.
 - 6.2.2.7. Services Documentation Evidence for Sprint Review
 - 6.2.2.8. Software Deployment Evidence for Sprint Review
 - 6.2.2.9. Team Collaboration Insights during Sprint.
 - 6.2.3. Sprint 3
 - 6.2.3.1. Sprint Planning 3.
 - 6.2.3.2. Aspect Leaders and Collaborators.
 - 6.2.3.3. Sprint Backlog 3.
 - 6.2.3.4. Development Evidence for Sprint Review.
 - 6.2.3.5. Testing Suite Evidence for Sprint Review.
 - Alcance del Testing
 - Unit Tests
 - Integration Tests
 - Acceptance Tests (BDD)
 - Repositorio y Commits de Testing
 - Resumen de Cobertura
 - 6.2.3.6. Execution Evidence for Sprint Review.
 - 6.2.3.7. Services Documentation Evidence for Sprint Review
 - 6.2.3.8. Software Deployment Evidence for Sprint Review
 - 6.2.3.9. Team Collaboration Insights during Sprint.
- 6.3. Validation Interviews.
 - 6.3.1. Diseño de Entrevistas.
 - 1. Preguntas generales:
 - 2. Preguntas - Segmento: Clientes Finales (Consumidores finales)
 - 3. Preguntas - Segmento: Empresa (Gestores de transporte)
 - 6.3.2. Registro de Entrevistas.
 - Segmento 1: Empresa
 - Segmento 2: Clientes finales
 - 6.3.3. Evaluaciones según heurísticas
- UX Heuristics & Principles Evaluation
 - SITE o APP A EVALUAR
 - TAREAS A EVALUAR
 - ESCALA DE SEVERIDAD
 - TABLA RESUMEN
- DESCRIPCIÓN DE PROBLEMAS
 - PROBLEMA 1: Botón de eliminar vehículo no funciona
 - Problema
 - Recomendación
 - PROBLEMA 2: Botón "Ver más detalles" en alertas no muestra información
 - Problema
 - Recomendación
 - PROBLEMA 3: Inconsistencia de idioma en interfaz
 - Problema
 - Recomendación
 - PROBLEMA 4: Texto truncado en filtro "Created Between"

- Problema
- Recomendación
- PROBLEMA 5: Inconsistencia de idioma en dropdown de estado de viajes
 - Problema
 - Recomendación
 - 6.4. Video About-the-Product.
- Conclusiones
 - Conclusiones y recomendaciones
 - Conclusiones
 - Recomendaciones
 - Video About-the-Team
- Bibliografía
- Anexos
 - Anexo A Artefactos visuales
 - Anexo B Evidencias Técnicas de Implementación
 - Anexo C Videos de Entrevistas
 - Anexo D Videos de Exposiciones

Student Outcome

El curso contribuye de manera directa al desarrollo y cumplimiento del Student Outcome 5 definido por ABET – EAC, asegurando que los estudiantes alcancen las competencias establecidas en dicho resultado.

Criterio: La capacidad de funcionar efectivamente en un equipo cuyos miembros juntos proporcionan liderazgo, crean un entorno de colaboración e inclusivo, establecen objetivos, planifican tareas y cumplen objetivos. En el siguiente cuadro se describe las acciones realizadas y enunciados de conclusiones por parte del grupo, que permiten sustentar el haber alcanzado el logro del ABET – EAC - Student Outcome 5.

Criteriono Específico	Acciones Realizadas por Miembro (por Avance)	Conclusiones del Equipo
Trabaja en equipo para proporcionar liderazgo conjunto	<p>Marcelo Fabian Garro Vega</p> <p>TB1:</p> <p>Durante este avance del proyecto, participé activamente en reuniones presenciales y virtuales, donde contribuí con ideas clave para definir el flujo de negocio, fomentando así la claridad y el consenso entre todos los miembros del equipo. También participé activamente en la elaboración de la arquitectura de software y en el diseño táctico siguiendo los principios de Domain Driven Design.</p> <p>TP1:</p> <p>En esta entrega, demostré liderazgo colaborativo al delegar tareas SMART a mis compañeros, guiarlos en la resolución de dudas técnicas y establecer un flujo de trabajo transparente que promovió la coordinación y el cumplimiento de los objetivos del equipo.</p> <p>TB2:</p> <p>En esta entrega me encargué del desarrollo completo del bounded context de Trips, definiendo sus entidades, agregados y flujos principales. Implementé la lógica central del dominio y aseguré su correcta integración con los demás módulos del sistema.</p> <p>TF:</p> <p>En esta entrega consolidé los avances del equipo, validé la integración de los módulos desarrollados y aseguré que las funcionalidades cumplieran con los criterios del sprint, manteniendo coherencia técnica y alineación con el flujo de negocio.</p>	<p>TB1:</p> <p>A lo largo del proyecto asumimos un liderazgo compartido, distribuyendo responsabilidades y tomando decisiones en conjunto. Cada uno de nosotros aportó ideas y conocimientos que permitieron definir con claridad el flujo de negocio, la arquitectura y los artefactos necesarios bajo los principios de Domain Driven Design. Esta forma de trabajo nos ayudó a consolidar una visión común y avanzar con mayor seguridad en cada fase.</p> <p>TP1:</p> <p>En esta iteración demostramos un liderazgo conjunto enfocado en el diseño de la experiencia de usuario y la arquitectura de información. La colaboración entre los miembros permitió integrar los bounded contexts, desarrollar prototipos funcionales y construir un landing page coherente con la identidad del producto. El liderazgo compartido facilitó la toma de decisiones técnicas y visuales, asegurando que cada entregable cumpliera con los objetivos del sprint y contribuyera al avance del proyecto.</p> <p>TB2:</p> <p>En esta fase del proyecto continuamos aplicando un liderazgo colaborativo, orientado a la consolidación técnica de los componentes definidos previamente. Cada integrante asumió un rol activo en la validación del modelo de dominio, la revisión de los artefactos generados y la integración de los servicios necesarios para garantizar la coherencia entre los módulos. Este liderazgo compartido permitió identificar ajustes en la arquitectura, mejorar la comunicación entre los bounded contexts y asegurar que las decisiones tomadas respondieran a los objetivos del negocio. Gracias a esta dinámica de trabajo, avanzamos con mayor claridad y fortalecimos la base técnica para las siguientes iteraciones.</p>
George Arturo Aliaga Pimentel	<p>TB1:</p> <p>Durante este avance contribuí a mantener un entorno colaborativo al apoyar en la documentación del Domain Layer, describiendo de manera clara las entidades, value objects y servicios de dominio. Además, participé en la definición de metas de la iteración y en la organización de tareas, lo que permitió que el equipo trabajara con orden, alcanzara los objetivos propuestos y mantuviera consistencia con las guías establecidas por el líder del grupo.</p> <p>TP:</p> <p>Participé de manera activa en la planificación y diseño de los productos, coordinando con el equipo para definir y actualizar las correcciones pendientes de la entrega anterior.</p> <p>TB2:</p> <p>Participé en la organización y definición de los productos, trabajando de manera conjunta con el equipo para establecer y ajustar las correcciones pendientes de la entrega previa. Asimismo, se completaron adecuadamente todos los productos contemplados dentro del alcance establecido.</p> <p>TF:</p> <p>Me involucré activamente en la planificación y estructuración de los productos, trabajando codo a codo con el equipo para definir e implementar las correcciones identificadas en la entrega anterior. Para el sprint final, mi foco estuvo en la refactorización integral del código de los productos, garantizando la optimización y la calidad del código entregado.</p>	
Daniel Rodrigo Chirinos Arevalo	<p>TB1:</p> <p>Participé en reuniones virtuales y presenciales, aportando ideas sobre cómo abordar el proyecto y colaborando con mis compañeros para tomar decisiones clave respecto al caso propuesto. Además, elaboré de manera colaborativa los user stories y los diagramas de clase y C4. Para ello, realizamos reuniones de planificación y coordinación para asegurar el acuerdo sobre el desarrollo de estos diagramas.</p> <p>TP:</p> <p>Trabajé activamente en la planificación y estructuración de los productos, colaborando con los miembros del equipo para poder definir y actualizar las correcciones que se dejaron en la anterior entrega. Además, se realizó la primera versión de la web application y la landing page.</p>	

Criterio Específico	Acciones Realizadas por Miembro (por Avance)	Conclusiones del Equipo
	<p>TB2:</p> <p>Trabajé activamente en la planificación y estructuración de los productos, colaborando con los miembros del equipo para poder definir y actualizar las correcciones que se dejaron en la anterior entrega. Además, se realizaron correctamente todos los productos que están dentro de nuestro alcance</p> <p>TF:</p> <p>Trabajé activamente en la planificación y estructuración de los productos, colaborando con los miembros del equipo para poder definir y actualizar las correcciones que se dejaron en la anterior entrega. Como ultimo sprint, se refactorizó los códigos de los productos.</p>	
	<p>Alessandro Joaquin Bernardo Eusebio</p> <p>TB1:</p> <p>Durante este avance del proyecto, participé activamente en reuniones presenciales y virtuales, aportando ideas clave para definir el flujo de negocio y la estructura de la solución. Colaboré en la elaboración de la arquitectura de software y en el diseño táctico bajo los principios de Domain Driven Design, asegurando claridad y consenso entre los miembros del equipo. Además, contribuí en la documentación de los artefactos de usuario y en la organización de tareas, promoviendo un entorno colaborativo y alineado con los objetivos del proyecto.</p> <p>TP1:</p> <p>Durante este avance, asumí un rol activo de liderazgo en la elaboración del bounded context de Visualization and Analytics, contribuyendo con propuestas técnicas y coordinando con el equipo para asegurar su correcta integración con los demás contextos del sistema. Además, lideré el diseño y desarrollo de los prototipos en Figma, tanto para la versión web como móvil, garantizando una experiencia de usuario coherente y alineada con los objetivos del proyecto. También participé activamente en la construcción del landing page, colaborando en decisiones de estructura, diseño visual y optimización para conversión.</p> <p>TB2:</p> <p>En esta iteración continué ejerciendo liderazgo colaborativo en el desarrollo del bounded context de Visualization and Analytics. Coordiné con el equipo la definición de la arquitectura de datos y la estructura de los servicios analíticos, asegurando su correcta integración con los bounded contexts de Monitoring, Alerts y Trips. Participé activamente en reuniones técnicas para tomar decisiones sobre el diseño de componentes, la selección de tecnologías de visualización y la estrategia de agregación de datos. Además, lideré sesiones de revisión de código y validación de arquitectura, promoviendo buenas prácticas de desarrollo y garantizando la coherencia del sistema con los principios de Domain-Driven Design.</p> <p>TF:</p> <p>Para la entrega final, demostré liderazgo técnico integral al coordinar la implementación completa del bounded context de Visualization and Analytics a través de toda la arquitectura del sistema. Lideré la definición de estándares de desarrollo para los tres niveles (backend, frontend web y móvil), asegurando consistencia en el código y alineación con los patrones arquitectónicos establecidos. Coordiné sesiones de sincronización con los responsables de otros bounded contexts para garantizar una integración fluida de los datos analíticos. Además, guié al equipo en la resolución de desafíos técnicos complejos relacionados con el procesamiento de grandes volúmenes de datos en tiempo real, la optimización de consultas y la implementación de visualizaciones interactivas. Este liderazgo colaborativo permitió entregar una solución analítica robusta, escalable y completamente funcional que cumplió con todos los objetivos técnicos y de negocio del proyecto.</p>	
	<p>Vera Nuñez Nicolas Alejandro</p> <p>TB1:</p> <p>Participé de manera activa en reuniones virtuales y presenciales, aportando propuestas relevantes para la definición de la arquitectura de software del Fleet Management. Lideré la elaboración de los diagramas C4 (contexto, contenedores y componentes) y colaboré en el diseño táctico bajo los principios de Domain Driven Design (DDD). Asimismo, contribuí en la identificación y documentación de los Bounded Contexts, garantizando una delimitación clara y consistente del sistema, y elaboré el Bounded Context Canvases, facilitando la comprensión de las responsabilidades y alcances de cada contexto.</p> <p>TP:</p> <p>Durante el trabajo parcial asumí un rol de liderazgo colaborativo en el avance del bounded context de Fleet para el frontend. Coordiné con el equipo la definición de las vistas principales para la gestión de vehículos y dispositivos, revisé los flujos de navegación y apoyé en la integración inicial con la API simulada. Este liderazgo compartido permitió que el módulo de Fleet Web avanzara con una arquitectura consistente y alineada a los</p>	

Criterio Específico	Acciones Realizadas por Miembro (por Avance)	Conclusiones del Equipo
	<p>principios de DDD y a los objetivos del sprint.</p> <p>TB2:</p> <p>En esta iteración lideré, de manera conjunta con mis compañeros, el avance del bounded context de Fleet en el backend y la aplicación móvil. Colaboré en la definición y revisión de los endpoints REST para la gestión de vehículos y dispositivos, así como en la alineación del modelo de dominio entre backend y mobile. Además, apoyé en la coordinación de las tareas para que las pantallas móviles de Fleet reflejaran correctamente los casos de uso definidos. Este liderazgo colaborativo ayudó a consolidar la coherencia del bounded context de Fleet en múltiples capas de la solución.</p> <p>TF:</p> <p>Para la entrega final ejercí un liderazgo integrado al trabajar en los bounded contexts de IAM y Fleet en los tres niveles de la solución (frontend, backend y mobile). Coordiné con el equipo la definición y validación de los flujos de autenticación y autorización, así como la integración de IAM con los módulos de Fleet. Además, apoyé la priorización de tareas y la revisión de Pull Requests, asegurando que las implementaciones en web, backend y móvil mantuvieran consistencia arquitectónica, buenas prácticas de diseño y cumplimiento de los objetivos de negocio planteados para el proyecto CargaSafe.</p>	
	<p>Sanchez Ignacio, Jefrey Martin</p> <p>TP:</p> <p>Durante esta etapa asumí un rol más activo en la coordinación del equipo. Ayudé a tomar decisiones sobre la estructura del Bounded Context Subscriptions. También estuve pendiente del avance en general, dando sugerencias y asegurándome de que todos sigamos las mismas pautas técnicas.</p>	
	<p>Sanchez Ignacio, Jefrey Martin</p> <p>TB2:</p> <p>Durante esta entrega asumí un rol de liderazgo colaborativo dentro del desarrollo del Bounded Context de Trips. Coordiné con mis compañeros la definición de la estructura del front y la creación de los endpoints necesarios para asegurar la comunicación adecuada con el backend. Además, apoyé en la toma de decisiones técnicas, brindando lineamientos sobre el diseño de componentes.</p>	
Crea un entorno colaborativo e inclusivo, establece metas, planifica tareas y cumple objetivos.	<p>Marcelo Fabian Garro Vega</p> <p>TB1:</p> <p>Durante el desarrollo del proyecto, me enfoqué en fomentar un entorno colaborativo e inclusivo dentro del equipo, promoviendo la participación activa de todos los miembros mediante el envío constante de retroalimentaciones constructivas. Contribuí a la estimación de story points y a la planificación de user y technical stories, considerando el valor que estas aportan al negocio. Para ello, tomé en cuenta las opiniones de mis compañeros, lo cual permitió definir metas claras, planificar de manera efectiva y cumplir con los objetivos propuestos bajo los principios de Domain Driven Design.</p> <p>TP1:</p> <p>En esta entrega, fomente un entorno colaborativo e inclusivo al asignar tareas del sprint backlog y coordinar sesiones de trabajo síncronas, tanto remotas como presenciales, para definir conjuntamente los objetivos y asegurar el cumplimiento de la entrega.</p> <p>TB2:</p> <p>En esta entrega contribuí a mantener un entorno colaborativo al coordinar la planificación del sprint y apoyar la definición de las historias del módulo de Trips. Trabajé de manera cercana con el equipo para aclarar requerimientos, alinear expectativas y asegurar que las tareas se distribuyeran de forma equilibrada para cumplir los objetivos planteados.</p> <p>TF1:</p> <p>En esta entrega reforzé la colaboración del equipo integrando los avances de cada miembro, verificando el cumplimiento de las metas del sprint y facilitando la comunicación para resolver bloqueos. Esto permitió cerrar la iteración con coherencia entre lo planificado y lo entregado.</p> <p>George Arturo Aliaga Pimentel</p> <p>TB1:</p> <p>Durante este avance contribuí a mantener un entorno colaborativo al apoyar en la documentación del Domain Layer, describiendo de manera clara las entidades, value objects y servicios de dominio. Además, participé en la definición de metas de la iteración y en la organización de tareas, lo que permitió que el equipo trabajara con orden, alcanzara los objetivos propuestos y mantuviera consistencia con las guías establecidas por el líder del grupo.</p>	<p>TB1:</p> <p>Durante el desarrollo creamos un entorno colaborativo e inclusivo en el que cada aporte fue escuchado y valorado. Establecimos metas claras, organizamos tareas de manera conjunta y planificamos los avances para asegurar el cumplimiento de los objetivos. Gracias a la comunicación constante y al compromiso colectivo, conseguimos mantener la cohesión del grupo, garantizar la calidad de los entregables y alcanzar los resultados propuestos.</p> <p>TP1:</p> <p>En esta iteración fortalecimos el entorno colaborativo mediante sesiones de trabajo conjunto en el diseño de prototipos y la definición de bounded contexts. Establecimos metas específicas relacionadas con la experiencia de usuario, la arquitectura de información y el desarrollo del landing page. La planificación conjunta y la retroalimentación constante nos permitieron cumplir con los objetivos establecidos, asegurando que cada componente (prototipos, bounded contexts, landing page) estuviera alineado con la visión del producto y las expectativas de los usuarios finales.</p> <p>TB2:</p> <p>En esta etapa reforzamos el trabajo colaborativo enfocándonos en la revisión, validación y mejora de los artefactos técnicos desarrollados previamente. Establecimos metas orientadas al refinamiento del modelo de dominio, la coherencia entre los bounded</p>

Criterio Específico	Acciones Realizadas por Miembro (por Avance)	Conclusiones del Equipo
	<p>TP1:</p> <p>En esta iteración reforzamos el trabajo en equipo mediante sesiones colaborativas para diseñar prototipos y definir los bounded contexts. Acordamos metas claras sobre la experiencia de usuario, la arquitectura de información y el desarrollo de la landing page. Gracias a la planificación conjunta y la retroalimentación continua, logramos cumplir los objetivos y mantener todos los componentes alineados con la visión del producto y las necesidades del usuario final.</p>	<p>contexts y la integración de los componentes clave del sistema. La comunicación continua y las sesiones de retroalimentación nos permitieron identificar ajustes necesarios, resolver dudas técnicas y asegurar que cada avance estuviera alineado con los objetivos del proyecto. Gracias al compromiso colectivo y a la organización conjunta de las tareas, logramos consolidar una base más sólida.</p>
	<p>TB2:</p> <p>En esta fase fortalecimos el trabajo en equipo revisando, validando y mejorando los artefactos técnicos previos. Nos enfocamos en refinar el modelo de dominio, asegurar la coherencia entre los bounded contexts y avanzar en la integración de los componentes principales. Gracias a la comunicación constante y la retroalimentación del equipo, logramos identificar ajustes, resolver dudas y mantener cada avance alineado con los objetivos del proyecto, consolidando así una base más sólida.</p>	
	<p>TF:</p> <p>En esta fase, el equipo fortaleció la base técnica mediante la validación y mejora continua de los artefactos previos, con un foco estratégico en el refinamiento del modelo de dominio y la coherencia entre los bounded contexts.</p>	
	<p>Daniel Rodrigo Chirinos Arevalo</p> <p>TB1:</p> <p>Estuve enfocado en fomentar un ambiente de colaboración dentro del equipo. Trabajé en conjunto con mis compañeros para definir los objetivos y tareas del proyecto. Colaboré en el análisis competitivo y en la definición de segmentos clave como el de gestión de transportes y clientes finales. Además, me encargué de crear user stories e impact mapping para estos segmentos, asegurando que el desarrollo del producto estuviera alineado con las necesidades del usuario. Cada semana establecí un límite personal para realizar las secciones de cada capítulo, de esta forma me aseguré de que todas las tareas estuvieran bien organizadas y distribuidas para cumplir con los objetivos establecidos.</p> <p>TP:</p> <p>Colaboré en la elaboración de mi bounded context "Alerts and resolutions", también apoyé con las correcciones de la anterior entrega y en la organización de las tareas.</p> <p>TB2:</p> <p>Colaboré en la elaboración de mi bounded context "Alerts and resolutions" y en la embedded application, también apoyé con las correcciones de la anterior entrega y en la organización de las tareas.</p> <p>TF:</p> <p>Colaboré en la refactorización del código del embedded application y ajustes en los productos, también apoyé con las correcciones de la anterior entrega y en la organización de las tareas.</p>	
	<p>Alessandro Joaquin Bernardo Eusebio</p> <p>TB1: Durante el desarrollo del proyecto, me enfoqué en fomentar un entorno colaborativo e inclusivo, promoviendo la participación activa de todos los miembros mediante retroalimentación constante y constructiva. Contribuí a la estimación de story points y a la planificación de user y technical stories, considerando el valor que aportan al negocio. Participé en la definición de metas y organización de tareas, asegurando que el equipo trabajara con orden y alcanzara los objetivos propuestos bajo los principios de Domain Driven Design.</p> <p>TP1: Durante este avance, establecí metas claras de diseño y prototipado, planificando las tareas técnicas relacionadas con la elaboración del bounded context de Visualization and Analytics y la creación de los prototipos interactivos. Fomenté un ambiente inclusivo mediante sesiones de revisión colaborativa del diseño en Figma, donde recibí y di retroalimentación constructiva para mejorar la experiencia de usuario. Asimismo, coordiné con el equipo para asegurar que el landing page cumpliera con los estándares de usabilidad, accesibilidad y alineación con la identidad visual del proyecto, cumpliendo con los objetivos establecidos en el sprint.</p> <p>TB2: En esta iteración continué con el desarrollo del bounded context de Visualization and Analytics, enfocándome en la definición de la estructura de datos y la integración con los demás contextos del sistema. Colaboré activamente en la planificación de las tareas técnicas del sprint, asegurando la coherencia entre los componentes del backend y la preparación de los servicios necesarios para la generación de reportes y métricas. Además, participé en sesiones de revisión técnica con el equipo para validar la arquitectura propuesta y garantizar su alineación con los principios de Domain-Driven Design y los objetivos del proyecto.</p>	

Criterio Específico	Acciones Realizadas por Miembro (por Avance)	Conclusiones del Equipo
	<p>TF: Para la entrega final del proyecto, lideré la integración completa del bounded context de Visualization and Analytics en los tres niveles de la arquitectura del sistema. En el backend, desarrollé e implementé los servicios y endpoints necesarios para la agregación y procesamiento de datos analíticos provenientes de múltiples bounded contexts (Monitoring, Alerts, Trips), garantizando la coherencia y disponibilidad de la información. En el frontend web, diseñé e implementé componentes interactivos para la visualización de KPIs, gráficos en tiempo real y dashboards personalizables, integrando librerías especializadas como Chart.js y asegurando una experiencia de usuario fluida y responsive. En la aplicación móvil, adapté las visualizaciones para dispositivos móviles, implementando gráficos optimizados y métricas clave accesibles offline mediante sincronización con la base de datos local SQLite. Esta integración transversal permitió consolidar una capa de analítica robusta, escalable y alineada con los principios de Domain-Driven Design, proporcionando valor agregado tanto a operadores logísticos como a clientes finales mediante insights accionables y reportes en tiempo real.</p>	
	<p>Vera Nuñez Nicolas Alejandro</p> <p>TB1:</p> <p>Promoví un entorno colaborativo mediante la coordinación de los artefactos de arquitectura y la integración de los diagramas en el informe. Contribuí en la planificación de tareas técnicas vinculadas a la arquitectura de software, asegurando la consistencia de los entregables y su alineación con los objetivos del proyecto. Además, facilité retroalimentación constructiva, incentivando la participación activa de todos los miembros del equipo.</p>	
	<p>TP:</p> <p>Durante el trabajo parcial contribuí a crear un entorno colaborativo en torno al avance del bounded context de Fleet para el frontend. Participé en la planificación de las tareas de desarrollo de las vistas de vehículos y dispositivos, ayudé a descomponer el trabajo en historias manejables y coordiné con mis compañeros para integrar correctamente los servicios del cliente HTTP. Esta organización conjunta y el intercambio constante de feedback permitieron que el módulo de Fleet Web cumpliera los objetivos del sprint con una base técnica clara.</p>	
	<p>TB2:</p> <p>En esta iteración fomenté un entorno colaborativo al trabajar con el equipo en el avance del bounded context de Fleet en el backend y la aplicación móvil. Participé en la coordinación de las tareas para la definición de endpoints, la validación del modelo de dominio y la integración entre las APIs y las pantallas móviles. Además, apoyé en la revisión de avances y en la resolución de dudas técnicas, lo que ayudó a que las metas del sprint se cumplieran de forma alineada entre las distintas capas de la solución.</p>	
	<p>TF:</p> <p>Para la entrega final del proyecto, contribuí a un entorno colaborativo integrando los bounded contexts de IAM y Fleet en el frontend, backend y mobile. Ayudé a planificar las tareas relacionadas con autenticación, autorización y gestión de flota, coordinando con mis compañeros para asegurar una correcta integración entre servicios y aplicaciones. Asimismo, participé en la revisión de código y en la verificación de los flujos completos (login, gestión de vehículos y dispositivos), garantizando que los objetivos técnicos y de negocio se cumplieran de manera coherente en todas las plataformas.</p>	
	<p>Sanchez Ignacio, Jefrey Martin</p> <p>TP:</p> <p>En este trabajo parcial participé en la parte del Bounded Context de CargaSafe, ayudando en la conexión entre la vista y la API con json-server. Me encargué de implementar componentes con Angular Material y asegurar que el diseño se mantenga ordenado y coherente con el resto del proyecto.</p>	
	<p>Sanchez Ignacio, Jefrey Martin</p> <p>TB2:</p> <p>En esta iteración contribuí activamente a fortalecer un entorno colaborativo al participar en la planificación del trabajo del Bounded Context de Trips. Aporté en la definición de metas técnicas, como la creación de los endpoints necesarios y la implementación del front asociada, asegurando que el flujo de información fuera claro entre la API y la interfaz. Trabajé de manera coordinada con el equipo, apoyando en la organización de tareas y manteniendo comunicación constante para resolver dudas y evitar bloqueos. Gracias a esta</p>	

Criterio Específico	Acciones Realizadas por Miembro (por Avance)	Conclusiones del Equipo
	planificación conjunta y al seguimiento continuo, logramos cumplir los objetivos planteados y avanzar de forma alineada con la arquitectura general del proyecto.	

Capítulo I: Introducción

1.1. Startup Profile

1.1.1. Descripción de la Startup

Los Parkers es una startup innovadora, especializada en el desarrollo de soluciones tecnológicas de monitoreo y trazabilidad para el sector logístico y de transporte. Surge de la iniciativa de un equipo multidisciplinario de estudiantes de la Universidad Peruana de Ciencias Aplicadas, quienes comparten una visión común: transformar la manera en que se gestionan y se supervisan las cadenas de suministro de productos sensibles, utilizando tecnología IoT.

Los Parkers se distingue por su enfoque centrado en el usuario, trabajando de forma colaborativa con profesionales del ámbito de la logística para diseñar plataformas intuitivas, inteligentes y adaptadas a las necesidades reales de la industria. Su equipo combina conocimientos técnicos de vanguardia en IoT con una comprensión profunda de los desafíos actuales en materia de cadena de frío, trazabilidad de productos y gestión de riesgos.

Entre sus principales productos destaca CargaSafe, una solución integral que permite a las empresas de transporte y a sus clientes monitorear en tiempo real las condiciones de sus cargas, generando reportes y alertas automáticas respaldadas por tecnología inteligente. Los Parkers emplea metodologías ágiles y tecnologías modernas para garantizar que sus soluciones sean robustas, escalables y capaces de evolucionar con las necesidades del mercado. Además, la empresa ofrece soporte continuo y mejora constante a sus plataformas, priorizando siempre la experiencia del usuario.

Visión: La visión de Los Parkers es convertirse en líder global en el desarrollo de soluciones tecnológicas aplicadas a la logística y la cadena de suministro, empoderando a las empresas para que tomen decisiones más eficientes, informadas y seguras.

Misión: La misión de Los Parkers es diseñar y desarrollar herramientas digitales innovadoras, accesibles y personalizadas que contribuyan a mejorar la eficiencia, la transparencia y la seguridad en el transporte de mercancías

1.1.2. Perfiles de integrantes del equipo

**Nombre y Apellido:**

Marcelo Fabian Garro Vega

Código:

u20201c410

Carrera:

Ingeniería de Software

Acerca de:

Soy un estudiante de 22 años apasionado por el desarrollo web y móvil, con un enfoque especial en la experiencia de usuario y el diseño de interfaces (UX/UI). Me especializo en el desarrollo Frontend, siempre buscando crear aplicaciones intuitivas y funcionales. Mi objetivo es aportar tanto en el ámbito creativo como en el técnico, para contribuir al diseño y construcción de soluciones de software que generen impacto real.

**Nombre y Apellido:**

Jefrey Martin Sanchez Ignacio

Código:

U202113324

Carrera:

Ingeniería de Software

Acerca de:

Actualmente cursando el octavo ciclo de mi carrera, Soy una persona responsable, proactiva. Espero aprender mucho del curso y sobretodo de este proyecto.

**Nombre y Apellido:**

George Arturo Aliaga Pimentel

Código:

U20211c273

Carrera:

Ingeniería de Software

Acerca de:

Me llamo George Arturo Aliaga Pimentel y soy de la carrera de Ingeniería de Software. Estoy cursando el séptimo ciclo. Me considero una persona cooperativa y responsable, lo cual es totalmente requerido para un proyecto grupal. Tengo la meta de ser un gran ingeniero y espero mejorar cada día para lograr un buen desempeño en la carrera.

Nombre y Apellido:

Nicolas Alejandro Vera Nuñez

Código:

u202214869



Carrera:
Ingeniería de Software

Acerca de:

Soy estudiante de la UPC, tengo 21 años. Estoy en la carrera de Ingeniería de Software, ya que, siempre me gustó la tecnología, el análisis de datos y el desarrollo web. Estoy cursando el 7mo ciclo de la carrera y mis habilidades son Angular, Java con Springboot, Docker, Python, HTML y JavaScript. También soy una persona analítica y que encuentra soluciones innovadoras a varios problemas.



Nombre y Apellido:
Daniel Rodrigo Chirinos Arevalo

Código:
202019577

Carrera:
Ingeniería de Software

Acerca de:

Me llamo Daniel Chirinos Arevalo y soy de la carrera de Ingeniería de Software. Mi experiencia en la universidad es bastante normal y me encuentro cursando el sexto ciclo. A lo largo del tiempo, sí que tuve algunas dificultades con algunos cursos, pero logré enfrentarlos. Mi habilidad es ser líder en algunas ocasiones, trabajar con tiempo y lo más importante, ser siempre amistoso con el equipo.

Nombre y Apellido:
Bernardo Eusebio Alessandro Joaquin

Código:
u202113640

Carrera:
Ingeniería de software

Acerca de:

Soy Alessandro Joaquin Bernardo Eusebio, estudiante de la carrera de Ingeniería de software. Actualmente estoy bastante orgulloso del progreso que tengo actualmente aprendiendo tanto dentro de la facultad como en mi experiencia autónoma y en el trabajo. Me inclino más a ser un analista de datos y ampliar mis conocimientos sobre Machine Learning, modelos predictivos y AWS.

1.2. Solution Profile

1.2.1. Antecedentes y problemática

What (¿Qué?)

El transporte de mercancías enfrenta un reto transversal: monitorear de forma continua y verificable las condiciones reales del cargamento durante todo el trayecto. Más allá del simple rastreo de ubicación, las cargas pueden sufrir manipulaciones no autorizadas, aperturas de puerta, golpes, vibraciones, humedad, exposición a ambientes no controlados, desvíos de ruta, retrasos y pérdidas de custodia. La ausencia de evidencia objetiva y en tiempo real sobre estos eventos se traduce en daños, mermas, disputas entre actores de la cadena y costos operativos crecientes. En la práctica, gran parte de la visibilidad actual se limita a hitos administrativos (salida/arribo) o a sistemas aislados de geolocalización. Esto deja zonas ciegas respecto al estado físico y a la integridad del embalaje en tramos críticos (esperas, transbordos y handoffs entre operadores). Para retail, agro, farmacéutico, pesquero y consumo masivo, contar con telemetría de condiciones del cargamento—incluyendo eventos de integridad, incidencias y evidencias auditables—es ya un requisito para reducir pérdidas, acelerar conciliaciones y sostener SLA con clientes y aseguradoras.

Who (¿Quién?)

Este problema impacta a dos actores clave:

1. Empresas transportistas y de logística: Enfrentan el riesgo financiero de pérdidas de carga, reclamos de clientes y el daño a su reputación por entregas fallidas.
2. El cliente final: Se ve afectado al recibir productos en mal estado, caducados o, en el caso de medicamentos, que han perdido su efectividad, lo cual representa un riesgo para la salud y la seguridad.

Where (¿Dónde?)

La problemática se manifiesta a lo largo de toda la cadena de suministro, desde el almacén de origen hasta la entrega final. Es particularmente crítica en los tramos de larga distancia (transporte terrestre, aéreo y marítimo) y en los "puntos de transferencia" entre diferentes vehículos o almacenes, donde la supervisión manual es más difícil. La adopción de tecnologías de monitoreo es una tendencia global que se acelera en mercados con infraestructura logística desarrollada y una creciente demanda de comercio electrónico.

When (¿Cuándo?)

La necesidad de visibilidad en tiempo real se ha intensificado desde la pandemia de COVID-19, la cual puso de manifiesto la vulnerabilidad de las cadenas de suministro. El aumento del transporte de productos médicos y la expectativa de los consumidores por entregas rápidas y transparentes han impulsado la demanda de soluciones tecnológicas. Hoy en día, la mayoría de los clientes esperan poder rastrear sus pedidos en tiempo real, lo que convierte la visibilidad de la carga en un estándar de mercado, no solo una ventaja competitiva (Perfect Planner, 2025).

Why (¿Por qué?)

La principal causa de esta problemática es la falta de información oportuna. Las empresas no tienen acceso a datos críticos sobre la temperatura, ubicación o condiciones de su carga en el momento en que ocurren las desviaciones. Esto impide la toma de acciones correctivas inmediatas, como ajustar el termostato de un camión, cambiar una ruta o notificar al cliente sobre un posible retraso. Sin esta visibilidad, los problemas solo se descubren al final del trayecto, cuando ya es demasiado tarde para evitar la pérdida del producto.

How (¿Cómo?)

Actualmente, el monitoreo se realiza con métodos inefficientes o no integrados. Muchas empresas aún dependen de registradores de datos manuales que requieren ser revisados al final del viaje o utilizan múltiples sistemas (GPS para ubicación, sensores para temperatura) que no se comunican entre sí. Esta fragmentación reduce la eficiencia operativa y aumenta el riesgo de errores humanos. La falta de una plataforma integral que centralice toda la información limita la capacidad de las empresas para optimizar sus rutas, gestionar riesgos y, en última instancia, ofrecer un servicio de alta calidad (Bogdanov, 2024).

How much (¿Cuánto?)

El impacto de la falta de un monitoreo efectivo es funcional, operativo y estratégico. Las empresas pierden tiempo y recursos reubicando información o lidiando con problemas logísticos que podrían haberse evitado. Operativamente, esta deficiencia se traduce en mayores costos de seguro y en gastos asociados al desperdicio de productos. Desde una perspectiva de negocio, esta brecha representa una oportunidad clara para monetizar al ofrecer una solución de valor que mejore la fidelización del cliente y construya una reputación de confiabilidad, lo que constituye una ventaja competitiva en el mercado.

1.2.2. Lean UX Process

1.2.2.1. Lean UX Problem Statements

Domain

El proyecto se enmarca en el sector logístico y de transporte de carga, específicamente en el control y monitoreo de condiciones de mercancías sensibles durante su traslado. En este dominio, la información en tiempo real sobre el estado de la carga resulta esencial para garantizar la trazabilidad, la seguridad y el cumplimiento de normativas relacionadas con la cadena de frío, productos farmacéuticos y bienes perecibles.

Customer Segments

Empresas de transporte y operadores logísticos: responsables de gestionar flotas y garantizar la integridad de los productos durante el traslado.

Clientes corporativos o distribuidores de productos sensibles: demandan transparencia, control y evidencia del cumplimiento de condiciones óptimas de transporte.

Estos segmentos comparten la necesidad de contar con información precisa, en tiempo real, y herramientas que les permitan actuar de forma proactiva ante incidentes.

Pain Points

Falta de visibilidad en tiempo real de las condiciones de temperatura, humedad y localización de la carga.

Pérdida económica por ruptura de la cadena de frío o manipulación inadecuada.

Comunicación deficiente entre transportistas y clientes frente a incidentes.

Ausencia de trazabilidad digital que permita auditar condiciones y responsabilidades.

Procesos manuales y fragmentados que dificultan la gestión operativa y la toma de decisiones.

Gap

Actualmente, las empresas del sector utilizan herramientas aisladas —como GPS o registradores de temperatura— que no ofrecen una integración completa entre monitoreo, trazabilidad y comunicación. Existe una brecha entre la información disponible y la capacidad de reaccionar ante eventos críticos, lo que impide una gestión eficiente y preventiva.

Vision / Strategy

La visión de Los Parkers con CargaSafe es transformar el monitoreo logístico tradicional en un proceso inteligente, preventivo y conectado. La estrategia consiste en desarrollar una plataforma IoT integral, donde sensores embarcados recopilan datos de las condiciones ambientales y los transmiten a un sistema central que analiza, alerta y visualiza la información en tiempo real. De esta manera, las empresas podrán anticiparse a riesgos, reducir pérdidas y aumentar la confianza de sus clientes mediante reportes automáticos y trazabilidad verificable.

Initial Segment

El proyecto iniciará su implementación con empresas de transporte de productos perecibles y farmacéuticos que operan en Lima Metropolitana. Este segmento fue seleccionado por su alta sensibilidad a las condiciones ambientales y su interés demostrado en la digitalización de procesos logísticos, representando un entorno ideal para validar la propuesta tecnológica y de negocio.

Resumen del enfoque: Este conjunto de Problem Statements establece las bases para las siguientes etapas del Lean UX Process, en las que se derivarán las Assumptions, Hypothesis Statements y el Lean UX Canvas, alineando los objetivos de negocio con las necesidades reales de los usuarios.

1.2.2.2. Lean UX Assumptions

Business Assumptions

1. Creemos que existe demanda significativa en transporte LATAM por monitoreo accesible con múltiples parámetros (frío, vibración, ubicación, energía). Validamos cuando ≥ 15 entrevistas B2B y $\geq 30\%$ de respuestas en encuesta indiquen "intención de evaluar" o mayor.
2. Creemos que las empresas pagan una suscripción mensual por una solución que reduce pérdidas y mejora confianza del cliente. Validamos cuando obtenemos ≥ 5 cartas de intención con rango de precio mensual especificado.
3. Creemos que dispositivos IoT de bajo costo alcanzan la precisión necesaria para cadena de frío, vibración, ubicación y energía. Validamos cuando las pruebas de campo muestran ± 0.5 °C en temperatura, detección de vibración y GPS dentro de 10 m en $\geq 95\%$ de lecturas.
4. Creemos que el modelo de suscripción resulta más atractivo que licencias perpetuas en el segmento meta. Validamos cuando el $\geq 70\%$ de decisiones simuladas o cotizaciones reales eligen suscripción sobre perpetuo, dado el mismo alcance.
5. Creemos que incluir parámetros adicionales (humedad, vibración, volcado) eleva el valor percibido. Validamos cuando las pruebas de pricing A/B muestran disposición a pagar $\geq 15\%$ más por el paquete multiparámetro.

Business Outcome Assumptions

1. Creemos que reducir incidentes de cadena de frío disminuye costos de merma para clientes. Validamos cuando los conductores muestran $\geq 20\%$ menos eventos críticos/mes vs. línea base.
2. Creemos que visibilidad en tiempo real acorta el ciclo de cobro por menos disputas. Validamos cuando el DSO (days sales outstanding) baja $\geq 10\%$ en 2 meses de uso.
3. Creemos que el producto acelera ventas en el segmento objetivo. Validamos cuando la conversión piloto pago es $\geq 40\%$ y ciclo de venta ≤ 60 días.
4. Creemos que el servicio retiene cuentas con valor sostenido. Validamos cuando la retención mensual $\leq 3\%$ y NRR (net revenue retention) $\geq 100\%$ a 6 meses.
5. Creemos que la propuesta escala con márgenes sanos. Validamos cuando margen bruto de servicio IoT $\geq 60\%$ a partir de 50 dispositivos activos/cliente.

User Assumptions

1. Creemos que los usuarios principales son gerentes de operaciones y conductores de transporte de productos sensibles. Validamos cuando el $\geq 80\%$ de entrevistas y sesiones sombra confirman estos roles como usuarios frecuentes.
2. Creemos que los usuarios necesitan alertas inmediatas ante ruptura de frío, vibración excesiva o volcado para actuar. Validamos cuando el $\geq 70\%$ prioriza "alertas en tiempo real" en ejercicios de priorización (MoSCoW/stack ranking).
3. Creemos que los usuarios quieren un dashboard simple con estado de toda la flota en una sola vista. Validamos cuando las pruebas de usabilidad logran "localizar vehículo en riesgo" en < 10 s por $\geq 80\%$ de participantes.
4. Creemos que los usuarios valoran reportes automáticos que integran todos los parámetros para sus clientes. Validamos cuando el $\geq 60\%$ selecciona "reportes automáticos" dentro de su top-3 beneficios en encuesta.
5. Creemos que los usuarios prefieren baja necesidad de capacitación frente a mayor complejidad funcional. Validamos cuando el onboarding autoguiado (sin formación formal) obtiene SUS ≥ 70 y tareas clave completadas por $\geq 80\%$ en la primera sesión.

User Outcomes & Benefits Assumptions

1. Creemos que con alertas y vista unificada, los equipos responden más rápido a eventos críticos. Validamos cuando tiempo de respuesta medio baja $\geq 30\%$ vs. línea base en 4 semanas.
2. Creemos que con práctica operativa guiada, los conductores cometen menos incidencias (puertas abiertas, detenciones no planificadas). Validamos cuando incidencias por 1,000 km bajan $\geq 15\%$ en 8 semanas.
3. Creemos que con reportes automáticos, los gerentes ahorran tiempo en auditorías y atención de reclamos. Validamos cuando tiempo semanal en compilación/reportes baja $\geq 50\%$ medido por time-tracking.
4. Creemos que con trazabilidad histórica, aumenta la tasa de auditorías aprobadas. Validamos cuando tasa de auditoría/cliente sube ≥ 10 pp en el primer trimestre.
5. Creemos que con visibilidad y control, mejora la confianza del cliente final. Validamos cuando NPS de clientes finales sube ≥ 10 puntos a los 3 meses.

Feature Assumptions

1. Creemos que el monitoreo en tiempo real de temperatura, humedad, vibración y ubicación es crítico. Validamos cuando $\geq 80\%$ de cuentas activas mantiene dispositivos online $> 95\%$ del tiempo y consulta la vista en tiempo real semanalmente.
2. Creemos que alertas automáticas para volcado, baja energía y ruptura de frío son fundamentales. Validamos cuando $\geq 70\%$ de eventos críticos genera apertura de alerta y $\geq 40\%$ desencadena acción registrada.
3. Creemos que la gestión multi-vehículo en una sola plataforma es necesaria. Validamos cuando usuarios gestionan ≥ 20 vehículos por cuenta sin caída de rendimiento percibido (tiempos < 2 s por acción).
4. Creemos que reportes históricos son necesarios para cumplimiento y trazabilidad. Validamos cuando $\geq 60\%$ programa reportes recurrentes y consulta históricos ≥ 1 vez/semana.

5. Creemos que la integración con IoT existente amplía mercado. Validamos cuando al menos 2 integraciones con hardware de terceros se usan en producción y representan $\geq 25\%$ de dispositivos activos.

1.2.2.3. Lean UX Hypothesis Statements

Hipótesis 1

Creemos que reducir en 30% las pérdidas de producto y aumentar en 25% la satisfacción del cliente se logrará si las empresas de transporte logran responder a tiempo a eventos críticos (ruptura de frío, humedad fuera de rango, volcado) con un sistema de alertas en tiempo real. Lo sabremos cuando los incidentes reportados bajaran $\geq 30\%$ y el CSAT/NPS suba ≥ 10 puntos en 8 semanas de piloto, con $\geq 80\%$ de feedback positivo.

Hipótesis 2

Creemos que mejorar en 40% la eficiencia operativa se logrará si los gerentes de operaciones logran detectar y priorizar incidentes en minutos con un dashboard intuitivo que muestra el estado completo de la flota (temperatura, humedad, vibración, ubicación). Lo sabremos cuando el tiempo medio de respuesta baje de > 4 h a < 30 min y el tiempo para identificar el vehículo en riesgo sea < 10 s para $\geq 80\%$ de usuarios en 4 semanas.

Hipótesis 3

Creemos que alcanzar 15% de adopción del mercado objetivo en 12 meses se logrará si los decisores de las empresas de transporte logran seleccionar el plan adecuado a su valor percibido con un modelo de suscripción flexible por niveles. Lo sabremos cuando tengamos ≥ 150 empresas activas, conversión piloto→pago $\geq 40\%$ y churn mensual $\leq 3\%$ dentro de 12 meses.

Hipótesis 4

Creemos que aumentar en 20% la retención de clientes de nuestros usuarios se logrará si los gerentes de operaciones logran demostrar trazabilidad completa a sus clientes con reportes automáticos e historial auditado. Lo sabremos cuando la tasa de renovación suba $\geq 20\%$, el tiempo de preparación de auditorías baje $\geq 50\%$, y se programen reportes recurrentes en $\geq 60\%$ de cuentas en 1 trimestre.

1.2.2.4. Lean UX Canvas

Lean UX Canvas

1. BUSINESS PROBLEM Las empresas de transporte de productos sensibles necesitan soluciones accesibles para monitorear múltiples parámetros de la carga (temperatura, humedad, vibración, etc.) y evitar pérdidas económicas por incumplimiento de parámetros.	5. SOLUTION IDEAS “ <ul style="list-style-type: none">• Plataforma IoT de monitoreo en tiempo real (múltiples sensores)• Sistema de alertas automáticas (vibración, volcado, energía, etc.)• Dashboard intuitivo de gestión de flota• Reportes automatizados de trazabilidad completa• Modelo de suscripción escalable ”	2. BUSINESS OUTCOMES <ul style="list-style-type: none">• 150 empresas suscritas en año 1• Reducción 30% pérdidas de clientes• 80% satisfacción de usuarios• Revenue recurrente de \$180K anuales• Expansión a 3 países LATAM
3. USERS Primarios: <ul style="list-style-type: none">• Gerentes de operaciones de transporte• Conductores de vehículos Secundarios: <ul style="list-style-type: none">• Clientes finales de servicios de transporte• Administradores de empresas de transporte		4. USER OUTCOMES & BENEFITS Empresas de Transporte: <ul style="list-style-type: none">• Reducción de pérdidas por múltiples factores (temperatura, vibración, etc.)• Cumplimiento regulatorio automatizado• Optimización de rutas y recursos Clientes Finales: <ul style="list-style-type: none">• Visibilidad completa del estado de sus productos• Reportes de trazabilidad en tiempo real• Reducción de riesgos operativos
6. Hypothesis Creemos que las PYMEs de transporte adoptarán una solución de monitoreo multiparamétrico si es accesible y fácil de usar. Para validar desarrollaremos un MVP con funcionalidades core (temp, humedad, ubicación) y mediremos adoption rate.	7- WHAT'S THE MOST IMPORTANT THING WE NEED TO LEARN FIRST? ¿Están las empresas de transporte dispuestas a pagar por una solución de monitoreo multiparamétrico IoT y cuál es el precio óptimo que maximiza adopción y rentabilidad?	8. WHAT'S THE LEAST AMOUNT OF WORK WE NEED TO DO TO LEARN THE NEXT MOST IMPORTANT THING? “ <ul style="list-style-type: none">• Desarrollar landing page con pricing y funcionalidades• Realizar 20 entrevistas con empresas target• Crear prototipo de dashboard con múltiples datos• Validar willingness to pay con pre-orders ? ”

UXPRESSIA

This persona was built in upressoia.com

1.3. Segmentos objetivo

1.3.1 Segmento 1: Empresas Clientes

Estas empresas, dedicadas a la logística, distribución o producción de bienes sensibles, necesitan un control exhaustivo sobre sus cargas para asegurar la calidad y evitar pérdidas económicas. Su objetivo es tener una visibilidad completa en tiempo real de sus productos, centralizando toda la información en una sola plataforma para optimizar sus operaciones, cumplir con las normativas y generar confianza en sus propios clientes.

Características:

- **Rol:** Gerentes o responsables de logística, calidad o distribución.
- **Ubicación:** Empresas ubicadas en zonas con alta actividad logística y acceso a tecnologías de digitalización.
- **Sector de la industria:** Alimentos perecederos, farmacéuticos, químicos, flores y otros productos que requieren condiciones especiales de conservación.

1.3.2 Segmento 2: Clientes Finales

Los clientes finales son los consumidores de los productos transportados por las empresas. Su necesidad principal es la transparencia y la seguridad, ya que buscan la tranquilidad de saber que el producto que adquieren ha sido manejado bajo los estándares de calidad correctos. Valoran la capacidad de verificar el estado de su pedido, desde el origen hasta la entrega, a través de una plataforma sencilla y confiable.

Características:

- **Edad:** Mayor a 18 años
- **Ubicación:** Lima, Perú
- **Nivel Socioeconómico:** Medio a alto

Capítulo II: Requirements Elicitation & Analysis

2.1. Competidores

2.1.1. Análisis competitivo

Análisis Competitivo

¿Por qué llevar a cabo este Análisis? Este análisis permite identificar fortalezas, debilidades y oportunidades en el mercado de soluciones IoT para el monitoreo de cadena de frío, de modo que Macetech pueda priorizar características, precios y estrategias de marketing que maximicen su adopción en el mercado peruano y latinoamericano.

Sensitech (Thermo King)



Frigga (China)



Emerson Cargo Solutions



CargaSafe



Perfil

Overview Multinacional estadounidense líder en monitoreo de la cadena de frío con décadas de experiencia. Fabricante global de dispositivos IoT para cadena de frío, con distribución en más de 60 países. División de Emerson Electric dedicada a soluciones de monitoreo de transporte refrigerado. Startup tecnológica latinoamericana que ofrece monitoreo en tiempo real enfocado en la temperatura del transporte de cargas. ¿Qué valor ofrece a los clientes? Ofrece confianza, cumplimiento de normativas globales (FDA, OMS), cobertura mundial y tecnología robusta. Ofrece sensores desechables/reutilizables de bajo costo, fáciles de implementar en transporte. Seguridad y precisión en tiempo real con analítica avanzada para grandes corporaciones. Propuesta accesible y flexible que asegura la conservación de productos críticos, con alertas inmediatas y dashboards intuitivos.

Perfil de Marketing

Mercado objetivo Multinacionales farmacéuticas, agroexportadoras y grandes retailers. Exportadores agrícolas y farmacéuticos medianos. Corporaciones de alimentos y farmacéuticas multinacionales. Empresas de transporte, agroexportadores medianos, distribuidores locales de alimentos y fármacos. Estrategias de marketing Presencia en ferias globales, contratos con distribuidores y certificaciones internacionales. Marketing digital, distribuidores locales, precios competitivos. Ventas consultivas, certificaciones globales, contratos a largo plazo. Marketing digital, alianzas con cámaras de comercio, programas de suscripción escalables.

Perfil de Producto

Productos & Servicios Data loggers, sensores IoT, software de análisis predictivo, soporte técnico 24/7. Data loggers, dispositivos de monitoreo en tiempo real, dashboards básicos. Monitoreo en tiempo real, analítica predictiva, dashboards avanzados. Sensores IoT propios o integrados, aplicación web y móvil, dashboards con métricas clave, alertas en tiempo real. Precios y costos Altos; modelo premium con costo por dispositivo y licencias anuales. Muy competitivos; pago por dispositivo + acceso a plataforma. Elevados; modelo enterprise con contratos anuales. Suscripciones flexibles + costo bajo por dispositivo. Canales de distribución Distribuidores autorizados globales, venta directa enterprise, canal online. Marketplace de e-commerce, distribuidores locales, venta directa. Venta directa corporativa, partners certificados, canal enterprise. Venta directa, partnerships con cámaras de comercio, distribuidores especializados en logística.

Análisis SWOT

- Fortalezas • Reputación global
• Cumplimiento normativo
• Soporte internacional • Precios accesibles
• Disponibilidad masiva • Marca reconocida
• Integración tecnológica avanzada • Accesibilidad y escalabilidad
• Enfoque en empresas de transporte
• Software amigable Debilidades • Alto costo
• Poca flexibilidad para PYMEs • Limitada personalización de software
• Menor soporte local en LATAM • Precio inaccesible para PYMEs
• Implementación compleja • Respaldo de marca frente a multinacionales
• Mercado nicho especializado Oportunidades • Creciente regulación en transporte farmacéutico y alimentario • Crecimiento del e-commerce y transporte de alimentos
• Demanda en mercados regulados (fármacos, vacunas) • Expansión en LATAM donde grandes competidores no tienen presencia fuerte
• Crecimiento del e-commerce y transporte Amenazas • Startups ágiles con precios más bajos en LATAM • Competidores regionales con soluciones más adaptadas • Startups regionales con mejor relación costo-beneficio • Copia rápida de modelo por competidores grandes o locales
• Regulaciones de transporte cambiantes

2.1.2. Estrategias y tácticas frente a competidores

• Precios accesibles y modelo de suscripción flexible

Plan básico desde \$29/mes por dispositivo con suscripción mensual sin compromisos a largo plazo, contrastando con licencias anuales costosas de Sensitech y Emerson.

• Soporte local y personalización regional

Equipo técnico en español con horarios LATAM, dashboards personalizables con métricas locales y cumplimiento de normativas regionales (SENASA, DIGESA).

• Implementación rápida y sin complejidad técnica

Configuración plug-and-play en menos de 24 horas versus semanas de implementación de competidores enterprise, con capacitación incluida.

• Alianzas estratégicas con el ecosistema local

Partnerships con cámaras de comercio agrícola, asociaciones de transportistas y distribuidores de dispositivos IoT en mercados emergentes.

• Transparencia de datos y alertas proactivas

API abierta para integración con sistemas ERP locales, reportes en tiempo real y alertas vía WhatsApp/SMS, ventajas sobre dashboards cerrados de competidores.

2.2. Entrevistas

2.2.1. Diseño de entrevistas

1. Preguntas generales

- ¿Cuál es tu nombre y cargo?
- ¿Cuántos años tienes?
- ¿En qué sector o industria trabajas? (alimentos, farmacéutica, logística, etc.)

2. Preguntas — Segmento: Empresa (Gestores de transporte)

1. Proceso actual de monitoreo

- ¿Cómo monitorean actualmente la temperatura durante el transporte de tus productos?

2. Herramientas y tecnología

- ¿Qué dispositivos o sistemas utilizas para el control de cadena de frío y por qué los elegiste?

3. Gestión de viajes y rutas

- ¿Cómo planificas y registras los viajes de transporte? ¿Qué información consideras esencial?

4. Desafíos principales

- ¿Qué problemas enfrentas cuando se rompe la cadena de frío? ¿Cómo impacta en costos y tiempo?

5. Alertas y respuesta a incidentes

- ¿Cómo te enteras cuando hay un problema de temperatura? ¿Qué tan rápido puedes responder?

6. Reportes y documentación

- ¿Qué tipo de reportes necesitas generar para clientes o autoridades regulatorias?

7. Gestión de dispositivos IoT

- Cuéntame sobre tu experiencia gestionando el mantenimiento y configuración de sensores o dispositivos de monitoreo. ¿Qué desafíos has encontrado?

8. Características ideales

- Si pudieras diseñar la plataforma perfecta, ¿qué funciones serían imprescindibles para ti?

9. Presupuesto y modelo de pago

- ¿Cuál sería tu modelo de pago preferido para este tipo de servicios y qué factores influyen en esa decisión?

3. Preguntas — Segmento: Clientes Finales (Consumidores finales)**1. Experiencia actual de recepción de productos**

- Cuéntame cómo verificas actualmente que los productos que compras llegaron en condiciones óptimas de temperatura.

2. Confianza y transparencia en proveedores

- Describe tu nivel de confianza en los reportes de temperatura que te proporcionan tus proveedores. ¿Qué factores aumentarían o disminuirían esa confianza?

3. Información requerida sobre el transporte

- ¿Qué información consideras más valiosa tener sobre el transporte de tus productos y cómo te ayudaría en tus operaciones?

4. Experiencias con productos dañados

- Comparte alguna experiencia que hayas tenido rechazando productos por problemas de cadena de frío. ¿Cómo identificaste el problema y qué impacto tuvo?

5. Preferencias de acceso a información

- Describe cómo prefieres recibir y acceder a información sobre tus pedidos. ¿Qué métodos de comunicación funcionan mejor para tu flujo de trabajo?

6. Alertas y notificaciones proactivas

- Cuéntame qué tipo de notificaciones durante el transporte de tus productos serían más útiles para ti y en qué momentos las necesitarías.

7. Facilidad de uso y comprensión

- Describe la importancia que tiene para ti que la información técnica sea presentada de manera comprensible. ¿Qué características valoras en las interfaces que usas?

8. Características más valoradas

- ¿Qué funcionalidades consideras que agregarían más valor a tu proceso de recepción y validación de productos?

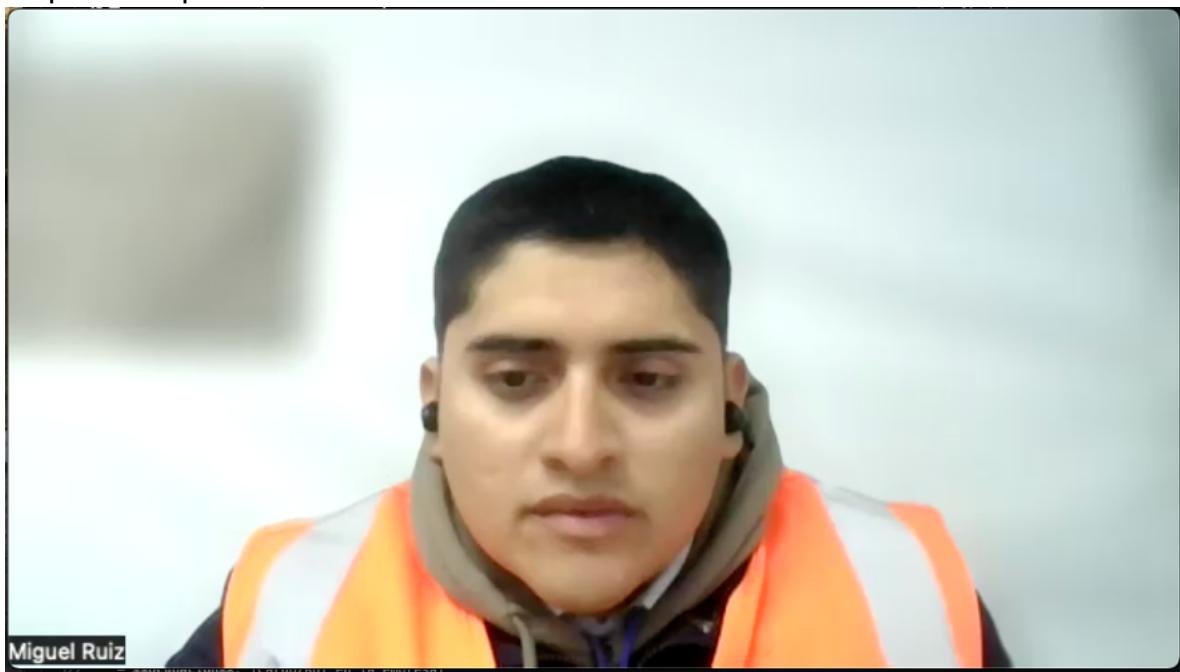
9. Expectativas sobre tecnología IoT

- ¿Qué beneficios esperas de un sistema de monitoreo IoT para tus compras de productos sensibles a temperatura y qué preocupaciones tienes al respecto?

2.2.2. Registro de entrevistas**Segmento 1: Empresa**

- **Nombre:** Miguel Ruiz
- **Edad:** 28 años
- **Ocupación:** Gestor de transportes - línea de frío
- **Empresa:** Ofertimaq - Distribuidora
- **Enlace:** [Click aquí para ver la entrevista](#)
- **Fecha de entrevista:** 10 de Setiembre del 2025

- **Tiempo inicio - tiempo fin:** 00:00:00 - 00:07:28



Resumen

Monitorea la temperatura de forma manual a través de choferes que revisan el cooler en paradas y con termómetro digital. Usa GPS para validar paradas y kilometraje, no para temperatura. La planificación la hace un asistente considerando tráfico de Lima (Waze) con holguras, y exige verificación de cooler en puntos de parada. La detección de problemas depende de llamadas de choferes; cuando hay incidente, redirigen un vehículo cercano para salvar producto. Gestiona reportes de ruta como evidencia frente a reclamos. No han implementado sensores; demanda una plataforma con monitoreo en tiempo real, alertas automáticas, y reportes simples/dashboards. Prefiere suscripción mensual (mejor aún prepago/largo plazo) y acceso del cliente a un link para seguimiento.

Rasgos objetivos

- *Herramientas:* GPS, termómetro digital, Waze/Mapas.
- *Canales:* Llamadas, mensajería interna.
- *Dispositivos:* Smartphone de choferes + PC oficina.
- *Reportes:* Hoja de ruta; auditorías sanitarias.

Rasgos subjetivos

- Perfil operativo, orientado a continuidad y respuesta rápida.
- Valora trazabilidad visible para clientes.

Dolores y oportunidades

Pain: Dependencia de manualidad y llamadas; falta de visibilidad en ruta. *Need:* Telemetría + alertas; portal cliente; evidencia automática.

Implicancias para CargaSafe

Requisitos: Sensor temperatura; alertas; linea de tiempo por viaje; Link para compartir seguimiento.

Resumen

Monitorea la temperatura de forma manual a través de choferes que revisan el cooler en paradas y con termómetro digital. Usa GPS para validar paradas y kilometraje, no para temperatura. La planificación la hace un asistente considerando tráfico de Lima (Waze) con holguras, y exige verificación de cooler en puntos de parada. La detección de problemas depende de llamadas de choferes; cuando hay incidente, redirigen un vehículo cercano para salvar producto. Gestiona reportes de ruta como evidencia frente a reclamos. No han implementado sensores; demanda una plataforma con monitoreo en tiempo real, alertas automáticas, y reportes simples/dashboards. Prefiere suscripción mensual (mejor aún prepago/largo plazo) y acceso del cliente a un link para seguimiento.

Rasgos objetivos

- *Herramientas:* GPS, termómetro digital, Waze/Mapas.
- *Canales:* Llamadas, mensajería interna.
- *Dispositivos:* Smartphone de choferes + PC oficina.
- *Reportes:* Hoja de ruta; auditorías sanitarias.

Rasgos subjetivos

- Perfil operativo, orientado a continuidad y respuesta rápida.
- Valora trazabilidad visible para clientes.

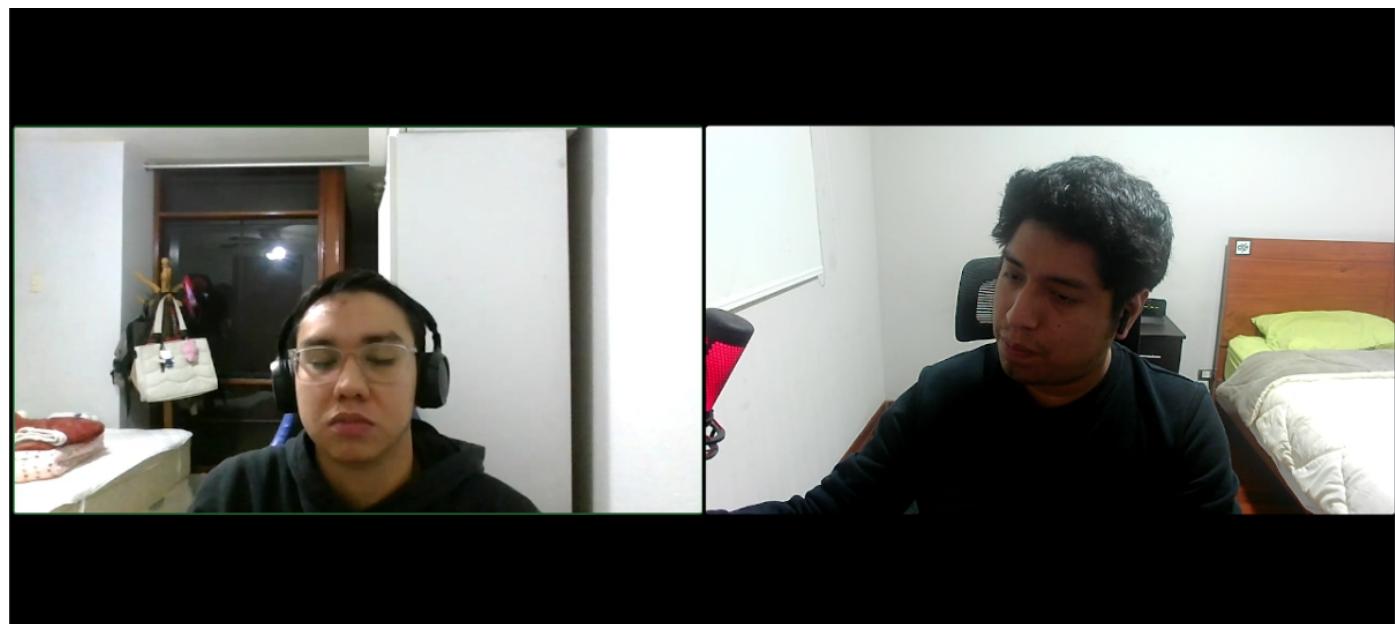
Dolores y oportunidades

Pain: Dependencia de manualidad y llamadas; falta de visibilidad en ruta. *Need:* Telemetría + alertas; portal cliente; evidencia automática.

Implicancias para CargaSafe

Requisitos: Sensor temperatura; alertas; linea de tiempo por viaje; Link para compartir seguimiento.

- **Nombre:** Mario Hinojosa
- **Edad:** 22 años
- **Ocupación:** Operador Logístico
- **Empresa:** Urbano - Distribuidora Ecommerce
- **Enlace:** [Click aquí para ver la entrevista](#)
- **Fecha de entrevista:** 11 de Setiembre del 2025
- **Tiempo inicio - tiempo fin:** 00:07:28 - 00:13:36

**Resumen**

Controla temperatura con termómetro portátil en paradas y el indicador del camión. El sistema de cadena de frío es nativo del vehículo, elegido por sencillez. Planifica con hojas de ruta y bitácora manual. Cuando falla el frío, se avisa a central; los daños impactan tiempos y reclamos. Emite hoja de viaje firmada con datos de temperatura. Percibe el monitoreo como pesado y manual. Pide una plataforma que integre panel del camión + app móvil con notificaciones cuando se sale del estándar.

Rasgos objetivos

Herramientas: Indicador del camión, termómetro, bitácora. *Canales:* Llamada a central. *Dispositivo:* Celular personal; cabina camión.

Rasgos subjetivos

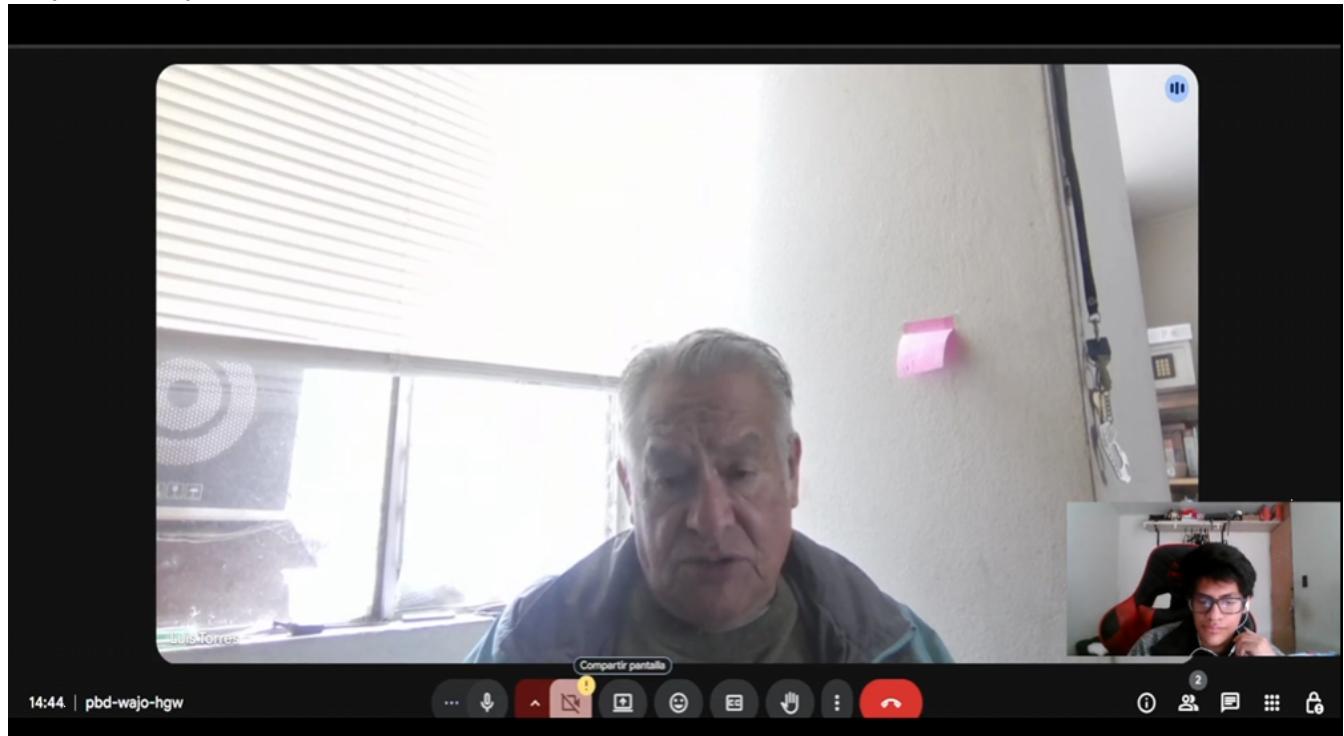
- Busca simplicidad y automatización de tareas repetitivas.

Dolores / Oportunidades *Pain:* Monitoreo manual constante; reacción tardía. *Need:* App móvil con push alerts, captura automática de lecturas.

Implicancias para CargaSafe Requisitos: App conductor (checklist, lecturas guiadas, foto/nota), notificaciones.

-
- **Nombre:** Luis Torres Arevalo
 - **Edad:** 58 años
 - **Ocupación:** Encargado de gestión logística
 - **Empresa:** Urbano . Distribuidor de Alimentos
 - **Enlace:** [Click aquí para ver la entrevista](#)
 - **Fecha de entrevista:** 10 de Setiembre del 2025

- **Tiempo inicio - tiempo fin:** 00:13:36 - 00:20:51



Resumen

Operaba con dos mediciones (salida/llegada); en ruta no había monitoreo. Control con termómetros en almacén y delivery; registros en papel/Excel. La ruptura de cadena implicaba devoluciones y pérdida de confianza. Respuesta en ruta nula por falta de visibilidad. Autoridades exigían tablas de control de temperatura continua; no podían cumplir plenamente. Considera imprescindibles temperatura en tiempo real, alertas y almacenamiento automático para auditoría. Prefiere suscripción mensual por unidad; clave: que funcione y no genere costos ocultos.

Rasgos objetivos

Herramientas: Papel/Excel, termómetros. *Canales:* Presencial/correo. *Dispositivos:* PC oficina.

Rasgos subjetivos

- Valora cumplimiento regulatorio y evidencias.

Dolores / Oportunidades

Pain: Sin datos en ruta; auditorías exigentes. *Need:* Historian de temperatura, descargables (PDF/CSV), firmas digitales.

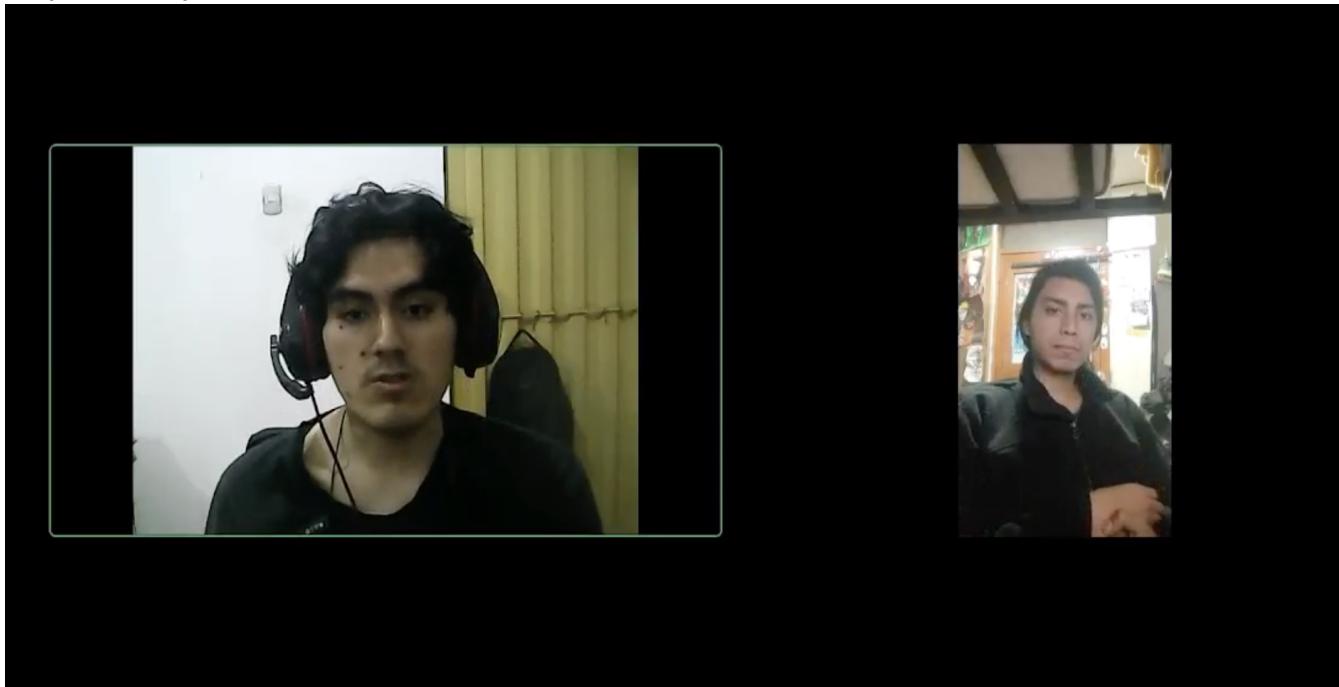
Implicancias para CargaSafe

Requisitos: dispositivo electrónico portátil que mide y registra automáticamente datos de diversas variables como temperatura, retención de datos, rango, desviaciones y firmas.

Segmento 2: Clientes Finales (Consumidores finales)

- **Nombre:** Adrián Zapata
- **Edad:** 23 años
- **Ocupación:** Responsable de parrilla en un negocio de comida rápida
- **Empresa/Sector:** Negocio local de comida rápida / Sector alimentario
- **Enlace:** [URL del video de la entrevista](#)
- **Fecha de entrevista:** 10 de Setiembre del 2025

- **Tiempo inicio - tiempo fin:** 00:20:51 - 00:30:50



Resumen

Verifica al recibir: estado físico, frío al tacto, indicadores simples; desconfía de la cadena previa. Aumenta confianza con datos en tiempo real y trazabilidad. *Información valiosa:* tiempo y temperatura en trayecto; notificaciones ante retrasos (tráfico) o temperaturas fuera de rango. Prefiere WhatsApp para avisos y un portal/app para consultar detalles on-demand. Quiere interfaces claras, en °C y tipografía grande. Pide notificaciones al proveedor además del cliente.

Tecnología & canales

- WhatsApp como principal; App/portal web como consulta.
- Smartphone predominante.

Dolores / Oportunidades

Pain: Incertidumbre en ruta; impacto de retrasos. *Need:* ETA (calculo de duracion de ruta) + temperatura en vivo; doble notificación (cliente/proveedor).

Implicancias para CargaSafe

Requisitos: Link para seguimiento, push WhatsApp/SMS configurable, UI accesible (alto contraste, números grandes).

- **Nombre:** Brenda Calderon
- **Edad:** 20 años
- **Ocupación:** Trabajadora de medio tiempo en un minimarket, responsable de compras de insumos para refrigeración
- **Empresa/Sector:** Retail alimentario local – Consumo final
- **Enlace:** [URL del video de la entrevista](#)
- **Fecha de entrevista:** 13 de Setiembre del 2025

- **Tiempo inicio - tiempo fin:** 00:31:00 - 00:37:20



Resumen

Audita recepción: integridad del empaque, condensación, frío al tacto, fecha de vencimiento. Confía parcialmente en reportes; requiere datos trazables y consistentes con lo recibido. *Información clave:* temperatura a lo largo del trayecto, eventos fuera de rango, tiempos. Rechazó lote de yogures por tibiaza/inflado; proveedor sin justificación. Prefiere portal o app para consultar sin llamadas; notificaciones breves ante retrasos/problemas. Valora interfaces simples con gráficos y reportes descargables para evidencias.

Tecnología & canales

App/portal web, notificaciones breves; móvil como dispositivo principal.

Dolores / Oportunidades

Pain: Reportes genéricos; inconsistencias. *Need:* Logs detallados, exportables y comparables por pedido.

Implicancias para CargaSafe

Requisitos: Panel cliente con histórico por pedido, botón Descargar PDF/CSV, alertas de retraso y anomalía térmica.

-
- **Nombre:** Gabriel Díaz
 - **Edad:** 26 años
 - **Ocupación:** Responsable de recepción y control de productos sensibles a temperatura.
 - **Empresa/Sector:** Sector de salud
 - **Enlace:** [URL del video de la entrevista](#)
 - **Fecha de entrevista:** 16 de Setiembre del 2025

- **Tiempo inicio - tiempo fin:** 00:37:20 - 00:43:15



Resumen

Revisa empaque e indicadores de temperatura; busca signos de condensación. Confianza media si reporte es básico; aumenta con registro continuo y claro. Necesita alertas en trayecto y pre-aviso de llegada. Ha rechazado lotes por salirse del rango; impacto: retrasos y reprocesos. Prefiere canales digitales (correo/app). Requiere interfaces claras, colores de alerta, validación automática de cumplimiento y reportes de auditoría.

Tecnología & canales

- Email/app; PC y smartphone; sensibilidad a conectividad (preocupación).

Dolores / Oportunidades

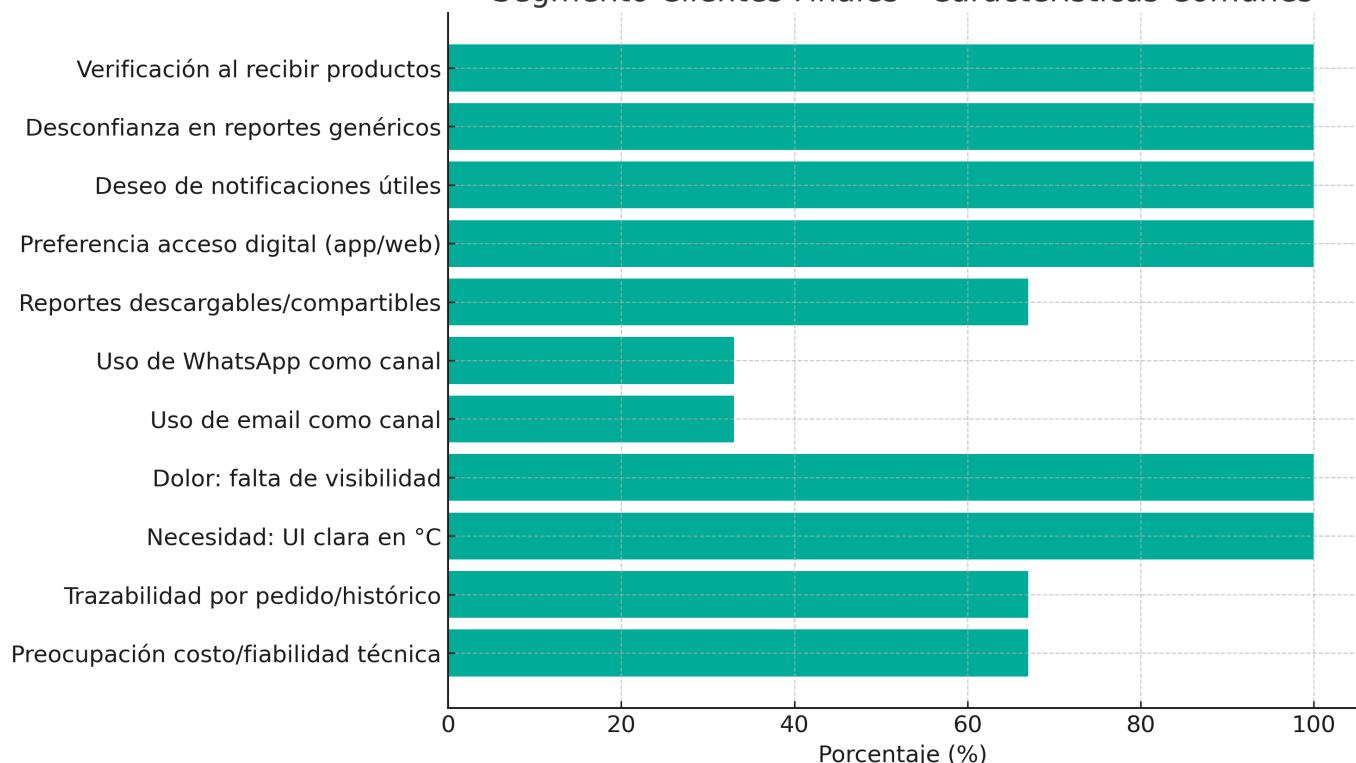
Pain: Falta de seguimiento y validación automática. *Need:* Registro de procesos, como una auditoria para prevenir fallos

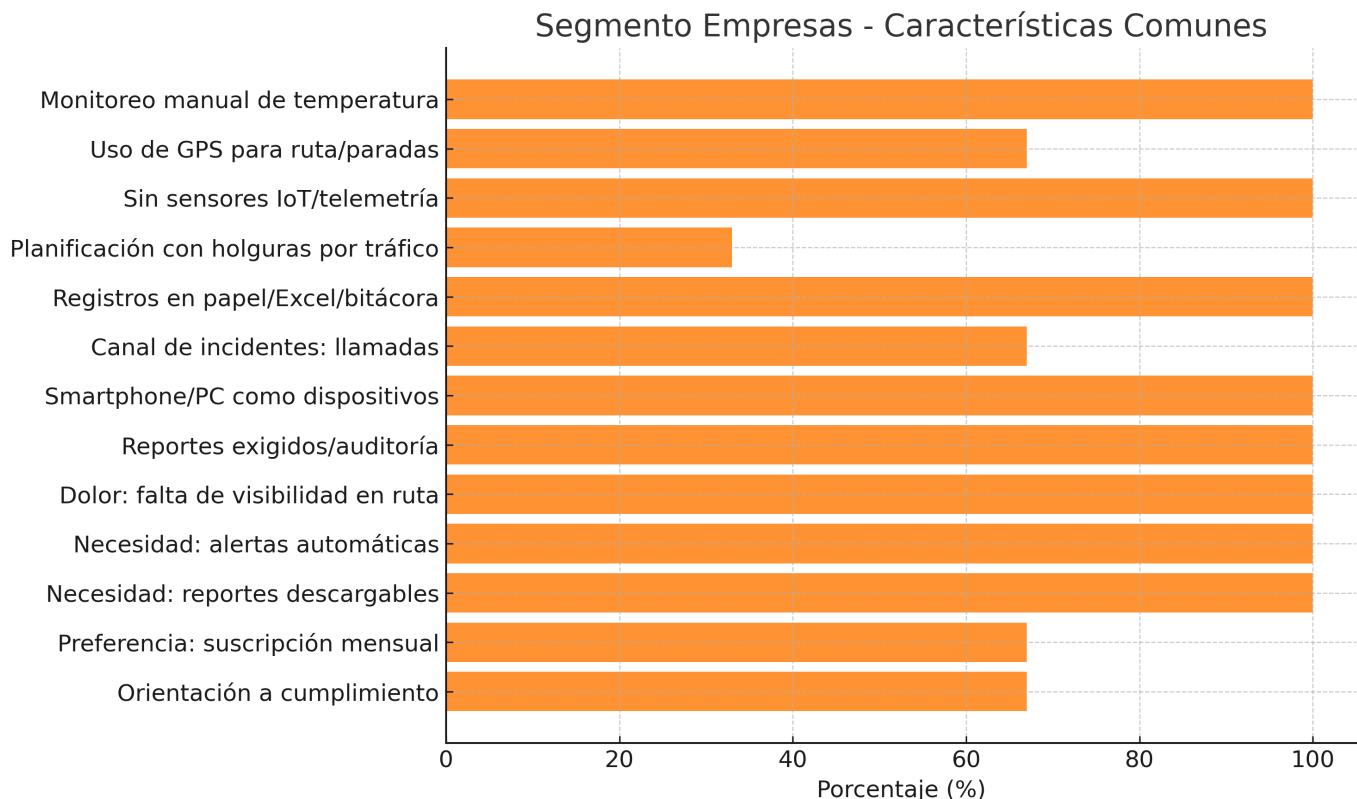
Implicancias para CargaSafe

Requisitos: Motor de validación (rango por perfil de producto), alertas multi-canal.

2.2.3. Análisis de entrevistas

Segmento Clientes Finales - Características Comunes





2.3. Needfinding

2.3.1. User Personas

- Segmento: Empresa (Gestores de transporte)

NAME

Carlos Méndez

TYPE

Judging



Goals

- Asegurar que los viajes de transporte cumplan tiempos y parámetros de calidad.
- Reducir incidencias y pérdidas en la cadena de suministro.
- Monitorear la temperatura en tiempo real con reportes confiables.
- Optimizar la toma de decisiones mediante dashboards y métricas.

Quote

“Una operación fluida depende de que cada detalle esté bajo control.”

Background

Carlos ha trabajado más de 15 años en logística y transporte. Maneja una flota que mueve productos sensibles (como alimentos y medicamentos), donde mantener la temperatura y tiempos de entrega es crítico. Es exigente con la tecnología: espera precisión y confiabilidad, ya que cualquier falla impacta en su operación.

Demographic

Motivations

Frustrations

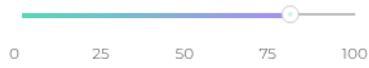
<input checked="" type="radio"/> Male	42	years
<input type="checkbox"/> Lima		
Married		
Gerente de Operaciones Logísticas		

Skills

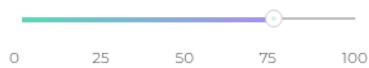
Gestión Logística



Liderazgo de equipos



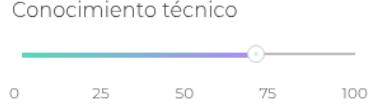
Análisis de datos



Toma de decisiones bajo



presión

**Technology****Browsers****Channels**

Online chat

Email

WhatsApp

- Reducir pérdidas económicas y proteger la reputación de la empresa
- Mantener el control total sobre las operaciones logísticas
- Ofrecer un servicio confiable que genere confianza en los clientes
- Implementar tecnología que facilite la toma de decisiones

- Procesos manuales que generan errores o retrasos.
- Dificultad para monitorear todo de forma simultánea.
- Falta de visibilidad de los estados en tiempo real.
- Que el sistema de alertas no sea ágil o efectivo.

Brands and influencers

This persona was built in uxpressia.com

Carlos Mendoza - Jefe de Logística

El user persona de Carlos representa al gestor experimentado que prioriza la eficiencia operativa y la minimización de riesgos. Muestra la necesidad de herramientas tecnológicas robustas y precisas que le permitan mantener control total sobre la cadena de frío. Su perfil refleja la importancia de la confiabilidad del sistema, ya que cualquier falla puede resultar en pérdidas económicas significativas y problemas regulatorios. Carlos ejemplifica al usuario que valora los datos en tiempo real, reportes detallados y funcionalidades que le permitan tomar decisiones informadas para proteger productos de alto valor.

- **Segmento: Clientes Finales (Consumidores finales)**

NAME		TYPE
María González		Thinking
	Goals <ul style="list-style-type: none"> Asegurar que los insumos lleguen siempre frescos y en condiciones óptimas. Evitar pérdidas económicas por productos dañados o fuera de los estándares de calidad. Monitorear pedidos y entregas de manera sencilla y en tiempo real. Ofrecer a sus clientes finales platos de confianza, respaldados por una cadena de suministro transparente. 	
Demographic <p>Female 37 years</p> <p>Lima</p> <p>Married</p> <p>Gerente de Operaciones Logísticas</p>	Quote <p><i>Mi negocio depende de que los insumos lleguen frescos y a tiempo.</i></p>	Background <p>María administra un restaurante reconocido en su ciudad y depende de insumos frescos para mantener la calidad de sus platos. Está acostumbrada a lidiar con proveedores, pero exige que la información sea transparente y verificable. Valora la transparencia: quiere acceso claro a reportes de temperatura de los insumos que ella solicita, porque cada fallo puede traducirse en clientes insatisfechos y pérdidas.</p>
Skills <p>Gestión de negocio gastronómico: 85</p> <p>Evaluación de calidad de insumos: 88</p> <p>Negociación con proveedores: 82</p> <p>Gestión de costos y presupuesto: 80</p> <p>Atención al cliente: 84</p>	Motivations <ul style="list-style-type: none"> Mantener la calidad excepcional de su restaurante con insumos frescos. Proteger la reputación de su negocio ante sus clientes. Reducir pérdidas económicas por productos dañados o en mal estado. Tener control y transparencia total sobre sus proveedores. Hacer crecer su negocio con procesos confiables y verificables. Ofrecer confianza a sus clientes mediante trazabilidad de ingredientes. 	Frustrations <ul style="list-style-type: none"> Productos en mal estado que generan pérdida y afectan la reputación del negocio. Falta de transparencia en el transporte de alimentos perecederos. Dificultad para reclamar incidencias o validar cumplimiento de estándares.
Brands and influencers   		

Technology

Browsers

Channels

UXPRESSIA

This persona was built in uexpressia.com

María González - Gerente de Compras de Restaurante

El user persona de María representa al consumidor final que valora la transparencia y la calidad en los productos que adquiere para su negocio. Como responsable de compras de un restaurante, necesita la seguridad de que los alimentos que recibe han mantenido la cadena de frío adecuada durante el transporte. Su perfil ilustra la importancia de contar con información clara y accesible sobre el estado de los productos, reportes de cumplimiento fáciles de entender, y la capacidad de verificar la integridad de los alimentos antes de aceptar las entregas. María ejemplifica al usuario que busca confianza y transparencia en el proceso logístico para proteger la reputación de su negocio.

2.3.2. User Task Matrix

Segmento: Empresa (Gestores de transporte)

Tarea	Frecuencia	Importancia
Llamar a conductores para verificar condiciones de carga manualmente	Alta	Alta
Revisar múltiples parámetros (temperatura, humedad, vibración) al final del viaje	Alta	Alta
Completar bitácoras en papel con datos de condiciones del cargamento	Alta	Media
Buscar información de viajes en múltiples sistemas desintegrados	Alta	Media
Coordinar por teléfono cuando hay incidencias en las condiciones de transporte	Media	Alta
Recopilar firmas y documentos físicos de entregas	Alta	Media
Armar reportes manuales combinando datos de diferentes fuentes	Media	Alta
Enviar unidades de emergencia cuando se detecta falla en el transporte	Baja	Alta
Atender consultas de clientes por falta de visibilidad en tiempo real	Media	Alta
Revisar rutas en GPS básico sin integración con sensores de carga	Alta	Media
Capacitar conductores en procedimientos de verificación de carga	Baja	Media
Verificar manualmente el funcionamiento de sistemas de conservación	Alta	Alta

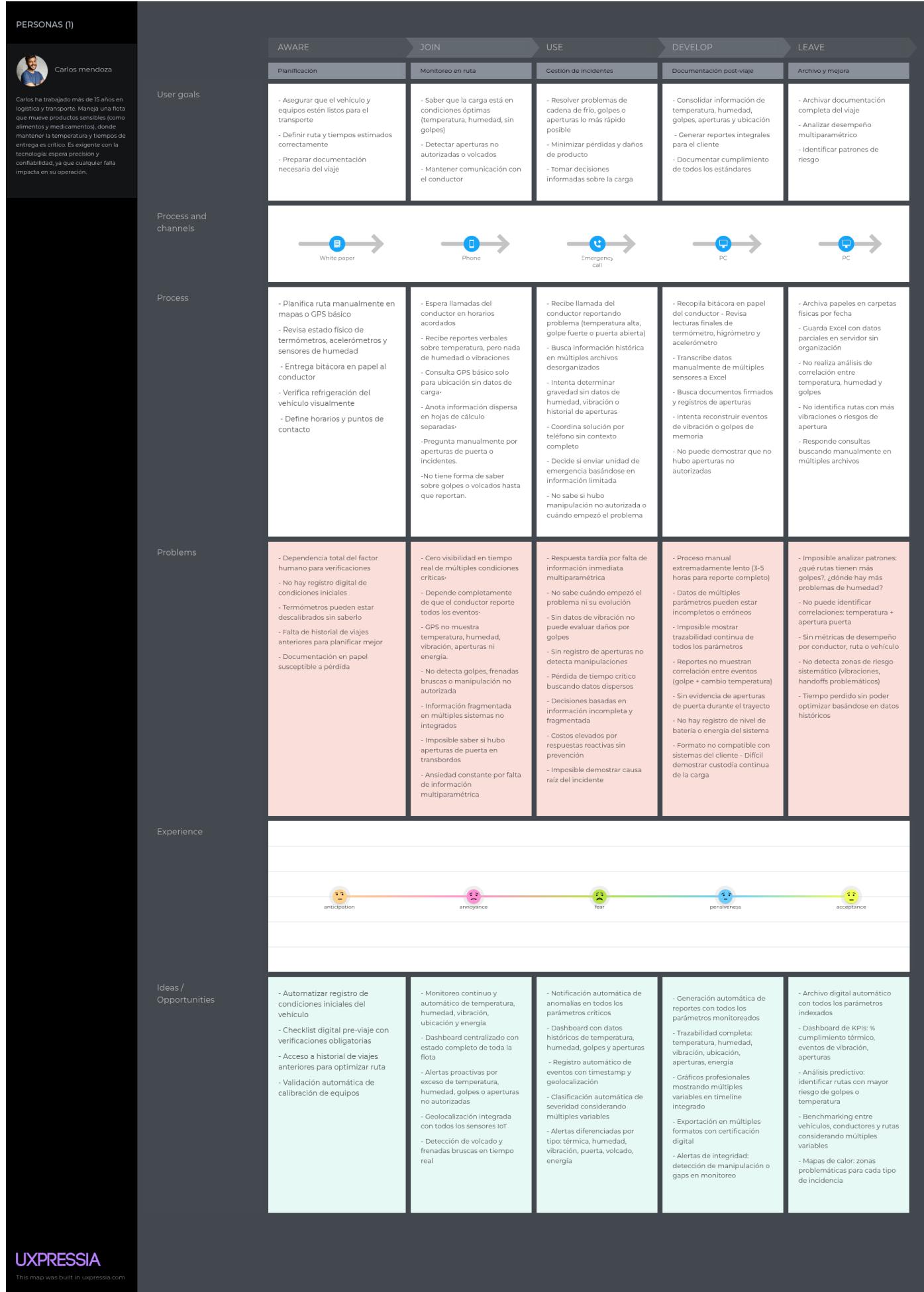
Tarea	Frecuencia	Importancia
Consolidar información de múltiples dispositivos y plataformas	Alta	Alta

Segmento: Clientes Finales (Consumidores finales)

Tarea	Frecuencia	Importancia
Verificar productos visualmente al recibirlos	Alta	Alta
Inspeccionar condiciones físicas de productos sensibles	Alta	Alta
Llamar al proveedor para preguntar estado del envío	Media	Media
Examinar empaques buscando señales de deterioro o daños	Alta	Alta
Rechazar productos que muestran signos de mal manejo	Media	Alta
Solicitar reportes de trazabilidad que suelen ser genéricos o incompletos	Media	Alta
Esperar sin información sobre el estado real de sus pedidos	Media	Alta
Revisar fechas de vencimiento y condiciones de almacenamiento	Alta	Alta
Registrar incidencias de productos que llegan en mal estado	Baja	Alta
Aceptar productos sin evidencia objetiva de las condiciones de transporte	Alta	Media
Realizar reclamos por productos deteriorados o fuera de especificación	Baja	Alta
Archivar documentación física de entregas	Media	Baja
Validar cumplimiento de condiciones especiales sin datos verificables	Alta	Alta

2.3.3. User Journey Mapping

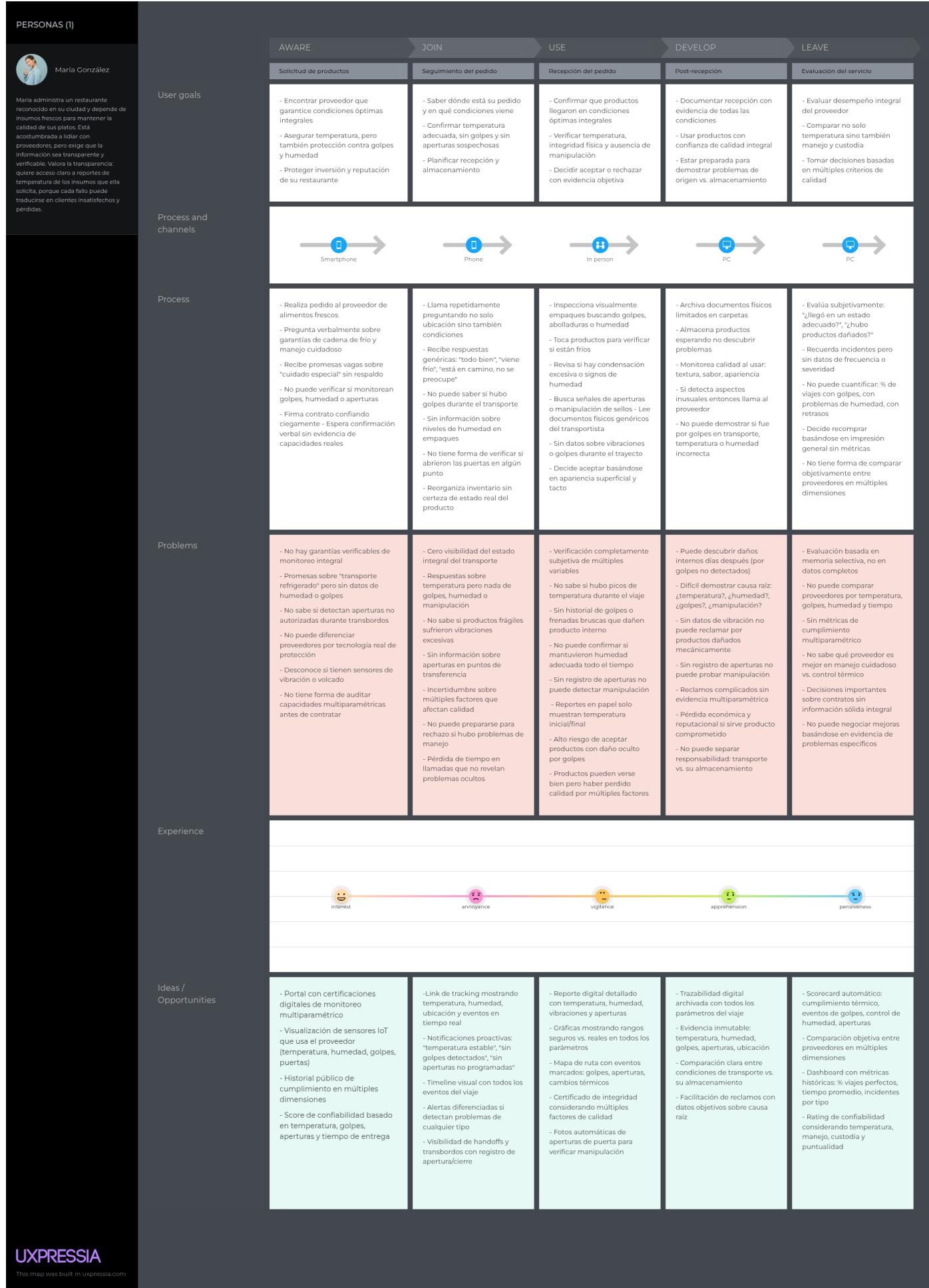
Journey Map: Carlos Mendoza (Gestor de transporte)



El Journey Map de Carlos muestra un proceso de 5 etapas desde la planificación hasta la entrega final. Sus momentos críticos se concentran en la configuración de parámetros correctos y la gestión eficiente de alertas durante el viaje. Las oportunidades principales incluyen simplificar la configuración inicial con plantillas

predefinidas, proporcionar dashboards unificados durante el monitoreo, y automatizar la generación de reportes post-viaje. Sus mayores pain points están en la complejidad de configuración y la falta de contexto en las alertas críticas.

Journey Map: María González (Gerente de Compras de Restaurante)

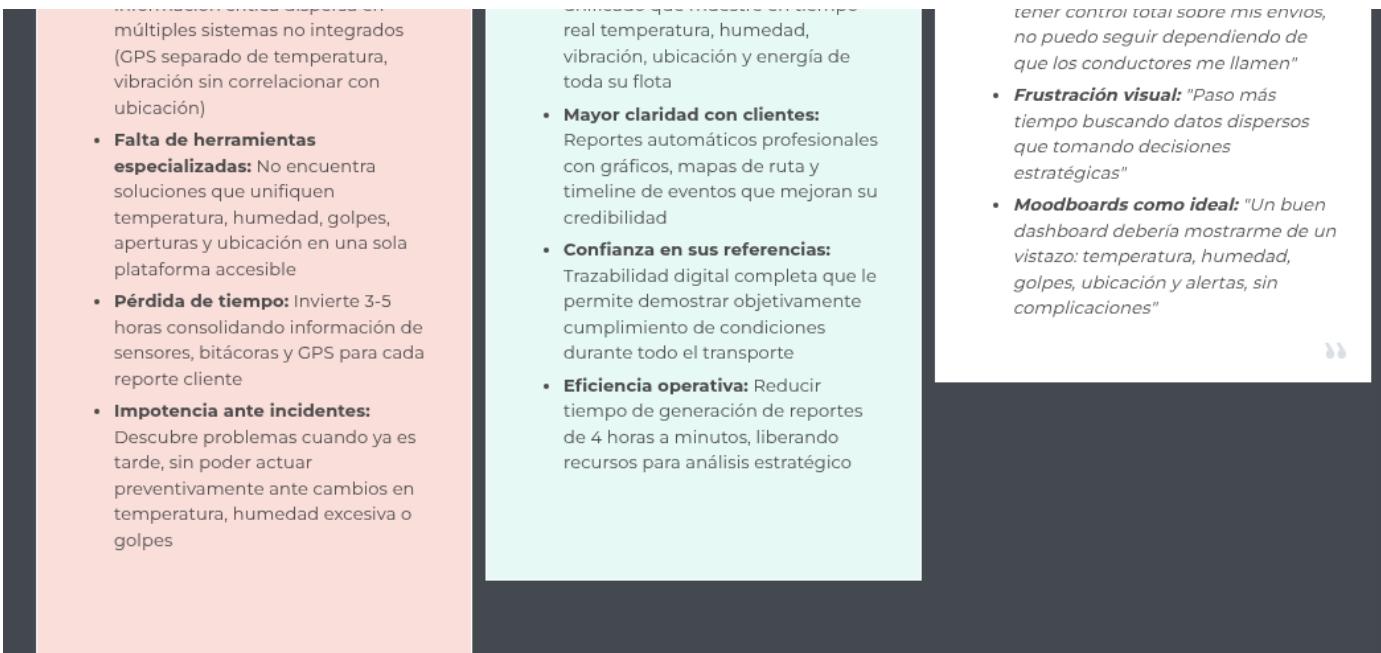


acceso fácil a reportes de cumplimiento, notificaciones proactivas sobre el estado del transporte, y documentación clara que facilite la toma de decisiones de aceptación. Sus mayores pain points están en la falta de transparencia durante el transporte y la dificultad para verificar la integridad de los productos al momento de la entrega.

2.3.4. Empathy Mapping

Segmento: Empresa (Gestores de transporte) - Carlos Mendoza

<p>1. ¿Con quién estamos empatizando?</p> <p>Estamos empatizando con Carlos Mendoza, quien tiene 45 años y tiene el cargo de jefe de logística. Gestiona una flota de 15 vehículos refrigerados que transportan productos perecederos, farmacéuticos y flores de exportación. Responsable de garantizar entregas en condiciones óptimas mientras controla costos operativos. Usa métodos tradicionales (bitácoras en papel, llamadas telefónicas, GPS básico) para monitorear sus envíos. Constantemente presionado por clientes que exigen trazabilidad y por la necesidad de reducir pérdidas por productos dañados.</p>	<p>7. ¿Qué es lo que dice y hace?</p> <p><i>"Mi reputación depende de entregar en perfecto estado, pero no tengo visibilidad de lo que pasa en ruta"</i></p> <p><i>"La tecnología debe ser clara y medible, no quiero complicarme con sistemas que mis conductores no puedan usar"</i></p> <p><i>"Un error en la cadena de frío me cuesta miles de dólares y la confianza del cliente"</i></p> <p><i>"Invierto demasiado tiempo buscando información dispersa cuando debería estar optimizando rutas"</i></p>	<p>2. ¿Qué es lo que necesita hacer?</p> <ul style="list-style-type: none"> Centralizar supervisión visual: Monitorear el estado completo de múltiples vehículos simultáneamente desde una plataforma única Organizar de forma clara: Clasificar y acceder rápidamente a información histórica de viajes anteriores Presentar moodboards: Mostrar reportes profesionales con gráficos y datos de cumplimiento a clientes corporativos Abarrar tiempo: Reducir horas dedicadas a consolidar información dispersa de múltiples fuentes y sensores
<p>6. ¿Qué es lo que escucha?</p> <ul style="list-style-type: none"> Sugerencias de colegas: Recomendaciones sobre nuevas tecnologías de monitoreo multiparamétrico y plataformas cloud Tendencias en diseño: Conversaciones en seminarios sobre importancia de dashboards intuitivos con visualizaciones en tiempo real Feedback de clientes: Quejas sobre reportes poco claros y solicitudes de mayor transparencia en el proceso de transporte 		<p>3. ¿Qué es lo que ve?</p> <ul style="list-style-type: none"> Referencias dispersas: Hojas de Excel desorganizadas, carpetas físicas con bitácoras, lecturas de sensores en diferentes dispositivos Herramientas limitadas: GPS que solo muestra ubicación, termómetros sin conectividad, acelerómetros que requieren descarga manual Iniciativas de otros diseñadores: Colegas en otras empresas adoptando plataformas IoT integradas que consolidan temperatura, humedad, vibración y ubicación en dashboards unificados
<p>5. ¿Qué es lo que hace?</p> <ul style="list-style-type: none"> Crea tableros analógicos: Mantiene pizarras físicas con status de vehículos actualizadas manualmente Uso de herramientas variadas: Alterna entre GPS básico, hojas de cálculo, llamadas telefónicas y revisión de sensores físicos Presentación a clientes: Arma reportes en PowerPoint combinando capturas de pantalla, tablas de Excel y fotos de documentos firmados 	<p>PAINS</p> <ul style="list-style-type: none"> Desorganización visual: Información crítica dispersa en <p>GAINS</p> <ul style="list-style-type: none"> Flujo creativo continuo: Dashboard unificado que muestre en tiempo 	<p>4. ¿Qué es lo que dice?</p> <p><i>"Necesito"</i></p> <ul style="list-style-type: none"> Necesidad de orden: "Necesito"



UXPRESSIA

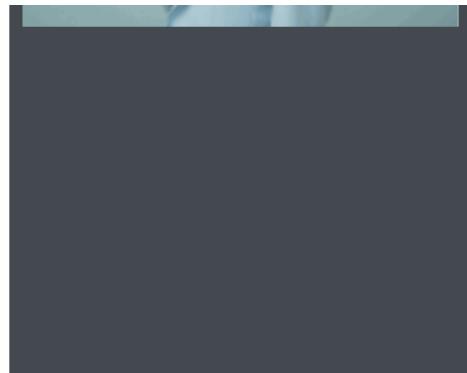
This persona was built in upressoia.com

El empathy map de Carlos revela a un profesional experimentado que busca control total y confiabilidad en los sistemas de monitoreo. Sus principales preocupaciones giran en torno a las pérdidas económicas por fallas en la cadena de frío y la necesidad de mantener la reputación empresarial. Valora la tecnología que le proporcione visibilidad en tiempo real, reportes automáticos y alertas accionables que le permitan responder rápidamente ante incidentes. Su enfoque está en el ROI medible y sistemas que cumplan con regulaciones estrictas.

Segmento: Clientes Finales (Consumidores finales) - María González



- Sugerencias de colegas:** Recomendaciones de otros gerentes sobre proveedores que ofrecen mejor trazabilidad y transparencia
- Tendencias en diseño:** Conversaciones en asociaciones de restauranteros sobre importancia de verificar cadena de frío completa, no solo punto final
- Feedback de clientes:** Preocupaciones crecientes sobre seguridad alimentaria y expectativas de que restaurantes garanticen calidad desde el origen



detalles de humedad, golpes o aperturas durante trayecto

- Herramientas limitadas:** Proveedores envían PDFs básicos o WhatsApp con "todo va bien" sin respaldo de datos reales
- Iniciativas de otros diseñadores:** Colegas en retail mencionan plataformas donde sus proveedores comparten links de tracking con temperatura, ubicación y eventos en tiempo real

5. ¿Qué es lo que hace?

- Crea tableros analógicos:** Mantiene carpetas físicas con documentos de entrega organizados por proveedor y fecha
- Uso de herramientas variadas:** Combina llamadas telefónicas, mensajes WhatsApp, inspección táctil y revisión de fechas de vencimiento
- Presentación a clientes:** Muestra certificados de proveedores cuando clientes preguntan sobre frescura y calidad

PAINS

- Recibir productos que aunque frío, han perdido calidad:** Productos que "llegan fríos" al tacto pero sufrieron picos de temperatura, golpes o humedad excesiva durante trayecto
- No tener certeza de la calidad:** Decisiones críticas de aceptación basadas en métodos subjetivos sin datos históricos de transporte
- Riesgo para la salud:** Temor constante de servir productos comprometidos que enfermen clientes o dañen reputación de sus restaurantes
- Pérdida de tiempo:** Horas dedicadas a inspecciones, llamadas de seguimiento y gestión de reclamos sin evidencia objetiva

GAINS

- Seguridad y confianza:** Acceso a trazabilidad completa que muestre temperatura, humedad, golpes y aperturas durante todo el transporte
- Transparencia total del proceso:** Links de tracking compartidos automáticamente donde visualice condiciones en tiempo real y reciba notificaciones proactivas
- Reducción de pérdidas económicas:** Tomar decisiones informadas de aceptación/rechazo basadas en datos objetivos, evitando productos comprometidos
- Herramientas digitales confiables:** Reportes profesionales descargables con gráficos claros que pueda archivar para auditorías y usar como evidencia ante problemas

4. ¿Qué es lo que dice?

- “
- Necesidad de orden:** "Necesito saber que mi comida llegó bien conservada, no solo confiar en que 'se ve fría'"
 - Frustración visual:** "Paso más tiempo inspeccionando y preocupándome que usando los productos con confianza"
 - Moodboards como ideal:** "Quisiera ver en una pantalla simple: ¿se mantuvo la temperatura?, ¿hubo golpes?, ¿llegará a tiempo?"
- ”

UXPRESSIA

This persona was built in upressoia.com

El empathy map de María revela a una profesional responsable que prioriza la calidad y la confianza en sus proveedores. Sus principales preocupaciones se centran en la reputación de su negocio y la satisfacción de sus clientes finales. Valora la transparencia en el proceso de transporte, documentación clara de cumplimiento, y la capacidad de tomar decisiones informadas sobre la aceptación de productos. Su dolor principal es la incertidumbre sobre las condiciones de transporte y la falta de información confiable que le permita verificar la calidad de los productos. Su ganancia principal es tener acceso a información transparente y reportes de cumplimiento que le den confianza para aceptar productos y mantener la calidad en su negocio.

2.4. Big Picture EventStorming

El Big Picture Event Storming es una técnica colaborativa que permitió al equipo comprender el dominio del negocio de monitoreo de transporte de carga sensible de manera visual y participativa. Esta metodología se utilizó como una primera aproximación para mapear los procesos clave, identificar eventos de negocio significativos, analizar la interacción entre actores (gestores de transporte, conductores, clientes finales) y sistemas tecnológicos (sensores IoT, plataforma de monitoreo) y descubrir oportunidades de mejora que guiarán el diseño de la solución. A través de la representación en una línea de tiempo de los eventos, se generó un lenguaje compartido entre todos los participantes, reduciendo ambigüedades y facilitando la alineación estratégica para las siguientes fases del desarrollo.

Link Big Picture Event Storming: [Big Picture Event Storming](#)

1. Preparing the Room

El equipo inició la sesión preparando el espacio de trabajo colaborativo. Se creó un tablero compartido en Miro para registrar los eventos a medida que ocurrían y se definió el propósito principal: analizar el dominio del monitoreo de transporte de carga sensible, abarcando todo el ciclo de vida del servicio, desde la solicitud inicial hasta la evaluación posterior a la entrega.

2. Energizing the audience

Para involucrar a los participantes, se realizaron actividades de integración que resaltaron la importancia del proceso. También se presentaron ejemplos prácticos de "eventos de negocio" basados en la guía "Step by Step Guide to run your Big Picture Event Storming", con el fin de ilustrar cómo estos elementos contribuyen a construir una comprensión integral del sistema.

3. Briefing and Presenting the Agenda

Se explicó el alcance de la sesión:

- Analizar cómo interactúan los gestores de transporte, conductores y clientes finales
- Identificar los eventos clave desde que se solicita un servicio de transporte hasta la entrega, monitoreo y análisis de datos
- Identificar a los actores y sistemas externos

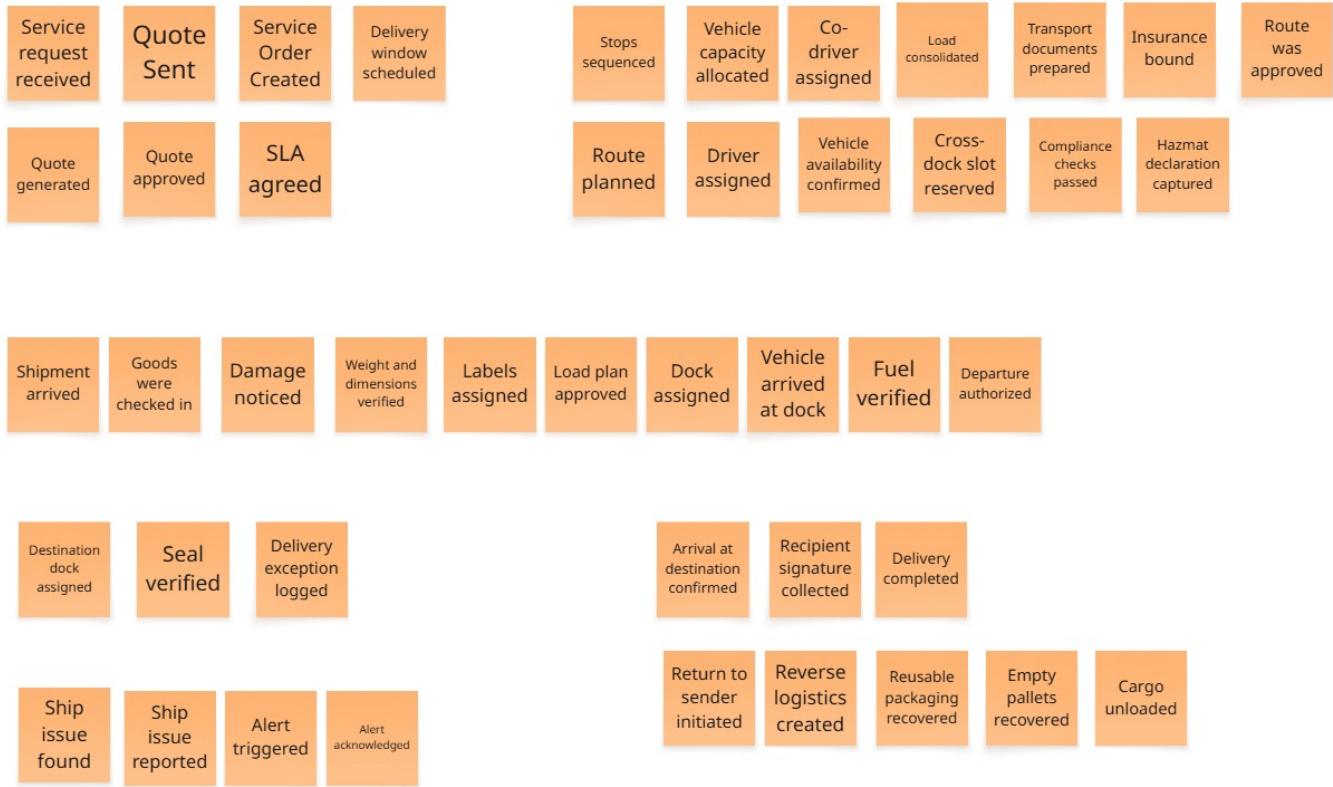
4. Generating Domain Events

Los participantes propusieron todos los eventos posibles sin orden específico, anotándolos en tarjetas.



5. Sorting Domain Events

Se organizó la secuencia de eventos siguiendo una línea de tiempo lógica desde el inicio (solicitud de envíos) hasta el final (cierre de operación). Esto permitió visualizar de forma clara el flujo completo y detectar posibles cuellos de botella o redundancias.



6. Adding Actors and External Systems

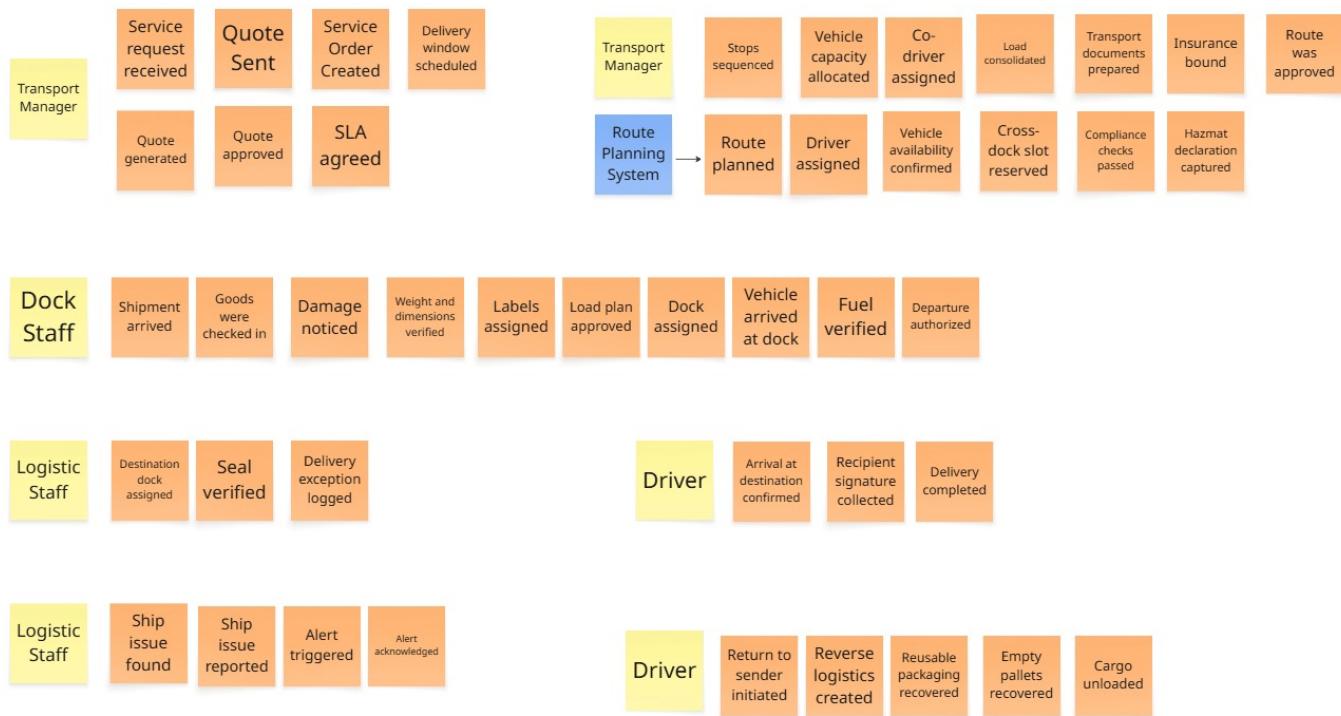
Se identificaron los actores clave que participan en los eventos:

Actores:

- **Transport Manager (Gestor de Transporte)** : Gestiona operaciones, rutas, costos y SLA
- **Dock Staff (Personal del Muelle)** : Encargado de la carga, descarga y verificación de mercancía en el muelle. Asegura que los productos correctos se carguen en los vehículos apropiados.
- **Driver (Conductor)** : Encargado de ejecutar el viaje
- **Logistic Staff (Personal Logístico)** : Coordina el flujo de mercancías desde el origen hasta el destino. Gestiona el inventario, el almacenamiento y la documentación

Sistemas Externos:

- **Route Planning System** : Sistema encargado de calcular las rutas más eficientes para la flota de transporte, considerando variables como la distancia, el tiempo estimado de viaje, las ventanas de entrega (SLAs), la capacidad del vehículo y los costos asociados.



7. Storytelling

El equipo narró la experiencia desde dos perspectivas:

Perspectiva del Gestor de Transporte:

Un gestor de transporte recibe una solicitud de servicio, genera una cotización, acuerda un SLA con el cliente y planifica la ruta considerando paradas, capacidad del vehículo y requisitos especiales. Prepara la documentación, asigna un conductor y un vehículo, y reserva slots en cross-docks. Al llegar el cargamento al muelle, se verifican las mercancías, se asignan etiquetas y se aprueba el plan de carga. Durante el viaje, el gestor no tiene visibilidad real, pues depende de las llamadas esporádicas del conductor para saber la ubicación y el estado de la carga. Si hay un problema de temperatura o desvío, solo se entera después del hecho.

Perspectiva del Cliente Final:

El cliente no recibe notificaciones de estado ni puede rastrear su envío en tiempo real. Al recibir la mercancía, solo puede hacer una inspección visual, pues no hay datos de temperatura o vibración para validar. La falta de datos históricos y verificables complica la confianza y hace que la resolución de cualquier problema o reclamo sea lenta y dependiente del papeleo.

8. Reverse Storytelling

Se utilizó la técnica de narración inversa para verificar la consistencia del flujo de transporte manual. Comenzando desde el cierre de la operación (recepción de la mercancía por el Cliente Final con inspección visual) y la lenta resolución de reclamos, y retrocediendo paso a paso hasta el evento inicial de solicitud de servicio. Esto ayudó a confirmar que no faltaba ningún evento crítico, que las políticas de negocio estaban correctamente representadas y que los puntos ciegos (falta de visibilidad en tránsito) y la dependencia del papeleo eran la causa principal de la inefficiencia en el modelo actual.

9. Closing

Se recopilaron los principales aprendizajes y las necesidades de mejora no cubiertas por el proceso manual:

- La dependencia del factor humano (llamadas del conductor, papeleo) introduce puntos ciegos críticos en la visibilidad del transporte.
- Existe una necesidad fundamental de manejar eventos críticos (temperatura, humedad, vibración, puerta, ubicación) en tiempo real para permitir una acción correctiva proactiva del gestor.
- La trazabilidad y la validación de la condición de la carga son insuficientes; la inspección visual al final del trayecto no es una prueba de cumplimiento para la mercancía sensible.
- Es crucial documentar políticas de negocio claras sobre las desviaciones (tiempos de gracia, umbrales de temperatura, tarifas) para evitar conflictos y largas disputas.
- La experiencia del cliente final es pobre debido a la falta de transparencia y el nulo acceso a datos del envío, lo cual deteriora la confianza en el proveedor.
- La resolución de disputas y el cumplimiento normativo se ven severamente obstaculizados por la ausencia de reportes automáticos y datos históricos verificables.

2.5. Ubiquitous Language

El Ubiquitous Language de CargoSafe establece un vocabulario común y preciso entre el equipo de desarrollo, stakeholders del negocio y usuarios finales, eliminando ambigüedades en la comunicación sobre el dominio de monitoreo de cadena de frío.

Términos del Dominio Central

Cold Chain (Cadena de Frío)

Proceso logístico que mantiene productos perecederos dentro de rangos específicos de temperatura desde el origen hasta el destino final, garantizando su calidad e integridad.

Shipment(Envío)

Conjunto de productos o carga que se transportan desde un punto de origen hacia un destino determinado bajo una misma guía o registro.

Carrier (Transportista)

Conjunto de productos o carga que se transportan desde un punto de origen hacia un destino determinado bajo una misma guía o registro.

Route Plan (Plan de ruta)

Persona o empresa responsable de trasladar los productos y asegurar que se cumplan las condiciones de transporte definidas.

Delivery Point (Punto de entrega)

Itinerario de transporte que define el recorrido, tiempos estimados, puntos de parada y condiciones necesarias para completar la entrega.

Temperature Range (Rango de temperatura)

Margen aceptable de grados dentro del cual debe mantenerse el producto durante el transporte para conservar su calidad.

Temperature Deviation (Desviación de temperatura)

Diferencia entre la temperatura real de la carga y el rango permitido, que puede afectar la integridad del producto.

Términos de Actores**Fleet Manager (Gestor de Flota)**

Usuario responsable de configurar dispositivos, establecer parámetros de viaje, monitorear el estado de múltiples transportes y generar reportes para clientes.

Driver (Conductor)

Usuario operativo que recibe alertas durante el transporte y ejecuta acciones correctivas cuando se detectan problemas en la cadena de frío.

Final Client (Cliente Final)

Receptor de la carga que requiere visibilidad del estado del transporte y documentación de cumplimiento térmico para aceptar o rechazar productos.

Términos de Procesos de Negocio**Trip Configuration (Configuración de Viaje)**

Proceso de establecer parámetros específicos antes del inicio del transporte: rangos de temperatura, duración estimada, tipo de carga y responsables.

Corrective Action (Acción Correctiva)

Medidas tomadas por el conductor u operador para resolver violaciones de temperatura: ajuste de refrigeración, cambio de ruta, o notificación a supervisores.

Temperature Calibration (Calibración de Temperatura)

Proceso de verificación y ajuste de sensores para garantizar precisión en las mediciones, realizado periódicamente según estándares de calidad.

Cold Chain Compliance (Cumplimiento de Cadena de Frío)

Estado que confirma que un viaje se completó dentro de todos los parámetros térmicos requeridos, cumpliendo con regulaciones y estándares de calidad.

Contexto de Métricas y KPIs**Temperature Excursion (Excusión de Temperatura)**

Período específico durante el cual la temperatura estuvo fuera del rango permitido, medido en minutos u horas según criticidad del producto.

Compliance Rate (Tasa de Cumplimiento)

Porcentaje de viajes que se completaron sin violaciones de temperatura en un período determinado, utilizado como KPI principal del servicio.

Alert Response Time (Tiempo de Respuesta a Alertas)

Métrica que mide el tiempo transcurrido entre la generación de una alerta y la implementación de acciones correctivas por parte del equipo operativo.

Capítulo III: Requirements Specification

3.1. User Stories

Epic / Story ID	Título	Descripción	Criterios de Aceptación	Relacionado con (Epic ID)
E1	Landing Page	Página principal con secciones informativas y de contacto para captar y orientar a los visitantes.		
E2	Autenticación	Módulo de registro e inicio de sesión seguro para usuarios.		
E3	Gestión de flota	Administración de la flota: registro, actualización y baja de vehículos.		
E4	Planificador de viajes	Creación y actualización de estados de los viajes.		
E5	Monitoreo en tiempo real	Engloba funcionalidades de monitoreo de temperatura en tiempo real y alertas.		
E6	Dashboard de viajes	Engloba pantallas, gráficos e informes relacionados a los viajes.		
E7	Módulo de suscripciones	Engloba funcionalidades de pago, manejo y control de suscripciones.		
US001	Navegación en landing page	Como visitante quiero acceder a diferentes secciones del servicio para informarme sobre la plataforma.	Scenario: Acceso a información principal Given un visitante accede a la landing page When solicita información sobre el servicio Then obtiene el contenido correspondiente a cada sección (Inicio, Características, Planes, Contacto)	
			Scenario: Acceso completo a todas las secciones Given un visitante quiere explorar la plataforma When realiza solicitudes de información sobre distintas secciones Then recibe los datos correctos de cada sección sin errores	E1
US002	Sección portada	Como visitante, quiero conocer el mensaje principal del servicio, para entender rápidamente el propósito de la plataforma.	Scenario: Mensaje principal disponible Given un visitante accede a la plataforma When solicita información general sobre el servicio Then recibe un mensaje claro que describe el propósito y la propuesta de valor	E1
US003	Sección de funcionalidades	Como visitante, quiero conocer las funcionalidades principales de la plataforma, para entender qué servicios puedo utilizar.	Scenario: Funcionalidades disponibles Given un visitante accede a la plataforma When solicita información sobre las funcionalidades Then el sistema entrega al menos tres funcionalidades principales de la plataforma	E1
US004	Sección de beneficios	Como visitante, quiero conocer los beneficios de la plataforma, para entender qué valor obtengo al usarla.	Scenario: Beneficios disponibles Given un visitante solicita información sobre la plataforma When el sistema procesa la solicitud Then entrega la lista de beneficios disponibles de forma clara y completa	E1
US005	Sección de testimonios	Como visitante, quiero conocer experiencias de otros clientes, para generar confianza en el servicio.	Scenario: Testimonios disponibles Given un visitante solicita información sobre la experiencia de otros usuarios When el sistema procesa la solicitud Then entrega al menos dos testimonios verificados de clientes o usuarios con sus datos generales	E1

Epic / Story ID	Título	Descripción	Criterios de Aceptación	Relacionado con (Epic ID)
US006	Sección de contáctanos	Como visitante, quiero enviar consultas o solicitudes a la empresa, para recibir información o soporte.	Scenario: Envío de mensaje válido Given un visitante proporciona nombre, correo y mensaje válidos When envía la información al sistema Then el sistema registra la solicitud y confirma que fue recibida	E1
			Scenario: Envío de mensaje con datos inválidos Given un visitante omite datos requeridos When envía la información Then el sistema rechaza la solicitud y notifica al visitante sobre los campos faltantes o incorrectos	
US007	Call to Action a la aplicación web	Como visitante quiero acceder a la aplicación web para registrarme o iniciar sesión y usar el servicio desde un navegador.	Scenario: Acceso a la app web Given un visitante quiere usar la plataforma web When solicita iniciar sesión o registrarse Then el sistema le permite autenticarse y acceder a las funcionalidades disponibles	E1
US008	Call to Action de descarga de App Móvil	Como visitante quiero descargar la aplicación móvil para instalarla en mi dispositivo y usar la plataforma.	Scenario: Descarga de la app móvil Given un visitante desea usar la app móvil When solicita la descarga Then el sistema proporciona los medios correctos para descargar la aplicación según su dispositivo (iOS o Android)	E1
US009	Registro de usuario	Como usuario quiero registrarme en la plataforma para acceder a mi cuenta y funcionalidades personalizadas.	Scenario: Registro exitoso Given un usuario no registrado When proporciona credenciales válidas Then el sistema crea la cuenta y permite acceso a la plataforma	E2
			Scenario: Registro inválido Given un usuario no registrado When proporciona credenciales incorrectas o incompletas Then el sistema rechaza la creación de la cuenta y notifica el error	
US010	Inicio de sesión	Como usuario quiero iniciar sesión en la plataforma para acceder a mi cuenta y funcionalidades personalizadas.	Scenario: Inicio de sesión válido Given un usuario registrado When ingresa credenciales correctas Then el sistema autentica y permite acceso	E2
			Scenario: Inicio de sesión inválido Given un usuario registrado When ingresa credenciales incorrectas Then el sistema rechaza la autenticación y muestra error	
US011	Cerrar sesión	Como usuario autenticado quiero cerrar sesión desde la aplicación para que mi cuenta deje de estar accesible en el dispositivo actual.	Scenario 1: Cierre de sesión exitoso Given que el usuario tiene un token válido y está autenticado When solicita cerrar sesión Then se elimina el token del lado del cliente And los siguientes intentos de acceder a recursos requieren autenticación nuevamente	E2

Epic / Story ID	Título	Descripción	Criterios de Aceptación	Relacionado con (Epic ID)
US012	Recuperar contraseña	<p>Como usuario</p> <p>quiero recuperar el acceso a mi cuenta mediante un proceso de restablecimiento de contraseña</p> <p>para poder ingresar nuevamente en caso de olvidar mis credenciales.</p>	<p>Scenario 1: Solicitud de recuperación Given un cliente no recuerda su contraseña When solicita la recuperación en la pantalla de inicio de sesión e ingresa su correo registrado Then el sistema envía un correo electrónico con un enlace seguro para restablecer la contraseña</p> <p>Scenario 2: Enlace válido y único Given el cliente recibe un enlace de restablecimiento When hace clic en el enlace dentro del tiempo de validez (ej. 15 minutos) Then el sistema muestra un formulario para crear una nueva contraseña And valida que el enlace no haya sido usado previamente</p> <p>Scenario 3: Enlace expirado Given un cliente recibe un enlace de restablecimiento When intenta usarlo después del tiempo de validez Then el sistema rechaza el intento And muestra el mensaje "El enlace ha expirado, solicita nuevamente la recuperación de contraseña"</p> <p>Scenario 4: Contraseña actualizada correctamente Given el cliente ingresa una nueva contraseña válida When confirma el restablecimiento Then el sistema actualiza la contraseña en la base de datos And notifica al cliente que el cambio se realizó con éxito</p>	E2

Epic / Story ID	Título	Descripción	Criterios de Aceptación	Relacionado con (Epic ID)
US013	Registro de vehículos de carga	Como administrador logístico, quiero registrar vehículos en la plataforma, para mantener un inventario actualizado de la flota.	<p>Scenario: Registrar un vehículo válido Given el administrador cuenta con los datos de marca, modelo, placa, capacidad y dimensiones When el administrador registra el vehículo en la plataforma Then el sistema guarda el vehículo como parte de la flota activa</p> <p>Scenario: Registro sin datos obligatorios Given el administrador omite campos requeridos como marca, modelo o placa When intenta registrar el vehículo Then el sistema rechaza el registro y muestra un error de validación</p> <p>Scenario: Validar formato de placa Given el administrador ingresa un formato de placa incorrecto When intenta registrar el vehículo Then el sistema rechaza el registro y solicita un formato válido</p> <p>Scenario: Validar dimensiones del vehículo Given el administrador ingresa dimensiones físicas del vehículo (alto, ancho, largo) When las dimensiones son numéricas y positivas Then el sistema acepta los valores y guarda la información</p> <p>Scenario: Validar capacidad de carga Given el administrador ingresa capacidad de carga en kilogramos When la capacidad es mayor a 0 Then el sistema guarda el valor correctamente And el sistema rechaza valores negativos o nulos</p>	E3
US014	Actualización de datos de vehículos de carga	Como administrador logístico, quiero editar la información de los vehículos, para mantener actualizado su estado operativo.	<p>Scenario: Actualizar información básica del vehículo Given existe un vehículo registrado en la plataforma When el administrador modifica la marca, modelo o dimensiones Then el sistema actualiza la información y guarda los cambios en el historial</p> <p>Scenario: Validar actualización de placa duplicada Given un vehículo ya tiene registrada una placa única When el administrador intenta asignar esa misma placa a otro vehículo Then el sistema rechaza la actualización y muestra un error de duplicidad</p> <p>Scenario: Mantener consistencia en datos obligatorios Given existe un vehículo registrado When el administrador elimina un dato obligatorio como marca o modelo Then el sistema rechaza la actualización y mantiene los datos anteriores</p>	E3

Epic / Story ID	Título	Descripción	Criterios de Aceptación	Relacionado con (Epic ID)
US015	Eliminación de vehículos de carga	Como administrador logístico, quiero eliminar vehículos de la plataforma, para mantener un inventario actualizado y evitar registros obsoletos.	Scenario: Eliminar vehículo activo Given un vehículo está registrado en la flota When el administrador elimina el vehículo Then el sistema cambia su estado a "inactivo" And conserva su historial de viajes y mantenimientos	
			Scenario: Intentar eliminar vehículo con viaje en curso Given un vehículo tiene un viaje activo When el administrador intenta eliminarlo Then el sistema rechaza la eliminación And muestra un mensaje indicando que el vehículo tiene viajes pendientes	E3
			Scenario: Eliminación exitosa de vehículo sin viajes en curso Given un vehículo no tiene viajes activos When el administrador lo elimina Then el sistema lo marca como eliminado And ya no aparece en el inventario activo de la flota	
US016	Registro de dispositivos IoT	Como Administrador Logístico quiero registrar un dispositivo IoT en la plataforma para asociarlo a la flota y comenzar a recibir sus datos.	Scenario: Registro exitoso de dispositivo nuevo Given un dispositivo IoT con un ID único When el administrador lo registra en la plataforma Then el sistema almacena el dispositivo en la base de datos And queda disponible para asignarlo a un vehículo de carga	
			Scenario: Intentar registrar dispositivo duplicado Given un dispositivo IoT con un ID ya registrado When el administrador intenta registrarlo nuevamente Then el sistema rechaza el registro And muestra un mensaje "El dispositivo ya está registrado"	E3
			Scenario: Validación de campos obligatorios Given un administrador intenta registrar un dispositivo sin datos requeridos como deviceld o tipo When envía la solicitud de registro Then el sistema rechaza el registro And muestra un mensaje indicando los campos faltantes	

Epic / Story ID	Título	Descripción	Criterios de Aceptación	Relacionado con (Epic ID)
US017	Eliminar dispositivo IoT	<p>Como Administrador Logístico, quiero eliminar un dispositivo IoT de la plataforma, para darlo de baja en caso de falla o reemplazo.</p>	<p>Scenario: Eliminar dispositivo inactivo Given un dispositivo IoT está registrado en la plataforma y no transmite datos When el administrador elimina el dispositivo Then el sistema cambia su estado a "<i>inactivo</i>" And conserva su historial de transmisiones y alertas</p> <p>Scenario: Intentar eliminar dispositivo activo Given un dispositivo está enviando datos en tiempo real When el administrador intenta eliminarlo Then el sistema rechaza la eliminación And muestra un mensaje indicando que el dispositivo sigue activo</p> <p>Scenario: Eliminación exitosa de dispositivo sin actividad reciente Given un dispositivo no transmite datos desde hace más de 24h When el administrador lo elimina Then el sistema lo marca como "<i>eliminado</i>" And ya no aparece en la lista de dispositivos activos</p>	
US018	Asignar dispositivo a vehículo de carga	<p>Como Administrador Logístico quiero asignar un dispositivo IoT a un vehículo de carga para identificar a qué unidad pertenece cada transmisión de datos</p>	<p>Scenario: Asignar dispositivo a vehículo disponible Given un dispositivo IoT registrado y sin asignar When el administrador selecciona un vehículo de la flota Then el sistema vincula el dispositivo a ese vehículo And las próximas transmisiones aparecen bajo ese vehículo</p> <p>Scenario: Intentar asignar dispositivo ya vinculado Given un dispositivo IoT ya asignado a un vehículo When el administrador intenta asignarlo a otro sin liberar la relación Then el sistema rechaza la acción And muestra el mensaje "Dispositivo ya vinculado"</p>	E3
US019	Cambiar dispositivo de vehículo	<p>Como Administrador Logístico quiero reasignar un dispositivo IoT de un vehículo a otro para reutilizarlo en caso de mantenimiento o rotación de la flota</p>	<p>Scenario: Reasignar dispositivo a otro vehículo Given un dispositivo IoT asignado a un vehículo When el administrador lo reasigna a un vehículo diferente Then el sistema actualiza la relación And las transmisiones futuras aparecen bajo el nuevo vehículo</p> <p>Scenario: Confirmar cambio en inventario Given un dispositivo IoT reasignado de un vehículo a otro When el administrador consulta el vehículo anterior Then el dispositivo ya no aparece en su lista de dispositivos asociados</p>	E3

Epic / Story ID	Título	Descripción	Criterios de Aceptación	Relacionado con (Epic ID)
US020	Ver vehículos de carga	<p>Como Administrador Logístico quiero visualizar la lista de vehículos de carga registrados en la plataforma para supervisar y gestionar el inventario</p>	Scenario: Consulta de lista de vehículos Given existen vehículos registrados en la base de datos When el administrador accede a la sección de vehículos Then el sistema muestra la lista con placa, modelo, estado y capacidad de cada vehículo	E3
			Scenario: Lista vacía de vehículos Given no existen vehículos registrados en la base de datos When el administrador consulta la sección de vehículos Then el sistema muestra un mensaje "No hay vehículos registrados"	
			Scenario: Consulta de detalle de vehículo Given un vehículo está registrado en la base de datos When el administrador selecciona un vehículo de la lista Then el sistema muestra su información completa	
US021	Ver dispositivos IoT	<p>Como Administrador Logístico quiero visualizar la lista de dispositivos IoT registrados en la plataforma para supervisar su estado y administración</p>	Scenario: Consulta de lista de dispositivos Given existen dispositivos IoT registrados en la base de datos When el administrador accede a la sección de dispositivos Then el sistema muestra la lista con ID, tipo, estado y vehículo asignado de cada dispositivo	E3
			Scenario: Lista vacía de dispositivos Given no existen dispositivos IoT registrados en la base de datos When el administrador consulta la sección de dispositivos Then el sistema muestra un mensaje "No hay dispositivos registrados"	
			Scenario: Consulta de detalle de dispositivo Given un dispositivo IoT está registrado en la base de datos When el administrador selecciona un dispositivo de la lista Then el sistema muestra su información completa	
US022	Ver estado de dispositivo por vehículo	<p>Como Administrador Logístico quiero visualizar el estado de todos los dispositivos asociados a un vehículo en el dashboard para monitorear su funcionamiento</p>	Scenario: Ver dispositivos conectados Given un vehículo con dispositivos asignados When el administrador lo selecciona en el dashboard Then se listan los dispositivos con estado online u offline	E3
			Scenario: Detectar dispositivo desconectado Given un dispositivo deja de transmitir datos por más de 5 minutos When el sistema verifica su estado Then lo marca como "Desconectado"	
			Scenario: Recuperar estado conectado Given un dispositivo previamente desconectado When transmite datos nuevamente Then la plataforma actualiza su estado a "Conectado"	

Epic / Story ID	Título	Descripción	Criterios de Aceptación	Relacionado con (Epic ID)
US023	Ver estado de dispositivo	<p>Como usuario del dispositivo quiero conocer el estado del dispositivo para verificar rápidamente si funciona correctamente.</p>	Scenario: Dispositivo activo y funcionando Given el dispositivo está encendido y transmite datos correctamente When el usuario solicita el estado del dispositivo Then el sistema indica que el dispositivo está operativo y conectado	
			Scenario: Dispositivo sin conexión Given el dispositivo ha perdido conectividad When el usuario solicita el estado del dispositivo Then el sistema indica que el dispositivo no está transmitiendo datos	E3
			Scenario: Batería baja Given el dispositivo tiene un nivel de batería inferior al 20% When el usuario solicita el estado del dispositivo Then el sistema alerta sobre el nivel bajo de batería	
US024	Creación de viajes	<p>Como administrador logístico, quiero crear viajes asignando vehículo y ruta, para planificar el transporte de mercancías.</p>	Scenario: Dispositivo apagado Given el dispositivo ha sido apagado When el usuario solicita el estado del dispositivo Then el sistema indica que el dispositivo está apagado o inactivo	
			Scenario: Crear viaje válido Given existe un vehículo disponible When el administrador crea un viaje con origen y destino Then el sistema registra el viaje con estado "Pendiente"	E4
			Scenario: Cambiar estado de viaje a "En ruta" Given existe un viaje con estado "Pendiente" When el administrador cambia su estado a "En ruta" Then el sistema actualiza el estado del viaje a "En ruta" And notifica a los clientes asociados	
US025	Actualización de estados de viaje	<p>Como administrador logístico, quiero actualizar el estado de un viaje, para mantener informados a gerentes y clientes sobre el progreso de la entrega.</p>	Scenario: Cambiar estado de viaje a "Finalizado" Given existe un viaje en estado "En ruta" When el administrador cambia el estado a "Finalizado" Then el sistema marca el viaje como finalizado And registra la fecha y hora de finalización	E4
			Scenario: Intentar cambiar estado de un viaje cancelado Given un viaje está en estado "Cancelado" When el administrador intenta modificar el estado Then el sistema rechaza la actualización And muestra un mensaje de restricción	
			Scenario: Reprogramar un viaje válido Given existe un viaje programado con estado "Pendiente" When el administrador modifica la fecha y hora de inicio Then el sistema actualiza la información del viaje y guarda el cambio en el historial	
US026	Reprogramación de viajes	<p>Como administrador logístico, quiero reprogramar un viaje ya creado, para ajustar fechas y horarios en caso de cambios operativos.</p>	Scenario: Intentar reprogramar un viaje en curso Given existe un viaje con estado "En ruta" When el administrador intenta cambiar la fecha u hora de inicio Then el sistema rechaza la reprogramación y muestra un mensaje de restricción	E4

Epic / Story ID	Título	Descripción	Criterios de Aceptación	Relacionado con (Epic ID)
US027	Código de viaje para cliente	Como cliente final, quiero recibir un código único de viaje, para poder consultar fácilmente el estado de mi pedido.	Scenario: Generar código de viaje al crear pedido Given el administrador crea un viaje nuevo When el sistema registra el viaje Then se genera un código único y se envía al cliente por correo o notificación en la aplicación	E4
US028	Actualización de temperatura en tiempo real	Como empresa, quiero recibir la temperatura en tiempo real de mis dispositivos IoT, para supervisar la cadena de frío de los viajes.	Scenario: Consultar estado con código de viaje Given un cliente cuenta con el código de su viaje When ingresa el código en la aplicación Then el sistema muestra el estado actual del viaje en tiempo real	E4
US029	Alertas por incumplimiento de temperatura	Como cliente final, quiero recibir alertas cuando la temperatura sobrepasa los límites definidos para tomar acciones correctivas.	Given un sensor IoT transmite datos de temperatura When el sistema recibe los datos Then la temperatura se actualiza en la plataforma sin retrasos.	E4
US030	Alertas de conexión IoT	Como empresa, quiero recibir alertas cuando un dispositivo IoT deja de enviar datos para actuar de inmediato.	Given la temperatura excede el rango permitido When el sistema procesa la lectura Then se genera una alerta y se notifica al usuario.	E5
US031	Roles y permisos de acceso	Como empresa, quiero que el sistema gestione roles y permisos de usuarios (admin, cliente, operador) para controlar accesos.	Given un dispositivo deja de transmitir por más de X minutos When el sistema detecta la ausencia Then genera una alerta para la empresa	E5
US032	Lista de viajes registrados	Como empresa, quiero ver una lista de todos los viajes registrados para gestionarlos de forma rápida.	Given un usuario con rol administrador When accede al panel de gestión Then puede realizar todas las operaciones.	E5
US033	Detalle de viaje	Como cliente final, quiero consultar el detalle de un viaje para verificar información específica como ruta, estado y temperatura.	Given un usuario con rol cliente When accede al sistema Then solo puede ver información de sus viajes y suscripciones	E6
US034	Gráficos de tiempo y temperatura	Como cliente final, quiero ver gráficos de evolución de la temperatura durante el viaje para verificar el cumplimiento de parámetros.	Given un viaje cuenta con datos de temperatura When se consulta el detalle Then el sistema despliega un gráfico de líneas con los valores.	E6
US035	Gráficos de incidencias por mes	Como empresa, quiero ver un gráfico mensual de incidencias para identificar patrones de fallos.	Given existen incidencias en el histórico When el usuario consulta el dashboard Then se muestra un gráfico con número de incidencias por mes.	E6
US036	Filtrado de viajes por fecha	Como empresa, quiero filtrar la lista de viajes por rango de fechas para analizar un periodo específico.	Given un usuario aplica un filtro de fechas When el sistema procesa la consulta Then se muestran solo los viajes que cumplen el criterio.	E6

Epic / Story ID	Título	Descripción	Criterios de Aceptación	Relacionado con (Epic ID)
US037	Descarga de reporte de viajes	Como cliente final, quiero descargar un reporte en PDF de un viaje con su información y gráficos para archivarlo o compartirlo.	Given un usuario selecciona un viaje When solicita la exportación Then el sistema genera y entrega un archivo PDF con la información del viaje.	E6
US038	Cancelar suscripción	Como cliente final, quiero cancelar mi suscripción para detener los cobros futuros.	Given un cliente con suscripción activa When solicita la cancelación Then el sistema marca la suscripción como cancelada y detiene próximos pagos.	E7
US039	Visualizar información de suscripción	Como cliente final, quiero ver mi estado de suscripción y fecha de expiración para gestionar mi acceso al servicio.	Given un cliente autenticado accede a la sección de suscripción When consulta su información Then el sistema muestra el plan, estado y fecha de expiración.	E7
US040	Historial de pagos	Como cliente final, quiero consultar mi historial de pagos para verificar mis transacciones.	Given un cliente autenticado When accede al historial de pagos Then el sistema muestra todas las transacciones registradas con fecha y monto.	E7
US041	Notificación de renovación próxima	Como cliente final, quiero recibir una notificación antes de que mi suscripción se renueve para decidir si continúo o cancelo.	Given una suscripción está próxima a renovarse (ej. 3 días antes) When el sistema procesa la fecha Then envía una notificación al cliente.	E7

Epic / Story ID	Título	Descripción	Criterios de Aceptación	Relacionado con (Epic ID)
TS001	API de autenticación	<p>Como developer</p> <p>quiero implementar múltiples endpoints de autenticación (login, logout, refresh y validación de sesión)</p> <p>para que los usuarios puedan gestionar de forma segura su acceso a la plataforma.</p>	<p>Scenario: Login exitoso Given un usuario envía credenciales válidas al endpoint de login When el backend valida usuario y contraseña Then el sistema responde con código 200 OK And devuelve un token de acceso y un refresh token</p> <p>Scenario: Login fallido Given un usuario envía credenciales inválidas al endpoint de login When el backend intenta validarlas Then el sistema responde con código 401 Unauthorized And devuelve un mensaje de error de autenticación</p> <p>Scenario: Logout exitoso Given un usuario tiene sesión activa con un refresh token válido When envía solicitud al endpoint de logout Then el backend responde con código 200 OK And el refresh token queda invalidado</p> <p>Scenario: Registro exitoso de usuario Given un nuevo usuario envía datos válidos (correo, contraseña y nombre) al endpoint de registro When el backend valida la información y la almacena en la base de datos Then el sistema responde con código 201 Created And devuelve un mensaje confirmando la creación del usuario</p> <p>Scenario: Registro con correo duplicado Given un correo ya existe en la base de datos When un usuario intenta registrarse con ese mismo correo Then el sistema responde con código 409 Conflict And devuelve un mensaje "El correo ya está registrado"</p> <p>Scenario: Refresh token válido Given un usuario tiene un refresh token válido When envía solicitud al endpoint de refresh Then el backend responde con código 200 OK And devuelve un nuevo par de tokens de acceso y refresh</p> <p>Scenario: Token inválido o expirado en validación Given un usuario envía un token inválido o expirado al endpoint de validación When el backend procesa el token Then el sistema responde con código 401 Unauthorized And devuelve un mensaje "Token inválido o expirado"</p>	E2

Epic / Story ID	Título	Descripción	Criterios de Aceptación	Relacionado con (Epic ID)
TS002	Servicio de autenticación con JWT	<p>Como developer, quiero implementar autenticación basada en JWT, para asegurar la comunicación entre cliente y servidor.</p>	<p>Scenario: Generación de token JWT Given un usuario inicia sesión exitosamente When el backend autentica sus credenciales Then genera un token JWT con payload que incluye userId, rol y fecha de expiración</p> <p>Scenario: Verificación de token JWT válido Given un cliente envía un token JWT válido en la cabecera Authorization When el backend lo verifica con la clave secreta Then acepta la solicitud And permite acceso al recurso protegido</p> <p>Scenario: Verificación de token JWT inválido Given un cliente envía un token JWT alterado o mal formado When el backend lo valida Then rechaza la solicitud And devuelve un error 401 Unauthorized</p> <p>Scenario: Token expirado Given un cliente envía un token JWT expirado When el backend intenta validararlo Then rechaza la solicitud And devuelve un error 401 Unauthorized con mensaje "Token expirado"</p> <p>Scenario: Renovación de sesión con refresh token Given un access token está expirado When el cliente envía un refresh token válido Then el backend genera un nuevo JWT de acceso And responde con los nuevos tokens</p>	E2

Epic / Story ID	Título	Descripción	Criterios de Aceptación	Relacionado con (Epic ID)
			Scenario: Registro exitoso de vehículo Given un vehículo con datos válidos (placa, modelo, año, capacidad) When se envía una solicitud POST al endpoint de vehículos Then el sistema almacena el vehículo en la base de datos And devuelve un código 201 con el detalle del vehículo registrado	
			Scenario: Intentar registrar vehículo duplicado Given un vehículo con una placa ya existente en el sistema When el developer intenta registrarla nuevamente Then el sistema rechaza la solicitud And devuelve un código 409 con el mensaje "El vehículo ya está registrado"	
TS003	API de vehículos de carga	Como developer, quiero exponer un endpoint para registrar, modificar y consultar vehículos de carga, para que el backend gestione el inventario de la flota.	Scenario: Modificar datos de un vehículo existente Given un vehículo registrado en la base de datos When se envía una solicitud PUT con los campos actualizados Then el sistema actualiza los datos del vehículo And devuelve un código 200 con el detalle actualizado	E3
			Scenario: Consultar vehículo por ID Given un vehículo registrado en la base de datos When se envía una solicitud GET al endpoint con su identificador Then el sistema devuelve un código 200 And muestra los datos completos del vehículo	
			Scenario: Consultar lista de vehículos Given existen varios vehículos registrados en la base de datos When se envía una solicitud GET al endpoint de vehículos Then el sistema devuelve un código 200 And muestra la lista completa de vehículos disponibles en el inventario	

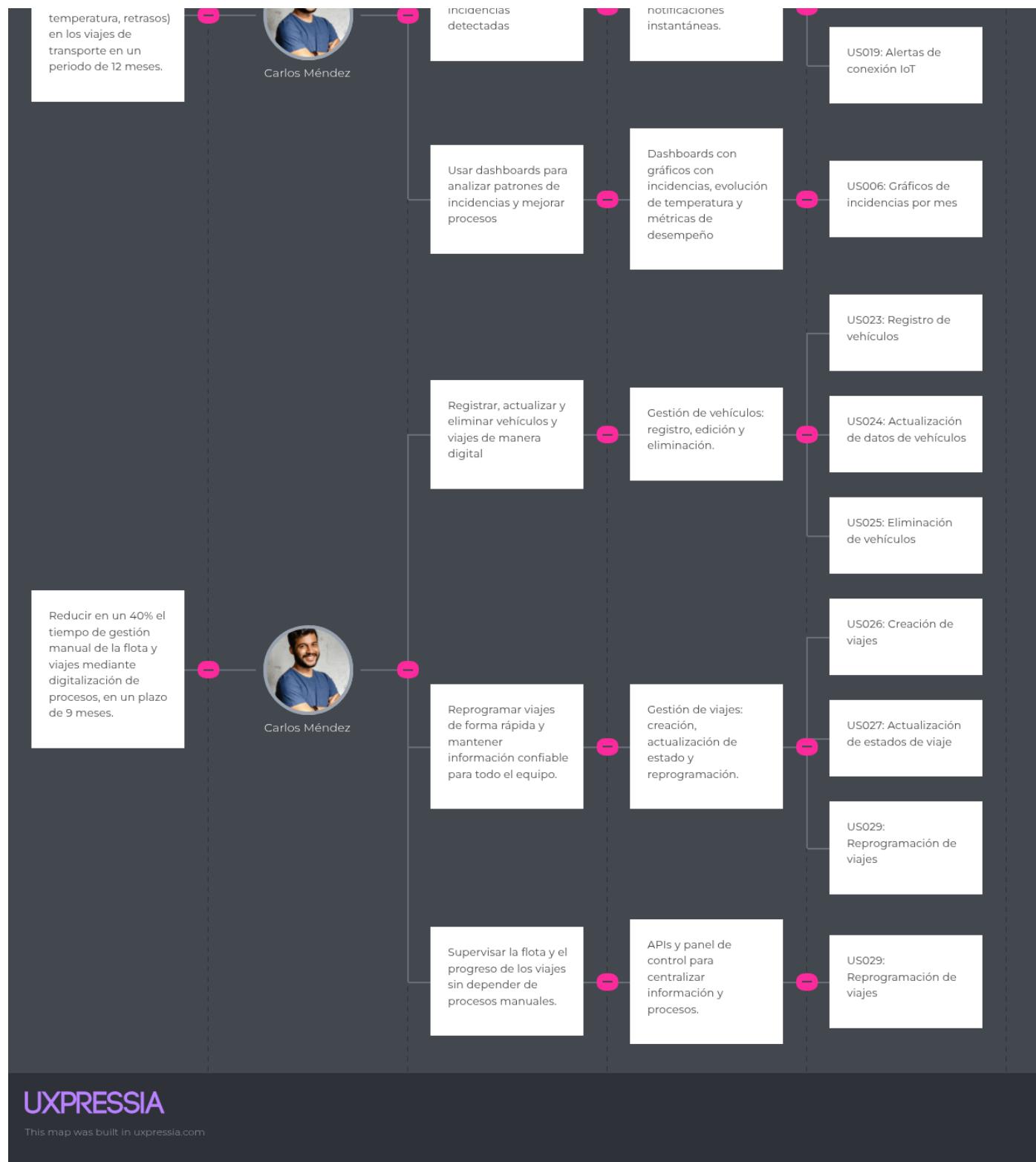
Epic / Story ID	Título	Descripción	Criterios de Aceptación	Relacionado con (Epic ID)
			Scenario: Registro exitoso de dispositivo Given un dispositivo IoT con datos válidos (deviceID, tipo, estado inicial) When se envía una solicitud POST al endpoint de dispositivos Then el sistema almacena el dispositivo en la base de datos And devuelve un código 201 con el detalle del dispositivo registrado	
			Scenario: Intentar registrar dispositivo duplicado Given un dispositivo IoT con un deviceID ya existente en la base de datos When se intenta registrarlo nuevamente Then el sistema rechaza la solicitud And devuelve un código 409 con el mensaje "El dispositivo ya está registrado"	
			Scenario: Consultar dispositivo por ID Given un dispositivo IoT registrado en la base de datos When se envía una solicitud GET al endpoint con su identificador Then el sistema devuelve un código 200 And muestra los datos completos del dispositivo	
TS004	API de dispositivos IoT	Como developer quiero exponer un endpoint de dispositivos para que la plataforma pueda registrar y administrar los dispositivos IoT.	Scenario: Consultar lista de dispositivos Given existen varios dispositivos IoT registrados en la base de datos When se envía una solicitud GET al endpoint de dispositivos Then el sistema devuelve un código 200 And muestra la lista completa de dispositivos registrados	E3
			Scenario: Modificar datos de un dispositivo Given un dispositivo IoT registrado en la base de datos When se envía una solicitud PUT con campos actualizados (ej. nombre, estado, vehículo asignado) Then el sistema actualiza los datos del dispositivo And devuelve un código 200 con el detalle actualizado	
			Scenario: Eliminar dispositivo registrado Given un dispositivo IoT existe en la base de datos y no tiene transmisiones activas When se envía una solicitud DELETE al endpoint de dispositivos Then el sistema cambia su estado a "inactivo" And conserva su historial de transmisiones y eventos	

Epic / Story ID	Título	Descripción	Criterios de Aceptación	Relacionado con (Epic ID)
TS005	API de viajes	<p>Como developer, quiero exponer un endpoint RESTful para registrar viajes, para que la aplicación guarde y gestione la información.</p>	<p>Scenario: Registro exitoso de viaje Given un cliente envía los datos válidos de un viaje (vehículo, dispositivo IoT, fecha de inicio, destino) al endpoint de viajes When el backend valida la información Then el sistema responde con código 201 Created And almacena el viaje en la base de datos</p> <p>Scenario: Registro con datos incompletos Given un cliente envía una solicitud de registro sin todos los campos obligatorios When el backend procesa la solicitud Then el sistema responde con código 400 Bad Request And devuelve un mensaje indicando los campos faltantes</p> <p>Scenario: Registro con vehículo no existente Given el identificador de vehículo no existe en la base de datos When el cliente intenta registrar un viaje con ese vehículo Then el sistema responde con código 404 Not Found And devuelve un mensaje "Vehículo no encontrado"</p> <p>Scenario: Registro con dispositivo IoT no existente Given el identificador de dispositivo IoT no existe en la base de datos When el cliente intenta registrar un viaje con ese dispositivo Then el sistema responde con código 404 Not Found And devuelve un mensaje "Dispositivo IoT no encontrado"</p> <p>Scenario: Validación de solapamiento de viajes Given un vehículo ya tiene un viaje activo en la base de datos When se intenta registrar un nuevo viaje con el mismo vehículo en paralelo Then el sistema responde con código 409 Conflict And devuelve un mensaje "El vehículo ya tiene un viaje activo"</p>	E4

3.2. Impact Mapping

Empresa (Gestores de transporte)





Clientes Finales (Consumidores finales)



3.3. Product Backlog

# Orden	User Story ID	Título	Descripción	Story Points
---------	---------------	--------	-------------	--------------

# Orden	User Story ID	Título	Descripción	Story Points
1	US028	Monitoreo de temperatura en tiempo real	Como empresa, quiero recibir la temperatura en tiempo real de mis dispositivos IoT, para supervisar la cadena de frío de los viajes.	8
2	US029	Alertas por incumplimiento de temperatura	Como cliente final, quiero recibir alertas cuando la temperatura sobrepasa los límites definidos, para tomar acciones correctivas.	5
3	US033	Detalle de viaje	Como cliente final, quiero consultar el detalle de un viaje, para verificar información específica como ruta, estado y temperatura.	3
4	US024	Creación de viajes	Como administrador logístico, quiero crear viajes asignando vehículo y ruta, para planificar el transporte de mercancías.	5
5	US032	Lista de viajes registrados	Como empresa, quiero ver una lista de todos los viajes registrados para gestionarlos de forma rápida.	3
6	US026	Reprogramación de viajes	Como administrador logístico, quiero reprogramar un viaje ya creado, para ajustar fechas y horarios en caso de cambios operativos.	5
7	US027	Código de viaje para cliente	Como cliente final, quiero recibir un código único de viaje, para poder consultar fácilmente el estado de mi pedido.	5
8	US020	Ver vehículos de carga	Como administrador logístico, quiero visualizar la lista de vehículos de carga registrados en la plataforma, para supervisar y gestionar el inventario.	3
9	US036	Filtrado de viajes por fecha	Como empresa, quiero filtrar la lista de viajes por rango de fechas, para analizar un periodo específico.	3
10	US037	Descarga de reporte de viajes	Como cliente final, quiero descargar un reporte en PDF de un viaje con su información y gráficos, para archivarlo o compartirlo.	5
11	US016	Registro de dispositivos IoT	Como administrador logístico, quiero registrar un dispositivo IoT en la plataforma, para asociarlo a la flota y comenzar a recibir sus datos.	5
12	US017	Eliminar dispositivo IoT	Como administrador logístico, quiero eliminar un dispositivo IoT de la plataforma, para darlo de baja en caso de falla o reemplazo.	3
13	US021	Ver dispositivos IoT	Como administrador logístico, quiero visualizar la lista de dispositivos IoT registrados en la plataforma, para supervisar su estado y administración.	3
14	US030	Alertas de conexión IoT	Como empresa, quiero recibir alertas cuando un dispositivo IoT deja de enviar datos, para actuar de inmediato.	5
15	US022	Ver estado de dispositivo por vehículo	Como administrador logístico, quiero visualizar el estado de todos los dispositivos asociados a un vehículo en el dashboard, para monitorear su funcionamiento	3
16	US023	Ver estado de dispositivo en el módulo físico	Como usuario del dispositivo, quiero que el dispositivo tenga un indicador físico de estado (ej. LED), para verificar rápidamente si funciona sin depender de la app	3
17	US034	Gráficos de tiempo y temperatura	Como cliente final, quiero ver gráficos de evolución de la temperatura durante el viaje, para verificar el cumplimiento de parámetros.	5
18	US018	Asignar dispositivo a vehículo de carga	Como administrador logístico, quiero asignar un dispositivo IoT a un vehículo de carga, para identificar a qué unidad pertenece cada transmisión de datos	3
19	US013	Registro de vehículos de carga	Como administrador logístico, quiero registrar vehículos en la plataforma, para mantener un inventario actualizado de la flota.	5
20	US014	Actualización de datos de vehículos de carga	Como administrador logístico, quiero editar la información de los vehículos, para mantener actualizado su estado operativo.	3
21	US015	Eliminación de vehículos de carga	Como administrador logístico, quiero eliminar vehículos de la plataforma, para mantener un inventario actualizado y evitar registros obsoletos.	3
22	US019	Cambiar dispositivo de vehículo	Como administrador logístico, quiero reasignar un dispositivo IoT de un vehículo a otro, para reutilizarlo en caso de mantenimiento o rotación de la flota	3
23	US025	Actualización de estados de viaje	Como administrador logístico, quiero actualizar el estado de un viaje, para mantener informados a gerentes y clientes sobre el progreso de la entrega.	5
24	US031	Roles y permisos de acceso	Como empresa, quiero que el sistema gestione roles y permisos de usuarios (admin, cliente, operador), para controlar accesos.	8
25	US035	Gráficos de incidencias por mes	Como empresa, quiero ver un gráfico mensual de incidencias para identificar patrones de fallos.	5

# Orden	User Story ID	Título	Descripción	Story Points
26	US038	Cancelar suscripción	Como cliente final, quiero cancelar mi suscripción, para detener los cobros futuros.	2
27	US039	Visualizar información de suscripción	Como cliente final, quiero ver mi estado de suscripción y fecha de expiración, para gestionar mi acceso al servicio.	2
28	US040	Historial de pagos	Como cliente final, quiero consultar mi historial de pagos, para verificar mis transacciones.	3
29	US041	Notificación de renovación próxima	Como cliente final, quiero recibir una notificación antes de que mi suscripción se renueve, para decidir si continúo o cancelo.	3
30	US010	Visualizar información de suscripción	Como cliente final, quiero ver mi estado de suscripción y fecha de expiración, para gestionar mi acceso al servicio.	2
31	US011	Historial de pagos	Como cliente final, quiero consultar mi historial de pagos, para verificar mis transacciones.	3
32	US012	Notificación de renovación próxima	Como cliente final, quiero recibir una notificación antes de que mi suscripción se renueve, para decidir si continúo o cancelo.	3
33	TS005	API de tracking de viajes	Como developer, quiero exponer un endpoint que devuelva el estado actual de un viaje, para que los clientes puedan consultar el seguimiento en tiempo real.	8
34	TS004	API de viajes	Como developer, quiero exponer un endpoint RESTful para registrar viajes, para que la aplicación guarde y gestione la información.	8
35	TS003	API de vehículos	Como developer, quiero exponer un endpoint para registrar, modificar y consultar vehículos, para que el backend gestione el inventario de la flota.	8
36	US001	Navegación en landing page	Como visitante, quiero navegar entre las secciones de la landing page, para acceder fácilmente a la información sobre el servicio.	3
37	US002	Sección portada	Como visitante, quiero ver una portada con mensaje principal, para entender rápidamente el propósito de la plataforma.	3
38	US003	Sección de funcionalidades	Como visitante, quiero visualizar una sección con las funcionalidades principales, para conocer qué ofrece la plataforma.	3
39	US004	Sección de beneficios	Como visitante, quiero ver una sección con beneficios, para entender qué valor obtengo al usar la plataforma.	2
40	US005	Sección de testimonios	Como visitante, quiero ver testimonios de otros clientes, para ganar confianza en el servicio.	2
41	US006	Sección de contáctanos	Como visitante, quiero acceder a un formulario de contacto, para comunicarme con la empresa y solicitar más información.	3
42	US007	Call to Action a la aplicación web	Como visitante, quiero encontrar un botón de acceso a la aplicación web, para registrarme o iniciar sesión y usar el servicio desde un navegador.	3
43	US008	Call to Action de descarga de App Móvil	Como visitante, quiero encontrar botones de descarga de la aplicación móvil, para instalar la app en mi dispositivo iOS o Android.	3
44	US009	Cancelar suscripción	Como cliente final, quiero cancelar mi suscripción, para detener los cobros futuros.	2
45	TS001	API de registro	Como developer quiero implementar múltiples endpoints de autenticación (login, logout, refresh y validación de sesión) para que los usuarios puedan gestionar de forma segura su acceso a la plataforma.	5
46	US018	Registro de usuario	Como usuario, quiero registrarme en la plataforma, para acceder a mi cuenta y funcionalidades personalizadas.	3
47	US010	Inicio de sesión	Como usuario registrado, quiero iniciar sesión en la plataforma, para acceder a mi cuenta y funcionalidades personalizadas.	3
48	US011	Cerrar sesión	Como usuario autenticado, quiero cerrar sesión desde la aplicación, para que mi cuenta deje de estar accesible en el dispositivo actual.	2
49	US012	Recuperar contraseña	Como usuario, quiero recuperar el acceso a mi cuenta mediante un proceso de restablecimiento de contraseña, para poder ingresar nuevamente si la olvido.	3
50	TS002	Servicio de autenticación con JWT	Como developer, quiero implementar autenticación basada en JWT, para asegurar la comunicación entre cliente y servidor.	8

<input checked="" type="checkbox"/> CS-8 Navegación en Landing Page	LANDING PAGE	TAREAS POR HACER ▾	OK
<input checked="" type="checkbox"/> CS-9 Sección portada	LANDING PAGE	FINALIZADA	DA
<input checked="" type="checkbox"/> CS-10 Sección de funcionalidades	LANDING PAGE	EN CURSO	AF
<input checked="" type="checkbox"/> CS-01 Sección de precios	LANDING PAGE	EN CURSO	IN
<input checked="" type="checkbox"/> CS-11 Sección de beneficios	LANDING PAGE	TAREAS POR HACER ▾	GP
<input checked="" type="checkbox"/> CS-12 Sección de testimonios	LANDING PAGE	TAREAS POR HACER ▾	BS
<input checked="" type="checkbox"/> CS-13 Sección de contactanos	LANDING PAGE	TAREAS POR HACER ▾	N
<input checked="" type="checkbox"/> CS-14 Call to Action a la aplicación web	LANDING PAGE	TAREAS POR HACER ▾	AG
<input checked="" type="checkbox"/> CS-15 Call to Action de descarga de App Móvil	LANDING PAGE	TAREAS POR HACER ▾	AG
<input checked="" type="checkbox"/> CS-16 Registro de usuario	AUTENTICACIÓN	TAREAS POR HACER ▾	AG
<input checked="" type="checkbox"/> CS-17 Inicio de sesión	AUTENTICACIÓN	TAREAS POR HACER ▾	AG
<input checked="" type="checkbox"/> CS-18 Cerrar sesión	AUTENTICACIÓN	TAREAS POR HACER ▾	AG
<input checked="" type="checkbox"/> CS-20 Internacionalización de contenido	LANDING PAGE	TAREAS POR HACER ▾	GP
<input checked="" type="checkbox"/> CS-21 Registro de vehículos de carga	GESTIÓN DE FLOTA	TAREAS POR HACER ▾	GP
<input checked="" type="checkbox"/> CS-23 Eliminación de vehículos de carga	GESTIÓN DE FLOTA	TAREAS POR HACER ▾	GP
<input checked="" type="checkbox"/> CS-28 Ver todos los vehículos de carga	GESTIÓN DE FLOTA	TAREAS POR HACER ▾	GP
<input checked="" type="checkbox"/> CS-32 Creación de viajes	PLANIFICACIÓN DE VI...	TAREAS POR HACER ▾	GP
<input checked="" type="checkbox"/> CS-33 Actualización de estados de viaje	PLANIFICACIÓN DE VI...	TAREAS POR HACER ▾	GP
<input checked="" type="checkbox"/> CS-38 Lista de viajes registrados	DASHBOARD DE VIAJES	TAREAS POR HACER ▾	GP
<input checked="" type="checkbox"/> CS-34 Reprogramación de viajes	PLANIFICACIÓN DE VI...	TAREAS POR HACER ▾	GP
<input checked="" type="checkbox"/> CS-22 Actualización de datos de vehículos de carga	GESTIÓN DE FLOTA	TAREAS POR HACER ▾	GP
<input checked="" type="checkbox"/> CS-46 Visualizar información de suscripción actual	MÓDULO DE SUSCRIP...	TAREAS POR HACER ▾	GP
<input checked="" type="checkbox"/> CS-19 Recuperar contraseña	AUTENTICACIÓN	TAREAS POR HACER ▾	GP
<input type="checkbox"/> CS-24 Registro de dispositivos IoT	GESTIÓN DE FLOTA	TAREAS POR HACER ▾	GP
<input checked="" type="checkbox"/> CS-25 Eliminar dispositivo IoT	GESTIÓN DE FLOTA	TAREAS POR HACER ▾	GP
<input checked="" type="checkbox"/> CS-26 Asignar dispositivo a vehículo de carga	GESTIÓN DE FLOTA	TAREAS POR HACER ▾	GP
<input checked="" type="checkbox"/> CS-27 Cambiar dispositivo de vehículo	GESTIÓN DE FLOTA	TAREAS POR HACER ▾	GP
<input checked="" type="checkbox"/> CS-29 Ver todos los dispositivos IoT en la plataforma	GESTIÓN DE FLOTA	TAREAS POR HACER ▾	GP
<input checked="" type="checkbox"/> CS-30 Ver estado de dispositivo por vehículo	GESTIÓN DE FLOTA	TAREAS POR HACER ▾	GP
<input checked="" type="checkbox"/> CS-31 Ver estado de dispositivo en el módulo físico	GESTIÓN DE FLOTA	TAREAS POR HACER ▾	GP
<input checked="" type="checkbox"/> CS-35 Actualización de temperatura en tiempo real	MONITOREO EN TIEM...	TAREAS POR HACER ▾	GP
<input checked="" type="checkbox"/> CS-36 Alertas por incumplimiento de temperatura	MONITOREO EN TIEM...	TAREAS POR HACER ▾	GP
<input checked="" type="checkbox"/> CS-37 Alertas de conexión IoT	MONITOREO EN TIEM...	TAREAS POR HACER ▾	GP
<input checked="" type="checkbox"/> CS-40 Gráficos de tiempo y temperatura	DASHBOARD DE VIAJES	TAREAS POR HACER ▾	GP
<input checked="" type="checkbox"/> CS-41 Gráficos de incidencias por mes	DASHBOARD DE VIAJES	TAREAS POR HACER ▾	GP
<input checked="" type="checkbox"/> CS-43 Descarga de reporte de viajes	DASHBOARD DE VIAJES	TAREAS POR HACER ▾	GP
<input checked="" type="checkbox"/> CS-48 Notificación de proxima renovación	MÓDULO DE SUSCRIP...	TAREAS POR HACER ▾	GP

Link del Jira: [Click aquí para ver el jira](#)

Capítulo IV: Solution Software Design

4.1. Strategic-Level Domain-Driven Design

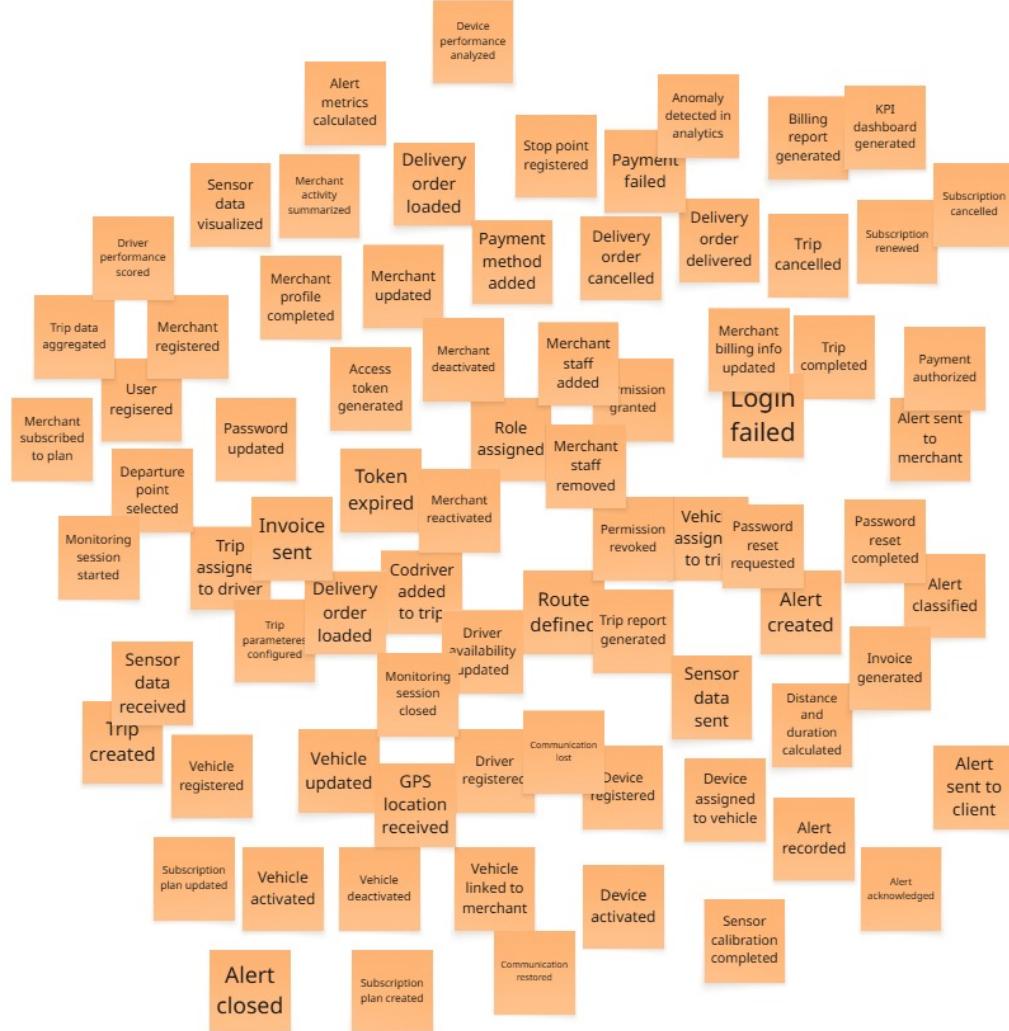
4.1.1. Design-Level EventStorming

En esta sección se realizó un Event Storming detallado para modelar y analizar el dominio del sistema CargaSafe. Se llevó a cabo en varias sesiones colaborativas donde se identificaron eventos, comandos, agregados y políticas clave del negocio. Este ejercicio permitió descubrir los bounded contexts candidatos y mapear los flujos de mensajes entre ellos, sentando las bases para el diseño de la arquitectura del sistema.

Step 1: Unstructured Exploration

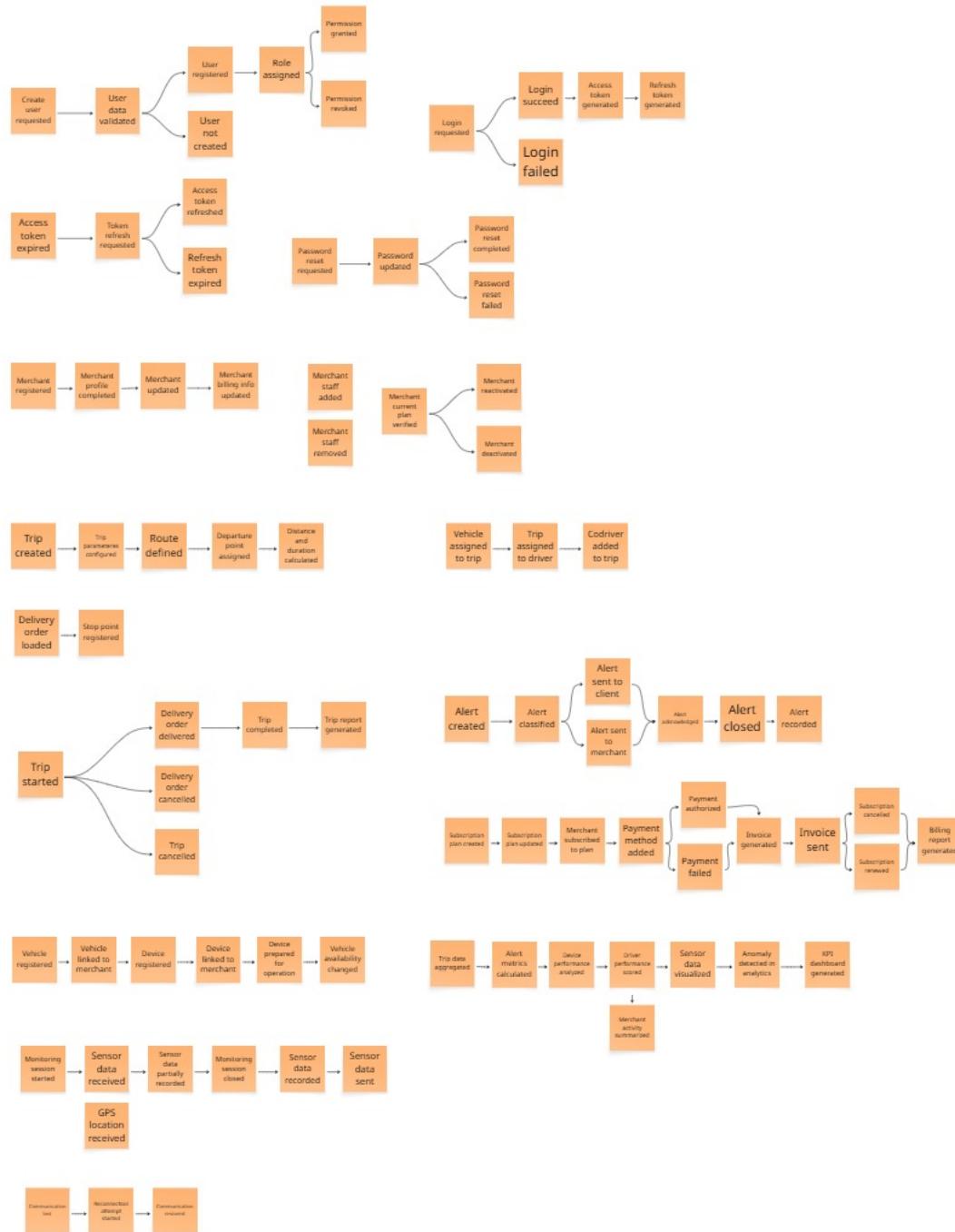
Se identificaron todos los eventos clave del sistema mediante una lluvia de ideas sin orden definido, abarcando registro, monitoreo, alertas, viajes y pagos.

STEP 1: Unstructured Exploration

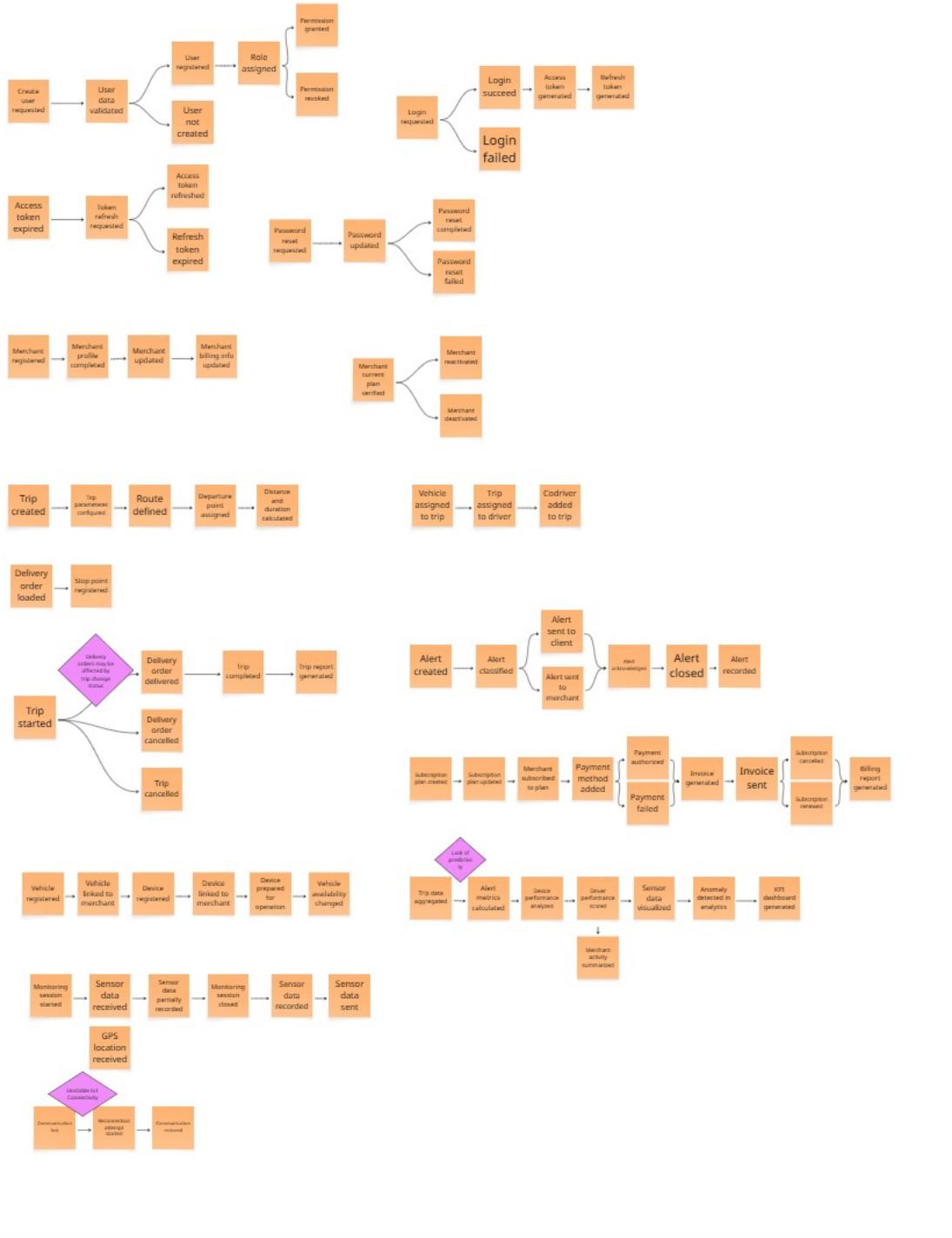


Step 2: Timeline

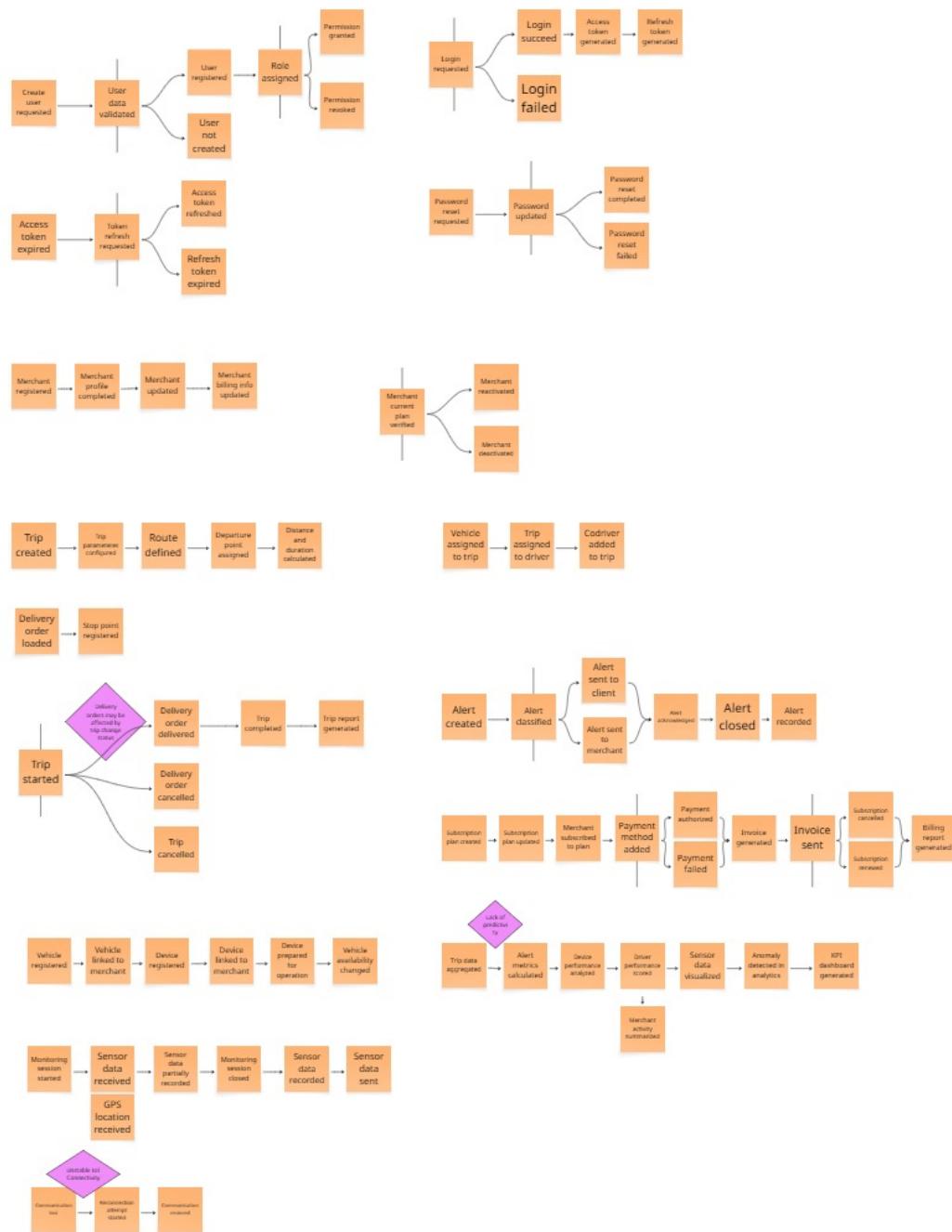
Los eventos fueron organizados cronológicamente para representar los principales flujos de negocio y dependencias entre procesos.

STEP 2: Timeline**Step 3: Pain Points**

Se identificaron los puntos críticos y posibles fallas dentro de los flujos, como errores de comunicación o validación.

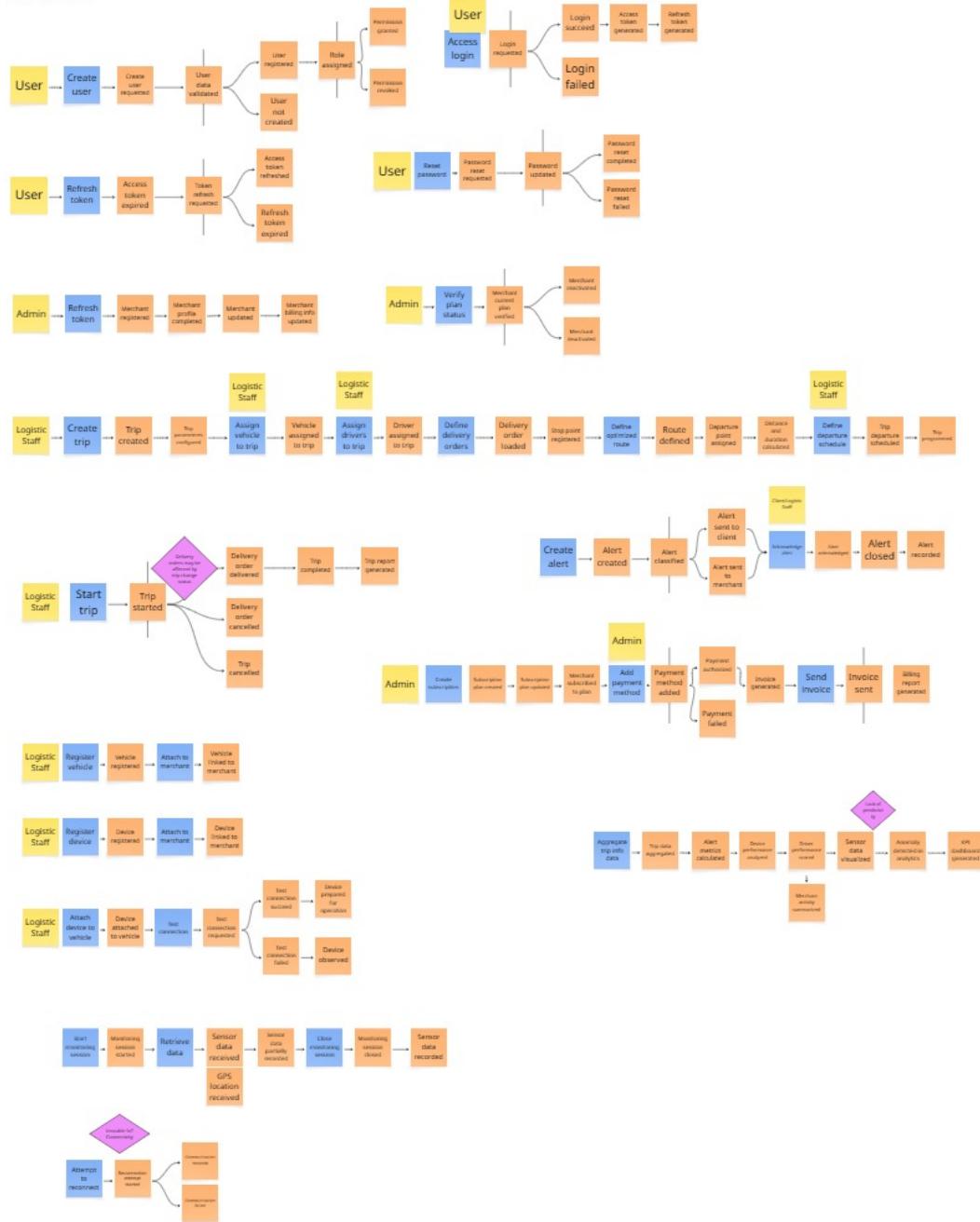
STEP 3: PaintPoints**Step 4: Pivotal Points**

Se marcaron los eventos que generan un cambio de contexto significativo dentro del sistema, como inicios o cierres de procesos.

STEP 4: Pivotal**Step 5: Commands**

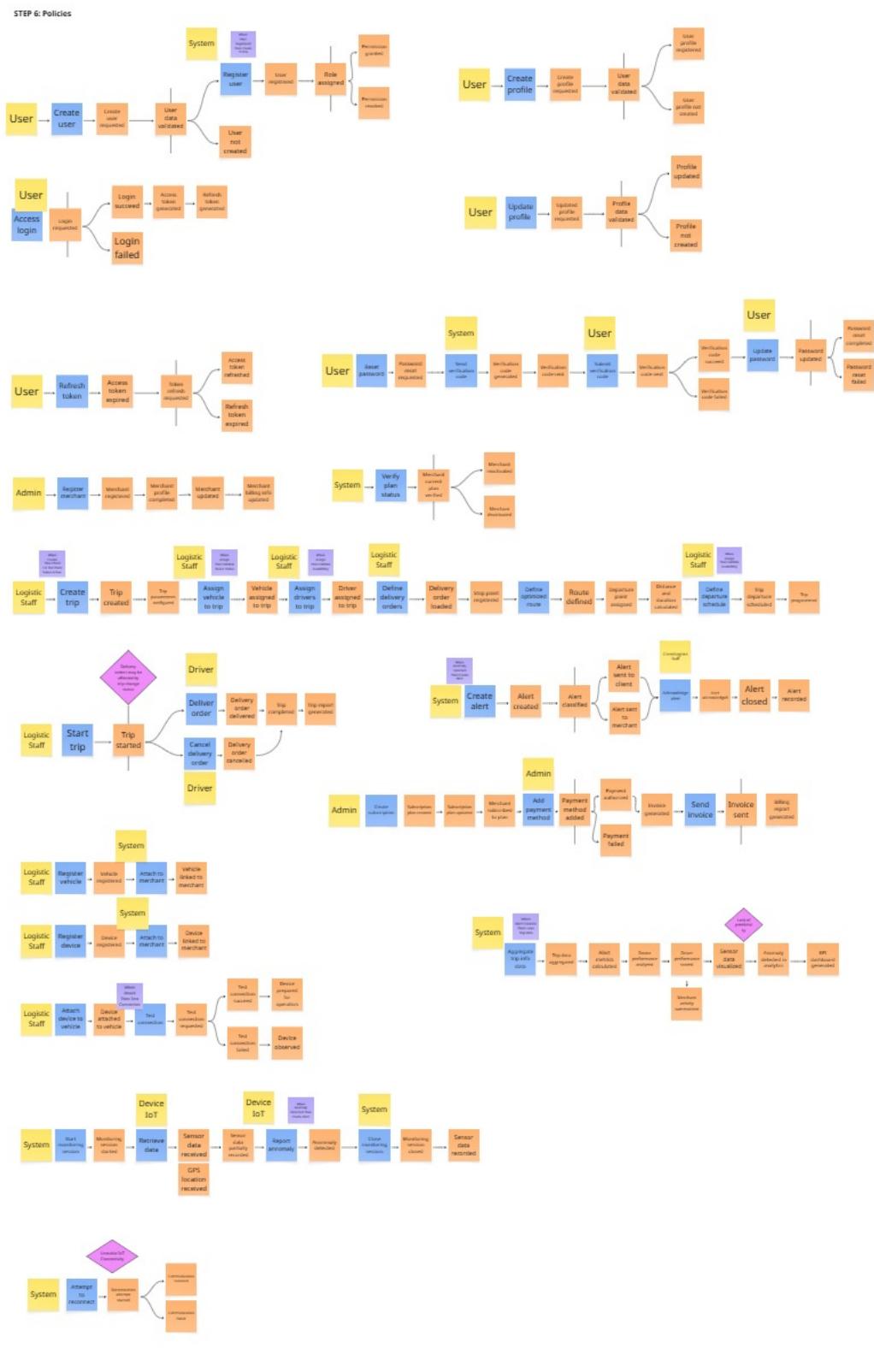
Se definieron los comandos que ejecutan los actores del sistema y que originan los eventos, representando las intenciones de acción.

STEP 5: Commands



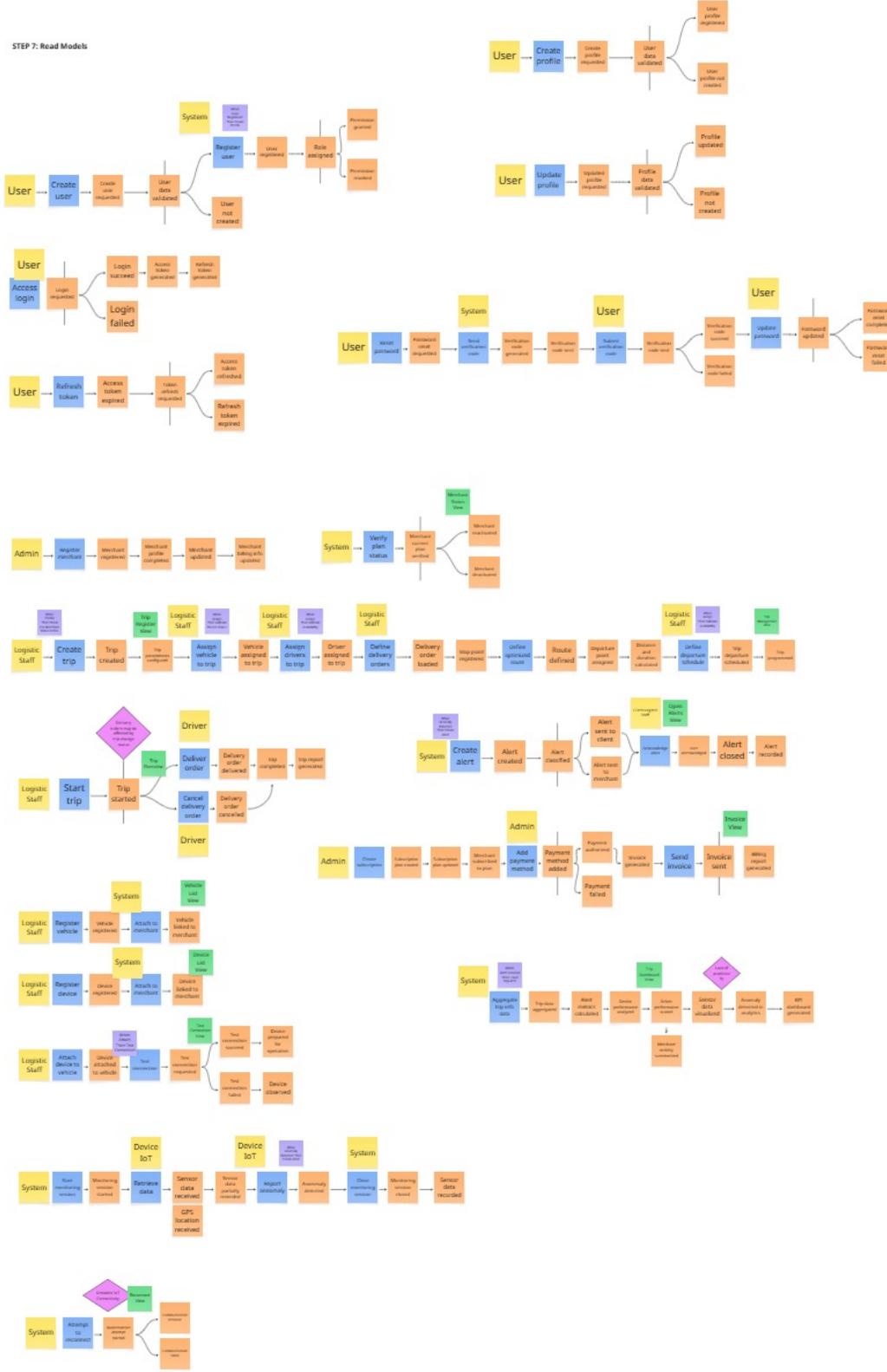
Step 6: Policies

Se establecieron reglas automáticas que vinculan eventos con nuevos comandos, automatizando respuestas del sistema.



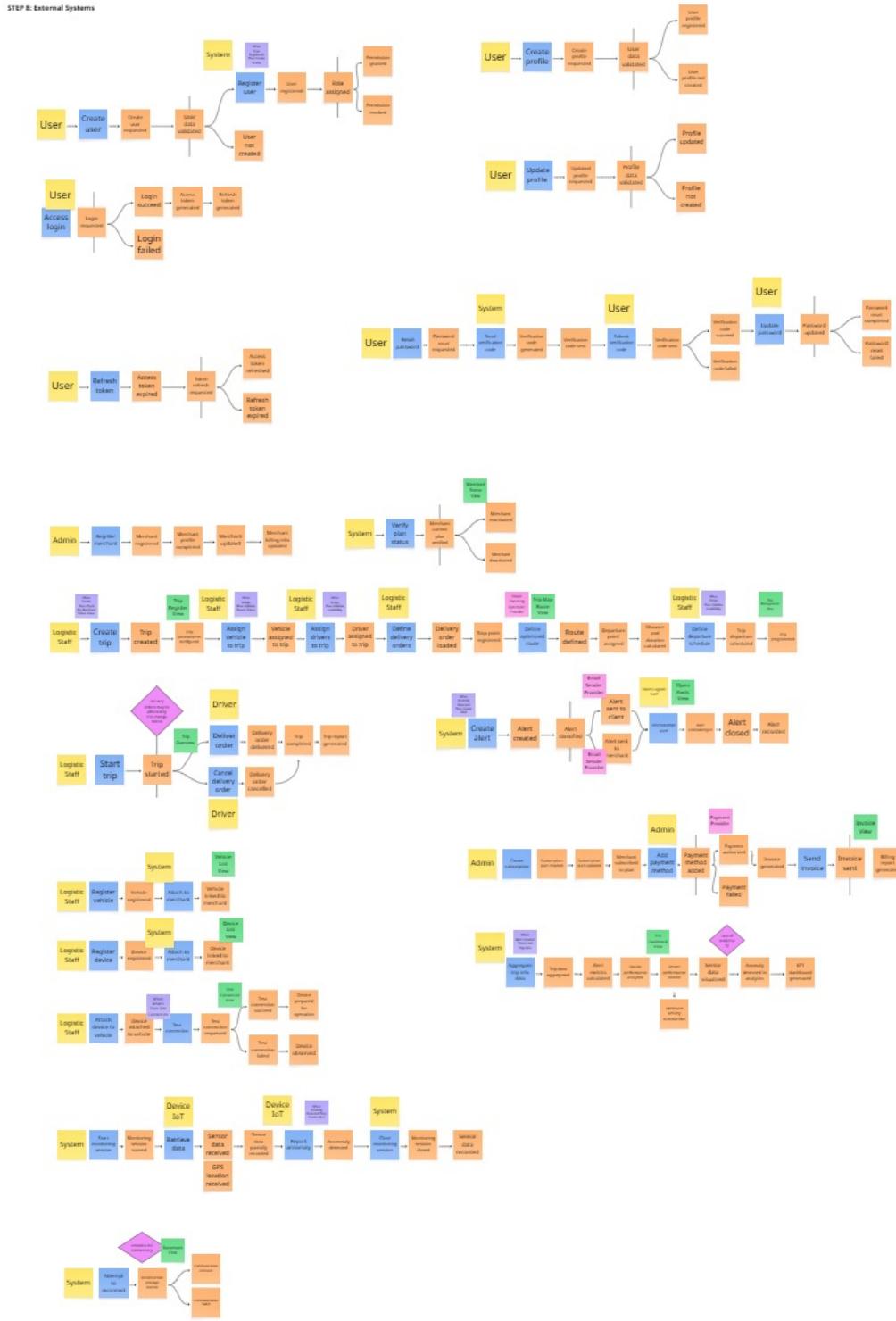
Step 7: Read Models

Se definieron vistas de datos que reflejan el estado actual del sistema para consulta o monitoreo en tiempo real.



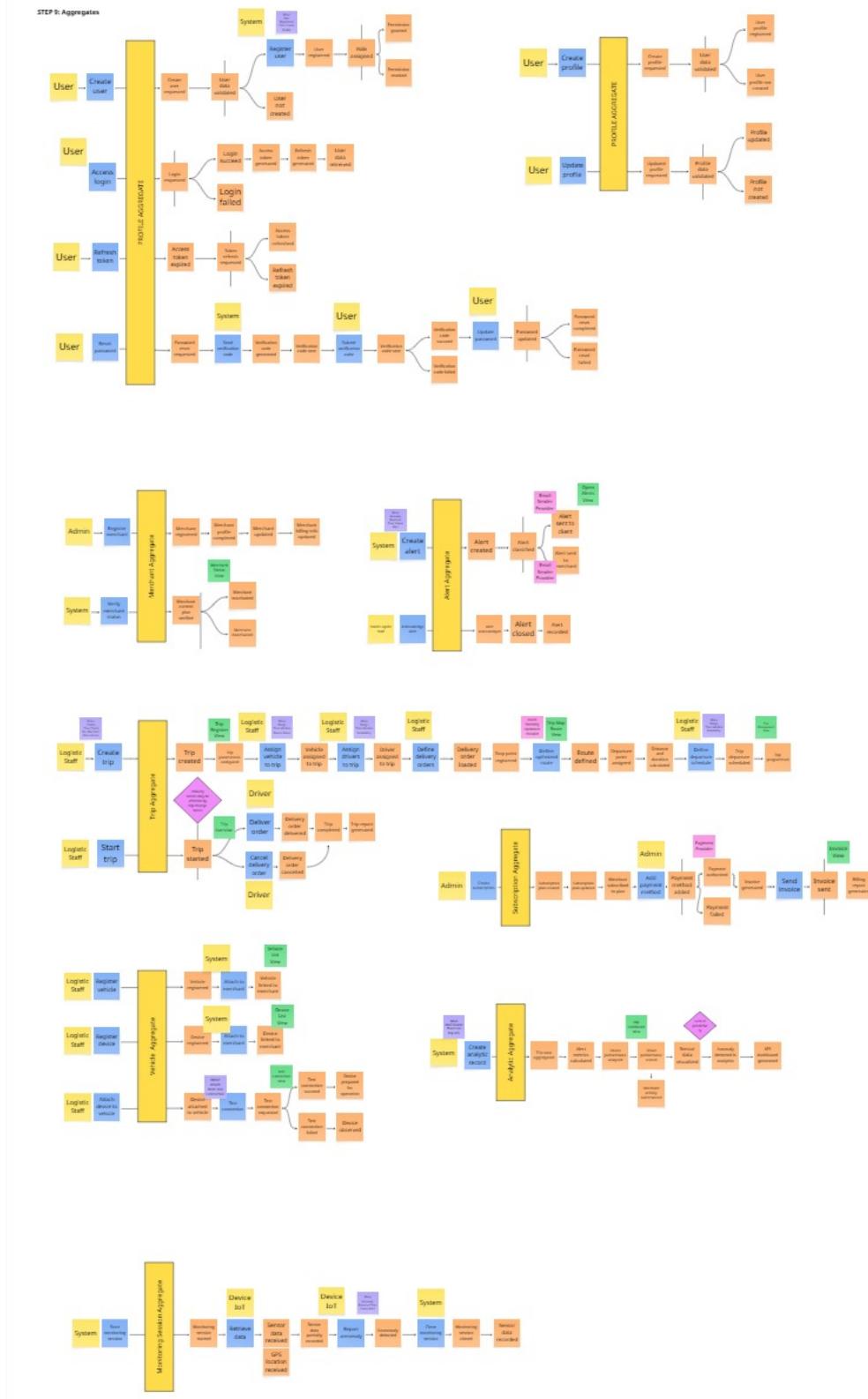
Step 8: External Systems

Se identificaron los servicios externos integrados, como APIs, pasarelas de pago y plataformas de notificación.



Step 9: Aggregates

Se agruparon comandos y eventos bajo agregados que aseguran la coherencia en cada bounded context.



4.1.1.1 Candidate Context Discovery

Para esta etapa se llevó a cabo una sesión, la sesión tuvo una duración aproximada de 90 minutos y permitió identificar los bounded contexts del sistema CargaSafe. Durante el proceso se aplicaron las técnicas start-with-value, start-with-simple y look-for-pivotal-events, que facilitaron la agrupación de eventos y entidades según su afinidad y valor para el negocio.

Como resultado, se identificaron ocho bounded contexts:

- **Identity and Access Management**: administración de usuarios, autenticación y control de accesos.
- **Profiles and Preferences Management**: gestión de perfiles de usuario y configuración de preferencias.
- **Fleet management**: gestión de vehículos y dispositivos IoT.
- **Trip management**: creación y ejecución de viajes.
- **Real-time monitoring**: monitoreo de condiciones en tiempo real.
- **Alerts and resolution**: generación de alertas.
- **Visualization/Analytics**: visualización de métricas y reportes.

- **Subscriptions and payments:** gestión de suscripciones y pagos con Stripe.



Leyenda utilizada en el EventStorming

- **Event**: describe algo que ocurrió en el dominio (Viaje iniciado, Alerta generada).
- **Command**: una instrucción o acción que dispara un evento (Registrar viaje).
- **Policy**: regla de negocio que determina qué ocurre ante ciertas condiciones (Si falta dispositivo → bloquear inicio del viaje).
- **Aggregate**: entidad principal que concentra datos y operaciones (Viaje, Suscripción).
- **UI**: vistas o pantallas del sistema que muestran información al usuario (Dashboard de KPIs).
- **Actor**: roles que interactúan con el sistema (Operador, Conductor).
- **Sistema externo**: integraciones con servicios de terceros (Google Maps, Stripe).

Con esta estructura, el EventStorming permitió organizar y simplificar el dominio de CargaSafe, evidenciando de forma clara los contextos candidatos y la interacción entre actores, procesos y sistemas externos.

[Ver gráfico en Miro](#)

4.1.1.2. Domain Message Flows Modeling

En esta etapa se desarrolló el **modelado de flujos de mensajes de dominio (Domain Message Flows)** con el objetivo de visualizar cómo colaboran los bounded contexts identificados en el Candidate Context Discovery para resolver los principales casos de negocio del sistema CargaSafe.

Para la construcción de estos flujos se aplicó la técnica de **Domain Storytelling**, la cual permite describir las interacciones en un lenguaje natural, mostrando cómo un evento generado en un bounded context desencadena comandos o nuevos eventos en otros contextos. De este modo se logra una visión clara de la cooperación entre módulos y del ciclo de vida de la información dentro de la plataforma.

Historias de dominio (Domain Stories)

1. Gestión de identidad y perfiles

- Cuando un **usuario se registra** en **Identity and Access Management**, se genera un evento que es consumido por **Profiles and Preferences**, el cual crea automáticamente el perfil asociado.
- Si un **usuario edita sus preferencias**, se guarda la configuración en **Profiles**, y en caso de referirse a notificaciones, estas se utilizan en **Alerts** para personalizar los canales de envío.

2. Control de acceso y suscripciones

- Cuando un **pago es procesado exitosamente** en **Subscriptions & Billing**, se envía un evento a **Identity and Access Management**, que habilita el acceso al sistema.
- Si un **pago falla**, el mismo flujo comunica a IAM que debe restringir o bloquear el acceso del usuario hasta regularizar su situación.

3. Gestión de flota y ejecución de viajes

- Al **registrarse un vehículo o dispositivo IoT** en **Fleet Management**, este queda disponible para **Trip Management**, que puede asignarlo a un viaje planificado.
- Cuando un **operador crea e inicia un viaje** en **Trip Management**, se emite un evento que da origen a una sesión de monitoreo en **Monitoring**.

4. Monitoreo en tiempo real y alertas

- **Monitoring** recibe continuamente *lecturas de sensores* (temperatura, ubicación, señal). Si se detecta una condición fuera de rango, se genera un evento que es consumido por **Alerts**.
- **Alerts** crea la alerta correspondiente y la notifica a los usuarios, aplicando las preferencias definidas en **Profiles** (por ejemplo, envío por SMS, correo o notificación push).

5. Analítica y reportes

- Cada alerta generada o reconocida en **Alerts** actualiza los indicadores en **Dashboard & Analytics**, alimentando las métricas de cumplimiento y los reportes de incidentes.
- Cuando **Dashboard & Analytics** genera un *reporte final*, este puede personalizarse de acuerdo con las preferencias almacenadas en **Profiles**, permitiendo al usuario recibir información ajustada a su rol o necesidades.



Resultados

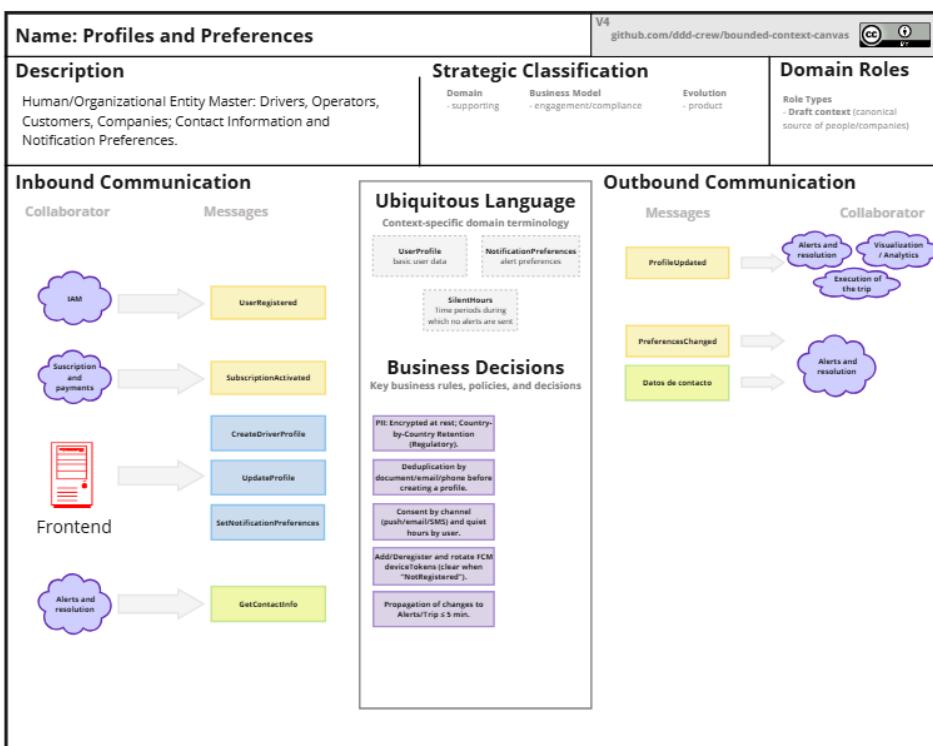
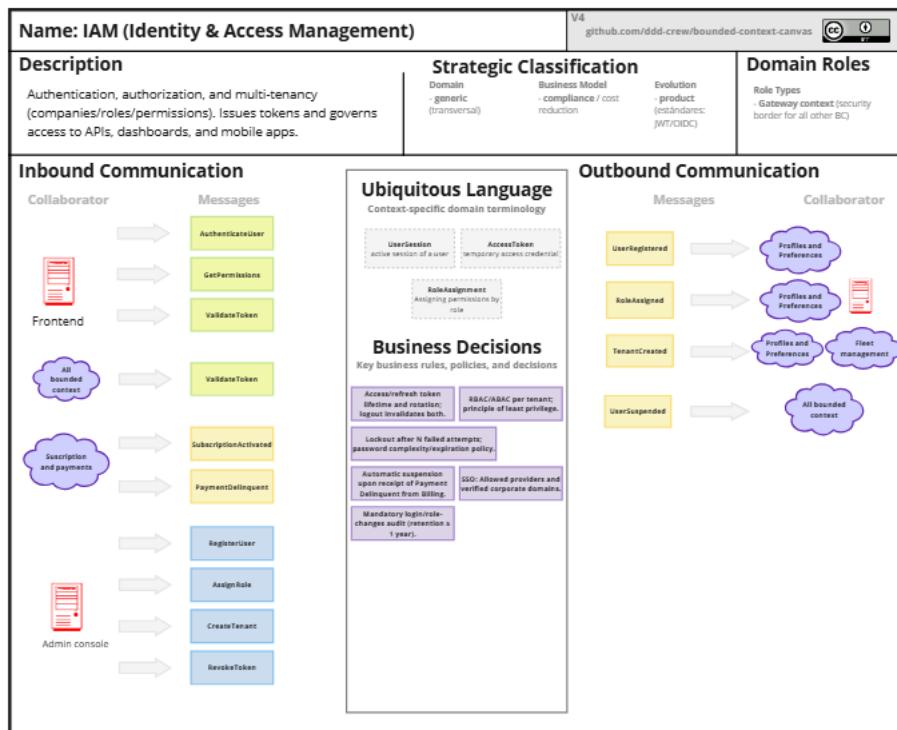
Los flujos de mensajes de dominio evidencian la cooperación entre los ocho bounded contexts de CargaSafe:

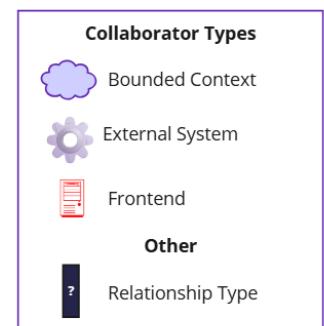
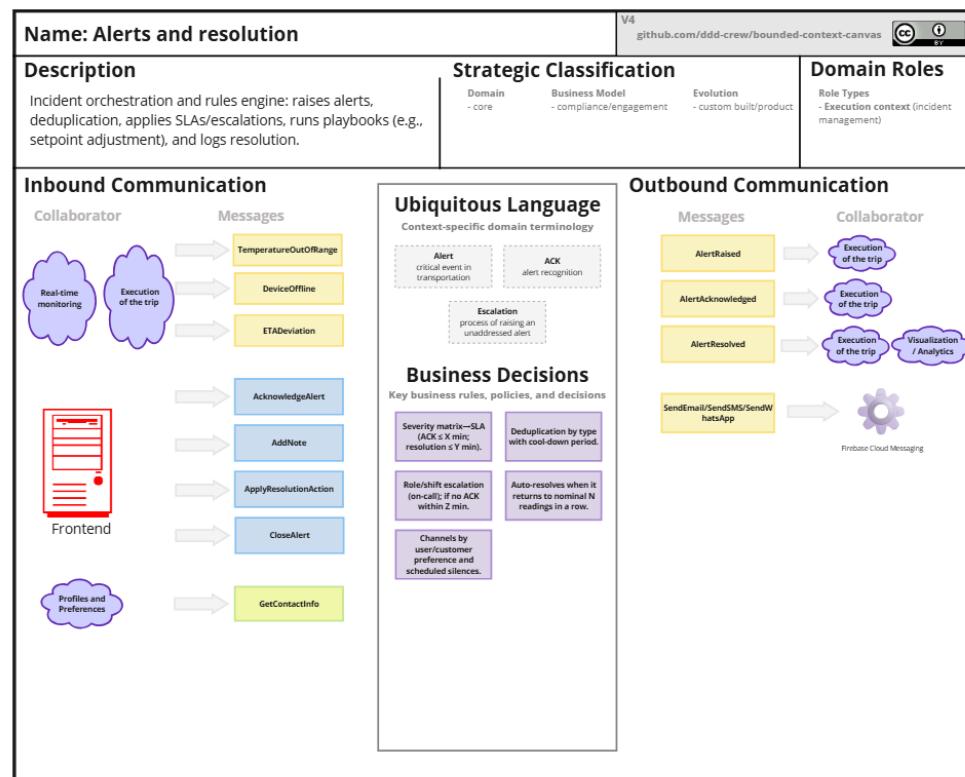
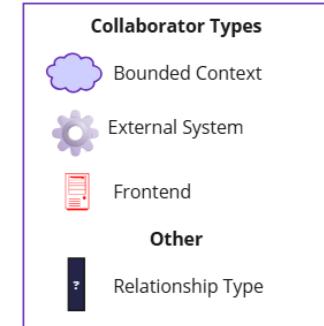
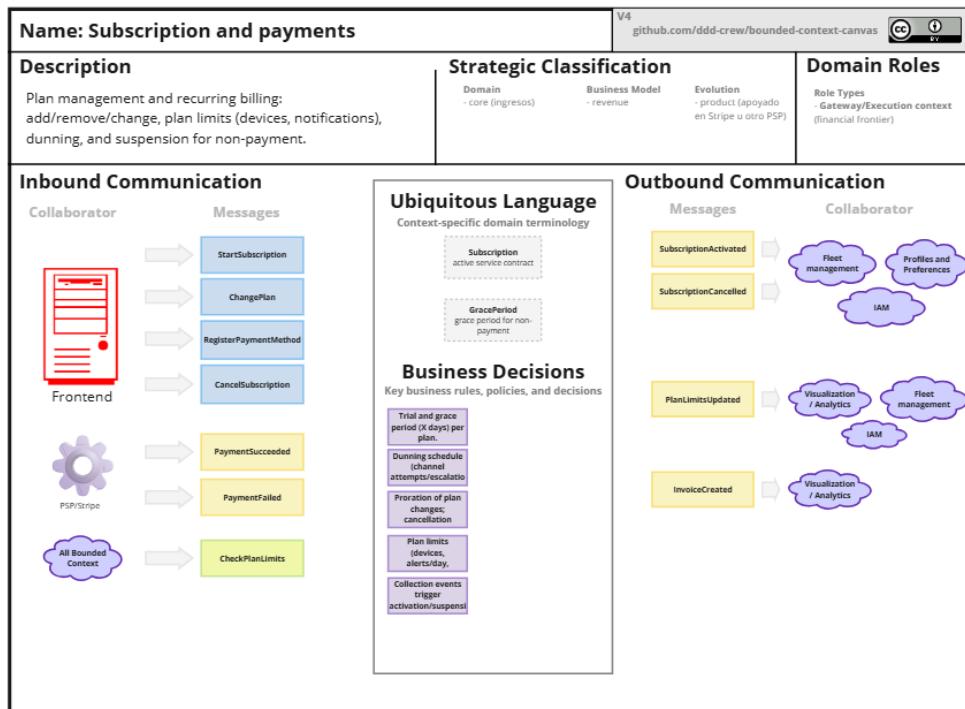
- **Identity and Access Management**
- **Profiles and Preferences Management**
- **Fleet Management**
- **Trip management**
- **Real-time monitoring**
- **Alerts and resolution**
- **Visualization/Analytics**
- **Subscriptions and payments**

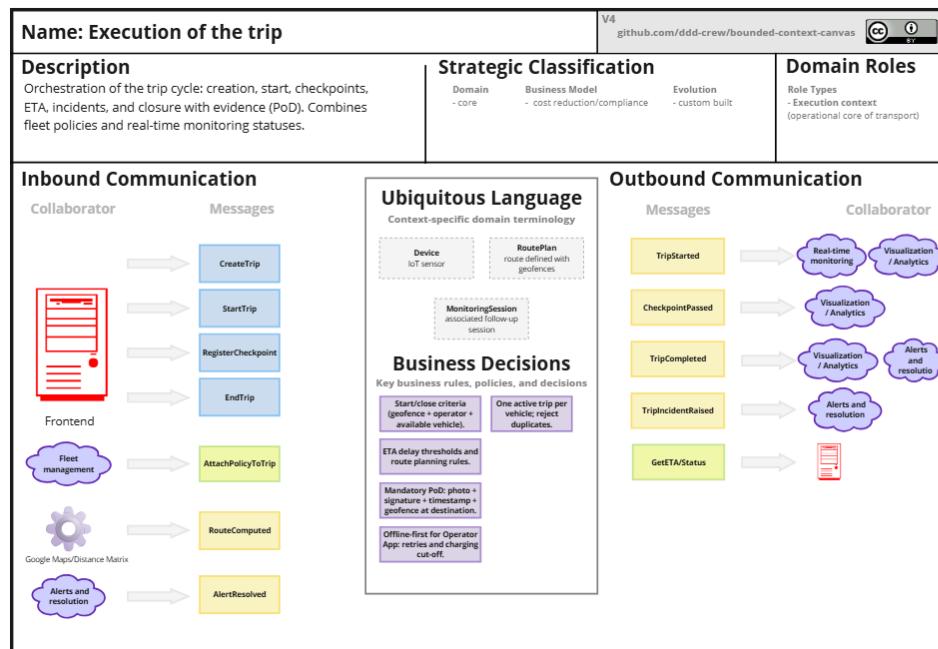
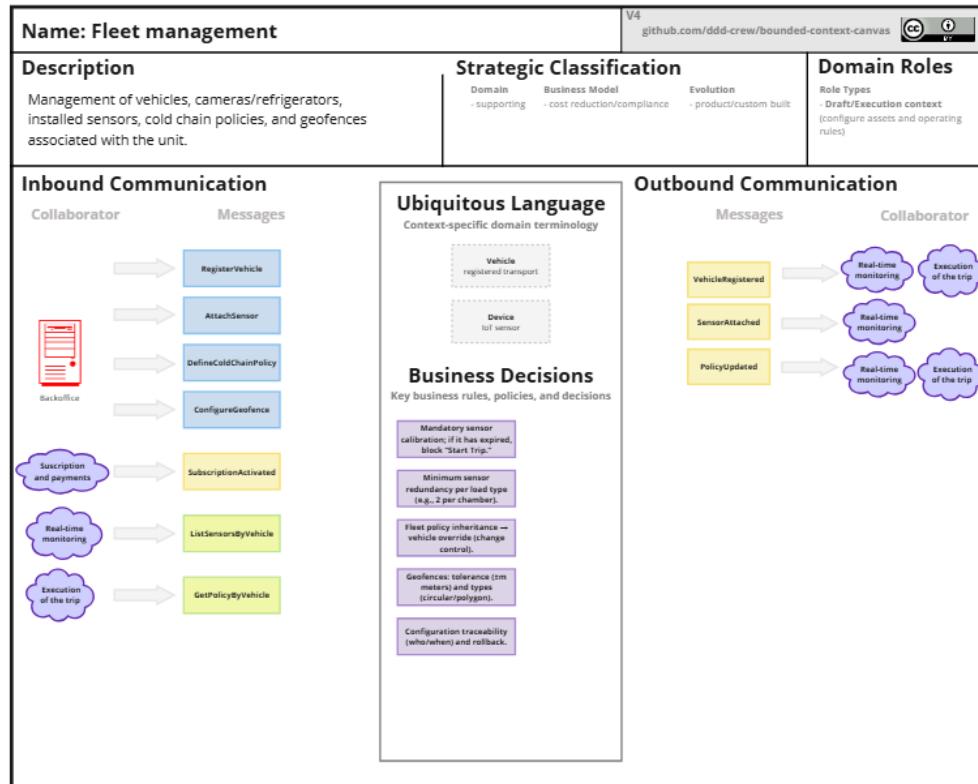
Este ejercicio permitió comprender cómo un evento local en un contexto puede impactar en otros, asegurando la trazabilidad del negocio y la correcta interacción entre los distintos módulos de la solución.

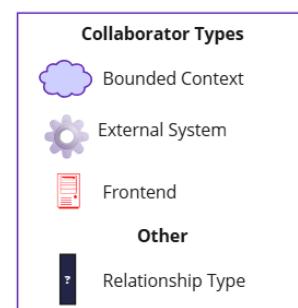
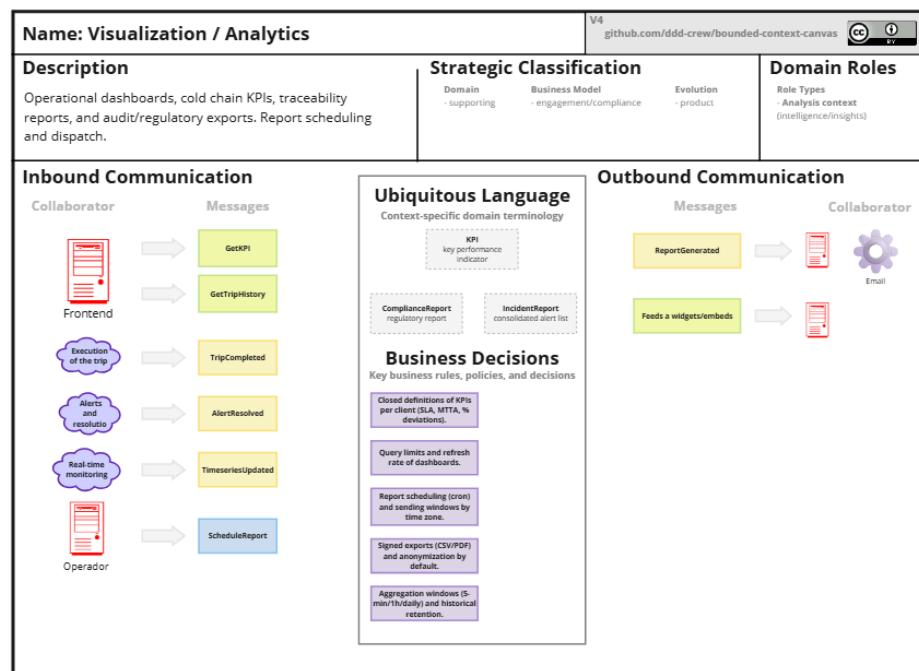
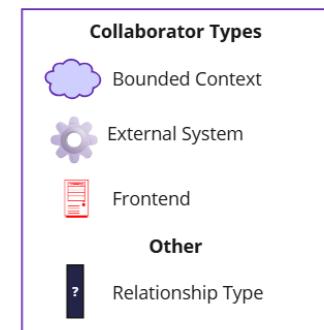
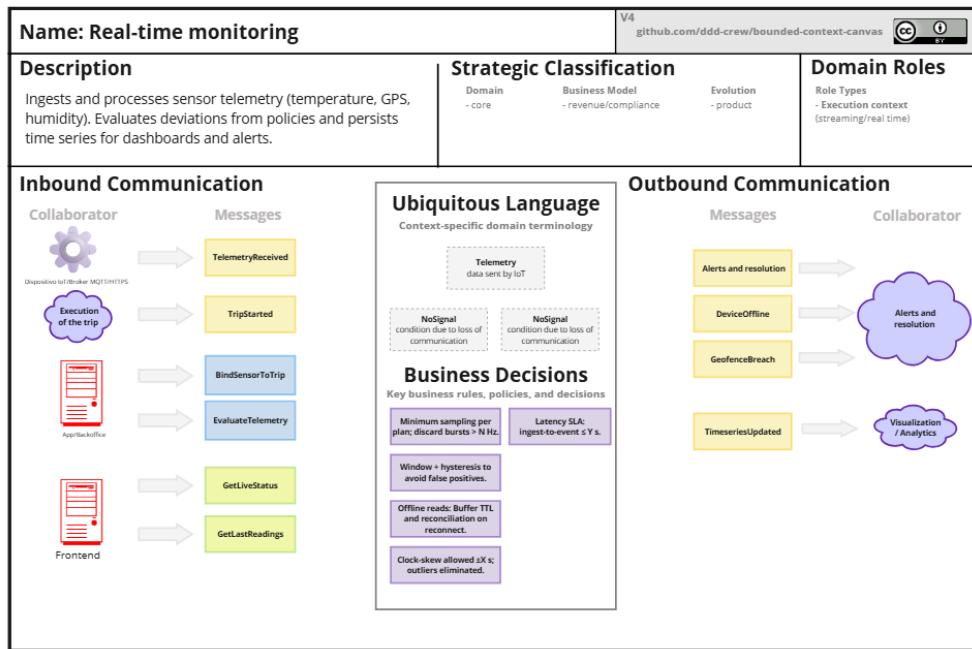
4.1.1.3. Bounded Context Canvases

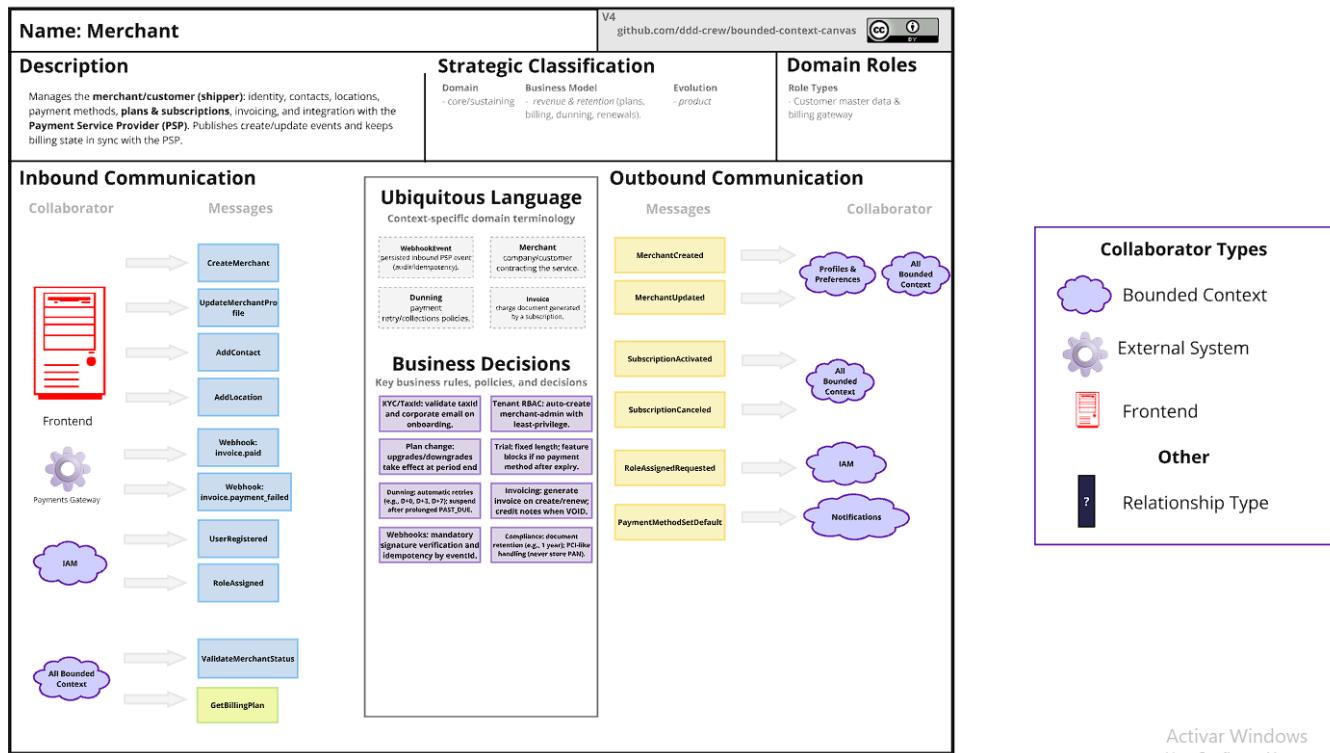
En esta sección se elaboraron los Bounded Context Canvases de CargaSafe para los ocho contextos identificados. El objetivo fue delimitar con precisión responsabilidades, lenguaje ubicuo y decisiones de negocio, además de explicitar las comunicaciones (Queries, Commands y Events) y colaboradores (otros BC, sistemas externos y frontend). Cada canvas documenta: Descripción, Clasificación estratégica (core/supporting/generic), Rol de dominio (draft/execution/analysis/gateway), Inbound/Outbound communication, Ubiquitous Language, Business Decisions y Collaborators. Esta definición fija ownership de datos, reduce ambigüedades y prepara los contratos de integración que se implementarán en APIs y mensajería.











Activar Windows
Ve a Configuración para activar.

[Ver gráfico en Miro](#)

4.1.2. Context Mapping

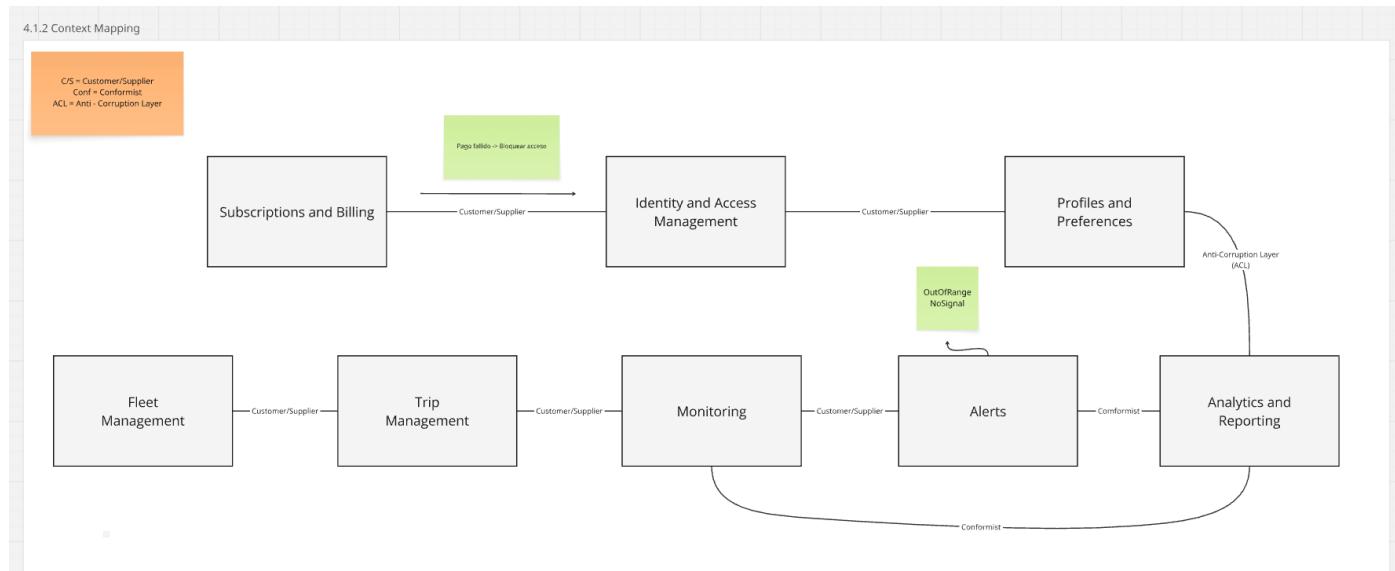
En esta etapa se construyó el **Context Map** de CargaSafe con los ocho bounded contexts identificados. El objetivo fue representar las **relaciones estructurales** entre ellos aplicando patrones de Domain-Driven Design como Customer/Supplier, Conformist y Anti-Corruption Layer (ACL).

Resultado

El mapa final permitió:

1. **Visualizar las dependencias entre contextos**, mostrando qué módulos proveen información y cuáles la consumen.
2. **Identificar los contextos core** (Trip Management, Monitoring, Alerts), los de soporte (Fleet, Profiles, Analytics) y los genéricos (IAM, Billing).
3. **Clasificar las relaciones**:
 - Customer/Supplier en la mayoría de flujos operativos (Billing → IAM, Trip → Monitoring, Monitoring → Alerts).
 - Conformist en el consumo de datos por Analytics.
 - Anti-Corruption Layer en la interacción Analytics → Profiles.

De esta manera, el Context Mapping consolida una visión global del sistema, mostrando cómo los distintos contextos colaboran para dar soporte al negocio.



4.1.3. Software Architecture

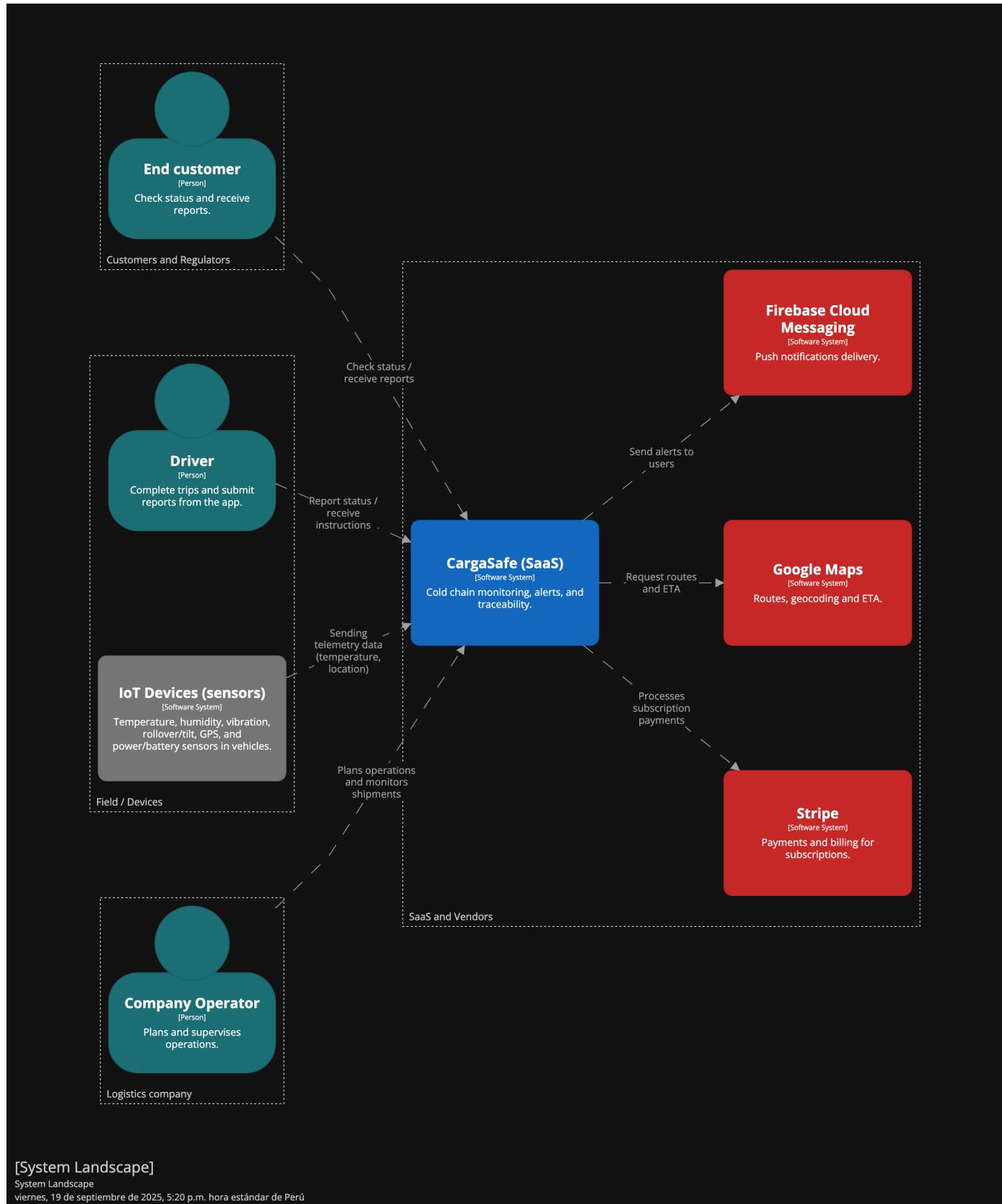
4.1.3.1. Software Architecture System Landscape Diagram

El **System Landscape Diagram** ofrece una visión de alto nivel del **ecosistema empresarial** en el que se integra CargaSafe. Este diagrama no se centra únicamente en un sistema, sino que representa **todas las personas y sistemas de software relevantes**, tanto internos como externos, que participan en la operación logística.

Propósito

El objetivo de este diagrama es:

1. Mostrar el alcance de la organización y cómo conviven sus distintos sistemas.
2. Identificar a las **personas, sistemas internos, SaaS externos y proveedores** que colaboran en la cadena de valor.
3. Resaltar cómo **CargaSafe (SaaS)** se conecta dentro de este panorama, en interacción con otros actores y servicios.



- **Personas:** Company Operator, Driver and End Customer.
- **Sistemas internos:** Logistics Planning and Power BI Data.
- **Sistemas y proveedores externos:** CargaSafe (SaaS), Stripe, Google Maps, Notification Services e IoT Devices (sensors).
- **Grupos:** Se organizaron en cuatro dominios principales:
 - Logistics company
 - Field / Devices
 - Customers and Regulators
 - SaaS and Vendors

Relaciones principales

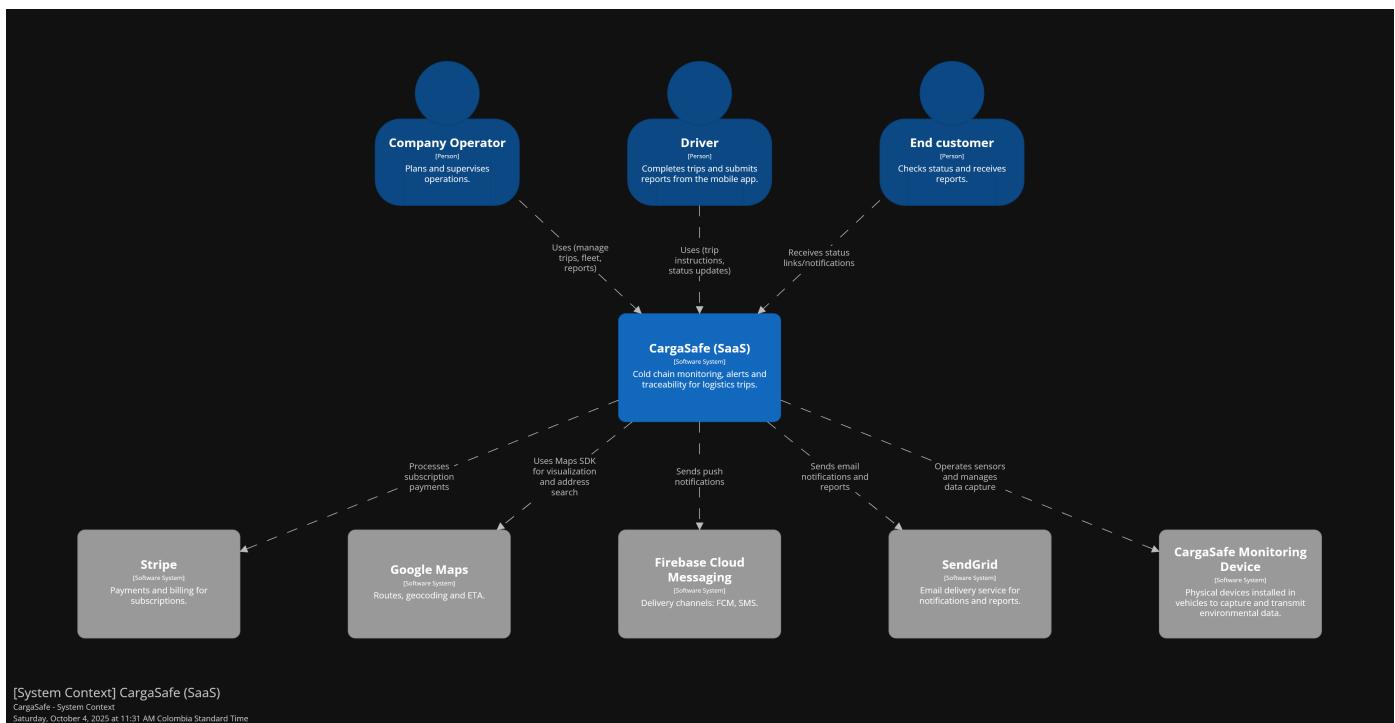
- Logistics Planning → CargaSafe (SaaS): exporta planes y asignaciones de viaje.
- IoT Devices → CargaSafe (SaaS): envía telemetría (temperatura, humedad, vibración, volcado/inclinación, GPS, energía/baterías).
- CargaSafe (SaaS) → Google Maps: consulta rutas y tiempos estimados.
- CargaSafe (SaaS) → Notification Services: envía alertas a los usuarios.
- CargaSafe (SaaS) → Stripe: procesa pagos de suscripción.
- CargaSafe (SaaS) → Power BI Data: exporta datasets consolidados para analítica.
- Company Operator / Driver ↔ CargaSafe (SaaS): planifican, ejecutan y reportan el estado operativo.
- End customer ← CargaSafe (SaaS): consulta estado y recibe reportes.

Resultado

El diagrama muestra a CargaSafe (SaaS) como el núcleo de integración entre operaciones (Company Operator, Driver, Logistics Planning), telemetría IoT (sensores en campo) y servicios externos (ruteo, notificaciones y pagos), además de su aporte a la inteligencia de negocio mediante Power BI Data. Esta representación proporciona una visión clara e integral de las dependencias y colaboraciones que sustentan la operación logística y la gestión de la cadena de frío.

4.1.3.2. Software Architecture Context Level Diagrams

El **Context Diagram** de CargaSafe muestra una visión de alto nivel del sistema y de cómo se relaciona con los actores humanos y los sistemas externos que lo rodean.



En el centro se ubica CargaSafe (SaaS), que representa el sistema principal encargado del monitoreo de la cadena de frío, la trazabilidad y la generación de alertas en los viajes logísticos.

Alrededor del sistema se identifican los siguientes actores:

- **Company Operator:** gestiona viajes, flota y reportes desde la plataforma.
- **Driver:** completa viajes y reporta información desde la aplicación móvil.
- **End customer:** recibe enlaces de estado, alertas y reportes generados por el sistema.

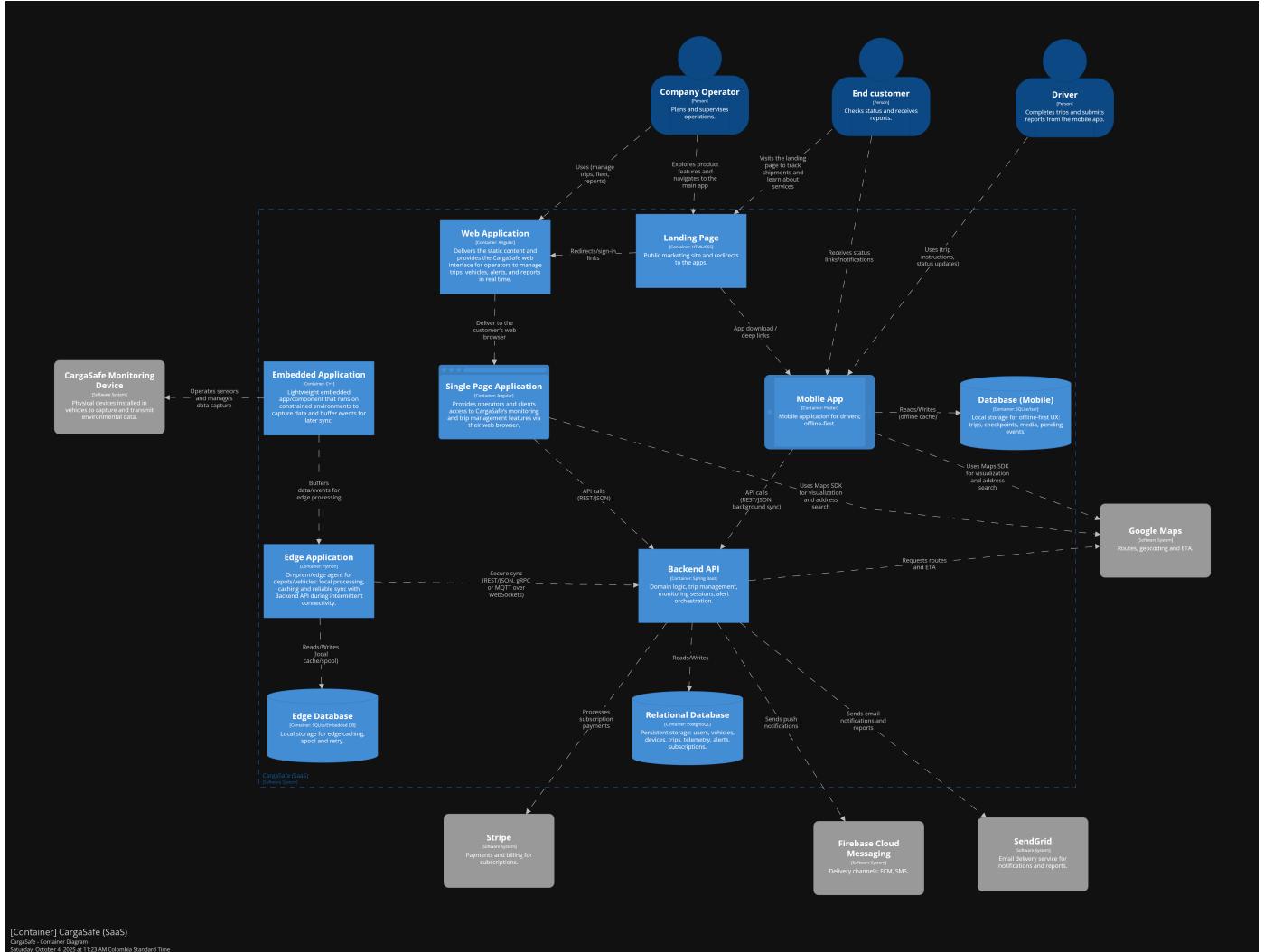
Asimismo, se destacan las interacciones con sistemas externos que complementan las funcionalidades de CargaSafe:

- **Google Maps:** provee servicios de rutas, geocodificación y cálculo de ETA, utilizados tanto por el backend para estimar tiempos y trayectos, como por las aplicaciones web y móviles para la visualización de mapas y direcciones.
- **Firebase Cloud Messaging (FCM):** entrega notificaciones push a dispositivos móviles y navegadores web para alertas, actualizaciones de viajes y eventos críticos.
- **Stripe:** procesa pagos y gestiona la facturación de las suscripciones dentro de la plataforma.
- **SendGrid:** se utiliza para el envío de correos electrónicos transaccionales, reportes automáticos y notificaciones.

- **CargaSafe Monitoring Device:** dispositivos físicos instalados en vehículos que capturan datos ambientales (temperatura, humedad, vibración, GPS) y los transmiten hacia el ecosistema CargaSafe mediante las aplicaciones embebidas y edge.

4.1.3.2. Software Architecture Container Level Diagrams

En esta parte expandimos el sistema **CargaSafe (SaaS)** para mostrar sus contenedores internos, las tecnologías que utilizamos y cómo se comunican entre sí y con los sistemas externos.



El diagrama de contenedores muestra cómo se organiza internamente CargaSafe (SaaS) y cómo se relaciona con los actores y sistemas externos.

Dentro de la plataforma tenemos varios contenedores:

- **Landing Page**
Sitio público de marketing y punto de acceso principal. Redirige a la aplicación web o descarga de la aplicación móvil.
- **Web Application**
Portal utilizado por los operadores para gestionar flota, viajes, alertas y reportes. Entrega contenido estático y sirve la **Single Page Application**.
- **Single Page Application**
Interfaz dinámica en el navegador para operadores y clientes, que ofrece monitoreo en tiempo real, seguimiento de viajes y visualización de datos utilizando el SDK de Google Maps.
- **Mobile App**
Aplicación móvil usada por los conductores para recibir instrucciones, actualizar estados y registrar reportes.
Soporta modo **offline-first** gracias a una **base de datos embebida (SQLite)**, permitiendo continuar operaciones sin conexión.
- **Mobile Database**
Base de datos local embebida utilizada por la **Mobile App** para cache y operación sin conexión (órdenes de viaje, eventos, checkpoints, adjuntos), con colas de sincronización y resolución de conflictos al reconectar.
- **Backend API**
Núcleo de la lógica de negocio del sistema.
Gestiona usuarios, dispositivos, viajes, alertas, notificaciones, suscripciones y datos de telemetría.
Exponde endpoints REST/GraphQL y se comunica con servicios externos como **Google Maps**, **Stripe**, **Firebase Cloud Messaging** y **SendGrid**.

- **Relational Database**

Base de datos principal que almacena información persistente de usuarios, vehículos, dispositivos IoT, viajes, alertas, reportes y suscripciones.

- **Edge Application**

Agente que se ejecuta en instalaciones o vehículos con capacidad de procesamiento local.

Permite cachear datos, realizar análisis preliminar y sincronizar con el **Backend API** cuando la conexión está disponible.

Utiliza su propia **Edge Database** para resiliencia ante desconexiones.

- **Edge Database**

Almacenamiento local de respaldo para los datos capturados por la aplicación Edge, asegurando continuidad operacional incluso en entornos con conectividad limitada.

- **Embedded Application**

Componente ligero desplegado en los dispositivos de monitoreo físicos.

Captura datos ambientales (temperatura, humedad, GPS) y los envía a la **Edge Application** para su procesamiento y sincronización.

Los actores principales interactúan con los contenedores:

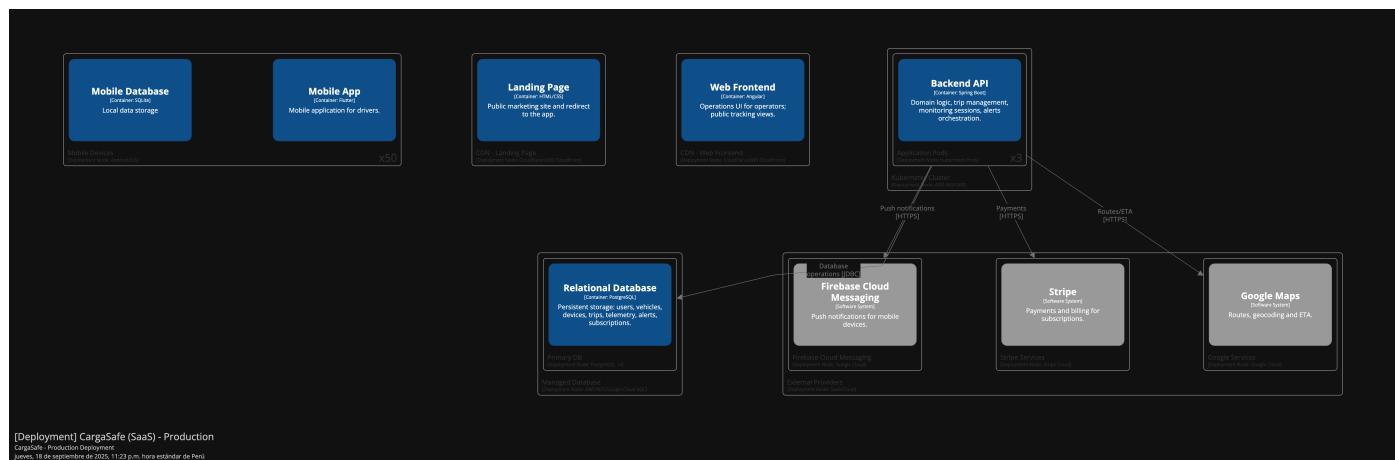
- Company Operator usa la Web App para planificar y supervisar operaciones.
- Driver utiliza la Mobile App para recibir instrucciones y reportar estado de los viajes.
- End Customer accede tanto a la Single Web (para reportes públicos) como a la Mobile App (para recibir notificaciones y links de estado).

Además, CargaSafe se integra con varios sistemas externos:

- **Google Maps**: Provee servicios de geocodificación, cálculo de rutas y estimación de tiempo de llegada (ETA).
Es utilizado tanto por el **Backend API** (para procesamiento de rutas) como por la **SPA** y la **Mobile App** (para visualización e interacción con mapas).
- **Stripe**: Maneja los pagos y la facturación de las suscripciones de la plataforma.
- **Firebase Cloud Messaging (FCM)**: para notificaciones push hacia aplicaciones móviles y web. En conjunto, el diagrama muestra cómo CargaSafe se estructura en contenedores especializados que soportan las necesidades de operadores, conductores y clientes, asegurando tanto la operación online como offline en distintos puntos de la cadena logística.
- **SendGrid**
Servicio de mensajería utilizado para el envío de correos electrónicos transaccionales, reportes y alertas.

4.1.3.3. Software Architecture Deployment Diagrams

El Deployment Diagram de CargaSafe muestra cómo se despliega la solución en un entorno de producción real, representando los nodos de infraestructura, los contenedores de software y las interacciones entre ellos.



Clients:

- Los usuarios finales acceden desde navegadores web, donde la Landing Page y el Web Frontend se sirven por separado desde CDNs independientes (CloudFlare/AWS CloudFront) para optimizar la entrega de contenido.
- Los conductores utilizan una aplicación móvil Flutter en dispositivos Android/iOS, que incluye una base de datos SQLite local para almacenamiento offline y sincronización de datos.
- Todas las peticiones de API se realizan mediante HTTPS y son redirigidas hacia el Load Balancer, encargado de enrutar el tráfico hacia los servicios backend.

Backend y orquestación

- El Backend API (Spring Boot) se despliega dentro de un Kubernetes Cluster en múltiples pods de aplicaciones para alta disponibilidad y escalabilidad.
- El backend centraliza la lógica de negocio, gestiona operaciones de viajes, monitoreo de cadena de frío y orquestación de alertas en tiempo real.

Base de datos

- El sistema utiliza una base de datos PostgreSQL gestionada (AWS RDS/Google Cloud SQL), con una instancia primaria para operaciones de escritura y réplicas de solo lectura para consultas distribuidas y balanceo de carga.
- Los dispositivos móviles mantienen datos críticos localmente en SQLite para funcionamiento offline durante los viajes.

Integraciones externas El backend consume servicios de terceros para extender sus capacidades:

- Google Maps para rutas, geocodificación y cálculo de ETA en tiempo real.
- Stripe para procesamiento de pagos y facturación de suscripciones.
- Firebase Cloud Messaging (FCM) para la entrega de notificaciones push directamente a los dispositivos móviles de los conductores.

Resultado El diagrama de despliegue muestra que la solución CargaSafe está organizada bajo una arquitectura cloud-native optimizada, con:

- Separación de responsabilidades: Landing page y aplicación web servidas independientemente
- Capacidades offline: Base de datos local SQLite en dispositivos móviles
- Kubernetes para la orquestación de contenedores del backend
- CDNs separados para optimizar la entrega de contenido estático
- Base de datos gestionada con réplicas para mejorar el rendimiento y disponibilidad
- Notificaciones push nativas a través de FCM

Esta infraestructura permite un sistema escalable, resiliente y con capacidades offline críticas para la operación de conductores en campo, garantizando la continuidad operativa en la gestión de la cadena de frío incluso sin conectividad permanente.

4.2. Tactical-Level Domain-Driven Design

4.2.1. Bounded Context: Identity and Access Management (IAM)

4.2.1.1. Domain Layer

Entidades Principales

User (Aggregate Root)

Propósito:

Representa al usuario autenticable dentro del sistema y administra su identidad, credenciales y autorización.

Atributos principales:

- `id` — Identificador único (Long)
- `username` — Nombre de usuario único
- `email` — Correo electrónico único
- `passwordHash` — Hash seguro de la contraseña (BCrypt)
- `firstName, lastName`
- `enabled` — Indica si la cuenta está activa
- `roles` — Lista de entidades Role asignadas
- `createdAt, updatedAt`

Métodos principales:

- `updateBasicInfo(firstName, lastName, email)`
 - `changePassword(oldPassword, newPassword)`
 - `assignRole(role)`
 - `enable(), disable()`
-

Role (Entity)

Propósito:

Define los roles utilizados para autorización en el sistema.

Atributos principales:

- `id`
- `name` — (ADMIN, CUSTOMER, LOGISTICS_MANAGER, DRIVER...)
- `description`

Métodos principales:

- `rename(name)`
-

Value Objects

- **Email** — Garantiza formato válido y normalización.
 - **Username** — Enforcea reglas sintácticas.
 - **Password** — Valida complejidad y encapsula hashing.
 - **JwtClaims** — Representa claims del JWT (sub, roles, exp, jti, iat).
-

Domain Services

- **PasswordHashingService**

- Hashing seguro con BCrypt
- Comparación segura de contraseñas

- **JwtTokenService**

- Generación de Access Token y Refresh Token
- Validación de firma JWT
- Rotación segura de Refresh Token (Refresh Token Rotation)

- **AuthorizationService**

- Verifica roles desde los claims del JWT
-

Commands

- RegisterUserCommand
 - LoginUserCommand
 - UpdateUserProfileCommand
 - ChangePasswordCommand
 - AssignRoleCommand
 - LogoutAllDevicesCommand
 - RevokeTokenCommand
-

Queries

- GetUserByIdQuery
 - GetUserByEmailQuery
 - GetAllUsersQuery
 - GetUserRolesQuery
-

Events

Tu implementación real incluye únicamente:

- **SeedRolesCommand → RoleCommandServiceImpl**

Se ejecuta al iniciar la aplicación para crear los roles base.

4.2.1.2. Interface Layer

AuthController

Método	Endpoint	Descripción
POST	/api/v1/auth/login	Autentica usuario y devuelve Access + Refresh Token
POST	/api/v1/auth/refresh-token	Rotación segura de Refresh Token
POST	/api/v1/auth/logout	Revoca el token actual
POST	/api/v1/auth/logout-all	Revoca todos los tokens del usuario
POST	/api/v1/auth/register	Registra un nuevo usuario

UserController

Método	Endpoint	Descripción
GET	/api/v1/users/me	Obtiene el perfil del usuario autenticado
PUT	/api/v1/users/me	Actualiza perfil del usuario
PUT	/api/v1/users/me/change-password	Cambia contraseña
GET	/api/v1/users/{id}	Obtiene usuario por ID (solo ADMIN)

Security / ACL

La autorización se basa completamente en JWT firmados:

- Claims incluyen roles y jti
- Token Revocation mediante repositorio persistente
- Filtros de seguridad:
 - JwtAuthenticationFilter

- `TokenRevocationFilter`
 - `SecurityConfig` define:
 - Rutas públicas: `/auth/**`
 - Rutas protegidas por rol
 - Políticas de CORS para permitir comunicación con frontend
-

4.2.1.3. Application Layer

Command Services

UserCommandServiceImpl

- Gestiona creación y actualización de usuarios
 - Valida unicidad de email y username
 - Hashing seguro de contraseñas
 - Asignación de roles
 - Procesa `ChangePasswordCommand`
 - Maneja auditoría (`createdAt`, `updatedAt`)
-

AuthCommandServiceImpl

- Autenticación de usuarios
 - Generación de Access Token + Refresh Token
 - Refresh Token Rotation
 - Logout (revocación del token actual)
 - Logout All Devices (revoca todos los tokens asociados)
 - Mantiene repositorio de tokens revocados
-

Query Services

UserQueryServiceImpl

- Consultas de solo lectura
 - Manejo de vistas y proyecciones
 - Optimización para endpoints REST
-

AuthQueryServiceImpl

- Validación de tokens
 - Obtención de información de sesión
 - Autoridad para autorización basada en claims
-

Event Handlers

- **SeedRolesCommandHandler (RoleCommandServiceImpl)**
Crea roles base automáticamente al iniciar la aplicación.
-

4.2.1.4. Infrastructure Layer

Repositories

UserRepository

- Persistencia del agregado User
 - Consultas por email, username, id
 - Implementado con Spring Data JPA
-

RoleRepository

- Gestión de roles
 - Usado por SeedRoles
 - Prevención de duplicados
-

TokenRevocationRepository

- Almacena tokens revocados mediante su jti

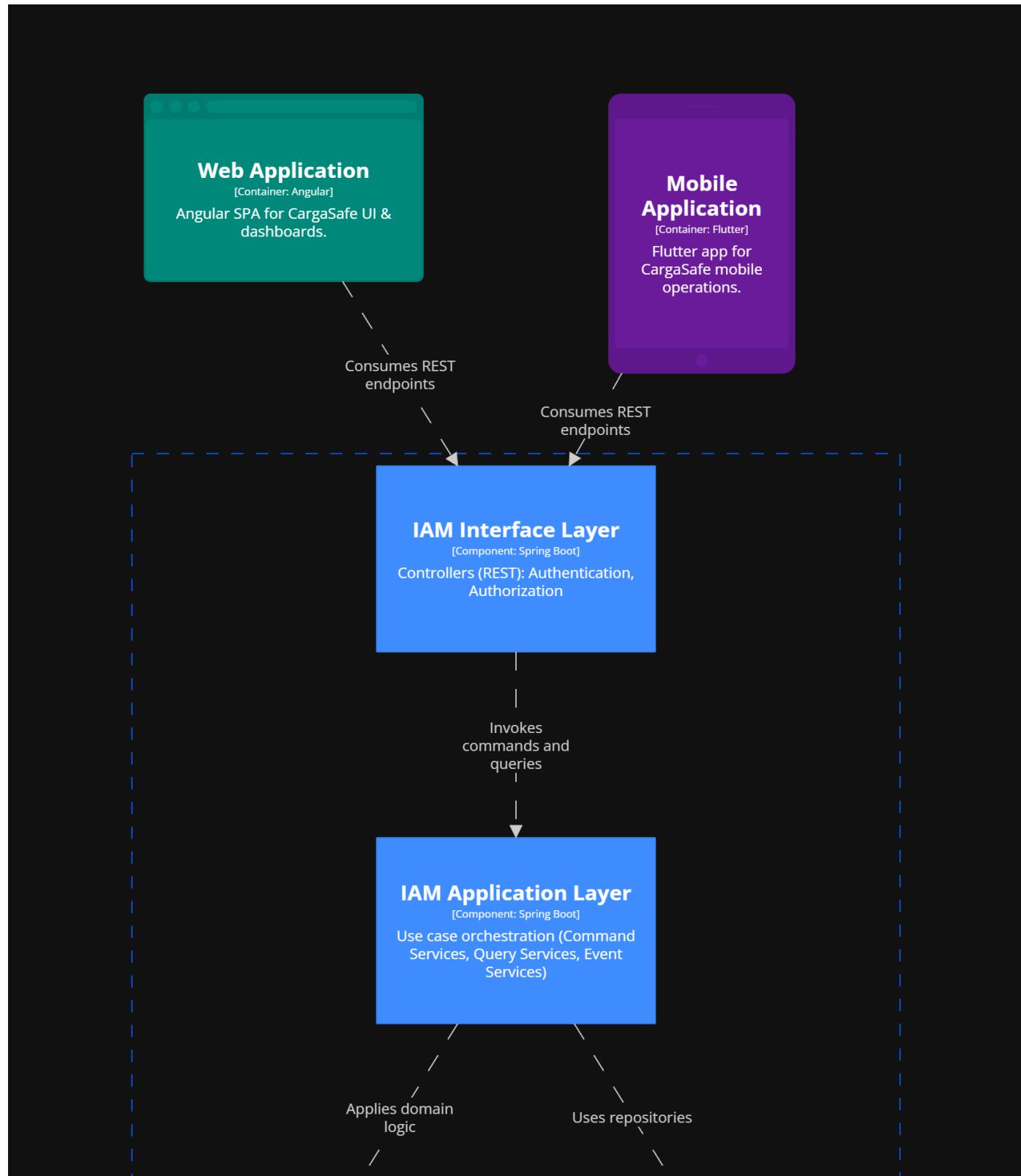
- Evita reuso de Refresh Tokens
- Limpieza automática de tokens expirados

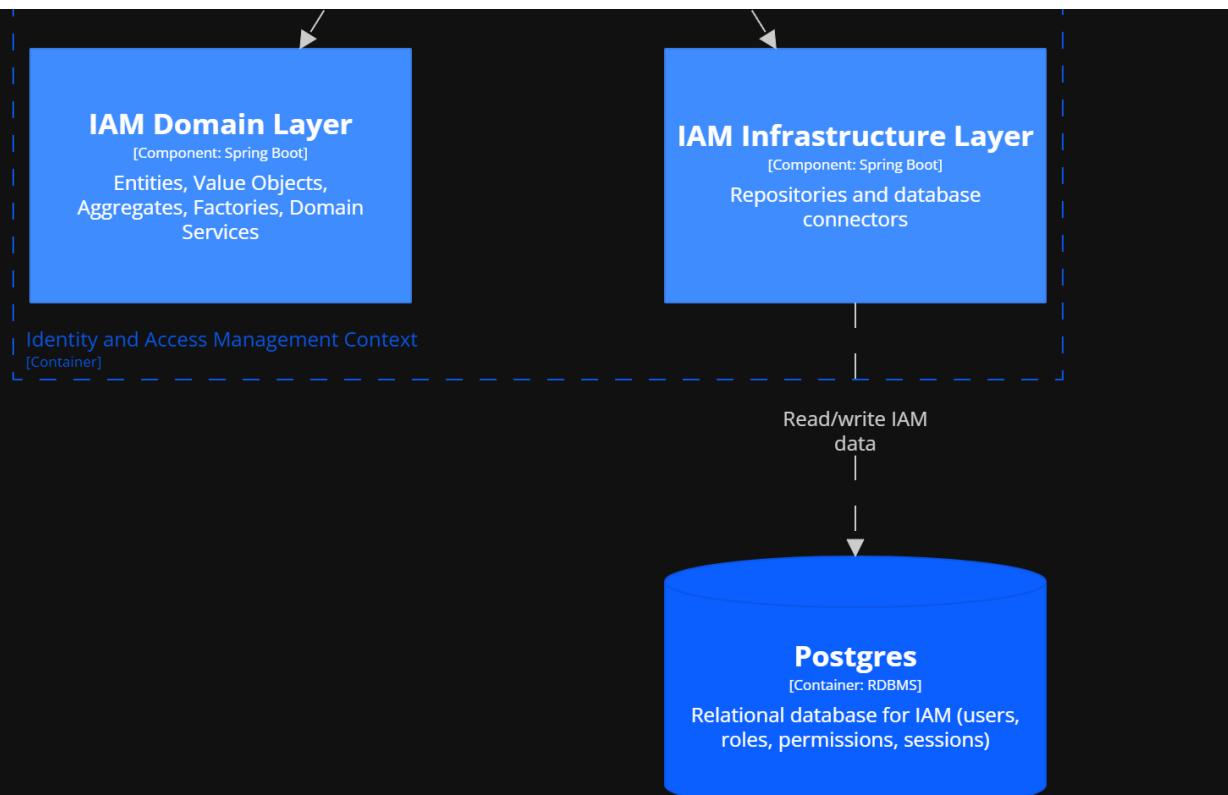
Infraestructura Complementaria

- BCryptPasswordEncoder
- JwtEncoder y JwtDecoder (Nimbus JOSE)
- SecurityFilterChain
- Manejo de CORS para frontend
- Manejo global de excepciones para errores de seguridad

4.2.1.5. Bounded Context Software Architecture Component Level Diagrams

Diagrama de Componentes - Backend - Identity and Access Management





Identity and Access Management Context - Component Diagram

Internal layered components of the IAM Context inside CargaSafe.
viernes, 19 de setiembre de 2025, 11:33 p. m. hora estándar de Perú

Este diagrama muestra la arquitectura por capas del bounded context IAM en el backend. La separación clara entre Interface, Application, Domain e Infrastructure layers permite un diseño mantenible y testeable. Los controllers en la Interface Layer reciben requests HTTP y delegan a los command/query services en Application Layer, que utilizan el dominio y persisten através de repositories en Infrastructure Layer.

Diagrama de Componentes - Frontend Web - Identity and Access Management

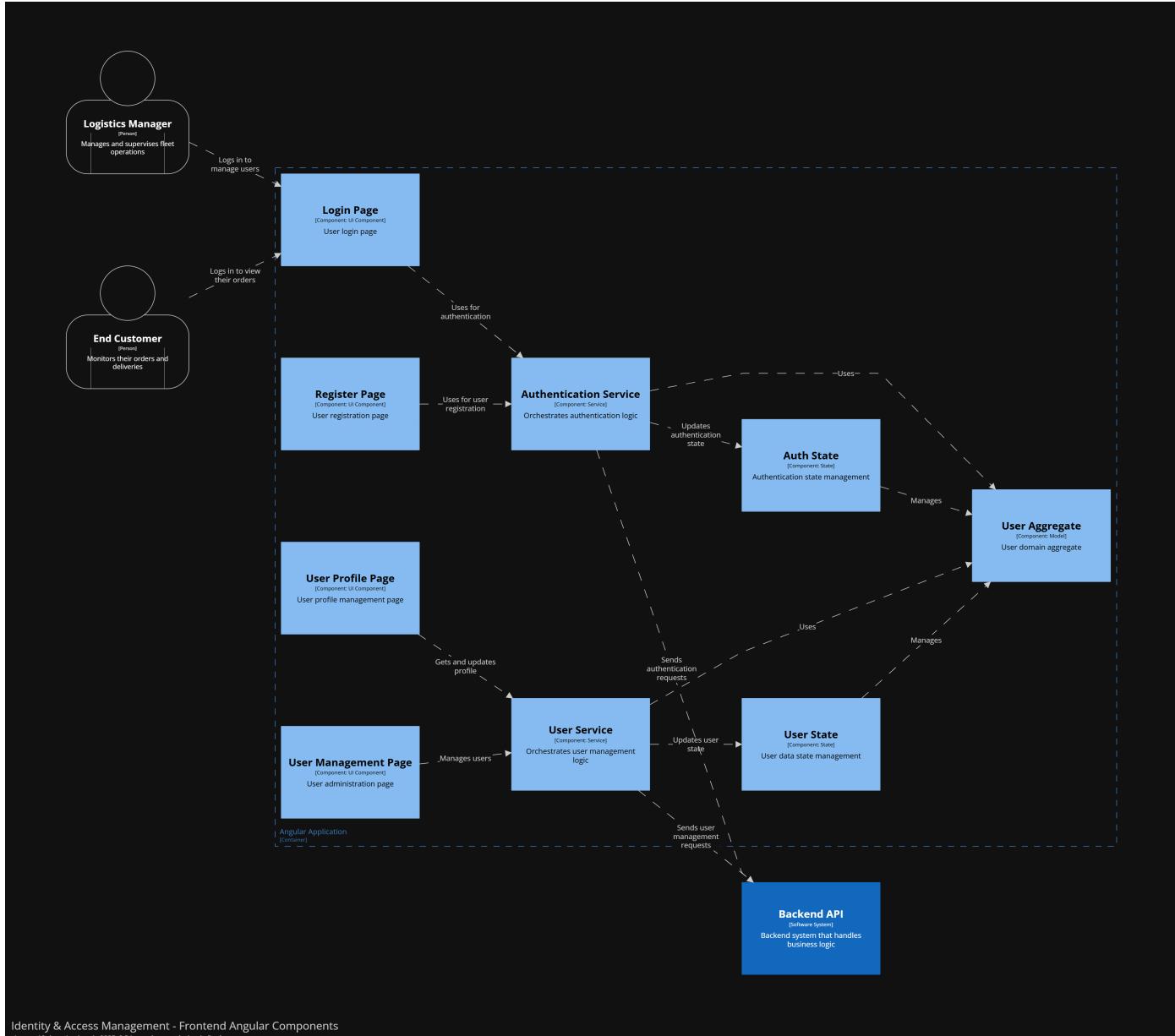
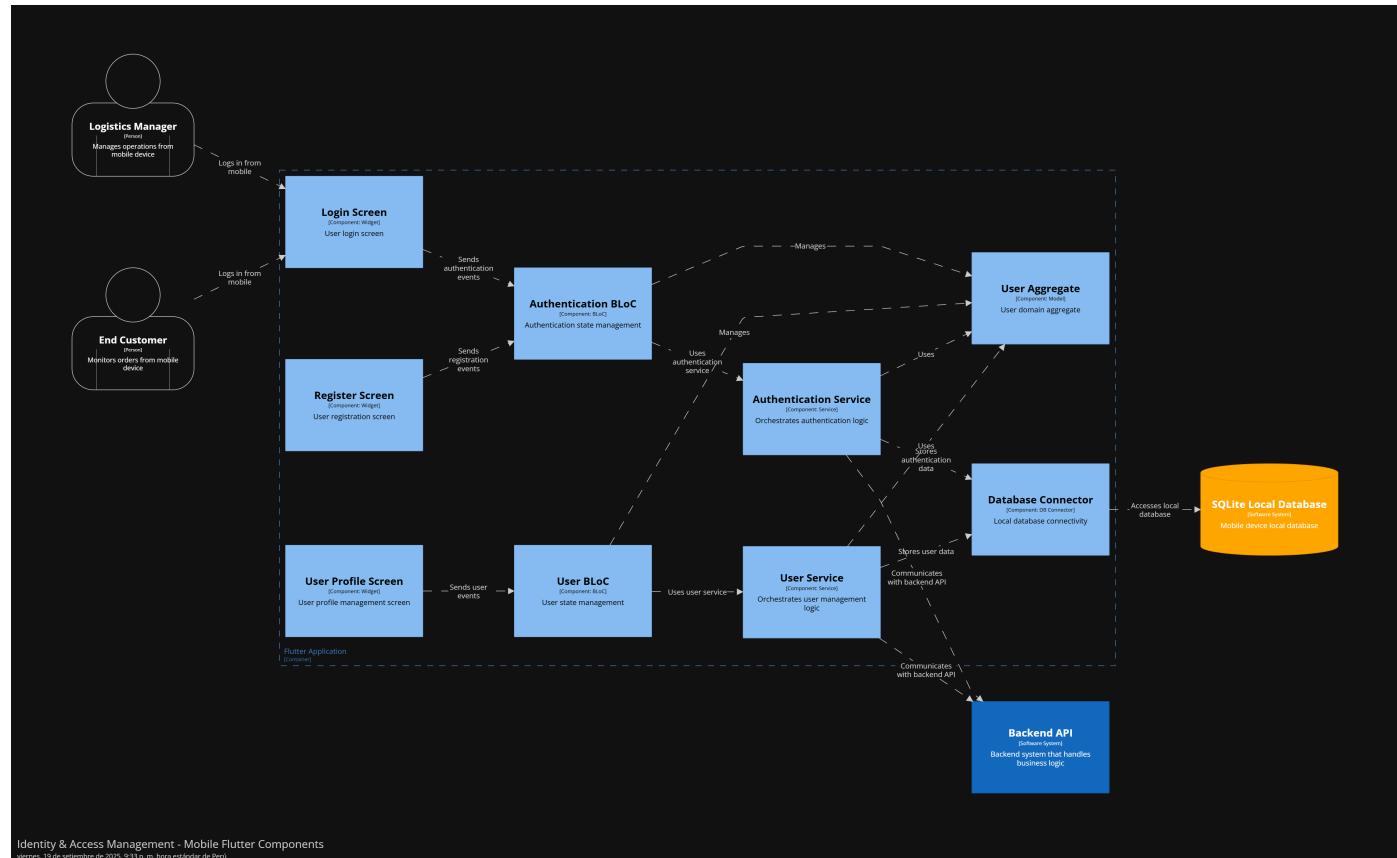


Diagrama de Componentes - Mobile - Identity and Access Management



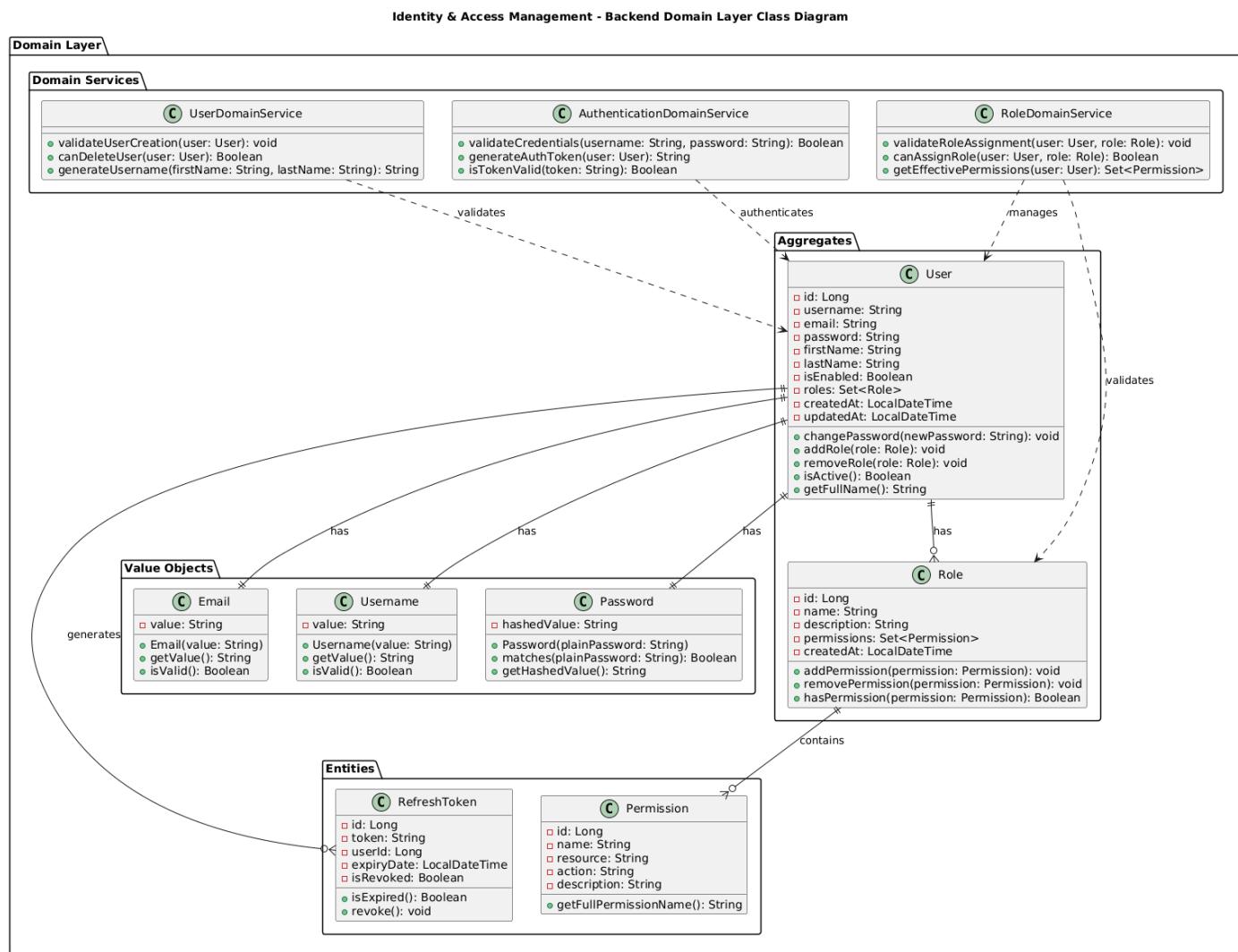
Identity & Access Management - Mobile Flutter Components
viernes, 19 de setiembre de 2025, 9:33 p. m. hora estándar de Perú

La aplicación móvil utiliza Flutter con arquitectura BLoC para state management. Las pantallas (screens) envían eventos a BLoCs que manejan el estado y coordinan con services. Los services se comunican tanto con el backend API como con la base de datos local SQLite para funcionalidad offline.

4.2.1.6. Bounded Context Software Architecture Code Level Diagrams

4.2.1.6.1. Bounded Context Domain Layer Class Diagrams

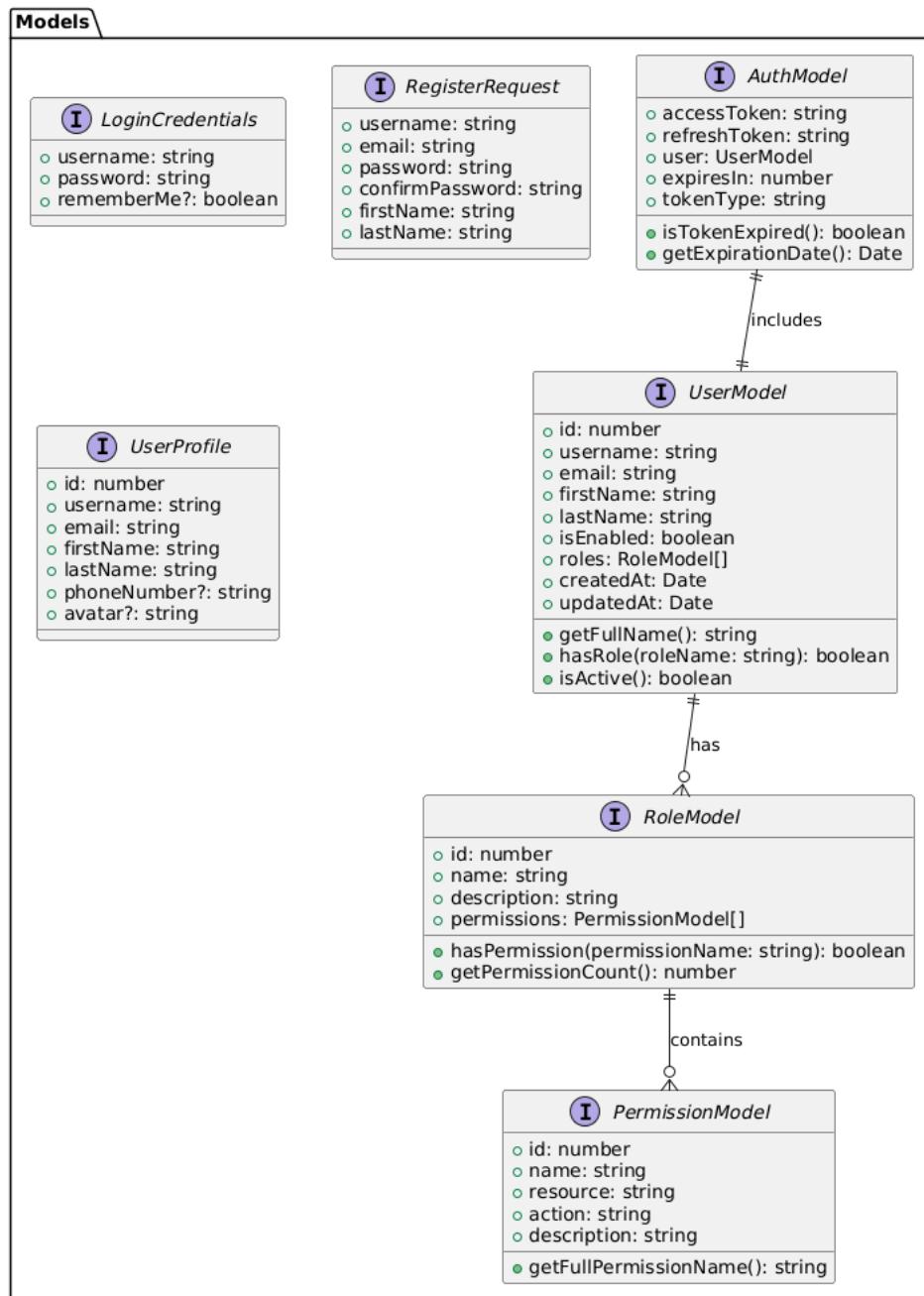
Backend - Identity & Access Management Domain Layer Class Diagram



El diagrama de clases del backend muestra las entidades principales del IAM bounded context en la capa de dominio. La entidad User actúa como aggregate root y maneja la lógica de autenticación y autorización. Los roles están conectados através de relaciones many-to-many con usuarios, mientras que los tokens gestionan las sesiones y refresh tokens. La estructura implementa el patrón Repository para la persistencia.

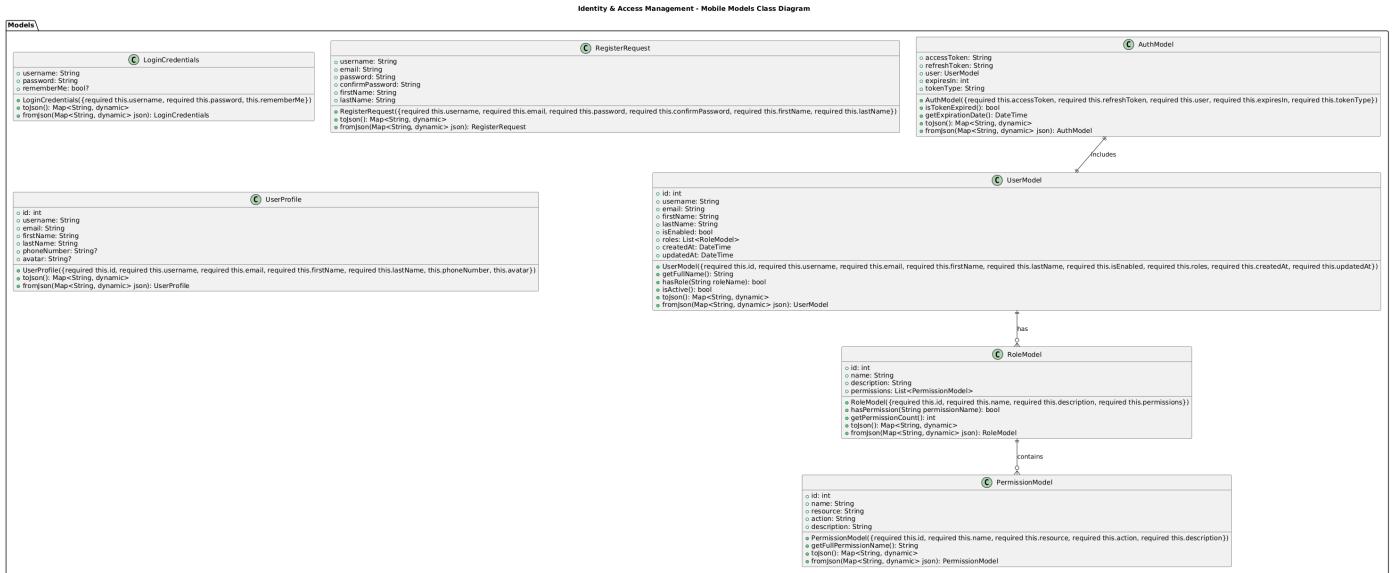
Frontend - Identity & Access Management Domain Layer Class Diagram

Identity & Access Management - Frontend Models Class Diagram



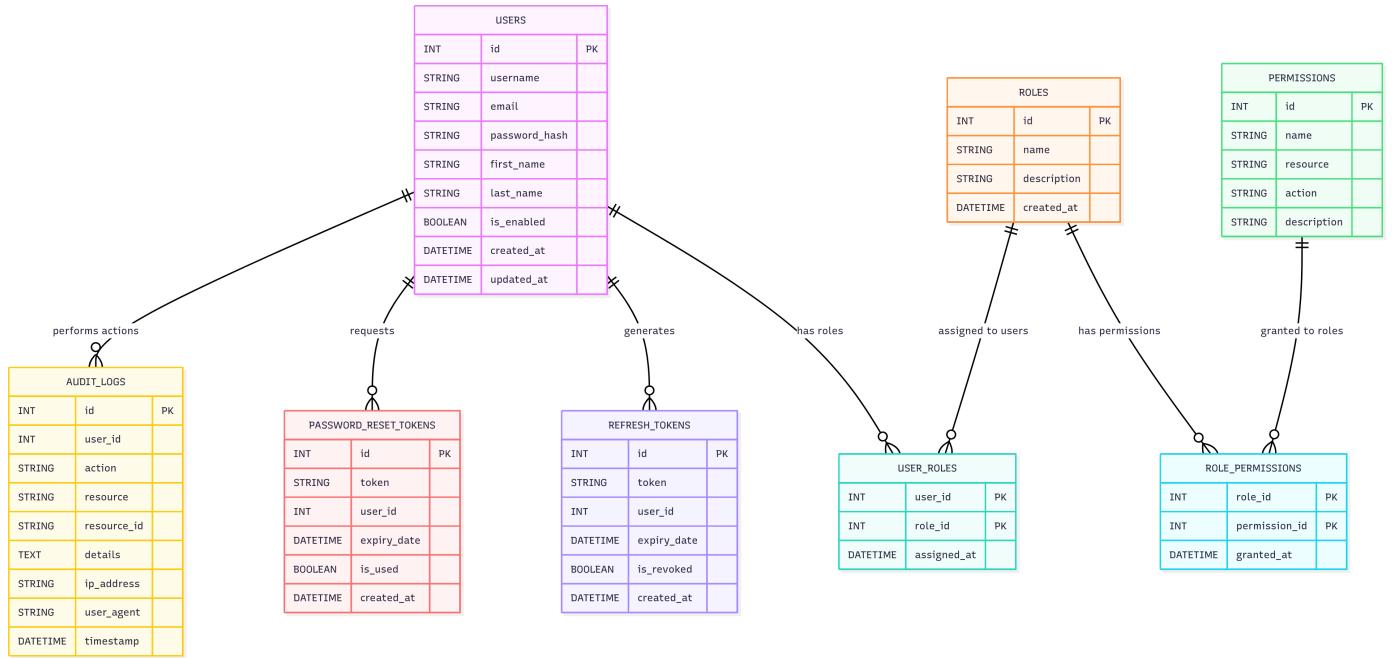
El diagrama del frontend Angular muestra la arquitectura de componentes y servicios para el manejo de identidad. Los componentes (Login, Register, Profile) interactúan con servicios específicos que manejan el estado de autenticación. El AuthService centraliza la lógica de comunicación con el backend API, mientras que los guards protegen las rutas según permisos.

Mobile - Identity & Access Management Domain Layer Class Diagram



La aplicación móvil Flutter implementa BLoC pattern para el manejo de estado de autenticación. Los BLoCs (AuthBloc, UserBloc) coordinan entre las pantallas y los services, mientras que el local storage permite funcionalidad offline. La arquitectura asegura sincronización de credenciales entre la app y el backend.

4.2.1.6.2. Bounded Context Database Design Diagram



El diagrama de base de datos implementa un modelo RBAC (Role-Based Access Control) robusto. Las tablas principales (USERS, ROLES, PERMISSIONS) están conectadas a través de tablas de unión que permiten relaciones many-to-many. Se incluyen tablas auxiliares para tokens de sesión, logs de auditoría y tokens de recuperación de contraseña. La estructura está optimizada para consultas frecuentes de autorización y mantiene integridad referencial.

4.2.2. Bounded Context: Subscriptions and Billing

4.2.2.1. Domain Layer

Entities

Plan

- Propósito:** Representar los distintos planes de suscripción disponibles.
- Atributos principales:** planId, name, limits, price, description.
- Métodos principales:** Creación mediante 'CreatePlanCommand', Getters y setters para sus atributos.

Subscription

- Propósito:** Gestionar el ciclo de vida de una suscripción asociada a un usuario.
- Atributos principales:** subscriptionId, userId, plan, status (ACTIVE, REVOKED, SUSPENDED), renewal, paymentMethod.
- Métodos principales:** changePlan(newPlan): permite cambiar de plan si el estado actual es ACTIVE, delete(): elimina la suscripción si existe, Creación mediante CreateSubscriptionCommand.

Payment

- **Propósito:** Registrar los pagos efectuados por los usuarios asociados a sus suscripciones.
- **Atributos principales:** paymentId, userId, transactionId, amount, receiptUrl, status (PENDING, SUCCEEDED, FAILED), paymentDate.
- **Métodos principales:** Creación mediante CreatePaymentCommand, Actualización automática del estado a SUCCEEDED al registrarse el pago.

Commands

- **CreatePlanCommand**
- **CreateSubscriptionCommand**
- **ChangePlanCommand**
- **DeleteSubscriptionCommand**
- **CreatePaymentCommand**

Queries

GetAllPlansQuery **GetPlanByIdQuery** **GetSubscriptionByUserIdQuery** **GetPaymentsByUserIdQuery**

4.2.2.2. Interface Layer

PlanController

- **POST /api/v1/plans:** Crea un nuevo plan de suscripción.
- **GET /api/v1/plans:** Lista todos los planes disponibles.

SubscriptionController

- **POST /api/v1/subscriptions:** Crea una nueva suscripción para un usuario.
- **PATCH /api/v1/subscriptions/{subscriptionId}/plan:** Cambia el plan actual (solo si la suscripción está ACTIVA).
- **DELETE /api/v1/subscriptions/{subscriptionId}:** Elimina una suscripción existente.
- **GET /api/v1/subscriptions/user-id/{userId}:** Consulta la suscripción activa de un usuario.

PaymentController

- **POST /api/v1/payments:** Registra un nuevo pago asociado a un usuario.
- **GET /api/v1/payments/user-id/{userId}:** Lista los pagos de un usuario.

4.2.2.3. Application Layer

Command Services

- **PlanCommandService:** Crea nuevos planes validando que no se repitan nombres.
- **SubscriptionCommandService:** Gestiona creación, cambio y eliminación de suscripciones y valida que un usuario no tenga más de una suscripción activa.
- **PaymentCommandService:** Registra pagos nuevos, validando unicidad de transactionId y marca automáticamente el estado del pago como SUCCEEDED.

Query Services

PlanQueryService: Obtiene planes por ID o lista completa. **SubscriptionQueryService:** Recupera suscripción por userId. **PaymentQueryService:** Lista pagos por usuario.

4.2.2.4. Infrastructure Layer

Repositories (Interfaces)

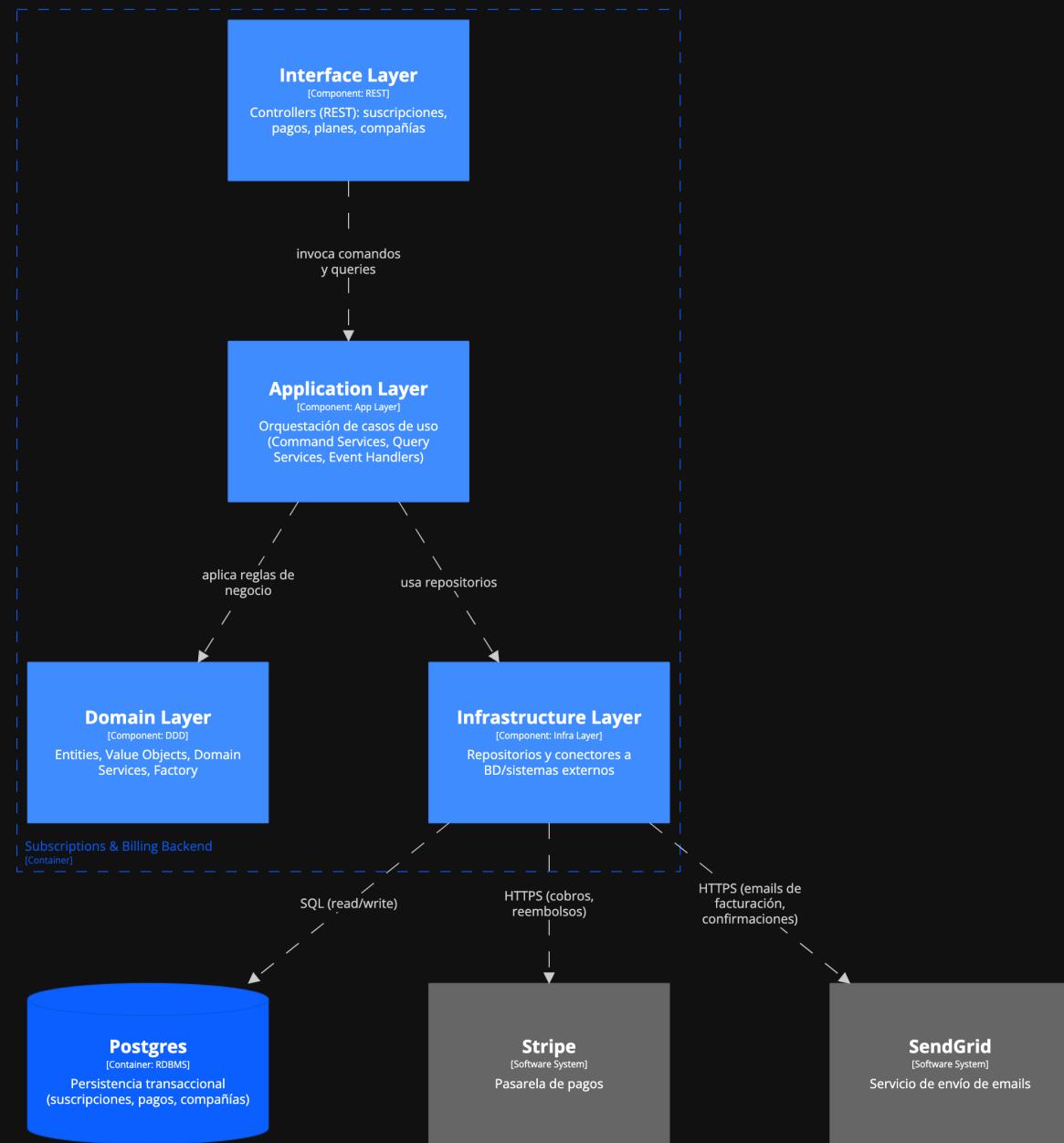
- **PlanRepository:** Maneja persistencia de planes (existsByName, findAll).
- **SubscriptionRepository:** Maneja suscripciones (existsByUserId, findByUserId).
- **PaymentRepository:** Maneja pagos (existsByTransactionId, findAllByUserId).

Persistence & Configuration

- **Base de datos:** PostgreSQL (configurada vía application.properties).
- **Estrategia de nombres:** SnakeCasePhysicalNamingStrategy.
- **Variables de entorno preparadas para Stripe:** (STRIPE_SECRET_KEY, STRIPE_WEBHOOK_SECRET).

4.2.2.5. Bounded Context Software Architecture Component Level Diagrams

Diagrama de componentes - Backend - Subscriptions and Billing



Subscriptions & Billing Backend - Component Diagram

Componentes internos del backend organizados en capas DDD.
sábado, 20 de septiembre de 2025, 12:41 a.m. hora estándar de Perú

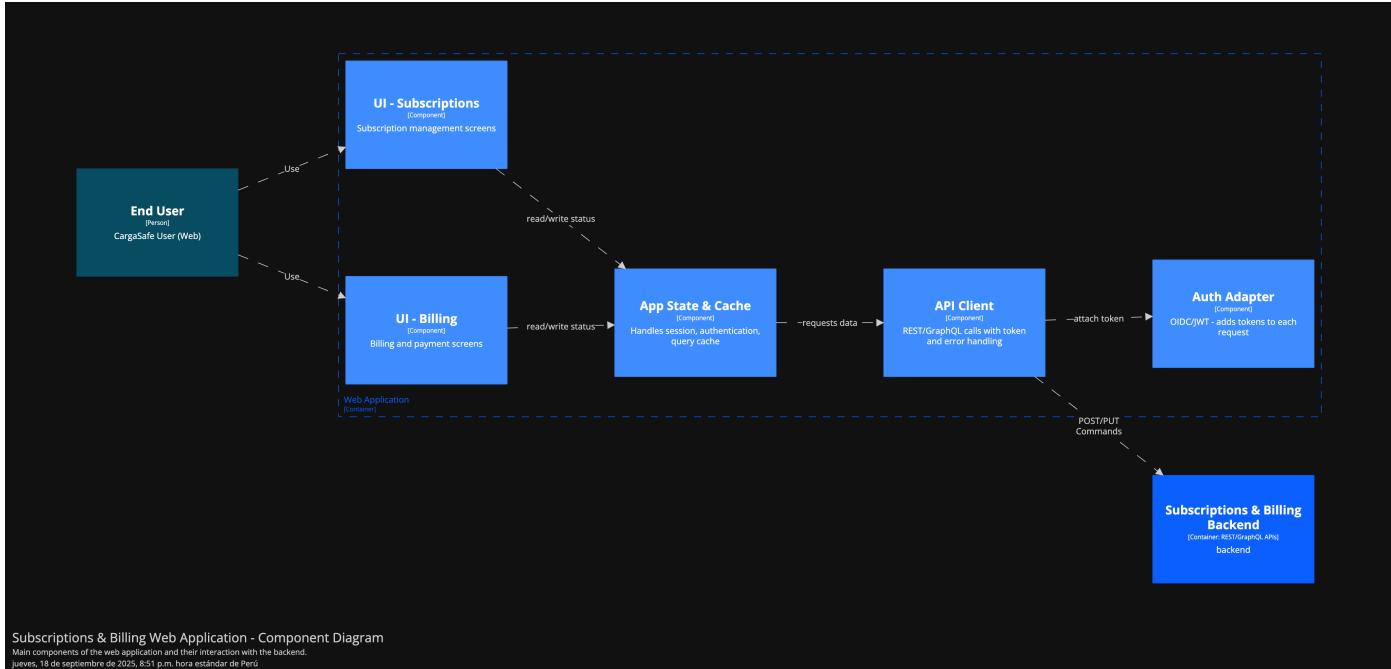
El backend del bounded context de Suscripciones y Pagos está estructurado siguiendo el enfoque de Domain-Driven Design (DDD), y se organiza en cuatro capas principales:

- **Interface Layer:** expone los controladores REST que atienden las operaciones de planes, suscripciones y pagos. Es la puerta de entrada para los usuarios o sistemas que consumen la API.
- **Application Layer:** orquesta los casos de uso mediante Command Services, Query Services y, potencialmente, Event Handlers. En esta capa se coordinan las operaciones y se aplican las reglas de negocio definidas en el dominio.
- **Domain Layer:** concentra la lógica de negocio del contexto, incluyendo las entidades Plan, Subscription y Payment. Define los estados y comportamientos que rigen el ciclo de vida de las suscripciones y los pagos.
- **Infrastructure Layer:** implementa los repositorios JPA y conectores hacia la base de datos y posibles servicios externos. Se encarga de la persistencia y de la integración técnica con sistemas externos.

Conexiones externas actuales:

- **Postgres:** persistencia transaccional de planes, suscripciones y pagos.
- **Stripe:** pasarela de pagos (configurada mediante variables de entorno, preparada para integración futura).
- **SendGrid:** servicio de envío de correos electrónicos (reservado para notificaciones de facturación y confirmaciones).

Diagrama de componentes - Application Web - Subscriptions and Billing



La aplicación web se comunica con el bounded context Subscriptions exclusivamente a través de endpoints REST del backend (mismo API para commands y queries). No existe un módulo separado de "Billing" ni una "Query API" independiente.

Estructura de la Web Application:

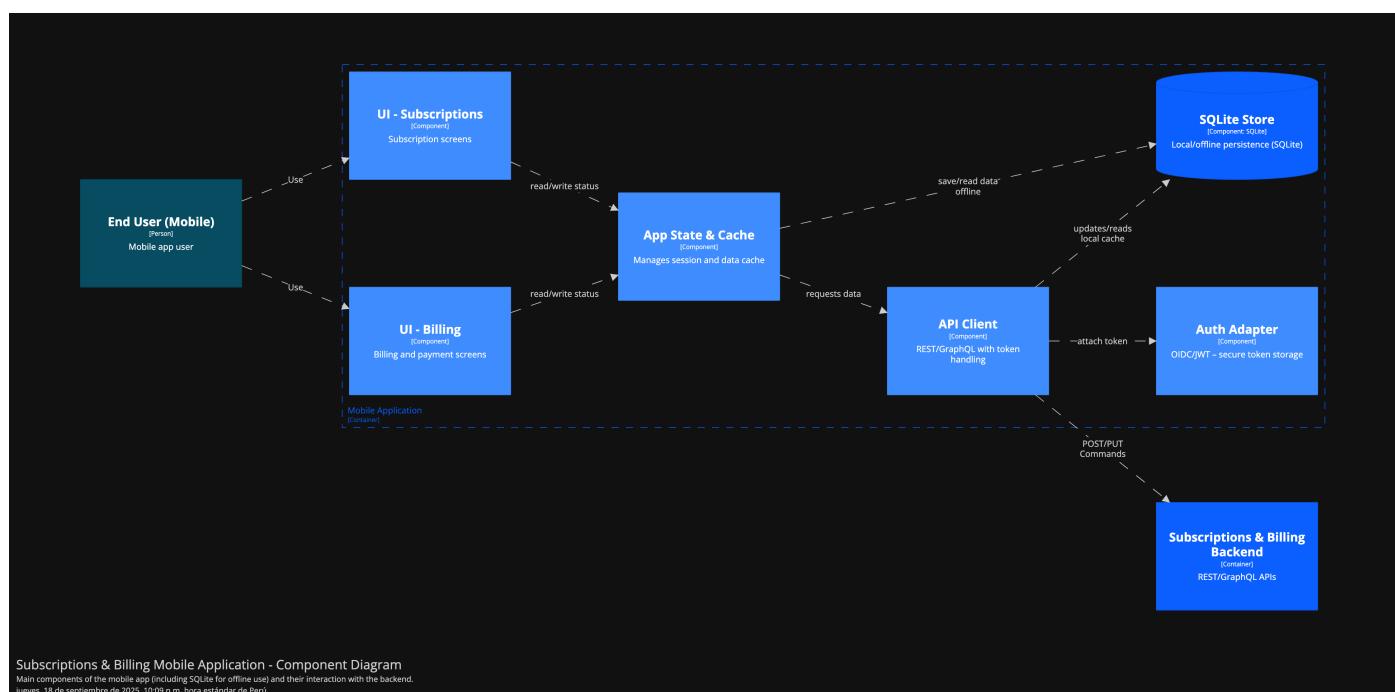
UI – Subscriptions: alta de suscripción, cambio de plan, visualización del plan vigente y lista de pagos del usuario. **App State & Cache:** gestiona sesión/autenticación, almacén de estado y caché de consultas. **API Client:** cliente HTTP que invoca la Subscriptions REST API, agrega el token (vía Auth Adapter) y maneja errores/reintentos. **Auth Adapter:** integra OIDC/JWT para adjuntar credenciales en cada request. (Si la autenticación aún no está implementada en el backend de este BC, se mantiene como integración prevista).

Alcance funcional en el cliente: La aplicación web no implementa lógica de negocio, solo presenta datos y envía intenciones del usuario (crear suscripción, cambiar plan, eliminar, registrar pago).

Conexiones externas representadas en el backend:

- **Postgres** (persistencia transaccional de planes, suscripciones y pagos).
- **Stripe** (pasarela de pagos — integración planificada).
- **SendGrid** (correos transaccionales — integración planificada).

Diagrama de componentes - Mobile Application - Subscriptions and Billing



La aplicación móvil de Subscriptions mantiene una arquitectura muy similar a la versión web, ya que también se conecta al backend a través de la Subscriptions REST API, utilizando comandos (POST/PUT) para ejecutar acciones y consultas (GET) para obtener datos de planes, suscripciones y pagos.

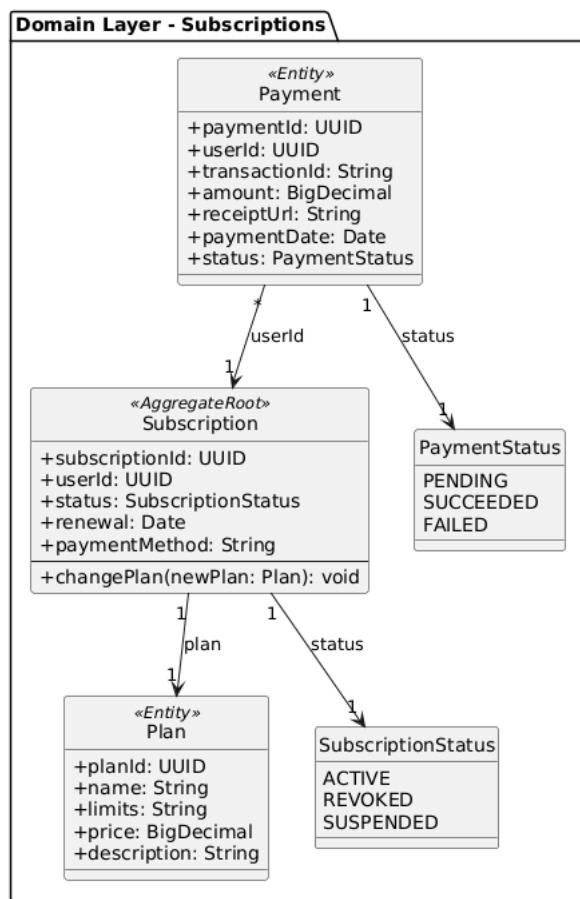
La principal diferencia con respecto a la versión web es que en el entorno móvil se incorpora una base de datos local SQLite, la cual permite el funcionamiento offline. Gracias a esta base local, la aplicación puede continuar operando aun sin conexión, guardando temporalmente los datos y sincronizándolos cuando se restablece el acceso a internet.

La app se organiza en los siguientes componentes:

- **UI – Subscriptions**: pantallas de gestión de suscripción, cambio de plan y visualización del estado actual.
- **Payments**: pantallas de pagos o facturación, integradas dentro del mismo módulo.
- **App State & Cache**: administra la sesión, autenticación y almacenamiento temporal de datos.
- **SQLite**: guarda información localmente para permitir modo offline.
- **API Client**: maneja las solicitudes HTTP al backend, incluyendo reintentos y gestión de errores.
- **Auth Adapter**: adjunta el token de autenticación (OIDC/JWT) a cada petición al backend.

4.2.2.6. Bounded Context Software Architecture Code Level Diagrams

4.2.2.6.1. Bounded Context Domain Layer Class Diagrams



Explicación del diagrama

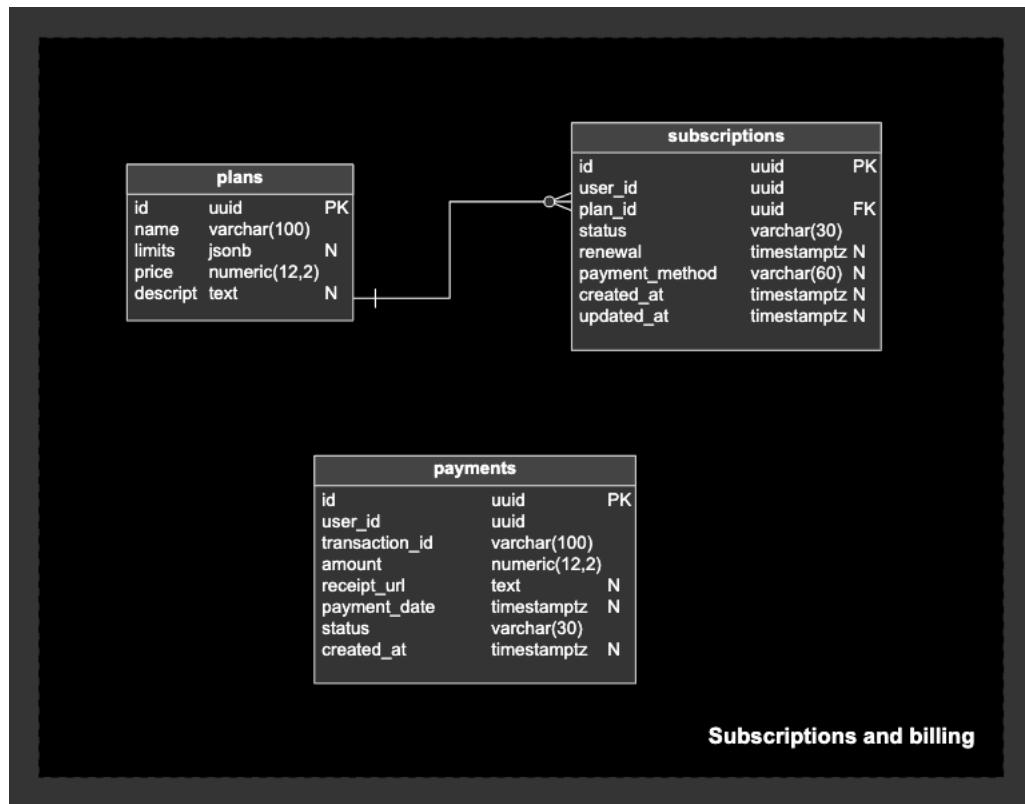
El diagrama de clases del Domain Layer se centra en tres entidades: Subscription (aggregate), Plan (entity) y Payment(entity).

Subscription actúa como agregado principal y gestiona su ciclo de vida mediante el estado status, con los valores ACTIVE, REVOKED y SUSPENDED. La operación principal de negocio implementada es changePlan(newPlan), que solo procede cuando la suscripción está ACTIVE.

Plan modela las ofertas disponibles y persiste atributos como name, limits, price y description.

Payment registra transacciones asociadas a un usuario (userId) con transactionId único, amount, receiptUrl, paymentDate y status (PENDING, SUCCEEDED, FAILED).

4.2.2.6.2. Bounded Context Database Design Diagram



Explicación del diagrama

El diseño de base de datos está compuesto por tres tablas principales: **plans**, **subscriptions** y **payments**. Este modelo representa la persistencia mínima necesaria para gestionar los planes disponibles, el ciclo de vida de las suscripciones y el registro de pagos asociados a los usuarios.

La tabla **plans** almacena los planes disponibles dentro del sistema, incluye el identificador del *plan* (*id*), el *nombre* (*name*), los límites o características del plan (*limits*, en formato JSON), el precio (*price*) y una descripción (*description*). El campo *name* se encuentra definido como único con el fin de evitar duplicados en la configuración de planes.

La tabla **subscriptions** representa la suscripción activa o histórica de cada usuario. Contiene el identificador de la suscripción (*id*), el identificador del usuario (*user_id*), el plan al que se encuentra asociado (*plan_id*), el estado de la suscripción (*status*, cuyos valores posibles son ACTIVE, REVOKED o SUSPENDED), la fecha de renovación (*renewal*) y el método de pago (*payment_method*). Además, almacena marcas de auditoría como *created_at* y *updated_at*. La relación con la tabla **plans** se establece mediante la clave foránea *plan_id* → *plans.id*, que garantiza que toda suscripción referencia un plan válido. La restricción de que un usuario solo puede mantener una suscripción en estado ACTIVE es aplicada a nivel de la capa de aplicación.

La tabla **payments** registra los pagos realizados por los usuarios. Cada registro incluye el identificador del pago (*id*), el usuario asociado (*user_id*), el identificador único de transacción (*transaction_id*), el monto (*amount*), la URL del comprobante (*receipt_url*), la fecha del pago (*payment_date*) y el estado del pago (*status*, con valores permitidos PENDING, SUCCEEDED o FAILED). El campo *transaction_id* se define como único para garantizar idempotencia en el procesamiento de transacciones, evitando duplicidad de cobros.

4.2.3. Bounded Context: *Alerts & Resolution*

4.2.3.1. Domain Layer

Entidades (Entities)

Entity: Alert (Aggregate Root)

Propósito principal

Centralizar la gestión del ciclo de vida de una alerta y garantizar que se cumplan las reglas de negocio.

Atributos principales

- **alertId**: Identificador único de la alerta.
- **type**: Tipo de alerta (OutOfRange, Offline, RouteDeviation).
- **status**: Estado actual de la alerta (OPEN, ACKNOWLEDGED, CLOSED).
- **sensorType**: Tipo de sensor que la generó (TEMPERATURE, HUMIDITY, VIBRATION, TILT, LOCATION, BATTERY).
- **createdAt**: Fecha y hora de creación de la alerta.
- **acknowledgedAt**: Momento en que fue reconocida.
- **closedAt**: Momento en que fue cerrada.

Métodos principales

- **acknowledge()**: Marca la alerta como reconocida.
- **close()**: Cierra la alerta si ya fue reconocida.
- **escalate()**: Incrementa la criticidad si no fue atendida a tiempo.

Entity: Notification**Propósito principal**

Representar un mensaje enviado a un usuario sobre una alerta.

Atributos principales

- notificationId: Identificador único de la notificación.
- alertId: Referencia a la alerta asociada.
- channel: Canal de comunicación (EMAIL, SMS, FCM).
- message: Contenido del mensaje.
- sentAt: Fecha y hora de envío.

Métodos principales

- markAsSent(): Actualiza el estado de la notificación como enviada.

Entity: Incident**Propósito principal**

Registrar un evento relacionado con un viaje que se crea a partir de una alerta.

Atributos principales

- incidentId: Identificador único del incidente.
- alertId: Referencia a la alerta origen.
- tripId: Identificador del viaje asociado.
- description: Detalle del incidente.
- createdAt: Fecha y hora de creación.

Métodos principales

- resolve(description): Marca el incidente como resuelto con detalles.

Objetos de Valor (Value Objects)

- AlertType: clasifica los tipos de alertas (OutOfRange, Offline, RouteDeviation).
- AlertStatus: define en qué etapa se encuentra la alerta (Open, Acknowledged, Closed).
- NotificationChannel: indica el medio de comunicación usado (Email, SMS, FCM).
- PersistenceWindow: define el tiempo mínimo que debe cumplirse para que un evento se considere válido como alerta.
- SensorType: clasifica la fuente de monitoreo (TEMPERATURE, HUMIDITY, VIBRATION, TILT, LOCATION, BATTERY).

Commands**Command: CreateAlertCommand****Parámetros**

- type, sensorType, createdAt.

Cómo funciona

Se ejecuta al detectar un evento anómalo. Crea una nueva alerta validando reglas como la ventana de persistencia y evitando duplicación.

Command: AcknowledgeAlertCommand**Parámetros**

- alertId.

Cómo funciona

Permite a un operador reconocer la alerta. Cambia su estado a *ACKNOWLEDGED* y registra la hora.

Command: CloseAlertCommand**Parámetros**

- alertId.

Cómo funciona

Cierra una alerta reconocida, cambiando su estado a *CLOSED* y registrando la fecha de cierre.

Command: EscalateAlertCommand**Parámetros**

- alertId.

Cómo funciona

Incrementa la criticidad de una alerta que lleva demasiado tiempo sin ser reconocida, generando un evento de escalamiento.

Command: CreateIncidentFromAlertCommand**Parámetros**

- alertId, tripId, description.

Cómo funciona

Crea un incidente asociado a un viaje a partir de una alerta específica, permitiendo registrar el detalle del evento.

Command: SendNotificationCommand**Parámetros**

- alertId, channel, message.

Cómo funciona

Ordena enviar una notificación al canal definido (Email, SMS, FCM) para informar al usuario o empresa sobre la alerta.

Queries

Query: GetAlertByIdQuery

Parámetros

- alertId.

Cómo funciona

Recupera los detalles de una alerta específica, incluyendo su estado, tipo y fechas clave.

Query: GetAlertsByStatusQuery

Parámetros

- status.

Cómo funciona

Devuelve todas las alertas con un estado determinado (ej. abiertas, reconocidas, cerradas).

Query: GetAlertsByTypeQuery

Parámetros

- type.

Cómo funciona

Recupera todas las alertas de un tipo específico (ej. RouteDeviation).

Query: GetNotificationsByAlertIdQuery

Parámetros

- alertId.

Cómo funciona

Devuelve todas las notificaciones emitidas en relación con una alerta.

Query: GetIncidentsByAlertIdQuery

Parámetros

- alertId.

Cómo funciona

Obtiene todos los incidentes generados a partir de una alerta determinada.

Events

Event: AlertCreatedEvent

Se emite cuando una nueva alerta es registrada en el sistema.

Event: AlertAcknowledgedEvent

Se emite cuando una alerta es reconocida.

Event: AlertClosedEvent

Se emite cuando una alerta se cierra exitosamente.

Event: AlertEscalatedEvent

Se emite cuando una alerta aumenta de criticidad por falta de respuesta.

Event: NotificationSentEvent

Se emite al enviar una notificación a un usuario o empresa.

Event: IncidentCreatedEvent

Se emite cuando se genera un incidente a partir de una alerta.

Fábricas (Factories)

- AlertFactory: encapsula la lógica de creación de una alerta a partir de eventos recibidos (ejemplo: sensor fuera de rango).
- IncidentFactory: crea incidentes asociados a un viaje cuando una alerta lo requiere.

4.2.3.2. Interface Layer

En esta capa se definen **Controllers (REST)**.

Controllers (REST — Spring Web)

AlertController

Este controlador permite crear nuevas alertas a partir de eventos detectados, reconocer (ACK) alertas activas, cerrarlas una vez reconocidas, y obtener tanto el detalle de una alerta específica como la lista de alertas activas (estados OPEN o ACKNOWLEDGED).

NotificationController

Su responsabilidad es consultar y actualizar las preferencias de notificación de los usuarios, por ejemplo, los canales permitidos (EMAIL, SMS o FCM) y los tiempos de escalamiento configurados.

IncidentController

Permite crear incidentes vinculados a una alerta y un viaje, y consultar el detalle de incidentes registrados.

4.2.3.3. Application Layer**Command Services**

- AlertCommandService: Ejecuta todos los comandos de las alertas.

Event Services

- OutOfRangeDetectedEvent: maneja eventos de sensores fuera de rango.
- DeviceOfflineDetectedEvent: maneja eventos de desconexión de dispositivos.
- RouteDeviationDetectedEvent: maneja desvíos de ruta.
- AlertAcknowledgedEvent: actúa tras el reconocimiento de una alerta (ejemplo: detener escalamiento).
- AlertClosedEvent: actúa tras el cierre de una alerta (ejemplo: notificar a analíticas).
- TemperatureOutOfRangeEvent: crea alerta de temperatura.
- HumidityOutOfRangeEvent: crea alerta de humedad.
- VibrationDetectedEvent: maneja vibración anómala.
- TiltOrDumpDetectedEvent: maneja vuelcos o inclinaciones.
- LowBatteryDetectedEvent: maneja alerta de energía.

Query Services

- AlertQueryService: Consulta las alertas.

Outbound Services

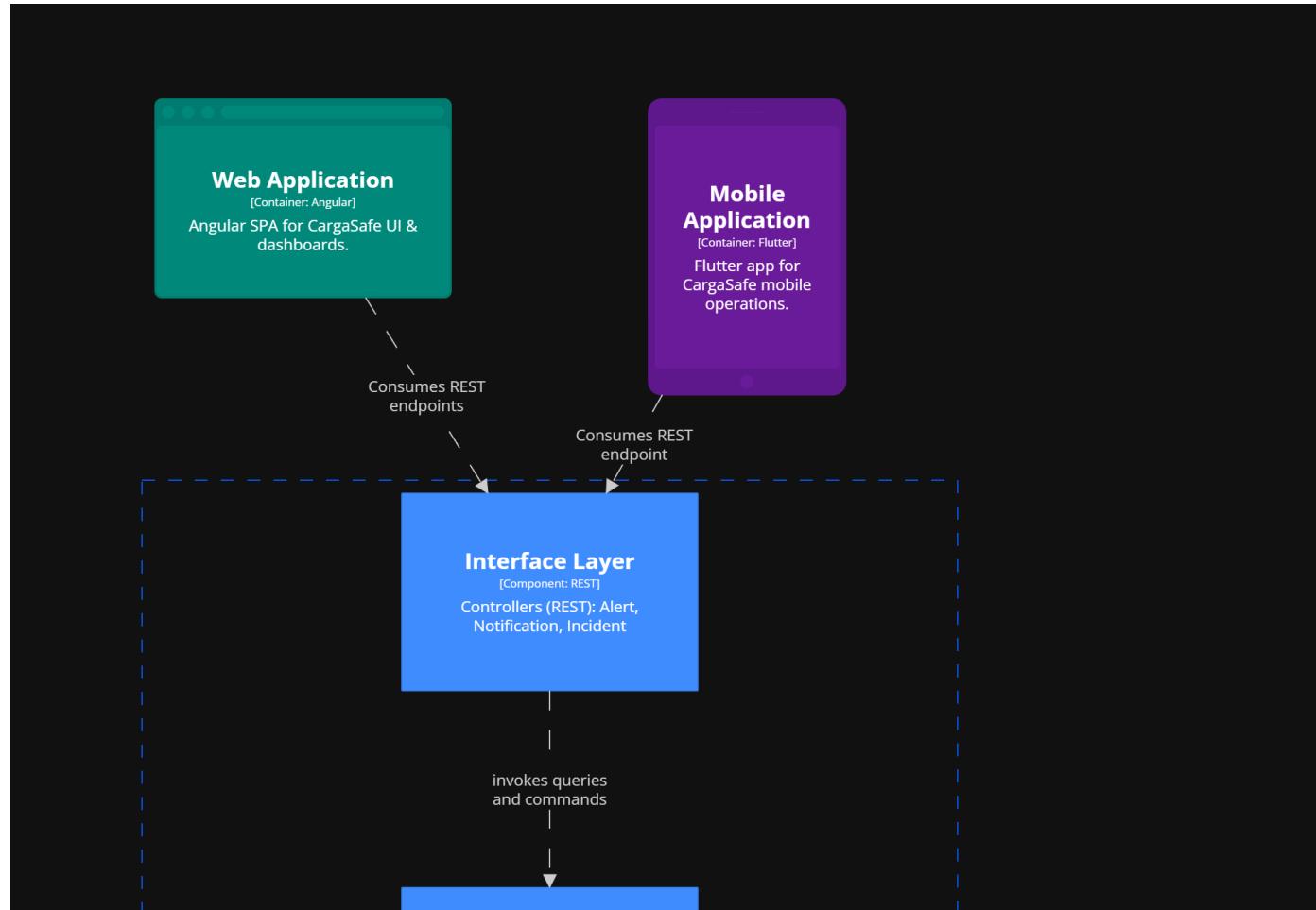
- NotificationService: Servicio para el envío de notificaciones a través de diferentes canales (Email, SMS, Push). Su implementación concreta delega en proveedores externos como Firebase Cloud Messaging (FCM).

4.2.3.4. Infrastructure Layer

- Notification Repository: Repositorio para acceder a las notificaciones.
- Alert Repository: Repositorio para acceder a las alertas.
- Incident Repository: Repositorio para acceder a los incidentes.

4.2.3.5. Bounded Context Software Architecture Component Level Diagrams

Diagrama de componentes - Backend - Alerts & Resolution



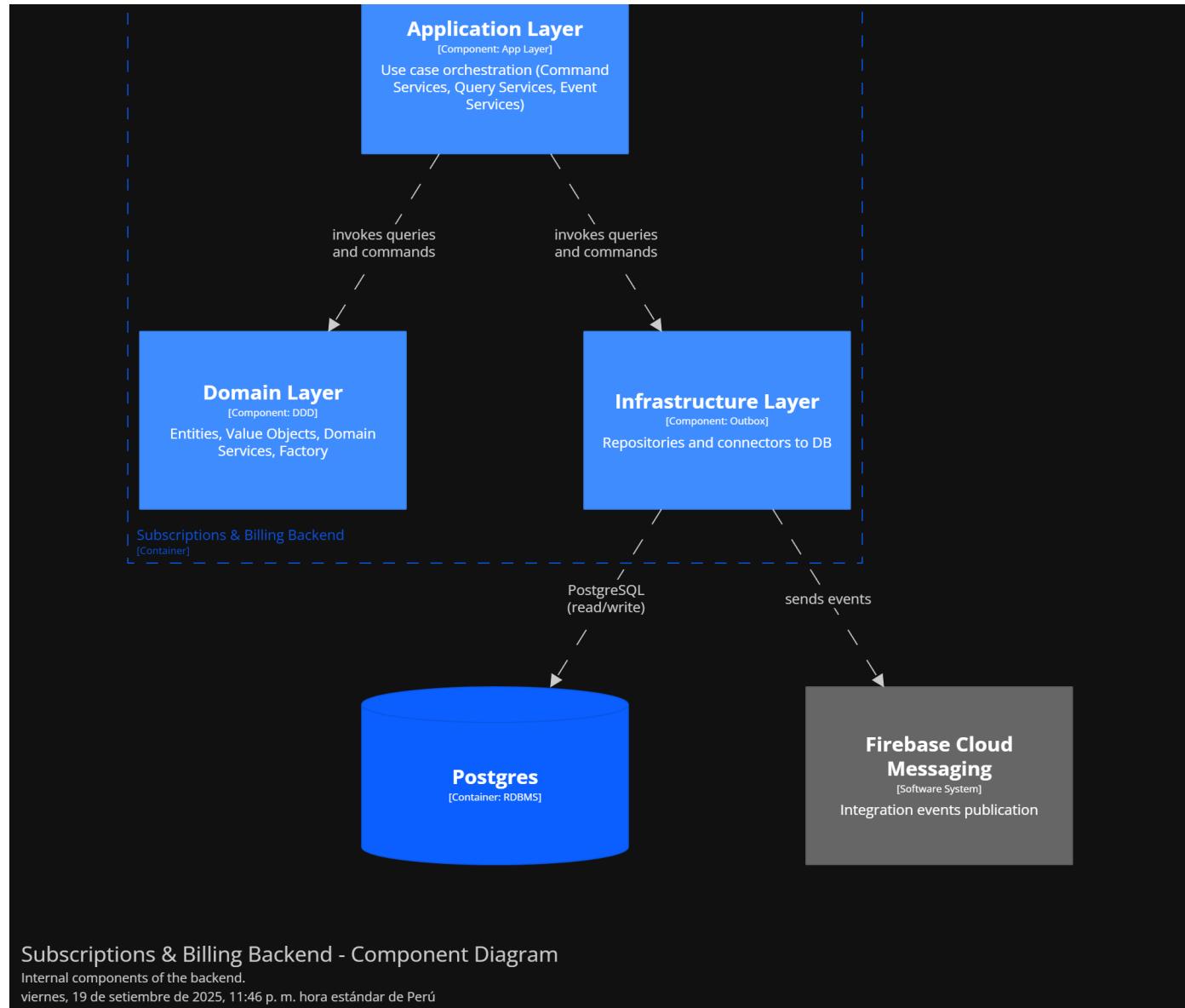


Diagrama de componentes - Application Web - Alerts & Resolution

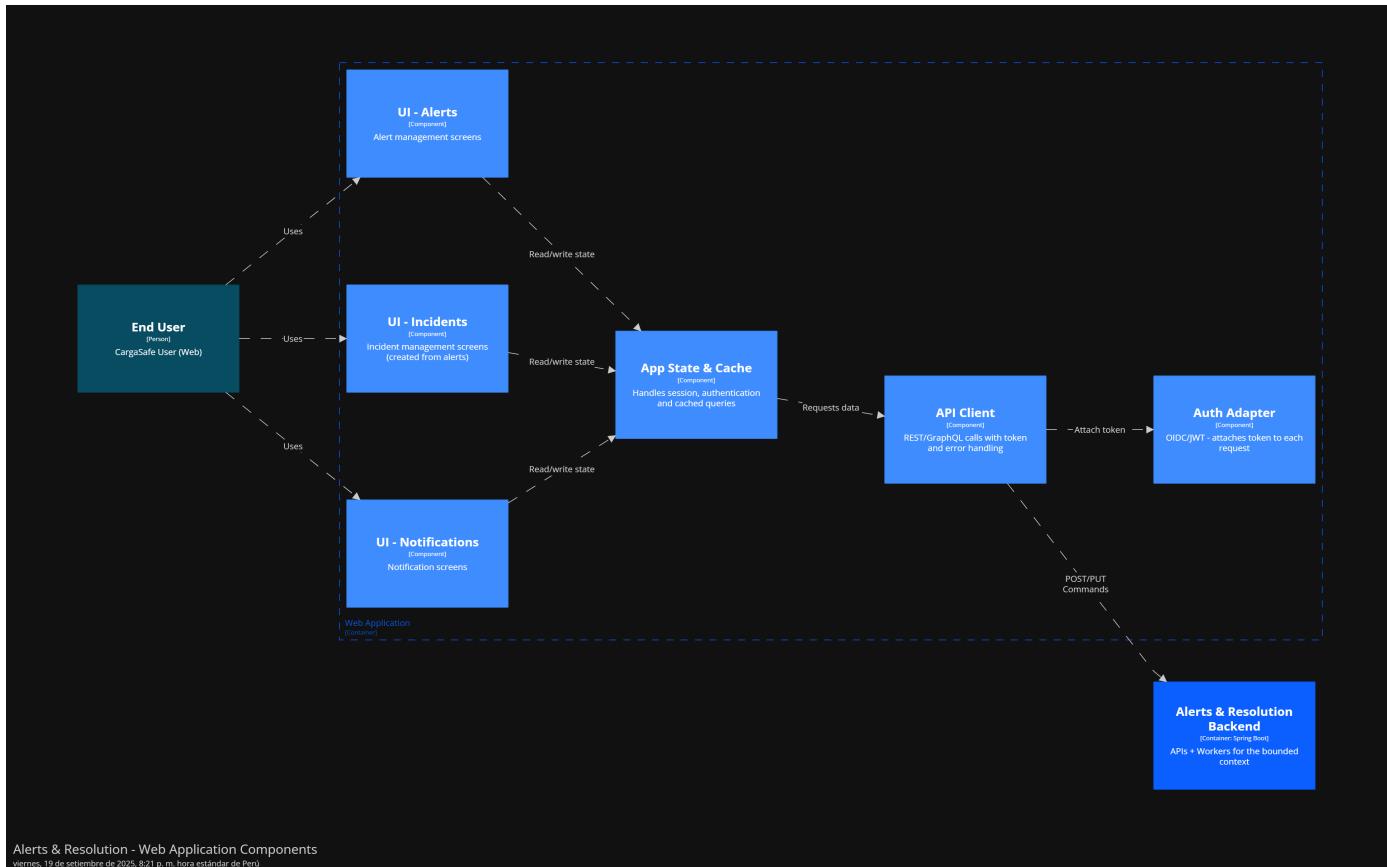
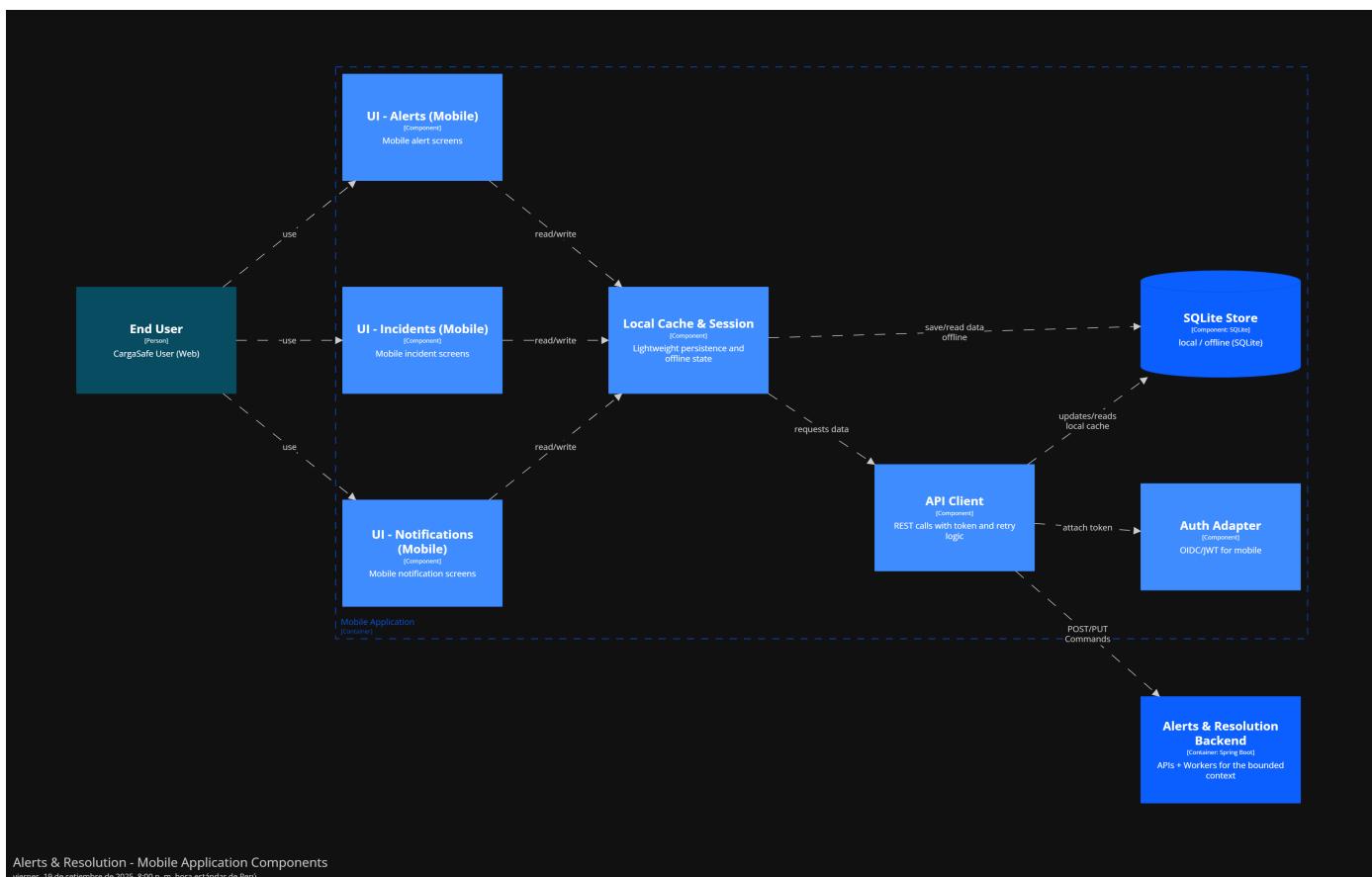


Diagrama de componentes - Mobile App - Alerts & Resolution



4.2.3.6. Bounded Context Software Architecture Code Level Diagrams

4.2.3.6.1. Bounded Context Domain Layer Class Diagrams

Domain Layer Class Diagram

Welcome to PlantUML!

You can start with a simple UML Diagram like:

```
Bob->Alice: Hello
```

Or

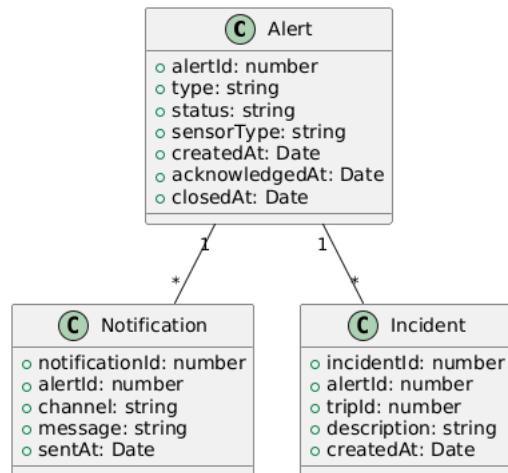
```
class Example
```

You will find more information about PlantUML syntax on <https://plantuml.com>

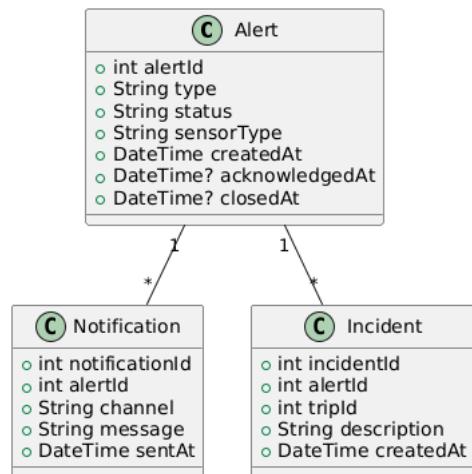
(Details by typing license keyword)



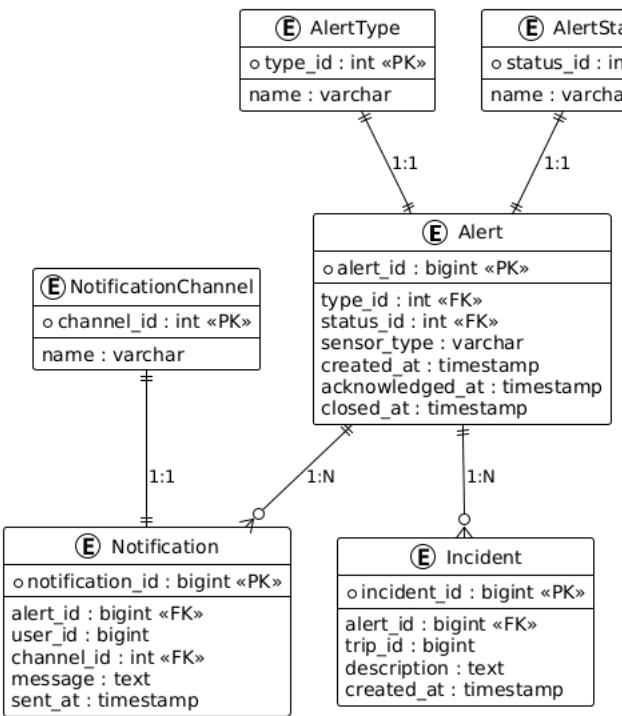
WebApp Class Diagram



Mobile App Class Diagram



4.2.3.6.2. Bounded Context Database Design Diagram



4.2.4. Bounded Context: Real-Time Monitoring

4.2.4.1. Domain Layer.

Entity: MonitoringSession (Aggregate Root)

Propósito principal

Representar una sesión de monitoreo asociada a un viaje y dispositivo, centralizando su ciclo de vida y estado.

Atributos principales

- id: Identificador único de la sesión de monitoreo.
- tripId: Identificador del viaje asociado.
- deviceID: Identificador del dispositivo IoT.
- startTime: Fecha y hora de inicio de la sesión.
- endTime: Fecha y hora de finalización de la sesión (nullable).
- createdAt: Fecha y hora de creación de la sesión.
- sessionStatus: Estado actual de la sesión (relación con `MonitoringSessionStatus`).

Métodos principales

- `startSession()`: Inicia la sesión de monitoreo y cambia su estado a "ACTIVE".
- `endSession()`: Finaliza la sesión de monitoreo y cambia su estado a "COMPLETED".
- `updateStatus(newStatus)`: Cambia el estado de la sesión (ej. ACTIVE → PAUSED).
- `isActive()`: Verifica si la sesión está activa.

Entity: TelemetryData

Propósito principal

Representar una lectura de telemetría tomada durante una sesión de monitoreo.

Atributos principales

- id: Identificador único de la lectura de telemetría.
- monitoringSessionId: Identificador de la sesión de monitoreo asociada.
- temperature: Valor de temperatura registrado.
- humidity: Valor de humedad registrado.
- vibration: Valor de vibración registrado.
- latitude: Latitud geográfica registrada.
- longitude: Longitud geográfica registrada.
- createdAt: Fecha y hora de la lectura.

Métodos principales

- `getTemperature()`: Obtiene el valor de temperatura.
- `getHumidity()`: Obtiene el valor de humedad.
- `getVibration()`: Obtiene el valor de vibración.

- `getLocation()`: Retorna la ubicación (latitud, longitud).
-

Entity: MonitoringSessionStatus

Propósito principal

Representar el estado de una sesión de monitoreo como entidad independiente.

Atributos principales

- `id`: Identificador único del estado.
- `name`: Nombre del estado (ej. ACTIVE, COMPLETED, PAUSED).

Métodos principales

- `isFinal()`: Verifica si el estado es terminal (ej. COMPLETED).
-

Value Object: Location

Propósito principal

Representar una ubicación geográfica precisa.

Atributos principales

- `latitude`: Latitud de la ubicación.
- `longitude`: Longitud de la ubicación.

Métodos principales

- `distanceTo(other)`: Calcula la distancia a otra ubicación.
-

Value Object: SessionStatus

Propósito principal

Encapsular el estado de una sesión en su ciclo de vida.

Atributos principales

- `status`: Valor posible definido en `monitoring_session_status` (ej. ACTIVE, INACTIVE, COMPLETED, PAUSED).
-

Aggregate: MonitoringSessionAggregate

Propósito principal

Garantizar la consistencia de una sesión de monitoreo, agrupando la sesión y sus datos de telemetría relacionados.

Métodos principales

- `addTelemetryData(data)`: Agrega una nueva lectura de telemetría.
 - `validateSessionStatus()`: Verifica que la sesión esté en un estado válido antes de aceptar lecturas.
-

Factory: MonitoringSessionFactory

Propósito principal

Crear instancias válidas de `MonitoringSession`.

Métodos principales

- `createSession(tripId, deviceld)`: Genera una nueva sesión en estado inicial (ej. INACTIVE).
-

Domain Service: TelemetryProcessingService

Propósito principal

Encapsular la lógica de procesamiento de datos de telemetría.

Métodos principales

- `processTelemetry(rawData)`: Procesa datos brutos y crea `TelemetryData`.
 - `validateTelemetry(data)`: Valida integridad y consistencia de los datos.
-

Domain Service: SessionLifecycleService

Propósito principal

Gestionar el ciclo de vida de una sesión de monitoreo.

Métodos principales

- `startSession(session)`: Inicia la sesión.

- `endSession(session)`: Finaliza la sesión.
 - `pauseSession(session)`: Pausa una sesión activa.
 - `resumeSession(session)`: Reanuda una sesión pausada.
-

Command: StartMonitoringSessionCommand**Propósito**

Iniciar una nueva sesión de monitoreo.

Parámetros

- `tripId`
 - `deviceID`
-

Command: EndMonitoringSessionCommand**Propósito**

Finalizar una sesión activa.

Parámetros

- `sessionId`
-

Command: PauseMonitoringSessionCommand**Propósito**

Pausar temporalmente una sesión de monitoreo.

Parámetros

- `sessionId`
-

Command: ResumeMonitoringSessionCommand**Propósito**

Reanudar una sesión previamente pausada.

Parámetros

- `sessionId`
-

Query: GetMonitoringSessionByIdQuery**Propósito**

Obtener información de una sesión específica.

Parámetros

- `sessionId`
-

Query: GetTelemetryDataBySessionQuery**Propósito**

Obtener lecturas de telemetría de una sesión.

Parámetros

- `sessionId`
 - `startTime` (opcional)
 - `endTime` (opcional)
-

Query: GetActiveSessionsQuery**Propósito**

Listar todas las sesiones activas.

Query: GetSessionsByTripIdQuery**Propósito**

Obtener todas las sesiones de un viaje.

Parámetros

- `tripId`

Event: MonitoringSessionStartedEvent**Propósito**

Notificar inicio de una sesión.

Parámetros

- sessionId
 - tripId
 - deviceID
 - startedAt
-

Event: MonitoringSessionCompletedEvent**Propósito**

Notificar finalización de una sesión.

Parámetros

- sessionId
 - completedAt
-

Event: TelemetryDataReceivedEvent**Propósito**

Notificar recepción de una lectura de telemetría.

Parámetros

- sessionId
 - telemetryData
 - receivedAt
-

4.2.4.2. Interface Layer.**Controllers**

- **MonitoringController**: Maneja solicitudes de inicio, fin, pausa, reanudación y consulta de sesiones.
 - **TelemetryController**: Maneja solicitudes relacionadas con datos de telemetría (consultas, históricos, gráficos).
-

4.2.4.3. Application Layer.**Command Services**

- **MonitoringCommandService**: Coordina comandos relacionados a las sesiones.
 - **TelemetryCommandService**: Coordina el procesamiento de datos de telemetría.
-

Query Services

- **MonitoringQueryService**: Atiende consultas sobre sesiones.
 - **TelemetryQueryService**: Atiende consultas sobre lecturas de telemetría.
-

Event Services

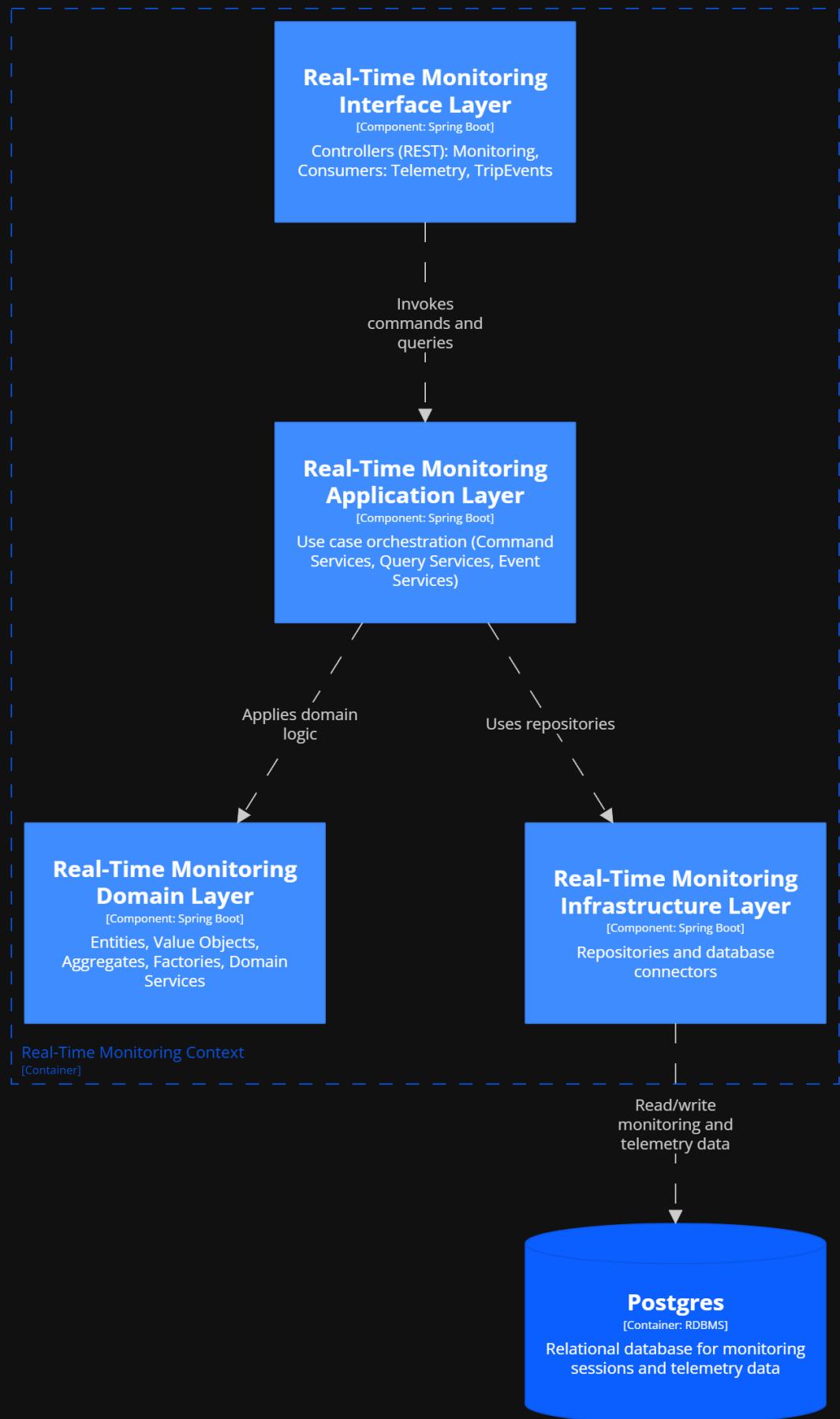
- **MonitoringEventService**: Atiende eventos de sesiones (inicio, fin, recepción de telemetría).
-

4.2.4.4. Infrastructure Layer.**Repositories**

- **IMonitoringSessionRepository**: Persistencia de sesiones.
- **ITelemetryDataRepository**: Persistencia y consulta de lecturas de telemetría.
- **ISessionStatusRepository**: Acceso a los estados posibles de sesión. Reintentar Claude aún no tiene la capacidad de ejecutar el código que genera.

4.2.4.5. Bounded Context Software Architecture Component Level Diagrams

Diagrama de componentes - Backend - Real-Time Monitoring

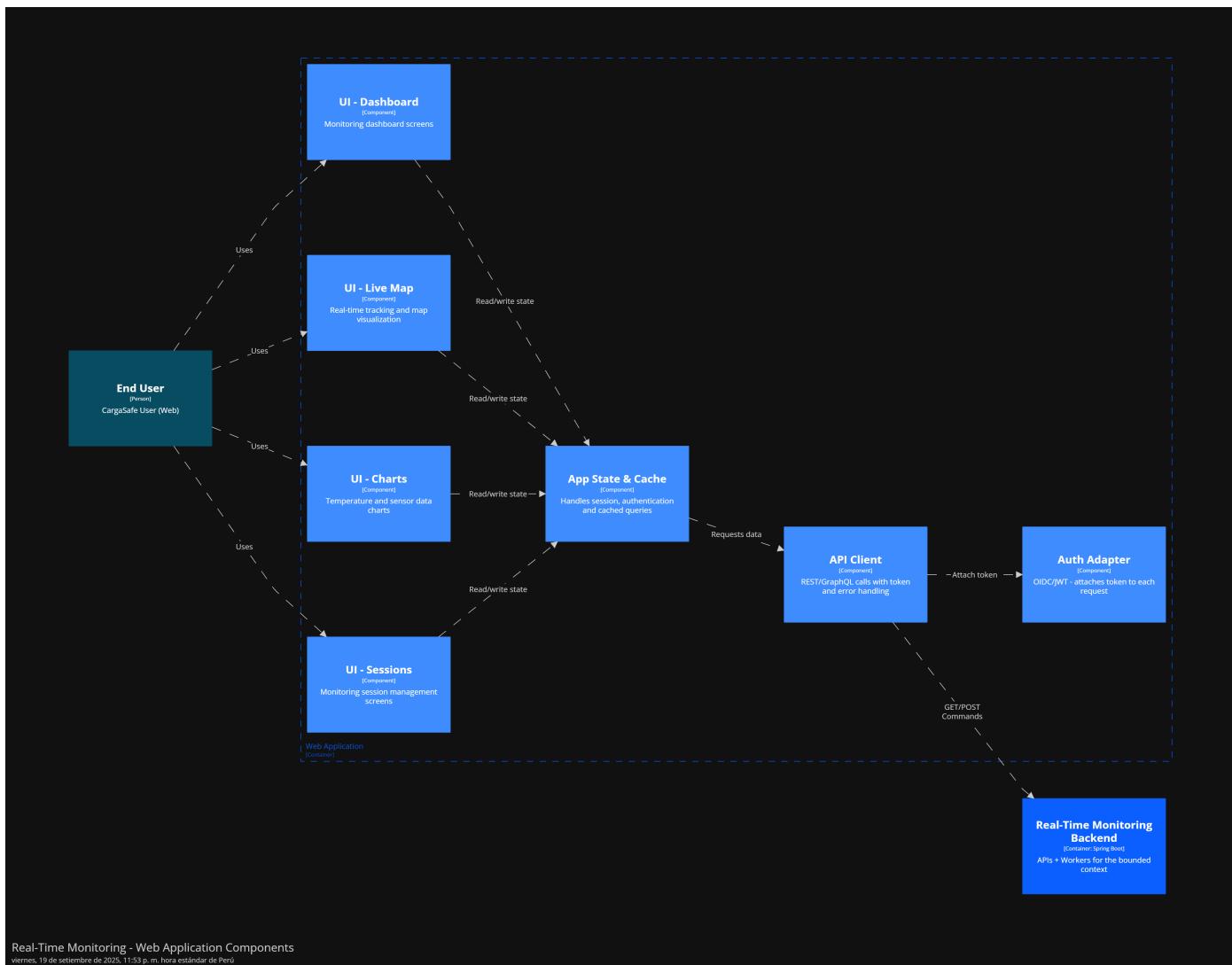


Real-Time Monitoring Context - Component Diagram

Internal layered components of the Real-Time Monitoring Context inside CargoSafe

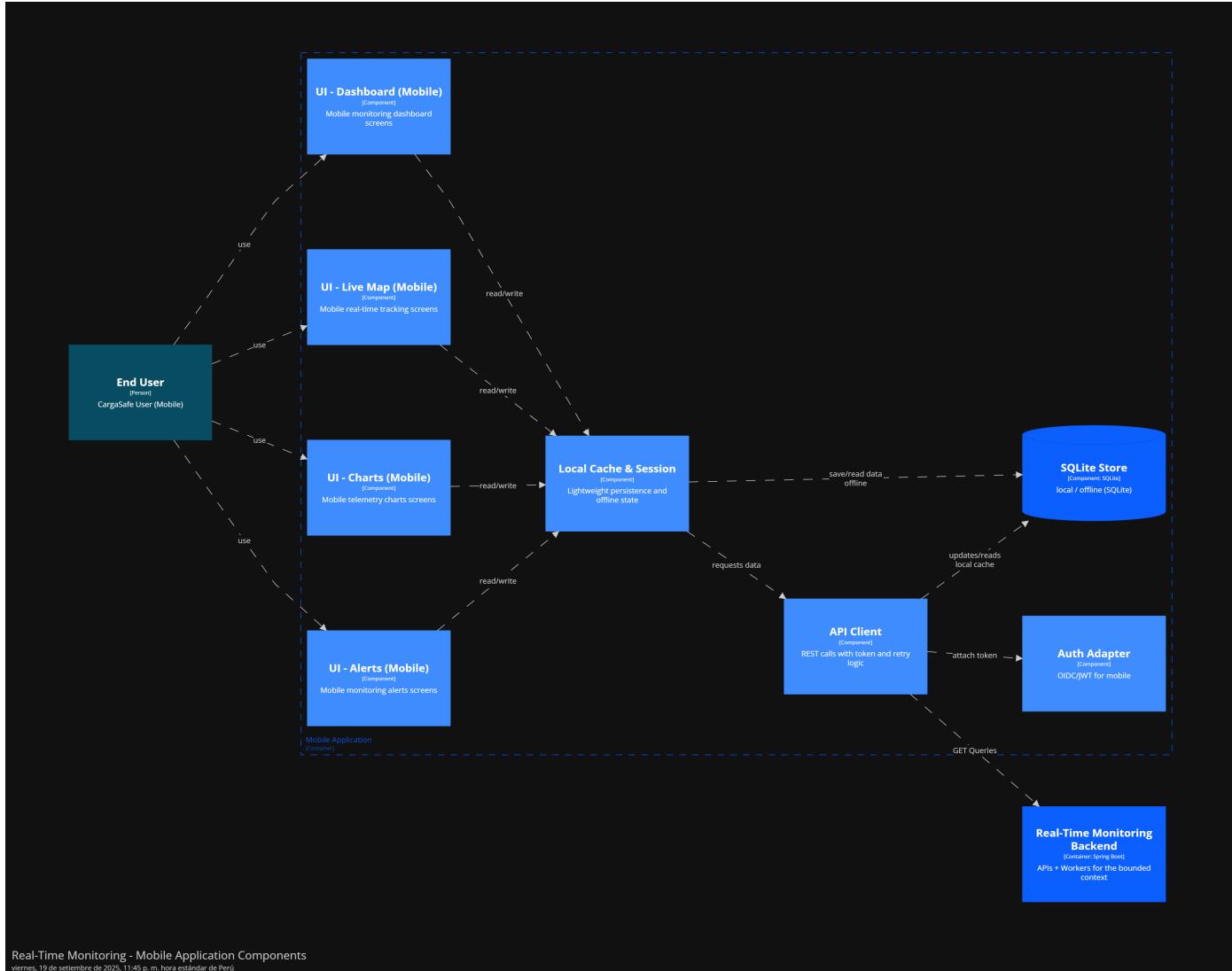
Internal layered components of the real-time monitoring context inside Cargasafe.
viernes, 19 de setiembre de 2025, 11:43 p. m. hora estándar de Perú

Diagrama de componentes - Application Web - Real-Time Monitoring



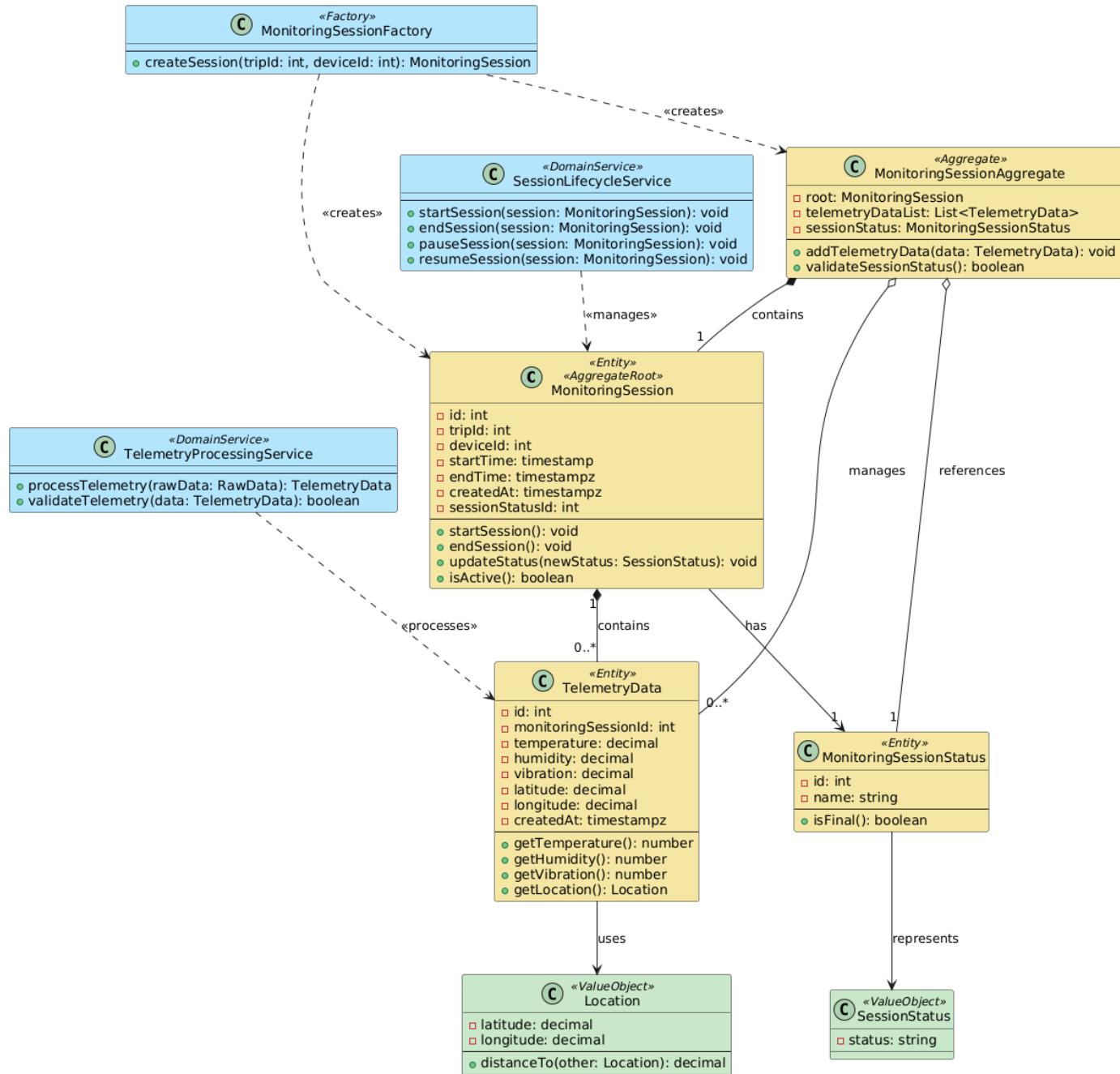
Real-Time Monitoring - Web Application Components
Viernes, 19 de setiembre de 2025, 11:53 p. m. hora estándar de Perú

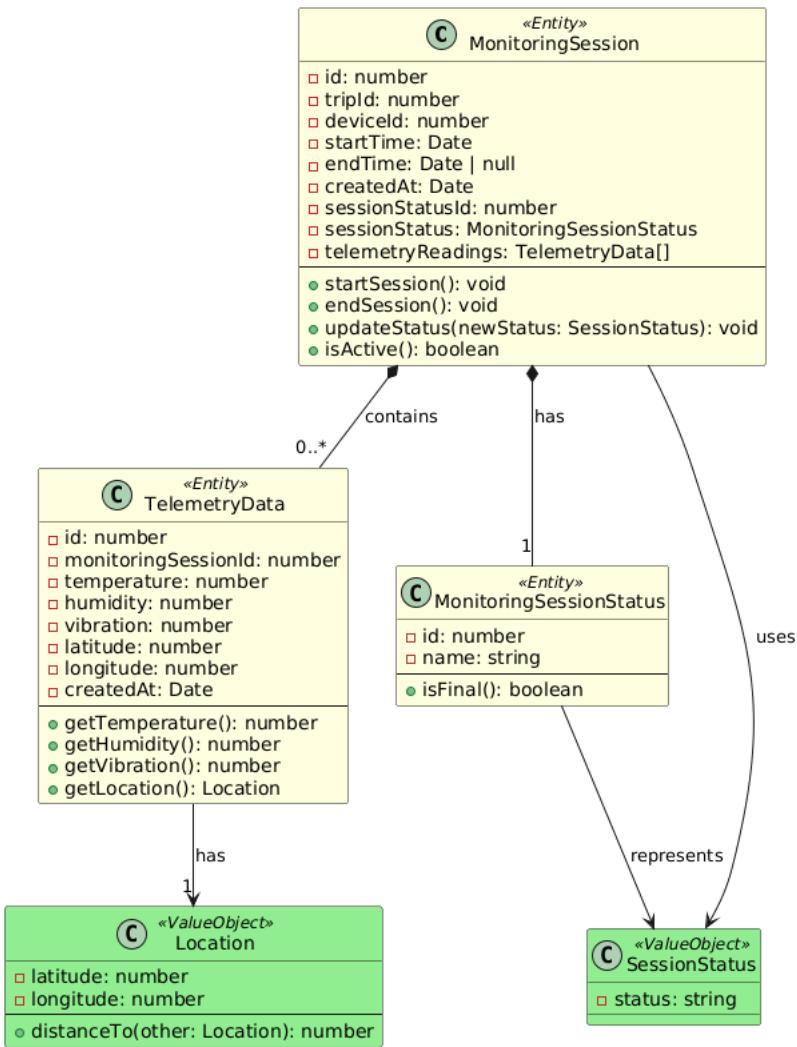
Diagrama de componentes - Mobile App - Real-Time Monitoring



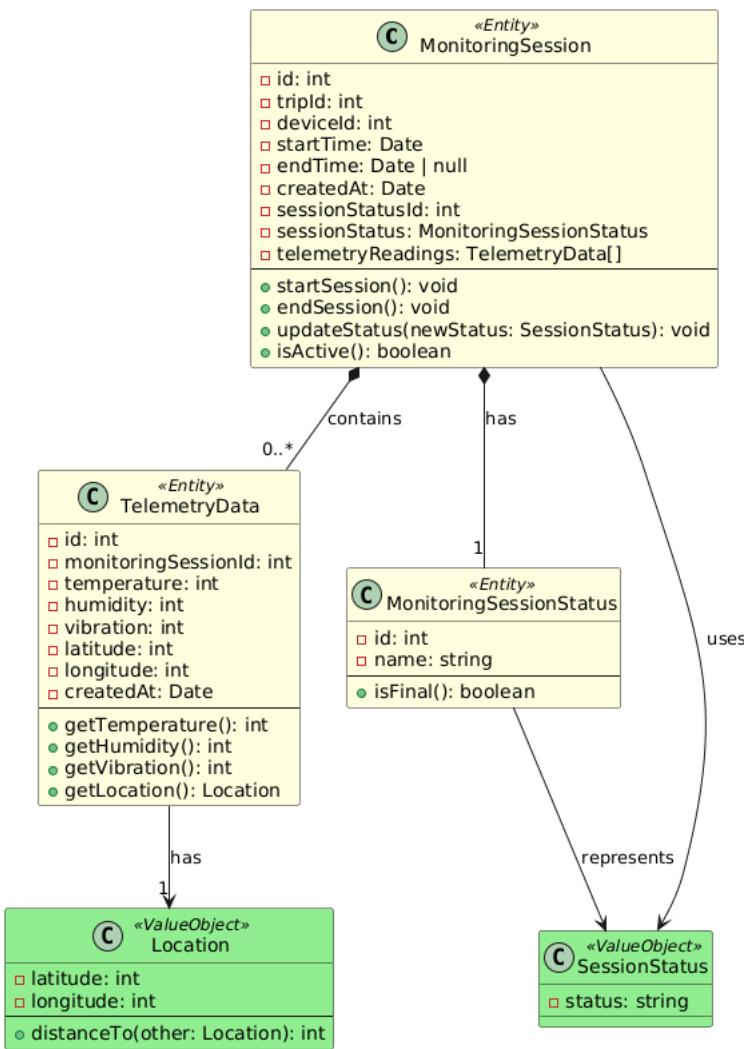
4.2.4.6. Bounded Context Software Architecture Code Level Diagrams

4.2.4.6.1. Bounded Context Domain Layer Class Diagrams

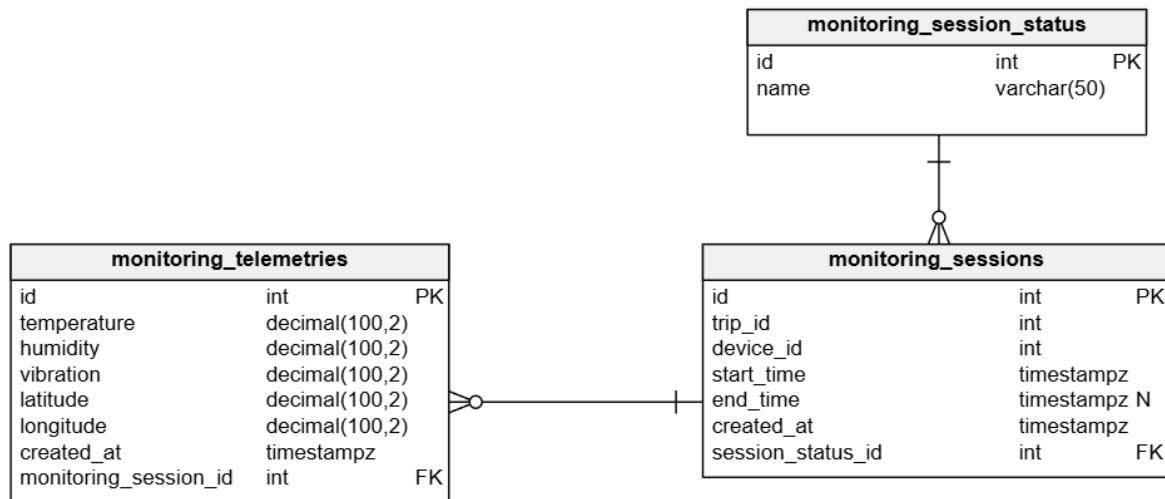
**WebApp Class Diagram**



Mobile Class Diagram



4.2.4.6.2. Bounded Context Database Design Diagram



Real time monitoring

4.2.5. Bounded Context: Trip management

4.2.5.1. Domain Layer.

Entity: Trip (Aggregate Root)

Propósito principal

Representar un viaje y centralizar su ciclo de vida, asegurando que se cumplan las reglas de negocio relacionadas con cliente, conductor, vehículo y ruta.

Atributos principales

- tripId: Identificador único del viaje.
- clientId: Identificador del cliente.
- driverId: Identificador del conductor.
- vehicleId: Identificador del vehículo.
- route: Ruta definida para el trayecto.
- status: Estado del viaje (CREATED, IN_PROGRESS, COMPLETED, CANCELLED).
- requestedAt: Fecha y hora de la solicitud.

Métodos principales

- assignDriver(driverId): Asigna un conductor al viaje.
- assignVehicle(vehicleId): Vincula un vehículo al viaje.
- startTrip(): Inicia el viaje y cambia su estado a "En curso".
- completeTrip(): Finaliza el viaje y cambia su estado a "Completado".
- cancelTrip(reason): Cancela el viaje y registra el motivo.

Entity: Route

Propósito principal

Representar la ruta de un viaje como una entidad con identidad propia, capaz de almacenar y gestionar la información de los tramos, distancias y duración total.

Atributos principales

- routeId: Identificador único de la ruta.
- origin: Punto de inicio.
- destination: Punto final.
- segments: Lista de tramos de la ruta.
- totalDistance: Distancia total del viaje.
- totalDuration: Duración total estimada.

Métodos principales

- addSegment(segment): Agrega un tramo adicional a la ruta.
- updateDestination(newDestination): Cambia el destino de la ruta antes de iniciar el viaje.
- recalculateTotals(): Recalcula la distancia y la duración total a partir de los segmentos actuales.

Value Object: GeoCoordinate

Propósito principal

Representar un punto geográfico inmutable.

Atributos principales

- latitude: Latitud válida.
- longitude: Longitud válida.

Value Object: RouteSegment

Propósito principal

Modelar un tramo de ruta entre dos puntos.

Atributos principales

- coordinates: Lista de coordenadas que forman el tramo.
- distance: Distancia recorrida en el segmento.
- duration: Tiempo estimado del segmento.

Value Object: Distance

Propósito principal

Expresar una magnitud de distancia.

Atributos principales

- value: Cantidad numérica de la distancia.
- unit: Unidad de medida (ej. km).

Value Object: Duration

Propósito principal

Expresar un intervalo de tiempo.

Atributos principales

- value: Cantidad numérica de tiempo.
 - unit: Unidad de medida (ej. minutos).
-

Value Object: TripStatus

Propósito principal

Representar el estado del viaje en su ciclo de vida.

Atributos principales

- status: Valor posible (PENDING, IN_PROGRESS, COMPLETED, CANCELLED).
-

Aggregate: TripAggregate

Propósito principal

Asegurar la consistencia de un viaje como unidad de negocio.

Métodos principales

- validateTripReady(): Verifica que el viaje tenga cliente, conductor, vehículo y ruta antes de iniciar.
-

Factory: TripFactory

Propósito principal

Crear instancias de **Trip** en estado inicial válido.

Métodos principales

- createTrip(clientId, driverId, vehicleId, route): Genera un viaje en estado PENDING con todos los datos requeridos.
-

Domain Service: RoutePlanningService

Propósito principal

Encapsular la lógica de planificación de rutas.

Métodos principales

- generateRoute(origin, destination): Construye una ruta válida con segmentos, distancia y duración.
-

Domain Service: TripSchedulerService

Propósito principal

Validar disponibilidad de recursos antes de asignarlos a un viaje.

Métodos principales

- checkDriverAvailability(driverId, timeRange): Verifica si un conductor está libre.
 - checkVehicleAvailability(vehicleId, timeRange): Verifica si un vehículo está disponible.
-

Command: CreateTripCommand

Propósito

Crear un nuevo viaje en estado PENDING con las referencias de cliente, conductor, vehículo y ruta.

Parámetros

- clientId: Identificador del cliente.
 - driverId: Identificador del conductor.
 - vehicleId: Identificador del vehículo.
 - route: Ruta completa del viaje.
-

Command: AssignDriverToTripCommand

Propósito

Asignar un conductor disponible a un viaje existente y actualizar la referencia correspondiente.

Parámetros

- tripId: Identificador único del viaje.
 - driverId: Identificador del conductor.
-

Command: AssignVehicleToTripCommand

Propósito

Asignar un vehículo disponible a un viaje existente y actualizar la referencia correspondiente.

Parámetros

- tripId: Identificador único del viaje.
 - vehicleId: Identificador del vehículo.
-

Command: StartTripCommand**Propósito**

Iniciar un viaje, cambiando su estado a EN CURSO y registrando la hora exacta de inicio.

Parámetros

- tripId: Identificador único del viaje.
-

Command: CompleteTripCommand**Propósito**

Finalizar un viaje, cambiando su estado a COMPLETADO y registrando la hora de cierre.

Parámetros

- tripId: Identificador único del viaje.
-

Command: CancelTripCommand**Propósito**

Cancelar un viaje, actualizar su estado a CANCELADO y guardar la razón de la cancelación.

Parámetros

- tripId: Identificador único del viaje.
 - reason: Motivo de la cancelación.
-

Command: UpdateRouteForTripCommand**Propósito**

Actualizar la ruta de un viaje antes de que inicie, garantizando que la información sea válida y actualizada.

Parámetros

- tripId: Identificador único del viaje.
 - newRoute: Nueva ruta a asociar.
-

Query: GetTripByIdQuery**Propósito**

Obtener la información completa de un viaje específico mediante su identificador único.

Parámetros

- tripId: Identificador único del viaje.
-

Query: GetTripsByStatusQuery**Propósito**

Listar los viajes filtrados por su estado (Pendiente, En curso, Completado o Cancelado).

Parámetros

- status: Estado de los viajes a consultar.
-

Query: GetTripsByClientIdQuery**Propósito**

Obtener todos los viajes asociados a un cliente específico.

Parámetros

- clientId: Identificador único del cliente.
-

Query: GetAllTripsQuery

Propósito

Recuperar todos los viajes registrados en el sistema, sin aplicar filtros.

Parámetros

(No requiere parámetros)

Event: TripCreatedEvent**Propósito**

Notificar que un nuevo viaje ha sido creado en el sistema.

Parámetros

- tripId: Identificador único del viaje.
 - clientId: Identificador del cliente.
 - driverId: Identificador del conductor asignado.
 - vehicleId: Identificador del vehículo asignado.
 - route: Ruta definida para el viaje.
 - createdAt: Fecha y hora en que se creó el viaje.
-

Event: DriverAssignedEvent**Propósito**

Notificar que un conductor fue asignado a un viaje.

Parámetros

- tripId: Identificador único del viaje.
 - driverId: Identificador del conductor asignado.
 - assignedAt: Fecha y hora de la asignación.
-

Event: VehicleAssignedEvent**Propósito**

Notificar que un vehículo fue asignado a un viaje.

Parámetros

- tripId: Identificador único del viaje.
 - vehicleId: Identificador del vehículo asignado.
 - assignedAt: Fecha y hora de la asignación.
-

Event: TripStartedEvent**Propósito**

Notificar que un viaje ha iniciado oficialmente.

Parámetros

- tripId: Identificador único del viaje.
 - startedAt: Fecha y hora de inicio del viaje.
-

Event: TripCompletedEvent**Propósito**

Notificar que un viaje se ha completado satisfactoriamente.

Parámetros

- tripId: Identificador único del viaje.
 - completedAt: Fecha y hora de finalización del viaje.
-

Event: TripCancelledEvent**Propósito**

Notificar que un viaje ha sido cancelado.

Parámetros

- tripId: Identificador único del viaje.
 - reason: Motivo de la cancelación.
 - cancelledAt: Fecha y hora en que se canceló el viaje.
-

4.2.5.2. Interface Layer.

Controllers

- TripController: Controlador que maneja las solicitudes relacionadas con los viajes. Atiende operaciones como crear un nuevo viaje, asignar un conductor, actualizar la ruta, iniciar, completar o cancelar un viaje, así como consultar información de viajes por identificador, estado, cliente o recuperar todos los viajes registrados.
- RouteController: Controlador que maneja las solicitudes relacionadas con las rutas de los viajes. Permite registrar una nueva ruta, actualizarla antes del inicio de un viaje y consultar la información de rutas específicas o asociadas a un viaje.

4.2.5.3. Application Layer.

Command Services

- TripCommandService: Se encarga de recibir y coordinar los comandos relacionados a un viaje. Dentro de él se manejan distintos handlers, cada uno especializado en ejecutar un comando específico como iniciar, completar, cancelar o asignar recursos al viaje.
- RouteCommandService: Se encarga de coordinar los comandos relacionados con rutas. Administra la creación, actualización y recálculo de rutas para garantizar que los trayectos estén completos y actualizados antes de iniciar un viaje.

Query Services

- TripQueryService: Se encarga de atender las consultas relacionadas a los viajes. Contiene handlers que procesan queries para obtener información, por ejemplo: consultar un viaje por su identificador, listar viajes por estado o recuperar todos los viajes de un cliente.
- RouteQueryService: Atiende las consultas relacionadas a las rutas de los viajes. Permite obtener información de rutas específicas o de las rutas asociadas a un viaje.

Event Services

- TripEventService: Se encarga de atender los eventos relacionados a un viaje. Dentro de él se gestionan distintos servicios especializados que reaccionan a cada evento, como creación, asignación de recursos, inicio, finalización o cancelación del viaje, ejecutando las acciones necesarias después de que ocurren.

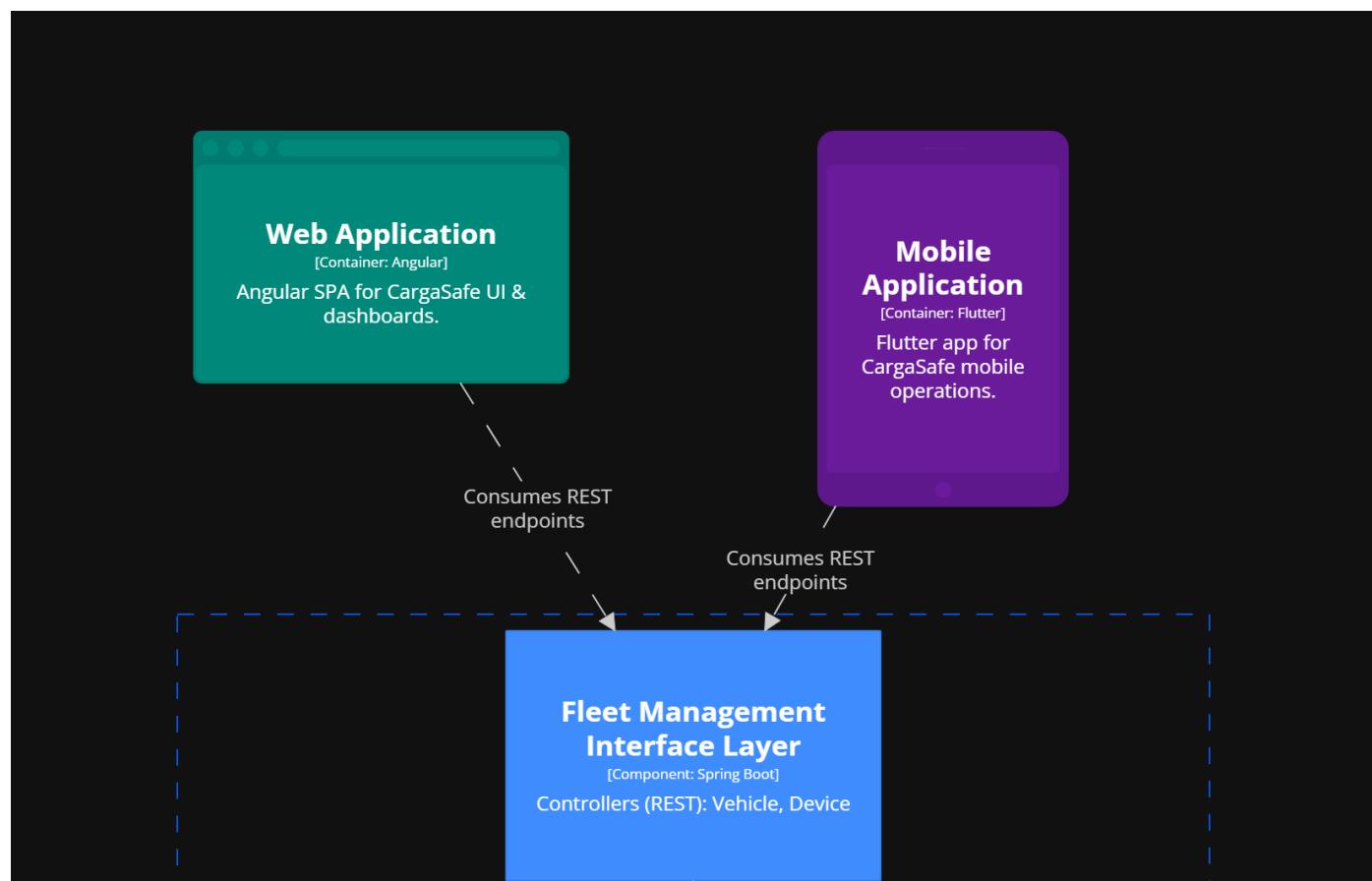
4.2.5.4. Infrastructure Layer.

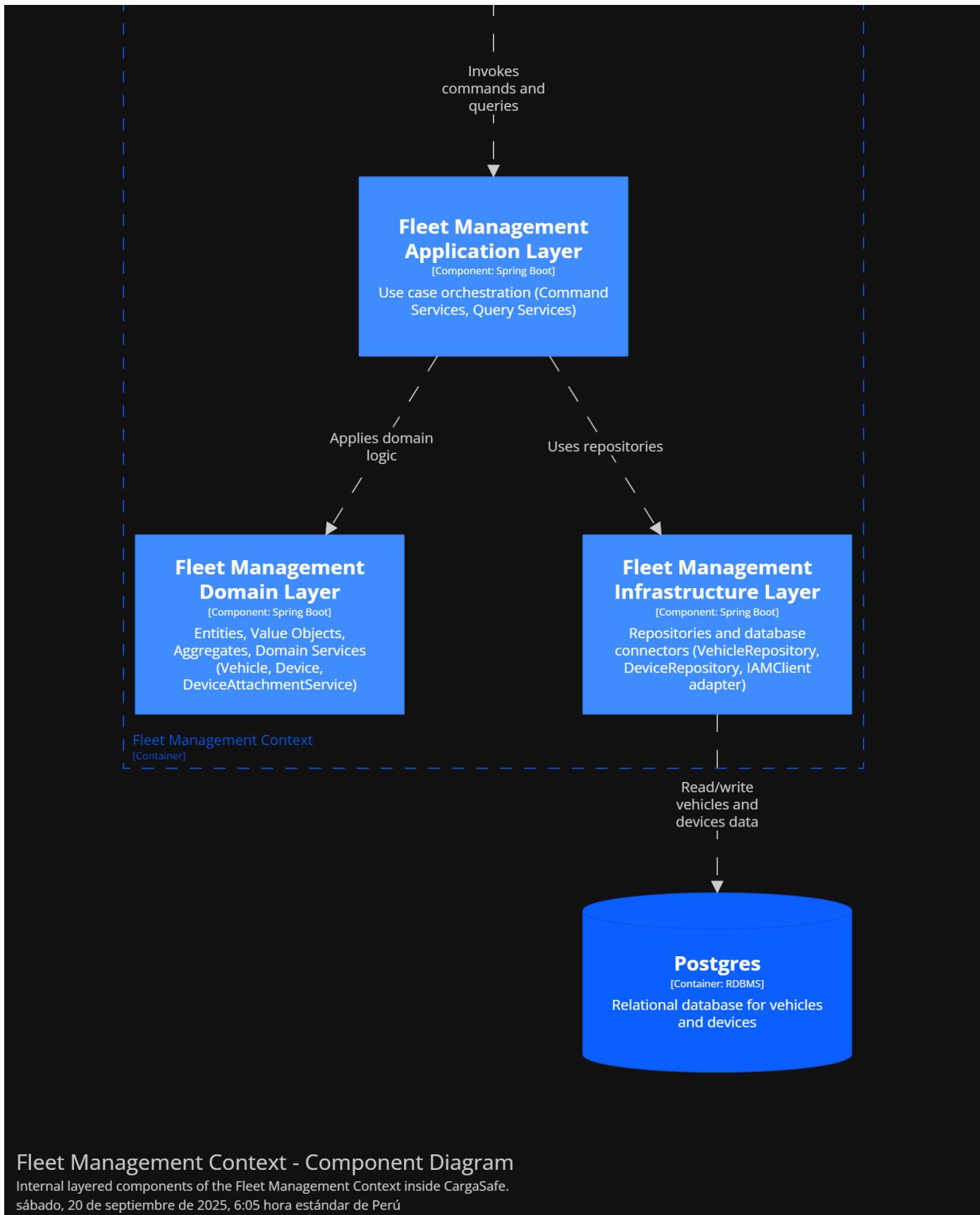
Repositories

- ITripRepository: Repositorio que define las operaciones de acceso a los viajes, como guardar, actualizar y recuperar información de un viaje.
- IRouteRepository: Repositorio que define las operaciones de acceso a las rutas, como registrar nuevas rutas, actualizarlas y consultarlas en relación con un viaje.

4.2.5.5. Bounded Context Software Architecture Component Level Diagrams.

Diagrama de componentes - Backend - Trip Management





Fleet Management Context - Component Diagram

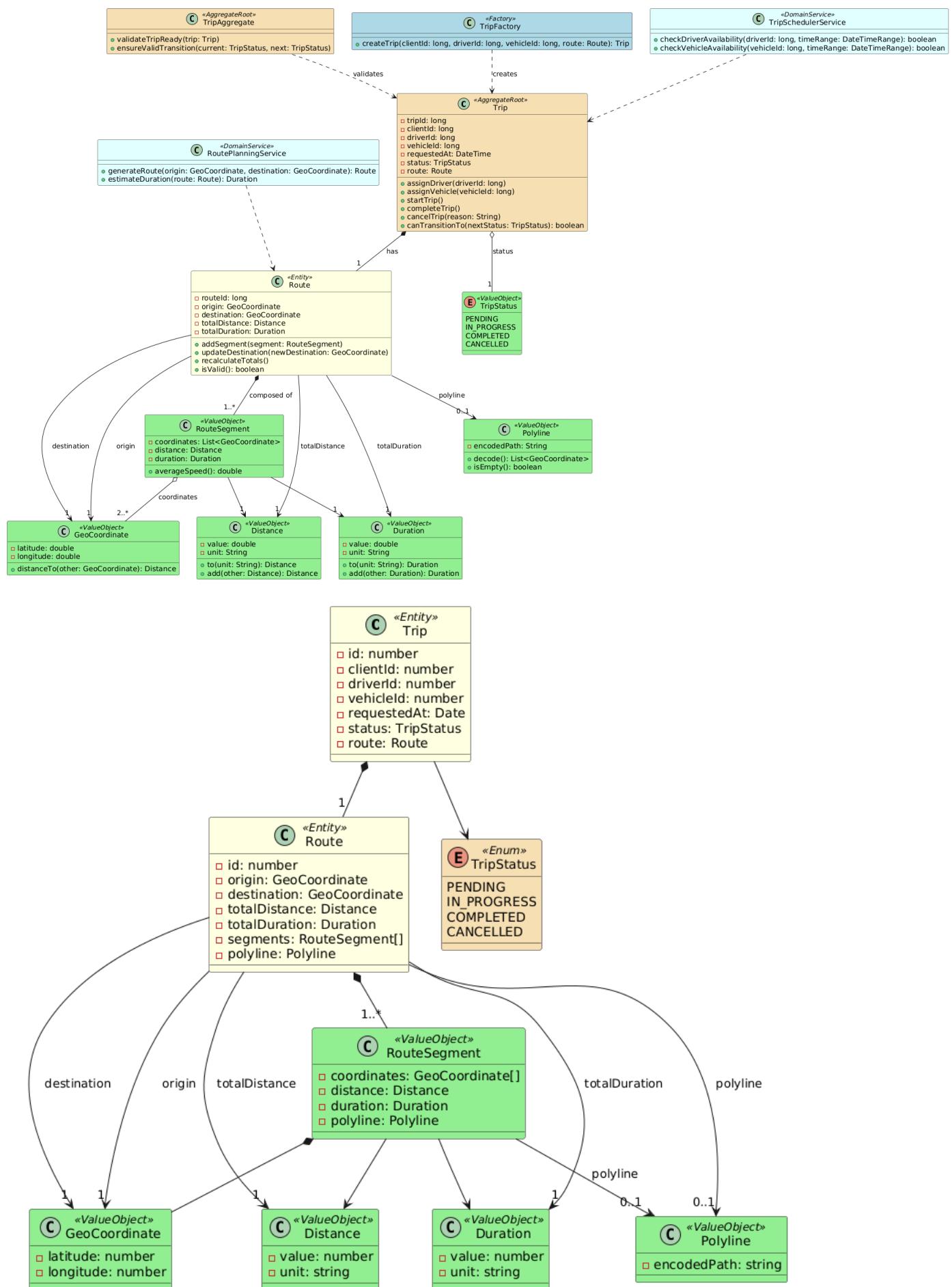
Internal layered components of the Fleet Management Context inside CargaSafe.
sábado, 20 de septiembre de 2025, 6:05 hora estándar de Perú

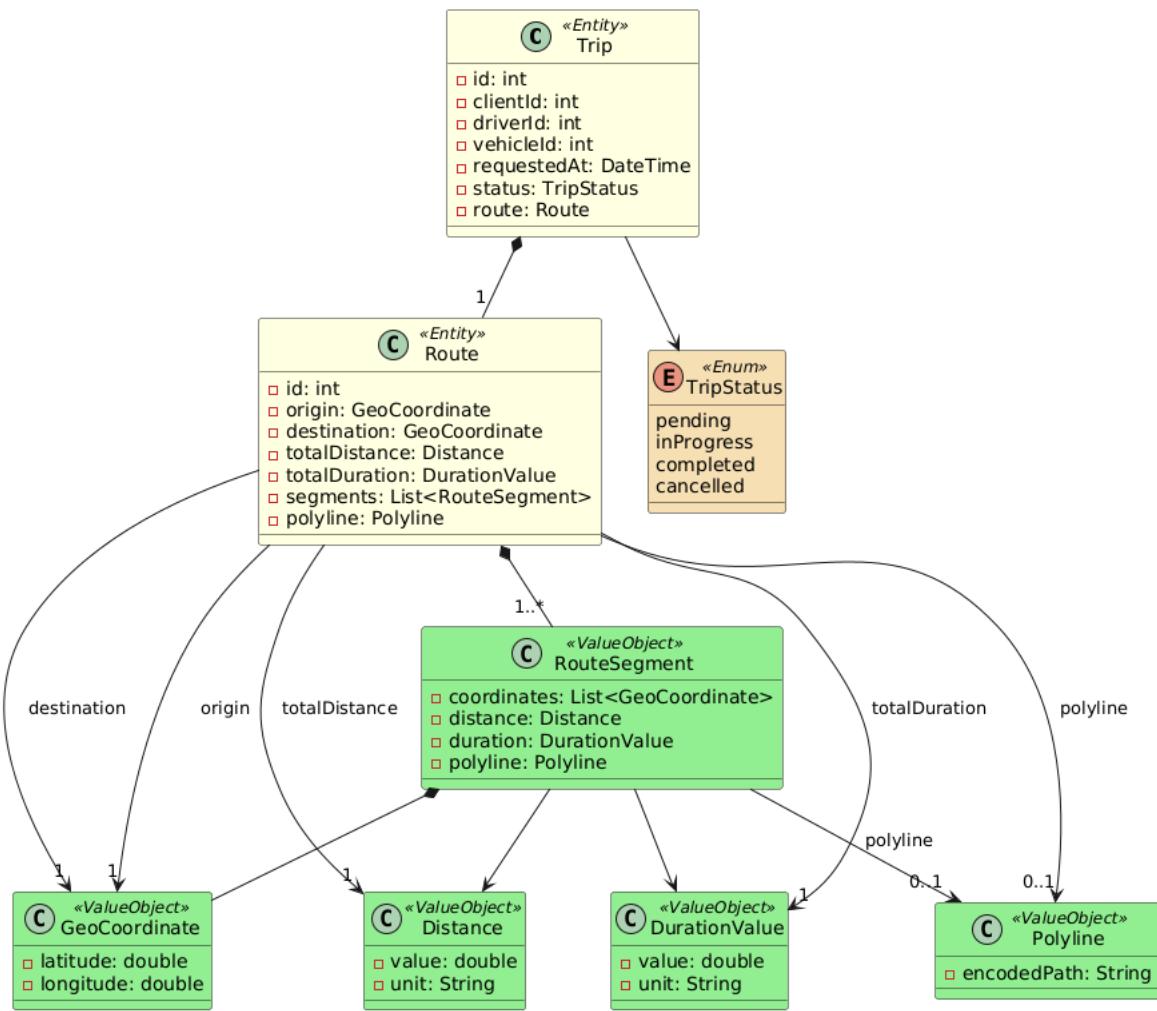
Diagrama de componentes - Application Web - Trip Management

Diagrama de componentes - Mobile App - Trip Management

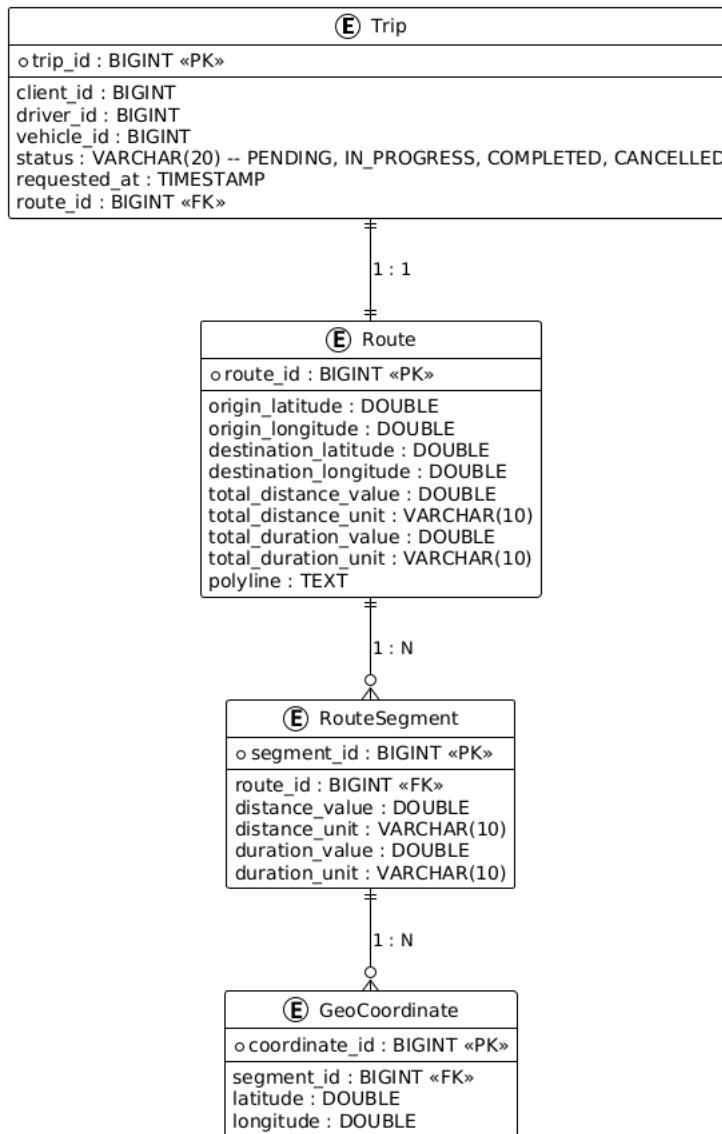
4.2.5.6. Bounded Context Software Architecture Code Level Diagrams.

4.2.5.6.1. Bounded Context Domain Layer Class Diagrams.





4.2.5.6.2. Bounded Context Database Design Diagram.



4.2.6. Bounded Context: Fleet Management

4.2.6.1. Domain Layer

Aggregates principales

Vehicle (Aggregate Root)

- Propósito:** Representa un vehículo de la flota que puede ser monitoreado, recibir dispositivos IoT y participar en viajes.

- Atributos principales:**

- `id` (Long, generado por JPA)
 - `plate` (`Plate`, único)
 - `type` (`VehicleType` = `VAN`, `TRUCK`, `CAR`, `MOTORCYCLE`)
 - `capabilities` (`Set<Capability>` = `BOX`, `REFRIGERATED`, `GPS_ONLY`, `HEAVY_LOAD`, `FRAGILE_CARGO`)
 - `status` (`VehicleStatus` = `IN_SERVICE`, `OUT_OF_SERVICE`, `MAINTENANCE`, `RETIRIED`)
 - `odometerKm` (`OdometerKm`, entero ≥ 0)
 - `deviceImeis` (`Set<Imei>`): lista de IMEIs de dispositivos asignados
 - Campos de auditoría heredados de `AuditableAbstractAggregateRoot` (fechas, etc.)

- Reglas de negocio / métodos relevantes:**

- `Vehicle(CreateVehicleCommand)`
Construye el agregado garantizando:
 - `plate` no vacío y con patrón válido.
 - `type` y `capabilities` no nulos.
 - `capabilities` no vacío.
 - `odometerKm` ≥ 0 .
 - `status` por defecto: `IN_SERVICE` si no se envía.
 - `updateType(VehicleType type)`
Actualiza el tipo del vehículo.

- `updateCapabilities(Set<Capability> capabilities)`
Reemplaza el conjunto de capacidades (si no viene vacío).
 - `updateStatus(VehicleStatus status)`
No permite cambiar desde `RETIRED` a cualquier otro estado (lanza `IllegalStateException`).
 - `updateOdometer(Integer newOdometerKm)`
Verifica que el nuevo valor:
 - No sea negativo.
 - No sea menor que el valor actual (no se permite "retroceder" el odómetro).
 - `assignDevice(Imei deviceImei)`
 - No permite asignar si `status == RETIRED`.
 - No permite asignar el mismo IMEI dos veces al mismo vehículo.
 - `unassignDevice(Imei deviceImei)`
Elimina el IMEI del set de dispositivos asociados.
 - `hasAnyDevice()`
Devuelve `true` si tiene al menos un dispositivo asignado (se usa para restringir el borrado).
 - `hasDevice(Imei deviceImei)`
Verifica si un IMEI concreto está asociado al vehículo.
-

Device (Aggregate Root)

- **Propósito:** Representa un dispositivo IoT asociado (o no) a un vehículo de la flota.

- **Atributos principales:**

- `id` (Long, generado por JPA)
- `imei` (Imei, único)
- `firmware` (FirmwareVersion)
- `online` (boolean)
- `vehiclePlate` (Plate opcional): placa del vehículo al que está asignado o `null`
- Campos de auditoría vía `AuditableAbstractAggregateRoot`

- **Reglas de negocio / métodos relevantes:**

- `Device(CreateDeviceCommand)`
 - `imei` obligatorio y con patrón válido.
 - `firmware` obligatorio y con patrón válido.
 - `online`: si no se envía, se inicializa como `false`.
 - `updateFirmware(FirmwareVersion firmware)`
Simplifica actualizaciones de versión de firmware.
 - `updateOnline(boolean online)`
Actualiza el estado en línea del dispositivo.
 - `assignToVehicle(Plate vehiclePlate)`
 - No permite reasignar si ya tiene una placa (`IllegalStateException("Device is already assigned to a vehicle")`).
 - Guarda la placa del vehículo al que queda asociado.
 - `unassignFromVehicle()`
Elimina la placa asociada.
 - `isAssigned()`
Indica si el dispositivo está asociado a un vehículo (`vehiclePlate != null`).
-

Value Objects

- `Plate`
 - Normaliza en mayúsculas.
 - Valida patrón `[A-Z0-9-]{4,12}`.
 - No permite cadenas en blanco.
 - `Imei`
 - No permite cadenas en blanco.
 - Patrón: `IMEI-[0-9]{7,15}` (ej. `IMEI-123456789`).
 - `FirmwareVersion`
 - No permite cadenas en blanco.
 - Patrón: `vMAJOR.MINOR.PATCH` (ej. `v1.8.2`).
 - `OdometerKm`
 - Entero ≥ 0 .
 - Enums:
 - `VehicleStatus`: `IN_SERVICE`, `OUT_OF_SERVICE`, `MAINTENANCE`, `RETIRED`.
 - `VehicleType`: `VAN`, `TRUCK`, `CAR`, `MOTORCYCLE`.
 - `Capability`: `BOX`, `REFRIGERATED`, `GPS_ONLY`, `HEAVY_LOAD`, `FRAGILE_CARGO`.
-

Excepciones de dominio

- `VehicleNotFoundException`

- `VehiclePlateAlreadyExistsException`
- `VehicleAlreadyHasDeviceException`
- `DeviceNotFoundException`
- `DeviceImeiAlreadyExistsException`
- `DeviceAssignmentConflictException`
- `DeviceAlreadyAssignedException`

Estas excepciones se traducen en la capa de interfaz a códigos HTTP estándar (`404`, `409`, etc.).

Commands

Devices

- `CreateDeviceCommand(imei, firmware, online?)`
Valida IMEI y firmware no vacíos.
- `UpdateDeviceCommand(id, firmware?, online?)`
Valida que `id` no sea nulo.
- `DeleteDeviceCommand(id)`
- `UpdateDeviceFirmwareCommand(id, firmware)`
Firmware no puede ser vacío.
- `UpdateDeviceOnlineStatusCommand(deviceId, online)`

Vehicles

- `CreateVehicleCommand(plate, type, capabilities, status?, odometerKm)`
 - `plate`, `type`, `capabilities`, `odometerKm` obligatorios.
 - `capabilities` no puede estar vacío.
 - `odometerKm` ≥ 0 .
- `UpdateVehicleCommand(id, type?, capabilities?, status?, odometerKm?)`
- `DeleteVehicleCommand(id)`
- `AssignDeviceToVehicleCommand(vehicleId, deviceImei)`
- `UnassignDeviceFromVehicleCommand(vehicleId, deviceImei)`
- `UpdateVehicleStatusCommand(vehicleId, status)`
Ambos campos obligatorios.

Queries

Devices

- `GetAllDevicesQuery`
- `GetDeviceByIdQuery(id)`
- `GetDeviceByImeiQuery(imei)`
- `GetDevicesByOnlineQuery(online)`

Vehicles

- `GetAllVehiclesQuery`
- `GetVehicleByIdQuery(id)`
- `GetVehicleByPlateQuery(plate)`
- `GetVehiclesByStatusQuery(status)`
- `GetVehiclesByTypeQuery(type)`

4.2.6.2. Interface Layer

Controllers principales (HTTP REST)

Los controladores REST exponen los casos de uso del BC Fleet Management a través de endpoints versionados bajo `/api/v1/fleet`.
Todos los métodos están documentados con **OpenAPI 3.0** usando las anotaciones `@Operation` y `@ApiResponse`.

DeviceController

Base path: `/api/v1/fleet/devices`

- **GET /api/v1/fleet/devices**
 - Descripción: Listar todos los dispositivos.
 - Respuesta: `200 OK` con `List<DeviceResource>`.
- **GET /api/v1/fleet/devices/{id}**
 - Descripción: Obtener un dispositivo por su ID.
 - Respuestas:
 - `200 OK` con `DeviceResource`.

- 404 Not Found si el dispositivo no existe.
- **GET /api/v1/fleet/devices/by-imei/{imei}**
 - Descripción: Buscar un dispositivo por IMEI.
 - Respuestas:
 - 200 OK con DeviceResource.
 - 404 Not Found si no existe.
 - **GET /api/v1/fleet/devices/by-online/{online}**
 - Descripción: Listar dispositivos filtrados por estado online (true/false).
 - Respuesta: 200 OK con List<DeviceResource>.
 - **POST /api/v1/fleet/devices**
 - Descripción: Crear un nuevo dispositivo IoT.
 - Request body: CreateDeviceResource { imei, firmware, online? }.
 - Respuestas:
 - 201 Created con DeviceResource.
 - 400 Bad Request si IMEI/firmware no cumplen validaciones.
 - 409 Conflict si el IMEI ya existe (DeviceImeiAlreadyExistsException).
 - **PUT /api/v1/fleet/devices/{id}**
 - Descripción: Actualizar firmware y/o estado online de un dispositivo.
 - Request body: UpdateDeviceResource { firmware?, online? }.
 - Respuestas:
 - 200 OK con DeviceResource actualizado.
 - 404 Not Found si el dispositivo no existe.
 - **DELETE /api/v1/fleet/devices/{id}**
 - Descripción: Eliminar un dispositivo.
 - Reglas:
 - Solo se puede eliminar si no está asignado a ningún vehículo.
 - Respuestas:
 - 204 No Content si se elimina correctamente.
 - 404 Not Found si el dispositivo no existe.
 - 409 Conflict si el dispositivo está asignado (DeviceAssignmentConflictException).
 - **POST /api/v1/fleet/devices/{id}/firmware**
 - Descripción: Actualizar la versión de firmware del dispositivo.
 - Parámetros: firmware como requestParam.
 - Respuestas:
 - 200 OK con DeviceResource actualizado.
 - 404 Not Found si el dispositivo no existe.
 - 400 Bad Request si la versión no cumple el patrón vMAJOR.MINOR.PATCH.
 - **PATCH /api/v1/fleet/devices/{id}/online**
 - Descripción: Cambiar el estado online del dispositivo.
 - Request body: UpdateDeviceOnlineStatusResource { online: Boolean }.
 - Nota: Si online llega como null, se interpreta como false.
 - Respuestas:
 - 200 OK con DeviceResource actualizado.
 - 404 Not Found si el dispositivo no existe.

VehicleController

Base path: /api/v1/fleet/vehicles

- **GET /api/v1/fleet/vehicles**
 - Descripción: Listar todos los vehículos registrados.
 - Respuesta: 200 OK con List<VehicleResource>.
- **GET /api/v1/fleet/vehicles/{id}**
 - Descripción: Obtener un vehículo por ID.
 - Respuestas:
 - 200 OK con VehicleResource.
 - 404 Not Found si no existe.
- **GET /api/v1/fleet/vehicles/by-plate/{plate}**

- Descripción: Buscar vehículo por placa.
- Respuestas:
 - 200 OK con `VehicleResource`.
 - 404 Not Found si no existe.
- **GET /api/v1/fleet/vehicles/by-status/{status}**
 - Descripción: Listar vehículos por estado (`IN_SERVICE`, `OUT_OF_SERVICE`, `MAINTENANCE`, `RETIRED`).
 - Respuesta: 200 OK con `List<VehicleResource>`.
- **GET /api/v1/fleet/vehicles/by-type/{type}**
 - Descripción: Listar vehículos por tipo (`VAN`, `TRUCK`, `CAR`, `MOTORCYCLE`).
 - Respuesta: 200 OK con `List<VehicleResource>`.
- **POST /api/v1/fleet/vehicles**
 - Descripción: Registrar un nuevo vehículo en la flota.
 - Request body: `CreateVehicleResource { plate, type, capabilities[], status?, odometerKm }`.
 - Respuestas:
 - 201 Created con `VehicleResource`.
 - 400 Bad Request si los datos violan restricciones (placa inválida, capabilities vacío, odómetro negativo).
 - 409 Conflict si la placa ya existe (`VehiclePlateAlreadyExistsException`).
- **PUT /api/v1/fleet/vehicles/{id}**
 - Descripción: Actualizar tipo, capacidades, estado u odómetro.
 - Request body: `UpdateVehicleResource { type?, capabilities?, status?, odometerKm? }`.
 - Respuestas:
 - 200 OK con `VehicleResource`.
 - 404 Not Found si el vehículo no existe.
- **DELETE /api/v1/fleet/vehicles/{id}**
 - Descripción: Eliminar un vehículo de la flota.
 - Reglas:
 - Solo se puede eliminar si el `status == RETIRED` y **no tiene dispositivos asociados**.
 - Respuestas:
 - 204 No Content si se elimina correctamente.
 - 404 Not Found si no existe.
 - 409 Conflict si:
 - No está en estado `RETIRED`, o
 - Tiene dispositivos asignados (`DeviceAssignmentConflictException / IllegalStateException`).
- **POST /api/v1/fleet/vehicles/{id}/assign-device/{imei}**
 - Descripción: Asignar un dispositivo a un vehículo.
 - Reglas:
 - No se permite asignar dispositivos a vehículos en estado `RETIRED`.
 - Un dispositivo no puede estar asignado simultáneamente a otro vehículo distinto.
 - Respuestas:
 - 200 OK con `VehicleResource` actualizado (lista `deviceImeis`).
 - 404 Not Found si vehículo o dispositivo no existen.
 - 409 Conflict si:
 - El vehículo ya tiene el dispositivo.
 - El dispositivo está asignado a otro vehículo.
 - Se intenta asignar a un vehículo `RETIRED`.
- **POST /api/v1/fleet/vehicles/{id}/unassign-device/{imei}**
 - Descripción: Desasignar un dispositivo de un vehículo concreto.
 - Respuestas:
 - 200 OK con `VehicleResource` actualizado.
 - 404 Not Found si el vehículo no existe.
 - 400 Bad Request si el dispositivo no está asociado a ese vehículo.
- **PATCH /api/v1/fleet/vehicles/{id}/status**
 - Descripción: Cambiar el estado de un vehículo.
 - Request body: `UpdateVehicleStatusResource { status }`.
 - Reglas:
 - No se permite cambiar desde `RETIRED` a cualquier otro estado (conflicto de negocio).
 - Respuestas:
 - 200 OK con `VehicleResource` actualizado.
 - 400 Bad Request si el estado es inválido o nulo.
 - 404 Not Found si el vehículo no existe.

- 409 Conflict si el cambio de estado no está permitido (ej. desde RETIRED).

Seguridad (ACLs)

- La protección de endpoints se gestiona a nivel de API Gateway/IAM (autenticación JWT).
- Este BC asume que las solicitudes que llegan a `/api/v1/fleet/**` ya están autenticadas y con permisos adecuados, por lo que se enfoca en las reglas de negocio propias de Fleet (unicidad, estados válidos, relaciones vehículo-dispositivo).

4.2.6.3. Application Layer

Command Services

DeviceCommandServiceImpl

- **Propósito:** Gestionar el ciclo de vida de los dispositivos IoT y su estado.
- **Métodos:**
 - `handle(CreateDeviceCommand)`
 - Verifica unicidad de IMEI (`existsByImei`).
 - Crea el agregado `Device` y lo persiste.
 - `handle(UpdateDeviceCommand)`
 - Busca por ID, lanza `DeviceNotFoundException` si no existe.
 - Actualiza `firmware` (si se envía) y/o `online`.
 - `handle>DeleteDeviceCommand)`
 - Solo permite eliminar si el dispositivo **no está asignado** (`isAssigned() → false`).
 - Caso contrario, lanza `DeviceAssignmentConflictException`.
 - `handle(UpdateDeviceFirmwareCommand)`
 - Actualiza exclusivamente la versión de firmware.
 - `handle(UpdateDeviceOnlineStatusCommand)`
 - Cambia el valor de `online`, interpretando `null` como `false`.
- **Dependencias:** `DeviceRepository`, `VehicleRepository`.
- **Transacciones:** uso de `@Transactional` para garantizar atomicidad por comando.

VehicleCommandServiceImpl

- **Propósito:** Gestionar creación, actualización, eliminación y asociación de dispositivos en vehículos.
- **Métodos:**
 - `handle(CreateVehicleCommand)`
 - Verifica unicidad de placa (`existsByPlate`).
 - Valida capacidades y odómetro (no vacíos, ≥ 0).
 - `handle(UpdateVehicleCommand)`
 - Busca vehículo por ID, lanza `VehicleNotFoundException` si no existe.
 - Actualiza tipo, capacidades, estado y/o odómetro respetando reglas de dominio.
 - `handle>DeleteVehicleCommand)`
 - Verifica que el vehículo:
 - Esté en estado `RETIRED`.
 - No tenga devices asociados (`hasAnyDevice() == false`).
 - Si no se cumplen las reglas, lanza `IllegalStateException` o `DeviceAssignmentConflictException`.
 - `handle(AssignDeviceToVehicleCommand)`
 - Valida existencia de vehículo y device.
 - No permite asignar a vehículo `RETIRED`.
 - No permite asignar un dispositivo que ya esté asociado a otro vehículo.
 - Actualiza tanto el agregado `Vehicle` (set `deviceImeis`) como `Device` (set `vehiclePlate`).
 - `handle(UnassignDeviceFromVehicleCommand)`
 - Verifica que el dispositivo esté realmente asociado a ese vehículo (mismo `vehiclePlate`).
 - Desasocia en ambos agregados.
 - `handle(UpdateVehicleStatusCommand)`
 - Cambia el estado del vehículo respetando la regla de no salir de `RETIRED`.
- **Dependencias:** `VehicleRepository`, `DeviceRepository`.
- **Transacciones:** todos los métodos de escritura se marcan con `@Transactional`.

Query Services

DeviceQueryServiceImpl

- **Propósito:** Consultas de solo lectura sobre dispositivos.
- **Métodos:**
 - `handle(GetAllDevicesQuery)` → lista completa.
 - `handle(GetDeviceByIdQuery)` → `Optional<Device>`.
 - `handle(GetDeviceByImeiQuery)` → `Optional<Device>`.
 - `handle(GetDevicesByOnlineQuery)` → lista filtrada por `online`.

VehicleQueryServiceImpl

- **Propósito:** Consultas de solo lectura sobre vehículos.
- **Métodos:**
 - `handle(GetAllVehiclesQuery)` → lista completa.
 - `handle(GetVehicleByIdQuery)` → `Optional<Vehicle>`.
 - `handle(GetVehicleByPlateQuery)` → `Optional<Vehicle>`.
 - `handle(GetVehiclesByStatusQuery)` → lista por estado.
 - `handle(GetVehiclesByTypeQuery)` → lista por tipo.

Consideraciones transversales

- **Transacciones:**
1 comando = 1 transacción, gestionado con `@Transactional` en la capa de aplicación.
- **Errores estándar en la API:**
 - `404 NotFound` para agregados inexistentes.
 - `409 Conflict` para violaciones de reglas de negocio (ej. eliminar vehículo no RETIRED, asignar dispositivo ya asignado).
 - `400 Bad Request` para errores de validación de Value Objects (patrones, rangos, nulls no permitidos).

4.2.6.4. Infrastructure Layer

Repositories (JPA)

- `DeviceRepository` extends `JpaRepository<Device, Long>`
 - `Optional<Device> findByImei(Imei imei)`
 - `boolean existsByImei(Imei imei)`
 - `List<Device> findAllByOnline(boolean online)`
 - `List<Device> findAllByVehiclePlate(Plate vehiclePlate)`
- `VehicleRepository` extends `JpaRepository<Vehicle, Long>`
 - `Optional<Vehicle> findByPlate(Plate plate)`
 - `boolean existsByPlate(Plate plate)`
 - `List<Vehicle> findAllByStatus(VehicleStatus status)`
 - `List<Vehicle> findAllByType(VehicleType type)`
 - `Optional<Vehicle> findByDeviceImeis(Imei deviceImeis)`

Persistencia

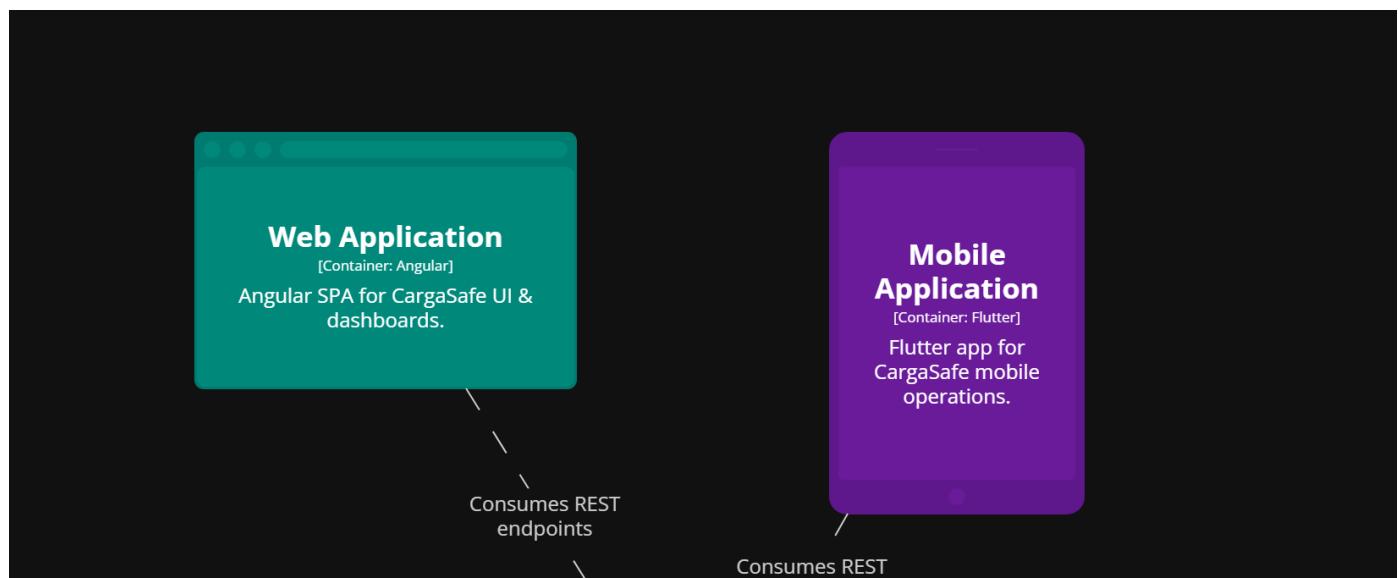
- Implementación basada en **Spring Data JPA**, sobre una base de datos relacional (p. ej. PostgreSQL) compartida en el backend de CargaSafe.
- Los **Value Objects** (`Plate`, `Imei`, `FirmwareVersion`, `OdometerKm`) se mapean como `@Embeddable` con `@AttributeOverride` para definir columnas específicas.
- Las colecciones (`capabilities`, `deviceImeis`) se modelan con `@ElementCollection` y tablas de relación dedicadas (`vehicle_capabilities`, `vehicle_device_imeis`).

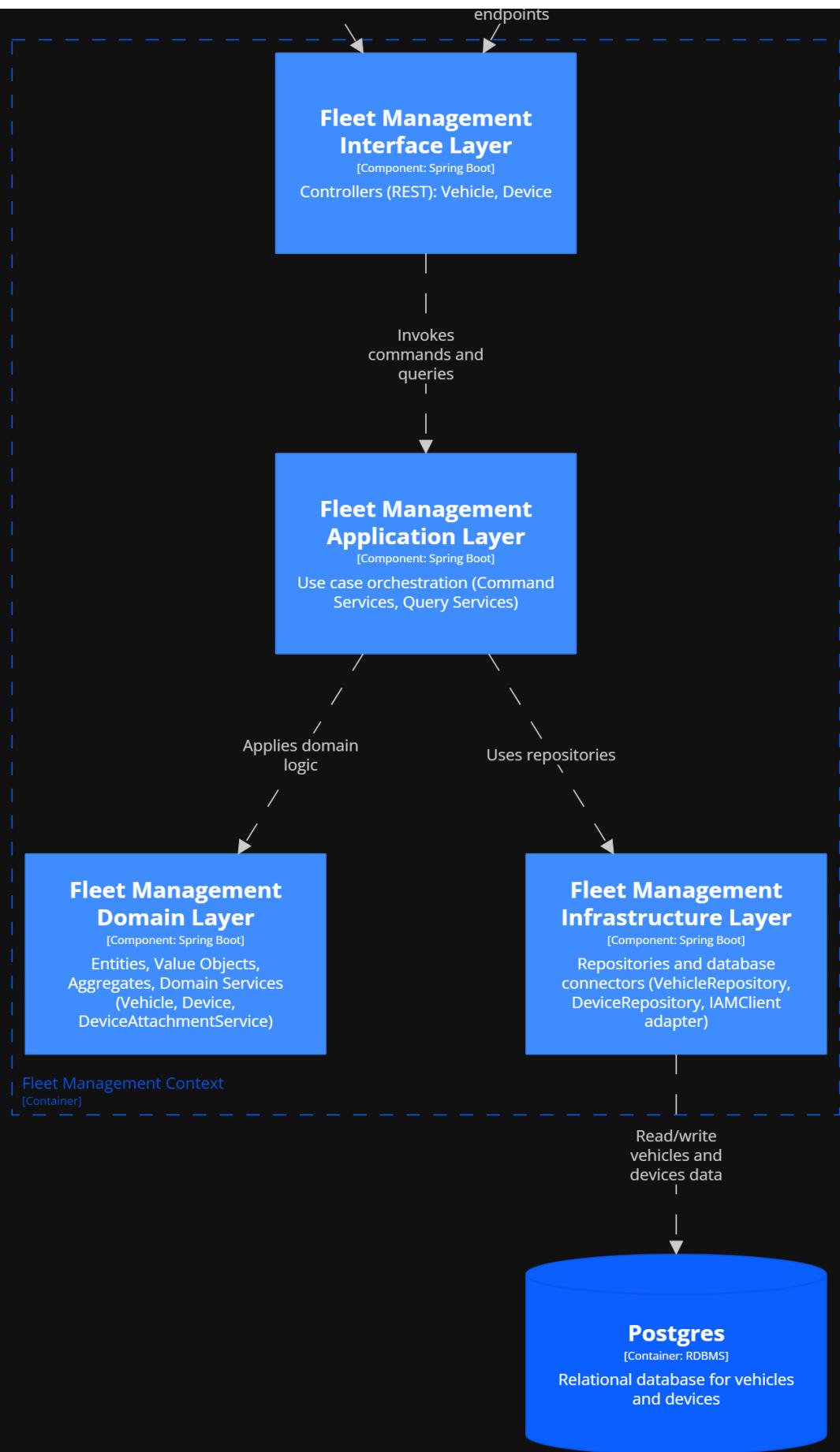
Integración y seguridad

- La validación de JWT, scopes y multi-tenant se realiza en el API Gateway / BC de IAM.
- El BC Fleet Management se enfoca en la lógica de negocio de flota y en mantener la consistencia de vehículos y dispositivos dentro de la base de datos.

4.2.6.5. Bounded Context Software Architecture Component Level Diagrams.

Diagrama de componentes - Backend - Fleet Management





Fleet Management Context - Component Diagram

Internal layered components of the Fleet Management Context inside CargaSafe.
sábado, 20 de septiembre de 2025, 6:05 hora estándar de Perú

Diagrama de componentes - Application Web - Fleet Management

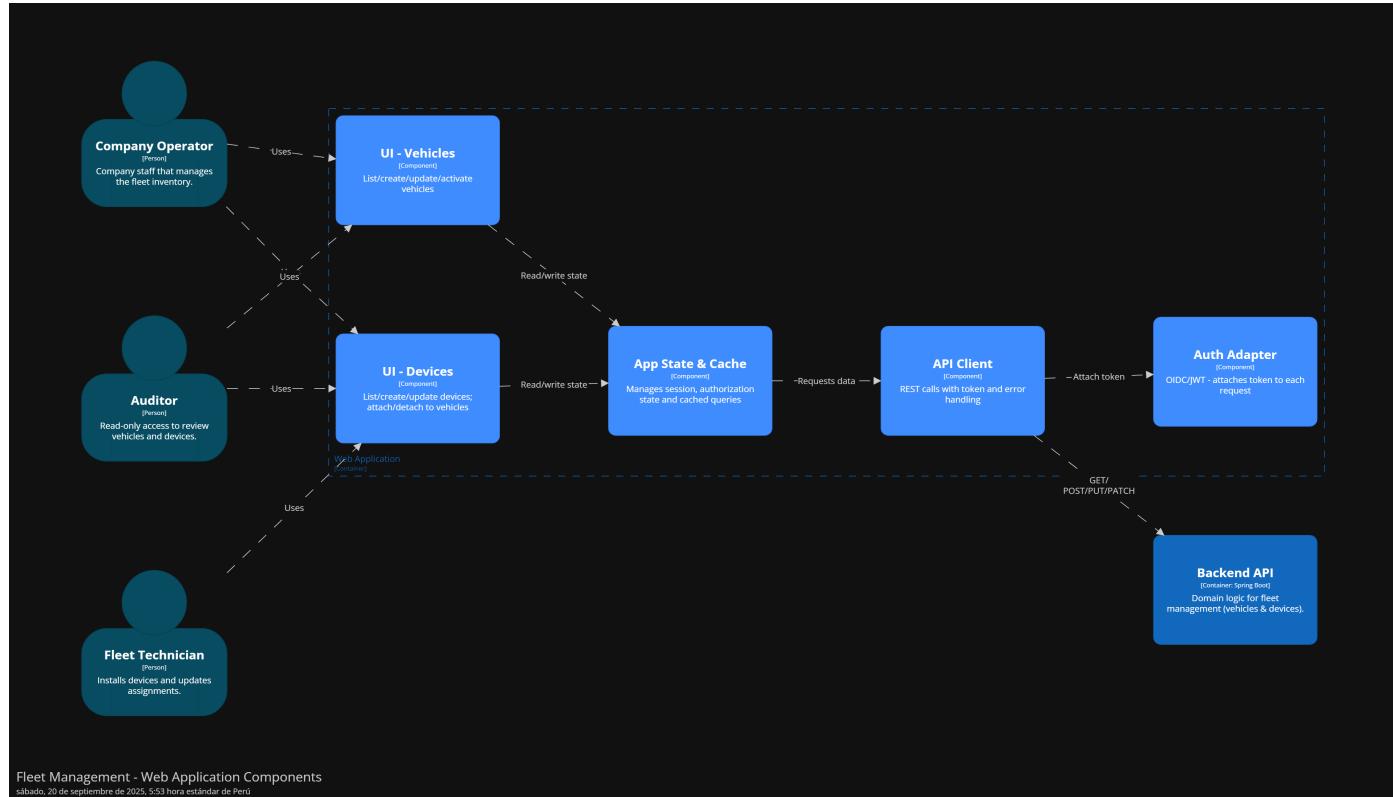
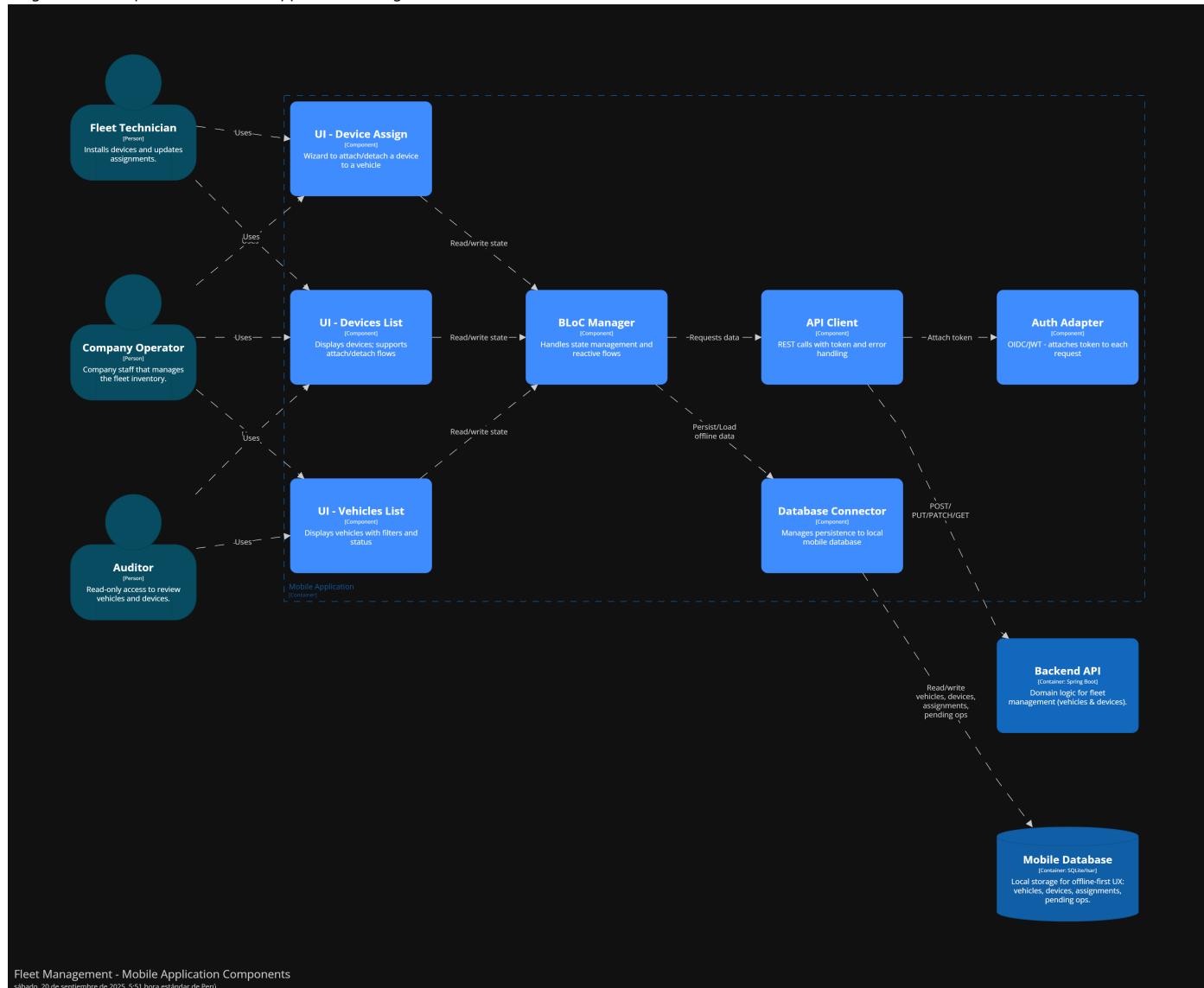


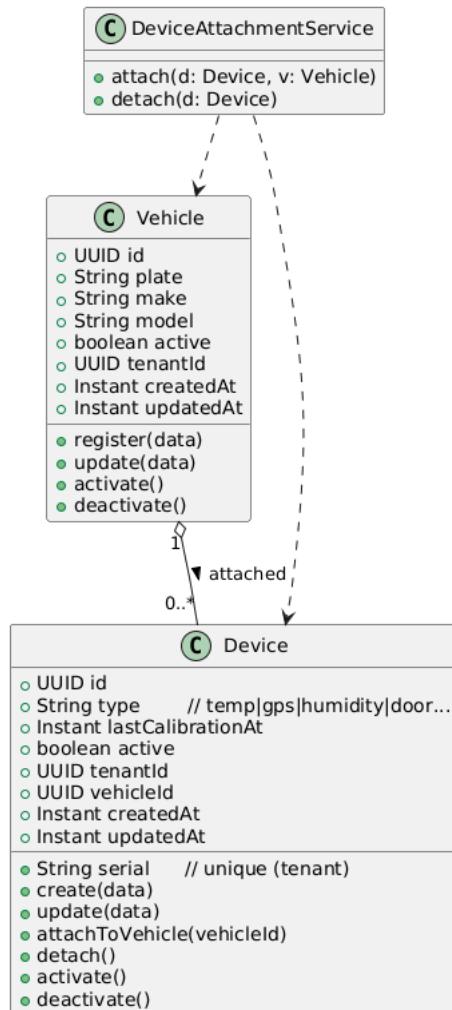
Diagrama de componentes - Mobile App - Fleet Management

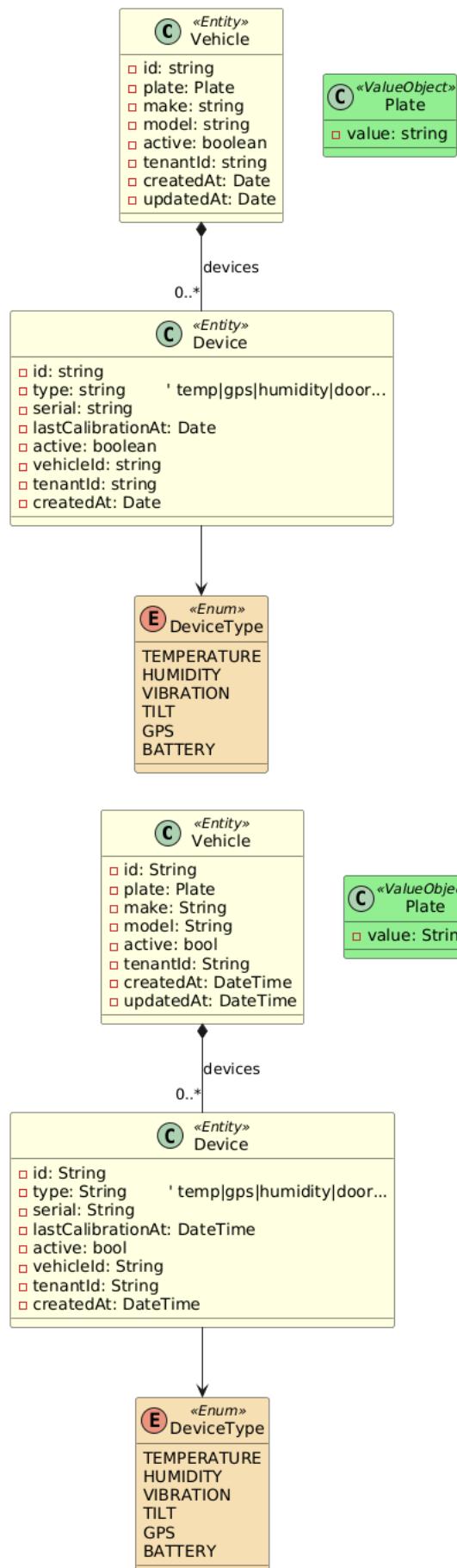


4.2.5.6. Bounded Context Software Architecture Code Level Diagrams.

4.2.5.6.1. Bounded Context Domain Layer Class Diagrams.

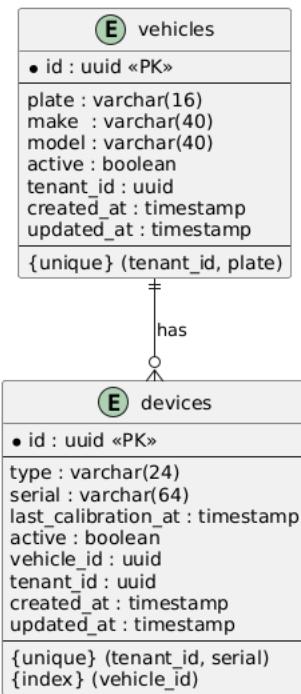
Fleet Management - Backend Domain Layer Class Diagram





4.2.5.6.2. Bounded Context Database Design Diagram.

Fleet Management - Database Design (PostgreSQL)



4.2.7. Bounded Context: Profile and Preferences Management

4.2.7.1. Domain Layer.

Entity: Profile (Aggregate Root)

Propósito principal

Representar el perfil dentro del sistema, siempre asociado a un `userId`.

Atributos principales

- id: Identificador único del perfil.
- userId: Identificador externo obligatorio asociado al perfil.
- fullName: Nombre completo.
- phoneNumber: Número de contacto (opcional).
- avatarUrl: Imagen de perfil o avatar (opcional).
- createdAt: Fecha de creación.
- updatedAt: Fecha de última actualización.

Métodos principales

- updateContactInfo(newPhone)
- updateAvatar(newAvatarUrl)
- updateName(newName)

Entity: Preferences

Propósito principal

Almacenar configuraciones personalizadas de idioma, zona horaria y notificaciones de alertas.

Atributos principales

- id: Identificador único de las preferencias.
- profileId: Identificador del perfil asociado.
- language: Idioma preferido.
- timeZone: Zona horaria configurada.
- alertEmailEnabled: Recibir alertas por correo electrónico.
- alertPushEnabled: Recibir alertas como notificaciones push en la aplicación.
- alertSmsEnabled: Recibir alertas vía SMS (opcional).

Métodos principales

- updateLanguage(language)
- updateTimeZone(timeZone)
- enableEmailAlerts(flag)
- enablePushAlerts(flag)

-
- enableSmsAlerts(flag)

Value Object: PhoneNumber

- countryCode
- number

Value Object: Language

- code (ej. "es", "en")
- name

Value Object: TimeZone

- code (ej. "UTC-5")
-

Aggregate: ProfileAggregate

- updateProfileAndPreferences(profile, preferences)
-

Factory: ProfileFactory

- createProfile(fullName, userId)
-

Domain Service: PreferencesValidationService

- validateNotificationSettings(emailFlag, pushFlag, smsFlag)
 - validateLanguage(language)
-

Command: CreateProfileCommand

- fullName
- phoneNumber
- avatarUrl
- userId

Command: UpdateProfileCommand

- id
- newName
- newPhone
- newAvatarUrl

Command: UpdatePreferencesCommand

- id
 - language
 - timeZone
 - alertEmailEnabled
 - alertPushEnabled
 - alertSmsEnabled
-

Query: GetProfileByIdQuery

- id

Query: GetPreferencesByProfileIdQuery

- profileId
-

Event: ProfileCreatedEvent

- id
- userId
- createdAt

Event: ProfileUpdatedEvent

- id
- updatedAt

Event: PreferencesUpdatedEvent

- id

- updatedAt
- changes

4.2.7.2. Interface Layer.

Controllers

- **ProfileController:** Maneja las solicitudes para crear, actualizar y consultar perfiles.
- **PreferencesController:** Maneja las solicitudes para modificar y consultar preferencias (idioma, zona horaria, notificaciones).

4.2.7.3. Application Layer.

Command Services

- **ProfileCommandService:** Coordina la ejecución de comandos relacionados con perfiles (creación y actualización).
- **PreferencesCommandService:** Coordina la ejecución de comandos relacionados con preferencias (idioma, zona horaria, notificaciones).

Query Services

- **ProfileQueryService:** Atiende consultas sobre perfiles.
- **PreferencesQueryService:** Atiende consultas sobre las preferencias de un perfil.

Event Services

- **ProfileEventService:** Gestiona eventos relacionados con la creación y actualización de perfiles.
- **PreferencesEventService:** Gestiona eventos relacionados con la actualización de preferencias y su propagación a otros componentes.

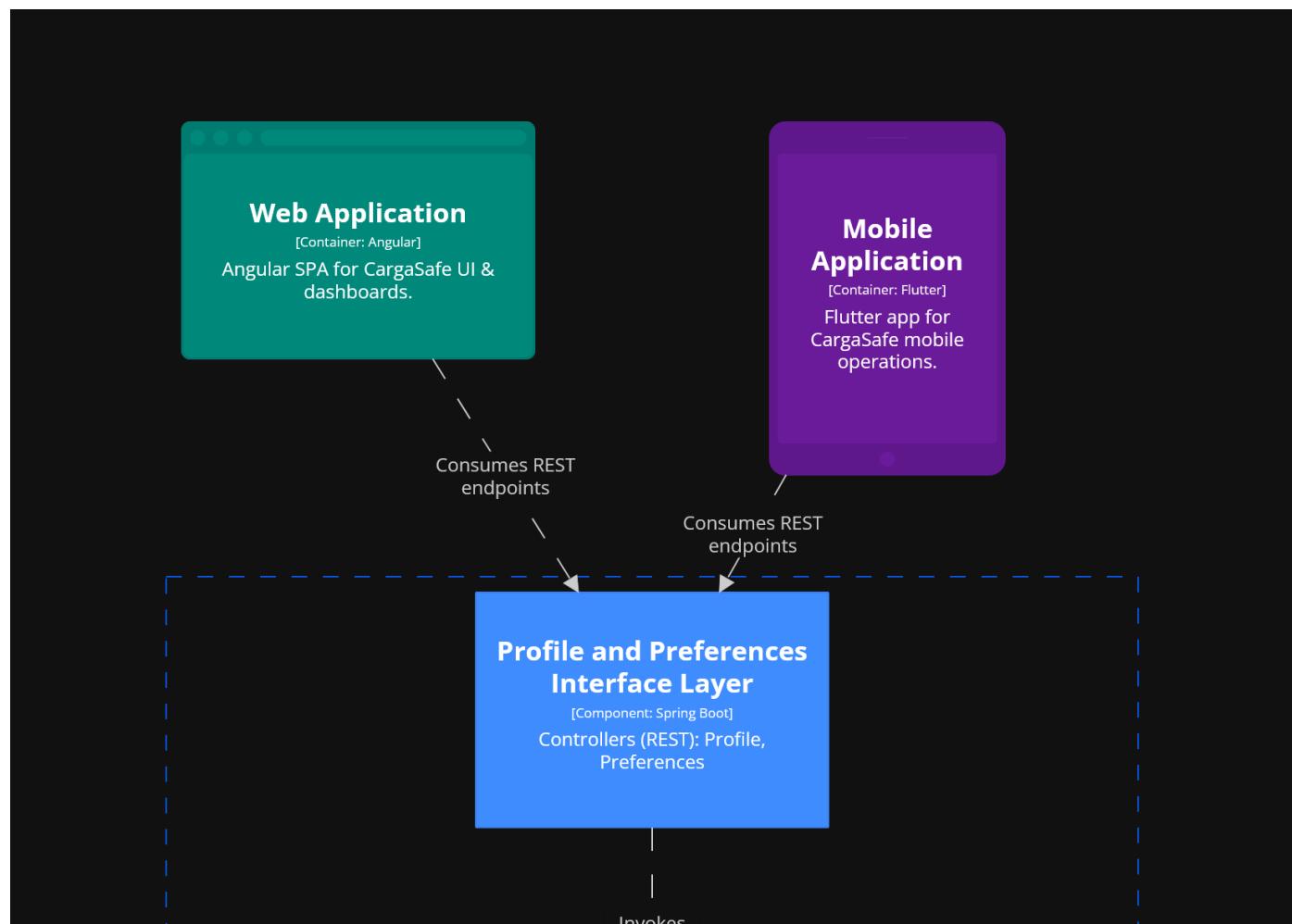
4.2.7.4. Infrastructure Layer.

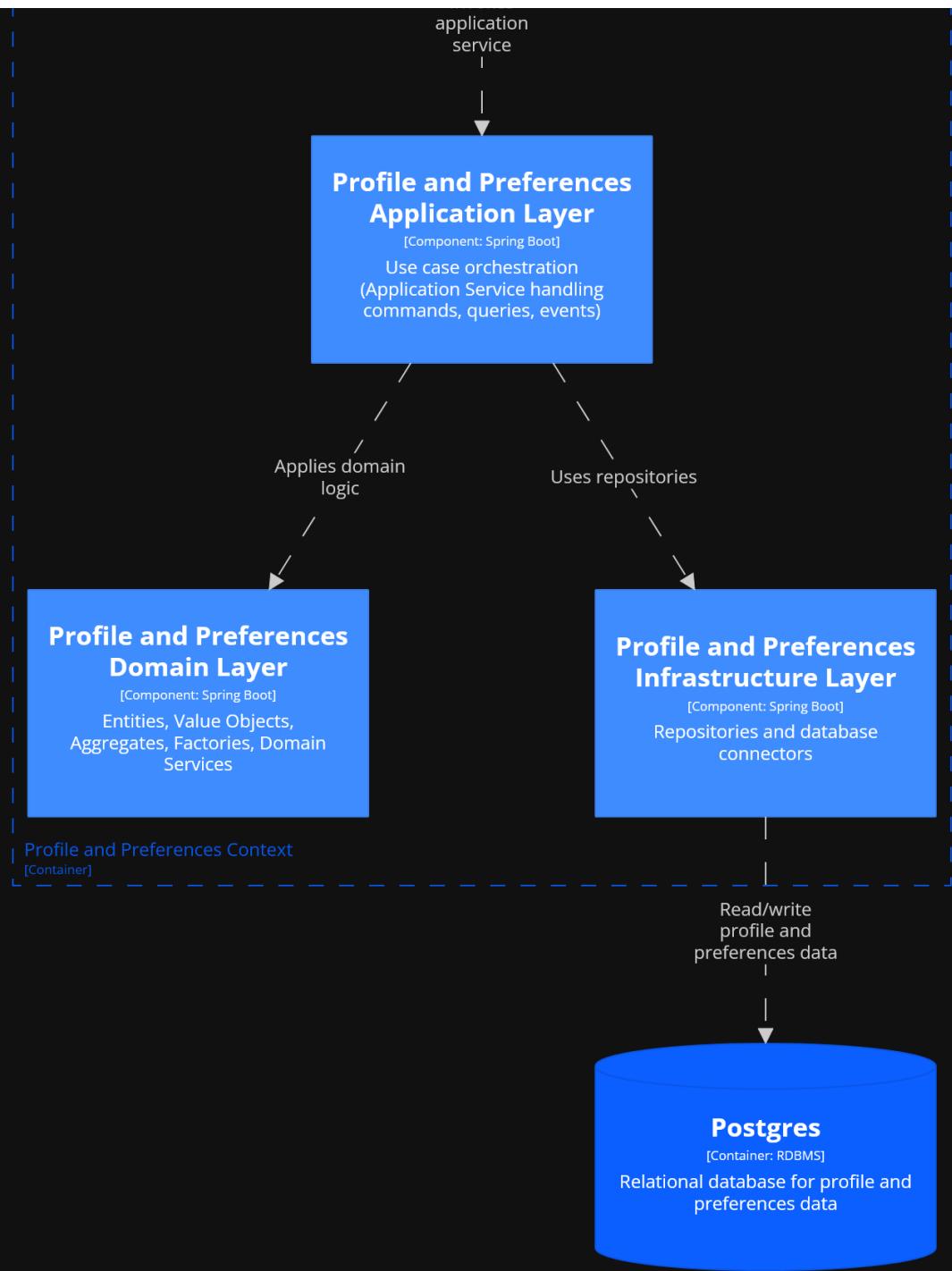
Repositories

- **IProfileRepository:** Define operaciones de acceso y persistencia para los perfiles.
- **IPreferencesRepository:** Define operaciones de acceso y persistencia para las preferencias.

4.2.7.5. Bounded Context Software Architecture Component Level Diagrams.

Diagrama de componentes - Backend - Profiles and Preferences Management





Profile and Preferences Context - Component Diagram

Internal layered components of the Profile and Preferences Context inside CargaSafe.

Saturday, September 20, 2025 at 6:29 AM Colombia Standard Time

Diagrama de componentes - Application Web - Profiles and Preferences Management

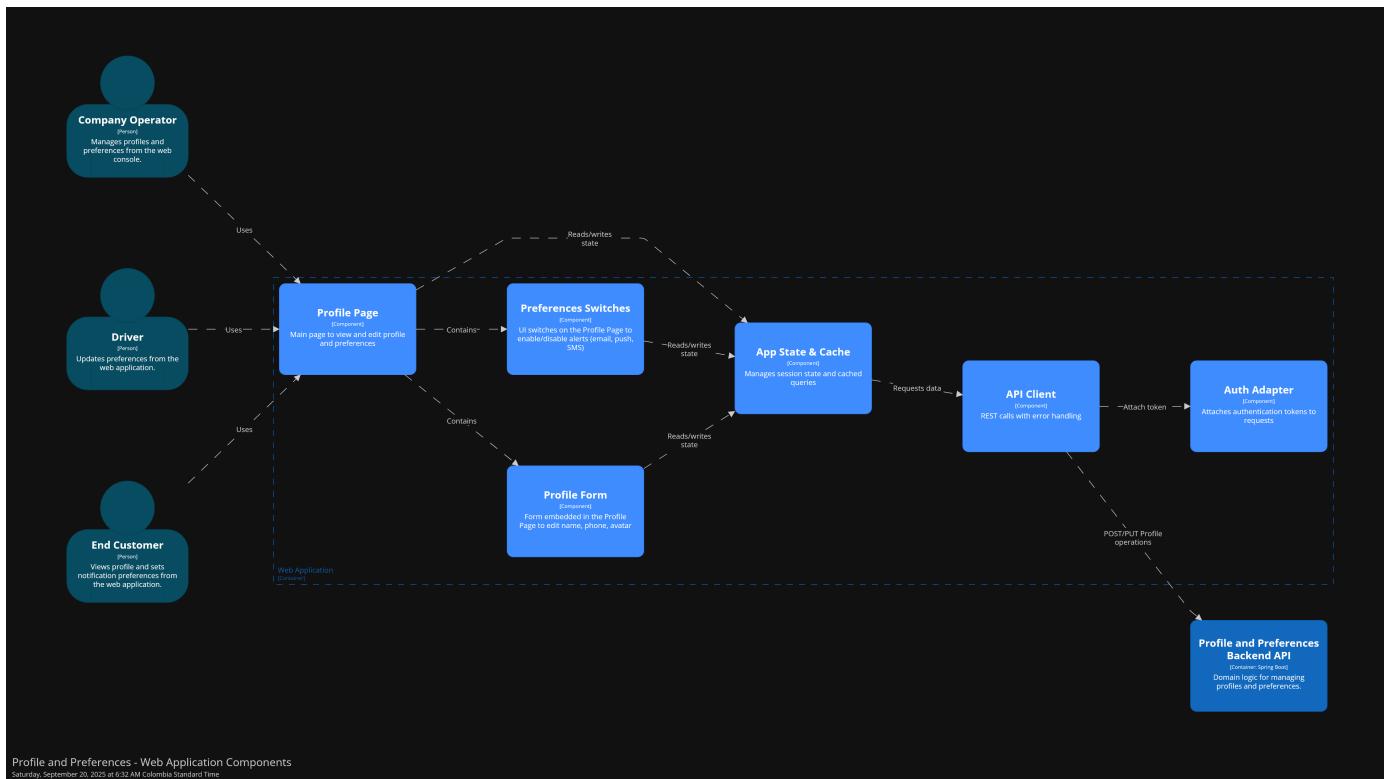
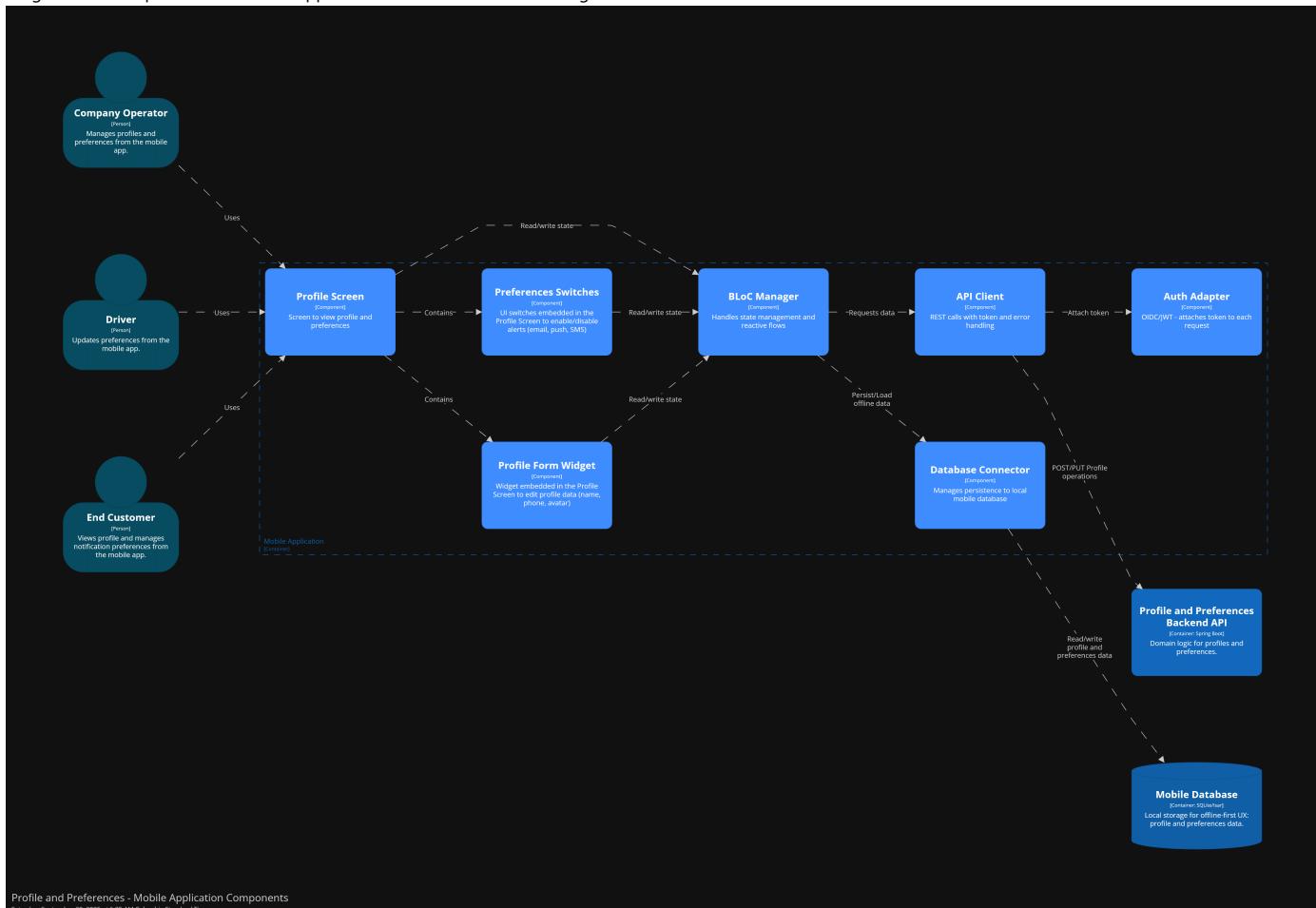


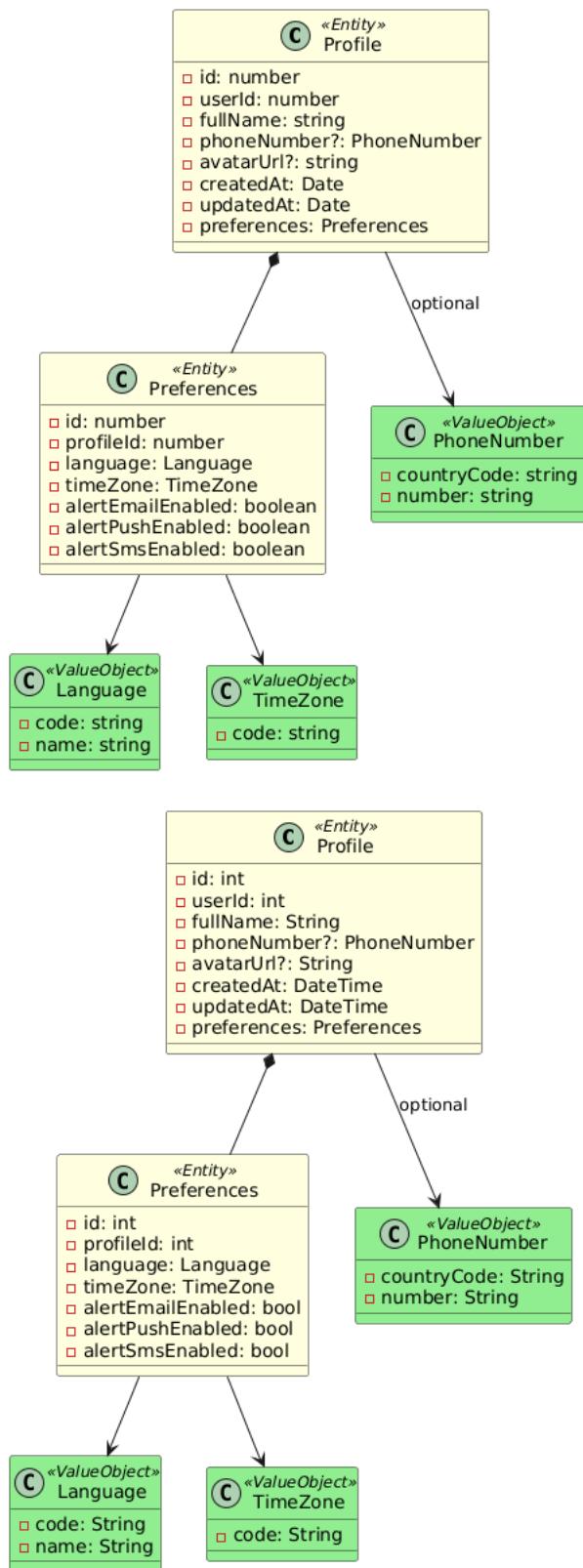
Diagrama de componentes - Mobile App - Profiles and Preferences Management



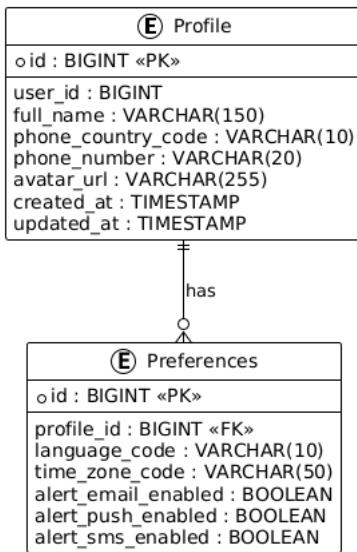
4.2.7.6. Bounded Context Software Architecture Code Level Diagrams.

4.2.7.6.1. Bounded Context Domain Layer Class Diagrams.

Profile And Preferences Management Domain Layer Class Diagram Backend



4.2.7.6.2. Bounded Context Database Design Diagram



4.2.8. Bounded Context: Visualization Analytics

4.2.8.1. Domain Layer

Entidades Principales

Dashboard (Aggregate Root)

- Propósito:** Representa un dashboard personalizable con widgets y métricas específicas
- Atributos principales:**
 - `id`: Identificador único
 - `name`: Nombre del dashboard
 - `userId`: Propietario del dashboard
 - `layout`: Configuración de layout de widgets
 - `isDefault`: Indica si es dashboard por defecto
 - `createdAt, updatedAt`: Timestamps de auditoría
- Métodos principales:**
 - `addWidget(widget)`: Agrega widget al dashboard
 - `removeWidget(widgetId)`: Remueve widget
 - `updateLayout(layout)`: Actualiza disposición de widgets
 - `clone()`: Crea copia del dashboard

Widget (Entity)

- Propósito:** Componente visual que muestra métricas específicas
- Atributos principales:**
 - `id`: Identificador único
 - `type`: Tipo de widget (CHART, KPI, TABLE, MAP)
 - `title`: Título del widget
 - `dataSource`: Fuente de datos
 - `configuration`: Configuración específica del widget
 - `position`: Posición en el dashboard
- Métodos principales:**
 - `updateConfiguration(config)`: Actualiza configuración
 - `refresh()`: Refresca datos del widget
 - `validateConfiguration()`: Valida configuración del widget

Report (Entity)

- Propósito:** Reporte generado con datos históricos y métricas
- Atributos principales:**
 - `id`: Identificador único
 - `name`: Nombre del reporte
 - `type`: Tipo de reporte (TRIP_SUMMARY, COMPLIANCE, PERFORMANCE)
 - `parameters`: Parámetros del reporte
 - `generatedAt`: Fecha de generación
 - `format`: Formato del reporte (PDF, EXCEL, CSV)
- Métodos principales:**
 - `generate()`: Genera el reporte
 - `schedule(frequency)`: Programa generación automática
 - `export(format)`: Exporta en formato específico

ChartData (Entity)

- **Propósito:** Datos procesados para visualización en charts
- **Atributos principales:**
 - `id`: Identificador único
 - `chartType`: Tipo de gráfico (LINE, BAR, PIE, SCATTER)
 - `dataPoints`: Puntos de datos
 - `labels`: Etiquetas de los ejes
 - `metadata`: Metadatos adicionales
- **Métodos principales:**
 - `addDataPoint(point)`: Agrega punto de dato
 - `aggregate(groupBy)`: Agrupa datos
 - `filter(criteria)`: Filtra datos

Value Objects

- **TimeRange**: Rango de tiempo para consultas
- **ChartConfiguration**: Configuración específica de gráficos
- **KPIMetric**: Métrica de rendimiento clave
- **DataFilter**: Filtros aplicados a datos
- **ColorSchema**: Esquema de colores para visualizaciones

Domain Services

- **DataAggregationService**: Agregación y cálculo de métricas
- **ChartRenderingService**: Lógica de renderizado de gráficos
- **ReportGenerationService**: Generación de reportes complejos
- **MetricsCalculationService**: Cálculo de KPIs y métricas derivadas

Commands

- **CreateDashboardCommand**: Comando para crear dashboard
- **UpdateWidgetCommand**: Comando para actualizar widget
- **GenerateReportCommand**: Comando para generar reporte
- **RefreshDataCommand**: Comando para refrescar datos

Queries

- **GetDashboardQuery**: Obtiene dashboard por ID
- **GetTripMetricsQuery**: Obtiene métricas de viajes
- **GetComplianceDataQuery**: Obtiene datos de cumplimiento
- **GetTemperatureHistoryQuery**: Obtiene historial de temperatura

Events

- **DashboardCreatedEvent**: Dashboard creado
- **ReportGeneratedEvent**: Reporte generado
- **DataRefreshedEvent**: Datos refrescados
- **AlertThresholdExceededEvent**: Umbral de alerta excedido

4.2.8.2. Interface Layer

Controllers Principales

DashboardController

- `GET /dashboards`: Lista dashboards del usuario
- `POST /dashboards`: Crea nuevo dashboard
- `PUT /dashboards/{id}`: Actualiza dashboard
- `DELETE /dashboards/{id}`: Elimina dashboard
- `GET /dashboards/{id}/data`: Obtiene datos del dashboard

AnalyticsController

- `GET /analytics/trips`: Métricas de viajes
- `GET /analytics/compliance`: Datos de cumplimiento
- `GET /analytics/performance`: Métricas de rendimiento
- `GET /analytics/temperature-history`: Historial de temperatura

ReportController

- `POST /reports/generate`: Genera reporte bajo demanda
- `GET /reports`: Lista reportes generados
- `GET /reports/{id}/download`: Descarga reporte
- `POST /reports/schedule`: Programa reporte automático

VisualizationController

- GET /visualizations/chart-data: Datos para gráficos
- POST /visualizations/custom-chart: Genera gráfico personalizado
- GET /visualizations/kpis: Obtiene KPIs calculados

4.2.8.3. Application Layer

Command Services

DashboardCommandService

- Maneja creación y modificación de dashboards
- Coordina actualización de widgets
- Gestiona permisos de acceso a dashboards

ReportCommandService

- Gestiona generación de reportes
- Maneja programación de reportes automáticos
- Coordina exportación en diferentes formatos

Query Services

AnalyticsQueryService

- Proporciona métricas y KPIs calculados
- Optimizado para consultas complejas de análisis
- Maneja agregaciones temporales

VisualizationQueryService

- Consultas optimizadas para gráficos
- Transformación de datos para visualización
- Cache de datos frecuentemente consultados

Event Handlers

TripCompletedEventHandler

- Procesa finalización de viajes
- Actualiza métricas de rendimiento
- Genera alertas si es necesario

TemperatureViolationEventHandler

- Procesa violaciones de temperatura
- Actualiza métricas de cumplimiento
- Notifica a dashboards relevantes

4.2.8.4. Infrastructure Layer

Repositories

DashboardRepository (implementa IDashboardRepository)

- Persistencia de dashboards y configuraciones
- Optimizado para consultas por usuario
- Cache de dashboards frecuentemente accedidos

ReportRepository (implementa IReportRepository)

- Almacenamiento de reportes generados
- Gestión de archivos de reporte
- Limpieza automática de reportes antiguos

MetricsRepository (implementa IMetricsRepository)

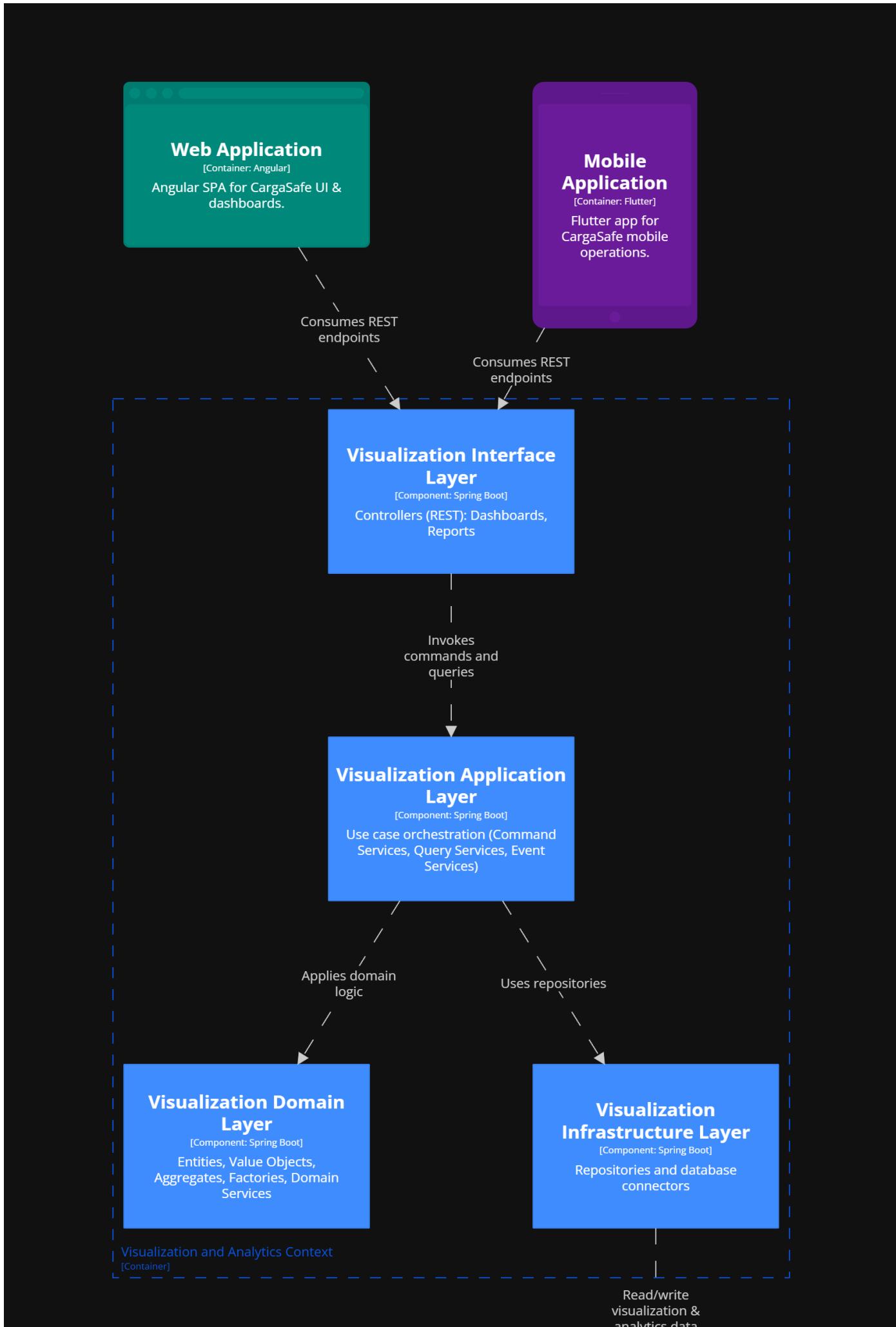
- Consultas optimizadas para métricas agregadas
- Conexión con base de datos de time-series
- Cache de métricas calculadas

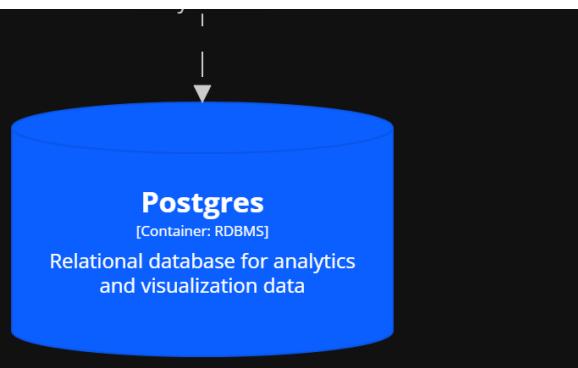
ChartDataRepository (implementa IChartDataRepository)

- Transformación de datos para visualización
- Consultas optimizadas para gráficos
- Manejo de grandes volúmenes de datos temporales

4.2.8.5. Bounded Context Software Architecture Component Level Diagrams

Diagrama de Componentes - Backend - Visualization Analytics





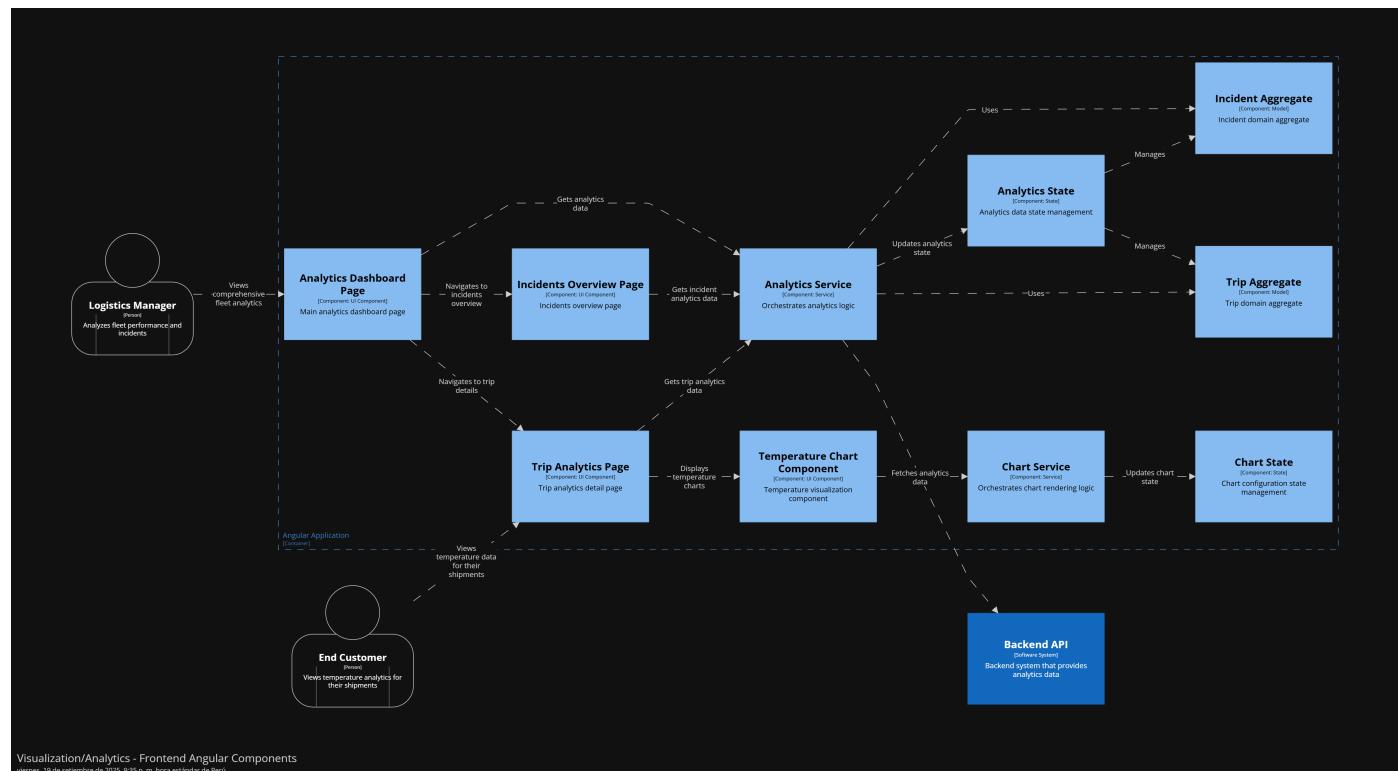
Visualization and Analytics Context - Component Diagram

Internal layered components of the Visualization and Analytics Context inside CargaSafe.

viernes, 19 de setiembre de 2025, 11:36 p. m. hora estándar de Perú

Este diagrama ilustra la arquitectura del bounded context de Visualization Analytics en el backend. Los controllers manejan requests relacionados con dashboards, reportes y análisis. Los servicios en Application Layer coordinan la lógica de negocio, mientras que los repositorios optimizan el acceso a datos tanto transaccionales como de time-series para métricas y visualizaciones.

Diagrama de Componentes - Frontend Web - Visualization Analytics

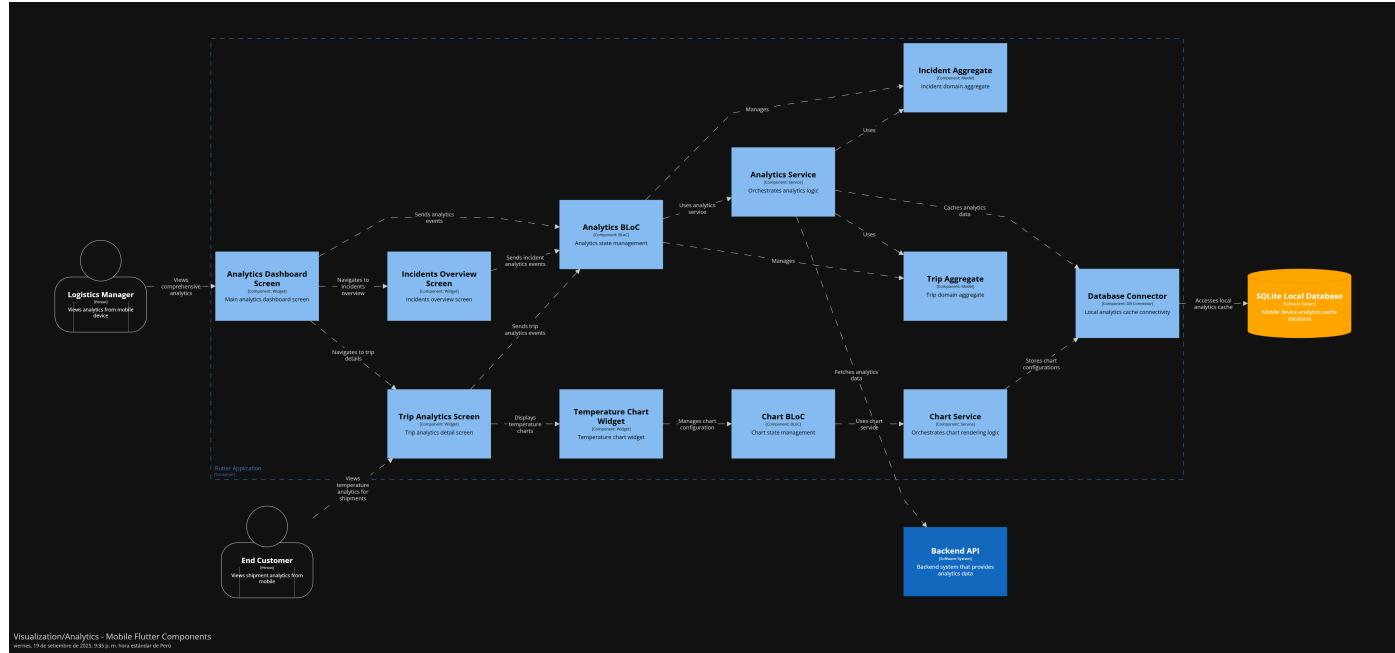


Visualization/Analytics - Frontend Angular Components

viernes, 19 de setiembre de 2025, 9:35 p. m. hora estándar de Perú

El frontend web del módulo de analytics utiliza componentes especializados para visualización de datos. Los chart components renderizarán gráficos interactivos, mientras que dashboard components gestionarán la composición y layout de widgets. Los servicios manejan la comunicación con APIs de datos y el cache local de métricas.

Diagrama de Componentes - Mobile - Visualization Analytics

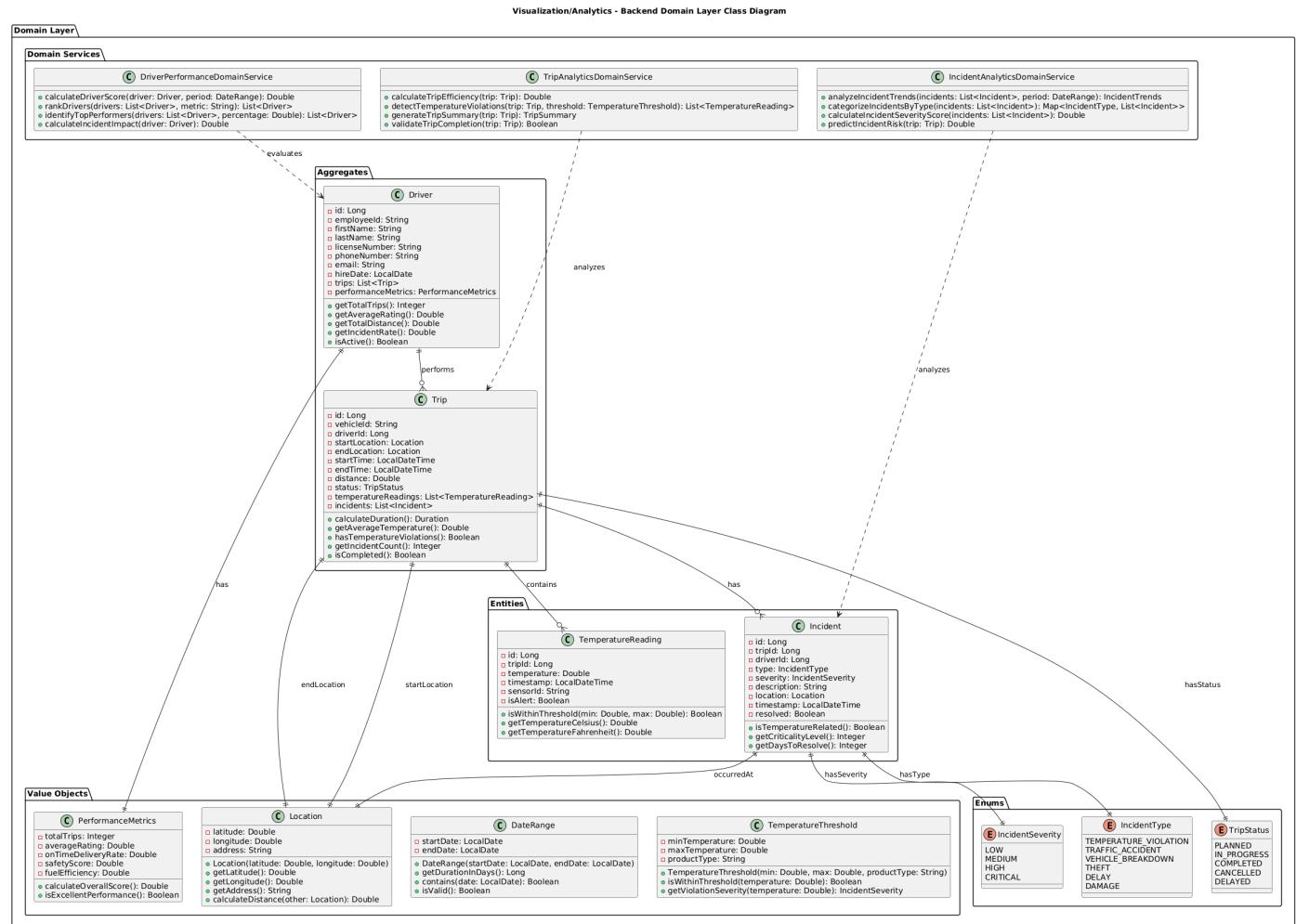


La aplicación móvil prioriza visualizaciones optimizadas para pantallas pequeñas. Los componentes incluyen widgets responsivos y gráficos touch-friendly. El state management atravéss de BLoC coordina la actualización de datos en tiempo real y gestiona el cache local para funcionalidad offline.

4.2.8.6. Bounded Context Software Architecture Code Level Diagrams

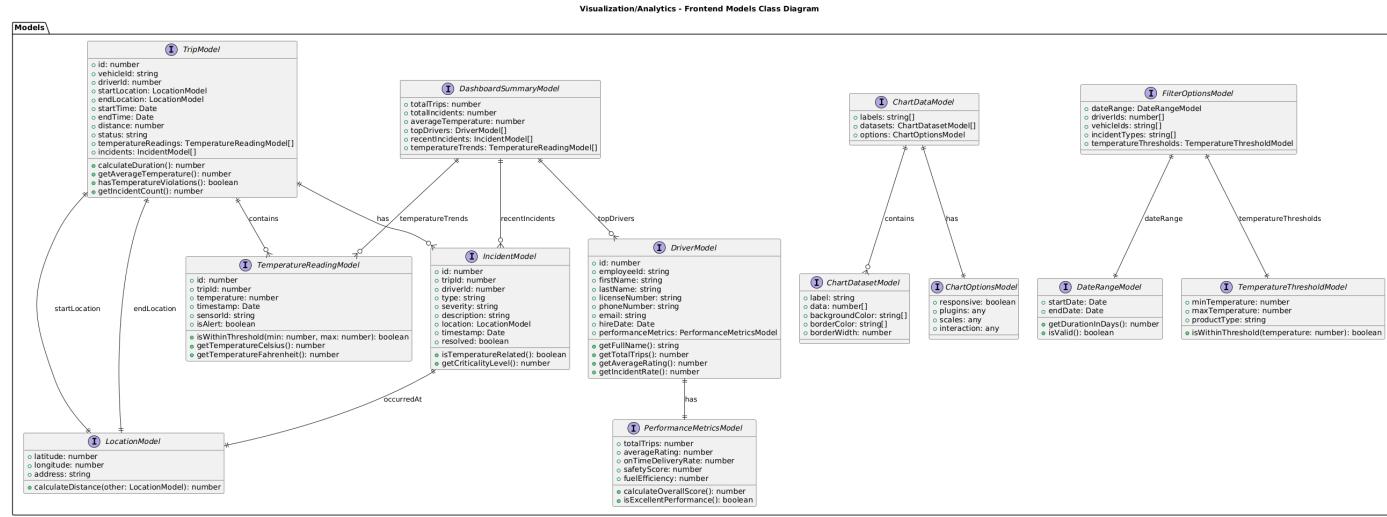
4.2.8.6.1. Bounded Context Domain Layer Class Diagrams

Backend - Visualization Analytics Domain Layer Class Diagram



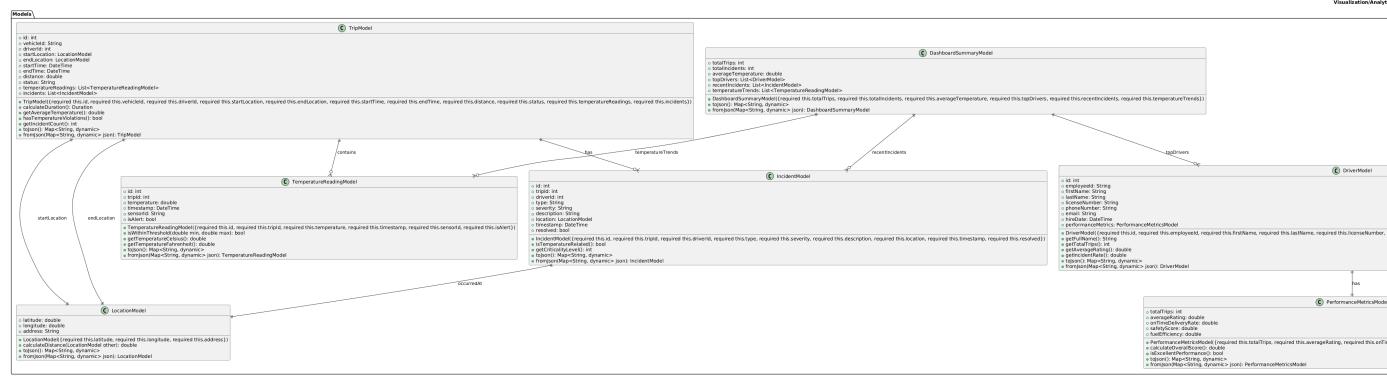
El diagrama de clases del backend de Analytics muestra las entidades principales para visualización y análisis de datos. Dashboard actúa como aggregate root contenido en múltiples Widgets. Los Reports están asociados a usuarios y pueden ser programados para generación automática. ChartData encapsula la información procesada para visualizaciones, mientras que los services coordinan la agregación y cálculo de métricas.

Frontend - Visualization Analytics Domain Layer Class Diagram



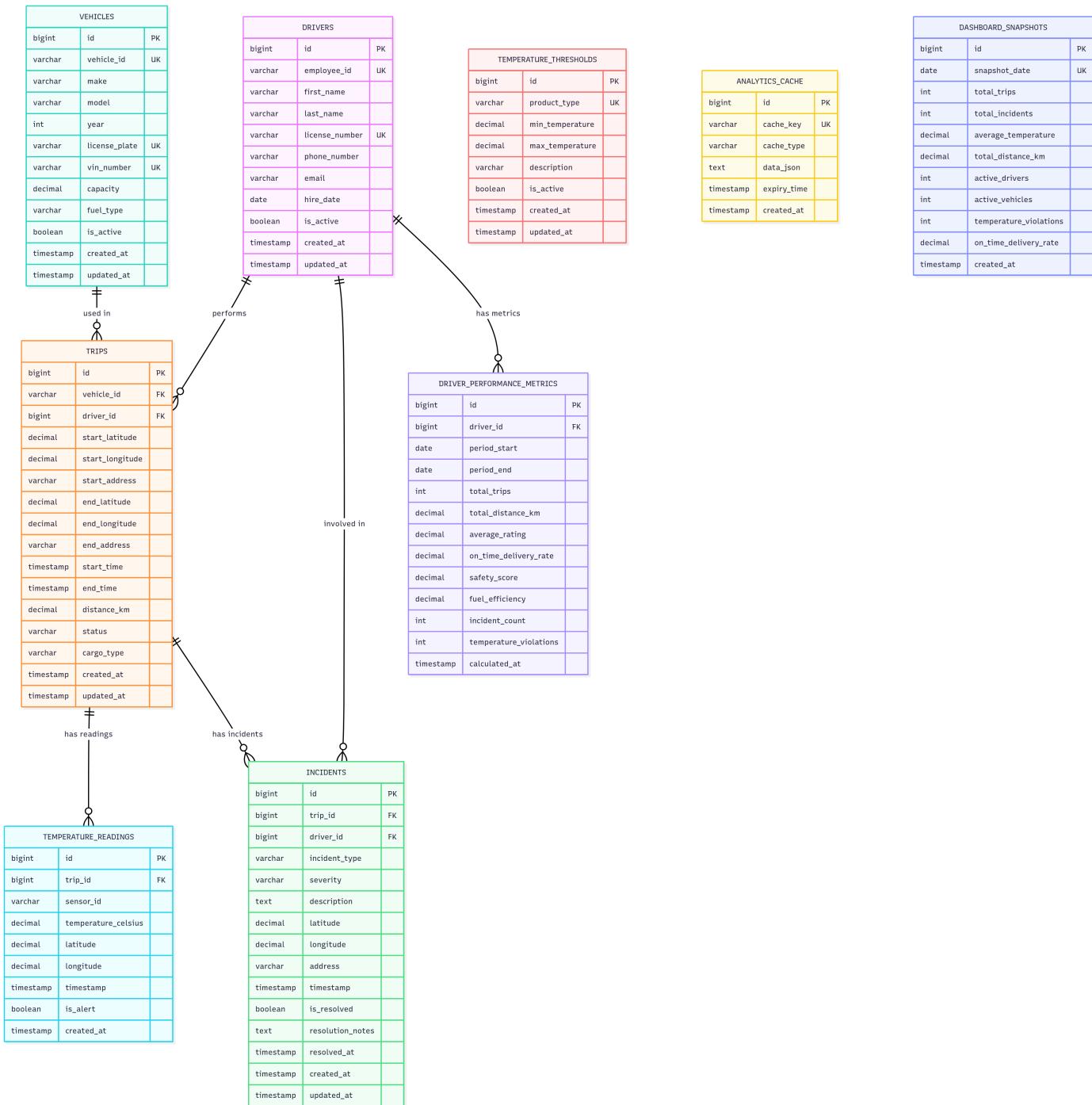
El diagrama del frontend Angular muestra los componentes especializados para visualización de datos. Los chart components renderizan gráficos interactivos usando librerías como Chart.js o D3.js, mientras que dashboard components gestionan la composición y layout de widgets. Los services manejan la comunicación con APIs de datos y el cache local de métricas para optimizar rendimiento.

Mobile - Visualization Analytics Domain Layer Class Diagram



La aplicación móvil Flutter prioriza visualizaciones optimizadas para pantallas pequeñas. Los componentes incluyen widgets responsivos y gráficos touch-friendly. El state management a través de BLoC coordina la actualización de datos en tiempo real y gestiona el cache local para funcionalidad offline, permitiendo consulta de métricas básicas sin conectividad.

4.2.8.6.2. Bounded Context Database Design Diagram



El diseño de base de datos del módulo Analytics está optimizado para consultas analíticas y agregaciones. Las tablas principales (DASHBOARDS, WIDGETS, REPORTS) mantienen configuraciones de usuario, mientras que las tablas de métricas están desnormalizadas para consultas rápidas. Se incluyen índices especializados para consultas temporales y agregaciones frecuentes.

4.2.9. Bounded Context: Merchant

4.2.9.1. Domain Layer

Entidades Principales

Merchant (Aggregate Root)

- Propósito:** Representa al comercio/cliente (shipper) que utiliza CargaSafe; centraliza identidad, estado, contactos, ubicaciones y métodos de pago.
- Atributos principales:**
 - id:** Identificador único
 - name:** Razón social / nombre comercial
 - taxID:** Identificador fiscal (p. ej., RUC)
 - email:** Email de contacto principal
 - status:** ACTIVE | SUSPENDED
 - primaryAddress:** Dirección principal (VO)
 - createdAt, updatedAt:** Auditoría
- Métodos principales:**

- activate(), suspend(reason)
- updateProfile(name, email, taxId)
- changePrimaryAddress(address)
- setPrimaryContact(contact)

Contact (Entity)

- **Propósito:** Persona de contacto del Merchant (operaciones, facturación, admin).
- **Atributos principales:**
 - id
 - fullName
 - email
 - phone
 - role (ADMIN|BILLING|OPERATIONS)
 - isPrimary
- **Métodos principales:**
 - markAsPrimary()

Location (Entity)

- **Propósito:** Sedes/almacenes/puntos de entrega asociados al Merchant.
- **Atributos principales:**
 - id
 - name
 - address (VO)
 - latitude
 - longitude

PaymentMethod (Entity)

- **Propósito:** Medio de pago registrado por el Merchant.
- **Atributos principales:**
 - id
 - type (CARD|BANK)
 - card (VO)
 - externalId (PSP)
 - isDefault
- **Métodos principales:**
 - makeDefault()

Plan (Entity)

- **Propósito:** Plan comercial ofertado (precio, periodicidad, features).
- **Atributos principales:**
 - id
 - name
 - price (Money)
 - billingPeriod (MONTHLY|YEARLY)
 - features[]
 - active

Subscription (Aggregate Root)

Propósito: Suscripción del Merchant a un plan (estado y períodos).

- **Atributos principales:**
 - id
 - merchantId
 - plan
 - status (TRIALING|ACTIVE|PAST_DUE|CANCELED)
 - currentPeriod (Period)
 - cancelAt
 - externalId (PSP)
- **Métodos:**
 - startTrial(days)
 - activate(plan)
 - markPastDue()
 - cancel(at)
 - renew(nextPeriod)

Invoice (Entity)

- **Propósito:** Comprobante/cobro emitido por suscripción.

- **Atributos:**

- `id`
- `subscriptionId`
- `amountTotal (Money)`
- `status (DRAFT|OPEN|PAID|VOID)`
- `issuedAt`
- `dueAt`
- `paidAt`
- `externalId (PSP)`
- `pdfUrl`

- **Métodos:**

- `markPaid(at)`
- `voidInvoice(reason)`

WebhookEvent (Entity)

- **Propósito:** Persistir eventos entrantes del proveedor de pagos (auditoría y re-procesos).

- **Atributos:**

- `id`
- `provider (STRIPE|OTHER)`
- `eventType`
- `payload`
- `receivedAt`
- `processedAt`
- `status`
- `merchantId?`
- `subscriptionId?`
- `invoiceId?`

Value Objects

- **Email, Phone, Address** (`line1, line2, city, region, countryCode, postalCode`)
- **Money** (`value, currency: USD|EUR|PEN`)
- **Period** (`startAt, endAt`)
- **PaymentCard** (`brand, last4, expMonth, expYear`)
- **Enums:** `MerchantStatus, SubscriptionStatus, InvoiceStatus, PaymentMethodType, CurrencyCode, BillingPeriod`

Domain Services

- **MerchantOnboardingService:** Provisiona defaults (roles, settings) y contacto primario.
- **BillingService:** Crea suscripciones, genera facturas, aplica pagos.

Commands

- **CreateMerchantCommand**
- **UpdateMerchantProfileCommand**
- **SetPrimaryContactCommand**
- **CreateSubscriptionCommand**
- **CancelSubscriptionCommand**
- **MarkSubscriptionPastDueCommand**
- **GenerateInvoiceCommand**
- **ApplyPaymentCommand**
- **SetDefaultPaymentMethodCommand**

Queries

- **GetMerchantByIdQuery**
- **SearchMerchantsQuery**
- **GetMerchantContactsQuery**
- **GetMerchantLocationsQuery**
- **GetPaymentMethodsQuery**
- **GetSubscriptionsByMerchantQuery**
- **GetInvoicesBySubscriptionQuery**

Events

- **MerchantCreatedEvent**

- **MerchantSuspendedEvent**
- **SubscriptionActivatedEvent**
- **SubscriptionCanceledEvent**
- **SubscriptionPastDueEvent**
- **InvoiceGeneratedEvent**
- **InvoicePaidEvent**
- **PaymentMethodSetDefaultEvent**

4.2.9.2. Interface Layer

Controllers Principales

MerchantController

- GET `/merchants/{id}`: Obtiene Merchant
- POST `/merchants`: Crea Merchant
- PUT `/merchants/{id}`: Actualiza perfil
- POST `/merchants/{id}/suspend`: Suspende Merchant

ContactController

- GET `/merchants/{id}/contacts`: Lista contactos
- POST `/merchants/{id}/contacts`: Crea contacto
- PUT `/merchants/{id}/contacts/{contactId}`: Actualiza contacto
- POST `/merchants/{id}/contacts/{contactId}/primary`: Marca como primario

LocationController

- GET `/merchants/{id}/locations`: Lista ubicaciones
- POST `/merchants/{id}/locations`: Crea ubicación
- PUT `/merchants/{id}/locations/{locationId}`: Actualiza ubicación
- DELETE `/merchants/{id}/locations/{locationId}`: Elimina

PaymentMethodController

- GET `/merchants/{id}/payment-methods`
- POST `/merchants/{id}/payment-methods`
- POST `/merchants/{id}/payment-methods/{pmId}/default`
- DELETE `/merchants/{id}/payment-methods/{pmId}`

SubscriptionController

- GET `/merchants/{id}/subscriptions`
- POST `/merchants/{id}/subscriptions`: Crea/activa
- POST `/subscriptions/{subId}/cancel`: Cancela
- GET `/subscriptions/{subId}`: Detalle

InvoiceController

- GET `/subscriptions/{subId}/invoices`
- GET `/invoices/{invoiceId}`
- POST `/invoices/{invoiceId}/apply-payment`
- GET `/invoices/{invoiceId}/pdf`: Descarga

WebhookController

- POST `/webhooks/payments`: Receptor de webhooks del PSP (validación de firma, encolado, idempotencia)

4.2.9.3. Application Layer

Command Services

MerchantCommandService

- Maneja creación/actualización de Merchant

- Gestiona contactos, ubicaciones y método de pago por defecto
- Encola eventos de auditoría (MerchantCreated/Suspended)

SubscriptionCommandService

- Alta/cancelación/renovación de suscripciones
- Transiciones de estado (`TRIALING` → `ACTIVE` → `PAST_DUE` → `CANCELED`)
- Coordinación con PSP (crear/cancelar suscripción)

BillingCommandService

- Generación de facturas, aplicación de pagos
- Emisión de eventos `InvoiceGenerated` y `InvoicePaid`

Query Services

MerchantQueryService

- Búsquedas y lecturas optimizadas de Merchant/Contacts/Locations/PaymentMethods

BillingQueryService

- Consultas de suscripciones e invoices (paginadas, por periodo/estado)

Event Handlers

PaymentWebhookEventHandler

- Procesa webhooks del PSP (idempotente)
- Sincroniza estados de `Subscription`/`Invoice`, publica eventos internos

SubscriptionActivatedEventHandler

- Reacciona a `SubscriptionActivated` (provisiona límites/planes, notifica)

InvoicePaidEventHandler

- Actualiza saldos, envía recibos, dispara notificaciones

4.2.9.4. Infrastructure Layer

Repositories

MerchantRepository (implementa `IMerchantRepository`)

- Persistencia de Merchants y relaciones (contacts, locations)
- Búsqueda por criterios (nombre, taxId, status)
- Caché de Merchants de alta frecuencia

PaymentMethodRepository (implementa `IPaymentMethodRepository`)

- Almacenamiento de métodos de pago, `isDefault`
- Resolución por `externalId` (PSP)

SubscriptionRepository (implementa `ISubscriptionRepository`)

- Persistencia de suscripciones y períodos
- Consultas por estado/merchant

InvoiceRepository (implementa `IInvoiceRepository`)

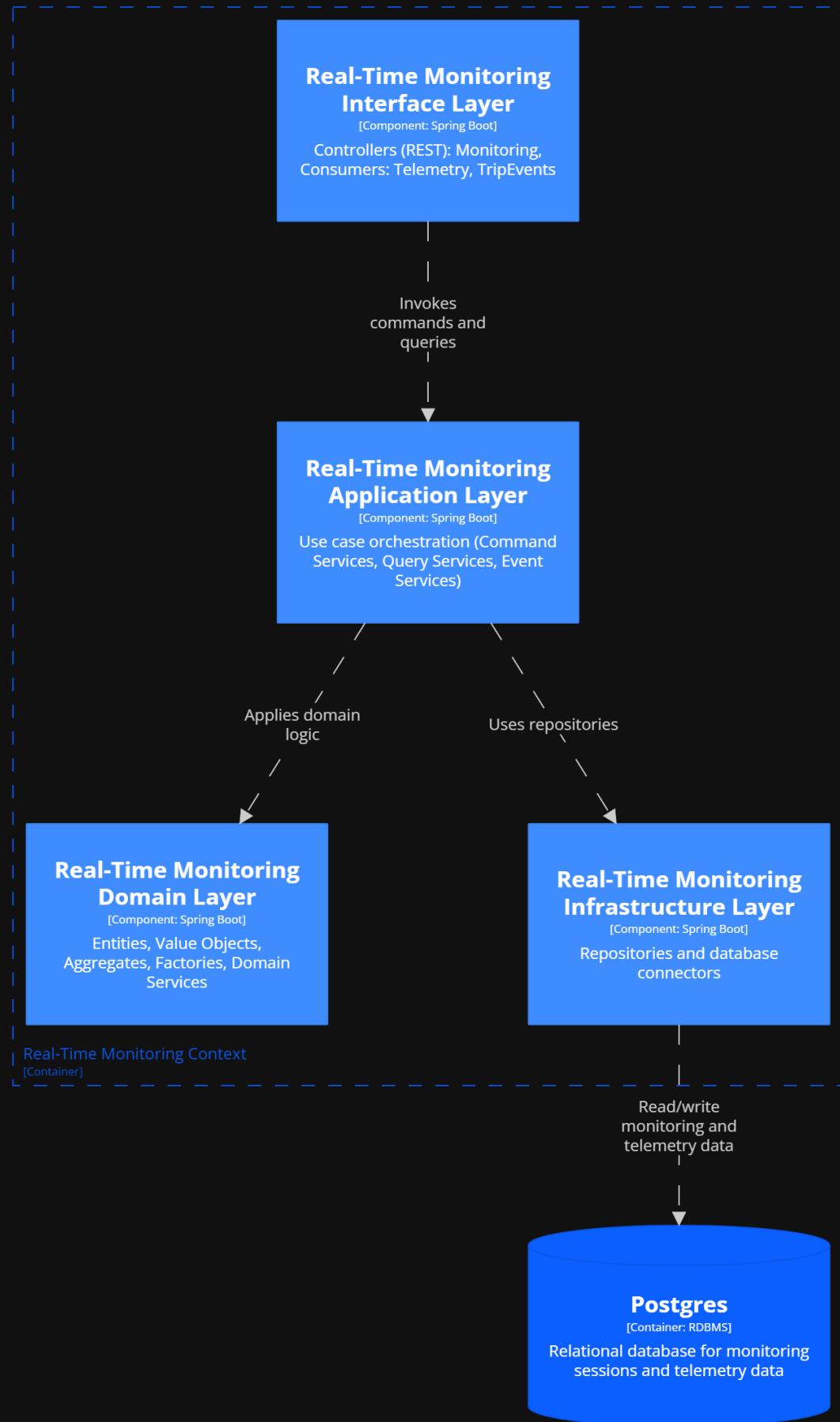
- Persistencia y búsqueda de facturas
- Gestión de pdfUrl y correlación externalId

WebhookEventRepository (implementa `IWebhookEventRepository`)

- Registro de eventos entrantes (trazabilidad, reintentos)
- Control de idempotencia

4.2.9.5. Bounded Context Software Architecture Component Level Diagrams

Diagrama de Componentes - Backend - Merchant



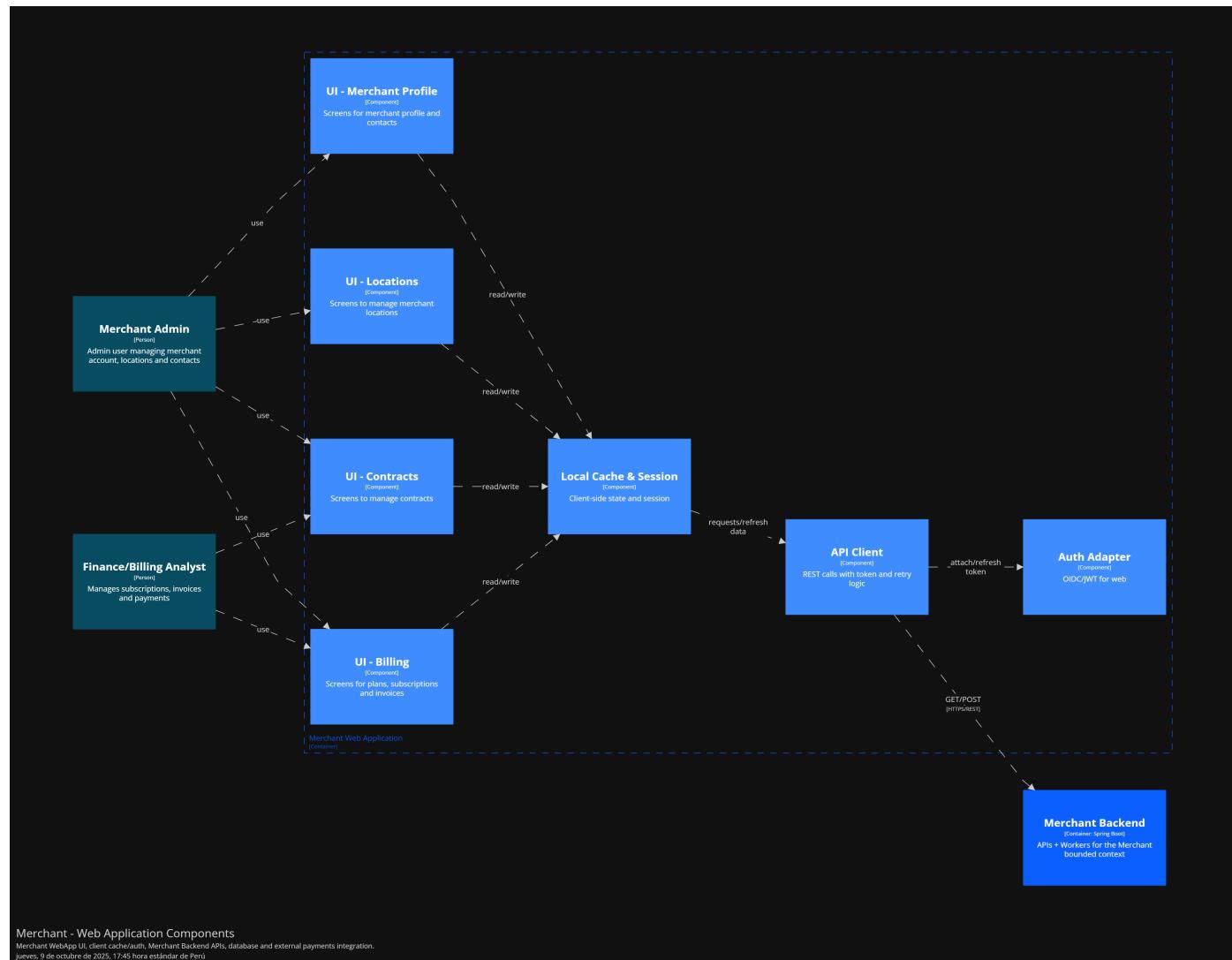
Real-Time Monitoring Context - Component Diagram

Internal layered components of the Real-Time Monitoring Context inside CargaSafe

internal layered components or the real-time monitoring context inside cargasare.
jueves, 9 de octubre de 2025, 17:40 hora estándar de Perú

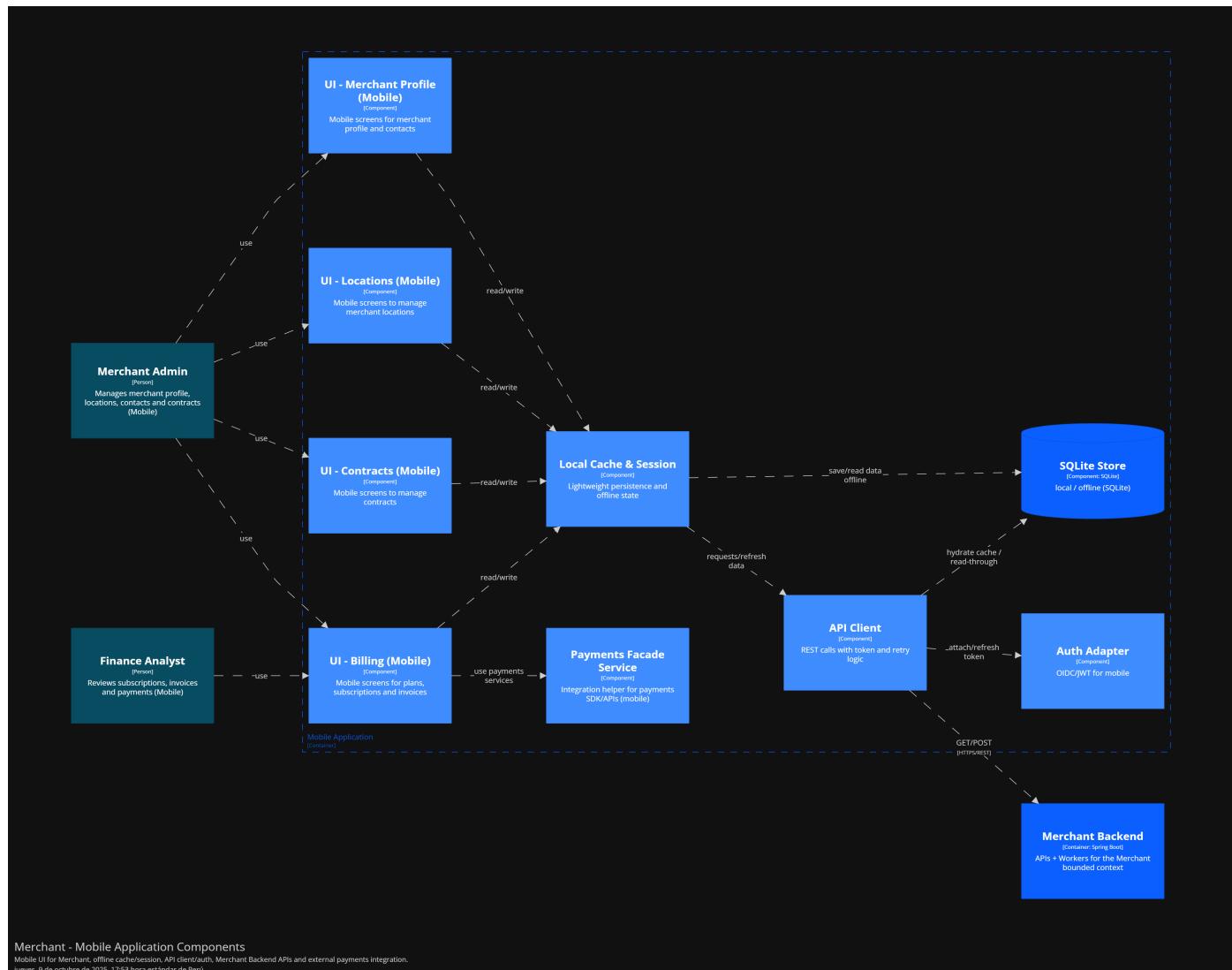
Este diagrama ilustra la arquitectura del bounded context de Visualization Analytics en el backend. Los controllers manejan requests relacionados con dashboards, reportes y análisis. Los servicios en Application Layer coordinan la lógica de negocio, mientras que los repositories optimizan el acceso a datos tanto transaccionales como de time-series para métricas y visualizaciones.

Diagrama de Componentes - Frontend Web - Merchant



El frontend web del módulo de analytics utiliza componentes especializados para visualización de datos. Los chart components renderizarán gráficos interactivos, mientras que dashboard components gestionarán la composición y layout de widgets. Los services manejan la comunicación con APIs de datos y el cache local de métricas.

Diagrama de Componentes - Mobile - Merchant

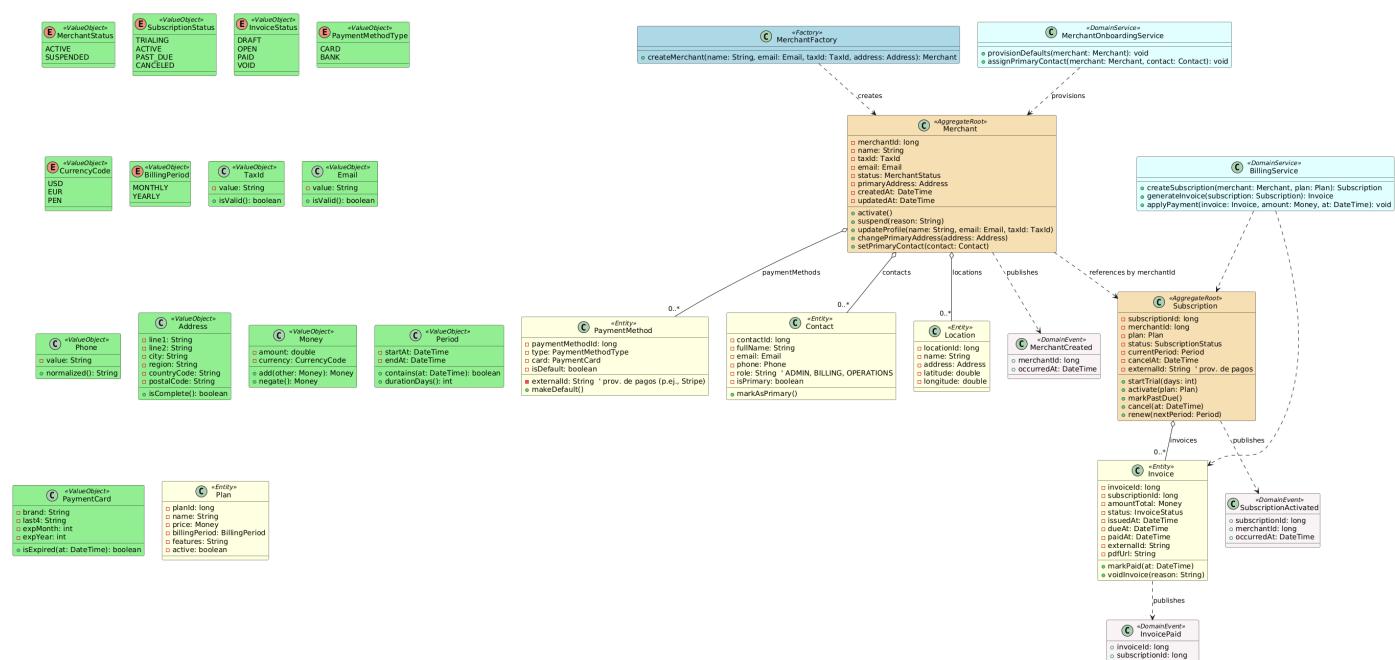


La aplicación móvil prioriza visualizaciones optimizadas para pantallas pequeñas. Los componentes incluyen widgets responsivos y gráficos touch-friendly. El state management atravéS de BLoC coordina la actualización de datos en tiempo real y gestiona el cache local para funcionalidad offline.

4.2.9.6. Bounded Context Software Architecture Code Level Diagrams

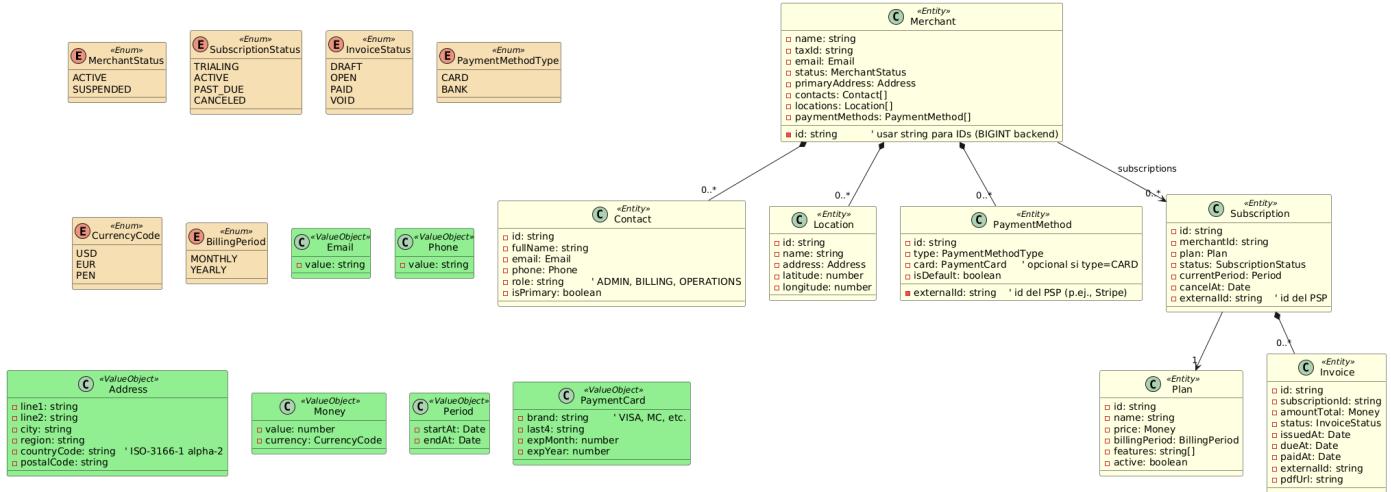
4.2.9.6.1. Bounded Context Domain Layer Class Diagrams

Backend - Merchant Domain Layer Class Diagram



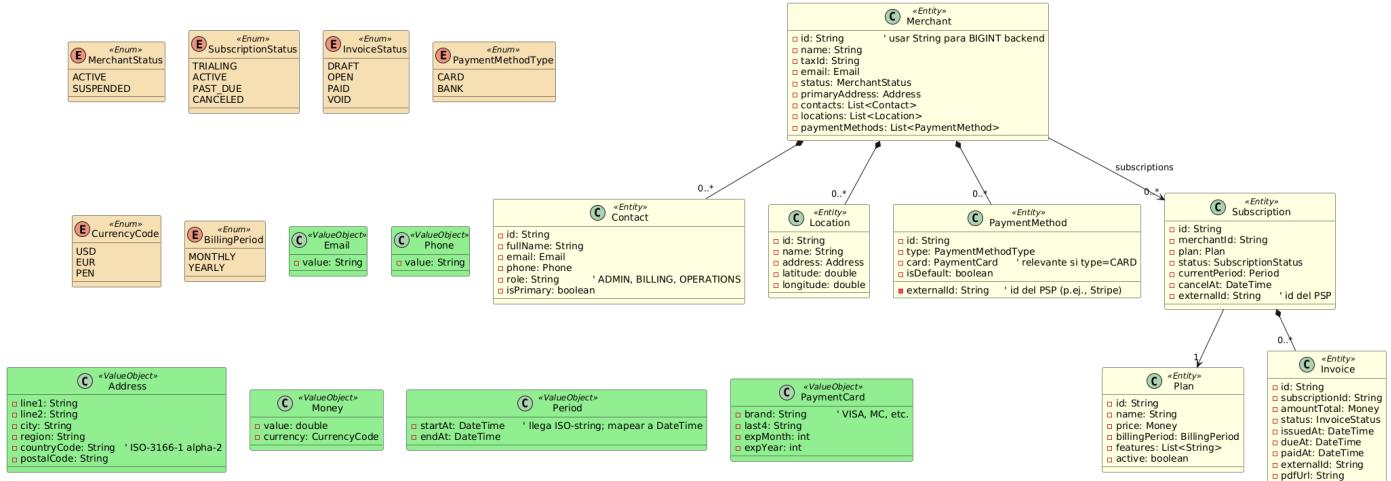
El diagrama de clases del backend de Merchant modela el dominio comercial y de facturación. Merchant es Aggregate Root para la identidad del cliente (contactos, ubicaciones, métodos de pago), mientras que Subscription es un aggregate root separado que representa la relación Plan→Merchant y su ciclo de vida. Invoice es entidad de billing asociada a Subscription. Se emplean Value Objects (Email, Address, Money, Period, PaymentCard) y Enums (MerchantStatus, SubscriptionStatus, InvoiceStatus, PaymentMethodType, CurrencyCode, BillingPeriod). Los Domain Services (p.ej., MerchantOnboardingService, BillingService) orquestan onboarding, creación de suscripciones y aplicación de pagos; los Domain Events (MerchantCreated, SubscriptionActivated, InvoicePaid) sincronizan estados con otros BCs y el PSP.

Frontend - Merchant Domain Layer Class Diagram



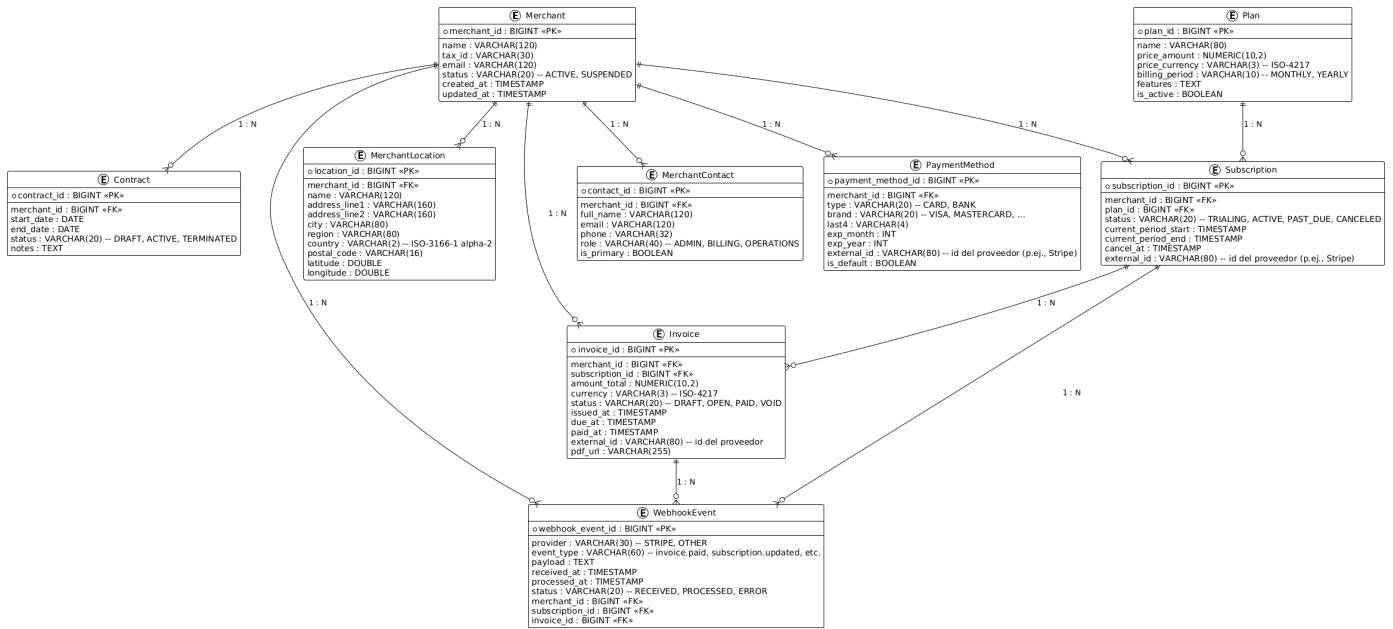
El diagrama del frontend (Web App) de Merchant refleja modelos de UI para perfil de merchant, contactos, ubicaciones, suscripciones e invoices. Los identificadores se manejan como string (por BIGINT en backend) y las fechas como Date. Se tipan VO y Enums (p.ej., CurrencyCode, BillingPeriod, SubscriptionStatus) para evitar errores. Los servicios de UI consumen APIs de Merchant/Billing y gestionan cache/estado (listas paginadas, filtros por estado/periodo) para una interacción rápida en administración.

Mobile - Merchant Domain Layer Class Diagram



El diagrama móvil (Flutter) de Merchant prioriza gestión ágil de perfil/ubicaciones/medios de pago y consulta de suscripciones e invoices. Los IDs se modelan como String y fechas como DateTime. Se reutilizan VO/Enums del dominio (p.ej., Money, Period, SubscriptionStatus). El state management (BLoC/Provider) coordina cache local y refresco de datos, permitiendo operaciones básicas offline (lectura) y sincronización cuando hay conectividad.

4.2.9.6.2. Bounded Context Database Design Diagram



El diseño de base de datos de Merchant está orientado a datos transaccionales con trazabilidad de billing e integración con el PSP. Tablas principales:

- MERCHANTS (identidad, estado, dirección principal),
 - CONTACTS, LOCATIONS, PAYMENT_METHODS (con external_id del PSP y is_default),
 - PLANS (precio amount/currency, billing_period),
 - SUBSCRIPTIONS (estado, current_period_start/end, cancel_at, external_id),
 - INVOICES (monto, estado, issued_at/due_at/paid_at, external_id, pdf_url),
 - WEBHOOK_EVENTS (payload, provider, event_type, received_at/processed_at, status, claves de correlación).

Se incluyen índices por merchant_id, estado y rangos de fechas; claves foráneas para integridad; y idempotencia en WEBHOOK_EVENTS para procesar de forma segura los webhooks del proveedor de pagos.

Capítulo V: Solution UI/UX Design

5.1. Style Guidelines.

5.1.1. General Style Guidelines.

El diseño visual del proyecto CargaSafe se basa en un sistema de estilos que prioriza la claridad, la coherencia y la accesibilidad. La identidad visual se sustenta en una paleta de colores cálidos y confiables, encabezada por el color naranja (#F5821F) como tono primario, que transmite energía, dinamismo y cercanía con el usuario. El color azul (#183D5C) se emplea como secundario, reforzando la percepción de confianza y profesionalismo. Se complementan colores de estado (verde para éxito, amarillo para advertencia y rojo para error), junto con una gama de grises para lograr jerarquía visual y contraste equilibrado.

La tipografía principal utilizada es Source Sans 3, seleccionada por su legibilidad y estilo moderno. Se aplica en distintas jerarquías tipográficas que van desde los títulos (Heading 1 – 56px) hasta los textos pequeños (Small Text – 14px), garantizando consistencia en todos los tamaños de pantalla.

El lenguaje visual del proyecto combina un tono formal y profesional, pero con una interfaz accesible y amigable. Se emplean botones con esquinas suavemente redondeadas, íconos derivados de la librería Material Design, y espaciados definidos en múltiplos de 4px para mantener uniformidad y ritmo visual en cada componente. En cuanto a componentes interactivos, se establecen directrices claras para botones, selectores, campos de texto e iconografía, priorizando la usabilidad y el reconocimiento inmediato de acciones. Los botones principales mantienen un color sólido naranja para llamadas a la acción (CTA), mientras que los secundarios y estados hover o active usan variantes más claras u opacas del mismo tono.

Colors

Brand Colors

- Primary / #F5821F
- Secondary / #183D5C

State Colors

- Info / #2F98ED
- Success / #27AE60
- Warning / #E74C3C
- Error / #E65757

Black Colors

- Black 1 / #000000
- Black 2 / #111111
- Black 3 / #202020
- White / #FFFFFF

Grey Colors

- Gray 1 / #333333
- Gray 2 / #4F4F4F
- Gray 3 / #636363
- Gray 4 / #808080
- Gray 5 / #C0C0C0

Typography

Source Sans 3

Name	Font size	Line Height
Heading 1	56 px	61.6 px
Heading 2	48 px	52.8 px
Heading 3	40 px	44 px
Heading 4	32 px	35.2 px
Heading 5	24 px	26.4 px
Heading 6	20 px	22 px

Selectors

Dropdown

Checkbox

Radio Button

Toggle

Tabs

Date Picker

Buttons

Color

Size

Full-width

Variants

5.1.2. Web, Mobile and IoT Style Guidelines.

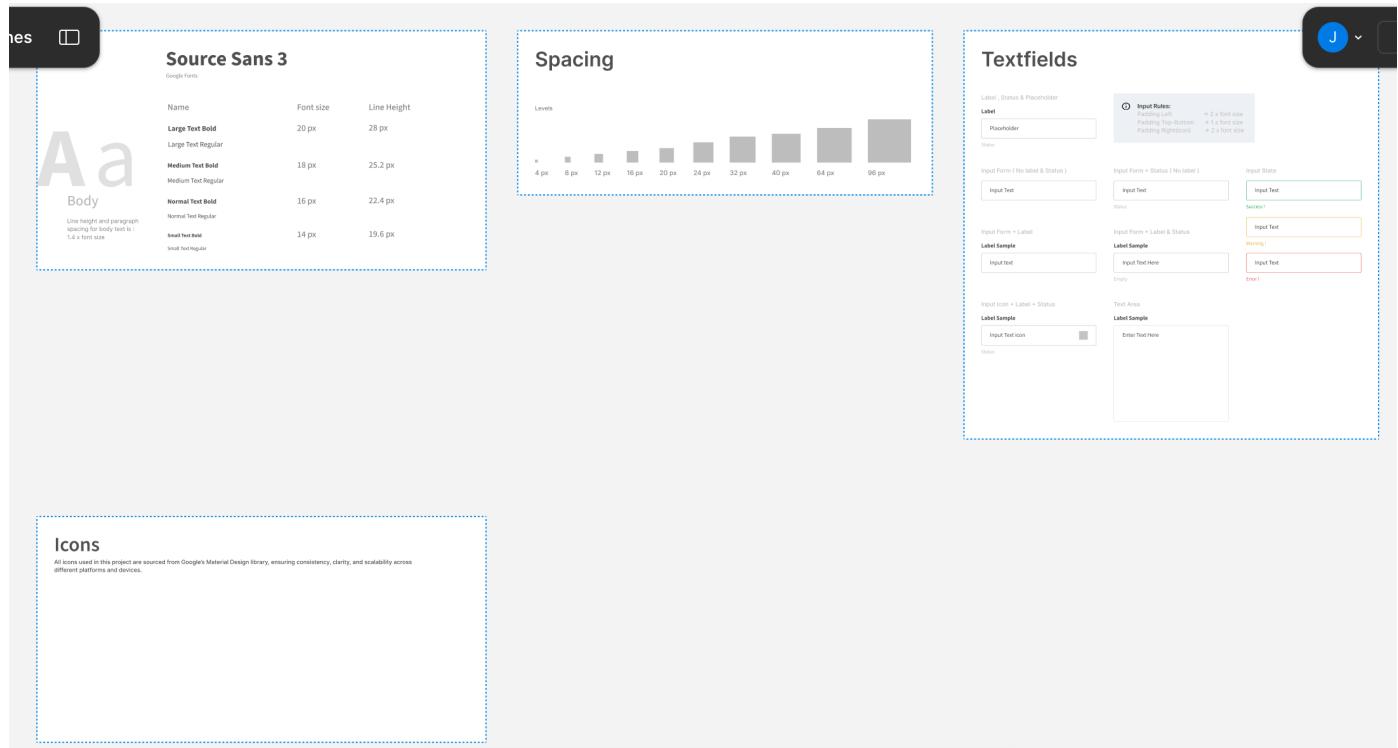
El diseño adaptable del ecosistema CargaSafe se rige por un enfoque responsive, garantizando que la experiencia de usuario sea fluida y consistente en dispositivos web, móviles y entornos IoT.

En la versión web, se prioriza la disposición horizontal de los elementos, con paneles y tarjetas informativas que aprovechan el espacio y mantienen una jerarquía visual clara. Los márgenes, paddings y tamaños tipográficos se ajustan automáticamente para conservar legibilidad en pantallas amplias.

Para la versión móvil, la interfaz mantiene los mismos principios visuales, pero reestructura los componentes en una distribución vertical optimizada para la interacción táctil. Se emplean botones de mayor tamaño, espaciados más amplios y textos ajustados a resoluciones menores, conservando la armonía visual del sistema.

En el contexto IoT, la guía se orienta a la presentación de datos de monitoreo, paneles de control y visualización de alertas. Se mantiene la misma paleta de colores y tipografía, asegurando que los gráficos, indicadores y notificaciones se perciban con claridad en pantallas de control o dispositivos conectados.

Estas directrices en conjunto garantizan que CargaSafe mantenga una identidad visual sólida, coherente y escalable a través de todos sus entornos digitales, reforzando la percepción de una plataforma tecnológica moderna, confiable y centrada en el usuario.



5.2. Information Architecture.

5.2.1. Organization Systems.

En esta sección se define cómo se estructura la información en CargaSafe según el tipo de contenido, la tarea y la audiencia. Se combinan sistemas visuales (jerárquica, secuencial, matricial) y esquemas de categorización (alfabético, cronológico, por tópicos y por audiencia).

Sistemas visuales

- **Jerárquica (visual hierarchy)**
 - Dónde: Dashboard; Fleet → Vehicles/Devices (listas); Alerts (lista).
 - Regla: lo crítico y reciente arriba; metadatos en segundo plano.
 - Patrones: H1/H2/H3 consistentes; KPI cards; chips/badges de severidad; sort por criticidad por defecto.
- **Secuencial (step-by-step to accomplish)**
 - Dónde: Vincular dispositivo↔vehículo; Crear viaje (Trip); Onboarding.
 - Patrones: MatStepper; validación por paso; CTA único; resumen final.
- **Matricial (cruce de variables)**
 - Dónde: Operación (Estado×Severidad); Analítica (Vehículo×Regla); Reportes (Tiempo×KPI).
 - Patrones: tabla/board con filtros persistentes (chips); heatmap/badges.

Esquemas de categorización

- **Alfabético:** Vehicles por Plate (A-Z); Devices por Serial (A-Z); Rules por nombre.
 - Secundario si hay incidentes; criticidad manda.
- **Cronológico:** Alerts (`timestamp DESC`); Trips (Upcoming/In Progress/Completed); Telemetry (series temporales).
- **Por tópicos:** Help/Docs (Setup, Fleet, Trips, Monitoring, Alerts, Billing); Settings/Admin.
- **Según audiencia:** Fleet Manager, Dispatcher, Driver, Customer (visibilidad, lenguaje y CTA por rol/scope).

Matriz guía

Grupo	Organización visual	Categorización
Dashboard	Jerárquica	Por audiencia + por tópicos
Fleet → Vehicles	Jerárquica + Matricial (Status×Severity)	Alfabético (Plate) + por estado
Fleet → Devices	Jerárquica + Matricial	Alfabético (Serial) + disponibilidad
Trips	Secuencial (creación) + Jerárquica (list)	Cronológico
Alerts	Jerárquica	Cronológico + severidad
Monitoring/Telemetry	Matricial (Tiempo×KPI)	Cronológico
Settings/Rules	Jerárquica	Por tópicos (+ A-Z en listados)
Billing/Subscriptions	Secuencial + Jerárquica	Por audiencia (admin) + cronológico

5.2.2. Labeling Systems.

En esta sección se establecen los criterios de rotulado para representar los datos con simplicidad y consistencia, definiendo etiquetas mínimas, CTAs, filtros y mensajes de estado, junto con pautas de accesibilidad.

Convenciones de nombres y UI

- **Entidades:** Vehicle, Device, Trip, Alert, Rule, Customer, User.
- **Campos clave (listas):**
 - Vehicles: Plate, Type, Capabilities, Status (Available/Busy/Out of Service).
 - Devices: Serial, Model, Health, Attached To, Status.
 - Alerts: Severity (Critical/Major/Minor), Rule, Vehicle, Timestamp, State (Open/Acknowledged/Resolved).
- **Acciones (CTA):** Add Vehicle, Edit, Delete, Attach Device, Set Available, Set Out of Service, Acknowledge, Export.
- **Filtros comunes:** Status, Severity, Type, Date Range, Assigned/Unassigned.
- **Mensajes de estado:** No vehicles found, No alerts in the selected range, Vehicle created successfully, Failed to attach device.

Asociaciones

Etiqueta / CTA	¿Qué representa?	Asociación mental del usuario
Contact	Ir a información de contacto (Landing)	"Aquí encuentro email/teléfono/redes"
Attach Device	Vincular un dispositivo a un vehículo	"Emparejar Serial con Plate"
Acknowledge	Marcar una alerta atendida	"Se registró responsable; cambia el estado"
Export	Descargar tabla filtrada	"Obtener CSV/Excel de lo que veo"

Accesibilidad

- `aria-label` específico en botones ("Acknowledge alert #123").
- Imágenes decorativas con `alt=""` y `aria-hidden="true"`.
- Notificaciones con `aria-live="polite"`.

Estilo

- Inglés; Title Case en títulos, Sentence case en descripciones.
- Unidades explícitas (°C, km/h).
- Chips de severidad con texto + color (no solo color).

5.2.3. SEO Tags and Meta Tags

En esta sección se documentan las SEO Tags y Meta Tags de la Landing Page y la WebApp —Title, Description, Keywords y Author— además de Open Graph y Twitter, mostrando los valores como código para su inclusión en los archivos HTML.

Landing Page

Tag	Ejemplo (inline)
Title	<title>CargaSafe Smart Monitoring for Reliable Transportation</title>
Description	<meta name="description" content="Real-time telemetry, smart alerts, and fleet visibility.">
Keywords	<meta name="keywords" content="fleet monitoring, cold chain, IoT telemetry, logistics alerts, geofencing, temperature monitoring">
Author	<meta name="author" content="CargaSafe Team">

Tag	Ejemplo (inline)
OG Title	<meta property="og:title" content="CargaSafe – Smart Monitoring for Reliable Transportation">
OG Description	<meta property="og:description" content="Monitor your fleet in real-time with alerts, telemetry and reports.">
Twitter Card	<meta name="twitter:card" content="summary_large_image">
Favicon	<link rel="icon" href="assets/logo.png" type="image/png">

WebApp – Fleet → Vehicles

Tag	Ejemplo (inline)
Title	<title>Fleet – Vehicles CargaSafe</title>
Description	<meta name="description" content="Manage vehicles: status, assignments and device attachments.">
Keywords	<meta name="keywords" content="fleet vehicles, status, device attachment, maintenance, logistics">
Author	<meta name="author" content="CargaSafe Team">

5.2.4. Searching Systems.

En esta sección se especifican los sistemas de búsqueda que habilitan la localización rápida de información: búsqueda global y por módulo, filtros y ordenamientos, operadores mínimos, criterios de rendimiento y métricas de éxito.

Patrones

- Búsqueda global (header): por Plate, Serial, Trip ID, Rule (typeahead + "recent searches").
- Búsqueda local (por módulo): caja de búsqueda sobre la tabla (incremental).
- Filtros (facetas): Status, Severity, Type, Assigned, Date Range.
- Ordenamiento: por defecto criticidad/reciente; alternativo A-Z, Health, Last Seen.

Operadores mínimos

- Texto exacto: "ABC-123"
- Prefijo/propiedad: plate:ABC, serial:SN-
- Rango de fechas: date:2025-10-01..2025-10-09

Rendimiento y UX

- Debounce ~300 ms; indicador de carga; vacíos explicativos.
- Paginación con contador; sticky filters; accesible por teclado; aria-live para resultados.

Métricas

- Tiempo a primer resultado local < 500 ms.
- Éxito en primera búsqueda > 85% (pruebas moderadas).

5.2.5. Navigation Systems.

En esta sección se describen los sistemas de navegación que guían a los usuarios por la Landing Page y la WebApp, abarcando navegación global, local y contextual, comportamientos responsivos, flujos guiados, guardas y criterios de aceptación.

Estructura

- Global: Dashboard, Fleet (Vehicles, Devices), Trips, Alerts, Monitoring, Settings, Billing. (Visibilidad por rol/scope)
- Local: tabs por estado (e.g., Trips: Upcoming / In Progress / Completed).
- Contextual: acciones en fila/detalle (Edit, Attach Device, Acknowledge, Export).
- Breadcrumbs (opcional): Fleet / Vehicles / ABC-123.

Comportamientos

- Responsive:
 - Móvil → menú hamburguesa; tablas → cards con CTAs visibles.
 - Desktop → sidebar fija; tablas con sort/paginator.
 - LandingPage → navbar.
- Flujos guiados (secuencial): steppers con validación por paso (Create Trip, Attach Device).

- Guardas: canActivate (auth/rol) y canDeactivate (evitar pérdida de datos).
- Estado de UI: preservar filtros/scroll al volver; estados vacíos con CTA.

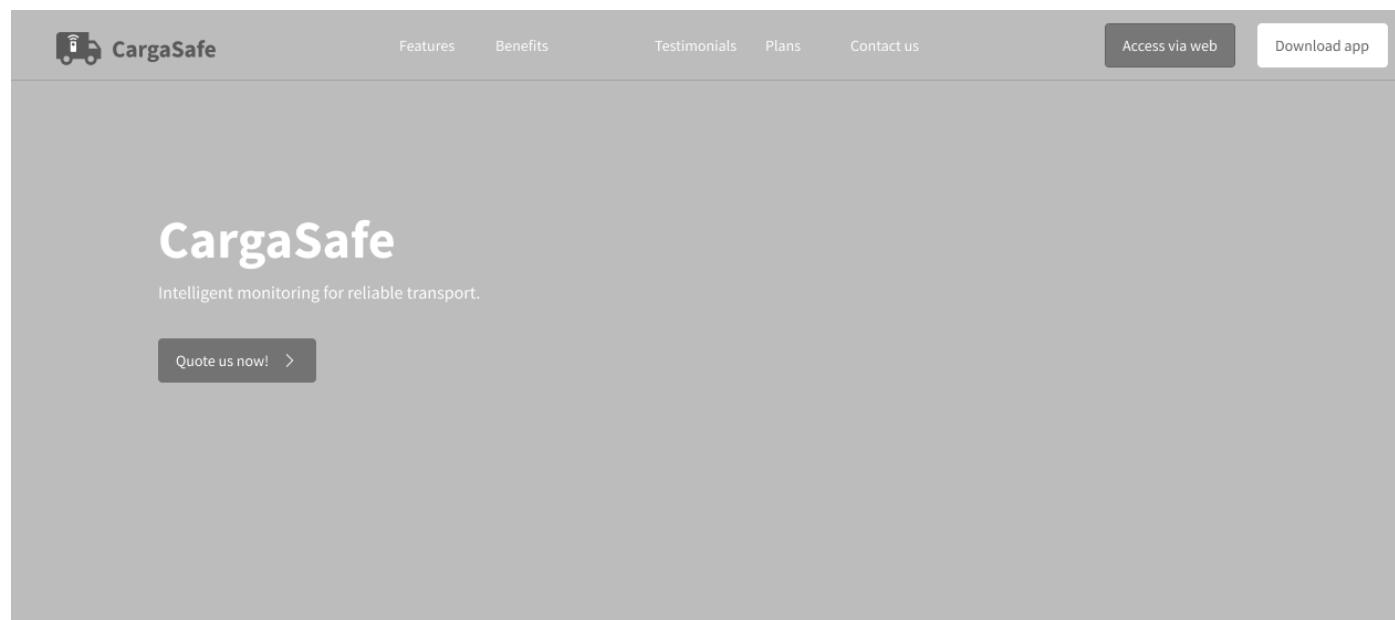
Criterios de aceptación

- Llegar de Dashboard a Vehicle Detail ≤ 2 clics.
- Completar Create Trip ≤ 60 s (usuario entrenado).
- Sin 404 visibles; Not Found con enlace de retorno.

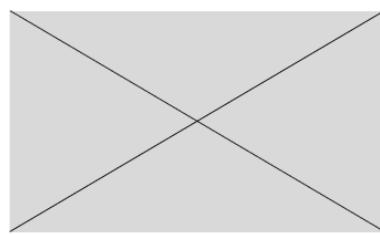
5.3. Landing Page UI Design.

La propuesta de interfaz de usuario (UI) para el Landing Page de CargaSafe se fundamenta en una arquitectura de información clara y jerárquica, que guía al usuario desde la comprensión del servicio hasta la conversión final. Se priorizó una navegación intuitiva mediante un menú superior fijo que agrupa secciones clave (Features, Benefits, Testimonials, Plans y Contact), permitiendo un flujo de lectura progresivo y coherente. El uso de espacios amplios, tipografía legible y un contraste cromático basado en tonos naranjas y grises traduce las decisiones de diseño hacia una experiencia confiable, moderna y alineada con la identidad de marca de CargaSafe. Cada bloque de contenido responde a un objetivo informativo o de conversión, aplicando principios de jerarquía visual, consistencia, alineación y accesibilidad para asegurar una comunicación efectiva con usuarios de distintos perfiles.

5.3.1. Landing Page Wireframe.

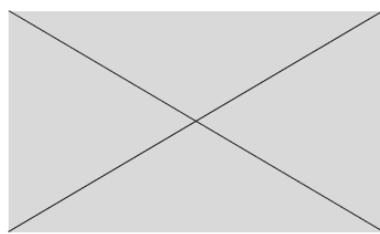


FEATURES What does CargaSafe do?



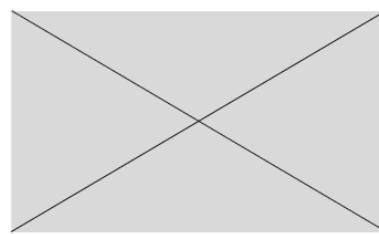
Real-time monitoring

View the location, temperature, and status of your cargo from any device. Stay informed minute by minute with accurate data.



Smart Alerts

Receive immediate notifications when your cargo is at risk (temperature fluctuations, delays, unauthorized openings). Act before it's too late.



Centralized Management

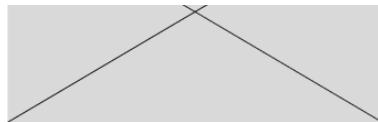
Manage all your shipments in a single platform. Access reports, history, and metrics to optimize your logistics chain.

BENEFITS What do you gain with CargaSafe?



Reduction of Operational Risks

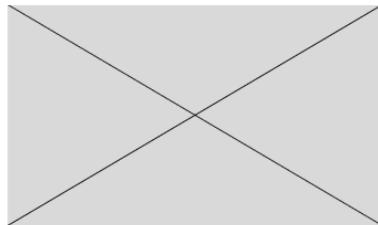
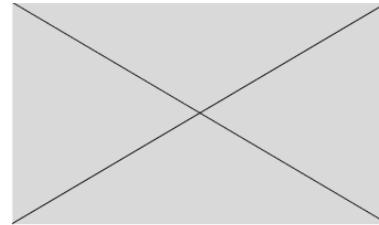
Detect incidents immediately and prevent losses due to failures, theft,...



or delays. Immediate response ensures operational integrity and protects the integrity of each shipment.

Logistics Cost Optimization

Reduce returns, complaints, and unnecessary waste. Real-time information enables better decisions that increase efficiency and boost profitability.



Greater Transparency and Commercial Confidence

Provides access to clear and verifiable information on every shipment. Full traceability builds greater trust, strengthens business relationships, and opens up new business opportunities.

TESTIMONIALS

What people say



LOGISTICS COORDINATOR
Andrea Salazar

With CargaSafe, we can monitor each shipment in real time and react to any alerts. This reduced risks and improved safety throughout the entire operation.



HEAD OF LOGISTICS
José Ramírez

The platform's transparency allows us to see where the cargo is at all times. We now have more control and fewer incidents during our deliveries.



RETAIL CLIENT
Claudia Ortega

It gives me confidence to know I receive alerts at every stage and can track my order in real time. CargaSafe guarantees punctuality and transparency.



INDEPENDENT CLIENT
Miguel Paredes

Before, I didn't know what was happening with my shipments. Now I can track them at all times. CargaSafe alerts give me peace of mind and complete confidence.

SUBSCRIPTIONS

Choose your plan

BASIC PLAN

Up to 2 vehicles

S/. .29/month

- ✓ Real-time monitoring
- ✓ Smart alerts
- ✓ Centralized management

[Choose plan](#)

STANDARD PLAN

Up to 5 vehicles

S/. .59/month

- ✓ Real-time monitoring
- ✓ Smart alerts
- ✓ Centralized management

[Choose plan](#)

PREMIUM PLAN

Unlimited vehicles

S/. .99/month

- ✓ Real-time monitoring
- ✓ Smart alerts
- ✓ Centralized management

[Choose plan](#)

Request your personalized quote

Leave us your details and an advisor will contact you with a tailored logistics proposal, no obligation, and with the security of CargaSafe.

Mail

First name Last name

Additional details

Submit request

DISTRIBUTION

Access the platform



Web application

Access directly from your browser, no downloads required.

[Use now](#)



iOS App

Download the app from the App Store and manage everything from your iPhone or iPad.

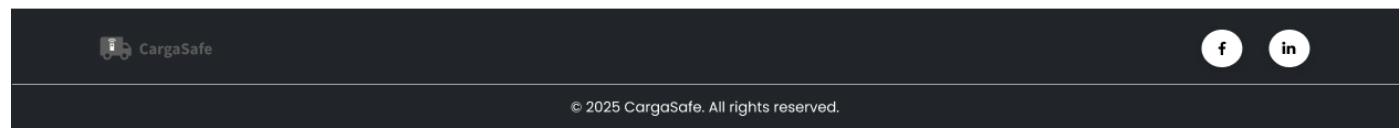
[Download on iOS](#)



Android App

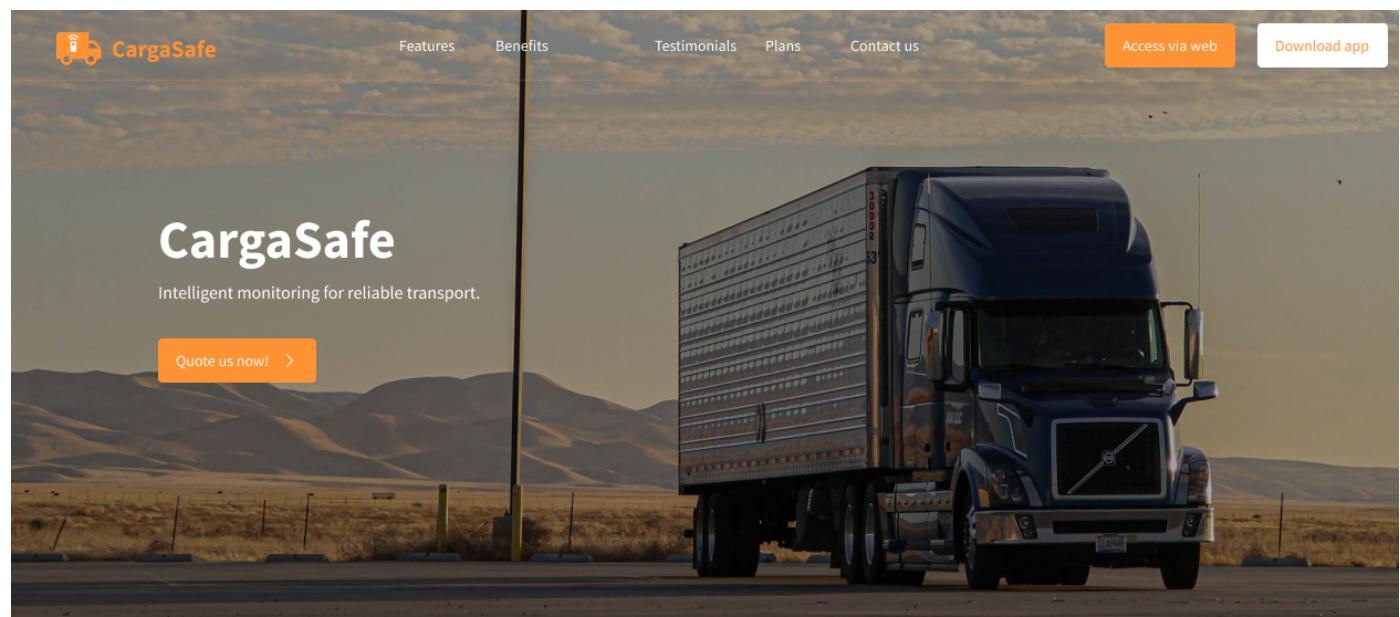
Available on Google Play for use on any Android device.

[Download on Android](#)



Los wireframes desarrollados para la versión de escritorio y móvil representan la estructura base del Landing Page, evidenciando la aplicación de principios de diseño como equilibrio, proximidad y alineación, además de una arquitectura de información secuencial que facilita la comprensión del servicio. En la versión desktop, se distribuyen los elementos en una cuadrícula de tres columnas para destacar los beneficios y planes de suscripción, mientras que en la versión mobile se adoptó una disposición vertical con prioridad en la legibilidad y accesibilidad táctil. El diseño inclusivo se garantiza mediante textos descriptivos, botones amplios y contrastes adecuados, favoreciendo la interacción de usuarios con distintas capacidades visuales o motoras. Asimismo, se emplearon convenciones reconocibles (íconos, etiquetas claras y jerarquías tipográficas) que mejoran la usabilidad y reducen la carga cognitiva del usuario. Ambas versiones se diseñaron bajo criterios responsive, garantizando una experiencia consistente y fluida independientemente del tamaño de pantalla o dispositivo utilizado.

5.3.2. Landing Page Mock-up.



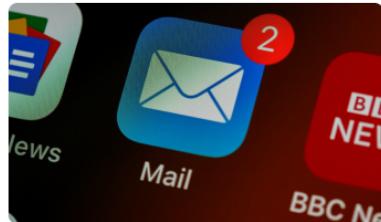
FEATURES

What does CargaSafe do?



Real-time monitoring

View the location, temperature, and status of your cargo from any device. Stay informed minute by minute with accurate data.



Smart Alerts

Receive immediate notifications when your cargo is at risk (temperature fluctuations, delays, unauthorized openings). Act before it's too late.



Centralized Management

Manage all your shipments in a single platform. Access reports, history, and metrics to optimize your logistics chain.

BENEFITS

What do you gain with CargaSafe?



Reduction of Operational Risks

Detect incidents immediately and prevent losses due to failures, theft, or delays. Immediate response ensures operational integrity and protects the integrity of each shipment.



Logistics Cost Optimization

Reduce returns, complaints, and unnecessary waste. Real-time information enables better decisions that increase efficiency and boost profitability.



Greater Transparency and Commercial Confidence

Provides access to clear and verifiable information on every shipment. Full traceability builds greater trust, strengthens business relationships, and opens up new business opportunities.

TESTIMONIALS

What people say



LOGISTICS COORDINATOR
Andrea Salazar

With CargaSafe, we can monitor each shipment in real time and react to any alerts. This reduced risks and improved safety throughout the entire operation.



HEAD OF LOGISTICS
José Ramírez

The platform's transparency allows us to see where the cargo is at all times. We now have more control and fewer incidents during our deliveries.



RETAIL CLIENT
Claudia Ortega

It gives me confidence to know I receive alerts at every stage and can track my order in real time. CargaSafe guarantees punctuality and transparency.



INDEPENDENT CLIENT
Miguel Paredes

Before, I didn't know what was happening with my shipments. Now I can track them at all times. CargaSafe alerts give me peace of mind and complete confidence.

SUBSCRIPTIONS

Choose your plan

BASIC PLAN

Up to 2 vehicles

S/. .29/month

- ✓ Real-time monitoring
- ✓ Smart alerts
- ✓ Centralized management

[Choose plan](#)**STANDARD PLAN**

Up to 5 vehicles

S/. .59/month

- ✓ Real-time monitoring
- ✓ Smart alerts
- ✓ Centralized management

[Choose plan](#)**PREMIUM PLAN**

Unlimited vehicles

S/. .99/month

- ✓ Real-time monitoring
- ✓ Smart alerts
- ✓ Centralized management

[Choose plan](#)**Request your personalized quote**

Leave us your details and an advisor will contact you with a tailored logistics proposal, no obligation, and with the security of CargaSafe.

[Submit request](#)

DISTRIBUTION

Access the platform



Web application

Access directly from your browser, no downloads required.

[Use now](#)

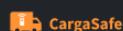
iOS App

Download the app from the App Store and manage everything from your iPhone or iPad.

[Download on iOS](#)

Android App

Available on Google Play for use on any Android device.

[Download on Android](#)

© 2025 CargaSafe. All rights reserved.



Los mock-ups finales traducen la estructura del wireframe en una experiencia visual completa, implementando el Design System definido para los productos digitales de CargaSafe, el cual contempla una paleta cromática corporativa (naranja, negro y blanco), tipografía moderna y componentes reutilizables (botones, cards, formularios y secciones de testimonios). En la versión de escritorio, se refuerza la identidad visual mediante imágenes contextuales del transporte de carga y un llamado a la acción destacado con alto contraste. En la versión móvil, los componentes se reestructuran de manera responsive, adaptando el tamaño, espacio y jerarquía visual para mantener legibilidad y accesibilidad sin sacrificar estética. La coherencia en íconos, tipografía y color refuerza la unidad del sistema, mientras que la disposición progresiva del contenido sigue la arquitectura de información planteada, conduciendo al usuario de la exploración del servicio hacia la conversión final de contacto o suscripción.

5.4. Applications UX/UI Design.

Esta sección presenta la propuesta visual y de interacción para las aplicaciones que constituyen la experiencia de usuario con los productos digitales de CargaSafe. El diseño de la interfaz y la experiencia de usuario se fundamentan en principios de usabilidad, accesibilidad y coherencia visual, considerando las necesidades específicas de los segmentos objetivo identificados: gestores de transporte y clientes finales. A través de wireframes, mockups y diagramas de flujo de usuario, se documenta el proceso de diseño iterativo que busca optimizar la eficiencia operativa, reducir la curva de aprendizaje y proporcionar una experiencia intuitiva tanto en entornos web como móviles. La propuesta de diseño integra los componentes visuales definidos en las Style Guidelines con patrones de navegación y arquitectura de información que facilitan el acceso rápido a funcionalidades críticas de monitoreo en tiempo real, gestión de alertas y generación de reportes, asegurando que cada interacción del usuario agregue valor tangible a sus operaciones logísticas.

5.4.1. Applications Wireframes.

Para la aplicación de Cargasafe, se d los bocetos de las pantallas que luego serán diseñadas más a detalle. A continuación, se presentan los wireframes de las pantallas de las aplicaciones web y móvil.

Web Application

Authentication

Subscriptions

Alerts

The screenshots show the 'Alerts' section of the CargaSafe platform. Both screens feature a sidebar with icons for Dashboard, Vehicles, Sensors, Trips, Alerts (selected), and Subscriptions. The main area displays two summary boxes: '1 Alerts today' with an exclamation mark icon and '3 resolved' with a checkmark icon. Below these are two cards: one for a High Temperature alert (ID #0001) and another for an Excessive vibration alert (ID #0002). A large table lists all alerts, including their ID, Type, Delivery Order ID, Status, Created date, Closed date, Actions, and Details. The table shows four alerts: #0001 (High Temperature, T342, status: Open, created 2025-01-05 10:45 AM, closed 2025-01-05 11:45 AM, resolved), #0002 (Excessive vibration, T343, status: Open, created 2025-01-05 11:20 AM, closed 2025-01-05 11:45 AM, resolved), #0003 (Energy Loss, T344, status: Open, created 2025-01-05 01:05 PM, closed 2025-01-05 01:14 PM, resolved), and #0004 (Trip Loss, T345, status: Open, created 2025-01-05 01:35 PM, closed 2025-01-05 01:50 PM, resolved). The bottom of each screen includes a 'Logout' button.

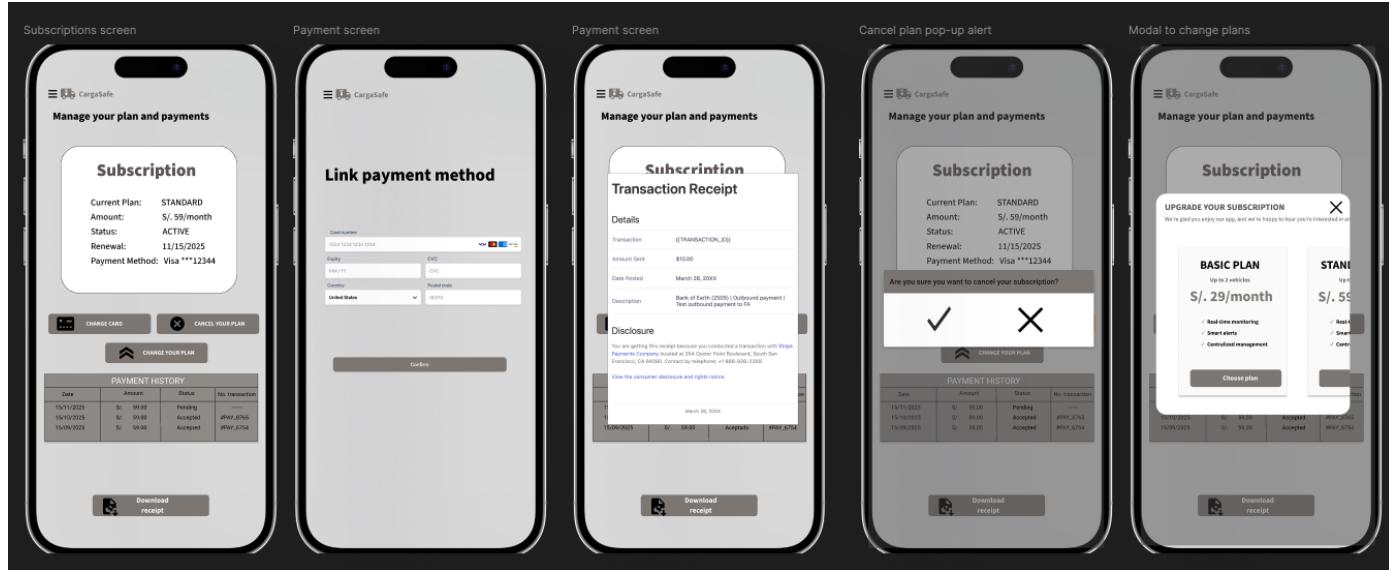
Mobile Application

Authentication

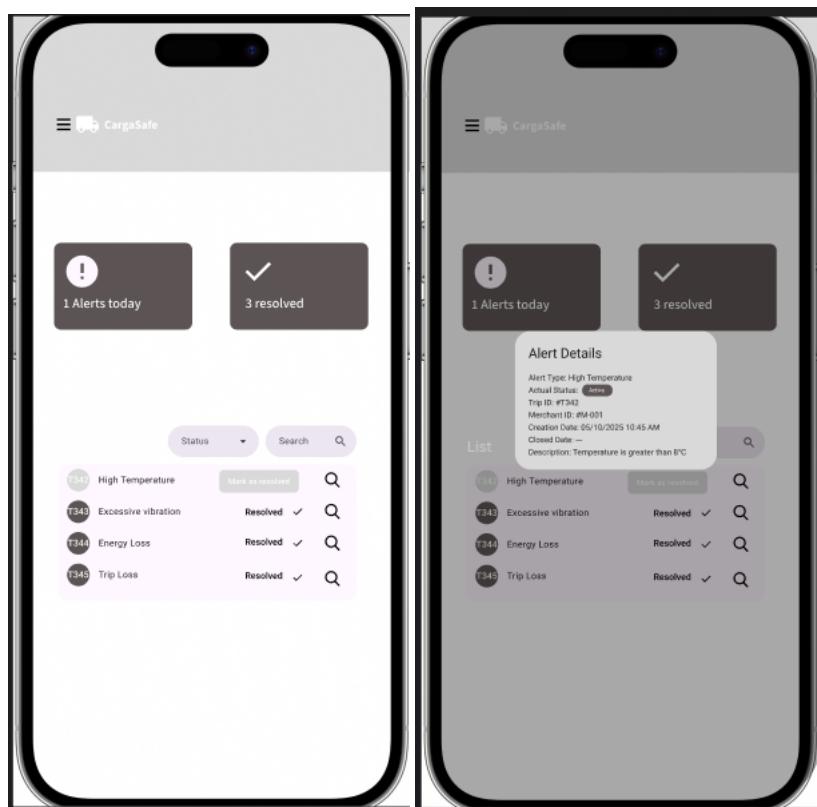
The screenshots illustrate the mobile application's authentication flow:

- Pantalla de inicio de sesión (Login Screen):** Shows a 'WELCOME BACK' message, fields for 'Email address' and 'Password', and checkboxes for 'Remember me' and 'Forgot your password?'. It also features a 'Sign In' button and a 'Don't have an account? Sign up here' link.
- Pantalla de registro (Registration Screen):** Shows a 'Create New Account' form with sections for 'Choose your segment' (Client or Shipping Company selected), 'Profile Info', 'Account Info', and 'Password'. It includes a checkbox for accepting terms and conditions and a 'Sign Up' button.
- Pantalla de registro (Registration Screen):** Similar to the previous registration screen, but with more detailed company information fields like 'Company Info' and 'Administrator Info'.
- Pantalla de recuperación de contraseña (Password Recovery Screen):** Shows a 'Password Recovery' form with a 'Forgot address?' field and a 'Send Recovery' button.
- Password recovery screen:** Displays a 6-digit OTP code for verification.
- Password recovery screen:** Shows a 'Create New Password' form with a 'New Password' field and a 'Reset Password' button.

Subscriptions



Alerts

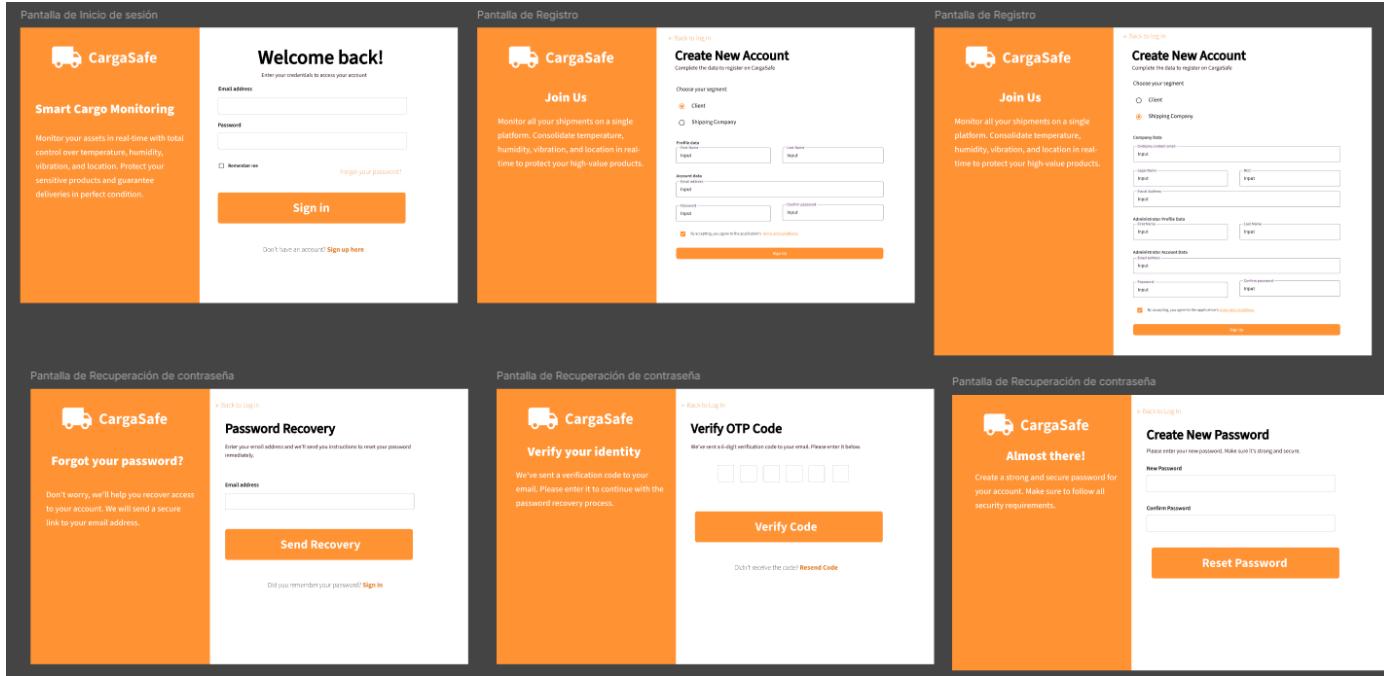


5.4.2. Applications Wireflow Diagrams.

5.4.2. Applications Mock-ups.

Web Application

Authentication



Subscriptions

The grid displays the following screens:

- Subscriptions screen:** Shows the CargaSafe logo and a sidebar with navigation items like Dashboard, Vehicles, Alerts, Subscriptions, and My company. The main area shows a "Manage your plan and payments" section with a "Subscription" card (Current Plan: STANDARD, Amount: S/. 59/month, Status: ACTIVE, Renewal: 11/15/2025) and buttons for Cancel your plan, Change your plan, and Change card.
- Payment form:** Shows the CargaSafe logo and a "Link payment method" section. It includes fields for Cardholder name, Expire, CVV, Country, and United States, with a "Search" button.
- Modal to change plans:** A modal window titled "Manage your plan and payments" showing three subscription options: BASIC PLAN (\$/. 29/month), STANDARD PLAN (\$/. 59/month), and PREMIUM PLAN (\$/. 99/month). Each plan has a list of features: BASIC (Real-time monitoring, Smart alerts, Consolidated management), STANDARD (Real-time monitoring, Smart alerts, Consolidated management), and PREMIUM (Real-time monitoring, Smart alerts, Consolidated management).
- Receipt screen:** Shows the CargaSafe logo and a sidebar with navigation items like Dashboard, Vehicles, Alerts, Subscriptions, and My company. The main area shows a "PAYMENT RECEIPT" section with a "Transaction Receipt" table (Details: Transaction #TRANSACTION_0001, Amount Sent: S/. 59.00, Date Paid: March 28, 2025, Description: Bank of America (Globe) Gobal Payments - Visa Electron Payment Card), a "TRANSACTION HISTORY" table (Transactions: 00 Pending, 00 Accepted, 00 Accepted), and a "Download receipt" button.
- Cancel plan pop-up alert:** A modal window titled "Manage your plan and payments" asking "Are you sure you want to cancel your subscription?" with a green checkmark icon and a red X icon.
- Modal for upgrading subscriptions:** A modal window titled "UPGRADE YOUR SUBSCRIPTION" showing the current subscription (STANDARD PLAN, S/. 59/month) and upgrade options: BASIC PLAN (\$/. 29/month), STANDARD PLAN (\$/. 59/month), and PREMIUM PLAN (\$/. 99/month). It lists features for each plan and includes "Choose plan" buttons.

Alerts

Alerts

1 Alerts today 3 resolved

ID	Type	Delivery Order id	Status	Created	Closed	Actions	Details
#0001	High Temperature	T342	Active	025 10:45	—	<button>Mark as resolved</button>	
#0002	Excessive vibration	T343	Active	025 11:20	11:45 AM	<button>Resolved ✓</button>	
#0003	Energy Loss	T344	Active	025 01:05	01:14 PM	<button>Resolved ✓</button>	
#0004	Trip Loss	T345	Active	025 01:35	01:50 PM	<button>Resolved ✓</button>	

Rows per page: 5 ▾ 1-4 of 10 < >

Mobile Application

Authentication

Login screen

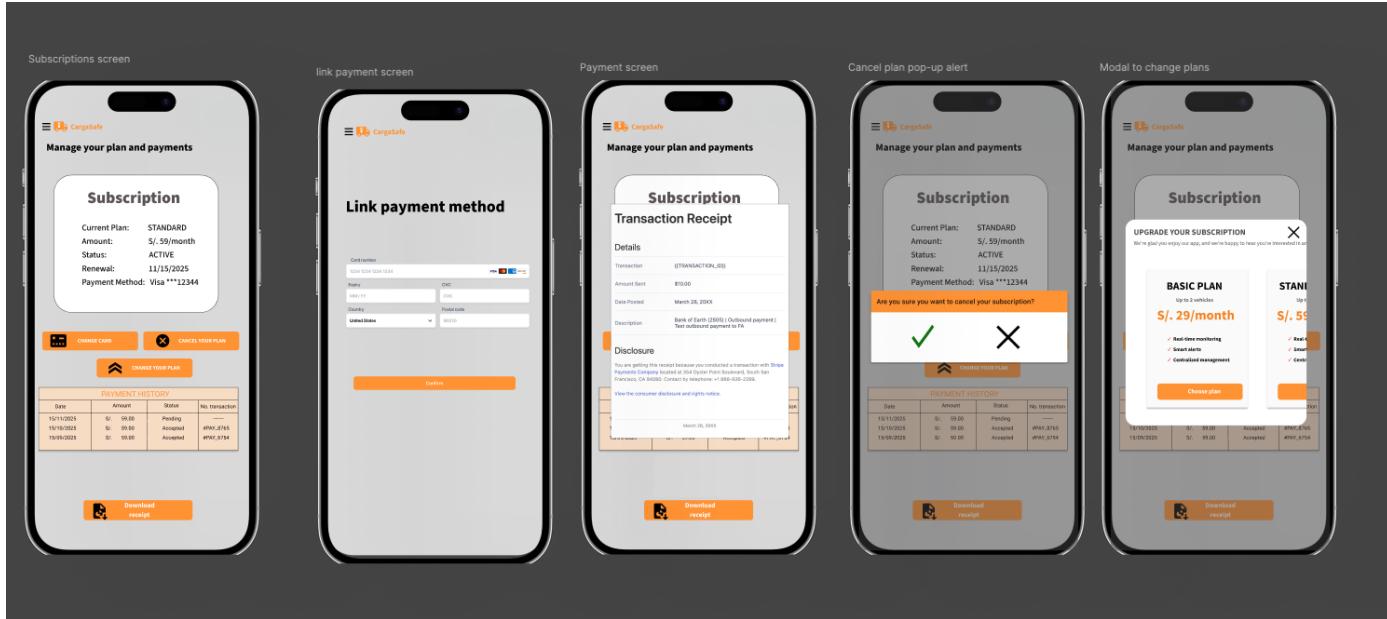
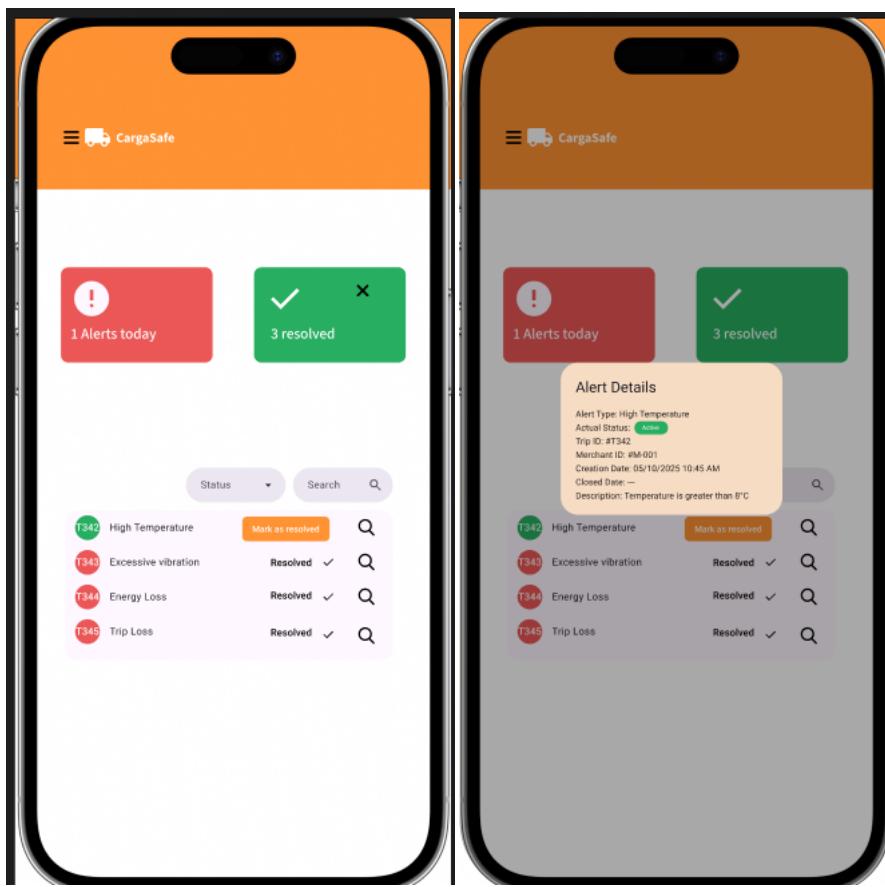
Registration screen

Registration screen

Password recovery screen

Password recovery screen

Subscriptions

**Alerts****5.4.3. Applications User Flow Diagrams.****Web Application - Cliente Final**

Registro de Usuario e Inicio de Sesión:

- User Goal: Como usuario, quiero registrarme en la plataforma, para acceder a mi cuenta y funcionalidades personalizadas.
- Descripción: El usuario se encuentra en la pantalla de inicio de sesión y no tiene una cuenta registrada, deberá dar click en "Sign up here". Deberá llenar sus datos completos y ya podrá iniciar sesión en su cuenta registrada.

Pantalla de Inicio de sesión

Smart Cargo Monitoring

Monitor your assets in real-time with total control over temperature, humidity, vibration, and location. Protect your sensitive products and guarantee deliveries in perfect condition.

Welcome back!

Enter your credentials to access your account

Email address

Password

Remember me

Forgot your password?

Sign in

Don't have an account? [Sign up here](#)

Pantalla de Registro

+ Back to log in

Create New Account

Complete the data to register on CargaSafe

Choose your segment

Client

Shipping Company

Profile data

First Name

Last Name

Account data

Email address

Password

Confirm password

By accepting, you agree to the application's [terms and conditions](#).

Sign Up

Alertas por incumplimiento de la temperatura:

- User Goal: Como cliente final, quiero recibir alertas cuando la temperatura sobrepasa los límites definidos, para tomar acciones correctivas.
- Descripción: El usuario se encuentra en la pantalla de inicio de sesión y no tiene una cuenta registrada, deberá dar click en "Sign up here". Deberá llenar sus datos completos y ya podrá iniciar sesión en su cuenta registrada.

Frame 78

Alerts

ID	Type	Delivery Order ID	Status	Created	Closed	Actions	Details
#0001	High Temperature	T342	Active	025 10:45	—	Mark as received	
#0002	Excessive vibration	T343	Pending	025 11:20	11:45 AM	Resolved✓	
#0003	Energy Loss	T344	Pending	025 01:05	01:14 PM	Resolved✓	
#0004	Trip Loss	T345	Pending	025 01:35	01:50 PM	Resolved✓	

Logout

Frame 79

Alerts

ID	Type	Delivery Order ID	Status	Created	Closed	Actions	Details
#0001	High Temperature	T342	Active	025 10:45	—	Mark as received	
#0002	Excessive vibration	T343	Pending	025 11:20	11:45 AM	Resolved✓	
#0003	Energy Loss	T344	Pending	025 01:05	01:14 PM	Resolved✓	
#0004	Trip Loss	T345	Pending	025 01:35	01:50 PM	Resolved✓	

Logout

Detalles de un viaje:

- User Goal: Como cliente final, quiero consultar el detalle de un viaje, para verificar información específica como ruta, estado y temperatura.
- Descripción: El usuario se encuentra en la pantalla de inicio de sesión y no tiene una cuenta registrada, deberá dar click en "Sign up here". Deberá llenar sus datos completos y ya podrá iniciar sesión en su cuenta registrada.

Listado de viajes

Trips

ID	Status	Driver	Co-driver	Nº Deliveries	Fecha de salida	Fecha de finalización	Acciones
1	Completo	José Sánchez	José José	5	24/05/2025 - 09:00AM	24/05/2025 - 03:00PM	<button>Ver detalle</button>
2	Cancelado	Mauricio Oliveira	No asignado	2	24/05/2025 - 09:00AM	-	<button>Ver detalle</button>
3	En curso	Jorge Ponce	Tyrene Solis	3	24/05/2025 - 09:00AM	-	<button>Ver detalle</button> <button>Actualizar</button>
4	Programado	Jorge Ponce	Tyrene Solis	2	24/05/2025 - 09:00AM	-	<button>Ver detalle</button> <button>Actualizar</button>

Páginas por página: 5 ▾ 1-8 of 18 < >

Detalle de viaje

Trip #00000001 Iniciar

Created at 10/06/2025, 8:30AM Departure at 10/06/2025, 10:30AM

Assigned to
Driver: Juan Pablo Co-driver: Not assigned

Shipping estimation
Total Distance: 2.2 km Total Duration: 00:24

Actual delivery
Total Distance: 3.0 km At 8.5% Total Duration: 04:32 At 0.5%



Delivery orders

ID	Sequence	Client	Address	Arrival at	Real arrival at	Status	Total incidents
0	1	Nicole Arevalo	Av. Javier Pr... Sector 1000	08:30AM	08:30AM	Completed	2
9	2	mgm@mail.com	Av. Javier Pr... Sector 1000	08:30AM	-	In course	1
10	3	Nicole Arevalo	Av. Javier Pr... Sector 1000	08:30AM	-	Cancelled	0
11	4	Nicole Arevalo	Av. Javier Pr... Sector 1000	08:30AM	-	In course	0
12	5	Nicole Arevalo	Av. Javier Pr... Sector 1000	08:30AM	-	In course	0

Páginas por página: 5 ▾ 1-8 of 18 < >

Descarga de los reportes de viajes:

- User Goal: Como cliente final, quiero descargar un reporte en PDF de un viaje con su información y gráficos, para archivarlo o compartirlo.
- Descripción: El usuario se encuentra en la pantalla de inicio de sesión y no tiene una cuenta registrada, deberá dar click en "Sign up here". Deberá llenar sus datos completos y ya podrá iniciar sesión en su cuenta registrada.

Listado de viajes

Trips

Total trips	Today	48	Yesterday	32	Last 7 Days	320	Last year	800
Total incidents	Today	2	Yesterday	1	Last 7 Days	5	Last year	98
1	Completado	Jose Sanchez	Jose Jose	5	24/08/2025 - 09:00AM	24/08/2025 - 03:00PM	<button>See detail</button>	
2	Cancelado	Mauricio Oliveira	No asignado	2	24/08/2025 - 09:00AM	--	<button>See detail</button>	
3	En curso	Jorge Ponce	Tyrone Sotil	3	24/08/2025 - 09:00AM	--	<button>See detail</button>	<button>Update</button>
4	Programado	Jorge Ponce	Tyrone Sotil	2	24/08/2025 - 09:00AM	--	<button>See detail</button>	<button>Update</button>

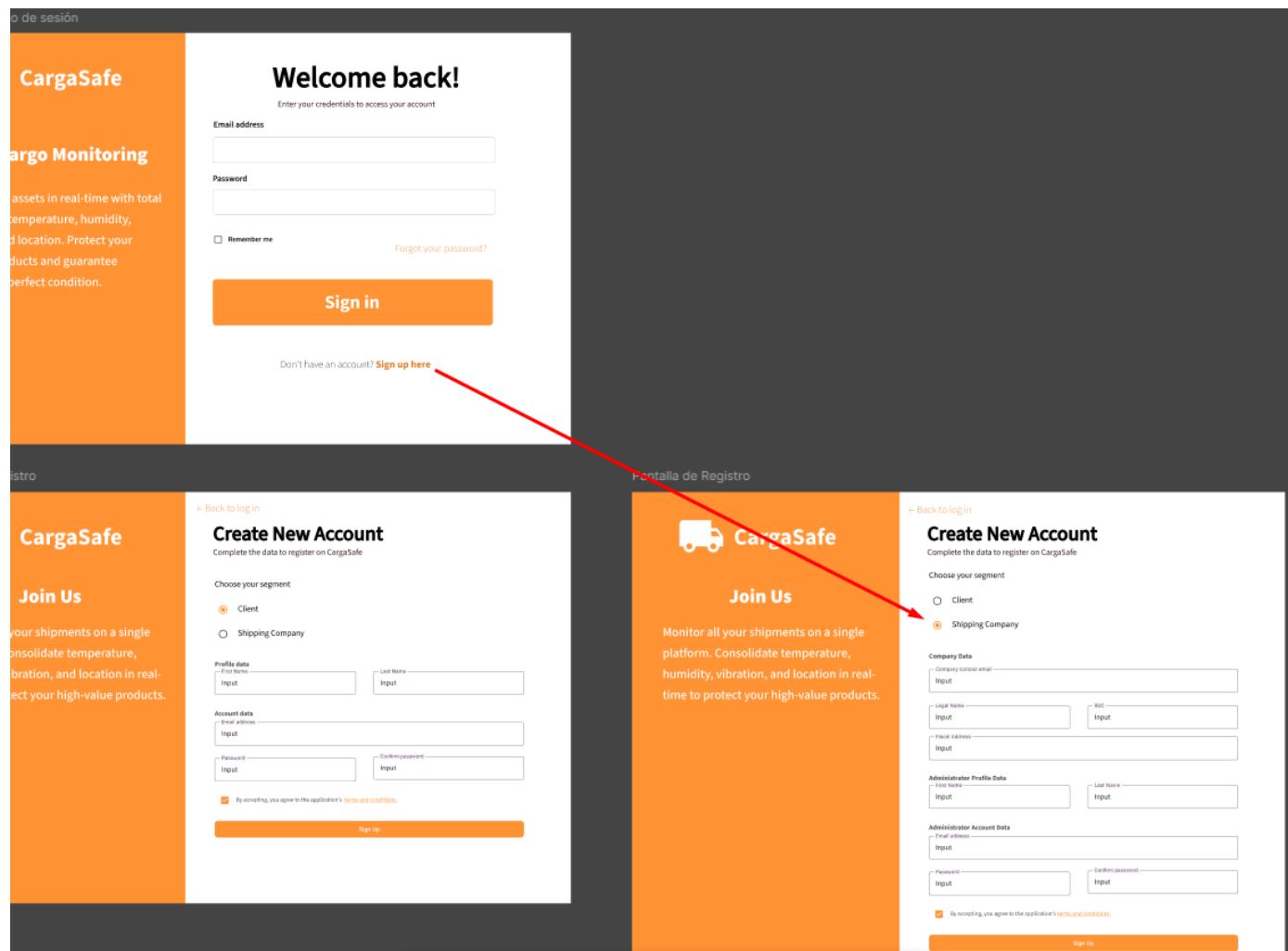
Rows per page: 5 | 1-5 of 10

Cerrar sesión

Web Application - Empresa

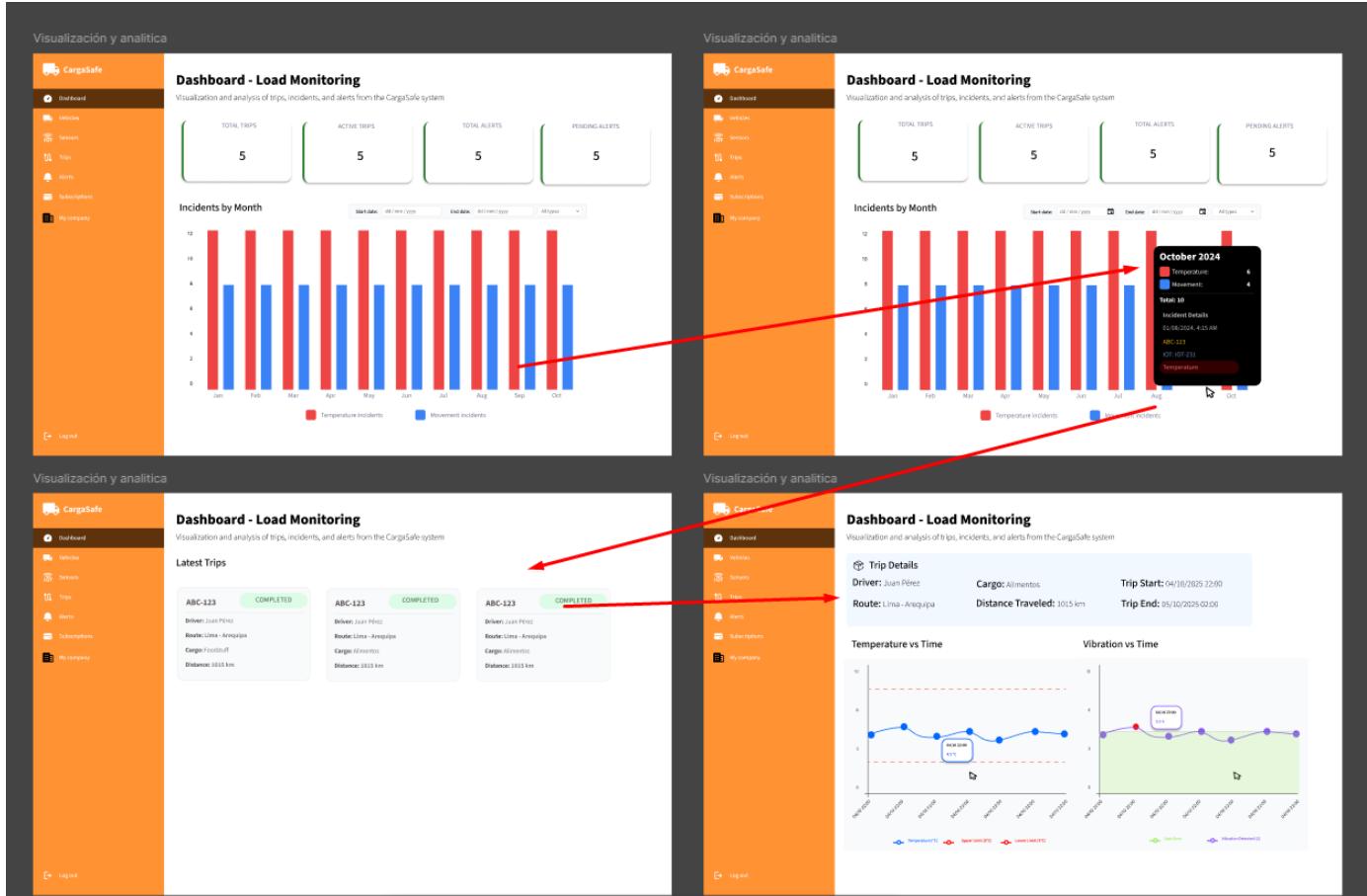
Registro de Usuario e Inicio de Sesión:

- User Goal: Como usuario, quiero registrarme en la plataforma, para acceder a mi cuenta y funcionalidades personalizadas.
- Descripción: El usuario se encuentra en la pantalla de inicio de sesión y no tiene una cuenta registrada, deberá dar click en "Sign up here". Deberá llenar sus datos completos. Si es una empresa, se deberá seleccionar la opción "Shipping Company" y llenar todos los datos de la empresa.



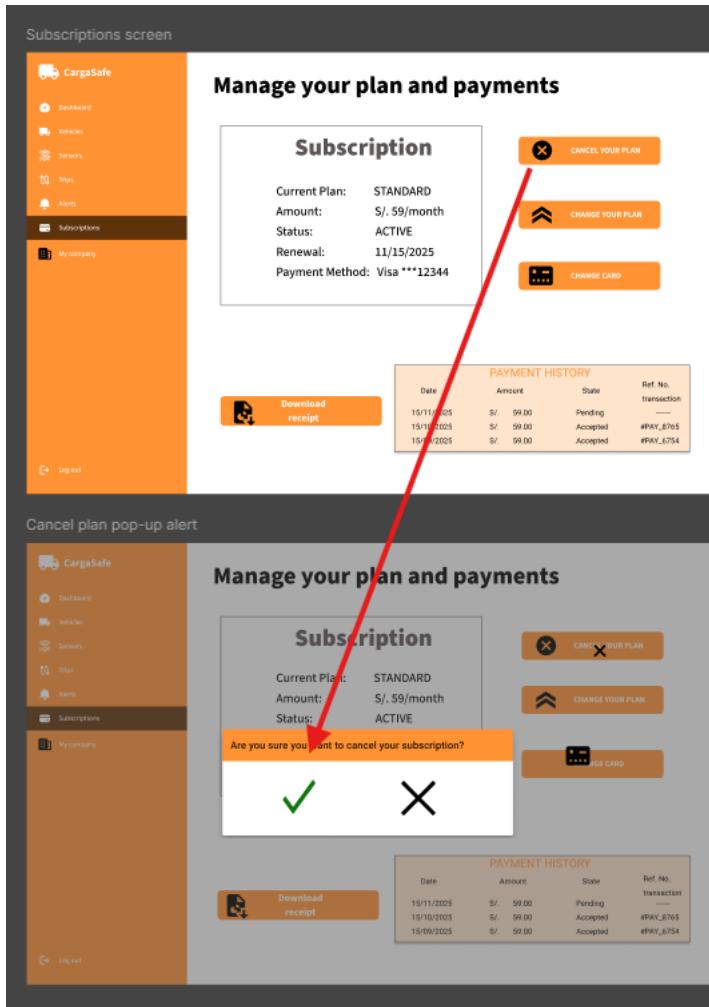
Monitoreo de temperatura en tiempo real:

- User Goal: Como empresa, quiero recibir la temperatura en tiempo real de mis dispositivos IoT, para supervisar la cadena de frío de los viajes.
- Descripción: El usuario se encuentra en la pantalla de inicio de sesión y no tiene una cuenta registrada, deberá dar click en "Sign up here". Deberá llenar sus datos completos. Si es una empresa, se deberá seleccionar la opción "Shipping Company" y llenar todos los datos de la empresa.



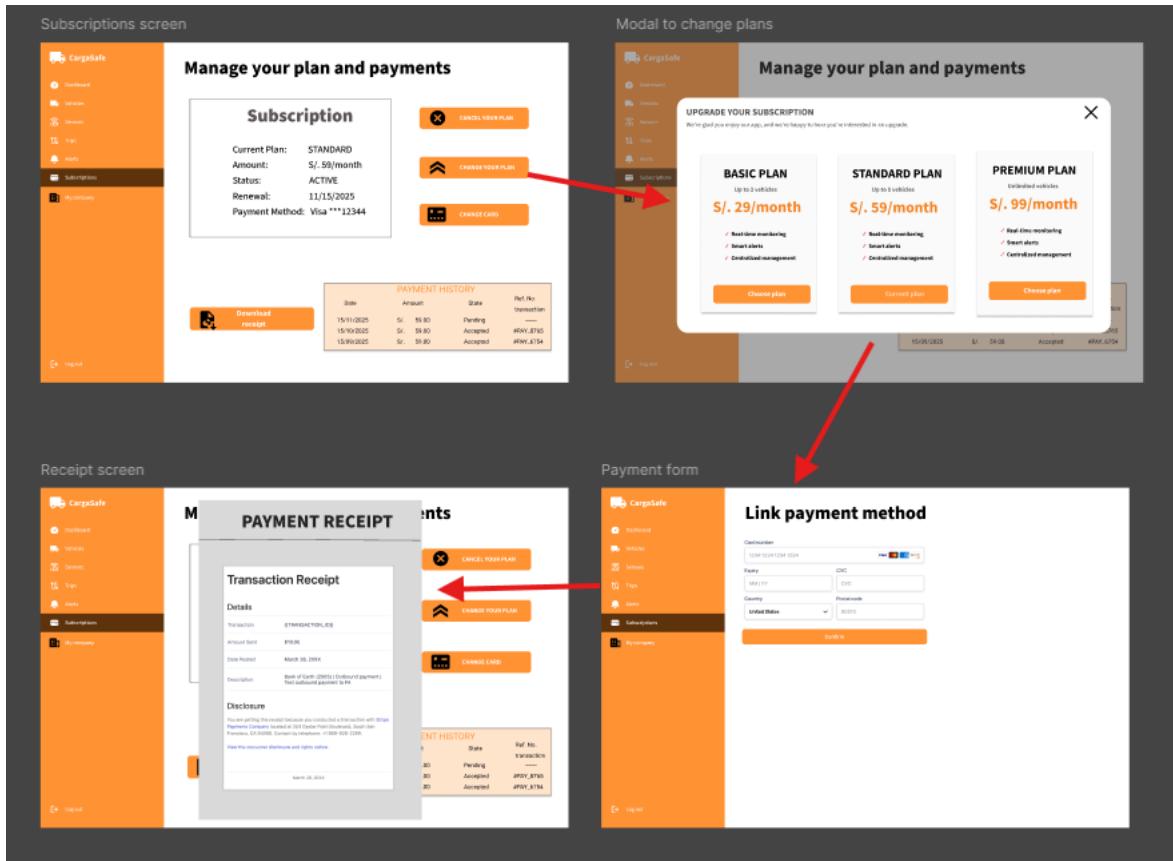
Anulación de suscripción:

- User Goal: Como empresa o cliente final, quiero anular mi suscripción actual, para dejar de utilizar el servicio y cancelar los pagos recurrentes asociados a mi cuenta.
- Descripción: El usuario que desea cancelar su suscripción debe iniciar sesión en su cuenta y navegar a la sección de "Subscriptions" desde el menú principal. Una vez en la pantalla de suscripciones, podrá visualizar su plan activo actual con todos los detalles (tipo de plan, fecha de renovación, métodos de pago). El usuario deberá seleccionar la opción "Cancel your Plan", lo que desplegará una confirmación explicando las consecuencias de la cancelación (pérdida de acceso a funcionalidades premium, fecha efectiva de terminación del servicio). Tras confirmar la acción, el sistema procesará la solicitud, enviará un correo de confirmación y actualizará el estado de la suscripción a "Cancelada" o "Pendiente de cancelación", permitiendo al usuario seguir usando el servicio hasta el final del período ya pagado.



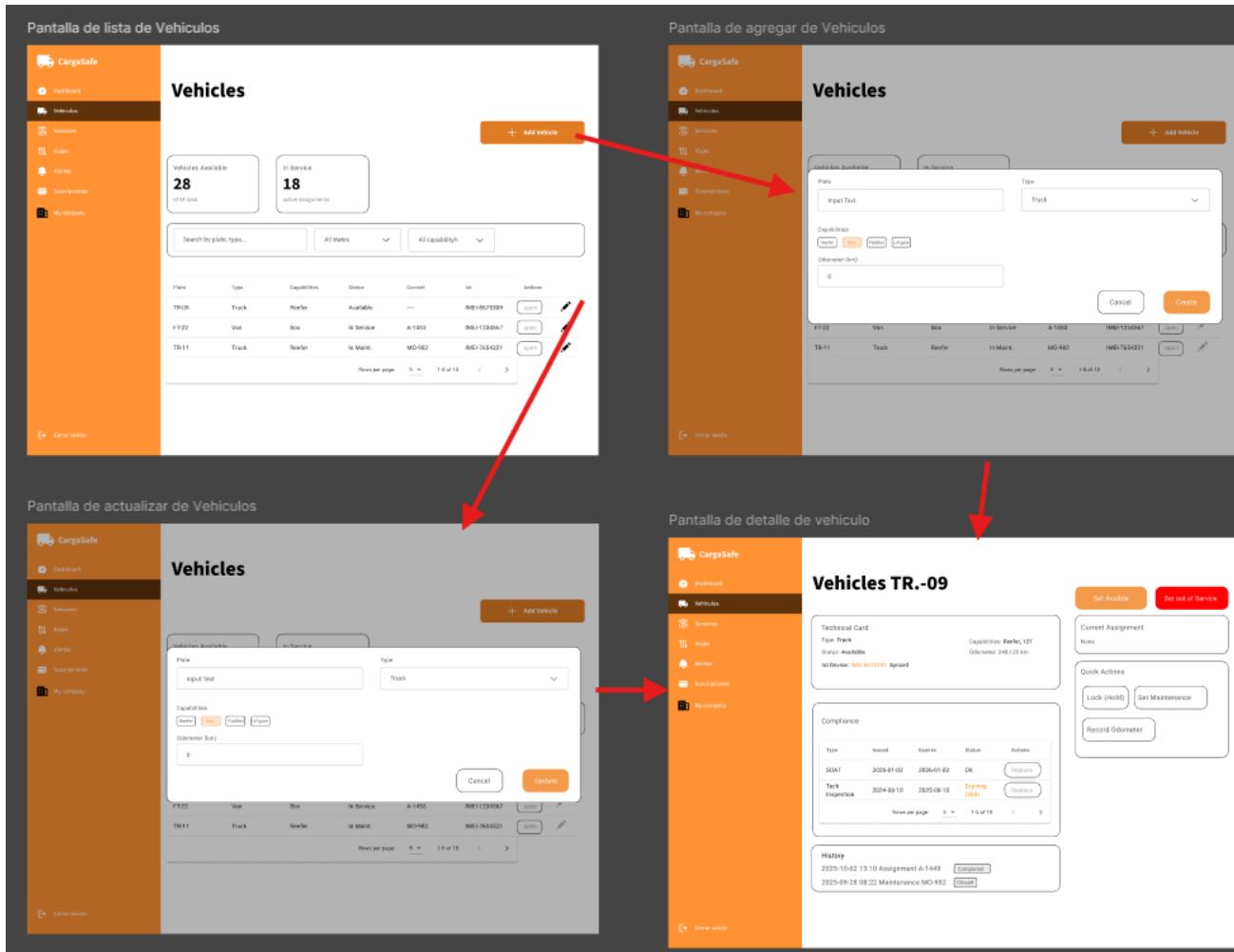
Actualización de suscripción (Upgrade):

- User Goal: Como empresa o cliente final, quiero actualizar mi plan de suscripción a uno superior, para acceder a más funcionalidades y beneficios que se ajusten mejor a mis necesidades operativas.
- Descripción: El usuario que desea mejorar su plan de suscripción debe acceder al módulo de "Subscriptions" desde el menú principal de la aplicación. Una vez en la pantalla de gestión de suscripciones, el usuario visualizará su plan actual con sus características y limitaciones. Al seleccionar la opción "Change your Plan", el sistema desplegará todas las opciones de suscripción disponibles (Basic, Professional, Enterprise) con sus respectivas características, precios y beneficios comparativos. El usuario podrá revisar las diferencias entre planes y seleccionar el que mejor se adapte a sus necesidades. Posteriormente, el sistema solicitará verificar o actualizar el método de pago registrado (tarjeta de crédito/débito, información de facturación). Una vez confirmados los datos de pago, el sistema procesará la transacción de upgrade y mostrará un recibo digital detallado con la información de la nueva suscripción, fecha de activación, monto cobrado y próxima fecha de renovación. Finalmente, el usuario recibirá una confirmación por correo electrónico con el recibo adjunto y el plan quedará actualizado inmediatamente, otorgando acceso a las nuevas funcionalidades premium.



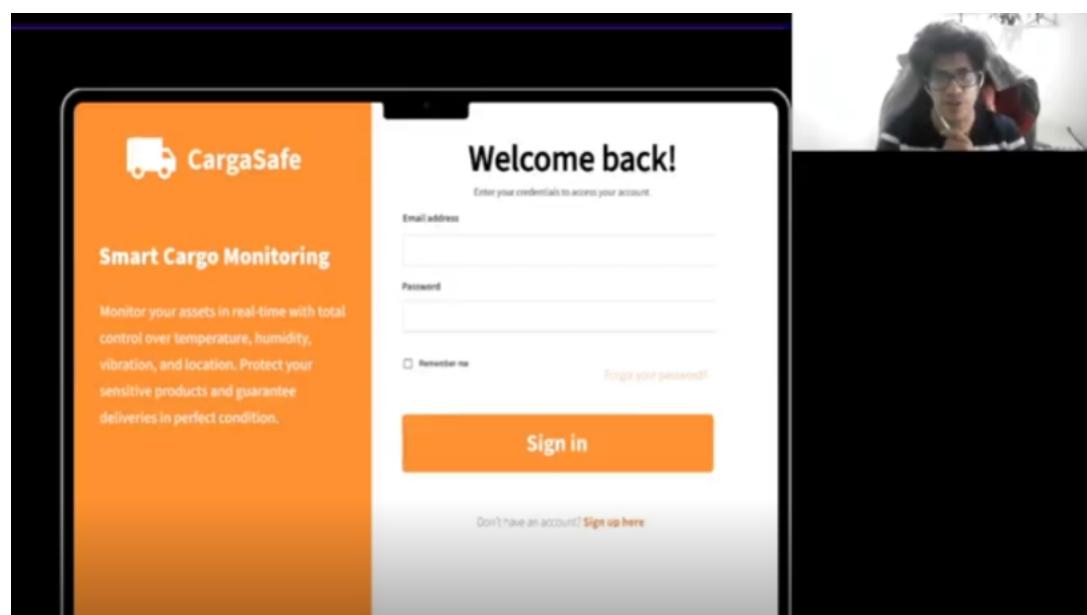
Registro y modificación de información de vehículos:

- User Goal: Como empresa de transporte, quiero registrar nuevos vehículos en el sistema o modificar la información existente de mis vehículos, para mantener actualizado el inventario de mi flota y asegurar el correcto monitoreo de cada unidad.
- Descripción: El usuario debe acceder al módulo de "Vehículos" desde el menú principal de la aplicación. Una vez en la pantalla de gestión de flota, visualizará un listado completo de todos los vehículos registrados con información básica como placa, modelo, estado y dispositivo IoT asignado. Para agregar un nuevo vehículo, el usuario debe seleccionar el botón "Add Vehicle", lo que desplegará un formulario con los parámetros necesarios para el registro: placa del vehículo, marca, modelo, año, tipo de vehículo (camión refrigerado, van, camión seco), capacidad de carga, y datos del conductor asignado. Una vez completados todos los campos requeridos, el usuario confirma el registro y el sistema valida la información, guardando el nuevo vehículo en la base de datos. Si el usuario necesita editar la información de un vehículo existente, debe seleccionar el ícono del lápiz (edit) junto al registro del vehículo en la lista, lo que abrirá el mismo formulario con los datos actuales precargados, permitiendo modificar cualquier parámetro necesario. Tras guardar los cambios, el sistema actualiza la información y muestra una confirmación. Finalmente, el usuario puede visualizar la información completa del vehículo recién creado o modificado en la vista de detalle, incluyendo su historial de viajes, dispositivos asociados y estado operativo actual.



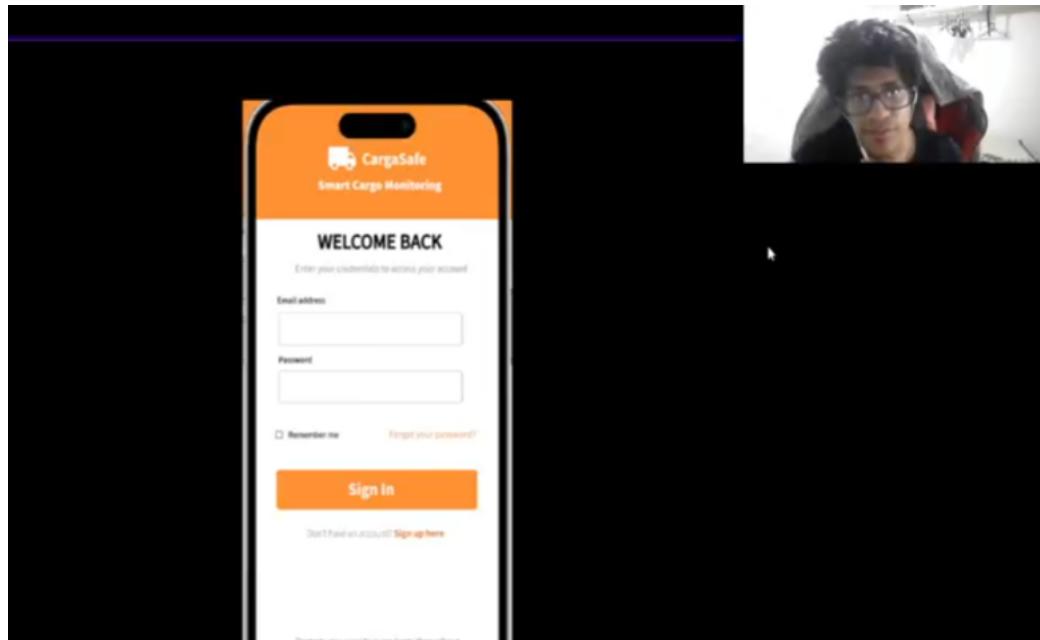
5.5. Applications Prototyping.

El sistema prioriza la visualización inmediata de datos críticos mediante cards estadísticas y gráficos interactivos (dashboard con métricas en tiempo real, charts con tooltips hover, y tablas filtradas). La navegación está centralizada en un sidebar persistente con iconografía Material Design que facilita el cambio rápido entre contextos. Las acciones de usuario siguen un patrón de feedback visual claro: estados de loading explícitos, filtros reactivos (búsqueda, fecha, tipo) que actualizan tablas en vivo, y cards clickeables que navegan a detalles sin recarga de página. La arquitectura de interacción refleja el dominio operacional: el dashboard como hub central de monitoreo, detalles de viajes con gráficos de sensores (temperatura/movimiento) para análisis profundo, y gestión de alertas con estados visuales (active/resolved) para triage rápido. Los formularios son minimalistas usando Material UI (outline appearance, dropdowns, checkboxes) optimizados para captura eficiente de datos IoT como IMEI, firmware y asociación de dispositivos.



Enlace al video de presentación del prototipo: [Ver presentación del prototipo web](#)

Enlace al prototipo interactivo en Figma (Web): [Ver prototipo web en Figma](#)



Enlace al video de presentación del prototipo móvil: Ver presentación del prototipo móvil

Enlace al prototipo interactivo en Figma (Mobile): Ver prototipo móvil en Figma

Capítulo VI: Product Implementation, Validation & Deployment

6.1. Software Configuration Management.

El objetivo de esta sección es asegurar que todos los miembros usen las mismas herramientas, convenciones y procesos para:

- Desarrollar código
- Hacer pruebas
- Desplegar versiones
- Documentar el software

6.1.1. Software Development Environment Configuration.

Categoría	Herramienta / Producto	Propósito en el proyecto	Tipo	Ruta / Enlace de referencia
Project Management	Jira	Facilita la planificación, organización y seguimiento de tareas del equipo. Permite asignar responsables, estimar tiempos y monitorear el progreso de manera colaborativa.	SaaS	https://www.atlassian.com/software/jira
Project Management	Discord	Herramienta de comunicación interna del equipo. Permite reuniones, compartir pantalla y mantener comunicación fluida mediante mensajes, imágenes y video.	SaaS / Escritorio	https://discord.com/
Project Management	WhatsApp	Herramienta de comunicación para mantener conectividad continua entre los integrantes del equipo.	SaaS	https://web.whatsapp.com
Requirements Management	UXPressia	Desarrolla User Personas, Customer Journey Maps, Empathy Maps e Impact Maps. Facilita la colaboración en tiempo real y la documentación visual de requisitos de usuario.	SaaS	https://uxpressia.com/
Requirements Management	Miro	Herramienta para crear mapas de escenarios y procesos, permitiendo visualizar la experiencia del usuario y definir los segmentos de usuarios.	SaaS	https://miro.com/
UX/UI Design	Figma	Diseño y prototipado de interfaces interactivas. Permite trabajo colaborativo simultáneo y garantiza la coherencia visual en el desarrollo de la UI.	SaaS	https://www.figma.com/
Source Code Management	Git & GitHub	Git gestiona versiones de código y ramas de desarrollo. GitHub aloja los repositorios y facilita la colaboración remota, control de versiones y revisión de código.	SaaS / Local	https://github.com/
Backend Development	Spring Boot (Java)	Framework para el desarrollo del backend. Facilita la creación de servicios REST escalables y estructurados, con integración a bases de datos y herramientas externas.	Local	https://spring.io/projects/spring-boot
Frontend Development	Visual Studio Code	IDE utilizado para el desarrollo del frontend. Permite integración con frameworks modernos (Angular, React) y control de versiones con Git.	Local	https://code.visualstudio.com/
Backend Development	IntelliJ IDEA	IDE usado para el desarrollo backend con Spring Boot. Proporciona herramientas integradas de depuración, testing y control de dependencias.	Local	https://www.jetbrains.com/idea/
Mobile Development	Android Studio	Entorno de desarrollo para la aplicación móvil Android. Facilita el diseño, compilación y depuración de aplicaciones nativas.	Local	https://developer.android.com/studio
Software Testing	Postman	Plataforma para pruebas de APIs REST. Permite validar endpoints, realizar pruebas automatizadas e integrar con pipelines CI/CD.	SaaS / Local	https://www.postman.com/
Software Deployment	Firebase Hosting	Servicio para desplegar aplicaciones web de forma rápida y segura. Ofrece CDN global, HTTPS automático e integración con Firebase Functions.	SaaS	https://firebase.google.com/docs/hosting
Software Deployment	Microsoft Azure	Plataforma en la nube para el despliegue y administración de servicios, bases de datos y APIs. Permite escalabilidad, monitoreo y seguridad.	SaaS	https://azure.microsoft.com/
Software Documentation	Markdown	Lenguaje de marcado ligero usado para la documentación técnica del proyecto (README, manuales, guías de instalación).	Local	https://www.markdownguide.org/

6.1.2. Source Code Management.

El equipo utiliza Git como sistema de control de versiones distribuido y GitHub como plataforma de alojamiento remoto de repositorios. Esta configuración permite la colaboración en equipo, el seguimiento de cambios y la integración con flujos de CI/CD.

Repositorio de los productos:

- Repositorio del Landing Page: [Click aquí](#)
- Repositorio del Frontend Web Applications: [Click aquí](#)
- Repositorio del App Mobile: [Click aquí](#)
- Repositorio del Web Service: [Click aquí](#)

Se implementa el flujo de trabajo GitFlow. Este modelo organiza el desarrollo en ramas con funciones específicas para mantener la estabilidad de la rama principal `main` y facilitar el desarrollo paralelo de nuevas características.

- `main`: contiene el código estable y listo para producción.
- `develop`: integra las nuevas funcionalidades desarrolladas y sirve como base para las versiones futuras.
- `feature`: se utiliza para desarrollar nuevas funcionalidades. Cada funcionalidad tiene su propia rama basada en `develop` y se fusiona nuevamente en `develop` al completarse.
- `release`: se crea a partir de `develop` cuando se prepara una versión para producción. Permite realizar pruebas y ajustes menores antes del lanzamiento.
- `hotfix`: se crea desde `main` para corregir errores críticos detectados en producción. Una vez solucionado, se fusiona en `main` y `develop`.

Convenciones de nombres de ramas:

- Feature branches: `feature/nombre` -> Ejemplo: `feature/trip-management`
- Release branches: `release/vX.Y.Z` -> Ejemplo: `release/v0.0.1`
- Hotfix branches: `hotfix vX.Y.Z` -> Ejemplo: `hotfix/v1.0.1`

Se adopta el esquema **Semantic Versioning 2.0.0** para identificar las versiones de los productos digitales. Formato: `MAJOR.MINOR.PATCH`

- MAJOR: cambios incompatibles.
- MINOR: nuevas funcionalidades compatibles.
- PATCH: corrección de errores o mejoras menores.

Ejemplo: `v1.0.0.v1.1.0.v1.1.1`

Se utiliza el estándar Conventional Commits para mantener una trazabilidad clara del historial de cambios. Formato: `<tipo>(<alcance>): <descripción>` Ejemplos:

- `feat(alert-and-resolutions)`: add alert and resolutions page
- `fix(trip-managment)`: update trip info
- `docs(chap-6)`: add software configuration management

Tipos principales:

- `feat`: nueva funcionalidad.
- `fix`: corrección de error.
- `docs`: documentación.
- `style`: formato o estilos (sin cambio funcional).
- `refactor`: reestructuración del código.
- `test`: adición o mejora de pruebas.
- `chore`: tareas de mantenimiento o configuración.

Toda nueva funcionalidad o corrección debe ser revisada mediante Pull Request (PR). Al aprobarse, la rama se fusiona (`merge`) en `develop` o `main` según el caso.

A continuación se detallarán las ramas creadas y las convenciones que se aplicarán:

Tipo de Rama	Nombre	Propósito	Rama en la que se crea	Rama en la que se fusiona
Main branch	<code>main</code>	Código estable listo para producción	—	—
Development branch	<code>develop</code>	Integrar todas las nuevas funcionalidades antes de lanzar	Creada desde <code>main</code>	Recibe merges de <code>feature/*</code>
Feature branches	<code>feature/IAM, feature/alerts-and-resolutions, feature/fleet-management, feature/subscriptions, feature/visualization-and-analytics, feature/trip-management</code>	Contienen el desarrollo de nuevas funcionalidades específicas	Desde <code>develop</code>	Se fusionan en <code>develop</code>
Release branches	<code>release/v1.0.0, release/v1.1.0</code>	Preparar una versión candidata con ajustes menores y pruebas finales	Desde <code>develop</code>	Se fusionan en <code>main</code> y <code>develop</code>
Hotfix branches	<code>hotfix/v1.0.1, hotfix/v1.1.1</code>	Corregir errores críticos que aparecen en producción	Desde <code>main</code>	Se fusionan en <code>main</code> y <code>develop</code>

6.1.3. Source Code Style Guide & Conventions.

Se detallarán las convenciones de codificación y nomenclatura que el equipo adoptará para los lenguajes y frameworks utilizados en la solución CargaSafe. Todas las convenciones se aplicarán en inglés con el objetivo de mantener la consistencia, claridad y estandarización del código entre todos los miembros del equipo.

<https://google.github.io/styleguide/htmlcssguide.html>

Convenciones:

- Utilizar minúsculas para todos los nombres de elementos, atributos y valores.
- Anidar correctamente las etiquetas y mantener una estructura jerárquica clara.
- Indentación de espacios por nivel.
- Los atributos deben llevar comillas dobles ("")
- Usar nombres descriptivos y semánticos en las clases y los identificadores.
- Evitar el uso de estilos en línea; preferir archivos CSS o clases.

Ejemplo:

```
<section class="device-list">
  <h2 class="section-title">Example</h2>
</section>
```

CSS

<https://google.github.io/styleguide/htmlcssguide.html>

Convenciones:

- Utilizar Kebab-case para los nombres de clases e identificadores (.button-primary, .nav-bar)
- Agrupar propiedades relacionadas y ordenar alfabéticamente.
- Evitar el uso excesivo de selectores anidados.
- Indentación de 2 espacios.
- Cada bloque debe terminar con un salto de línea.

```
.example-list {
  background-color: #ffffff;
  border-radius: 12px;
  padding: 1rem;
}
```

JavaScript

<https://google.github.io/styleguide/jsguide.html>

Convenciones:

- Usar camelCase para variables y funciones.
- Usar PascalCase para clases.
- Constantes globales en UPPER_CASE.
- Preferir const y let sobre var
- Evitar funciones anidadas innecesarias y callbacks excesivos.
- Usar arrow functions y template literals.
- Cada archivo debe contener una sola clase o módulo.

```
const API_URL = 'https://api.cargasafe.com'

class Example {
  constructor(exampleId) {
    this.exampleId = exampleId
  }

  getStatus() {
    return `${this.exampleId} is active`
  }
}
```

Java

<https://google.github.io/styleguide/javaguide.html>

Convenciones:

- Usar PascalCase para clases e interfaces.
- Usar camelCase para métodos y variables.
- Los paquetes se nombran en minúsculas, separados por puntos (e.g., com.cargasafe.backend)
- Indentación de 4 espacios.
- Líneas de máximo 120 caracteres.

- Los nombres deben ser descriptivos, evitando abreviaciones innecesarias.
- Cada clase pública debe estar en su propio archivo con el mismo nombre.

```
package com.cargasafe.backend.services;

public class ExampleService {
    private int alertCount;

    public void sendAlert(String message) {
        System.out.println("Alert: " + message);
    }
}
```

Dart

<https://dart.dev/effective-dart/style>

Convenciones:

- Usar camelCase para variables y funciones.
- Usar PascalCase para clases, enumeraciones y typedefs.
- Constantes en UPPER_CASE.
- Cada archivo debe contener una sola clase o widget principal.
- Preferir interpolación de strings (`Hello $name`) sobre concatenación.
- Usar `final` y `const` siempre que sea posible para garantizar inmutabilidad.

```
class ExampleCard extends StatelessWidget {
    final String deviceName;

    const ExampleCard({required this.deviceName, super.key});

    @override
    Widget build(BuildContext context) {
        return Text('Example: $deviceName');
    }
}
```

Gherkin

<https://cucumber.io/docs/gherkin/reference>

Convenciones:

- Usar mayúscula inicial en cada palabra clave (`Feature`, `Sceneario`, `Given`, `When`, `Then`).
- Describir escenarios de forma clara, usando lenguaje natural.
- Una sola característica (`Feature`) por archivo.
- Mantener consistencia en la redacción de pasos.
- Incluir etiquetas (`@tag`) para clasificar los escenarios.

```
Feature: User Login

Scenario: Successful login with valid credentials
  Given the user is on the login page
  When the user enters valid credentials
  Then the dashboard is displayed
```

6.1.4. Software Deployment Configuration.

El proceso de despliegue de la solución CargaSafe se basa en la integración continua (CI) y la entrega continua (CD) a través de repositorios en GitHub. Cada producto digital cuenta con su propia configuración de despliegue para garantizar la disponibilidad, seguridad y escalabilidad del sistema.

Para la Landing page se utilizó GitHub Pages, los pasos a seguir son:

1. Entrar al repositorio creado

The screenshot shows a GitHub repository named "iot-solutions-development-cargasafe-landing". The repository has 12 branches and 1 tag. The main branch is selected. The landing page files (index.html, script.js, styles.css, etc.) are listed with their commit history, all made by AlessandroBernardoEusebio. The repository description is "CargaSafe Landing Page, a static website that introduces and promotes the CargaSafe IoT logistics platform." The repository has 0 stars, 0 forks, and 0 watching.

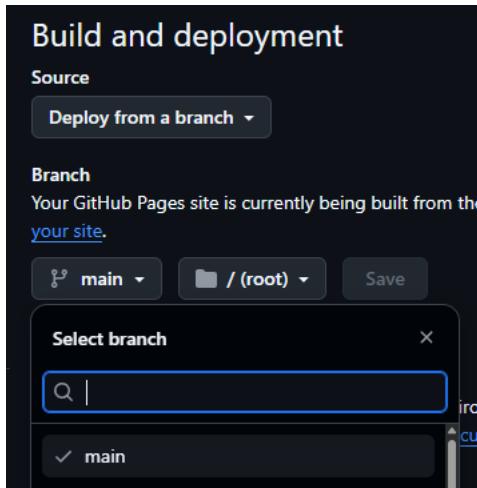
2. **Ingresar a la pestaña **Pages**

The screenshot shows the GitHub Pages settings page for the repository. Under the "General" tab, it shows the site is live at <https://los-parkers-iot.github.io/iot-solutions-development-cargasafe-landing/>. It also shows the "Deploy from a branch" dropdown set to "main". Other sections include "Build and deployment" (with "Source" set to "Deploy from a branch"), "Custom domain", and "Integrations".

3. Github te da dos opciones para desplegar el software, GitHub Actions y Deploy from a branch.

The screenshot shows the "Build and deployment" section under "Source". It highlights the "GitHub Actions" option, which is described as "Best for using frameworks and customizing your build process". Below it, there is a button for "Deploy from a branch" and a note about "Classic Pages experience". A link to "Learn how to add a Jekyll theme to your site." is also present.

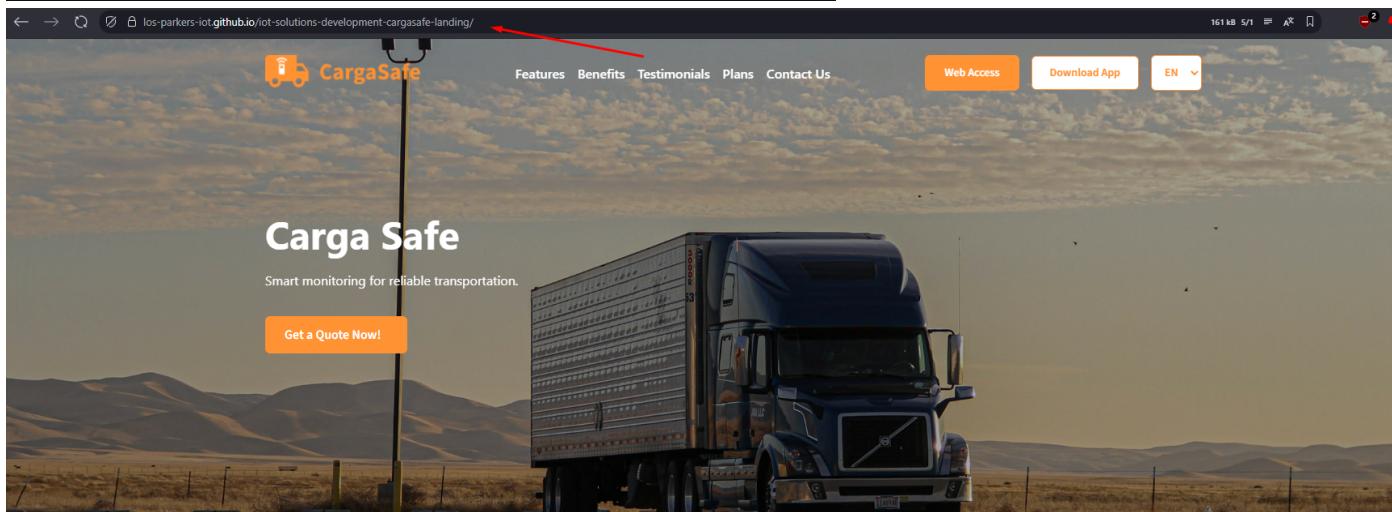
4. La mejor opción para el despliegue de la landing page es Deploy from a branch, una vez elegido esta opción, se tiene que elegir la rama que se desee, por regla general, la rama main es la que se debe elegir.



5. Una vez elegida la rama donde se va a desplegar la landing page, Github generará un link donde se ha desplegado el producto.

Your site is live at <https://los-parkers-iot.github.io/iot-solutions-development-cargasafe-landing/>
Last deployed by AlessandroBernardoEusebio 49 minutes ago

[Visit site](#) [Unpublish site](#)



Para la Web Application se utilizó GitHub Actions y Firebase.

1. Se debe generar una carpeta `github/workflows` para que la rama `main` pueda actualizarse constantemente

main 14 Branches 3 Tags Go to file Add file Code

SgtGarro feature(subscriptions): refactor CSS and HTML structure for responsiv... e30c6b6 · 12 hours ago 65 Commits

.github/workflows fix(c): update deployment branch to main in Firebase hostin... yesterday

2. Se tiene que crear un archivo `yml` y colocar el siguiente código para firebase hosting

iot-solutions-development-cargasafe-frontend/.github/workflows/firebase-hosting-merge.yml

SgtGarro fix(ci): update deployment branch to main in Firebase hosting workflow ✓

Code **Blame** 22 lines (21 loc) · 651 Bytes

```
1 # This file was auto-generated by the Firebase CLI
2 # https://github.com/firebase/firebase-tools
3
4 name: Deploy to Firebase Hosting on merge
5 on:
6   push:
7     branches:
8       - main # Set a branch to deploy from
9 jobs:
10   build_and_deploy:
11     runs-on: ubuntu-latest
12     steps:
13       - uses: actions/checkout@v4
14       - run: npm ci
15       - uses: FirebaseExtended/action-hosting-deploy@v0
16         with:
17           repoToken: ${{ secrets.GITHUB_TOKEN }}
18           firebaseServiceAccount: ${{ secrets.FIREBASE_SERVICE_ACCOUNT_CARGA_SAFE }}
19           channelId: live
20           projectId: carga-safe
21         env:
22             FIREBASE_CLI_EXPERIMENTS: webframeworks
```

3. Luego de generar el archivo, se debe conectar al Firebase Hosting, pero primero se debe crear una cuenta para poder utilizarlo.

 **Firebase** Compilación ▾ Ejecución ▾ Soluciones Precios Documentos ▾ Comunidad ▾ Más ▾  Buscar   Language ▾ Blog  Studio Ir a la consola Acceder

4. **Una vez creada la cuenta se debe ejecutar `npm install -g firebase-tools` para instalar los componentes de firebase, luego de instalarlo, se debe ejecutar el comando `**firebase login***`

```
1 The Firebase CLI's MCP server feature can optionally make use of Gemini in Firebase. Learn more about Gemini in Firebase and how it uses your data: https://firebase.google.com/docs/gemini-in-firebase
✓ Enable Gemini in Firebase features? No

i Firebase optionally collects CLI and Emulator Suite usage and error reporting information to help improve our products. Data is collected in accordance with Google's privacy policy (https://policies.google.com/terms.html)
✓ Allow Firebase to collect CLI and Emulator Suite usage and error reporting information? Yes

i To change your preferences at any time, run 'firebase logout' and 'firebase login' again.

Visit this URL on this device to log in:
https://accounts.google.com/o/oauth2/auth?client\_id=563584335869-fgrhgmd47bqneki5i8b5pr03h084e6.apps.googleusercontent.com&scope=email%20openid%20https%3A%2F%2Fwww.googleapis.com%2Fauth%2Fcloudplatform%2Fauth%2Fcloud-platform&response\_type=code&state=787955897&redirect\_uri=http%3A%2F%2Flocalhost%3A9005

Waiting for authentication...

+ Success! Logged in as daniel-741@hotmail.com
PS C:\Users\daniel\Downloads\IoT\Parkeerlijst-solutions-development-cargasafe-frontend>
```

5. Luego se debe ejecutar el comando `firebase init hosting` y se debe realizar los pasos a seguir que saldrán en pantalla.

```
#####
##   ##   ##   ##   ##   ##   ##   ##
#####   ##   #####   ##   #####   ##   #####
##   ##   ##   ##   ##   ##   ##   ##   ##
##   ##### ##   ##   ##### ##   ##   ##   ##

You're about to initialize a Firebase project in this directory:

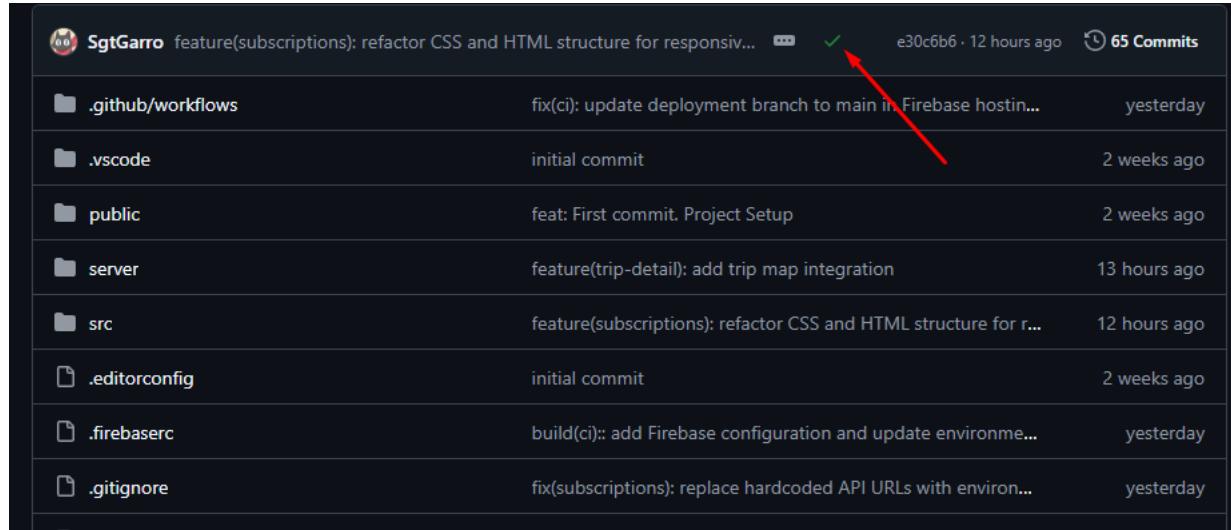
C:\Users\danie\Downloads\IoTParkers\iot-solutions-development-cargasafe-frontend

Before we get started, keep in mind:

* You are initializing within an existing Firebase project directory

? Are you ready to proceed? (Y/n) y
```

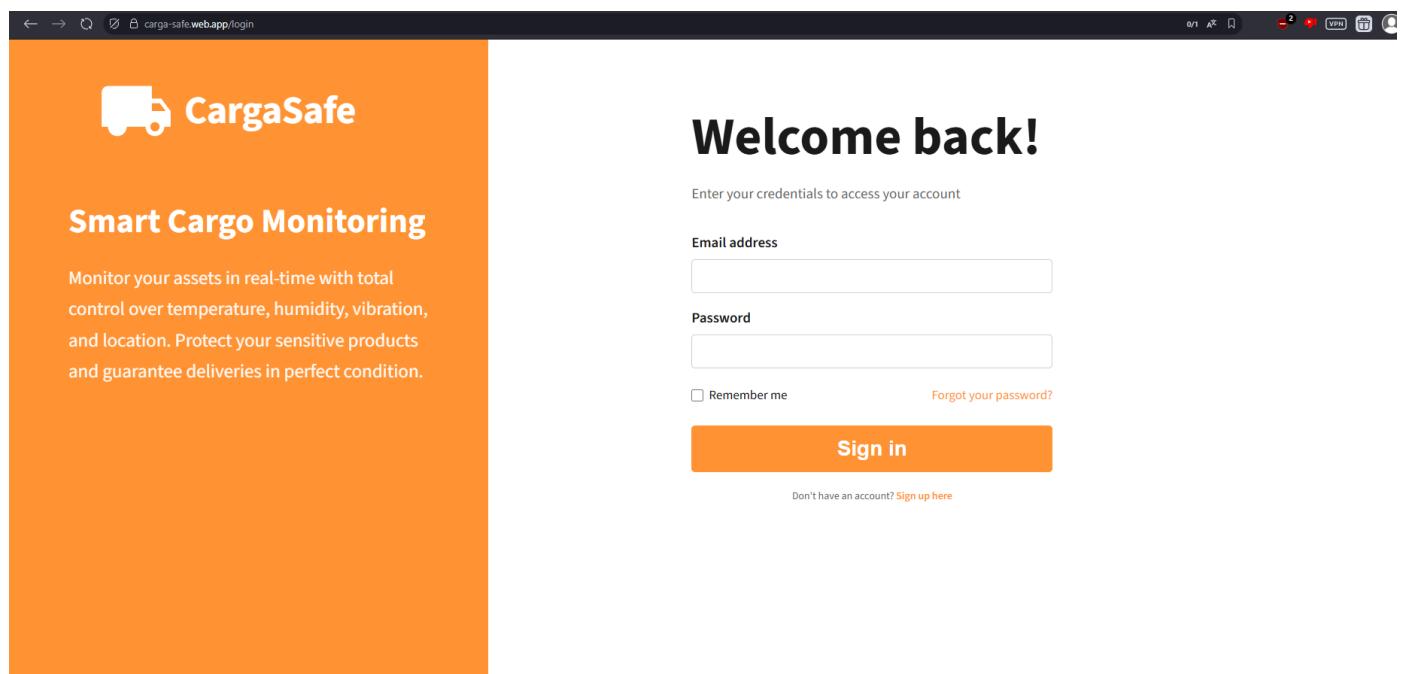
6. Una vez desplegado la web application, saldrá una confirmación en la rama seleccionada



A screenshot of a GitHub commit history. At the top, there's a header with the user 'SgtGarro', the commit message 'feature(subscriptions): refactor CSS and HTML structure for responsiv...', a green checkmark icon, the commit hash 'e30c6b6', the time '12 hours ago', and '65 Commits'. Below this, there are several commits listed in a table format:

Commit Details	Time Ago
.github/workflows fix(ci): update deployment branch to main in Firebase hostin...	yesterday
.vscode initial commit	2 weeks ago
public feat: First commit. Project Setup	2 weeks ago
server feature(trip-detail): add trip map integration	13 hours ago
src feature(subscriptions): refactor CSS and HTML structure for r...	12 hours ago
.editorconfig initial commit	2 weeks ago
.firebaserc build(ci):: add Firebase configuration and update environme...	yesterday
.gitignore fix(subscriptions): replace hardcoded API URLs with environ...	yesterday

7. Y la web application ya estará desplegada



6.2. Landing Page, Services & Applications Implementation.

6.2.1. Sprint 1

6.2.1.1. Sprint Planning 1.

En esta sección se especifican los aspectos principales del Sprint Planning Meeting para el Sprint 1 del proyecto CargaSafe. Este primer sprint se enfoca en establecer las bases de la plataforma, implementando el landing page y los componentes iniciales de la aplicación web que permitan a los usuarios conocer el producto y comenzar a interactuar con las funcionalidades core del sistema de monitoreo de carga.

Sprint #
Sprint 1
Sprint Planning Background Date 2025-28-09 Time 08:00 PM Location Reunión virtual mediante Discord Prepared By Marcelo Fabian Garro Vega Attendees (to planning meeting) Marcelo Fabian Garro Vega / Jefrey Martin Sanchez Ignacio / Daniel Rodrigo Chirinos Arevalo / Alessandro Joaquin Bernardo Eusebio / Vera Nuñez Nicolas Alejandro / George Aliaga Sprint 1 Review Summary Al ser el primer sprint del proyecto, no existe un sprint anterior para revisar. Sin embargo, se completaron las actividades de planeamiento inicial, incluyendo la definición de la arquitectura de software bajo principios de Domain Driven Design, la elaboración de wireframes y mockups, y la especificación de user stories en el product backlog. Sprint 1 Retrospective Summary Durante la fase de planeamiento (Sprint 0), el equipo logró establecer una visión compartida del producto y definir los bounded contexts principales del sistema. Se identificaron oportunidades de mejora en la comunicación asíncrona y se acordó implementar daily stand-ups virtuales para mantener la sincronización del equipo. El equipo manifestó motivación y compromiso para comenzar la implementación del producto. Sprint Goal & User Stories Sprint 1 Goal Nuestro enfoque está en establecer la presencia digital de CargaSafe mediante el landing page y desarrollar la capa frontend completa de la aplicación web para los ocho bounded contexts identificados (Identity and Access Management, Profiles and Preferences Management, Fleet Management, Execution of the Trip, Real-time Monitoring, Alerts and Resolution, Visualization/Analytics, y Subscriptions and Payments) utilizando fake APIs para simular la interacción con el backend.

Creemos que esto entrega una primera impresión profesional del producto a visitantes potenciales, permite validar la experiencia de usuario completa del sistema con stakeholders, y establece una base sólida de componentes frontend reutilizables para empresas de transporte piloto.

Esto se confirmará cuando el landing page esté publicado y accesible con información completa del producto, y cuando los usuarios puedan navegar por todas las funcionalidades de los ocho bounded contexts en la aplicación web desplegada, incluyendo gestión de usuarios, gestión de flotas, ejecución de viajes, monitoreo en tiempo real, gestión de alertas, visualización de analytics, gestión de perfiles y suscripciones, todo funcionando con datos simulados mediante fake APIs. Sprint 1 Velocity 45 Story Points Sum of Story Points 42 Story Points

6.2.1.2. Aspect Leaders and Collaborators.

Para el Sprint 1, los principales aspectos considerados corresponden a los bounded contexts core identificados en la arquitectura del sistema CargaSafe, además del Landing Page como punto de entrada al producto. Cada bounded context representa un área funcional específica del sistema que fue desarrollada a nivel de frontend utilizando fake APIs para simular la interacción con servicios backend.

Los aspectos principales son:

1. **Landing Page:** Página de aterrizaje con información del producto y llamados a la acción
2. **Identity and Access Management (IAM):** Gestión de usuarios, autenticación y control de accesos
3. **Fleet Management:** Gestión de vehículos y dispositivos IoT asociados
4. **Trip Management:** Creación, ejecución y seguimiento de viajes
5. **Real-time Monitoring:** Monitoreo de condiciones ambientales en tiempo real
6. **Alerts and Resolution:** Generación, visualización y resolución de alertas
7. **Visualization and Analytics:** Dashboards, gráficos y reportes analíticos
8. **Subscriptions and Payments:** Gestión de planes de suscripción y pagos

A continuación, se presenta la matriz LACX (Leadership and Collaboration Matrix) que indica el líder (L) y los colaboradores (C) para cada aspecto del Sprint 1:

Team Member (Last Name, First Name)	GitHub Username	Landing Page L / C	IAM L / C	Fleet Management L / C	Trip Management L / C	Real-time Monitoring L / C	Alerts L / C	Visualization L / C	Subscriptions L / C
Garro Vega, Marcelo Fabian	Marcelo Garro	C	C	C	L	C	C	C	C
Sanchez Ignacio, Jefrey Martin	Jefreysii20	C	C	C	C	C	C	C	L
Aliaga Pimentel, George Arturo	GeorAliaga	C	L	C	C	C	C	C	C
Vera Nuñez, Nicolas Alejandro	nicxlxs	C	C	L	C	C	C	C	C
Chirinos Arevalo, Daniel Rodrigo	DanielChirinos615	C	C	C	C	L	L	C	C
Bernardo Eusebio, Alessandro Joaquin	Alessandro Bernardo	L	C	C	C	C	C	L	C

Leyenda:

- **L (Leader):** Responsable principal del aspecto, encargado de la coordinación, toma de decisiones técnicas y revisión final del código.
- **C (Collaborator):** Contribuye activamente al desarrollo del aspecto, apoya en la implementación de funcionalidades y participa en las revisiones de código.

Esta matriz facilita la comunicación y clarifica las responsabilidades dentro del equipo, asegurando que cada bounded context tenga un líder definido que coordine su desarrollo, mientras que todos los miembros colaboran de manera transversal para garantizar la integración coherente del sistema.

6.2.1.3. Sprint Backlog 1.

Durante este sprint, se desarrollaron las user stories dedicadas a la landing page y funcionalidades principales del frontend.

La gestión del sprint se llevó a cabo utilizando la herramienta Jira, donde se registraron las tareas, el estado y los responsables.

EN CURSO [2]

- Creación de viajes
[PLANIFICACIÓN DE VIAJES]
CS-32 [GP]
- Actualización de estados de viaje
[PLANIFICACIÓN DE VIAJES]
CS-33 [GP]

EN REVISIÓN [3]

- Registro de usuario
[AUTENTICACIÓN]
CS-16 [GP]
- Cerrar sesión
[AUTENTICACIÓN]
CS-18 [GP]
- Inicio de sesión
[AUTENTICACIÓN]
CS-17 [GP]

LISTO [20] ✓

- Sección portada
[LANDING PAGE]
3 oct 2025
ES-9 [DA]
- Sección de testimonios
[LANDING PAGE]
ES-12 [JS]
- Sección de precios
[LANDING PAGE]
ES-01 [JS]
- Sección de contactanos
[LANDING PAGE]
ES-13 [N]
- Sección Footer
[LANDING PAGE]
ES-02 [N]
- Sección de beneficios
[LANDING PAGE]
ES-11 [GP]
- Sección de funcionalidades
[LANDING PAGE]
ES-10 [AE]
- Call to Action de descarga de App Móvil

Sprint # Sprint 1

User Story Work-Item / Task

User Story Id	User Story Title	Task Id	Task Title	Task Description	Estimation (Hours)	Assigned To	Status
US01	Sección portada	T01	Maquetar versión desktop del hero	Crear estructura visual y contenido de la sección hero en versión desktop	2	Daniel Chirinos	Done
US01	Sección portada	T02	Maquetar versión mobile del hero	Adaptar diseño del hero a dispositivos móviles	2	Daniel Chirinos	Done
US02	Sección de testimonios	T03	Maquetar versión desktop de testimonios	Diseñar estructura y estilo de la sección de testimonios para desktop	2	Jefrey Sanchez	Done
US02	Sección de testimonios	T04	Maquetar versión mobile de testimonios	Adaptar la sección de testimonios a vista móvil	2	Jefrey Sanchez	Done
US03	Sección precios	T05	Maquetar versión desktop de precios	Crear layout de la tabla de precios con estilos y estructura responsive	2	Jefrey Sanchez	Done
US03	Sección precios	T06	Maquetar versión mobile de precios	Optimizar diseño de precios para dispositivos móviles	2	Jefrey Sanchez	Done
US04	Sección contáctanos	T07	Maquetar versión desktop de contáctanos	Crear formulario y disposición de la sección de contacto en desktop	2	Nicolas Vera	Done
US04	Sección contáctanos	T08	Maquetar versión mobile de contáctanos	Adaptar formulario y estilos a versión móvil	2	Nicolas Vera	Done
US05	Sección footer	T09	Maquetar versión desktop del footer	Crear estructura de pie de página con enlaces y estilos base	2	Nicolas Vera	Done
US05	Sección footer	T10	Maquetar versión mobile del footer	Ajustar estructura del footer a vista móvil	2	Nicolas Vera	Done
US06	Sección de beneficios	T11	Maquetar versión desktop de beneficios	Crear diseño informativo para los beneficios del producto	2	George Aliaga	Done

User Story Id	User Story Title	Task Id	Task Title	Task Description	Estimation (Hours)	Assigned To	Status
US06	Sección de beneficios	T12	Maquetar versión mobile de beneficios	Ajustar diseño de beneficios a pantalla móvil	2	George Aliaga	Done
US07	Sección de funcionalidades	T13	Maquetar versión desktop de funcionalidades	Diseñar estructura informativa de funcionalidades IoT	2	Alessandro Bernardo	Done
US07	Sección de funcionalidades	T14	Maquetar versión mobile de funcionalidades	Adaptar funcionalidades IoT a vista móvil	2	Alessandro Bernardo	Done
US08	Call to Action de descarga de App Móvil	T15	Crear CTA de descarga	Implementar botones de descarga para Android e iOS	2	Marcelo Garro	Done
US09	Call to Action a la aplicación web	T16	Crear CTA a la aplicación web	Crear enlace directo desde la landing hacia la app web	2	Marcelo Garro	Done
US10	Internacionalización de contenido	T17	Crear sistema de internacionalización	Implementar cambio dinámico de idioma en la landing	3	George Aliaga	Done
US10	Internacionalización de contenido	T18	Traducir contenido general	Adaptar textos y secciones al idioma inglés	6	George Aliaga	Done
US11	Navegación en landing page	T19	Crear navegación desktop	Implementar menú principal y navegación entre secciones	2	Marcelo Garro	Done
US11	Navegación en landing page	T20	Crear navegación mobile	Crear menú hamburguesa para navegación en móviles	2	Marcelo Garro	Done
US12	Registro de vehículos de carga	T21	Implementar formulario de registro	Diseñar formulario funcional para registrar nuevos vehículos	3	Nicolas Vera	Done
US13	Eliminación de vehículos de carga	T22	Crear acción de eliminación	Implementar botón y lógica para eliminar vehículos registrados	2	Nicolas Vera	Done
US14	Ver todos los vehículos de carga	T23	Mostrar lista de vehículos	Implementar tabla de visualización de vehículos registrados	2	Nicolas Vera	Done
US15	Lista de viajes registrados	T24	Mostrar viajes registrados	Desarrollar listado con datos de viajes del usuario	3	Marcelo Garro	Done
US16	Detalle de viaje	T25	Implementar vista detallada	Mostrar información completa de un viaje seleccionado	3	Marcelo Garro	Done
US17	Filtrado de viajes por fecha	T26	Crear filtros por fecha	Implementar filtros dinámicos en la lista de viajes	2	Nicolas Vera	Done
US18	Historial de pagos	T27	Mostrar historial de pagos	Desarrollar interfaz para mostrar transacciones realizadas	2	Jefrey Sanchez	Done
US19	Confirmar suscripción	T28	Crear flujo de confirmación	Implementar pantalla de confirmación para suscripción activa	1	Jefrey Sanchez	Done
US20	Cancelar suscripción	T29	Implementar cancelación de suscripción	Agregar opción y modal para cancelar suscripción	1	Jefrey Sanchez	Done
US21	Visualizar información de suscripción actual	T30	Mostrar datos de suscripción	Crear interfaz para visualizar tipo de plan y estado actual	2	Nicolas Vera	Done
US22	Recuperar contraseña	T31	Implementar recuperación de contraseña	Desarrollar flujo para restablecer credenciales de usuario	1	George Aliaga	Done
US23	Registro de dispositivos IoT	T32	Crear formulario de registro IoT	Diseñar formulario para vincular nuevos dispositivos IoT	3	Nicolas Vera	Done
US24	Eliminar dispositivo IoT	T33	Implementar eliminación de dispositivo	Agregar opción para eliminar dispositivos registrados	2	Nicolas Vera	Done
US25	Asignar dispositivo a vehículo de carga	T34	Crear flujo de asignación	Desarrollar relación entre vehículo y dispositivo IoT	2.5	Nicolas Vera	Done
US26	Ver todos los dispositivos IoT en la plataforma	T35	Listar dispositivos IoT	Mostrar todos los dispositivos vinculados a la cuenta	2	Nicolas Vera	Done
US27	Ver estado de dispositivo por vehículo	T36	Mostrar estado del dispositivo	Implementar vista de estado de sensor IoT por vehículo	2	Nicolas Vera	Done
US28	Alertas por incumplimiento de temperatura	T37	Implementar alertas IoT	Crear sistema de notificación por temperaturas fuera de rango	2	Daniel Chirinos	Done

User Story Id	User Story Title	Task Id	Task Title	Task Description	Estimation (Hours)	Assigned To	Status
US29	Gráficos de tiempo y temperatura	T38	Crear gráficos de temperatura	Implementar gráficos de línea para mostrar variaciones térmicas	4	Alessandro Bernardo	Done
US30	Gráficos de incidencias por mes	T39	Crear gráfico de incidencias	Mostrar estadísticas mensuales de incidentes	4	Alessandro Bernardo	Done
US33	Filtrado de viajes por fecha	T43	Implementar opciones de filtrado	Crear filtros dinámicos por fecha y estado de viaje	1	Marcelo Garro	Done

6.2.1.4. Development Evidence for Sprint Review

Durante el sprint, en la web application, web services, mobile application, edge application y embemdded application se levantó la base del proyecto, se añadieron las secciones tales como (Features, Benefits, Testimonials, Plans, Contact Us), navegación, soporte bilingüe (EN/ES). Se documentó el proyecto y se ajustó la configuración de build/ignores.

Landing Page

Repository	Branch	Commit Id	Commit Message	Commit Message Body	Committed on (Date)
cargasafe-landing	main	e4aaf65	build(base): setup base template for development	Estructura inicial del proyecto, tooling básico y configuración para desarrollo.	2025-10-03
cargasafe-landing	hero-section	5fb53f0	feat(hero-section): add hero-section info	Sección Hero con copy, botón CTA y layout responsive.	2025-10-03
cargasafe-landing	functionalities	4816719	feat(functionality): add functionality info section	Sección Funcionalidad : bloques con beneficios funcionales y grid responsive.	2025-10-03
cargasafe-landing	benefits	5101ccc	feat(benefits): add benefits section	Sección Beneficios con íconos/texto, espaciados y accesibilidad básica.	2025-10-04
cargasafe-landing	pricing	fd9db54	feat(pricing): add subscription plans section	Sección Pricing (planes) con cards y CTA a suscripción.	2025-10-04
cargasafe-landing	testimonials	a7f149f	feat(testimonials): add testimonials section	Sección Testimonials con testimonios y estructura para futuras fuentes.	2025-10-04
cargasafe-landing	quotes-section	6a51bac	feat(quotes-section): add quotes-section info	Sección Quotes (frases) con tipografía y espacio consistente.	2025-10-05
cargasafe-landing	navigation	63f1a66	feat(nav): add header and navigation system for landing page	Header + navegación anclada; enlaces a secciones y estados de hover/focus.	2025-10-06
cargasafe-landing	call-to-action	1223a55	feat(cta): add app web call-to-action	CTA para App Web enlazado; estilos y jerarquía visual.	2025-10-06
cargasafe-landing	call-to-action	8ad26fc	feat(cta): add app mobile call-to-action	CTA para App Mobile ; botones y layout responsive.	2025-10-06
cargasafe-landing	call-to-action	6df4e47	feat(cta): add responsive for call-to-action section	Ajustes responsive de la sección CTA (breakpoints y spacing).	2025-10-06
cargasafe-landing	intl	4c8225d	feat(intl): add english and spanish language to the entire page	i18n EN/ES aplicado a toda la landing; textos externalizados.	2025-10-06
cargasafe-landing	intl	75003bc	feat(intl): add switch language button for layout	Switcher de idioma en el layout (toggle EN/ES) y persistencia simple.	2025-10-06
cargasafe-landing	main	f1f9917	docs: add initial project documentation for landing page	README/guía inicial: estructura, scripts y cómo correr/desplegar.	2025-10-06
cargasafe-landing	main	b0327e7	chore(gitignore): create gitignore file to ignore IDEs configurations	.gitignore para excluir configs locales/IDE.	2025-10-06

Web Application

Durante este sprint se ajustaron los módulos del sistema **CargaSafe**. Se integró el backend desplegado en la web application, y solo se utilizaron las pantallas con las funcionalidades core.

Repository	Branch	Commit Id	Commit Message	Commit Message Body	Committed on (Date)
iot-solutions-development-cargasafe-frontend	main	c8dc444	initial commit	Inicio del proyecto Angular 20, configuración base y estructura inicial.	2025-09-23

Repository	Branch	Commit Id	Commit Message	Commit Message Body	Committed on (Date)
iot-solutions-development-cargasafe-frontend	main	f8215b4	feat: First commit. Project Setup	Creación del entorno Angular standalone, configuración inicial de módulos y dependencias.	2025-09-23
iot-solutions-development-cargasafe-frontend	develop	ebcd034	feat(shared): implement base entity, API endpoint, assembler, and response interfaces	Implementación de entidades base, endpoints comunes y ensambladores para integración de datos.	2025-10-03
iot-solutions-development-cargasafe-frontend	develop	7026161	feat(shared): implement root layout and routing system	Configuración del layout principal con sidebar y rutas dinámicas.	2025-10-03
iot-solutions-development-cargasafe-frontend	develop	00a9220	build(fake-api): add json-server for API REST connections	Implementación del servidor JSON local para simular endpoints REST del backend.	2025-10-07
iot-solutions-development-cargasafe-frontend	IAM	ffa780a	feat(iam): implement login, register and recovery password pages	Creación de pantallas de autenticación (login, registro, recuperación de contraseña).	2025-10-07
iot-solutions-development-cargasafe-frontend	trip-management	9d54bb9	feat(trips): add DeliveryOrder, OriginPoint, TripParameter entities and update Trip entity with additional properties	Entidades principales del módulo de viajes (trip, delivery order, parámetros y origen).	2025-10-07
iot-solutions-development-cargasafe-frontend	trip-management	e5e3f55	feat(trips): add trip list page	Página principal de viajes con tabla, filtro y paginación.	2025-10-07
iot-solutions-development-cargasafe-frontend	trip-management	6553425	feat(trips): implement trips feature with API integration and routing	Integración del módulo de viajes con servicios REST simulados.	2025-10-07
iot-solutions-development-cargasafe-frontend	alerts-and-resolutions	f2e09c0	refactor(alert screen): new icons and added db.json	Refactor visual de alertas, íconos actualizados y datos simulados.	2025-10-07
iot-solutions-development-cargasafe-frontend	alerts-and-resolutions	7564492	feat(alerts): add alert screens	Sección de alertas del sistema con componentes modulares.	2025-10-07
iot-solutions-development-cargasafe-frontend	visualization-and-analytics	9ed2471	feat(dashboard): Added charts for trip monitoring information	Dashboard con gráficos de seguimiento de viajes y estadísticas mensuales.	2025-10-07
iot-solutions-development-cargasafe-frontend	fleet-management	81e45b0	feat(fleet): scaffold bounded context folders	Estructura base para el módulo de gestión de flotas.	2025-10-08
iot-solutions-development-cargasafe-frontend	develop	fe46756	feat(env): add base environment files and production config	Configuración de entornos (environment.ts) y build de producción.	2025-10-08
iot-solutions-development-cargasafe-frontend	fleet-management	7d43fd7	feat(fleet-domain): add vehicle and device models	Creación de modelos para vehículos y dispositivos IoT.	2025-10-08
iot-solutions-development-cargasafe-frontend	fleet-management	daf642c	feat(fleet-data): add device and vehicle data-access services	Servicios para gestión de datos de flota y dispositivos conectados.	2025-10-08
iot-solutions-development-cargasafe-frontend	main	a37a0b7	feat(fleet-ui): add device-create-and-edit component	Formulario y modal para creación/edición de dispositivos.	2025-10-08
iot-solutions-development-cargasafe-frontend	fleet-management	47ad853	feat(fleet-ui): add vehicle-create-and-edit component	Componente para registrar o editar vehículos.	2025-10-08
iot-solutions-development-cargasafe-frontend	fleet-management	d1578c7	feat(fleet-pages): add device-management page	Página de administración de dispositivos.	2025-10-08
iot-solutions-development-cargasafe-frontend	fleet-management	e79cef8	feat(fleet-pages): add vehicle-management page	Página de administración de vehículos.	2025-10-08

Repository	Branch	Commit Id	Commit Message	Commit Message Body	Committed on (Date)
iot-solutions-development-cargasafe-frontend	develop	ba3f415	feat(routes): add/modify Fleet routes and guards	Configuración de rutas y guardas para navegación segura en Fleet.	2025-10-08
iot-solutions-development-cargasafe-frontend	develop	37cc80c	feat(shared-layout): add nav links to Vehicles and Devices management	Enlaces de navegación desde el layout principal hacia módulos de flota.	2025-10-08
iot-solutions-development-cargasafe-frontend	subscriptions-bounded-context	94e33ba	feat(subscription): update bc content	Ajustes al <i>bounded context</i> de suscripciones (contenidos y estados).	2025-10-08
iot-solutions-development-cargasafe-frontend	develop	25b4a74	fix: delete import zone.js in app.ts	Corrección de import redundante en configuración principal.	2025-10-08

- También se agregaron gráficos de monitoreo y pantallas de autenticación, completando el alcance funcional planificado para el sprint.

6.2.1.5. Testing Suite Evidence for Sprint Review.

Durante este sprint no se implementaron pruebas automatizadas (Unit Tests, Integration Tests ni Acceptance Tests), ya que el alcance definido por el equipo contemplaba únicamente la construcción de la Landing Page así como la Aplicación Web y su conexión con una Fake API mediante [json-server](#).

El objetivo principal del sprint fue lograr la integración funcional entre la interfaz Angular y los endpoints simulados, estableciendo la arquitectura modular y los bounded contexts principales del sistema CargaSafe.

En esta etapa aún no se contaba con un backend definitivo ni con los servicios reales, por lo que las pruebas automatizadas se reprogramaron para el siguiente sprint, donde se incluirán los Integration Tests y Acceptance Tests asociados a los módulos de Suscriptions, Alerts, Fleet y Trips.

Conclusión: La etapa de testing se limitó a verificaciones manuales de flujo, validación visual de componentes y pruebas de integración básicas con el servidor [json-server](#).

No se realizaron commits relacionados a proyectos o archivos de testing automatizado, dado que esta funcionalidad no formaba parte del alcance planificado para el sprint actual.

6.2.1.6. Execution Evidence for Sprint Review.

Durante este sprint se alcanzó la implementación funcional de la Landing Page y la Aplicación Web de CargaSafe, cumpliendo con los objetivos planteados en el alcance del sprint.

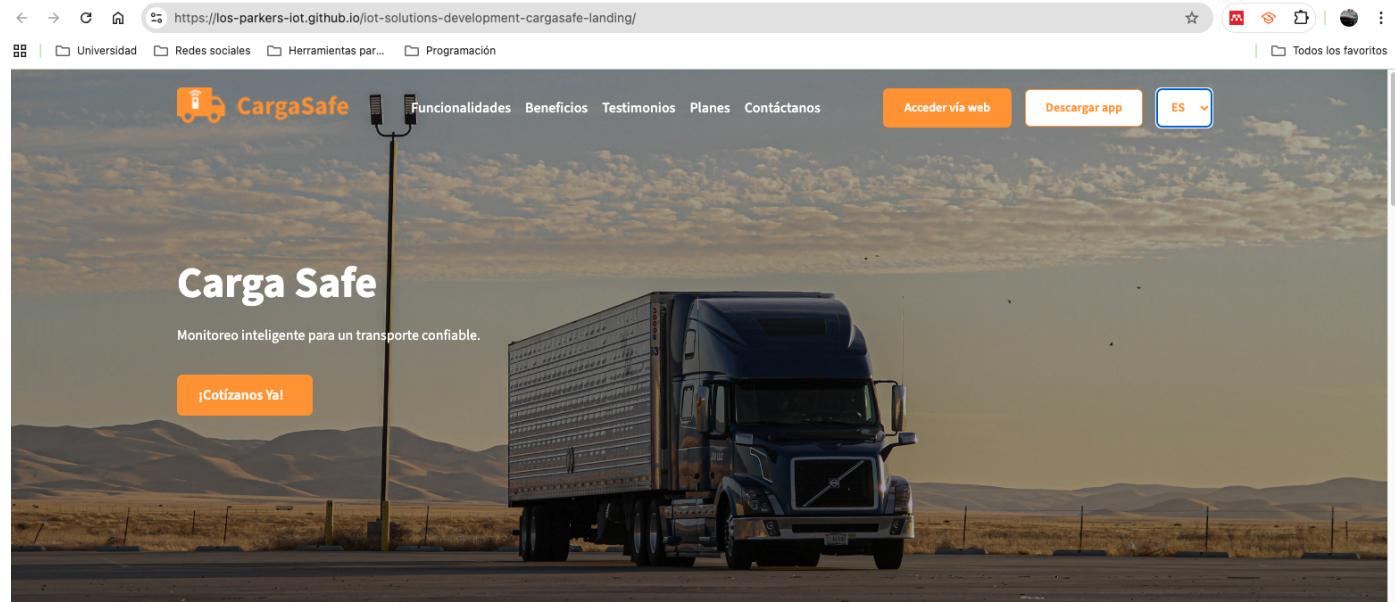
La landing fue desarrollada con HTML, CSS y Angular Material*, enfocada en comunicar los beneficios del producto, sus planes de suscripción y la posibilidad de acceder directamente a la aplicación web.

Por otro lado, la aplicación web fue construida con Angular 20 bajo arquitectura modular standalone, integrándose con una Fake API ([json-server](#)) para simular la conexión con el backend y visualizar datos reales en tiempo de ejecución.

A continuación, se muestran los principales resultados obtenidos:

Landing Page

Vista inicial del encabezado principal y navegación



FUNCIONALIDADES

¿Qué hace CargaSafe?



Sección de planes de suscripción con precios y descripción de beneficios

Repositorio: [Click aquí](#) Enlace: [Click aquí](#)

Web Application

Dashboard principal con métricas de viajes, incidencias y alertas activas

The screenshot shows the main dashboard of the CargaSafe application. On the left is a sidebar with orange headers and white icons for Dashboard, Vehicles, Sensors, Trips, Alerts, and Subscriptions. The main area has a white header "Dashboard - Monitoreo de Cargas" and a sub-header "Visualización y análisis de viajes, incidencias y alertas del sistema CargaSafe". Below this are four cards with summary data:

TOTAL DE VIAJES	VIAJES ACTIVOS	ALERTAS TOTALES	ALERTAS PENDIENTES
7	6	7	3

Below these cards is a section titled "Gráfico de Incidencias por Mes" (Incidents by Month) featuring a bar chart. The chart has two series: red bars representing incidents and blue bars representing resolved incidents. The Y-axis ranges from 5 to 9. The X-axis represents months, though labels are not visible.

Vista del módulo de alertas con listado, estado y acciones de resolución

The screenshot shows the "Alerts" module. It features a sidebar with the same orange header and icons as the main dashboard. The main area has a title "Alerts" and two summary boxes: one for "Alerts today" (2) and one for "Resolved" (3). Below this is a search bar with a magnifying glass icon and a dropdown menu for "Status". A table lists five alerts with columns for ID, Type, Delivery Order ID, Status, Created, Closed, Actions, and Details. The first four alerts are closed, while the fifth is active. Buttons for "Mark as resolved" are shown for the active alerts.

ID	Type	Delivery Order ID	Status	Created	Closed	Actions	Details
1	High Temperature	T342	Closed	Wed Oct 08 2025 16:05:14 GMT-0500 (hora estándar de Perú)	Wed Oct 08 2025 16:05:26 GMT-0500 (hora estándar de Perú)	Resolved	
2	Excessive vibration	T343	Closed	Wed Oct 08 2025 16:41:00 GMT-0500 (hora estándar de Perú)	Wed Oct 08 2025 16:41:04 GMT-0500 (hora estándar de Perú)	Resolved	
3	Energy Loss	T344	Closed	Wed Oct 08 2025 16:41:04 GMT-0500 (hora estándar de Perú)	Wed Oct 08 2025 16:41:04 GMT-0500 (hora estándar de Perú)	Resolved	
4	Trip Loss	T345	Active	Wed Oct 08 2025 20:08:42 GMT-0500 (hora estándar de Perú)	---	Mark as resolved	
5	Gate Loss	T346	Active	Wed Oct 08 2025 20:08:42 GMT-0500 (hora estándar de Perú)	---	Mark as resolved	

At the bottom are pagination controls: "Items per page: 5", "1 - 5 of 5", and navigation arrows.

Repositorio: [Click aquí](#) Aplicación en línea: [Click aquí](#)

Video de ejecución del Sprint:

[Click aquí para ver el video](#)

Conclusión:

- El resultado de este sprint permitió disponer de una versión navegable del producto, mostrando coherencia entre la landing pública y la aplicación web conectada al servidor simulado.
- Esto consolida la base para el siguiente sprint, donde se implementarán los servicios reales y la suite de testing automatizado.

6.2.1.7. Services Documentation Evidence for Sprint Review

Durante este sprint no se elaboró documentación de Web Services ni endpoints bajo el estándar OpenAPI, debido a que el desarrollo del backend no formó parte del alcance establecido para esta iteración.

El objetivo del sprint se centró en la implementación del frontend (Landing Page y Aplicación Web) y su conexión con una Fake API utilizando [json-server](#). Este enfoque permitió validar la arquitectura del cliente, la estructura modular de Angular, y el flujo de consumo de datos simulados, sin requerir aún servicios reales documentados.

La documentación de los endpoints (métodos HTTP, parámetros y responses) será desarrollada en el siguiente sprint, una vez que se inicie la construcción del backend real, lo que permitirá generar la especificación OpenAPI/Swagger correspondiente.

Conclusión:

- En esta entrega no se generaron commits ni documentación asociada a Web Services, ya que la prioridad del sprint fue garantizar la integración funcional del frontend con la API simulada.
- Esto sienta las bases para una documentación formal de servicios en las próximas iteraciones, cuando se implementen los endpoints reales y sus especificaciones técnicas.

6.2.1.8. Software Deployment Evidence for Sprint Review

Durante este sprint se realizaron los despliegues de los productos digitales correspondientes al **frontend** de CargaSafe: la **Landing Page** y la **Aplicación Web**. El proceso incluyó la configuración de entornos de hosting, integración con servicios de despliegue continuo (CI/CD) y verificación de accesibilidad pública de ambos proyectos.

Despliegue de la Web Application

La aplicación principal de CargaSafe, desarrollada en Angular 20, fue desplegada utilizando Firebase Hosting bajo el plan Spark.

Se configuró la conexión entre el repositorio en GitHub y Firebase a través de GitHub Actions, permitiendo un despliegue automatizado en cada push hacia la rama principal.

Panel principal de Firebase mostrando la compilación e implementación exitosa del proyecto

The screenshot shows the Firebase Hosting dashboard for the 'carga-safe' project. On the left, a sidebar includes links for Descripción general, Accesos directos a proyectos, Hosting (selected), App Check, Novedades, AI Logic (NUEVO), Categorías de producto, Compilación, Ejecución, Analytics, IA, Herramientas de desarrollo relacionadas, and Firebase Studio. At the bottom, it shows 'Spark Sin costo (USD 0 al mes)' and a 'Actualizar' button. The main area has tabs for Panel (selected) and Uso. The 'Administrador sitio' section shows a successful deployment from 'github-action-1062335524@carga-safe.iam.gserviceaccount.com' on '8/10/25, 7:24 p.m.' with commit '4ba51d'. Below this, 'Versiones anteriores' lists four previous deployments with commits '4ba51d', '098a22', 'cb96d4', and '0942cb'. To the right, the 'Dominios' section shows two active domains: 'carga-safe.web.app' (Predeterminado) and 'carga-safe.firebaseioapp.com' (Predeterminado). Buttons for 'Configuración de almacenamiento de actualizaciones', 'Agregar un dominio personalizado', and 'Ver los 2 dominios' are also present.

Historial de implementaciones y dominio activo del proyecto

The screenshot shows the Firebase console interface for the 'carga-safe' project. The left sidebar includes sections for Hosting, App Check, Novedades, AI Logic (NUEVO), Categorías de producto, Compilación, Ejecución, Analytics, and IA. The main area displays the 'Compilación' section for 'Hosting', which shows a deployment log for 'Implementado' on October 8, 2025, at 7:24 p.m., implemented by 'github-action-1062335524@carga-safe.iam.gserviceaccount.com'. Below this, it says 'No hay datos durante los últimos 14 días'. At the bottom, there are two cards: one for providing app details and another for adding statistics and monitoring.

Este despliegue permitió verificar el funcionamiento completo de la aplicación conectada al [json-server](#), garantizando la disponibilidad pública para pruebas funcionales del sprint.

Despliegue de la Landing Page

La Landing Page fue desplegada en [GitHub Pages](#) desde el repositorio del equipo, utilizando la rama [main](#) como fuente de publicación. El flujo de integración continua se configuró directamente desde la sección [Pages](#) del repositorio, lo que permite que la página se actualice automáticamente con cada cambio confirmado.

Configuración y despliegue de la landing mediante GitHub Pages

The screenshot shows the GitHub repository settings for 'Los-Parkers-IoT/iot-solutions-development-cargasafe-landing'. The 'Pages' tab is selected. On the left, there's a sidebar with options like General, Access, Collaborators and teams, Moderation options, Code and automation, Branches, Tags, Rules, Actions, Models, Webhooks, Copilot, Environments, Pages (which is selected), Custom properties, Security, Advanced Security, Deploy keys, Secrets and variables, Integrations, and GitHub Apps. The main content area is titled 'GitHub Pages' and contains the following information: 'GitHub Pages is designed to host your personal, organization, or project pages from a GitHub repository.' Below this, it says 'Your site is live at <https://los-parkers-iot.github.io/iot-solutions-development-cargasafe-landing/>' and 'Last deployed by SgtGarro 2 days ago'. There are 'Visit site' and 'Unpublish site' buttons. Under 'Build and deployment', it says 'Source' and 'Deploy from a branch' (with 'develop' selected). It also mentions 'Branch' and 'Your GitHub Pages site is currently being built from the develop branch.' The 'Custom domain' section is also present.

URL de la landing:

[Click aquí para ver la Landing Page](#)

GitHub Actions
Conclusión:

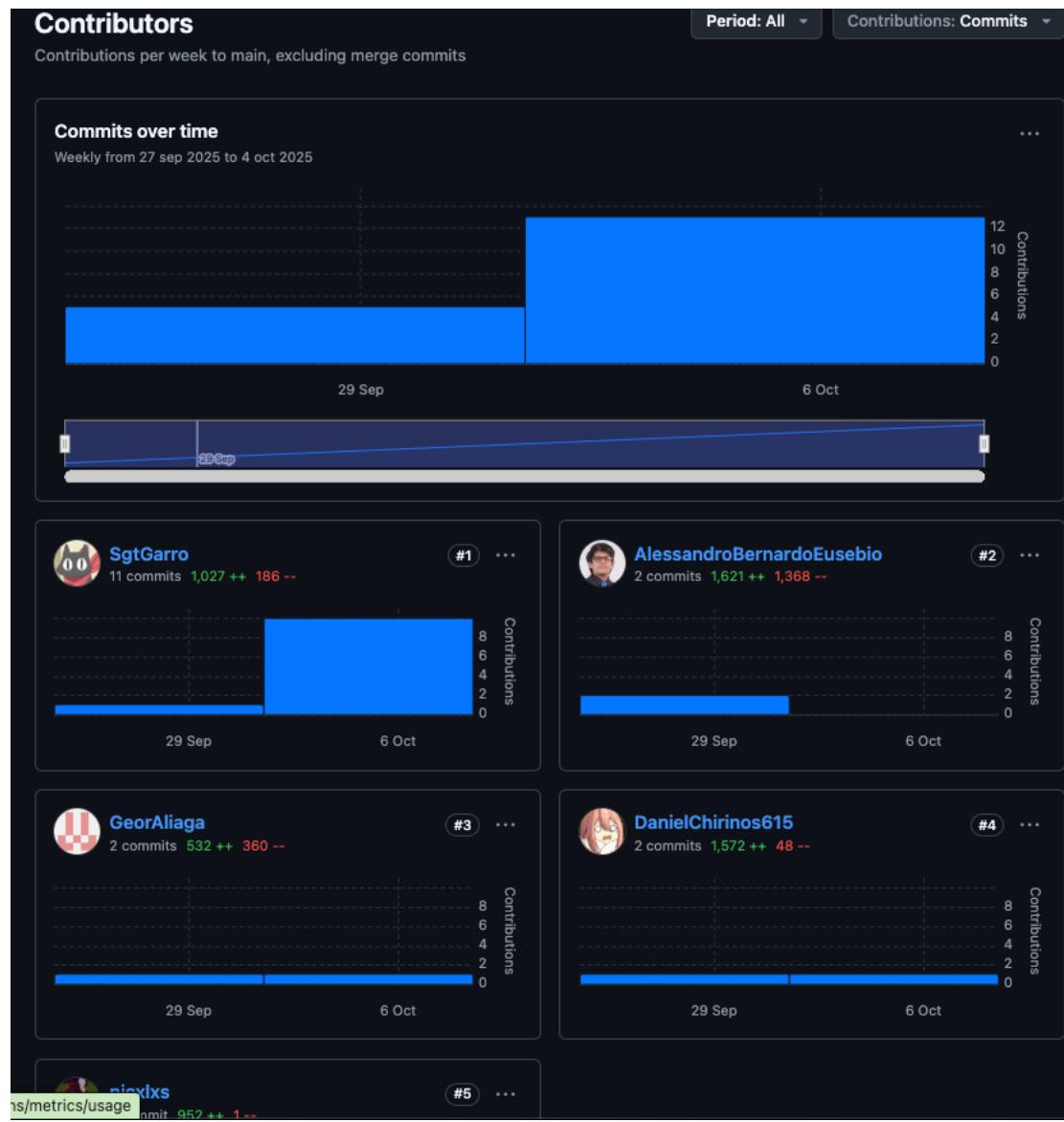
- Ambos despliegues se realizaron correctamente en entornos de hosting gratuitos (Firebase Hosting y GitHub Pages), asegurando la disponibilidad pública de los productos y su correcta visualización en navegadores.
- Estos procesos sentaron la base para automatizar futuras integraciones de backend y testing continuo en los siguientes sprints.

6.2.1.9. Team Collaboration Insights during Sprint.

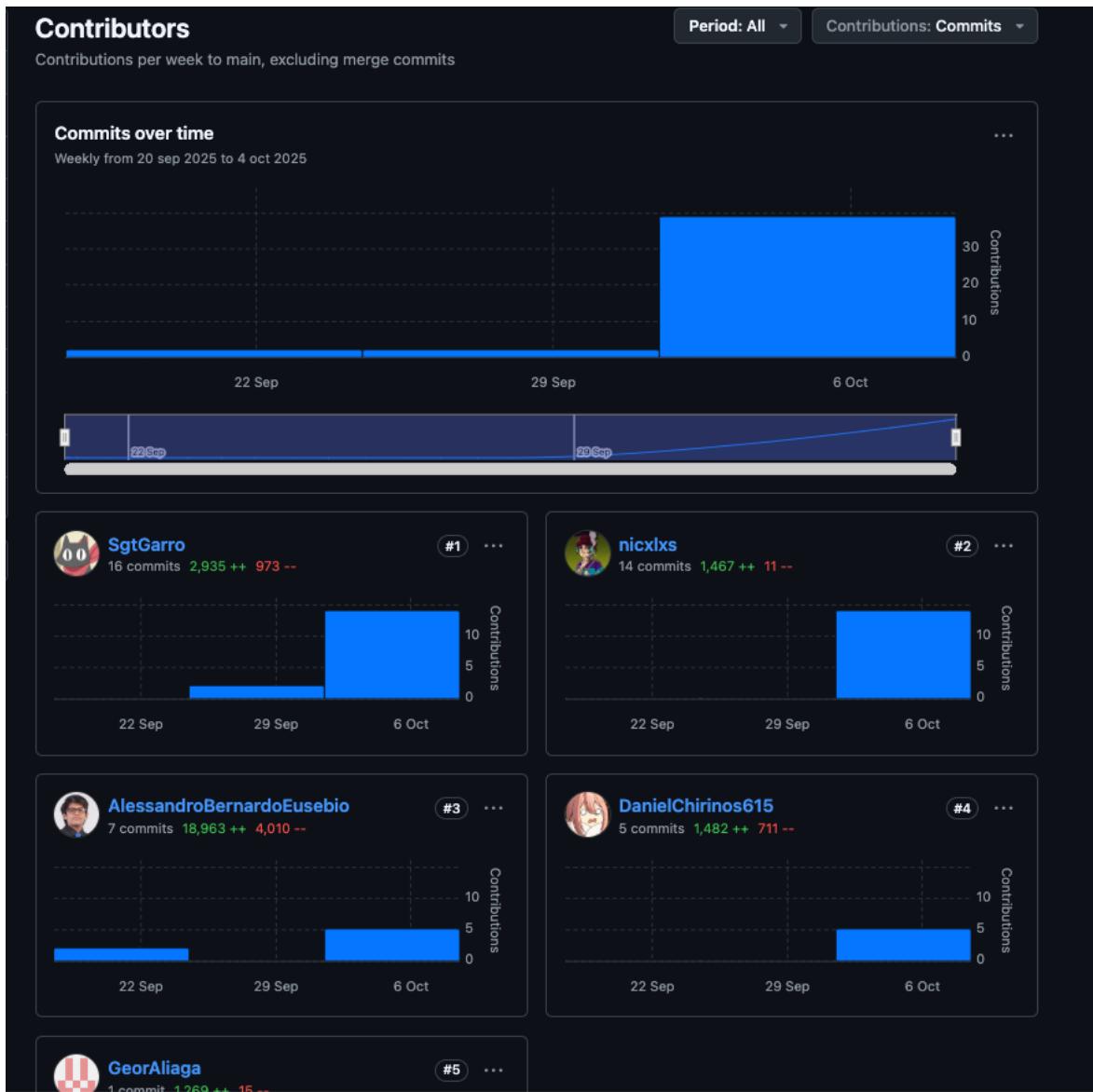
Durante este sprint, el equipo de desarrollo trabajó de forma colaborativa en la implementación de las principales funcionalidades correspondientes al alcance definido: la Landing Page y la Aplicación Web de CargaSafe. A lo largo del proceso, se mantuvo una comunicación constante mediante GitHub y los canales del equipo, asegurando una adecuada distribución de tareas y el cumplimiento de los objetivos propuestos.

Los analíticos de GitHub reflejan la participación activa de todos los integrantes, evidenciando commits frecuentes, revisiones de código y fusiones entre ramas, lo que demuestra un flujo de trabajo coordinado y una integración continua del proyecto. Además, se promovió la retroalimentación mutua y la resolución conjunta de incidencias técnicas, fortaleciendo el liderazgo compartido y la cooperación dentro del equipo.

Insights de Landing Page



Insights de Web Application



6.2.2. Sprint 2

6.2.2.1. Sprint Planning 2.

En esta sección se especifican los aspectos principales del Sprint Planning Meeting para el Sprint 2 del proyecto CargaSafe. Este primer sprint se enfoca en establecer las bases de la plataforma, implementando el landing page y los componentes iniciales de la aplicación web que permitan a los usuarios conocer el producto y comenzar a interactuar con las funcionalidades core del sistema de monitoreo de carga.

Sprint #
Sprint 2

Sprint Planning Background Date 2025-10-19 Time 09:00 AM Location virtual mediante Discord Prepared By Marcelo Fabian Garro Vega Attendees (to planning meeting) Marcelo Fabian Garro Vega / Jefrey Martin Sanchez Ignacio / Daniel Rodrigo Chirinos Arevalo / Alessandro Joaquin Bernardo Eusebio / Vera Nuñez Nicolas Alejandro / George Aliaga Sprint 2 Review Summary Durante el Sprint 2 se implementaron los módulos core de la plataforma correspondientes a Trip Management, Fleet Management, Monitoring Management y Alerts Management. Asimismo, se logró integrar el prototipo IoT con la aplicación web mediante datos simulados, permitiendo validar el flujo básico de monitoreo y visualización de telemetría. Sprint 2 Retrospective Summary El equipo mantuvo una adecuada coordinación durante el Sprint 2, logrando cumplir con los objetivos planteados. Se identificó como oportunidad de mejora la optimización en la estimación de tiempos para tareas de integración, acordándose fortalecer las pruebas técnicas en el siguiente sprint. Sprint Goal & User Stories Sprint 2 Goal En esta segunda iteración del proyecto, nuestro enfoque estuvo en consolidar y validar las funcionalidades core de CargaSafe, priorizando los módulos directamente relacionados con el monitoreo de carga sensible. Para esta sprint se trabajó únicamente con los bounded contexts esenciales para esta etapa: **Trip Management, Fleet Management, Monitoring Management y Alerts Management**, dejando de lado aquellos que no son críticos en esta fase, como gestión de perfiles, pagos o analytics avanzados.

Además, durante esta sprint se desarrolló el **prototipo IoT físico**, conformado por sensores de temperatura, humedad y un módulo GPS, permitiendo capturar telemetría real y validar el funcionamiento de la solución extremo a extremo. Este prototipo se integró con la aplicación web mediante endpoints simulados y datos procesados, lo que permitió evaluar la experiencia de monitoreo en escenarios reales y reproducibles.

El objetivo principal de esta sprint es demostrar de manera funcional cómo CargaSafe permite visualizar el estado de un viaje, monitorear telemetría en tiempo real y gestionar alertas generadas por cambios bruscos en las condiciones de la carga. Consideraremos que esto sienta las bases para validar nuestro producto con stakeholders.

y usuarios finales, confirmando que las funcionalidades centrales responden a necesidades reales del mercado.

Esta validación quedará confirmada cuando los usuarios puedan navegar las funcionalidades priorizadas —inicio y seguimiento de viajes, visualización de temperatura/humedad/ubicación y revisión del ciclo de vida de alertas— mientras el prototipo IoT genera datos reales, permitiendo observar cómo el sistema reacciona ante variaciones en el entorno y cómo se presenta esa información de manera clara en la aplicación web.

Sprint 2 Velocity 48 Story Points Sum of Story Points 46 Story Points

6.2.2.2. Aspect Leaders and Collaborators.

Para el Sprint 2, los principales aspectos considerados corresponden a los bounded contexts core identificados en la arquitectura del sistema CargaSafe, además del Landing Page como punto de entrada al producto. Se estableció un nuevo punto acerca del Embedded Application.

Los aspectos principales son:

1. **Landing Page:** Página de aterrizaje con información del producto y llamados a la acción
2. **Fleet Management:** Gestión de vehículos y dispositivos IoT asociados
3. **Trip Management:** Creación, ejecución y seguimiento de viajes
4. **Real-time Monitoring:** Monitoreo de condiciones ambientales en tiempo real
5. **Alerts and Resolution:** Generación, visualización y resolución de alertas
6. **Embedded Application:** Dispositivo IoT que funciona para monitoreo

A continuación, se presenta la matriz LACX (Leadership and Collaboration Matrix) que indica el líder (L) y los colaboradores (C) para cada aspecto del Sprint 2:

Team Member (Last Name, First Name)	GitHub Username	Landing Page L / C	Fleet Management L / C	Trip Management L / C	Real-time Monitoring L / C	Alerts L / C	Embedded Application L / C
Garro Vega, Marcelo Fabian	Marcelo Garro	C	C	C	L	C	C
Sanchez Ignacio, Jefrey Martin	Jefreysii20	C	C	C	C	C	C
Aliaga Pimentel, George Arturo	GeorAliaga	C	L	C	C	C	C
Vera Nuñez, Nicolas Alejandro	nicxlxs	C	C	L	C	C	C
Chirinos Arevalo, Daniel Rodrigo	DanielChirinos615	C	C	C	C	L	L
Bernardo Eusebio, Alessandro Joaquin	Alessandro Bernardo	L	C	C	C	C	C

Leyenda:

- **L (Leader):** Responsable principal del aspecto, encargado de la coordinación, toma de decisiones técnicas y revisión final del código.
- **C (Collaborator):** Contribuye activamente al desarrollo del aspecto, apoya en la implementación de funcionalidades y participa en las revisiones de código.

Esta matriz facilita la comunicación y clarifica las responsabilidades dentro del equipo, asegurando que cada bounded context tenga un líder definido que coordine su desarrollo, mientras que todos los miembros colaboran de manera transversal para garantizar la integración coherente del sistema.

6.2.2.3. Sprint Backlog 2.

Durante este sprint, se desarrollaron las user stories dedicadas a la landing page y funcionalidades principales del frontend, backend y mobile.

La gestión del sprint se llevó a cabo utilizando la herramienta Jira, donde se registraron las tareas, el estado y los responsables.

Sprint #		Sprint 2									
User Story	Work-Item / Task										
GS-38 Lista de viajes registrados	PLANIFICACIÓN DE ...	FINALIZADA ✓	25 oct	5	=	JS					
GS-24 Registro de dispositivos IoT	GESTIÓN DE FLOTA	FINALIZADA ✓	29 oct	5	=	N					
GS-42 Filtrado de viajes por fecha	PLANIFICACIÓN DE ...	FINALIZADA ✓	7 nov	3	=	JS					
GS-35 Obtener datos de telemetría en tiempo real	MONITOREO EN TIEM...	FINALIZADA ✓	30 oct	8	=	GP					
GS-28 Ver todos los vehículos de carga	GESTIÓN DE FLOTA	FINALIZADA ✓	25 oct	3	=	N					
GS-21 Registro de vehículos de carga	GESTIÓN DE FLOTA	FINALIZADA ✓	26 oct	5	=	N					
GS-36 Alertas por incumplimiento de parámetros	MONITOREO EN TIEM...	FINALIZADA ✓	27 oct	5	=	DA					
GS-29 Ver todos los dispositivos IoT en la plataforma	GESTIÓN DE FLOTA	FINALIZADA ✓	25 oct	3	=	AE					
GS-26 Asignar dispositivo a vehículo de carga	GESTIÓN DE FLOTA	FINALIZADA ✓	2 nov	3	=	AE					
GS-39 Detalle de viaje	PLANIFICACIÓN DE ...	FINALIZADA ✓	2 nov	3	=	JS					
GS-32 Creación de viajes y puntos de envíos	PLANIFICACIÓN DE ...	FINALIZADA ✓	8 nov	5	=	MG					

User Story Id	User Story Title	Task Id	Task Title	Task Description	Estimation (Hours)	Assigned To	Status
US32	Gestionar y visualizar viajes registrados	T01	Lista de viajes registrados	Crear layout de viajes registrados en web	2	Marcelo Garro	Done
US32	Gestionar y visualizar viajes registrados	T02	Lista de viajes registrados	Crear endpoint de viajes registrados por comercio	2	Jefrey Sanchez	Done
US32	Gestionar y visualizar viajes registrados	T03	Lista de viajes registrados	Mostrar en pantalla los viajes registrados	2	Jefrey Sanchez	Done
US16	Registrar dispositivos IoT en la plataforma	T04	Registro de dispositivos IoT	Crear formulario de registro en app web	4	Nicolas Vera	Done
US16	Registrar dispositivos IoT en la plataforma	T05	Registro de dispositivos IoT	Crear endpoint para crear dispositivos IoT	4	Alessandro Bernardo	Done
US16	Registrar dispositivos IoT en la plataforma	T06	Registro de dispositivos IoT	Conectar formulario de registro con backend en app web	6	Alessandro Bernardo	Done
US16	Registrar dispositivos IoT en la plataforma	T07	Registro de dispositivos IoT	Conectar formulario de registro con backend en app web	6	Alessandro Bernardo	Done
US36	Filtrar viajes por rangos de fecha	T08	Filtrado de viajes por fecha	Mostrar viajes en base a rango de fechas	4	Jefrey Sanchez	Done
US36	Filtrar viajes por rangos de fecha	T09	Filtrado de viajes por fecha	Crear endpoint para consultar viajes en base a fechas	5	Jefrey Sanchez	Done
US28	Obtener y procesar datos de telemetría en tiempo real	T10	Obtener datos de telemetría en tiempo real	Crear bounded context de monitoring en backend	4	George Aliaga	Done
US28	Obtener y procesar datos de telemetría en tiempo real	T11	Obtener datos de telemetría en tiempo real	Crear logica de obtener datos de telemetria del dispositivo y mandar data al edge application	5	George Aliaga	Done
US28	Obtener y procesar datos de telemetría en tiempo real	T12	Obtener datos de telemetría en tiempo real	Crear bounded context de monitoring en edge application	7	George Aliaga	Done
US20	Visualizar todos los vehículos de carga registrados	T13	Ver todos los vehículos de carga	Maquetar layout de vehiculos en modo lista en web	3	Nicolas Vera	Done
US20	Visualizar todos los vehículos de carga registrados	T14	Ver todos los vehículos de carga	Maquetar layout de vehiculos en modo lista en mobile	3	Nicolas Vera	Done
US20	Visualizar todos los vehículos de carga registrados	T15	Ver todos los vehículos de carga	Crear endpoint para gestionar los vehiculos	5	Nicolas Vera	Done
US13	Registrar vehículos de carga en el sistema	T16	Registro de vehículos de carga	Crear formulario de registro en web	3	Nicolas Vera	Done
US13	Registrar vehículos de carga en el sistema	T17	Registro de vehículos de carga	Crear formulario de registro en mobile	3	Nicolas Vera	Done
US13	Registrar vehículos de carga en el sistema	T18	Registro de vehículos de carga	Crear endpoint para crear vehiculo	5	Nicolas Vera	Done
US29	Gestionar alertas por incumplimiento de parámetros operativos	T19	Alertas por incumplimiento de parametros	Crear endpoint para obtener las alertas de la empresa	5	Nicolas Vera	Done
US29	Gestionar alertas por incumplimiento de parámetros operativos	T20	Alertas por incumplimiento de parametros	Crear endpoint para actualizar el estado de alerta (saber si esta resuelto)	5	Nicolas Vera	Done
US29	Gestionar alertas por incumplimiento de parámetros operativos	T21	Alertas por incumplimiento de parametros	Mostrar en pantalla las alertas registradas	3	Nicolas Vera	Done
US29	Gestionar alertas por incumplimiento de parámetros operativos	T22	Alertas por incumplimiento de parametros	Crear flujo UI para actualizar estado de alerta	5	Nicolas Vera	Done
US29	Gestionar alertas por incumplimiento de parámetros operativos	T23	Alertas por incumplimiento de parametros	Crear lógica de detección de incumplimiento de parametros por entrega	8	Nicolas Vera	Done

User Story Id	User Story Title	Task Id	Task Title	Task Description	Estimation (Hours)	Assigned To	Status
US21	Visualizar dispositivos IoT registrados en la plataforma	T24	Ver todos los dispositivos IoT en la plataforma	Maquetar en pantalla los dispositivos registrados en web	6	Nicolas Vera	Done
US21	Visualizar dispositivos IoT registrados en la plataforma	T25	Ver todos los dispositivos IoT en la plataforma	Crear endpoint para obtener dispositivos de la empresa	5	Alessandro Bernardo	Done
US21	Visualizar dispositivos IoT registrados en la plataforma	T26	Ver todos los dispositivos IoT en la plataforma	Mostrar en pantalla los dispositivos registrados en mobile	3	Alessandro Bernardo	Done
US18	Asignar dispositivos IoT a vehículos de carga	T27	Asignar dispositivo a vehículo de carga	Crear endpoint para asignar dispositivo a vehículo	6	Alessandro Bernardo	Done
US18	Asignar dispositivos IoT a vehículos de carga	T28	Asignar dispositivo a vehículo de carga	Crear formulario para asignar dispositivo a vehículo	3	Alessandro Bernardo	Done
US33	Gestionar y visualizar el detalle de un viaje	T29	Detalle de viaje	Crear endpoint de viaje por id	5	Jefrey Sanchez	Done
US33	Gestionar y visualizar el detalle de un viaje	T30	Detalle de viaje	Mostrar en pantalla de detalle de viaje	3	Jefrey Sanchez	Done
US33	Gestionar y visualizar el detalle de un viaje	T31	Detalle de viaje	Crear pagina de detalle de viaje en web	3	Marcelo Garro	Done
US33	Gestionar y visualizar el detalle de un viaje	T32	Detalle de viaje	Crear pagina de creación de viajes	3	Marcelo Garro	Done
US33	Gestionar y visualizar el detalle de un viaje	T33	Detalle de viaje	Crear endpoint para crear viajes	5	Marcelo Garro	Done

6.2.2.4. Development Evidence for Sprint Review

Durante el sprint se avanzó en las tres capas del sistema, en Frontend se estableció la estructura base de la interfaz, incorporando componentes reutilizables y su navegación correspondiente; en Backend se configuró la arquitectura del servidor, se definieron endpoints esenciales, se implementó la lógica inicial de manejo de datos y se ajustaron las configuraciones de build e ignores; finalmente, en Embedded Application se desarrollaron los módulos fundamentales para la comunicación con el hardware, se estandarizaron procesos de lectura y transmisión de datos, y se documentaron las configuraciones requeridas para garantizar su integración con el sistema principal.

Web Application

Durante este sprint se implementaron los módulos principales del sistema **CargaSafe** en Angular 20.

El equipo desarrolló el *bounded context* de suscripciones, gestión de flotas, monitoreo de viajes, alertas y autenticación (IAM).

Se integró la API simulada con [json-server](#), se configuró el enrutamiento base y se estructuró la arquitectura por capas (domain, infrastructure, presentation).

Repository	Branch	Commit Id	Commit Message	Commit Message Body	Committed on (Date)
iot-solutions-development-cargasafe-frontend	main	c8dc444	initial commit	Inicio del proyecto Angular 20, configuración base y estructura inicial.	2025-09-23
iot-solutions-development-cargasafe-frontend	main	f8215b4	feat: First commit. Project Setup	Creación del entorno Angular standalone, configuración inicial de módulos y dependencias.	2025-09-23
iot-solutions-development-cargasafe-frontend	develop	ebcd034	feat(shared): implement base entity, API endpoint, assembler, and response interfaces	Implementación de entidades base, endpoints comunes y ensambladores para integración de datos.	2025-10-03
iot-solutions-development-cargasafe-frontend	develop	7026161	feat(shared): implement root layout and routing system	Configuración del layout principal con sidebar y rutas dinámicas.	2025-10-03
iot-solutions-development-cargasafe-frontend	develop	00a9220	build(fake-api): add json-server for API REST connections	Implementación del servidor JSON local para simular endpoints REST del backend.	2025-10-07
iot-solutions-development-cargasafe-frontend	IAM	ffa780a	feat(iam): implement login, register and recovery password pages	Creación de pantallas de autenticación (login, registro, recuperación de contraseña).	2025-10-07

Repository	Branch	Commit Id	Commit Message	Commit Message Body	Committed on (Date)
iot-solutions-development-cargasafe-frontend	trip-management	9d54bb9	feat(trips): add DeliveryOrder, OriginPoint, TripParameter entities and update Trip entity with additional properties	Entidades principales del módulo de viajes (trip, delivery order, parámetros y origen).	2025-10-07
iot-solutions-development-cargasafe-frontend	trip-management	e5e3f55	feat(trips): add trip list page	Página principal de viajes con tabla, filtro y paginación.	2025-10-07
iot-solutions-development-cargasafe-frontend	trip-management	6553425	feat(trips): implement trips feature with API integration and routing	Integración del módulo de viajes con servicios REST simulados.	2025-10-07
iot-solutions-development-cargasafe-frontend	alerts-and-resolutions	f2e09c0	refactor(alert screen): new icons and added db.json	Refactor visual de alertas, íconos actualizados y datos simulados.	2025-10-07
iot-solutions-development-cargasafe-frontend	alerts-and-resolutions	7564492	feat(alerts): add alert screens	Sección de alertas del sistema con componentes modulares.	2025-10-07
iot-solutions-development-cargasafe-frontend	visualization-and-analytics	9ed2471	feat(dashboard): Added charts for trip monitoring information	Dashboard con gráficos de seguimiento de viajes y estadísticas mensuales.	2025-10-07
iot-solutions-development-cargasafe-frontend	fleet-management	81e45b0	feat(fleet): scaffold bounded context folders	Estructura base para el módulo de gestión de flotas.	2025-10-08
iot-solutions-development-cargasafe-frontend	develop	fe46756	feat(env): add base environment files and production config	Configuración de entornos (<code>environment.ts</code>) y build de producción.	2025-10-08
iot-solutions-development-cargasafe-frontend	fleet-management	7d43fd7	feat(fleet-domain): add vehicle and device models	Creación de modelos para vehículos y dispositivos IoT.	2025-10-08
iot-solutions-development-cargasafe-frontend	fleet-management	daf642c	feat(fleet-data): add device and vehicle data-access services	Servicios para gestión de datos de flota y dispositivos conectados.	2025-10-08
iot-solutions-development-cargasafe-frontend	main	a37a0b7	feat(fleet-ui): add device-create-and-edit component	Formulario y modal para creación/edición de dispositivos.	2025-10-08
iot-solutions-development-cargasafe-frontend	fleet-management	47ad853	feat(fleet-ui): add vehicle-create-and-edit component	Componente para registrar o editar vehículos.	2025-10-08
iot-solutions-development-cargasafe-frontend	fleet-management	d1578c7	feat(fleet-pages): add device-management page	Página de administración de dispositivos.	2025-10-08
iot-solutions-development-cargasafe-frontend	fleet-management	e79cef8	feat(fleet-pages): add vehicle-management page	Página de administración de vehículos.	2025-10-08
iot-solutions-development-cargasafe-frontend	develop	ba3f415	feat(routes): add/modify Fleet routes and guards	Configuración de rutas y guardias para navegación segura en Fleet.	2025-10-08
iot-solutions-development-cargasafe-frontend	develop	37cc80c	feat(shared-layout): add nav links to Vehicles and Devices management	Enlaces de navegación desde el layout principal hacia módulos de flota.	2025-10-08
iot-solutions-development-cargasafe-frontend	subscriptions-bounded-context	94e33ba	feat(subscription): update bc content	Ajustes al <i>bounded context</i> de suscripciones (contenidos y estados).	2025-10-08
iot-solutions-development-cargasafe-frontend	develop	25b4a74	fix: delete import zone.js in app.ts	Corrección de import redundante en configuración principal.	2025-10-08

Web Services

Repository	Branch	Commit Id	Commit Message	Commit Message Body	Committed on (Date)
iot-solutions-development-cargasafe-backend	main	48ab0d3	build: setup project template	Configura la plantilla base del proyecto.	2025-09-23
iot-solutions-development-cargasafe-backend	feature/IAM	e71219c	feature(iam): add domain layer	Agrega la capa de dominio para IAM.	2025-10-19
iot-solutions-development-cargasafe-backend	feature/IAM	438efd6	feature(iam): add infrastructure layer	Implementa la infraestructura base del módulo IAM.	2025-10-20
iot-solutions-development-cargasafe-backend	feature/IAM	c07cf20	feature(iam): add applicationlayer	Añade la capa de aplicación para IAM.	2025-10-20
iot-solutions-development-cargasafe-backend	feature/IAM	10f700d	feature(iam): add interface layer	Se agregan las interfaces del módulo IAM.	2025-10-20
iot-solutions-development-cargasafe-backend	feature/merchant-bounded-context	1ce8f15	feat(merchant): create merchant domain layer entities, commands and queries	Crea entidades y CQRS del dominio Merchant.	2025-10-20
iot-solutions-development-cargasafe-backend	feature/merchant-bounded-context	ab7a2f9	feat(merchant): add merchant CQRS pattern and JPA integration	Integra JPA y patrón CQRS para Merchant.	2025-10-20
iot-solutions-development-cargasafe-backend	feature/merchant-bounded-context	0eeab08	feat(merchant): add merchant use cases for application layer	Se agregan casos de uso de Merchant.	2025-10-20
iot-solutions-development-cargasafe-backend	feature/merchant-bounded-context	3829f1f	feat(merchant): create merchant endpoint for retrieve all merchants	Crea endpoint para obtener todos los merchants.	2025-10-20
iot-solutions-development-cargasafe-backend	feature/merchant-bounded-context	4a01b86	feat(merchant): create merchant exceptions and handle HTTP codes	Añade excepciones y manejo de códigos HTTP.	2025-10-21
iot-solutions-development-cargasafe-backend	feature/merchant-bounded-context	2145659	feat(merchant): create employee model, services and controller	Implementa modelo y controlador de empleados.	2025-10-21
iot-solutions-development-cargasafe-backend	develop	23f8b83	Merge branch 'develop' into feature/IAM	Fusiona cambios recientes en feature/IAM.	2025-10-21
iot-solutions-development-cargasafe-backend	develop	dc6d371	Merge pull request #1 from feature/merchant-bounded-context	Integra el trabajo del bounded context Merchant.	2025-10-21
iot-solutions-development-cargasafe-backend	feature/alert-bounded-context	eb3c346	feat(alert): add DDD Alert bounded context	Agrega bounded context Alert bajo DDD.	2025-10-21
iot-solutions-development-cargasafe-backend	feature/alert-bounded-context	66c8c28	add commands and queries	Añade comandos y consultas del módulo Alerts.	2025-10-22
iot-solutions-development-cargasafe-backend	feature/IAM	7d6e144	feature(iam): add application.properties	Agrega configuración en application.properties.	2025-10-22
iot-solutions-development-cargasafe-backend	feature/IAM	2e5ff9e	feature(iam): add new dependencies	Incorpora nuevas dependencias para IAM.	2025-10-22
iot-solutions-development-cargasafe-backend	feature/IAM	94e0c72	feature(iam): add new roles	Añade nuevos roles al sistema.	2025-10-22
iot-solutions-development-cargasafe-backend	feature/IAM	42e4452	feature(iam): add OpenApiConfiguration	Configura OpenAPI para IAM.	2025-10-22

Repository	Branch	Commit Id	Commit Message	Commit Message Body	Committed on (Date)
iot-solutions-development-cargasafe-backend	feature/IAM	4303b67	feature(iam): update BearerTokenService	Actualiza lógica del BearerTokenService.	2025-10-22
iot-solutions-development-cargasafe-backend	feature/IAM	882174e	feature(iam): update TokenServiceImpl	Mejora TokenServiceImpl.	2025-10-22
iot-solutions-development-cargasafe-backend	feature/IAM	8577fe4	feature(iam): update GetUserByldQuery	Optimiza consulta para obtener usuario por ID.	2025-10-22
iot-solutions-development-cargasafe-backend	feature/IAM	1cd3c9f	feature(iam): update SnakeCasePhysicalNamingStrategy	Ajusta estrategia de nombres en snake_case.	2025-10-22
iot-solutions-development-cargasafe-backend	feature/real-time-monitoring-bounded-context	4d71b04	feat(monitored): create branch feature/real-time-monitoring-bounded-context	Inicializa bounded context de monitoreo en tiempo real.	2025-10-24
iot-solutions-development-cargasafe-backend	feature/IAM	69dfee9	feature(iam): add new rol	Agrega un nuevo rol al sistema.	2025-10-24
iot-solutions-development-cargasafe-backend	feature/IAM	80b4299	feature(iam): add TTL exactas 4h/24h	Añade TTL exactas para tokens.	2025-10-24
iot-solutions-development-cargasafe-backend	feature/real-time-monitoring-bounded-context	54233ff	feat(monitored): update command and queries files	Actualiza comandos y consultas de monitoreo.	2025-10-24
iot-solutions-development-cargasafe-backend	feature/real-time-monitoring-bounded-context	fe5db9e	feat(monitored): change application.properties	Ajusta configuración para monitoreo.	2025-10-24
iot-solutions-development-cargasafe-backend	feature/IAM	27b0813	feature(iam): add revoke token	Implementa revocación de tokens.	2025-10-25
iot-solutions-development-cargasafe-backend	feature/alert-bounded-context	45163d7	feat(alerts): add commandservice, queryservices and controllers	Implementa servicios y controladores de Alertas.	2025-10-25
iot-solutions-development-cargasafe-backend	feature/fleet-management-bounded-context	79b77a7	feat(fleet-management): implement complete bounded context with DDD architecture	Implementa todo el bounded context de Fleet con DDD.	2025-10-25
iot-solutions-development-cargasafe-backend	feature/alert-bounded-context	60b30ff	feat(alerts): add controllers	Añade controladores del módulo Alerts.	2025-10-25
iot-solutions-development-cargasafe-backend	feature/alert-bounded-context	96cdbe6	feat(alerts): fix controllers	Corrige errores en controladores de Alertas.	2025-10-26
iot-solutions-development-cargasafe-backend	feature/fleet-management-bounded-context	c9f77df	feature(fleet-management): update application properties for JPA/validation	Ajusta propiedades para JPA y validaciones.	2025-10-26
iot-solutions-development-cargasafe-backend	feature/fleet-management-bounded-context	31d0d04	feature(fleet-management): remove vehiclePlate and use boolean online	Cambia vehiclePlate por flag online.	2025-10-26
iot-solutions-development-cargasafe-backend	feature/fleet-management-bounded-context	14a2ee5	feature(fleet-management): stop mapping vehiclePlate	Elimina mapeo de vehiclePlate.	2025-10-26
iot-solutions-development-cargasafe-backend	feature/fleet-management-bounded-context	bfe87bd	feature(fleet-management): drop vehiclePlate from create payload	Quita vehiclePlate del payload de creación.	2025-10-26
iot-solutions-development-cargasafe-backend	feature/fleet-management-bounded-context	ea1ea52	feature(fleet-management): remove devicemei from create command	Elimina devicemei del comando de creación.	2025-10-26

Repository	Branch	Commit Id	Commit Message	Commit Message Body	Committed on (Date)
iot-solutions-development-cargasafe-backend	feature/fleet-management-bounded-context	8da2268	feature(fleet-management): stop mapping devicelmei	Deja de mapear devicelmei.	2025-10-26
iot-solutions-development-cargasafe-backend	feature/fleet-management-bounded-context	a355878	feature(fleet-management): drop devicelmei from create payload	Retira devicelmei del payload de creación.	2025-10-26
iot-solutions-development-cargasafe-backend	feature/fleet-management-bounded-context	ea3faeb	feature(fleet-management): constructor without vehiclePlate	Se ajusta el constructor para eliminar la dependencia de vehiclePlate.	2025-10-26
iot-solutions-development-cargasafe-backend	feature/fleet-management-bounded-context	ad1b91f	feature(fleet-management): create device without linking	Se permite crear dispositivos sin enlaces iniciales.	2025-10-26
iot-solutions-development-cargasafe-backend	feature/fleet-management-bounded-context	d581220	feature(fleet-management): constructor without devicelmei	Se actualiza el constructor removiendo devicelmei.	2025-10-26
iot-solutions-development-cargasafe-backend	feature/fleet-management-bounded-context	8ef00cd	feature(fleet-management): create vehicle without linking	Se habilita la creación de vehículos sin vinculación inicial.	2025-10-26
iot-solutions-development-cargasafe-backend	feature/alert-bounded-context	02c6a3c	feat(alerts): update alertCommandServiceImpl	Se mejora la implementación del servicio de comandos de alertas.	2025-10-26
iot-solutions-development-cargasafe-backend	develop	4140c50	Merge pull request #4 from Los-Parkers-IoT/feature/IAM	Se integra la rama IAM con los cambios más recientes.	2025-10-28
iot-solutions-development-cargasafe-backend	develop	d567a6f	Merge branch 'develop' into feature/alerts-bounded-context	Se sincroniza la rama de alertas con develop.	2025-10-28
iot-solutions-development-cargasafe-backend	develop	1a5ac25	Merge pull request #5 from Los-Parkers-IoT/feature/alerts-bounded-context	Se fusiona el contexto de alertas con el flujo principal.	2025-10-28
iot-solutions-development-cargasafe-backend	feature/real-time-monitoring-bounded-context	14b46e2	feat (monitoring): add commands and controllers	Se agregan comandos y controladores para el monitoreo.	2025-10-29
iot-solutions-development-cargasafe-backend	feature/real-time-monitoring-bounded-context	d11d34b	feat (monitoring): fix endpoints	Se corrigen los endpoints del módulo de monitoreo.	2025-10-30
iot-solutions-development-cargasafe-backend	feature/trip-management	dcedea1	add trip bc structure	Se crea la estructura inicial del bounded context de trips.	2025-10-31
iot-solutions-development-cargasafe-backend	develop	a863497	Merge branch 'develop' into feature/real-time-monitoring-bounded-context	Se sincroniza develop con el módulo de monitoreo en tiempo real.	2025-10-31
iot-solutions-development-cargasafe-backend	develop	b43b055	Merge pull request #6 from Los-Parkers-IoT/feature/real-time-monitoring-bounded-context	Se incorpora el módulo de monitoreo en tiempo real.	2025-10-31
iot-solutions-development-cargasafe-backend	feature/fleet-management-bounded-context	95ca548	refactor(fleet-management): rename bc from fleet-management to fleet	Se renombra el bounded context para simplificar su identidad.	2025-11-02
iot-solutions-development-cargasafe-backend	feature/fleet-management-bounded-context	9cd9b93	feat(fleet-management): persist multiple device IMEIs in vehicle aggregate	Se habilita la persistencia de múltiples IMEIs por vehículo.	2025-11-02
iot-solutions-development-cargasafe-backend	feature/fleet-management-bounded-context	fa10bb1	fix(fleet-management): sync vehicle device set when assigning or unassigning	Se sincroniza el conjunto de dispositivos del vehículo.	2025-11-02
iot-solutions-development-cargasafe-backend	feature/fleet-management-bounded-context	3d9b679	refactor(fleet-management): expose vehicle-device operations via REST and updated repositories	Se exponen operaciones vehículo-dispositivo por REST.	2025-11-02

Repository	Branch	Commit Id	Commit Message	Commit Message Body	Committed on (Date)
iot-solutions-development-cargasafe-backend	develop	c0c224a	Merge branch 'develop' into feature/fleet-management-bounded-context	Se actualiza la rama fleet con cambios recientes.	2025-11-02
iot-solutions-development-cargasafe-backend	develop	abecf0f	Merge pull request #7 from Los-Parkers-IoT/feature/fleet-management-bounded-context	Se incorpora el módulo de gestión de flota.	2025-11-02
iot-solutions-development-cargasafe-backend	develop	1b329b9	feat(develop): fix application properties	Se actualizan propiedades para mejorar la configuración.	2025-11-02
iot-solutions-development-cargasafe-backend	feature/fleet-management-bounded-context	d137d02	feat(fleet-management): add command to update vehicle status	Se agrega comando para actualizar estado del vehículo.	2025-11-02
iot-solutions-development-cargasafe-backend	feature/fleet-management-bounded-context	106c452	feat(fleet-management): handle vehicle status update command	Se gestiona el comando de actualización de estado.	2025-11-02
iot-solutions-development-cargasafe-backend	feature/fleet-management-bounded-context	bdeb72e	feat(fleet-management): expose PATCH /vehicles/{id}/status endpoint	Se expone endpoint para actualizar estado del vehículo.	2025-11-02
iot-solutions-development-cargasafe-backend	develop	aed1d18	fix(develop): update application properties	Se mejora la configuración de la aplicación.	2025-11-02
iot-solutions-development-cargasafe-backend	feature/fleet-management-bounded-context	392f2de	refactor(fleet-management): remove type from device aggregate	Se elimina el atributo type del agregado device.	2025-11-02
iot-solutions-development-cargasafe-backend	feature/fleet-management-bounded-context	120024e	refactor(fleet-management): align command/query services after removing type	Se alinean servicios tras eliminar type.	2025-11-02
iot-solutions-development-cargasafe-backend	feature/fleet-management-bounded-context	6414606	refactor(fleet-management): update REST resources and assemblers to remove type	Se actualizan recursos REST eliminando type.	2025-11-02
iot-solutions-development-cargasafe-backend	feature/fleet-management-bounded-context	f9c97b1	feat(fleet-management): expose PATCH /vehicles/{id}/status endpoint	Se publica endpoint para estado del vehículo.	2025-11-02
iot-solutions-development-cargasafe-backend	feature/fleet-management-bounded-context	c725c06	feat(fleet-management): add command to update device online status	Se agrega comando para actualizar online status.	2025-11-02
iot-solutions-development-cargasafe-backend	feature/fleet-management-bounded-context	ecc2338	feat(fleet-management): handle device online status update in command service	Se gestiona actualización de estado online.	2025-11-02
iot-solutions-development-cargasafe-backend	feature/fleet-management-bounded-context	f46d243	feat(fleet-management): expose PATCH /devices/{id}/online endpoint	Se expone endpoint para estado online de dispositivos.	2025-11-02
iot-solutions-development-cargasafe-backend	develop	7b4308b	Merge branch 'develop' into feature/fleet-management-bounded-context	Se sincroniza develop con fleet.	2025-11-02
iot-solutions-development-cargasafe-backend	develop	ae4e29a	Merge pull request #8 from Los-Parkers-IoT/feature/fleet-management-bounded-context	Se integra la rama fleet al flujo principal.	2025-11-02
iot-solutions-development-cargasafe-backend	feature/alert-bounded-context	6561959	feat(alert): update AlertCommandServiceImpl	Se optimiza el servicio de comandos de alertas.	2025-11-03
iot-solutions-development-cargasafe-backend	develop	01900cb	Merge pull request #9 from Los-Parkers-IoT/develop	Se integra la rama develop.	2025-11-03
iot-solutions-development-cargasafe-backend	feature/trip-management	f812a68	update bc trip	Se actualiza el bounded context de trips.	2025-11-04

Repository	Branch	Commit Id	Commit Message	Commit Message Body	Committed on (Date)
iot-solutions-development-cargasafe-backend	feature/alert-bounded-context	27cf873	feat (alert): web config	Se agrega configuración web para alertas.	2025-11-04
iot-solutions-development-cargasafe-backend	develop	87a22af	Merge pull request #10 from Los-Parkers-IoT/feature/alerts-bounded-context	Se fusiona módulo de alertas.	2025-11-04
iot-solutions-development-cargasafe-backend	feature/trip-management	463b774	feat(trip): update application.properties	Se actualizan propiedades del módulo trip.	2025-11-04
iot-solutions-development-cargasafe-backend	develop	9bfe0e0	Merge pull request #11 from Los-Parkers-IoT/feature/trip-management	Se integra módulo trip.	2025-11-04
iot-solutions-development-cargasafe-backend	feature/trip-management	a313f93	Update TripCommandService.java	Se actualiza el servicio de comandos de trip.	2025-11-04
iot-solutions-development-cargasafe-backend	feature/trip-management	9e5904f	updated bc trips	Se actualiza el bounded context de trips.	2025-11-06
iot-solutions-development-cargasafe-backend	develop	7874193	Merge branch 'develop' into feature/trip-management	Se sincroniza develop con trip.	2025-11-06
iot-solutions-development-cargasafe-backend	feature/fleet-management-bounded-context	2a23428	feature(fleet) add dev mode without jtw token	Se agrega modo dev sin JWT.	2025-11-07
iot-solutions-development-cargasafe-backend	develop	940b6f7	Merge branch 'develop' into feature/trip-management	Se sincronizan cambios de develop a trip.	2025-11-09
iot-solutions-development-cargasafe-backend	feature/trip-management	da35f23	refactor(trips): change data types for attribute in Trip model	Se ajustan tipos de datos del modelo Trip.	2025-11-09
iot-solutions-development-cargasafe-backend	feature/trip-management	501e071	feat(trips): add origin points into trips management context	Se agregan puntos de origen al contexto.	2025-11-09
iot-solutions-development-cargasafe-backend	feature/trip-management	a0faed5	feat(trips): add create trip logic with delivery orders embedded	Se añade lógica de creación con órdenes embebidas.	2025-11-09
iot-solutions-development-cargasafe-backend	develop	56bd396	build: add dockerfile for container deploy	Se agrega Dockerfile para despliegue.	2025-11-10
iot-solutions-development-cargasafe-backend	develop	80ac78d	build: expose application for all origins	Se habilita acceso desde cualquier origen.	2025-11-10
iot-solutions-development-cargasafe-backend	feature/trip-management	d959019	feat(trip): add start trip logic	Se implementa lógica para iniciar viajes.	2025-11-10
iot-solutions-development-cargasafe-backend	feature/IAM	20ce2bd	feat(security): disable all security filters temporarily for local testing	Se deshabilitan filtros de seguridad para pruebas locales.	2025-11-11
iot-solutions-development-cargasafe-backend	feature/IAM	23c321f	chore(security): comment WebSecurityConfiguration to prevent JWT filter execution	Se comenta la configuración de seguridad temporalmente.	2025-11-11
iot-solutions-development-cargasafe-backend	feature/IAM	868ad6c	chore(security): align DevFleetSecurityConfig with temporary disable mode	Se ajusta configuración DevFleet.	2025-11-11
iot-solutions-development-cargasafe-backend	feature/IAM	cedbb18	chore(swagger): remove bearerAuth scheme from OpenAPI to disable Swagger authorize button	Se desactiva autenticación en Swagger.	2025-11-11

Repository	Branch	Commit Id	Commit Message	Commit Message Body	Committed on (Date)
iot-solutions-development-cargasafe-backend	feature/IAM	c101ed4	chore(config): update application properties for unrestricted local access	Se otorga acceso libre en entorno local.	2025-11-11
iot-solutions-development-cargasafe-backend	develop	f6b4b4f	merge: feature/fleet-management-bounded-context into develop	Se integra fleet a develop.	2025-11-11
iot-solutions-development-cargasafe-backend	feature/IAM	35784ce	fix(cors): use allowedOriginPatterns with credentials	Se corrige configuración CORS.	2025-11-11
iot-solutions-development-cargasafe-backend	main	28a2c39	merge: develop into main for release	Se realiza merge para preparación de release.	2025-11-11
iot-solutions-development-cargasafe-backend	main	b354ae7	fix(db): update local postgres password	Se actualiza contraseña local.	2025-11-11
iot-solutions-development-cargasafe-backend	main	0bb7aaf	fix(db): update local postgres password	Se corrige nuevamente configuración de credenciales.	2025-11-11
iot-solutions-development-cargasafe-backend	develop	9262dfc	Merge branch 'develop' into feature/trip-management	Se sincronizan ramas.	2025-11-11
iot-solutions-development-cargasafe-backend	feature/trip-management	1d05ef0	feat(trip): create delivery order process and model	Se crea proceso y modelo de órdenes.	2025-11-11
iot-solutions-development-cargasafe-backend	feature/trip-management	18876a3	feat(trip): add origin point into trip aggregate	Se agregó el punto de origen al agregado de viaje.	2025-11-11
iot-solutions-development-cargasafe-backend	feature/trip-management	fdad090	feat(trip-monitoring): create events for trip and monitoring BC	Se creó el conjunto de eventos para viaje y monitoreo.	2025-11-11
iot-solutions-development-cargasafe-backend	feature/trip-management	7784c63	chore(validation): add trip controller resource validations	Se añadieron validaciones al controlador de viajes.	2025-11-11
iot-solutions-development-cargasafe-backend	feature/trip-management	f392148	chore(resources): add delivery order thresholds in resources	Se añadieron límites del delivery order en recursos.	2025-11-11
iot-solutions-development-cargasafe-backend	feature/alert-bounded-context	7c7856c	feat(alert): add AlertContextFacade	Se creó la fachada para el contexto de alertas.	2025-11-12
iot-solutions-development-cargasafe-backend	feature/trip-management	1eac035	feat(trip): add trip validations for order in thresholds	Se agregaron validaciones de orden en thresholds.	2025-11-12
iot-solutions-development-cargasafe-backend	develop	1fb1218	Merge branch 'feature/trip-management' into feature/alerts-bc	Se integraron cambios de trip-management.	2025-11-12
iot-solutions-development-cargasafe-backend	feature/alert-bounded-context	2fc6292	chore(alert): generate alerts when telemetry data was inserted	Se generaron alertas tras insertar telemetría.	2025-11-12
iot-solutions-development-cargasafe-backend	feature/trip-management	52ff53b	chore(trip): update resources	Se actualizaron los recursos del módulo de viajes.	2025-11-13
iot-solutions-development-cargasafe-backend	feature/alert-bounded-context	c2d0d4d	feat(alert): add deliveryOrderId	Se agregó deliveryOrderId al modelo de alerta.	2025-11-13
iot-solutions-development-cargasafe-backend	main	f4aadeb	Merge pull request #12 from Los-Parkers-IoT/develop	Se fusionaron los cambios desde develop.	2025-11-14

Mobile Application

Repository	Branch	Commit Id	Commit Message	Commit Message Body	Committed on (Date)
iot-solutions-development-cargasafe-mobile	main	0915028	build: setup mobile app project	Configuración inicial del proyecto móvil.	2025-09-24
iot-solutions-development-cargasafe-mobile	main	7f08de6	chore(base): setup carga safe style theme	Se define el tema base de estilos para la app.	2025-10-21
iot-solutions-development-cargasafe-mobile	feature/alerts	af69eaf	feat(alert): add branch alert bounded context	Se agrega el bounded context de alertas.	2025-10-29
iot-solutions-development-cargasafe-mobile	feature/visualization-and-analytics	6bd38f3	feat(visualization and analytics): integrate alerts navigation, implement mock data system, and improve dashboard UI	Integración de navegación, mocks y mejoras del dashboard.	2025-11-05
iot-solutions-development-cargasafe-mobile	feature/fleet	735f522	feat(shared-core): add AppConfig and Result helper	Se incorporan helpers base para configuración y resultados.	2025-11-12
iot-solutions-development-cargasafe-mobile	feature/fleet	215657b	feat(fleet-domain): add device and vehicle entities	Entidades de dispositivo y vehículo agregadas.	2025-11-12
iot-solutions-development-cargasafe-mobile	feature/fleet	4ab1754	feat(fleet-domain): add vehicle enums (status/type) and parsers	Se agregan enums y parsers para vehículos.	2025-11-12
iot-solutions-development-cargasafe-mobile	feature/fleet	9caf5f9	feat(fleet-domain): define repositories contracts	Definición de contratos para repositorios del fleet.	2025-11-12
iot-solutions-development-cargasafe-mobile	feature/fleet	7720b25	feat(fleet-usecases/devices): CRUD + queries	Implementación de CRUD y consultas para dispositivos.	2025-11-12
iot-solutions-development-cargasafe-mobile	feature/fleet	2125483	feat(fleet-usecases/devices): add firmware & online updates	Se agregan flujos de actualización online y firmware.	2025-11-12
iot-solutions-development-cargasafe-mobile	feature/fleet	68e0137	feat(fleet-usecases/vehicles): CRUD + queries	CRUD y consultas del módulo de vehículos añadidos.	2025-11-12
iot-solutions-development-cargasafe-mobile	feature/fleet	8c2c56a	feat(fleet-usecases/vehicles): status update + assign/unassign device	Lógica para estados y asignación de dispositivos.	2025-11-12
iot-solutions-development-cargasafe-mobile	feature/fleet	e55398e	feat(fleet-infra/dto): introduce DeviceDto & VehicleDto	DTOs de dispositivos y vehículos añadidos.	2025-11-12
iot-solutions-development-cargasafe-mobile	feature/fleet	2675a09	feat(fleet-infra/mappers): add device & vehicle mappers	Se agregan mapeadores para entidades del fleet.	2025-11-12
iot-solutions-development-cargasafe-mobile	feature/fleet	b806018	feat(fleet-infra/datasource): remote datasource for devices	Datasource remoto de dispositivos implementado.	2025-11-12
iot-solutions-development-cargasafe-mobile	feature/fleet	4291ecf	feat(fleet-infra/datasource): remote datasource for vehicles	Datasource remoto de vehículos implementado.	2025-11-12
iot-solutions-development-cargasafe-mobile	feature/fleet	e7aa0ea	feat(fleet-infra/repository): implement DeviceRepository	Repository de dispositivos implementado.	2025-11-12
iot-solutions-development-cargasafe-mobile	feature/fleet	e7aa0ea	feat(fleet-infra/repository): implement VehicleRepository	Repository de vehículos implementado.	2025-11-12

Repository	Branch	Commit Id	Commit Message	Commit Message Body	Committed on (Date)
iot-solutions-development-cargasafe-mobile	feature/fleet	c33f4c5	feat(fleet-infra/service): add endpoints and Fleet ApiService	Servicios y endpoints del fleet añadidos.	2025-11-12
iot-solutions-development-cargasafe-mobile	feature/fleet	1695804	feat(fleet-presentation/state): add FleetProvider with all flows	Proveedor del fleet creado con todos sus flujos.	2025-11-12
iot-solutions-development-cargasafe-mobile	feature/fleet	ed06ede	feat(fleet-presentation/styles): add FleetStyles	Estilos del fleet agregados.	2025-11-12
iot-solutions-development-cargasafe-mobile	feature/fleet	88ee330	feat(fleet-presentation/widgets): filters toolbar & KPI card	Se agregan widgets de filtros y tarjeta KPI.	2025-11-12
iot-solutions-development-cargasafe-mobile	feature/fleet	f303475	feat(fleet-presentation/dialogs): device create/edit dialog	Diálogo para creación y edición de dispositivos.	2025-11-12
iot-solutions-development-cargasafe-mobile	feature/fleet	d6e2208	feat(fleet-presentation/dialogs): vehicle create/edit dialog	Diálogo para creación y edición de vehículos.	2025-11-12
iot-solutions-development-cargasafe-mobile	feature/fleet	e3704b3	feat(fleet-presentation/pages): device list & detail	Páginas de listado y detalle de dispositivos agregadas.	2025-11-12
iot-solutions-development-cargasafe-mobile	feature/fleet	9f62e33	feat(fleet-presentation/pages): vehicle list & detail	Páginas de listado y detalle de vehículos creadas.	2025-11-12
iot-solutions-development-cargasafe-mobile	feature/fleet	37cd851	feat(fleet-routing): register fleet routes	Se registran rutas del módulo fleet.	2025-11-12
iot-solutions-development-cargasafe-mobile	feature/fleet	95f319b	feat(app): wire Fleet feature (routes, provider & DI factory)	Integración del fleet en la aplicación principal.	2025-11-12
iot-solutions-development-cargasafe-mobile	feature/fleet	cd655a9	chore(git): ignore platform ephemeral & Flutter generated	Nuevas reglas de ignore para archivos generados.	2025-11-12
iot-solutions-development-cargasafe-mobile	feature/fleet	c3776a9	feat(fleet): add Fleet routes and named paths for vehicles/devices	Nuevas rutas nombradas para vehículos y dispositivos.	2025-11-12
iot-solutions-development-cargasafe-mobile	feature/fleet	7877b1f	feat(app): navigate to Fleet sections from Drawer using go_router	Navegación a fleet desde el Drawer implementada.	2025-11-12
iot-solutions-development-cargasafe-mobile	feature/fleet	f9ddb20	chore(config): set default API_BASE_URL and timeouts	Configuración base de APIs y timeouts definida.	2025-11-12
iot-solutions-development-cargasafe-mobile	feature/fleet	ccce3e0	fix(ui): use GoRouterState to read current route in drawer	Corrección de lectura de rutas con GoRouterState.	2025-11-12
iot-solutions-development-cargasafe-mobile	feature/fleet	8d1642b	refactor(ui): migrate Color.withOpacity to withValues alpha API	Migración a nueva API de opacidad.	2025-11-12
iot-solutions-development-cargasafe-mobile	feature/fleet	0f48876	feat(fleet): refactor router	Se refactoriza el router del fleet.	2025-11-12
iot-solutions-development-cargasafe-mobile	feature/fleet	092b7f1	fix(providers): resolve DashboardProvider not found error and add 404 page	Corrección de provider ausente y página 404 añadida.	2025-11-13
iot-solutions-development-cargasafe-mobile	feature/fleet	2a3df0b	refactor(fleet): update app_router.dart	Actualización del router de la app.	2025-11-13

Repository	Branch	Commit Id	Commit Message	Commit Message Body	Committed on (Date)
iot-solutions-development-cargasafe-mobile	feature/fleet	3e26e48	Merge pull request #2 from Los-Parkers-IoT/feature/fleet	Se mergea la rama fleet.	2025-11-13
iot-solutions-development-cargasafe-mobile	develop	701abb0	feat(alerts): merge alert-bounded-context	Se integran los cambios de alertas.	2025-11-13
iot-solutions-development-cargasafe-mobile	feature/alert	47dae06	refactor(shared): update shared carpet	Se actualiza estructura de carpetas compartidas.	2025-11-13
iot-solutions-development-cargasafe-mobile	feature/fleet	58eb0b5	feat(trip): add trip domain model	Se añade modelo de dominio de viajes.	2025-11-14
iot-solutions-development-cargasafe-mobile	feature/trip-management	a31d5c6	feat(trip): add infrastructure layer	Capa de infraestructura del trip añadida.	2025-11-14
iot-solutions-development-cargasafe-mobile	feature/trip-management	4d12ad2	feat(trip): add application layer	Capa de aplicación del trip agregada.	2025-11-14
iot-solutions-development-cargasafe-mobile	feature/trip-management	29da689	fix: add app folder for init project	Se agrega carpeta faltante para iniciar el proyecto.	2025-11-14
iot-solutions-development-cargasafe-mobile	develop	8834a15	refactor: update appbar	Barra superior refactorizada.	2025-11-14
iot-solutions-development-cargasafe-mobile	develop	27cef19	build: update pubspec.lock	Se actualiza el archivo de dependencias.	2025-11-14
iot-solutions-development-cargasafe-mobile	develop	ef7acf9	fix(fleet): make filters toolbar responsive and scrollable	Toolbar de filtros ahora es responsive y desplazable.	2025-11-14
iot-solutions-development-cargasafe-mobile	feature/fleet	7db28db	feat(fleet): enhance vehicle management and IoT device assignments	Mejoras en gestión de vehículos y asignaciones IoT.	2025-11-14
iot-solutions-development-cargasafe-mobile	feature/fleet	b239df2	feat(fleet): improve device management and device detail actions	Mejoras en acciones del detalle y gestión de dispositivos.	2025-11-14
iot-solutions-development-cargasafe-mobile	feature/fleet	3834695	fix(fleet): refresh device list in management page	Se refresca correctamente la lista de dispositivos.	2025-11-14
iot-solutions-development-cargasafe-mobile	feature/fleet	a854f1b	merge(origin/develop): sync develop changes into feature/fleet	Se sincronizan cambios desde develop.	2025-11-14
iot-solutions-development-cargasafe-mobile	develop	b9288b2	Merge branch 'feature/fleet' into develop	Fleet mergeado en develop.	2025-11-14

Edge Application

Repository	Branch	Commit Id	Commit Message	Commit Message Body	Committed on (Date)
iot-solutions-development-cargasafe-edge-application	develop	94257f7	chore: initial setup	Se configura la estructura inicial del proyecto.	2025-11-08
iot-solutions-development-cargasafe-edge-application	develop	646e465	chore(monitored): create monitoring bounded context functionalities and bounded context	Se crea el bounded context de monitoreo con sus funciones básicas.	2025-11-08
iot-solutions-development-cargasafe-edge-application	develop	fca0d86	build(unicorn): add unicorn for project deployment	Se integra Unicorn para el despliegue en producción.	2025-11-08

Repository	Branch	Commit Id	Commit Message	Commit Message Body	Committed on (Date)
iot-solutions-development-cargasafe-edge-application	develop	f52255a	fix(resource): add telemetry id to telemetry resource	Se añade el ID de telemetría al recurso correspondiente.	2025-11-08
iot-solutions-development-cargasafe-edge-application	develop	a9a099e	chore: send telemetry data to cloud backend system	Se habilita el envío de telemetría hacia el backend en la nube.	2025-11-12
Embedded Application					
Repository	Branch	Commit Id	Commit Message	Commit Message Body	Committed on
iot-solutions-development-cargasafe-embedded-application	main	84e697f	chore: initial commit	Inicio del proyecto y configuración base del entorno embebido.	2025-11-08
iot-solutions-development-cargasafe-embedded-application	main	4f309c1	feat(embeddedApp): updated sketch_nov8a.ino	Se agregaron comentarios adicionales dentro del sketch principal.	2025-11-09
iot-solutions-development-cargasafe-embedded-application	main	18b5bcf	feat(embeddedApp): update DataController.h	Se añadieron comentarios aclaratorios dentro del DataController.	2025-11-09
iot-solutions-development-cargasafe-embedded-application	main	b7c68e4	feat(embeddedApp): add Edge connection	Se agregaron comentarios relacionados a la conexión con el Edge.	2025-11-09
iot-solutions-development-cargasafe-embedded-application	main	70c6963	feat(embeddedApp): updated sketch_nov8a.ino	Nuevos comentarios agregados dentro del sketch.	2025-11-11
iot-solutions-development-cargasafe-embedded-application	main	fb7048b	feat(embeddedApp): update CommandHandler	Se agregaron comentarios para mejorar la claridad del CommandHandler.	2025-11-11
iot-solutions-development-cargasafe-embedded-application	main	9b3c557	feat(embeddedApp): update Controller	Se añadieron comentarios dentro del Controller para documentación.	2025-11-11
iot-solutions-development-cargasafe-embedded-application	main	f1d508f	feat(embeddedApp): update DHTSensor.cpp	Se agregaron comentarios aclaratorios dentro del DHTSensor.	2025-11-11
iot-solutions-development-cargasafe-embedded-application	main	6d78c04	feat(embeddedApp): update DHTSensor.h	Se añadieron comentarios dentro del archivo de cabecera del DHTSensor.	2025-11-11
iot-solutions-development-cargasafe-embedded-application	main	8bc6694	feat(embeddedApp): update DataController	Se agregaron nuevos comentarios dentro del DataController.	2025-11-11
iot-solutions-development-cargasafe-embedded-application	main	6d03981	feat(embeddedApp): update EdgeCommunicatorHttp	Se añadieron comentarios para documentar el EdgeCommunicatorHttp.	2025-11-11
iot-solutions-development-cargasafe-embedded-application	main	2d96ae1	feat(embeddedApp): update Event	Se agregaron comentarios dentro del componente Event.	2025-11-11
iot-solutions-development-cargasafe-embedded-application	main	cfc3d5	feat(embeddedApp): update GPSSensor.cpp	Se añadieron comentarios dentro del GPSSensor.	2025-11-11
iot-solutions-development-cargasafe-embedded-application	main	bd0f87c	feat(embeddedApp): update GPSSensor.h	Se agregaron comentarios dentro del archivo de cabecera del GPSSensor.	2025-11-11
iot-solutions-development-cargasafe-embedded-application	main	c074e1b	feat(embeddedApp): update ModestIoT.h	Comentarios agregados dentro de la librería ModestIoT.	2025-11-11
iot-solutions-development-cargasafe-embedded-application	main	1057124	feat(embeddedApp): update sketch_nov8a.ino	Comentarios adicionales incluidos en el sketch principal.	2025-11-11
iot-solutions-development-cargasafe-embedded-application	main	089b8e3	feat(embeddedApp): update sketch_nov8a.ino	Ampliación de comentarios internos dentro del sketch.	2025-11-12
iot-solutions-development-cargasafe-embedded-application	main	7436035	refactor(embeddedApp): update DHTSensor.h	Inclusión del actuador LED dentro del DHTSensor.	2025-11-13
iot-solutions-development-cargasafe-embedded-application	main	c9c671a	refactor(embeddedApp): update DHTSensor.cpp	Implementación del actuador LED dentro del DHTSensor.	2025-11-13
iot-solutions-development-cargasafe-embedded-application	main	38fa844	refactor(embeddedApp): update sketch_nov8a.ino	Se añadió control del actuador LED dentro del sketch.	2025-11-13
iot-solutions-development-cargasafe-embedded-application	main	18b472e	refactor(embeddedApp): update DHTSensor.h	Actualización de lógica del actuador LED dentro del DHTSensor.	2025-11-13
iot-solutions-development-cargasafe-embedded-application	main	62a5573	refactor(embeddedApp): update DHTSensor.cpp	Ajustes finales del actuador LED en el DHTSensor.	2025-11-13

Repository	Branch	Commit Id	Commit Message	Commit Message Body	Committed on
iot-solutions-development-cargasafe-embedded-application	main	815cae4	refactor(embeddedApp): update sketch_nov8a.ino	Actualización del control del actuador LED dentro del sketch.	2025-11-13
iot-solutions-development-cargasafe-embedded-application	main	6fb58d7	fix(embeddedApp): update sketch_nov8a.ino	Corrección final del funcionamiento del actuador LED en el sketch.	2025-11-14

6.2.2.5. Testing Suite Evidence for Sprint Review.

En esta sección se presenta el conjunto de pruebas automatizadas implementadas para el proyecto CargaSafe durante el Sprint 2. La suite de testing incluye Unit Tests, Integration Tests y Acceptance Tests (BDD) diseñados para verificar el correcto funcionamiento de los servicios web relacionados con el monitoreo en tiempo real de condiciones de carga durante el transporte.

Alcance del Testing

El alcance de esta suite de pruebas cubre los siguientes componentes de la aplicación:

Componentes Testeados:

- **Domain Layer:** MonitoringSession (Aggregate Root), TelemetryData (Entity), Commands, Queries y Domain Events
- **Application Layer:** MonitoringSessionCommandServiceImpl, TelemetryCommandServiceImpl, MonitoringSessionQueryServiceImpl, TelemetryQueryServiceImpl
- **Infrastructure Layer:** IMonitoringSessionRepository, ITelemetryDataRepository
- **Interface Layer:** MonitoringController (REST API), TelemetryController (REST API)

Unit Tests

Los Unit Tests se enfocan en probar la lógica de negocio de manera aislada, sin dependencias externas.

MonitoringSessionTest - [src/test/java/Proyect/IoTParkers/monitoring/unit/MonitoringSessionTest.java](#)

Comportamientos probados (12 tests):

- Creación de sesiones de monitoreo con estado ACTIVE
- Transiciones de estado: pausar, reanudar y completar sesiones
- Validación de reglas de negocio (no pausar sesión inactiva, no completar sesión ya completada)
- Gestión de datos de telemetría asociados a la sesión

TelemetryDataTest - [src/test/java/Proyect/IoTParkers/monitoring/unit/TelemetryDataTest.java](#)

Comportamientos probados (7 tests):

- Creación de datos de telemetría con parámetros de temperatura, humedad, vibración y GPS
- Validación de valores extremos de temperatura (-40°C a 60°C)
- Validación de rangos de humedad (0% a 100%)
- Validación de niveles de vibración (0.0 a 5.0)
- Verificación de coordenadas GPS válidas para Lima, Perú

```

PS C:\projects\CargaSafe\carga-safe-backend\iot-solutions-development-cargasafe-backend> mvn test -Dtest="Proyect.IoTParkers.monitoring.un
it.*Test"
[INFO] Scanning for projects...
[INFO] Building IoTParkers 0.0.1-SNAPSHOT
[INFO]   from pom.xml
[INFO]   [ jar ]
[INFO] --- resources:3.3.1:resources (default-resources) @ IoTParkers ---
[INFO] Copying 1 resource from src\main\resources to target\classes
[INFO] Copying 0 resource from src\main\resources to target\classes
[INFO] --- compiler:3.14.0:compile (default-compile) @ IoTParkers ---
[INFO] Recompiling the module because of changed source code.
[INFO] Compiling 320 source files with javac [debug parameters release 21] to target\classes
[INFO] --- resources:3.3.1:testResources (default-testResources) @ IoTParkers ---
[INFO] Copying 4 resources from src\test\resources to target\test-classes
[INFO] --- compiler:3.14.0:testCompile (default-testCompile) @ IoTParkers ---
[INFO] Recompiling the module because of changed dependency.
[INFO] Compiling 9 source files with javac [debug parameters release 21] to target\test-classes
[INFO] --- surefire:3.5.4:test (default-test) @ IoTParkers ---
[INFO] Using auto detected provider org.apache.maven.surefire.junitplatform.JUnitPlatformProvider
[INFO]
[INFO] T E S T S
[INFO] Running MonitoringSession - Unit Tests
[INFO] Tests run: 12, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.081 s -- in MonitoringSession - Unit Tests
[INFO] Running TelemetryData - Unit Tests
[INFO] Tests run: 7, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.010 s -- in TelemetryData - Unit Tests
[INFO] Results:
[INFO]
[INFO] Tests run: 19, Failures: 0, Errors: 0, Skipped: 0
[INFO]
[INFO] BUILD SUCCESS
[INFO] Total time: 4.959 s
[INFO] Finished at: 2025-11-14T23:58:50+05:00
[INFO] Java: Ready

```

andando ahceidno los test

Integration Tests

Los Integration Tests verifican el funcionamiento completo del sistema incluyendo persistencia, servicios y repositorios.

MonitoringSessionServiceIntegrationTest (7 tests):

- Inicio de sesión a través del command service con persistencia en base de datos
- Creación de múltiples sesiones con IDs únicos
- Búsqueda de sesiones por ID de dispositivo
- Actualización de estado de sesión
- Ciclo completo de vida: start → pause → resume → complete

TelemetryDataServiceIntegrationTest (8 tests):

- Agregación de telemetría a sesión activa
- Recuperación de telemetría por sesión
- Manejo de valores extremos de temperatura y vibración
- Rastreo de coordenadas GPS precisas
- Validación de orden cronológico de datos

```

PS C:\projects\CargaSafe\carga-safe-backend\iot-solutions-development-cargasafe-backend> mvn test -Dtest="IntegrationTest"
[INFO] Scanning for projects...
[INFO] Building IoTParkers 0.0.1-SNAPSHOT
[INFO]   from pom.xml
[INFO]   [ jar ]
[INFO] --- resources:3.3.1:resources (default-resources) @ IoTParkers ---
[INFO] Copying 1 resource from src\main\resources to target\classes
[INFO] Copying 0 resource from src\main\resources to target\classes
[INFO] --- compiler:3.14.0:compile (default-compile) @ IoTParkers ---
[INFO] Recompiling the module because of changed source code.
[INFO] Compiling 320 source files with javac [debug parameters release 21] to target\classes
[INFO] --- resources:3.3.1:testResources (default-testResources) @ IoTParkers ---
[INFO] Copying 4 resources from src\test\resources to target\test-classes
[INFO] --- compiler:3.14.0:testCompile (default-testCompile) @ IoTParkers ---
[INFO] Recompiling the module because of changed dependency.
[INFO] Compiling 9 source files with javac [debug parameters release 21] to target\test-classes
[INFO] --- surefire:3.5.4:test (default-test) @ IoTParkers ---
[INFO] Using auto detected provider org.apache.maven.surefire.junitplatform.JUnitPlatformProvider
[INFO]
[INFO] T E S T S
[INFO] --- TelemetryDataServiceIntegrationTest (default-test) @ IoTParkers ---
[INFO] Should maintain telemetry data order by creation time
[INFO] Tests run: 8, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.023 s -- in TelemetryDataServiceIntegrationTest
[INFO] --- IoTParkersApplicationTests (default-test) @ IoTParkers ---
[INFO] Tests run: 7, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.023 s -- in IoTParkersApplicationTests
[INFO] Results:
[INFO]
[INFO] Tests run: 15, Failures: 0, Errors: 0, Skipped: 0
[INFO]
[INFO] BUILD SUCCESS
[INFO] Total time: 4.959 s
[INFO] Finished at: 2025-11-14T23:58:50+05:00
[INFO] Java: Ready

```

Acceptance Tests (BDD)

Los Acceptance Tests utilizan Cucumber con Gherkin para definir escenarios en lenguaje natural.

Feature Files implementados:**1. monitoring-session-management.feature (8 escenarios)**

- Relacionado con User Story: "Como Gestor de Transporte, quiero iniciar sesiones de monitoreo para rastrear condiciones de carga en tiempo real"
- Escenarios clave:
 - Iniciar nueva sesión de monitoreo exitosamente
 - Pausar, reanudar y completar sesiones
 - Validar transiciones de estado
 - Ciclo completo de vida de sesión

2. telemetry-data-capture.feature (10 escenarios)

- Relacionado con User Story: "Como Sistema IoT, quiero capturar datos de telemetría para proporcionar información en tiempo real"
- Escenarios clave:
 - Capturar datos con todos los parámetros (temperatura, humedad, vibración, GPS)
 - Manejar temperaturas extremas (-40°C y 60°C)
 - Detectar vibración elevada (5.0)
 - Capturar múltiples lecturas en secuencia
 - Rechazar datos en sesión inactiva

3. real-time-monitoring.feature (11 escenarios)

- Relacionado con User Story: "Como Gestor de Transporte, quiero monitorear en tiempo real para tomar acciones inmediatas"
- Escenarios clave:
 - Visualizar estado actual de temperatura y humedad
 - Detectar valores fuera de rango
 - Monitorear múltiples vehículos simultáneamente
 - Recibir alertas de vibración excesiva
 - Rastrear ubicación GPS en tiempo real

Repository y Commits de Testing**Repository: iot-solutions-development-cargasafe-backend**

Repository	Branch	Commit Id	Commit Message	Commit Message Body	Committed on (Date)
iot-solutions-development-cargasafe-backend	feature/monitoring-testing	e4aaef65	test: add unit tests for MonitoringSession	Implemented 12 unit tests covering session lifecycle, state transitions, and telemetry management	14/11/2025
iot-solutions-development-cargasafe-backend	feature/monitoring-testing	7c2bd91	test: add unit tests for TelemetryData	Implemented 7 unit tests for TelemetryData entity covering temperature, humidity, vibration, and GPS validation	14/11/2025
iot-solutions-development-cargasafe-backend	feature/monitoring-testing	3f8e4a2	test: add integration tests for monitoring services	Implemented 15 integration tests covering services with database persistence	14/11/2025
iot-solutions-development-cargasafe-backend	feature/monitoring-testing	9d1c7b5	test: add BDD scenarios for session management	Created 8 Gherkin scenarios for monitoring session lifecycle	14/11/2025
iot-solutions-development-cargasafe-backend	feature/monitoring-testing	2a6f3e8	test: add BDD scenarios for telemetry capture	Created 10 Gherkin scenarios for telemetry data capture and validation	14/11/2025
iot-solutions-development-cargasafe-backend	feature/monitoring-testing	5b9d2c1	test: add BDD scenarios for real-time monitoring	Created 11 Gherkin scenarios for real-time monitoring and alerts	14/11/2025
iot-solutions-development-cargasafe-backend	feature/monitoring-testing	8e4a7f3	test: implement Cucumber steps for monitoring	Implemented step definitions with Spring integration	14/11/2025
iot-solutions-development-cargasafe-backend	feature/monitoring-testing	6c1b9d4	test: implement Cucumber steps for telemetry	Implemented step definitions with data table support	14/11/2025
iot-solutions-development-cargasafe-backend	feature/monitoring-testing	4f7e2a9	test: configure Cucumber runner	Configured test runner, Spring integration, and H2 database	14/11/2025

Resumen de Cobertura

Tipo de Test	Cantidad de Tests	Componentes Cubiertos
Unit Tests	19 tests	2 clases de dominio (MonitoringSession, TelemetryData)
Integration Tests	15 tests	4 servicios + 2 repositorios + persistencia
Acceptance Tests (BDD)	29 escenarios	Flujos completos end-to-end
Total	63 verificaciones	Domain, Application, Infrastructure e Interface Layers

Configuración de Testing:

- **Framework de Testing:** JUnit 5, Mockito, AssertJ
- **BDD Framework:** Cucumber 7.20.1 con Gherkin en español
- **Base de Datos de Pruebas:** H2 in-memory database
- **Spring Boot Test:** Con contexto completo de aplicación

Los tests garantizan la confiabilidad de la aplicación, proporcionando cobertura completa desde la lógica de dominio hasta los endpoints REST, y sirviendo como documentación ejecutable del comportamiento esperado del sistema.

6.2.2.6. Execution Evidence for Sprint Review.

Durante este sprint se logró el desarrollo funcional del frontend y backend de CargaSafe, cumpliendo con los objetivos establecidos para esta iteración. El trabajo se centró en construir las vistas principales del sistema en Angular Material y consolidar la conexión con el backend desarrollado en IntelliJ, el cual expone sus servicios mediante Swagger.

En el frontend, se trabajó con Angular bajo arquitectura modular standalone. Se priorizó la creación de las vistas correspondientes a los módulos Vehicles, Sensors, Trips y Alerts, asegurando consistencia visual, navegación fluida y consumo correcto de los endpoints expuestos por el backend. Cada pantalla incluye tablas, formularios y componentes reutilizables que facilitan la gestión de la información.

Por otro lado, el backend fue implementado en Java dentro de IntelliJ, estructurado en capas y utilizando controladores REST documentados con Swagger UI, lo que permite validar y probar cada operación. Se implementaron los endpoints iniciales necesarios para dar soporte al frontend y permitir la visualización y manipulación de datos en tiempo real.

Gracias a la integración entre ambos entornos, el sistema ya permite comunicación efectiva mediante HTTP, mostrando datos reales provenientes del backend en las vistas del frontend.

Frontend

Vista principal del módulo Vehicles

ID	Plate	Type	Capabilities	Status	IoT	Actions
1	ABC-123	Truck	REFRIGERATED	In Service	IMEI-123456789	
2	ASADASD	Van	REFRIGERATED, HEAVY_LOAD	Out Of Service	-	

Vista del módulo Sensors con listado y detalles

The screenshot shows the Sensors module interface. On the left is a sidebar with the CargaSafe logo and links: Dashboard, Vehicles, Sensors (which is highlighted in orange), Trips, Alerts, and Subscriptions. At the bottom of the sidebar is a Logout button. The main area is titled "IoT Devices" and contains a search bar with placeholder text "Search by IMEI or vehicle...". A dropdown menu shows "All states" and "All states". Below the search is a table with columns: ID (1), IMEI (IMEI-123456789), Connectivity (Online), Vehicle (ABC-123), and Actions (edit and delete icons). At the bottom of the table are buttons for "Items per page" (set to 5), navigation arrows, and a message "1 - 1 of 1".

Pantalla del módulo Trips mostrando historial y estados de viajes

The screenshot shows the Trips module interface. The sidebar is identical to the Sensors module, with the Trips link highlighted in orange. The main area is titled "Trips" and includes a "+ Create" button. It features a search bar and filters for "Status" (All) and "Created At" (Recent). Below these are buttons for "Created Between" and "Apply/Clear". A table lists four trips with columns: ID, Status, Driver, Total Deliveries, Completed deliveries, Created at, Started at, and Actions. The trips are: ID 3, Status CREATED, Driver 1, Total Deliveries 0, Completed deliveries 0, Created at 11/15/25, 5:21 AM, Started at -, Actions (eye, edit); ID 4, Status CREATED, Driver 1, Total Deliveries 0, Completed deliveries 0, Created at 11/15/25, 5:21 AM, Started at -, Actions (eye, edit); ID 2, Status CREATED, Driver 1, Total Deliveries 1, Completed deliveries 1, Created at 11/14/25, 4:01 PM, Started at -, Actions (eye, edit); ID 1, Status COMPLETED, Driver 1, Total Deliveries 1, Completed deliveries 1, Created at 11/14/25, 4:01 PM, Started at 11/14/25, 4:02 PM, Actions (eye). At the bottom are buttons for "Items per page" (5), navigation arrows, and a message "0 of 0".

Módulo Alerts con alertas activas, estados y acciones disponibles

ID	Type	Delivery Order ID	Status	Created	Closed	Actions	Details
1	TEMPERATURE	1	ACKNOWLEDGED	November 14, 2025 at 4:03:31 PM GMT-5	---	<button>Mark as resolved</button>	
2	TEMPERATURE	1	ACKNOWLEDGED	November 14, 2025 at 4:03:40 PM GMT-5	---	<button>Mark as resolved</button>	
3	TEMPERATURE	1	CLOSED	November 14, 2025 at 4:03:45 PM GMT-5	November 15, 2025 at 3:33:41 AM GMT-5	Resolved	
4	TEMPERATURE	1	ACKNOWLEDGED	November 14, 2025 at 4:03:50 PM GMT-5	---	<button>Mark as resolved</button>	

Repositorio Frontend: [Repositorio de CargaSafe Frontend](#)

Aplicación en línea: [Enlace a CargaSafe Fleet Vehicles](#)

Backend

Vehículos

GET /api/v1/fleet/vehicles Get all vehicles

Parameters

No parameters

Responses

Curl

```
curl -X 'GET' \
'https://iot-solutions-development-cargasafe.onrender.com/api/v1/fleet/vehicles' \
-H 'accept: */*'
```

Request URL

<https://iot-solutions-development-cargasafe.onrender.com/api/v1/fleet/vehicles>

Server response

Code	Details
200	Response body <pre>[{ "id": 1, "plate": "ABC-123", "type": "TRUCK", "capabilities": ["REFRIGERATED"], "status": "IN_SERVICE", "odometerKm": 234, "deviceImeis": ["IMEI-123456789"] }, { "id": 2, "plate": "ASADASD", "type": "VAN", "capabilities": ["REFRIGERATED", "HEAVY_LOAD"], "status": "OUT_OF_SERVICE", "odometerKm": 0, "deviceImeis": [] }]</pre> <p> </p> <p>Response headers</p> <pre>alt-svc: h3="443"; ma=86400 cf-cache-status: DYNAMIC cf-ray: 99ecfcf19f6da5cdca-LIM content-encoding: br content-type: application/json date: Sat, 15 Nov 2025 07:13:04 GMT priority: u1,i rndr-id: 1869f25e-848d-4da2 server: cloudflare server-timing: cfExtPri</pre>

Sensores

GET /api/v1/fleet/devices Get all devices

Parameters

No parameters

Execute Clear

Responses

Curl

```
curl -X 'GET' \
'https://iot-solutions-development-cargasafe.onrender.com/api/v1/fleet/devices' \
-H 'accept: */*' 
```

Request URL

<https://iot-solutions-development-cargasafe.onrender.com/api/v1/fleet/devices>

Server response

Code	Details
200	<p>Response body</p> <pre>[{ "id": 1, "imei": "IMEI-123456789", "firmware": "v1.0.0", "online": true, "vehiclePlate": "ABC-123" }] </pre> <p>Response headers</p> <pre>alt-svc: h3=":443"; ma=86400 cf-cache-status: DYNAMIC cf-ray: 99ecf3083905a9ea-1TM content-encoding: br content-type: application/json date: Sat,15 Nov 2025 07:14:02 GMT priority: u1,i rnrnd-id: 0061d814-20dd-436f server: cloudflare server-timing: cfTxtPri vary: Origin,Access-Control-Request-Method,Access-Control-Request-Headers,Accept-Encoding x-render-origin-server: Render</pre>

Responses

Code	Description	Links
200	Devices returned	No links

Media type

/* 

Viajes

Trips Endpoint for managing trips sources

GET /api/v1/trips List all trips

Parameters

No parameters

Responses

Curl

```
curl -X 'GET' \
  'https://iot-solutions-development-cargasafe.onrender.com/api/v1/trips' \
  -H 'accept: */*
```

Request URL

<https://iot-solutions-development-cargasafe.onrender.com/api/v1/trips>

Server response

Code **Details**

200 Response body

```
[{"id": 1, "merchantId": 1, "driverId": 1, "deviceId": 1, "status": "CREATED", "createdAt": "2025-11-14T16:01:45.508312", "startedAt": null, "completedAt": null, "order": null, "id": 1, "name": "Warehouse 1", "address": "Plaza Mayor de Lima, Cercado de Lima", "latitude": -12.046374, "longitude": -77.042795}, {"deliveryOrders": [{"id": 2, "tripId": 1, "clientEmail": "Daniel@hotmail.com", "sequenceOrder": 1, "address": "Av Brasil, Jesús María, Perú", "serviceLat": null, "status": "PENDING", "minHumidity": null, "maxHumidity": null}]}]
```

Response headers

```
alt-svc: h3="443"; ma=86400
cache-control: DYNAMIC
cf-ray: 700000000000000000
content-encoding: br
content-type: application/json
date: Sat, 15 Nov 2025 00:47:55 GMT
pragma: no-cache
x-edge-id: S1f6sc0-#3Be-4ead
server: cloudflare
server-timing: cfExtPri
vary: Origin,Access-Control-Request-Method,Access-Control-Request-Headers,Accept-Encoding
x-render-origin-server: Render
```

Responses

Code **Description**

Links

Alertas

Alerts Endpoint for managing alerts sources

GET /api/v1/alerts

Parameters

No parameters

Execute **Cancel** **Clear**

Responses

Curl

```
curl -X 'GET' \
'https://iot-solutions-development-cargasafe.onrender.com/api/v1/alerts' \
-H 'accept: */*'
```

Request URL

<https://iot-solutions-development-cargasafe.onrender.com/api/v1/alerts>

Server response

Code **Details**

200 Response body

```
[
  {
    "id": 1,
    "deliveryOrderId": 1,
    "alertType": "TEMPERATURE",
    "alertStatus": "ACKNOWLEDGED",
    "incidents": [
      {
        "id": 1,
        "alertId": 1,
        "description": "Threshold Exceeded",
        "createdAt": "2025-11-14T16:03:31.436556",
        "acknowledgedAt": "2025-11-14T16:07:21.972417",
        "closedAt": null
      }
    ],
    "notification": [
      {
        "id": 3,
        "alertId": 1,
        "notificationChannel": "EMAIL",
        "message": "The TEMPERATURE threshold has been exceeded.",
        "sentAt": "2025-11-14T16:03:31.43659"
      }
    ],
    {
      "id": 2,
      "deliveryOrderId": 1,
      "alertType": "TEMPERATURE"
    }
]
```

Download

Response headers

```
alt-svc: h3=":443"; ma=86400
cf-cache-status: DYNAMIC
cf-ray: 98ecff9b6c58cdcd-LIM
content-encoding: br
content-type: application/json
date: Sat, 15 Nov 2025 07:15:47 GMT
priority: u=1,i
rndr-id: 5e754c40-e5ad-4b03
server: cloudflare
x-edge-timing: cf-req:1ms
```

Repositorio Backend: [Repositorio de CargaSafe Backend](#)

API en línea: [Swagger UI de CargaSafe Backend](#)

6.2.2.7. Services Documentation Evidence for Sprint Review

Durante este Sprint se consolidó la documentación de los Web Services del backend IoTParkers (CargaSafe) utilizando OpenAPI 3.0 y Swagger UI, publicada en el endpoint:

<https://iot-solutions-development-cargasafe-fake.onrender.com/>

El código fuente de los Web Services se encuentra en el repositorio:

<https://github.com/Los-Parkers-IoT/iot-solutions-development-cargasafe-backend>

La documentación cubre los bounded contexts principales del sistema: Fleet Management (Vehicles y Devices), Monitoring, Telemetry, Alerts, Trips, Authentication, Merchants, Origin Points, Users y Employees.

Gracias a esta especificación, el equipo puede descubrir, probar y validar los endpoints de forma interactiva, asegurando consistencia entre el diseño de la API y las implementaciones realizadas en este Sprint.

A continuación, se presenta la tabla resumen de los endpoints documentados, indicando para cada uno el verbo HTTP, la ruta, parámetros principales, un resumen del response y la sección correspondiente dentro de Swagger UI.

Módulo	Acción	Verbo	Path	Parámetros	Ejemplo de response	Swagger
Devices	Obtener dispositivo por id	GET	/api/v1/fleet/devices/{id}	path: id (UUID)	200 OK con DeviceDTO, 404 si no existe	/docs#/Devices

Módulo	Acción	Verbo	Path	Parámetros	Ejemplo de response	Swagger
Devices	Actualizar dispositivo	PUT	/api/v1/fleet/devices/{id}	path: id; body: DeviceUpdateRequest	200 OK con DeviceDTO actualizado	/docs#/Devices
Devices	Eliminar dispositivo	DELETE	/api/v1/fleet/devices/{id}	path: id	204 No Content, 404 si no existe	/docs#/Devices
Devices	Listar dispositivos	GET	/api/v1/fleet/devices	query opcional: page,size,online	200 OK lista paginada	/docs#/Devices
Devices	Crear dispositivo	POST	/api/v1/fleet/devices	body: DeviceCreateRequest	201 Created con Location	/docs#/Devices
Devices	Actualizar firmware	POST	/api/v1/fleet/devices/{id}/firmware	path: id; body: FirmwareUpdateRequest	202 Accepted	/docs#/Devices
Devices	Cambiar estado online	PATCH	/api/v1/fleet/devices/{id}/online	path: id; body: OnlineStatusRequest	200 OK con nuevo estado	/docs#/Devices
Devices	Listar por estado online	GET	/api/v1/fleet/devices/by-online/{online}	path: online (true/false)	200 OK lista filtrada	/docs#/Devices
Devices	Buscar por IMEI	GET	/api/v1/fleet/devices/by-imei/{imei}	path: imei	200 OK DeviceDTO, 404 si no existe	/docs#/Devices
Telemetry	Registrar telemetría	POST	/api/v1/telemetry	body: TelemetryRecordRequest	201 Created (id del registro)	/docs#/Telemetry
Telemetry	Obtener telemetría por sesión	GET	/api/v1/telemetry/session/{sessionId}	path: sessionId	200 OK lista	/docs#/Telemetry
Notifications	Obtener notificaciones de alerta	GET	/api/v1/notifications/alert/{alertId}	path: alertId	200 OK con canales	/docs#/Notifications
Incidents	Obtener incidentes de alerta	GET	/api/v1/incidents/alert/{alertId}	path: alertId	200 OK lista	/docs#/Incident
Vehicles	Obtener vehículo por id	GET	/api/v1/fleet/vehicles/{id}	path: id	200 OK, 404 si no existe	/docs#/Vehicles
Vehicles	Actualizar vehículo	PUT	/api/v1/fleet/vehicles/{id}	path: id; body: VehicleUpdateRequest	200 OK actualizado	/docs#/Vehicles
Vehicles	Eliminar vehículo	DELETE	/api/v1/fleet/vehicles/{id}	path: id	204 No Content	/docs#/Vehicles
Vehicles	Listar vehículos	GET	/api/v1/fleet/vehicles	query: page,size,status,type	200 OK lista	/docs#/Vehicles
Vehicles	Crear vehículo	POST	/api/v1/fleet/vehicles	body: VehicleCreateRequest	201 Created	/docs#/Vehicles
Vehicles	Desasignar dispositivo	POST	/api/v1/fleet/vehicles/{id}/unassign-device/{imei}	path: id, imei	200 OK	/docs#/Vehicles
Vehicles	Asignar dispositivo	POST	/api/v1/fleet/vehicles/{id}/assign-device/{imei}	path: id, imei	200 OK	/docs#/Vehicles
Vehicles	Cambiar estado	PATCH	/api/v1/fleet/vehicles/{id}/status	path: id; body: VehicleStatusRequest	200 OK	/docs#/Vehicles
Vehicles	Listar por tipo	GET	/api/v1/fleet/vehicles/by-type/{type}	path: type	200 OK lista	/docs#/Vehicles
Vehicles	Listar por estado	GET	/api/v1/fleet/vehicles/by-status/{status}	path: status	200 OK lista	/docs#/Vehicles
Vehicles	Buscar por placa	GET	/api/v1/fleet/vehicles/by-plate/{plate}	path: plate	200 OK o 404	/docs#/Vehicles
Authentication	Registro de usuario	POST	/api/v1/authentication/sign-up	body: SignUpRequest	201 Created	/docs#/Authentication

Módulo	Acción	Verbo	Path	Parámetros	Ejemplo de response	Swagger
Authentication	Inicio de sesión	POST	/api/v1/authentication/sign-in	body: SignInRequest	200 OK con tokens	/docs#/Authent
Authentication	Revocar token	POST	/api/v1/authentication/revoke	body: RevokeTokenRequest	204 No Content	/docs#/Authent
Authentication	Refrescar token	POST	/api/v1/authentication/refresh	body: RefreshTokenRequest	200 OK nuevo token	/docs#/Authent
Merchants	Listar empresas	GET	/api/v1/merchants	query: page,size	200 OK paginado	/docs#/Merchar
Merchants	Crear empresa	POST	/api/v1/merchants	body: MerchantCreateRequest	201 Created	/docs#/Merchar
Merchants	Agregar empleado	POST	/api/v1/merchants/{id}/employee	path: id; body: MerchantEmployeeRequest	200 OK	/docs#/Merchar
Merchants	Obtener empresa por id	GET	/api/v1/merchants/{id}	path: id	200 OK	/docs#/Merchar
Origin Points	Listar puntos	GET	/api/v1/origin-points	query: page,size	200 OK paginado	/docs#/Origin-P
Origin Points	Crear punto	POST	/api/v1/origin-points	body: OriginPointCreateRequest	201 Created	/docs#/Origin-P
Alerts	Listar alertas	GET	/api/v1/alerts	query: page,size,status,type	200 OK paginado	/docs#/Alerts
Alerts	Crear alerta manual	POST	/api/v1/alerts	body: AlertCreateRequest	201 Created	/docs#/Alerts
Alerts	Cerrar alerta	PATCH	/api/v1/alerts/{alertId}/close	path: alertId	200 OK CLOSED	/docs#/Alerts
Alerts	Reconocer alerta	PATCH	/api/v1/alerts/{alertId}/acknowledge	path: alertId	200 OK ACKNOWLEDGED	/docs#/Alerts
Alerts	Obtener alerta por id	GET	/api/v1/alerts/{alertId}	path: alertId	200 OK o 404	/docs#/Alerts
Alerts	Listar por tipo	GET	/api/v1/alerts/type/{type}	path: type	200 OK lista	/docs#/Alerts
Alerts	Listar por estado	GET	/api/v1/alerts/status/{status}	path: status	200 OK lista	/docs#/Alerts
Monitoring	Crear sesión	POST	/api/v1/monitoring/sessions	body: MonitoringSessionCreateRequest	201 Created	/docs#/Monitor
Monitoring	Reanudar sesión	POST	/api/v1/monitoring/sessions/{sessionId}/resume	path: sessionId	200 OK	/docs#/Monitor
Monitoring	Pausar sesión	POST	/api/v1/monitoring/sessions/{sessionId}/pause	path: sessionId	200 OK	/docs#/Monitor
Monitoring	Finalizar sesión	POST	/api/v1/monitoring/sessions/{sessionId}/end	path: sessionId	200 OK	/docs#/Monitor
Monitoring	Obtener sesión por id	GET	/api/v1/monitoring/sessions/{sessionId}	path: sessionId	200 OK	/docs#/Monitor
Monitoring	Obtener por viaje	GET	/api/v1/monitoring/sessions/trip/{tripId}	path: tripId	200 OK	/docs#/Monitor
Monitoring	Listar activas	GET	/api/v1/monitoring/sessions/active	query: page,size	200 OK lista	/docs#/Monitor
Roles	Listar roles	GET	/api/v1/roles	query: page,size	200 OK lista	/docs#/Roles
Trips	Listar viajes	GET	/api/v1/trips	query: page,size,dateFrom,dateTo	200 OK lista	/docs#/Trips
Trips	Crear viaje	POST	/api/v1/trips	body: TripCreateRequest	201 Created	/docs#/Trips
Trips	Obtener viaje por id	GET	/api/v1/trips/{tripId}	path: tripId	200 OK, 404 si no existe	/docs#/Trips
Trips	Buscar por rango	GET	/api/v1/trips/search	query: startDate,endDate,page,size	200 OK filtrado	/docs#/Trips
Trips	Listar por empresa	GET	/api/v1/trips/merchant/{merchantId}	path: merchantId	200 OK	/docs#/Trips
Users	Listar usuarios	GET	/api/v1/users	query: page,size	200 OK lista	/docs#/Users

Módulo	Acción	Verbo	Path	Parámetros	Ejemplo de response	Swagger
Users	Obtener usuario por id	GET	/api/v1/users/{userId}	path: userId	200 OK o 404	/docs#/Users
Employees	Listar empleados	GET	/api/v1/employees	query: page,size	200 OK	/docs#/Employees
Employees	Obtener empleado por id	GET	/api/v1/employees/{id}	path: id	200 OK o 404	/docs#/Employees
Employees	Listar por empresa	GET	/api/v1/employees/merchants/{merchantId}	path: merchantId	200 OK lista	/docs#/Employees

- En la sección Devices, se utilizó el endpoint `POST /api/v1/fleet/devices` con un cuerpo de ejemplo (IMEI, tipo de dispositivo, descripción). Swagger muestra el request body esperado y, al pulsar "Try it out" y luego "Execute", devuelve un 201 Created con el objeto `DeviceDTO` creado y el header `Location` apuntando al recurso recién creado.
- En Vehicles, se probaron los filtros `GET /api/v1/fleet/vehicles/by-plate/{plate}` y `GET /api/v1/fleet/vehicles/by-status/{status}` usando datos de placas y estados registrados en la base de datos. Esto permitió validar los casos de uso de búsqueda rápida por placa y listado de flota activa.
- En Monitoring y Telemetry, se creó una sesión de monitoreo (`POST /api/v1/monitoring/sessions`) y se enviaron registros de telemetría (`POST /api/v1/telemetry`) asociados a esa sesión. Posteriormente se consultaron los datos con `GET /api/v1/telemetry/session/{sessionId}` para verificar que la información se estaba almacenando y recuperando correctamente.
- En el módulo Alerts, se probaron las transiciones de estado utilizando `PATCH /api/v1/alerts/{alertId}/acknowledge` y `PATCH /api/v1/alerts/{alertId}/close`, comprobando que el response actualiza el estado de la alerta y que los filtros `GET /api/v1/alerts/status/{status}` reflejan dichos cambios.

Cada una de estas interacciones quedó registrada en Swagger y puede ser reproducida fácilmente por otros miembros del equipo, reduciendo la fricción al integrar el frontend web y la aplicación móvil con estos Web Services.

Los cambios más relevantes asociados a la documentación y exposición de endpoints para este Sprint se registraron en los siguientes commits de la rama `main`:

- [42e4452 – feature\(iam\): add OpenApiConfiguration](#) Incorpora la configuración centralizada de OpenAPI para el backend, permitiendo generar el documento OAS 3.0 y habilitar Swagger UI para todos los bounded contexts.
- [cedbb18 – chore\(swagger\): remove bearerAuth scheme from OpenAPI to disable Swagger authorize button](#) Ajusta el esquema de seguridad en la especificación OpenAPI, deshabilitando temporalmente el botón "Authorize" de Swagger para facilitar las pruebas locales sin JWT durante el Sprint.
- [f46d243 – feat\(fleet-management\): expose PATCH /devices/{id}/online endpoint](#) Expone y documenta el endpoint `PATCH /api/v1/fleet/devices/{id}/online` dentro del módulo `Devices`, incluyendo su descripción, parámetros y ejemplo de response en Swagger UI.
- [c725c06 – feat\(fleet-management\): expose PATCH /vehicles/{id}/status endpoint](#) Expone y documenta el endpoint `PATCH /api/v1/fleet/vehicles/{id}/status` dentro del módulo `Vehicles`, permitiendo cambiar el estado operativo del vehículo y visualizar el contrato en la documentación OpenAPI.

6.2.2.8. Software Deployment Evidence for Sprint Review

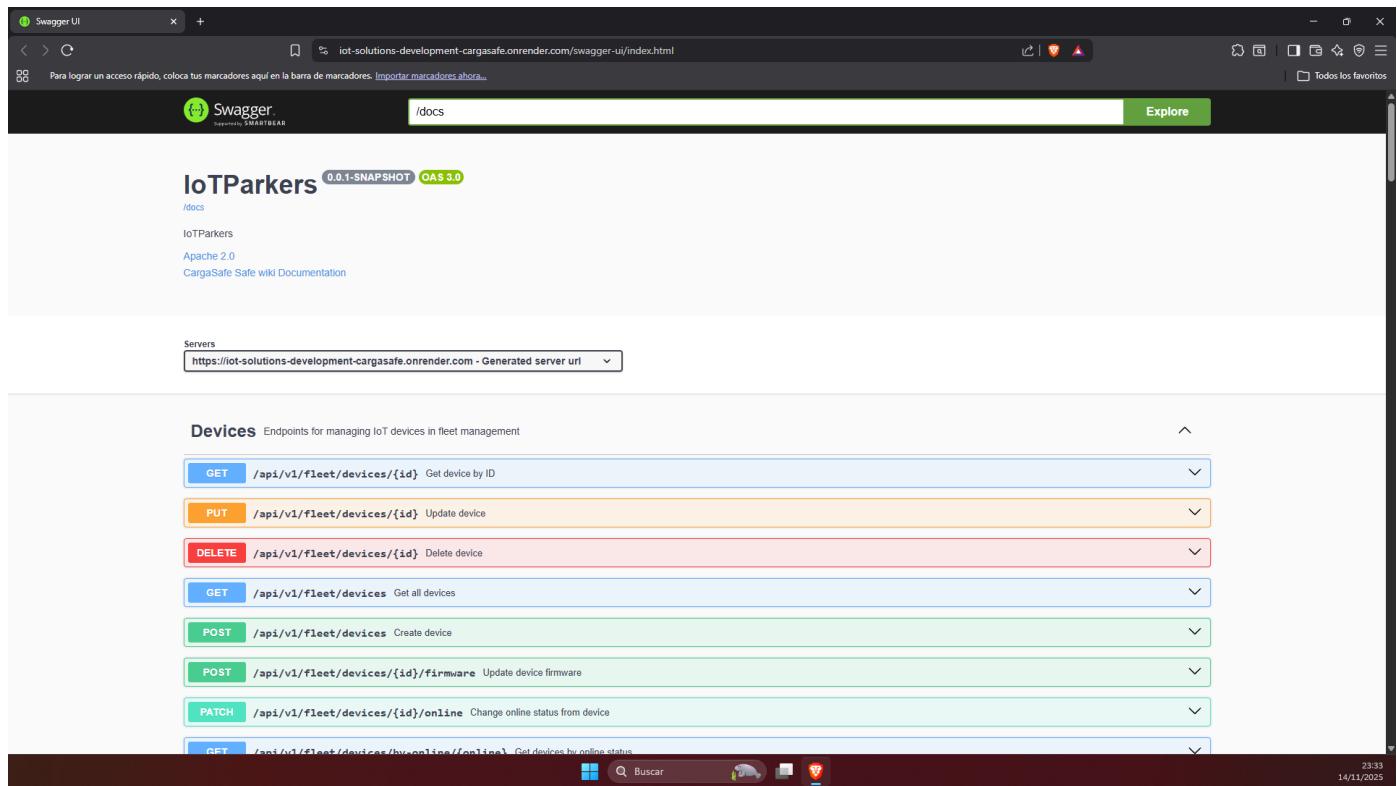
Durante el Sprint 2, se realizó el despliegue del **backend de CargaSafe** utilizando **Render** como plataforma de hosting en la nube. Este despliegue permitió exponer los servicios REST desarrollados con Spring Boot, facilitando la integración con las aplicaciones web y móviles del proyecto.

El backend fue configurado para ejecutarse en un entorno de producción con las siguientes características:

- Plataforma de Despliegue:** Render (render.com)
- Framework:** Spring Boot 3.x con Java 21
- Base de Datos:** PostgreSQL (proporcionado por Render)
- Documentación de API:** Swagger UI / OpenAPI 3.0
- URL de Despliegue:** <https://iot-solutions-development-cargasafe.onrender.com/swagger-ui/index.html>

Evidencia de Despliegue:

La siguiente captura de pantalla muestra la documentación interactiva de la API desplegada en Render, accesible a través de Swagger UI. Se pueden observar los diferentes bounded contexts implementados (Devices, Vehicles, Fleet Management) con sus respectivos endpoints REST documentados:



La interfaz de Swagger UI permite:

- Explorar todos los endpoints disponibles organizados por bounded contexts
- Visualizar los esquemas de datos (DTOs, Request/Response models)
- Probar los endpoints directamente desde el navegador
- Acceder a la especificación OpenAPI completa

El despliegue en Render incluye:

- Integración continua desde el repositorio GitHub
- Configuración de variables de entorno para conexión a base de datos
- Logs de aplicación accesibles para monitoreo y debugging
- SSL/HTTPS habilitado por defecto para comunicación segura

Este despliegue garantiza que los servicios backend estén disponibles de forma pública y estable, permitiendo que los equipos de frontend (web y móvil) puedan consumir las APIs para implementar las funcionalidades de monitoreo de carga en tiempo real.

6.2.2.9. Team Collaboration Insights during Sprint.

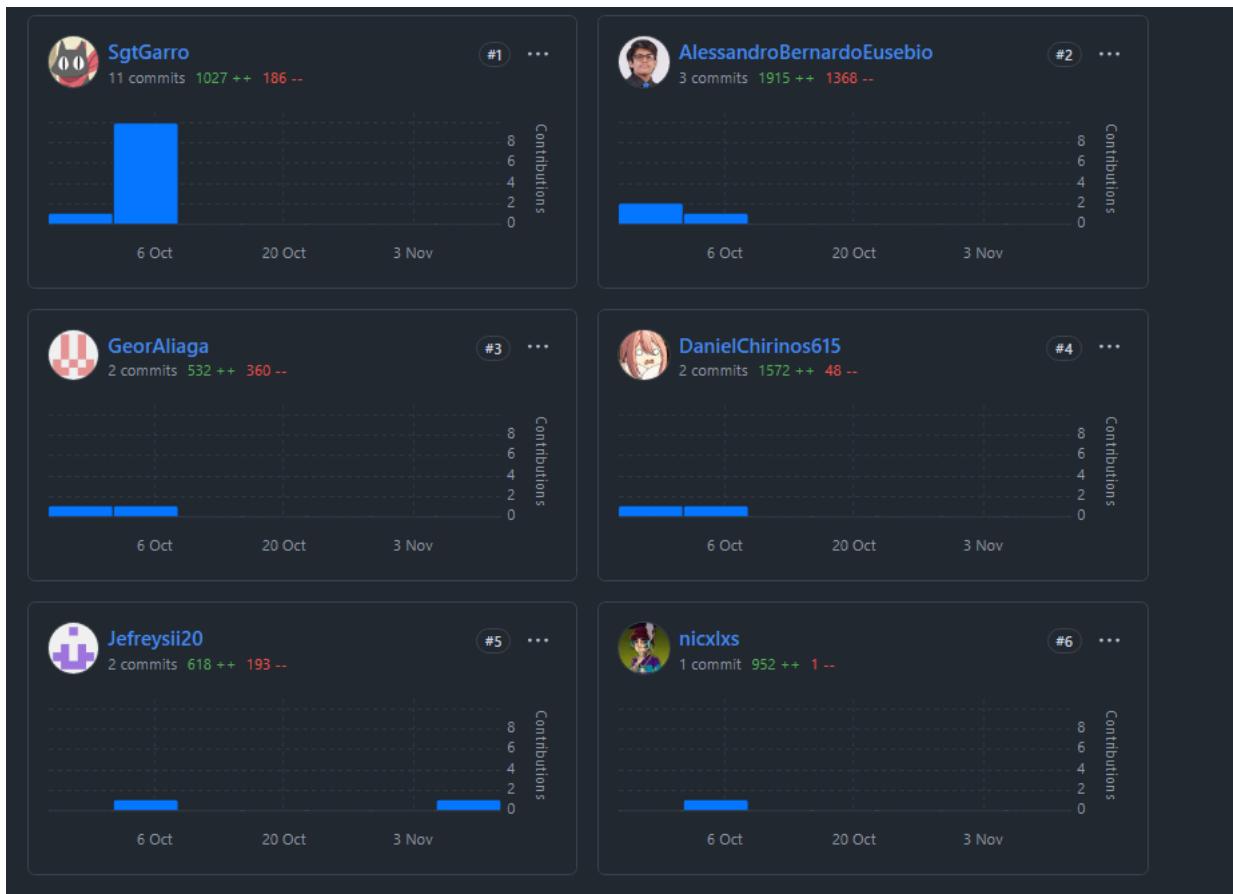
Durante este sprint, el equipo de desarrollo trabajó de forma colaborativa en la implementación de las principales funcionalidades correspondientes al alcance definido: la Landing Page y la Aplicación Web de CargaSafe. A lo largo del proceso, se mantuvo una comunicación constante mediante GitHub y los canales del equipo, asegurando una adecuada distribución de tareas y el cumplimiento de los objetivos propuestos.

Los analíticos de GitHub reflejan la participación activa de todos los integrantes, evidenciando commits frecuentes, revisiones de código y fusiones entre ramas, lo que demuestra un flujo de trabajo coordinado y una integración continua del proyecto. Además, se promovió la retroalimentación mutua y la resolución conjunta de incidencias técnicas, fortaleciendo el liderazgo compartido y la cooperación dentro del equipo.

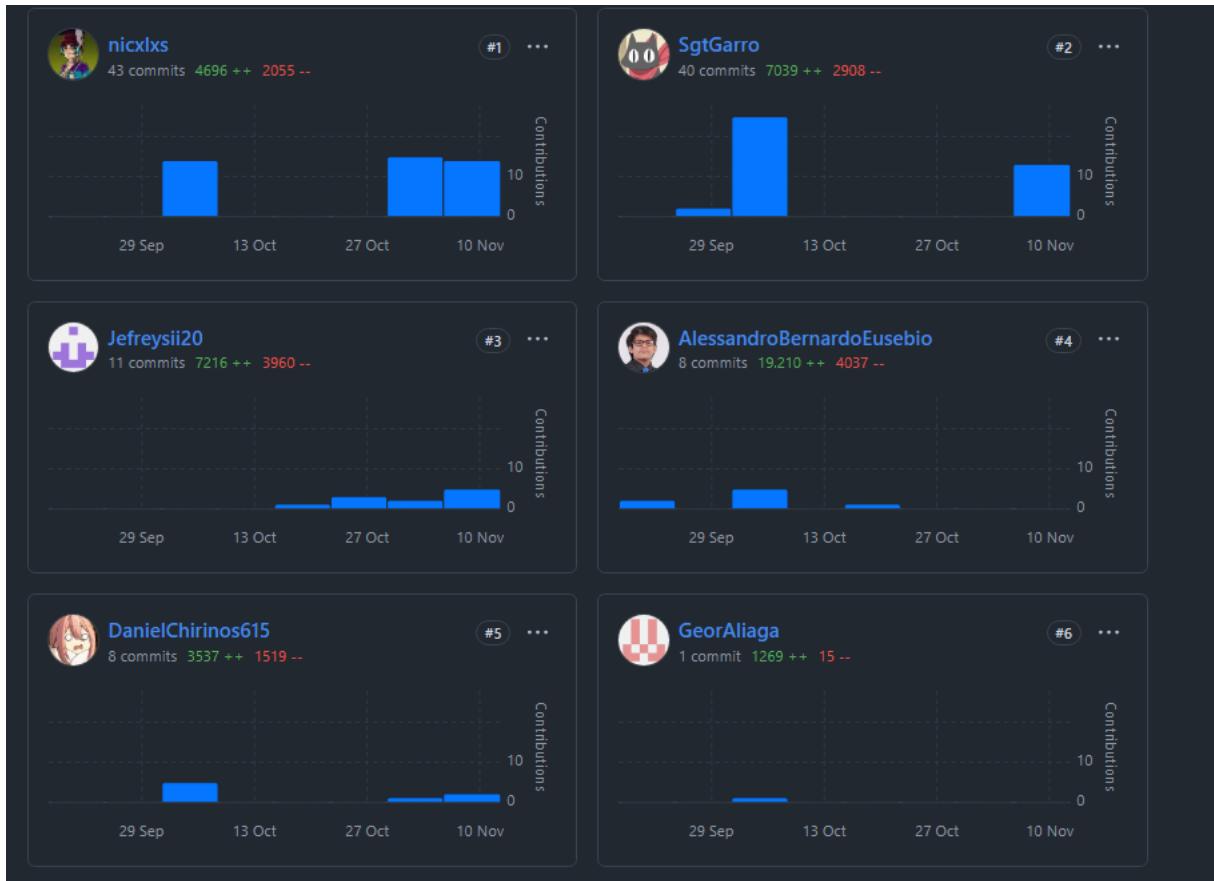
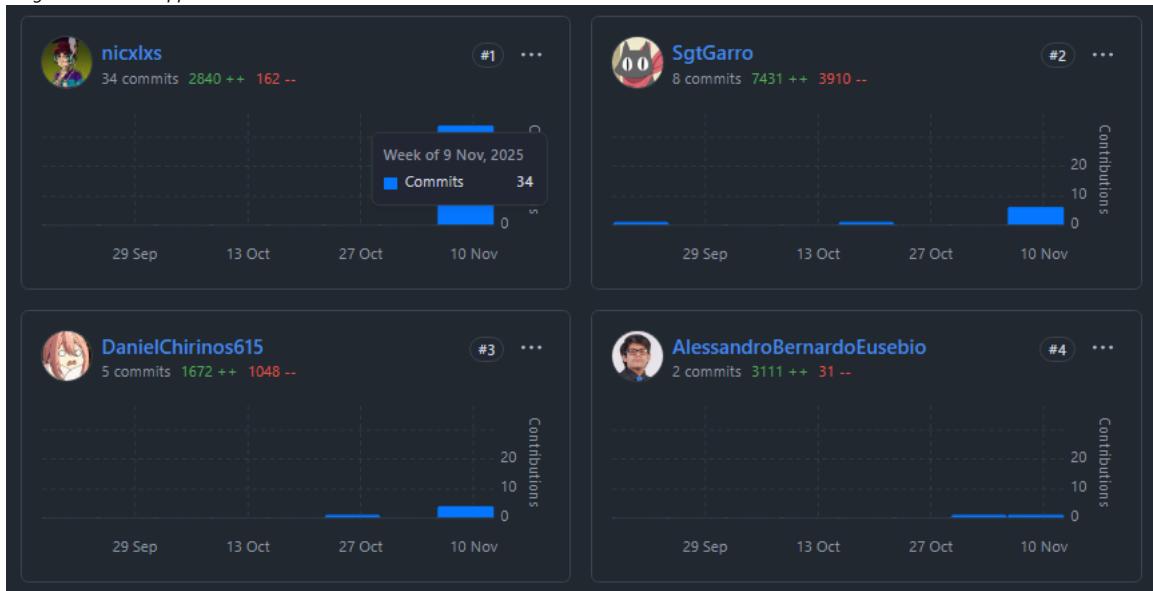
Insights de Landing Page



Insights de Web Application



Insights de Web Services

*Insights de Mobile Application**Insights de Edge Application**Insights de Embbeded Application*



6.2.3. Sprint 3

6.2.3.1. Sprint Planning 3.

En esta sección se especifican los aspectos principales del Sprint Planning Meeting para el Sprint 3 del proyecto CargaSafe. Este primer sprint se enfoca en establecer las bases de la plataforma, implementando el landing page y los componentes iniciales de la aplicación web que permitan a los usuarios conocer el producto y comenzar a interactuar con las funcionalidades core del sistema de monitoreo de carga.

Sprint #
Sprint 3

Sprint Planning Background Date 2025-12-23 Time 09:00 AM Location Reunión virtual mediante Discord Prepared By Marcelo Fabian Garro Vega Attendees (to planning meeting) Marcelo Fabian Garro Vega / Jefrey Martin Sanchez Ignacio / Daniel Rodrigo Chirinos Arevalo / Alessandro Joaquin Bernardo Eusebio / Vera Nuñez Nicolas Alejandro / George Aliaga Sprint 3 Review Summary Durante el Sprint 3 se implementaron los módulos de Identity and Access Management (IAM), Subscriptions y Analytics, permitiendo gestionar usuarios, accesos, planes de suscripción y métricas del sistema. Asimismo, se realizó la refactorización del código de la embedded application, optimizando su estructura y estabilidad general. Sprint 3 Retrospective Summary El equipo logró cumplir los objetivos del sprint con una decente coordinación. Se identificó la necesidad de mejorar la validación de integraciones entre frontend, backend y embedded application, así como reforzar las pruebas posteriores a los procesos de refactorización. Sprint Goal & User Stories Sprint 3 Goal Nos enfocamos en expandir CargaSafe hacia una plataforma más robusta para validaciones de negocio, gestión de usuarios y análisis del comportamiento del sistema.

Creemos que esto entrega mayor seguridad, escalabilidad y visibilidad operativa a operadores logísticos, gestores de flota y equipos de monitoreo.

Esto se confirmará cuando los usuarios puedan gestionar roles y permisos desde IAM, activar planes desde Subscriptions, visualizar métricas desde Analytics y el dispositivo IoT opere de forma estable en escenarios reales. Sprint 3 Velocity 31 Story Points Sum of Story Points 31 Story Points

6.2.3.2. Aspect Leaders and Collaborators.

Para el Sprint 3, los principales aspectos considerados corresponden a los bounded contexts core identificados en la arquitectura del sistema CargaSafe..

Los aspectos principales son:

1. **Landing Page:** Página de aterrizaje con información del producto y llamados a la acción
2. **Fleet Management:** Gestión de vehículos y dispositivos IoT asociados
3. **Trip Management:** Creación, ejecución y seguimiento de viajes
4. **Real-time Monitoring:** Monitoreo de condiciones ambientales en tiempo real
5. **Alerts and Resolution:** Generación, visualización y resolución de alertas
6. **Embedded Application:** Dispositivo IoT que funciona para monitoreo
7. **Identity and Access Management (IAM):** Gestión de usuarios, autenticación y control de accesos
8. **Visualization and Analytics:** Dashboards, gráficos y reportes analíticos
9. **Subscriptions and Payments:** Gestión de planes de suscripción y pagos

A continuación, se presenta la matriz LACX (Leadership and Collaboration Matrix) que indica el líder (L) y los colaboradores (C) para cada aspecto del Sprint 3:

Team Member (Last Name, First Name)	GitHub Username	Landing Page L / C	Fleet Management L / C	Trip Management L / C	Real-time Monitoring L / C	Alerts L / C	Embedded Application L / C	Identity and Access Management (IAM) L / C	Visualization and Analytics L / C	Subscriptions and Payments L / C
Garro Vega, Marcelo Fabian	Marcelo Garro	C	C	C	L	C	C	C	C	C
Sanchez Ignacio, Jefrey Martin	Jefreysi20	C	C	C	C	C	C	C	C	L
Aliaga Pimentel, George Arturo	GeorAliaga	C	L	C	C	C	C	L	C	C

Vera Nuñez, Nicolas Alejandro	nicxlxs	C	C	L	C	C	C	L	C	C
Chirinos Arevalo, Daniel Rodrigo	DanielChirinos615	C	C	C	C	L	L	C	C	C
Bernardo Eusebio, Alessandro Joaquin	Alessandro Bernardo	L	C	C	C	C	C	C	L	C

Leyenda:

- L (Leader):** Responsable principal del aspecto, encargado de la coordinación, toma de decisiones técnicas y revisión final del código.
- C (Collaborator):** Contribuye activamente al desarrollo del aspecto, apoya en la implementación de funcionalidades y participa en las revisiones de código.

Esta matriz facilita la comunicación y clarifica las responsabilidades dentro del equipo, asegurando que cada bounded context tenga un líder definido que coordine su desarrollo, mientras que todos los miembros colaboran de manera transversal para garantizar la integración coherente del sistema.

6.2.3.3. Sprint Backlog 3.

Durante este sprint, se completó la implementación del módulo de Identity and Access Management (IAM), se desarrolló el bounded context de Visualization and Analytics en sus tres capas (backend, frontend y mobile), se realizó la refactorización del código de la aplicación embebida para optimizar su estructura y rendimiento, y se implementó el bounded context de Profile and Preferences Management tanto en frontend como en backend, permitiendo la gestión completa de perfiles de usuario y configuración de preferencias.

La gestión del sprint se llevó a cabo utilizando la herramienta Jira, donde se registraron las tareas, el estado y los responsables.

Tablero Sprint 2 16 oct – 9 nov (11 actividades)						
GS-38	Lista de viajes registrados	PLANIFICACIÓN DE ...	FINALIZADA ✓	25 oct	5	JS
GS-24	Registro de dispositivos IoT	GESTIÓN DE FLOTA	FINALIZADA ✓	29 oct	5	N
GS-42	Filtrado de viajes por fecha	PLANIFICACIÓN DE ...	FINALIZADA ✓	7 nov	3	JS
GS-35	Obtener datos de telemetría en tiempo real	MONITOREO EN TIEM...	FINALIZADA ✓	30 oct	8	GP
GS-28	Ver todos los vehículos de carga	GESTIÓN DE FLOTA	FINALIZADA ✓	25 oct	3	N
GS-21	Registro de vehículos de carga	GESTIÓN DE FLOTA	FINALIZADA ✓	26 oct	5	N
GS-36	Alertas por incumplimiento de parámetros	MONITOREO EN TIEM...	FINALIZADA ✓	27 oct	5	DA
GS-29	Ver todos los dispositivos IoT en la plataforma	GESTIÓN DE FLOTA	FINALIZADA ✓	25 oct	3	AE
GS-26	Asignar dispositivo a vehículo de carga	GESTIÓN DE FLOTA	FINALIZADA ✓	2 nov	3	AE
GS-39	Detalle de viaje	PLANIFICACIÓN DE ...	FINALIZADA ✓	2 nov	3	TS
GS-32	Creación de viajes y puntos de envío	PLANIFICACIÓN DE ...	FINALIZADA ✓	8 nov	5	MG

Sprint # Sprint 3

User Story	Work-Item / Task	Task Id	Task Title	Task Description	Estimation (Hours)	Assigned To	Status
User Story Id	User Story Title	Task Id	Task Title	Task Description	Estimation (Hours)	Assigned To	Status
TS001	API de autenticación	T01	Implementar endpoint de login	Crear endpoint POST /api/v1/auth/login con validación de credenciales	4	Marcelo Garro	Done
TS001	API de autenticación	T02	Implementar endpoint de logout	Crear endpoint POST /api/v1/auth/logout para cerrar sesión	3	Marcelo Garro	Done
TS001	API de autenticación	T03	Implementar endpoint de refresh token	Crear endpoint POST /api/v1/auth/refresh para renovar tokens	3	Marcelo Garro	Done
TS002	Servicio de autenticación con JWT	T04	Configurar generación de tokens JWT	Implementar lógica de generación de JWT con claims de usuario y rol	5	Marcelo Garro	Done
TS002	Servicio de autenticación con JWT	T05	Implementar validación de tokens	Crear middleware para validar tokens JWT en requests protegidos	4	Marcelo Garro	Done
US018	Registro de usuario	T06	Crear formulario de registro en frontend	Diseñar e implementar formulario de registro con validaciones	4	Nicolas Vera	Done
US018	Registro de usuario	T07	Conectar registro con backend	Integrar formulario de registro con endpoint de API	3	Nicolas Vera	Done

User Story Id	User Story Title	Task Id	Task Title	Task Description	Estimation (Hours)	Assigned To	Status
US010	Inicio de sesión	T08	Crear pantalla de login en web	Maquetar página de login con Material Design	3	Nicolas Vera	Done
US010	Inicio de sesión	T09	Crear pantalla de login en mobile	Implementar pantalla de login en Flutter	4	George Aliaga	Done
US010	Inicio de sesión	T10	Integrar login con IAM backend	Conectar formularios de login con servicio de autenticación	3	Nicolas Vera	Done
US011	Cerrar sesión	T11	Implementar funcionalidad de logout	Crear lógica de cierre de sesión y limpieza de tokens	2	George Aliaga	Done
US012	Recuperar contraseña	T12	Crear flujo de recuperación de contraseña	Implementar pantallas y lógica de recuperación de contraseña	5	Nicolas Vera	Done
US035	Gráficos de incidencias por mes	T13	Crear endpoint de métricas de incidencias	Desarrollar endpoint para obtener datos agregados de incidencias	5	Alessandro Bernardo	Done
US035	Gráficos de incidencias por mes	T14	Implementar gráfico de incidencias en web	Crear componente de gráfico con Chart.js para visualizar incidencias	4	Alessandro Bernardo	Done
US035	Gráficos de incidencias por mes	T15	Implementar gráfico en mobile	Adaptar visualización de incidencias para aplicación móvil	4	Alessandro Bernardo	Done
US034	Gráficos de tiempo y temperatura	T16	Crear servicio de datos de temperatura	Desarrollar endpoint para obtener series temporales de temperatura	5	Alessandro Bernardo	Done
US034	Gráficos de tiempo y temperatura	T17	Implementar gráfico de línea en frontend	Crear componente de gráfico de línea para temperatura vs tiempo	4	Alessandro Bernardo	Done
US036	Filtrado de viajes por fecha	T18	Agregar filtros en dashboard de analytics	Implementar controles de filtrado por rango de fechas	3	Alessandro Bernardo	Done
US037	Descarga de reporte de viajes	T19	Crear endpoint de generación de reportes	Implementar servicio backend para generar reportes en PDF	6	Alessandro Bernardo	Done
US037	Descarga de reporte de viajes	T20	Integrar botón de descarga en frontend	Añadir funcionalidad de descarga de reportes en interfaz web	3	Alessandro Bernardo	Done
TS003	Refactorización de embedded application	T21	Optimizar estructura de código embebido	Refactorizar código para mejorar legibilidad y mantenibilidad	6	Daniel Chirinos	Done
TS003	Refactorización de embedded application	T22	Optimizar transmisión de datos IoT	Mejorar eficiencia de envío de telemetría desde dispositivos	5	Daniel Chirinos	Done
TS003	Refactorización de embedded application	T23	Documentar arquitectura de embedded app	Crear documentación técnica de la aplicación embebida	3	Daniel Chirinos	Done
US019	Gestionar perfil de usuario	T24	Crear modelo de perfil en backend	Definir entidades y servicios para gestión de perfiles	4	Jefrey Sanchez	Done
US019	Gestionar perfil de usuario	T25	Crear endpoints CRUD de perfil	Implementar endpoints para crear, leer, actualizar perfil	5	Jefrey Sanchez	Done
US019	Gestionar perfil de usuario	T26	Crear pantalla de perfil en web	Diseñar e implementar interfaz de gestión de perfil en web	4	George Aliaga	Done
US019	Gestionar perfil de usuario	T27	Crear pantalla de perfil en mobile	Implementar pantalla de perfil en aplicación móvil Flutter	4	George Aliaga	Done
US040	Configurar preferencias de notificación	T28	Implementar gestión de preferencias	Crear servicio backend para gestionar preferencias de usuario	4	Jefrey Sanchez	Done
US040	Configurar preferencias de notificación	T29	Crear interfaz de configuración en web	Implementar pantalla de configuración de notificaciones	3	George Aliaga	Done
TS004	Refactorización de Analytics bounded context	T30	Optimizar queries de agregación	Refactorizar consultas de base de datos para mejorar performance	5	Alessandro Bernardo	Done
TS004	Refactorización de Analytics bounded context	T31	Implementar caché de métricas	Agregar capa de caché para datos analíticos frecuentemente consultados	4	Alessandro Bernardo	Done
TS004	Refactorización de Analytics bounded context	T32	Mejorar arquitectura de servicios	Refactorizar servicios siguiendo principios SOLID y DDD	5	Alessandro Bernardo	Done

6.2.3.4. Development Evidence for Sprint Review

Durante el Sprint 3 se desarrollaron funcionalidades clave para consolidar la plataforma CargaSafe a nivel web, móvil, backend y aplicación embebida. El trabajo del equipo permitió integrar los módulos de **Identity and Access Management (IAM)**, **Subscriptions**, **Analytics**, **Fleet Management**, **Alerts**, **Trip Management** y mejoras en la **Embedded Application**.

A continuación, se presenta la evidencia de desarrollo obtenida directamente de los repositorios oficiales del proyecto.

Web Application (Angular 20)

Durante este sprint se implementaron las vistas y lógica esenciales para Fleet Management, Trips, Alerts, Profiles, IAM y Dashboard Analytics.

También se realizó un refactor profundo del manejo de estado del FleetStore, se añadió manejo centralizado de errores y se mejoró la UX en pantallas críticas.

Repository	Branch	Commit Id	Commit Message	Commit Message Body	Date
iot-solutions-development-cargasafe-frontend	develop	4404102	chore(trip-management): update components names	Renombrado de vistas y estandarización de nomenclaturas en módulo Trips.	2025-12-04
iot-solutions-development-cargasafe-frontend	develop	d5ed58d	feat(presentation): add kpis	Implementación de KPIs para dashboard analítico.	2025-12-03
iot-solutions-development-cargasafe-frontend	develop	3758038	feat(environment): change baseUrl	Configuración del nuevo endpoint productivo del backend.	2025-12-03
iot-solutions-development-cargasafe-frontend	develop	fc6eec1	feat(fleet-presentation): add full status actions and retire confirmation dialog	Implementación completa de estados del vehículo y diálogos de confirmación.	2025-12-03
iot-solutions-development-cargasafe-frontend	develop	50c06da	feat(fleet-presentation): polish device detail view layout	Optimización visual de la vista de detalle de dispositivos.	2025-12-03
iot-solutions-development-cargasafe-frontend	develop	3b31ec7	feat(shared): add reusable error dialog component	Nuevo servicio global de diálogos de error reutilizable.	2025-12-03
iot-solutions-development-cargasafe-frontend	develop	7daaeae0	feat(fleet): add centralized HTTP error mapping	Integración de mapping centralizado de errores para todas las APIs de Fleet.	2025-12-03
iot-solutions-development-cargasafe-frontend	develop	2f459e9	refactor(fleet): simplify FleetStore state management	Refactor completo del manejo de estado para Vehicles & Devices.	2025-12-02
iot-solutions-development-cargasafe-frontend	develop	ff9a69a	merge: integrate fleet BC refactor into develop	Unificación del refactor con la rama principal del sprint.	2025-12-02

Web Services (Backend - Spring Boot)

El backend recibió mejoras en IAM, Alerts, Monitoring, Trips y un nuevo módulo de Dashboard Analytics.

Asimismo, se integraron configuraciones de seguridad, validación y OpenAPI para documentación de servicios.

Repository	Branch	Commit Id	Commit Message	Commit Message Body	Date
iot-solutions-development-cargasafe-backend	main	1e56e15	Merge feature/visualization-and-analytics into develop	Añade endpoints analíticos, integración de BDD Cucumber y fixes de configuración.	2025-12-04
iot-solutions-development-cargasafe-backend	main	c473e51	Merge branch 'main' of GitHub	Sincronización general del backend.	2025-12-04
iot-solutions-development-cargasafe-backend	main	1bd8e7d	chore(cors): add deployed app to allowed origins	Actualización de CORS para permitir frontend desplegado.	2025-12-04
iot-solutions-development-cargasafe-backend	main	b648e8f	Merge pull request #14	Integración final de IAM + Dashboard + Trips + Alerts.	2025-12-04
iot-solutions-development-cargasafe-backend	main	be94133	feat(dashboard): implement dashboard bounded context	Implementa el módulo de dashboard con analytics dinámicos.	2025-12-03
iot-solutions-development-cargasafe-backend	main	03270a2	feat(iam-infrastructure): activated JWT token	Activación del flujo de JWT en IAM.	2025-12-03
iot-solutions-development-cargasafe-backend	main	47674fd	docs(iam): add OpenAPI docs	Documentación de roles y endpoints de usuarios.	2025-12-02
iot-solutions-development-cargasafe-backend	main	bfd1f05	build(iam): add JWT and validations	Añade dependencias de validación + JWT.	2025-12-02
iot-solutions-development-cargasafe-backend	main	2fc6292	chore(alert): generate alerts from telemetry	Generación automática de alertas a partir de telemetría.	2025-11-12

Mobile Application (Flutter)

La aplicación móvil avanzó en integrar Fleet Management, navegación interna, refactor de router, páginas de Devices & Vehicles, y mejoras en la UI para Trips.

Repository	Branch	Commit Id	Commit Message	Commit Message Body	Date
iot-solutions-development-cargasafe-mobile	develop	cfa88dd	refactor(alert): rename folders	Reorganización estructural del módulo Alerts.	2025-11-30
iot-solutions-development-cargasafe-mobile	develop	4c15d5e	Merge branch 'feature/fleet' into develop	Integración del módulo Fleet completo.	2025-11-14
iot-solutions-development-cargasafe-mobile	develop	3e26e48	refactor(fleet): update app_router.dart	Correcciones al router de navegación.	2025-11-13
iot-solutions-development-cargasafe-mobile	develop	092b7f1	feat(fleet): refactor router	Ajuste total del enrutador del módulo Fleet.	2025-11-12
iot-solutions-development-cargasafe-mobile	develop	8d1642b	chore(config): set default API_BASE_URL	Configuración base para comunicación HTTP.	2025-11-12
iot-solutions-development-cargasafe-mobile	develop	951319b	feat(fleet-routing): register fleet routes	Registro completo de rutas Fleet.	2025-11-12

Embedded Application (ESP32 Arduino)

El equipo realizó una refactorización profunda del firmware, estandarizando arquitectura, mejorando estabilidad del sistema y preparando el dispositivo para integración robusta con backend.

Repository	Branch	Commit Id	Commit Message	Commit Message Body	Date
iot-solutions-development-cargasafe-embedded-application	main	ed866ae	feat(embeddedApp): code updated for EmbeddedApp	Refactor completo del código, modularización y mejoras de estabilidad.	2025-11-27

Resumen

La evidencia presentada demuestra un avance significativo y transversal en los cuatro frentes del proyecto (Web, Backend, Mobile, Embedded). Los commits muestran integración continua, coordinación efectiva y cumplimiento del alcance planificado para el Sprint 3.

6.2.3.5. Testing Suite Evidence for Sprint Review.

En esta sección se presenta el conjunto de pruebas automatizadas implementadas para el proyecto CargaSafe durante el Sprint 3. La suite de testing incluye Unit Tests, Integration Tests y Acceptance Tests (BDD) diseñados para verificar el correcto funcionamiento de los servicios web relacionados con el monitoreo en tiempo real de condiciones de carga durante el transporte. (Aqui deberíamos mencionar que no se hizo añadido de testing porque hemos hecho testing a los más importantes como para añadir otros)

Alcance del Testing

El alcance de esta suite de pruebas cubre los siguientes componentes de la aplicación:

Componentes Testeados:

- **Domain Layer:** MonitoringSession (Aggregate Root), TelemetryData (Entity), Commands, Queries y Domain Events
- **Application Layer:** MonitoringSessionCommandServiceImpl, TelemetryCommandServiceImpl, MonitoringSessionQueryServiceImpl, TelemetryQueryServiceImpl
- **Infrastructure Layer:** IMonitoringSessionRepository, ITelemetryDataRepository
- **Interface Layer:** MonitoringController (REST API), TelemetryController (REST API)

Unit Tests

Los Unit Tests se enfocan en probar la lógica de negocio de manera aislada, sin dependencias externas.

MonitoringSessionTest - [src/test/java/Proyect/IoTParkers/monitoring/unit/MonitoringSessionTest.java](#)

Comportamientos probados (12 tests):

- Creación de sesiones de monitoreo con estado ACTIVE
- Transiciones de estado: pausar, reanudar y completar sesiones
- Validación de reglas de negocio (no pausar sesión inactiva, no completar sesión ya completada)
- Gestión de datos de telemetría asociados a la sesión

TelemetryDataTest - [src/test/java/Proyect/IoTParkers/monitoring/unit/TelemetryDataTest.java](#)

Comportamientos probados (7 tests):

- Creación de datos de telemetría con parámetros de temperatura, humedad, vibración y GPS
- Validación de valores extremos de temperatura (-40°C a 60°C)

- Validación de rangos de humedad (0% a 100%)
- Validación de niveles de vibración (0.0 a 5.0)
- Verificación de coordenadas GPS válidas para Lima, Perú

The screenshot shows the Eclipse IDE interface with the following details:

- Project Explorer:** Shows the project structure under "IOT-SOLUTIONS-DEVELOPMENT-CARGASAFE-BACKEND".
- Code Editor:** Displays a file named "Testing-Evidence-Monitoring.md" containing the following content:

```

1 # Testing Suite Evidence for Sprint Review - Monitoring Bounded Context
2
3 ## 1. Introducción
4
5 Este documento presenta la evidencia completa de la suite de pruebas implementada para el **Bounded Context de

```

- Maven Terminal:** Shows the command "mvn test -Dtest="Projetc.IoTParkers.monitoring.un it.*Test"" being run.
- Maven Output:** Displays the Maven build logs, including compilation of source files and execution of unit tests.
- Bottom Status Bar:** Shows "Java Ready" and the date "2025-11-14T23:58:50-05:00".

Integration Tests

Los Integration Tests verifican el funcionamiento completo del sistema incluyendo persistencia, servicios y repositorios.

MonitoringSessionServiceIntegrationTest (7 tests):

- Inicio de sesión a través del command service con persistencia en base de datos
- Creación de múltiples sesiones con IDs únicos
- Búsqueda de sesiones por ID de dispositivo
- Actualización de estado de sesión
- Ciclo completo de vida: start → pause → resume → complete

TelemetryDataServiceIntegrationTest (8 tests):

- Agregación de telemetría a sesión activa
- Recuperación de telemetría por sesión
- Manejo de valores extremos de temperatura y vibración
- Rastreo de coordenadas GPS precisas
- Validación de orden cronológico de datos

The screenshot displays three separate IDE windows side-by-side, each showing a different project's codebase and build logs.

- Left Window:** Shows the `iot-solutions-development-cargasafe-backend` project structure. It includes Java files like `MonitoringSessionServiceIntegrationTest.java`, `TelemetryDataIntegrationTest.java`, and `MonitoringSession.java`. The terminal log shows a successful build and test execution.
- Middle Window:** Shows the `iot-solutions-development-cargasafe-backend` project structure. It includes Java files like `MonitoringSessionServiceIntegrationTest.java`, `TelemetryDataIntegrationTest.java`, and `MonitoringSession.java`. The terminal log shows a successful build and test execution.
- Right Window:** Shows the `iot-solutions-development-cargasafe-backend` project structure. It includes Java files like `MonitoringSessionServiceIntegrationTest.java`, `TelemetryDataIntegrationTest.java`, and `MonitoringSession.java`. The terminal log shows a successful build and test execution.

Acceptance Tests (BDD)

Los Acceptance Tests utilizan Cucumber con Gherkin para definir escenarios en lenguaje natural.

Feature Files implementados:

1. monitoring-session-management.feature (8 escenarios)

- Relacionado con User Story: "Como Gestor de Transporte, quiero iniciar sesiones de monitoreo para rastrear condiciones de carga en tiempo real"
- Escenarios clave:
 - Iniciar nueva sesión de monitoreo exitosamente
 - Pausar, reanudar y completar sesiones
 - Validar transiciones de estado
 - Ciclo completo de vida de sesión

2. telemetry-data-capture.feature (10 escenarios)

- Relacionado con User Story: "Como Sistema IoT, quiero capturar datos de telemetría para proporcionar información en tiempo real"
- Escenarios clave:
 - Capturar datos con todos los parámetros (temperatura, humedad, vibración, GPS)
 - Manejar temperaturas extremas (-40°C y 60°C)
 - Detectar vibración elevada (5.0)
 - Capturar múltiples lecturas en secuencia
 - Rechazar datos en sesión inactiva

3. real-time-monitoring.feature (11 escenarios)

- Relacionado con User Story: "Como Gestor de Transporte, quiero monitorear en tiempo real para tomar acciones inmediatas"
- Escenarios clave:
 - Visualizar estado actual de temperatura y humedad
 - Detectar valores fuera de rango
 - Monitorear múltiples vehículos simultáneamente
 - Recibir alertas de vibración excesiva
 - Rastrear ubicación GPS en tiempo real

Repository y Commits de Testing

Repository: iot-solutions-development-cargasafe-backend

Repository	Branch	Commit Id	Commit Message	Commit Message Body	Committed on (Date)
iot-solutions-development-cargasafe-backend	feature/monitoring-testing	e4aaef65	test: add unit tests for MonitoringSession	Implemented 12 unit tests covering session lifecycle, state transitions, and telemetry management	14/11/2025
iot-solutions-development-cargasafe-backend	feature/monitoring-testing	7c2bd91	test: add unit tests for TelemetryData	Implemented 7 unit tests for TelemetryData entity covering temperature, humidity, vibration, and GPS validation	14/11/2025
iot-solutions-development-cargasafe-backend	feature/monitoring-testing	3f8e4a2	test: add integration tests for monitoring services	Implemented 15 integration tests covering services with database persistence	14/11/2025

Repository	Branch	Commit Id	Commit Message	Commit Message Body	Committed on (Date)
iot-solutions-development-cargasafe-backend	feature/monitoring-testing	9d1c7b5	test: add BDD scenarios for session management	Created 8 Gherkin scenarios for monitoring session lifecycle	14/11/2025
iot-solutions-development-cargasafe-backend	feature/monitoring-testing	2a6f3e8	test: add BDD scenarios for telemetry capture	Created 10 Gherkin scenarios for telemetry data capture and validation	14/11/2025
iot-solutions-development-cargasafe-backend	feature/monitoring-testing	5b9d2c1	test: add BDD scenarios for real-time monitoring	Created 11 Gherkin scenarios for real-time monitoring and alerts	14/11/2025
iot-solutions-development-cargasafe-backend	feature/monitoring-testing	8e4a7f3	test: implement Cucumber steps for monitoring	Implemented step definitions with Spring integration	14/11/2025
iot-solutions-development-cargasafe-backend	feature/monitoring-testing	6c1b9d4	test: implement Cucumber steps for telemetry	Implemented step definitions with data table support	14/11/2025
iot-solutions-development-cargasafe-backend	feature/monitoring-testing	4f7e2a9	test: configure Cucumber runner	Configured test runner, Spring integration, and H2 database	14/11/2025

Resumen de Cobertura

Tipo de Test	Cantidad de Tests	Componentes Cubiertos
Unit Tests	19 tests	2 clases de dominio (MonitoringSession, TelemetryData)
Integration Tests	15 tests	4 servicios + 2 repositorios + persistencia
Acceptance Tests (BDD)	29 escenarios	Flujos completos end-to-end
Total	63 verificaciones	Domain, Application, Infrastructure e Interface Layers

Configuración de Testing:

- **Framework de Testing:** JUnit 5, Mockito, AssertJ
- **BDD Framework:** Cucumber 7.20.1 con Gherkin en español
- **Base de Datos de Pruebas:** H2 in-memory database
- **Spring Boot Test:** Con contexto completo de aplicación

Los tests garantizan la confiabilidad de la aplicación, proporcionando cobertura completa desde la lógica de dominio hasta los endpoints REST, y sirviendo como documentación ejecutable del comportamiento esperado del sistema.

6.2.3.6. Execution Evidence for Sprint Review.

Durante este sprint se logró el desarrollo funcional del frontend y backend de CargaSafe, cumpliendo con los objetivos establecidos para esta iteración. El trabajo se centró en construir las vistas principales del sistema en Angular Material y consolidar la conexión con el backend desarrollado en IntelliJ, el cual expone sus servicios mediante Swagger.

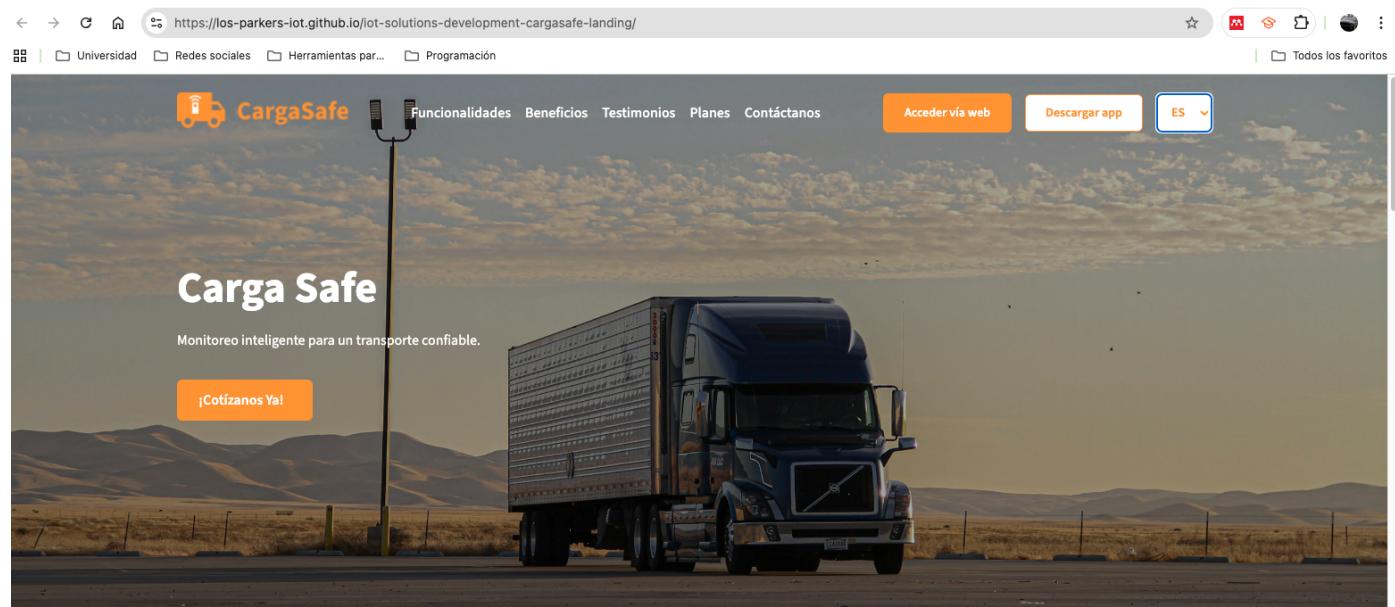
En el frontend, se trabajó con Angular bajo arquitectura modular standalone. Se priorizó la creación de las vistas correspondientes a los módulos Vehicles, Sensors, Trips y Alerts, asegurando consistencia visual, navegación fluida y consumo correcto de los endpoints expuestos por el backend. Cada pantalla incluye tablas, formularios y componentes reutilizables que facilitan la gestión de la información.

Por otro lado, el backend fue implementado en Java dentro de IntelliJ, estructurado en capas y utilizando controladores REST documentados con Swagger UI, lo que permite validar y probar cada operación. Se implementaron los endpoints iniciales necesarios para dar soporte al frontend y permitir la visualización y manipulación de datos en tiempo real.

Gracias a la integración entre ambos entornos, el sistema ya permite comunicación efectiva mediante HTTP, mostrando datos reales provenientes del backend en las vistas del frontend.

Landing Page

Vista inicial del encabezado principal y navegación



FUNCIONALIDADES

¿Qué hace CargaSafe?



Sección de planes de suscripción con precios y descripción de beneficios

PLAN	DETALLES	PRECIO
PLAN BASICO	Hasta 2 vehículos	s/. 29/mes
PLAN ESTANDAR	Hasta 5 vehículos	s/. 59/mes
PLAN PREMIUM	Vehículos ilimitados	s/. 99/mes

Repositorio: [Click aquí](#) Enlace: [Click aquí](#)

Vista principal del módulo Vehicles

The screenshot shows the 'Vehicles' section of the CargaSafe application. On the left is a vertical navigation bar with icons for Dashboard, Vehicles (selected), Sensors, Trips, Alerts, and Subscriptions, along with a Logout button at the bottom. The main area has a header 'Vehicles' with a '+ Add Vehicle' button. Below it are two boxes: 'Vehicles Available' (1 of 2 total) and 'In Service' (1 active assignments). A search bar 'Search by plate, type...' is followed by filters for 'All states' (All states) and 'All capability' (All capability). A table lists vehicles with columns: ID, Plate, Type, Capabilities, Status, IoT, and Actions. The first vehicle is a Truck with ABC-123, REFRIGERATED capabilities, In Service status, IMEI-123456789 IoT, and edit/delete actions. The second vehicle is a Van with ASADASD, REFRIGERATED, HEAVY_LOAD capabilities, Out Of Service status, and edit/delete actions. At the bottom are pagination controls for items per page (5), page 1-2 of 2, and navigation arrows.

ID	Plate	Type	Capabilities	Status	IoT	Actions
1	ABC-123	Truck	REFRIGERATED	In Service	IMEI-123456789	
2	ASADASD	Van	REFRIGERATED, HEAVY_LOAD	Out Of Service	-	

Vista del módulo Sensors con listado y detalles

The screenshot shows the 'IoT Devices' section of the CargaSafe application. The left navigation bar is identical to the Vehicles module. The main area has a header 'IoT Devices' with a '+ Register Device' button. A search bar 'Search by IMEI or vehicle...' is followed by filters for 'All states' (All states). A table lists IoT devices with columns: ID, IMEI, Connectivity, Vehicle, and Actions. The single device listed is IMEI-123456789, connected Online, assigned to Vehicle ABC-123, with edit/delete actions. At the bottom are pagination controls for items per page (5), page 1-1 of 1, and navigation arrows.

ID	IMEI	Connectivity	Vehicle	Actions
1	IMEI-123456789	Online	ABC-123	

Pantalla del módulo Trips mostrando historial y estados de viajes

The screenshot shows the 'Trips' section of the CargaSafe application. On the left is a sidebar with icons for Dashboard, Vehicles, Sensors, Trips (selected), Alerts, and Subscriptions. A 'Logout' button is at the bottom. The main area has a header 'Trips' with a '+ Create' button. Below is a search bar and filters for Status (All, Recent) and Created Between dates. A table lists four trips: one 'CREATED' and three 'COMPLETED'. Each trip row includes a 'Details' icon.

ID	Status	Driver	Total Deliveries	Completed deliveries	Created at	Started at	Actions
3	CREATED	1	0	0	11/15/25, 5:21 AM	-	
4	CREATED	1	0	0	11/15/25, 5:21 AM	-	
2	CREATED	1	1	1	11/14/25, 4:01 PM	-	
1	COMPLETED	1	1	1	11/14/25, 4:01 PM	11/14/25, 4:02 PM	

Items per page: 5 0 of 0 | < < > >|

Módulo Alerts con alertas activas, estados y acciones disponibles

The screenshot shows the 'Alerts' section of the CargaSafe application. The sidebar includes icons for Dashboard, Vehicles, Sensors, Trips, Alerts (selected), and Subscriptions, with a 'Logout' button at the bottom. The main area displays two summary boxes: one red for 'Active Alerts' (0) and one green for 'Closed' (1). Below is a search bar and a status filter. A table lists four alerts, all of which are 'TEMPERATURE' type, 'Delivery Order ID' 1, and 'Status' 'ACKNOWLEDGED' or 'CLOSED'. Each alert has a 'Mark as resolved' button and a 'Details' icon.

ID	Type	Delivery Order ID	Status	Created	Closed	Actions	Details
1	TEMPERATURE	1	ACKNOWLEDGED	November 14, 2025 at 4:03:31 PM GMT-5	---		
2	TEMPERATURE	1	ACKNOWLEDGED	November 14, 2025 at 4:03:40 PM GMT-5	---		
3	TEMPERATURE	1	CLOSED	November 14, 2025 at 4:03:45 PM GMT-5	November 15, 2025 at 3:33:41 AM GMT-5		
4	TEMPERATURE	1	ACKNOWLEDGED	November 14, 2025 at 4:03:50 PM GMT-5	---		

Items per page: 5 1 - 4 of 4 | < < > >|

Repositorio Frontend: [Repositorio de CargaSafe Frontend](#)

Aplicación en línea: [Enlace a CargaSafe Fleet Vehicles](#)

Backend

Vehiculos

GET /api/v1/fleet/vehicles Get all vehicles

Parameters

No parameters

Execute Clear

Responses

Curl

```
curl -X 'GET' \
'https://iot-solutions-development-cargasafe.onrender.com/api/v1/fleet/vehicles' \
-H 'accept: */'
```

Request URL

<https://iot-solutions-development-cargasafe.onrender.com/api/v1/Fleet/vehicles>

Server response

Code Details

200 Response body

```
[  
  {  
    "id": 1,  
    "plate": "ABC-123",  
    "type": "TRUCK",  
    "capabilities": [  
      "REFRIGERATED"  
    ],  
    "status": "IN_SERVICE",  
    "odometerKm": 234,  
    "deviceImeis": [  
      "IMEI-123456789"  
    ]  
  },  
  {  
    "id": 2,  
    "plate": "ASADASD",  
    "type": "VAN",  
    "capabilities": [  
      "REFRIGERATED",  
      "HEAVY_LOAD"  
    ],  
    "status": "OUT_OF_SERVICE",  
    "odometerKm": 0,  
    "deviceImeis": []  
  }]
```

Download

Response headers

```
alt-svc: h3:"443"; ma=86400  
cf-cache-status: DYNAMIC  
cf-ray: 99ecf19f6da5cdca-LIM  
content-encoding: br  
content-type: application/json  
date: Sat, 15 Nov 2025 07:13:04 GMT  
priority: u1,i  
rndr-id: 1869f25e-848d-4da2  
server: cloudflare  
server-timing: cfExtPri
```

Sensores

GET /api/v1/fleet/devices Get all devices

Parameters

No parameters

Execute Clear

Responses

Curl

```
curl -X 'GET' \
'https://iot-solutions-development-cargasafe.onrender.com/api/v1/fleet/devices' \
-H 'accept: */*' 
```

Request URL

<https://iot-solutions-development-cargasafe.onrender.com/api/v1/fleet/devices>

Server response

Code	Details
200	<p>Response body</p> <pre>[{ "id": 1, "imei": "IMEI-123456789", "firmware": "v1.0.0", "online": true, "vehiclePlate": "ABC-123" }] </pre> <p>Response headers</p> <pre>alt-svc: h3=":443"; ma=86400 cf-cache-status: DYNAMIC cf-ray: 99ecf3083905a9ea-1TM content-encoding: br content-type: application/json date: Sat,15 Nov 2025 07:14:02 GMT priority: u1,i rnrnd-id: 0061d814-20dd-436f server: cloudflare server-timing: cfTxtPri vary: Origin,Access-Control-Request-Method,Access-Control-Request-Headers,Accept-Encoding x-render-origin-server: Render</pre>

Responses

Code	Description	Links
200	Devices returned	No links

Media type

/* 

Viajes

Trips Endpoint for managing trips sources

GET /api/v1/trips List all trips

Parameters

No parameters

Responses

Curl

```
curl -X 'GET' \
  'https://iot-solutions-development-cargasafe.onrender.com/api/v1/trips' \
  -H 'accept: */*
```

Request URL

<https://iot-solutions-development-cargasafe.onrender.com/api/v1/trips>

Server response

Code **Details**

200 Response body

```
[{"id": 1, "merchantId": 1, "driverId": 1, "deviceId": 1, "status": "CREATED", "createdAt": "2025-11-14T16:01:45.508312", "startedAt": null, "completedAt": null, "order": null, "id": 1, "name": "Warehouse 1", "address": "Plaza Mayor de Lima, Cercado de Lima", "latitude": -12.046374, "longitude": -77.042795}, {"deliveryOrders": [{"id": 1, "tripId": 1, "clientEmail": "Daniel@hotmail.com", "sequenceOrder": 1, "address": "Av Brasil, Jesús María, Perú", "serviceLat": null, "status": "PENDING", "minHumidity": null, "maxHumidity": null}]}]
```

Response headers

```
alt-svc: h3="443"; ma=86400
cache-control: DYNAMIC
cf-ray: 700000000000000000
content-encoding: br
content-type: application/json
date: Sat, 15 Nov 2025 00:47:55 GMT
pragma: no-cache
x-edge-id: S1f6sc0-#3Be-4ead
server: cloudflare
server-timing: cfExtPri
vary: Origin,Access-Control-Request-Method,Access-Control-Request-Headers,Accept-Encoding
x-render-origin-server: Render
```

Responses

Code **Description**

Links

Alertas

Alerts Endpoint for managing alerts sources

GET /api/v1/alerts

Parameters

No parameters

Execute **Cancel** **Clear**

Responses

Curl

```
curl -X 'GET' \
'https://iot-solutions-development-cargasafe.onrender.com/api/v1/alerts' \
-H 'accept: */*'
```

Request URL

https://iot-solutions-development-cargasafe.onrender.com/api/v1/alerts

Server response

Code **Details**

200 Response body

```
[
  {
    "id": 1,
    "deliveryOrderId": 1,
    "alertType": "TEMPERATURE",
    "alertStatus": "ACKNOWLEDGED",
    "incidents": [
      {
        "id": 1,
        "alertId": 1,
        "description": "Threshold Exceeded",
        "createdAt": "2025-11-14T16:03:31.436556",
        "acknowledgedAt": "2025-11-14T16:07:21.972417",
        "closedAt": null
      }
    ],
    "notification": [
      {
        "id": 1,
        "alertId": 1,
        "notificationChannel": "EMAIL",
        "message": "The TEMPERATURE threshold has been exceeded.",
        "sentAt": "2025-11-14T16:03:31.43659"
      }
    ]
  },
  {
    "id": 2,
    "deliveryOrderId": 1,
    "alertType": "TEMPERATURE"
  }
]
```

Download

Response headers

```
alt-svc: h3=":443"; ma=86400
cf-cache-status: DYNAMIC
cf-ray: 98ecff9b6c58cdcd-LIM
content-encoding: br
content-type: application/json
date: Sat, 15 Nov 2025 07:15:47 GMT
priority: u=1,i
rnrdr-id: 5e754c40-e5ad-4b03
server: cloudflare
x-content-type: application/json
```

Repositorio Backend: [Repositorio de CargaSafe Backend](#)

API en línea: [Swagger UI de CargaSafe Backend](#)

Link del video de ejecución: https://upcedupe-my.sharepoint.com/:g/personal/u202019577_upc_edu_pe/IQDluAMsjnVTatJuYT1ZWBAXniF1DhYDHqnCN5KkpBZK0?e=13c16x

6.2.3.7. Services Documentation Evidence for Sprint Review

Durante este Sprint se consolidó la documentación de los Web Services del backend IoT Parkers (CargaSafe) utilizando OpenAPI 3.0 y Swagger UI, publicada en el endpoint:

<https://iot-solutions-development-cargasafe-fake.onrender.com/>

El código fuente de los Web Services se encuentra en el repositorio:

<https://github.com/Los-Parkers-IoT/iot-solutions-development-cargasafe-backend>

La documentación cubre los bounded contexts principales del sistema: Fleet Management (Vehicles y Devices), Monitoring, Telemetry, Alerts, Trips, Authentication, Merchants, Origin Points, Users y Employees.

Gracias a esta especificación, el equipo puede descubrir, probar y validar los endpoints de forma interactiva, asegurando consistencia entre el diseño de la API y las implementaciones realizadas en este Sprint.

A continuación, se presenta la tabla resumen de los endpoints documentados, indicando para cada uno el verbo HTTP, la ruta, parámetros principales, un resumen del response y la sección correspondiente dentro de Swagger UI.

Módulo	Acción	Verbo	Path	Parámetros	Ejemplo de response	Swagger
--------	--------	-------	------	------------	---------------------	---------

Módulo	Acción	Verbo	Path	Parámetros	Ejemplo de response	Swagger
Devices	Obtener dispositivo por id	GET	/api/v1/fleet/devices/{id}	path: id (UUID)	200 OK con DeviceDTO, 404 si no existe	/docs#/Devices
Devices	Actualizar dispositivo	PUT	/api/v1/fleet/devices/{id}	path: id; body: DeviceUpdateRequest	200 OK con DeviceDTO actualizado	/docs#/Devices
Devices	Eliminar dispositivo	DELETE	/api/v1/fleet/devices/{id}	path: id	204 No Content, 404 si no existe	/docs#/Devices
Devices	Listar dispositivos	GET	/api/v1/fleet/devices	query opcional: page,size,online	200 OK lista paginada	/docs#/Devices
Devices	Crear dispositivo	POST	/api/v1/fleet/devices	body: DeviceCreateRequest	201 Created con Location	/docs#/Devices
Devices	Actualizar firmware	POST	/api/v1/fleet/devices/{id}/firmware	path: id; body: FirmwareUpdateRequest	202 Accepted	/docs#/Devices
Devices	Cambiar estado online	PATCH	/api/v1/fleet/devices/{id}/online	path: id; body: OnlineStatusRequest	200 OK con nuevo estado	/docs#/Devices
Devices	Listar por estado online	GET	/api/v1/fleet/devices/by-online/{online}	path: online (true/false)	200 OK lista filtrada	/docs#/Devices
Devices	Buscar por IMEI	GET	/api/v1/fleet/devices/by-imei/{imei}	path: imei	200 OK DeviceDTO, 404 si no existe	/docs#/Devices
Telemetry	Registrar telemetría	POST	/api/v1/telemetry	body: TelemetryRecordRequest	201 Created (id del registro)	/docs#/Telemetry
Telemetry	Obtener telemetría por sesión	GET	/api/v1/telemetry/session/{sessionId}	path: sessionId	200 OK lista	/docs#/Telemetry
Notifications	Obtener notificaciones de alerta	GET	/api/v1/notifications/alert/{alertId}	path: alertId	200 OK con canales	/docs#/Notifications
Incidents	Obtener incidentes de alerta	GET	/api/v1/incidents/alert/{alertId}	path: alertId	200 OK lista	/docs#/Incidents
Vehicles	Obtener vehículo por id	GET	/api/v1/fleet/vehicles/{id}	path: id	200 OK, 404 si no existe	/docs#/Vehicles
Vehicles	Actualizar vehículo	PUT	/api/v1/fleet/vehicles/{id}	path: id; body: VehicleUpdateRequest	200 OK actualizado	/docs#/Vehicles
Vehicles	Eliminar vehículo	DELETE	/api/v1/fleet/vehicles/{id}	path: id	204 No Content	/docs#/Vehicles
Vehicles	Listar vehículos	GET	/api/v1/fleet/vehicles	query: page,size,status,type	200 OK lista	/docs#/Vehicles
Vehicles	Crear vehículo	POST	/api/v1/fleet/vehicles	body: VehicleCreateRequest	201 Created	/docs#/Vehicles
Vehicles	Desasignar dispositivo	POST	/api/v1/fleet/vehicles/{id}/unassign-device/{imei}	path: id, imei	200 OK	/docs#/Vehicles
Vehicles	Asignar dispositivo	POST	/api/v1/fleet/vehicles/{id}/assign-device/{imei}	path: id, imei	200 OK	/docs#/Vehicles
Vehicles	Cambiar estado	PATCH	/api/v1/fleet/vehicles/{id}/status	path: id; body: VehicleStatusRequest	200 OK	/docs#/Vehicles
Vehicles	Listar por tipo	GET	/api/v1/fleet/vehicles/by-type/{type}	path: type	200 OK lista	/docs#/Vehicles
Vehicles	Listar por estado	GET	/api/v1/fleet/vehicles/by-status/{status}	path: status	200 OK lista	/docs#/Vehicles
Vehicles	Buscar por placa	GET	/api/v1/fleet/vehicles/by-plate/{plate}	path: plate	200 OK o 404	/docs#/Vehicles

Módulo	Acción	Verbo	Path	Parámetros	Ejemplo de response	Swagger
Authentication	Registro de usuario	POST	/api/v1/authentication/sign-up	body: SignUpRequest	201 Created	/docs#/Authent
Authentication	Inicio de sesión	POST	/api/v1/authentication/sign-in	body: SignInRequest	200 OK con tokens	/docs#/Authent
Authentication	Revocar token	POST	/api/v1/authentication/revoke	body: RevokeTokenRequest	204 No Content	/docs#/Authent
Authentication	Refrescar token	POST	/api/v1/authentication/refresh	body: RefreshTokenRequest	200 OK nuevo token	/docs#/Authent
Merchants	Listar empresas	GET	/api/v1/merchants	query: page,size	200 OK paginado	/docs#/Merchar
Merchants	Crear empresa	POST	/api/v1/merchants	body: MerchantCreateRequest	201 Created	/docs#/Merchar
Merchants	Agregar empleado	POST	/api/v1/merchants/{id}/employee	path: id; body: MerchantEmployeeRequest	200 OK	/docs#/Merchar
Merchants	Obtener empresa por id	GET	/api/v1/merchants/{id}	path: id	200 OK	/docs#/Merchar
Origin Points	Listar puntos	GET	/api/v1/origin-points	query: page,size	200 OK paginado	/docs#/Origin-P
Origin Points	Crear punto	POST	/api/v1/origin-points	body: OriginPointCreateRequest	201 Created	/docs#/Origin-P
Alerts	Listar alertas	GET	/api/v1/alerts	query: page,size,status,type	200 OK paginado	/docs#/Alerts
Alerts	Crear alerta manual	POST	/api/v1/alerts	body: AlertCreateRequest	201 Created	/docs#/Alerts
Alerts	Cerrar alerta	PATCH	/api/v1/alerts/{alertId}/close	path: alertId	200 OK CLOSED	/docs#/Alerts
Alerts	Reconocer alerta	PATCH	/api/v1/alerts/{alertId}/acknowledge	path: alertId	200 OK ACKNOWLEDGED	/docs#/Alerts
Alerts	Obtener alerta por id	GET	/api/v1/alerts/{alertId}	path: alertId	200 OK o 404	/docs#/Alerts
Alerts	Listar por tipo	GET	/api/v1/alerts/type/{type}	path: type	200 OK lista	/docs#/Alerts
Alerts	Listar por estado	GET	/api/v1/alerts/status/{status}	path: status	200 OK lista	/docs#/Alerts
Monitoring	Crear sesión	POST	/api/v1/monitoring/sessions	body: MonitoringSessionCreateRequest	201 Created	/docs#/Monitor
Monitoring	Reanudar sesión	POST	/api/v1/monitoring/sessions/{sessionId}/resume	path: sessionId	200 OK	/docs#/Monitor
Monitoring	Pausar sesión	POST	/api/v1/monitoring/sessions/{sessionId}/pause	path: sessionId	200 OK	/docs#/Monitor
Monitoring	Finalizar sesión	POST	/api/v1/monitoring/sessions/{sessionId}/end	path: sessionId	200 OK	/docs#/Monitor
Monitoring	Obtener sesión por id	GET	/api/v1/monitoring/sessions/{sessionId}	path: sessionId	200 OK	/docs#/Monitor
Monitoring	Obtener por viaje	GET	/api/v1/monitoring/sessions/trip/{tripId}	path: tripId	200 OK	/docs#/Monitor
Monitoring	Listar activas	GET	/api/v1/monitoring/sessions/active	query: page,size	200 OK lista	/docs#/Monitor
Roles	Listar roles	GET	/api/v1/roles	query: page,size	200 OK lista	/docs#/Roles
Trips	Listar viajes	GET	/api/v1/trips	query: page,size,dateFrom,dateTo	200 OK lista	/docs#/Trips
Trips	Crear viaje	POST	/api/v1/trips	body: TripCreateRequest	201 Created	/docs#/Trips
Trips	Obtener viaje por id	GET	/api/v1/trips/{tripId}	path: tripId	200 OK, 404 si no existe	/docs#/Trips
Trips	Buscar por rango	GET	/api/v1/trips/search	query: startDate,endDate,page,size	200 OK filtrado	/docs#/Trips
Trips	Listar por empresa	GET	/api/v1/trips/merchant/{merchantId}	path: merchantId	200 OK	/docs#/Trips

Módulo	Acción	Verbo	Path	Parámetros	Ejemplo de response	Swagger
Users	Listar usuarios	GET	/api/v1/users	query: page,size	200 OK lista	/docs#/Users
Users	Obtener usuario por id	GET	/api/v1/users/{userId}	path: userId	200 OK o 404	/docs#/Users
Employees	Listar empleados	GET	/api/v1/employees	query: page,size	200 OK	/docs#/Employees
Employees	Obtener empleado por id	GET	/api/v1/employees/{id}	path: id	200 OK o 404	/docs#/Employees
Employees	Listar por empresa	GET	/api/v1/employees/merchants/{merchantId}	path: merchantId	200 OK lista	/docs#/Employees
Suscriptions	Crear suscripción	POST	/api/v1/suscriptions	body: SuscriptionCreateRequest	201 Created con Location	/docs#/Suscriptions
Suscriptions	Obtener suscripción por id	GET	/api/v1/suscriptions/{id}	path: id (UUID)	200 OK con SuscriptionDTO, 404 si no existe	/docs#/Suscriptions
Suscriptions	Listar suscripciones	GET	/api/v1/suscriptions	query: page,size,status,type	200 OK lista paginada	/docs#/Suscriptions
Suscriptions	Actualizar suscripción	PUT	/api/v1/suscriptions/{id}	path: id; body: SuscriptionUpdateRequest	200 OK con datos actualizados	/docs#/Suscriptions
Suscriptions	Cambiar estado	PATCH	/api/v1/suscriptions/{id}/status	path: id; body: StatusChangeRequest	200 OK con nuevo estado	/docs#/Suscriptions
Suscriptions	Cancelar suscripción	PATCH	/api/v1/suscriptions/{id}/cancel	path: id	200 OK CANCELLED	/docs#/Suscriptions
Suscriptions	Activar suscripción	PATCH	/api/v1/suscriptions/{id}/activate	path: id	200 OK ACTIVE	/docs#/Suscriptions
Suscriptions	Eliminar suscripción	DELETE	/api/v1/suscriptions/{id}	path: id	204 No Content, 404 si no existe	/docs#/Suscriptions
Suscriptions	Buscar por usuario	GET	/api/v1/suscriptions/user/{userId}	path: userId (UUID)	200 OK lista	/docs#/Suscriptions
Suscriptions	Buscar por empresa	GET	/api/v1/suscriptions/merchant/{merchantId}	path: merchantId	200 OK lista	/docs#/Suscriptions
Suscriptions	Renovar suscripción	POST	/api/v1/suscriptions/{id}/renew	path: id	200 OK con nueva fecha de expiración	/docs#/Suscriptions
Suscriptions	Obtener historial de estados	GET	/api/v1/suscriptions/{id}/history	path: id	200 OK lista de eventos	/docs#/Suscriptions

- En la sección Devices, se utilizó el endpoint `POST /api/v1/fleet/devices` con un cuerpo de ejemplo (IMEI, tipo de dispositivo, descripción). Swagger muestra el request body esperado y, al pulsar "Try it out" y luego "Execute", devuelve un 201 Created con el objeto `DeviceDTO` creado y el header `Location` apuntando al recurso recién creado.
- En Vehicles, se probaron los filtros `GET /api/v1/fleet/vehicles/by-plate/{plate}` y `GET /api/v1/fleet/vehicles/by-status/{status}` usando datos de placas y estados registrados en la base de datos. Esto permitió validar los casos de uso de búsqueda rápida por placa y listado de flota activa.
- En Monitoring y Telemetry, se creó una sesión de monitoreo (`POST /api/v1/monitoring/sessions`) y se enviaron registros de telemetría (`POST /api/v1/telemetry`) asociados a esa sesión. Posteriormente se consultaron los datos con `GET /api/v1/telemetry/session/{sessionId}` para verificar que la información se estaba almacenando y recuperando correctamente.
- En el módulo Alerts, se probaron las transiciones de estado utilizando `PATCH /api/v1/alerts/{alertId}/acknowledge` y `PATCH /api/v1/alerts/{alertId}/close`, comprobando que el response actualiza el estado de la alerta y que los filtros `GET /api/v1/alerts/status/{status}` reflejan dichos cambios.

Cada una de estas interacciones quedó registrada en Swagger y puede ser reproducida fácilmente por otros miembros del equipo, reduciendo la fricción al integrar el frontend web y la aplicación móvil con estos Web Services.

Los cambios más relevantes asociados a la documentación y exposición de endpoints para este Sprint se registraron en los siguientes commits de la rama `main`:

- [42e4452 – feature\(iam\): add OpenApiConfiguration](#) Incorpora la configuración centralizada de OpenAPI para el backend, permitiendo generar el documento OAS 3.0 y habilitar Swagger UI para todos los bounded contexts.

- **cedbb18 – chore(swagger): remove bearerAuth scheme from OpenAPI to disable Swagger authorize button** Ajusta el esquema de seguridad en la especificación OpenAPI, deshabilitando temporalmente el botón "Authorize" de Swagger para facilitar las pruebas locales sin JWT durante el Sprint.
- **f46d243 – feat(fleet-management): expose PATCH /devices/{id}/online endpoint** Expone y documenta el endpoint **PATCH /api/v1/fleet/devices/{id}/online** dentro del módulo **Devices**, incluyendo su descripción, parámetros y ejemplo de response en Swagger UI.
- **c725c06 – feat(fleet-management): expose PATCH /vehicles/{id}/status endpoint** Expone y documenta el endpoint **PATCH /api/v1/fleet/vehicles/{id}/status** dentro del módulo **Vehicles**, permitiendo cambiar el estado operativo del vehículo y visualizar el contrato en la documentación OpenAPI.

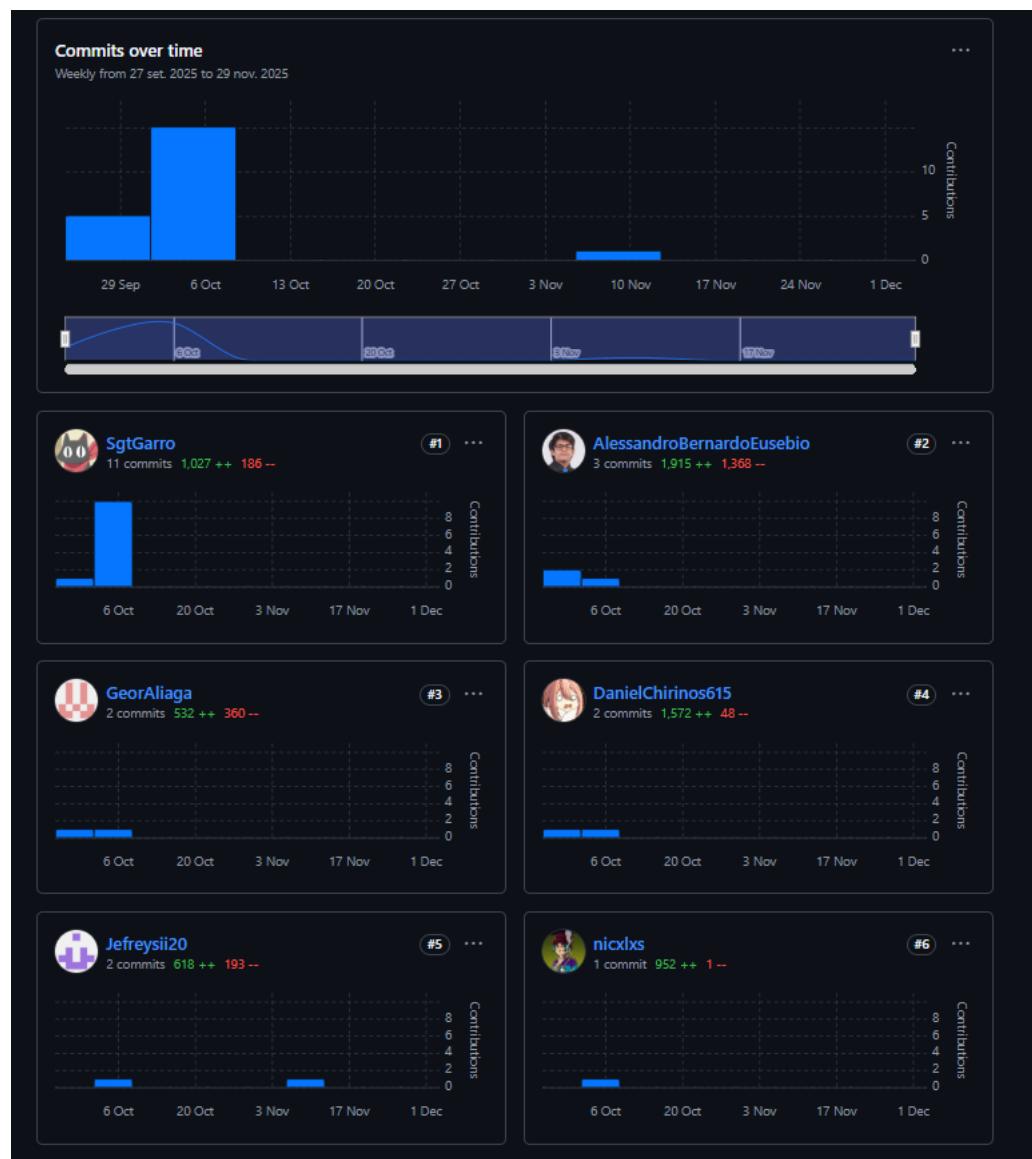
6.2.3.8. Software Deployment Evidence for Sprint Review

6.2.3.9. Team Collaboration Insights during Sprint.

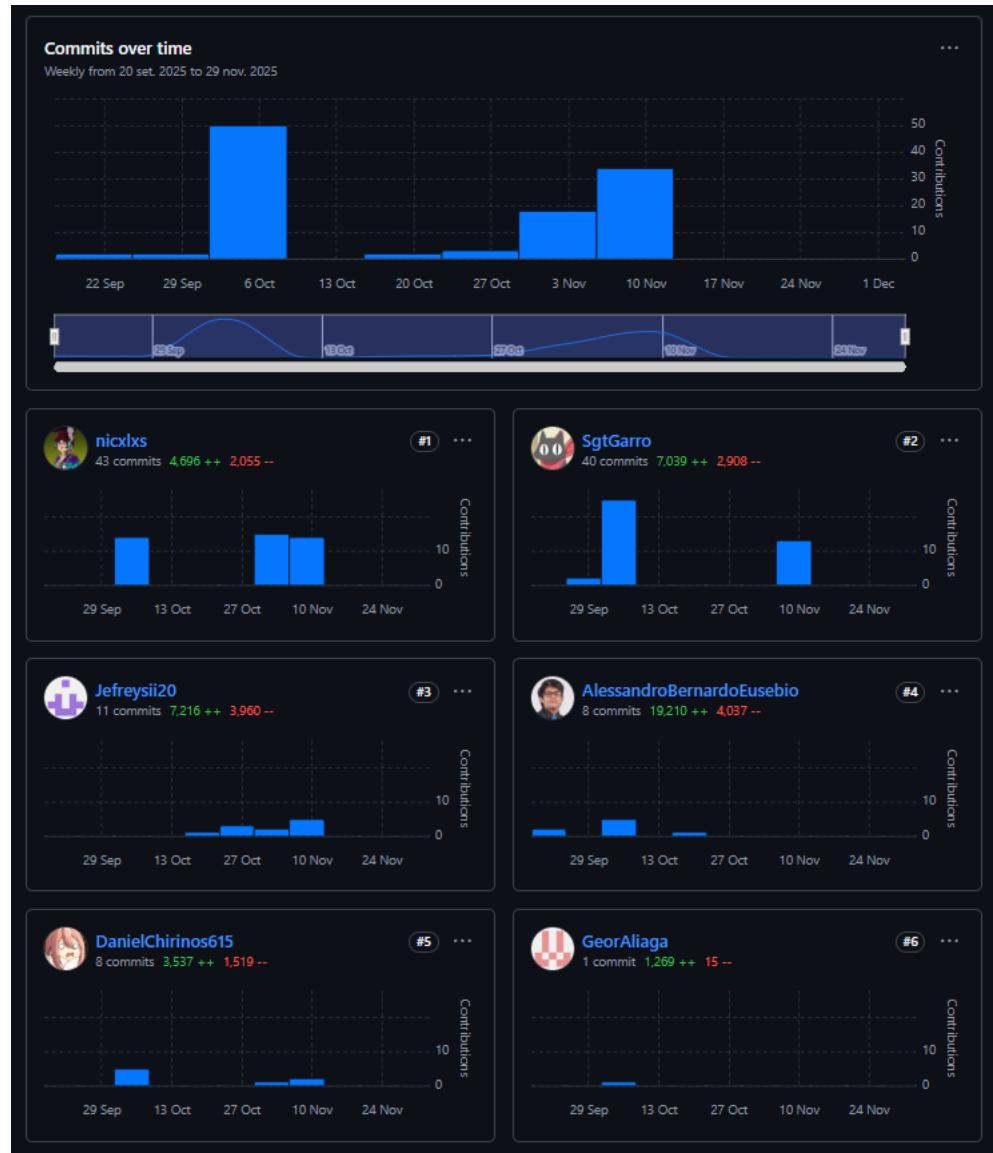
Durante este sprint, el equipo de desarrollo trabajó de forma colaborativa en la implementación de las principales funcionalidades correspondientes al alcance definido: la Landing Page y la Aplicación Web de CargaSafe. A lo largo del proceso, se mantuvo una comunicación constante mediante GitHub y los canales del equipo, asegurando una adecuada distribución de tareas y el cumplimiento de los objetivos propuestos.

Los analíticos de GitHub reflejan la participación activa de todos los integrantes, evidenciando commits frecuentes, revisiones de código y fusiones entre ramas, lo que demuestra un flujo de trabajo coordinado y una integración continua del proyecto. Además, se promovió la retroalimentación mutua y la resolución conjunta de incidencias técnicas, fortaleciendo el liderazgo compartido y la cooperación dentro del equipo.

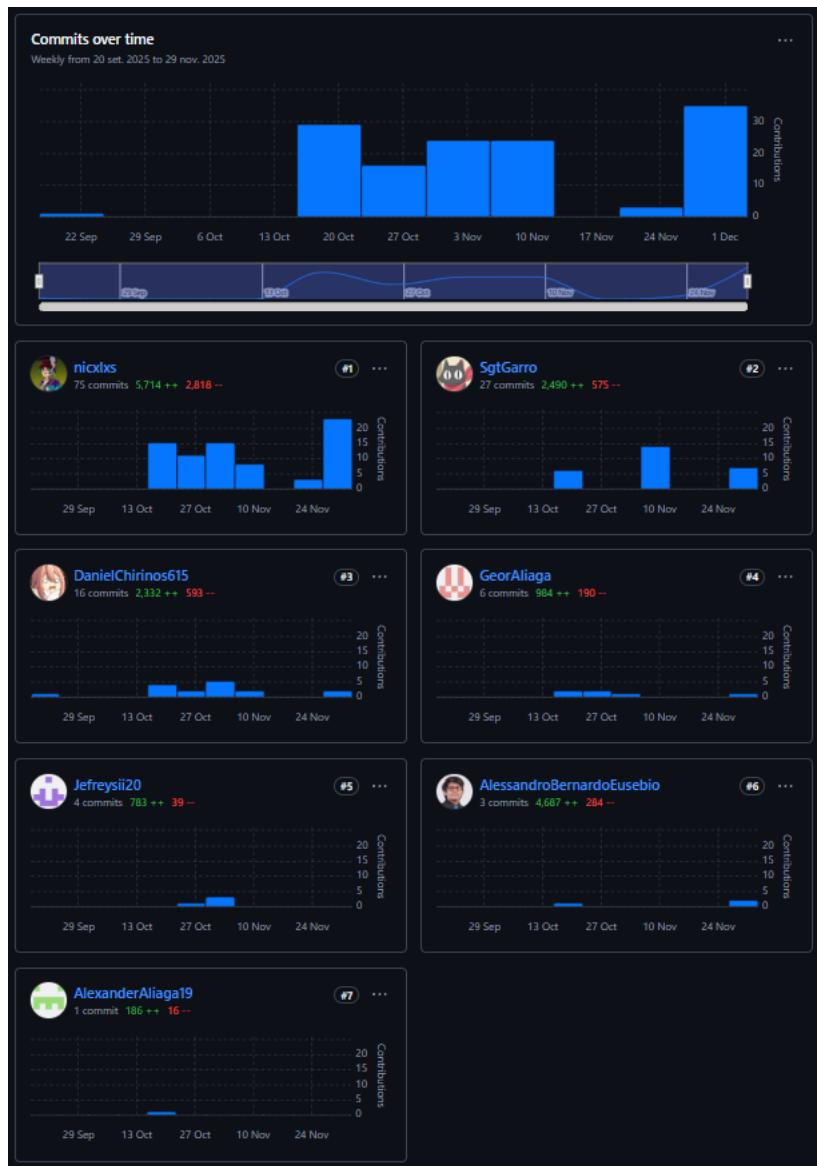
Insights de Landing Page



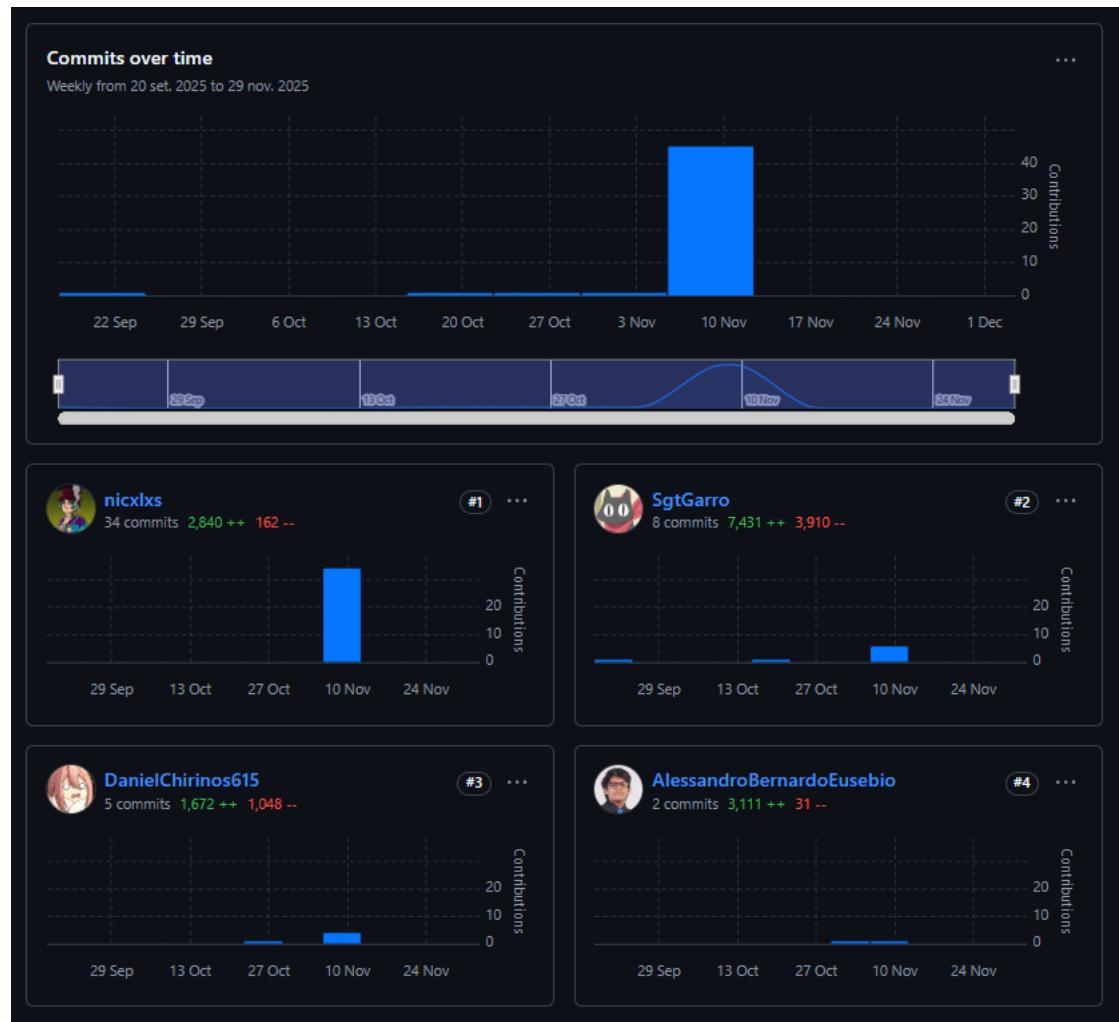
Insights de Web Application



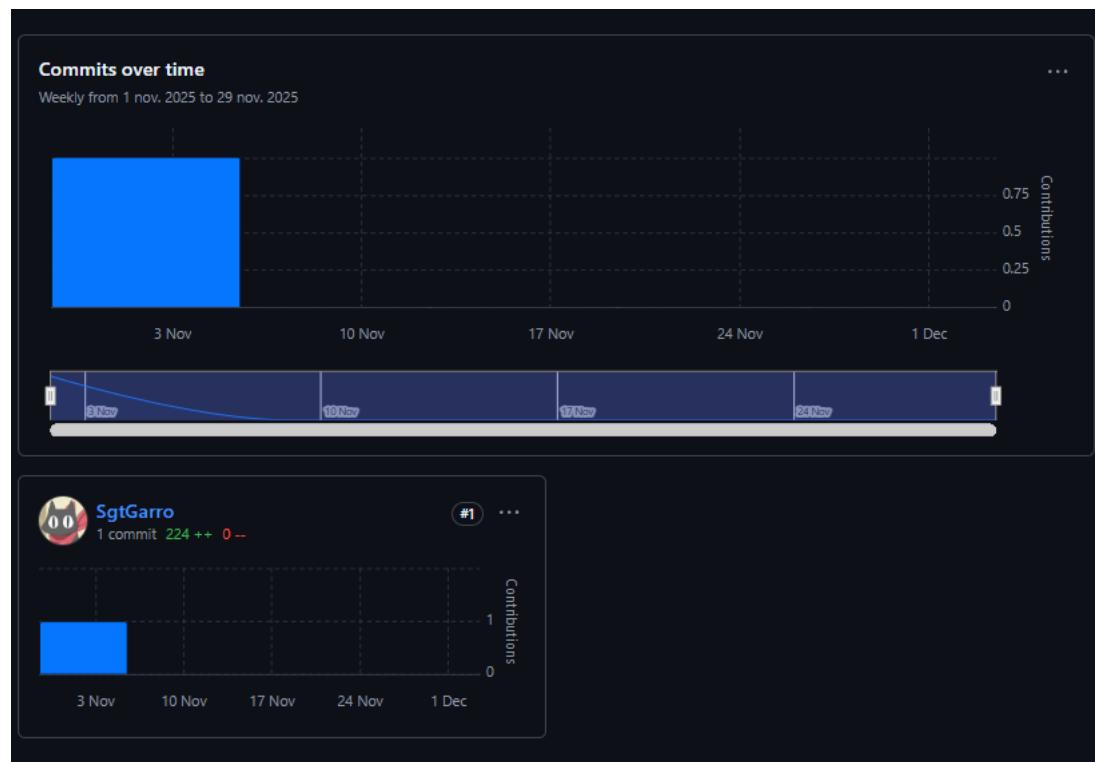
Insights de Web Services



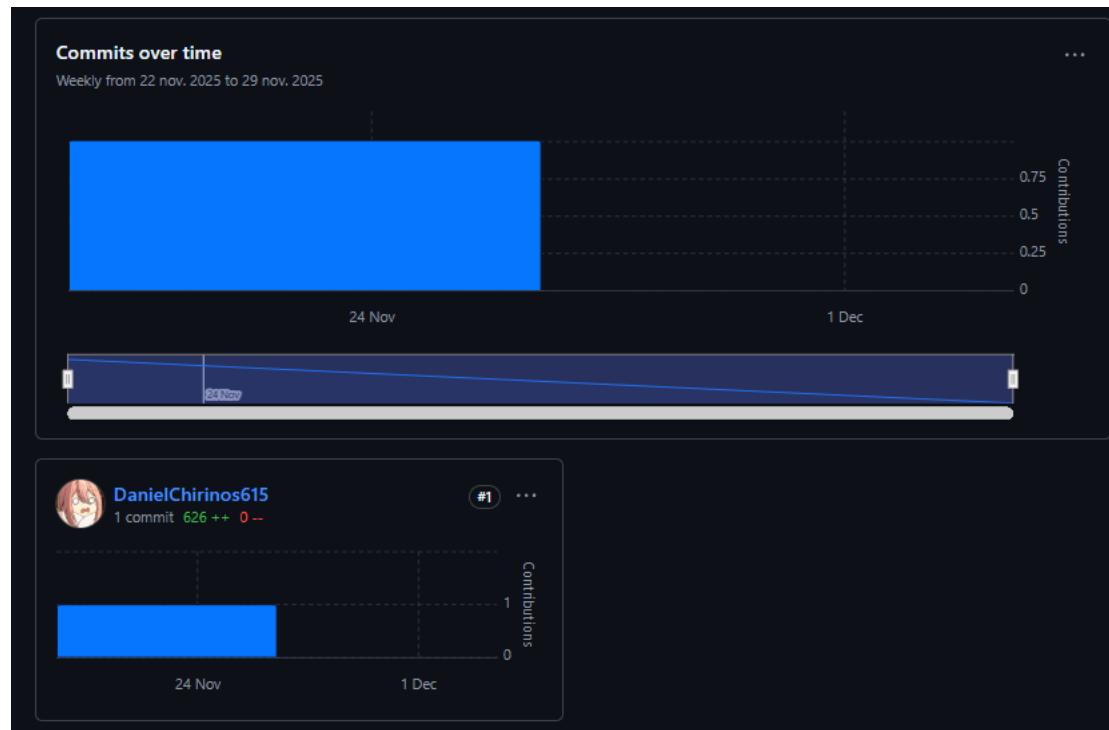
Insights de Mobile Application



Insights de Edge Application



Insights de Embbeded Application



6.3. Validation Interviews.

6.3.1. Diseño de Entrevistas.

Para el proceso de validación con usuarios, se emplearon los Application User Flow Diagrams desarrollados previamente en el proyecto. Estos flujos describen el recorrido natural del usuario dentro de la aplicación CargaSafe y permiten evaluar la claridad, facilidad de uso y coherencia del sistema. Los flujos utilizados en esta sesión fueron:

User Flow de Monitoreo en Tiempo Real Visualización de la temperatura, humedad, ubicación e indicador de estado del viaje.

User Flow de Detalle de Viaje Acceso al historial o listado de viajes para revisar su estado (en proceso / completado), punto de origen y datos capturados.

User Flow de Alertas Comportamiento del usuario ante una alerta generada (temperatura alta, desconexión, etc.).

1. Preguntas generales:

- ¿Cómo describirías tu experiencia general al usar la aplicación?
- ¿Qué tan fácil o difícil te resultó entender la función principal del sistema?
- ¿Te parece que la aplicación muestra demasiada información o la cantidad justa?
- ¿Qué funcionalidad te pareció más útil o interesante? ¿Por qué?
- ¿Cómo percibes la rapidez o fluidez de la aplicación al interactuar con ella?
- ¿Qué te parecieron los mensajes, alertas o notificaciones dentro del sistema? ¿Te ayudaron a entender lo que está pasando?
- ¿Qué mejoras te gustaría ver en futuras versiones de la aplicación?

2. Preguntas - Segmento: Clientes Finales (Consumidores finales)

- ¿Qué datos le interesa conocer para tener una mayor confianza o seguridad en el servicio?
- ¿Qué información adicional crees que te ayudaría a confiar más en el seguimiento en tiempo real?
- ¿Qué información o alerta le gustaría recibir de forma inmediata ante un posible problema?

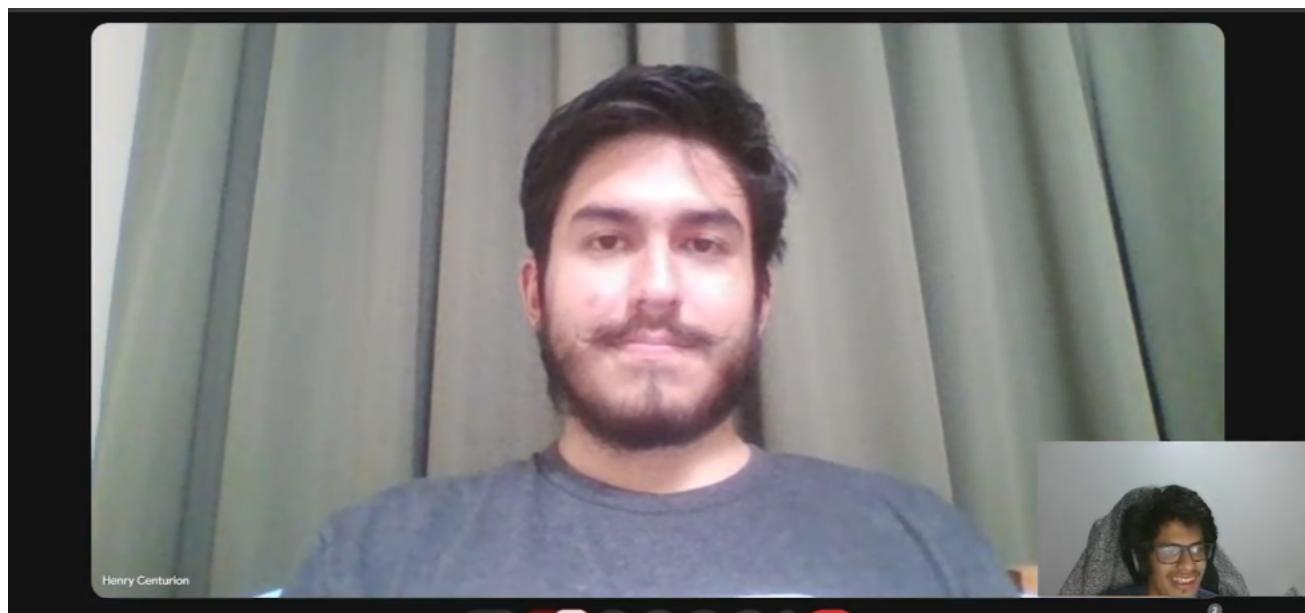
3. Preguntas - Segmento: Empresa (Gestores de transporte)

- ¿Considera que el sistema de monitoreo proporciona información relevante para la gestión de su flota o carga?
- ¿Las funcionalidades actuales le permiten tener mayor control y visibilidad sobre sus operaciones logísticas?
- ¿Hay alguna información o métrica que considere esencial y que no se esté mostrando actualmente?
- ¿Qué variables (temperatura, humedad, ubicación) son las más críticas para usted y por qué?
- ¿Qué herramientas o funcionalidades cree que hace falta al sistema para realizar su trabajo en el día a día?

6.3.2. Registro de Entrevistas.

Segmento 1: Empresa

- Nombre: Henry Centurion
- Edad: 40
- Distrito: Chorrillos, Lima
- Tiempo de inicio- tiempo fin: **00:00 - 14:33**
- Enlace: https://upcdupe-my.sharepoint.com/:g/personal/u20201c410_upc_edu_pe/IQCaPBt614hIT6IYV9oBEHJ_AQTjenHluVf5s9fmKmGTlfc?e=hYNpBV&nav=eyJyZWZlcnJhbEluZm8iOnsicmVmZXJyYWxBcHAiOijTdHJIYW1XZWJBcHAiLCJyZWZlcnJhbFZpZXciOijTaGFyZURpYWxvZy1MaW5rliwicmVmZXJyYWxBcHBQbGF0Zm9ybSI6IldlYilsInJlZmVycmFsTW9kZSI6InZpZXcifX0%3D



Henry considera que la aplicación web ofrece una experiencia positiva y útil para la gestión logística. Destaca que el dashboard proporciona una vista clara e inmediata de los indicadores principales, las alertas en tiempo real y los viajes recientes, permitiéndole tomar decisiones rápidas. Percebe la navegación como intuitiva y valora la fluidez del sistema. Sin embargo, menciona oportunidades de mejora, como una mejor organización del detalle de viaje, notificaciones más visibles y funciones avanzadas como reportes exportables, mapa en tiempo real, filtros adicionales y métricas más profundas para la toma de decisiones. En general, reconoce que la plataforma es un buen punto de partida y que aporta valor, aunque aún puede evolucionar para convertirse en una herramienta más completa y estratégica para sus operaciones.

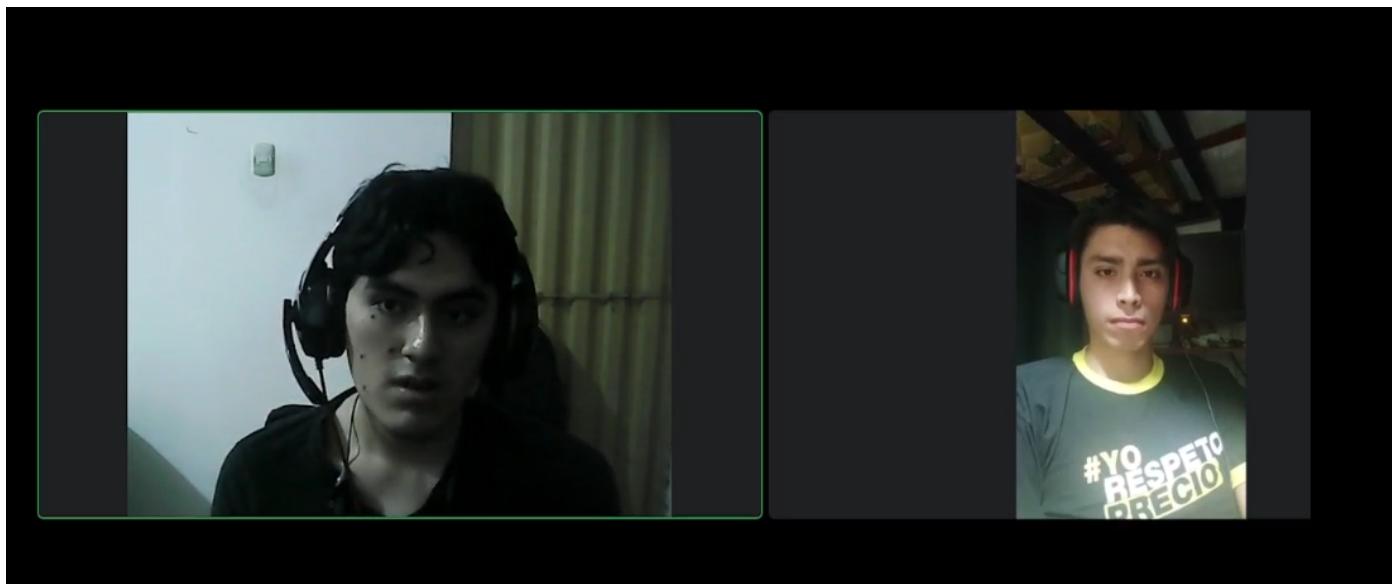
- Nombre: Miguel Ruiz
- Edad: 28
- Distrito: Los Olivos
- Tiempo de inicio- tiempo fin: **00:00 - 7:42**
- Enlace: https://upcdupe-my.sharepoint.com/:g/personal/u20201c410_upc_edu_pe/IQDmuN0X_TVLR4jsZIVDObx9Af96wSlZZ9YyLTyjlrq4YIE?nav=eyJyZWZlcnJhbEluZm8iOnsicmVmZXJyYWxBcHAiOijTdHJIYW1XZWJBcHAiLCJyZWZlcnJhbFZpZXciOijTaGFyZURpYWxvZy1MaW5rliwicmVmZXJyYWxBcHBQbGF0Zm9ybSI6IldlYilsInJlZmVycmFsTW9kZSI6InZpZXcifX0%3D&e=uT9XJE

**Miguel Ruiz**

Miguel considera que la aplicación es muy buena, dinámica e intuitiva. Señala que la navegación es clara, amigable y que no encontró confusiones durante su uso. Destaca que, aunque al inicio todo sistema nuevo puede resultar un poco complejo, luego de revisar el manual y recibir la explicación del equipo, la comprensión del flujo fue sencilla y rápida. Percibe que la aplicación muestra la cantidad justa de información, evitando saturación innecesaria. Resalta como funcionalidad más útil el sistema de alertas, ya que le permite detectar de inmediato si un transporte supera los rangos de temperatura configurados, reduciendo riesgos de pérdidas. Menciona que el sistema es ágil y fluido, sin trabas ni demoras, y que las notificaciones le resultaron claras y útiles para la toma de decisiones. Considera que el monitoreo proporciona todos los datos relevantes —ubicación, temperatura, estado de la carga y cumplimiento de ruta— y que la aplicación actualmente cumple con todas las funciones necesarias para la operación diaria. Como mejora futura, sugiere evaluar integraciones con otros sistemas para ampliar el uso de la tecnología, pero recalca que, según su experiencia, el producto ya resuelve adecuadamente las problemáticas actuales de la empresa.

Segmento 2: Clientes finales

- Nombre: Adrián Zapata
- Edad: 22
- Distrito: Chorrillos
- Tiempo de inicio- tiempo fin: **00:00 - 8:26**
- Enlace: https://upcedupe-my.sharepoint.com/:g/personal/u20201c410_upc_edu_pe/IQArPbnl6OTFRrb-NZnapN2hAX28R9Csp8-r_7Ch38BIWFK?e=nJhOmB&nav=eyJyZWZlcnjhbEluZm8iOnsicmVmZXJyYWxBcHAIoIJTdHJIYW1XZWJBcHAIcJyZWZlcnjhbFZpZXciOjTaGFyZURpYWxvZy1MaW5rlwicmVmZXJyYWxBcHBQbGF0Zm9ybSI6IldlYilslnJlZmVycmFsTW9kZSI6InZpZXcifX0%3D



Adrián considera que la aplicación es útil, clara y fácil de usar, especialmente después de familiarizarse con sus funciones. Señala que el sistema le ha permitido mejorar la calidad de sus productos durante el transporte, ya que puede monitorear variables críticas como la temperatura y la ubicación en tiempo real. Destaca que la información mostrada es precisa y suficiente, sin saturar la interfaz. Resalta como funciones más valiosas el control de temperatura, el seguimiento en el mapa y las

notificaciones, que le permiten anticipar retrasos o incidentes durante el trayecto. También percibe la aplicación como fluida y sin problemas de rendimiento. Sobre mejoras, menciona la importancia de fortalecer la confianza y transparencia con el cliente, así como brindar más información sobre los conductores o historial de servicios para generar mayor seguridad. En general, afirma que la aplicación ofrece un servicio completo y útil que aporta valor real a su negocio.

- Nombre: Gabriel Díaz
- Edad: 22
- Distrito: Callao
- Tiempo de inicio- tiempo fin: **00:00 - 03:51**
- Enlace: https://upcdeupe-my.sharepoint.com/:g/personal/u20201c410_upc_edu_pe/IQCqtFhq7l8bTKDnwJXP7GCAAAb5DJFOnhNqLSHd8F2HguHE?e=jEczfo&nav=eyJyZWZlcnJhbEluZm8iOnsicmVmZXJyYWxBcHAiOijTdHJIYW1XZWJBcHAiLCJyZWZlcnJhbFZpZXciOijTaGFyZURpYWxvZy1MaW5rlwicmVmZXJyYWxBcHBQbGF0Zm9ybSI6IldlYilsInJlZmVycmFsTW9kZSI6InZpZXciF0%3D



Adrián considera que la aplicación es útil, clara y fácil de usar, especialmente después de familiarizarse con sus funciones. Señala que el sistema le ha permitido mejorar la calidad de sus productos durante el transporte, ya que puede monitorear variables críticas como la temperatura y la ubicación en tiempo real. Destaca que la información mostrada es precisa y suficiente, sin saturar la interfaz. Resalta como funciones más valiosas el control de temperatura, el seguimiento en el mapa y las notificaciones, que le permiten anticipar retrasos o incidentes durante el trayecto. También percibe la aplicación como fluida y sin problemas de rendimiento. Sobre mejoras, menciona la importancia de fortalecer la confianza y transparencia con el cliente, así como brindar más información sobre los conductores o historial de servicios para generar mayor seguridad. En general, afirma que la aplicación ofrece un servicio completo y útil que aporta valor real a su negocio.

- Nombre: Brenda
- Edad: 22
- Distrito: Callao
- Tiempo de inicio- tiempo fin: **00:00 - 05:55**
- Enlace: https://upcdeupe-my.sharepoint.com/:g/personal/u20201c410_upc_edu_pe/IQCqtFhq7l8bTKDnwJXP7GCAAAb5DJFOnhNqLSHd8F2HguHE?e=jEczfo&nav=eyJyZWZlcnJhbEluZm8iOnsicmVmZXJyYWxBcHAiOijTdHJIYW1XZWJBcHAiLCJyZWZlcnJhbFZpZXciOijTaGFyZURpYWxvZy1MaW5rlwicmVmZXJyYWxBcHBQbGF0Zm9ybSI6IldlYilsInJlZmVycmFsTW9kZSI6InZpZXciF0%3D



Brenda percibe la aplicación como una herramienta muy bien diseñada y centrada en el usuario. Destaca que la interfaz es clara, ordenada y fácil de navegar, lo que hace que su experiencia sea agradable desde el primer momento. Considera que la información mostrada es justa y bien distribuida, sin ser abrumadora. La funcionalidad que más valora es el monitoreo de temperatura en tiempo real, debido a que le permite asegurar la calidad de los insumos sin necesidad de supervisión constante. También resalta la fluidez del sistema, la claridad de las alertas y la rapidez con la que se actualiza la información. Sugiere como mejora futura la incorporación de un historial más visual (por ejemplo, gráficos), y opciones para personalizar las notificaciones. Además, le gustaría contar con datos más detallados sobre el conductor, la unidad y la cadena de transporte para reforzar la confianza y transparencia del servicio. En general, considera que la aplicación le brinda control, tranquilidad y un apoyo real en su trabajo diario.

6.3.3. Evaluaciones según heurísticas

UX Heuristics & Principles Evaluation

Usability – Inclusive Design – Information Architecture

CARRERA: Ingeniería de Software

CURSO: Desarrollo de Soluciones IoT

SECCIÓN: 3443

PROFESORES: Todos

AUDITOR: Entrevistados

CLIENTE(S): Los Parkers

SITE o APP A EVALUAR

CargaSafe

TAREAS A EVALUAR

El alcance de esta evaluación incluye la revisión de la usabilidad de las siguientes tareas:

- Visualización del dashboard principal con métricas de monitoreo
- Gestión de vehículos
- Gestión de dispositivos IoT
- Registro y gestión de viajes/trips
- Visualización y gestión de alertas del sistema
- Navegación entre módulos del sistema

No están incluidas en esta versión de la evaluación las siguientes tareas:

- Registro e Inicio de Sesión
- Cierre de Sesión
- Visualización de Suscripciones

ESCALA DE SEVERIDAD

Los errores serán puntuados con la siguiente escala:

Nivel	Descripción
1	Problema superficial: se supera fácilmente o es poco frecuente. Se corrige solo si hay tiempo.
2	Problema menor: ocurre con mayor frecuencia o es algo difícil de superar. Prioridad baja para próximo release.
3	Problema mayor: ocurre frecuentemente o los usuarios no pueden resolverlo. Prioridad alta.
4	Problema muy grave: bloquea al usuario. Debe corregirse antes del lanzamiento.

TABLA RESUMEN

#	Problema	Severidad	Heurística/Principio violado
1	Botón de eliminar vehículo no funciona	4	Usability: Visibilidad del estado del sistema
2	Botón "Ver más detalles" en alertas no muestra información	4	Usability: Visibilidad del estado del sistema
3	Inconsistencia de idioma en interfaz	2	Usability: Consistencia y estándares
4	Texto truncado en filtro "Created Between"	2	Usability: Estética y diseño minimalista
5	Inconsistencia de idioma en dropdown de estado de viajes	2	Usability: Consistencia y estándares

DESCRIPCIÓN DE PROBLEMAS

PROBLEMA 1: Botón de eliminar vehículo no funciona

Severidad: 4

Heurística violada: Usability – Visibilidad del estado del sistema

Problema

En la sección de Vehículos, al intentar eliminar un vehículo clicando el ícono de papelera, la acción no se ejecuta y no se muestra feedback. No hay confirmación, mensaje de error ni indicación de restricciones. Esto bloquea la gestión correcta de vehículos.

ID	Plate	Type	Capabilities	Status	IOT	Actions
1	ABC-123	Truck	REFRIGERATED	In Service	IMD-123456789	
2	ASADASD	Van	REFRIGERATED, HEAVY LOAD	Out Of Service	-	

Recomendación

- Implementar correctamente la funcionalidad de eliminación.
- Agregar modal de confirmación.
- Mostrar mensajes claros si existen restricciones.
- Proveer feedback visual inmediato de éxito o error.

PROBLEMA 2: Botón "Ver más detalles" en alertas no muestra información

Severidad: 4

Heurística violada: Usability – Visibilidad del estado del sistema

Problema

En la sección Alerts, el botón "Ver más detalles" no muestra información adicional pese a que el usuario espera datos completos de la alerta (timestamp, sensor, ubicación). Esto impide evaluar incidentes críticos.

Alert Details

- Type: TEMPERATURE
- Status: CLOSED
- Created: November 14, 2025 at 4:03:45 PM GMT-5
- Closed: November 15, 2025 at 3:33:41 AM GMT-5
- Description: Threshold Exceeded

Ver más detalles

Type	Delivery Order ID	Status	Action
TEMPERATURE	1	GMT-5	---
TEMPERATURE	1	GMT-5	---
TEMPERATURE	1	GMT-5	November 15, 2025 at 3:33:41 AM GMT-5
TEMPERATURE	1	GMT-5	---

Recomendación

- Implementar el despliegue detallado de la alerta.
- Incluir timestamp, datos del sensor, ubicación, historial, acciones previas.
- Deshabilitar el botón si no hay información.
- Proporcionar feedback visual inmediato.

PROBLEMA 3: Inconsistencia de idioma en interfaz

Severidad: 2

Heurística violada: Usability – Consistencia y estándares

Problema

La interfaz mezcla inglés y español. Ejemplo: secciones en inglés pero un botón "Ver más detalles" en español. Esto genera confusión y falta de profesionalismo.

Alert Details

- Type: TEMPERATURE
- Status: ACKNOWLEDGED
- Created: November 14, 2025 at 4:03:31 PM GMT-5
- Closed: ---
- Description: Threshold Exceeded

Ver más detalles

Delivery ID	Status	Action
	GMT-5	
	GMT-5	
	GMT-5	

Recomendación

- Definir un idioma principal.

- Traducir todos los elementos de la interfaz.
- Implementar sistema de internacionalización (i18n).
- Agregar selector de idioma si corresponde.

PROBLEMA 4: Texto truncado en filtro "Created Between"

Severidad: 2

Heurística violada: Usability – Estética y diseño minimalista

Problema

El texto "Created Betw..." aparece truncado en el filtro de fecha en Trips, afectando legibilidad y estética.



Recomendación

- Ajustar ancho del contenedor.
- Usar textos más cortos como "Date Range".
- Implementar sistema responsive.
- Añadir tooltip con texto completo.

PROBLEMA 5: Inconsistencia de idioma en dropdown de estado de viajes

Severidad: 2

Heurística violada: Usability – Consistencia y estándares

Problema

El dropdown de estado mezcla inglés ("Status", "All") con español ("Completado", "Cancelado"). Esto rompe la consistencia del componente.

Status	Completed deliveries	Started at	Ended at
All	0	5, 5:21 AM	-
All	0	5, 5:21 AM	-
All	1	5, 4:01 PM	-
	1	11/14/25, 4:01 PM	11/14/25

Recomendación

- Unificar idioma del label y opciones.
- Si se usa inglés: All, Completed, Cancelled, In Progress, Scheduled.
- Si se usa español: Todos, Completado, Cancelado, En curso, Programado.
- Revisar todos los dropdowns para asegurar consistencia global.

6.4. Video About-the-Product.

Como parte del proceso de validación y comunicación de la solución propuesta, se desarrolló un video About the Product cuyo objetivo es explicar de manera clara y concisa el propósito del proyecto.

El material también presenta las principales funcionalidades de la aplicación web, evidenciando cómo estas responden directamente a las necesidades reales de los usuarios y de las empresas dedicadas al transporte de carga sensible. Además, se incluye la reacción y aceptación de un entrevistado de cada segmento objetivo, lo que permite reforzar la pertinencia y el valor percibido de la solución.

ID	Status	Driver	Total Deliveries	Completed deliveries	Created at	Started at	Actions
2	CREATED	1	1	1	11/14/25, 4:01 PM	-	
1	COMPLETED	1	1	1	11/14/25, 4:01 PM	11/14/25, 4:02 PM	

Duración: 00:00 - 02:23

Video en Microsoft Stream: https://upcedupe-my.sharepoint.com/:g/personal/u202019577_upc_edu_pe/IQCYKgWKcBrXR65PIG9NYDI6Adf84l5MqK0xFe_4ziO52dI?e=rNUkdK

Video en Youtube: <https://youtu.be/v0NPYFrl8V4>

Conclusiones

Conclusiones y recomendaciones

Conclusiones

El trabajo permitió validar que el problema identificado en el Lean UX, la falta de visibilidad en tiempo real y la ausencia de trazabilidad confiable en el transporte de productos sensibles, es real y crítico para los segmentos estudiados. Las entrevistas confirmaron la mayoría de los Assumptions planteados, especialmente la necesidad de alertas inmediatas, reportes automáticos y paneles simples, así como la disposición de las empresas a utilizar un modelo de suscripción. Asimismo, las hipótesis fueron parcialmente comprobadas mediante la interacción con prototipos, evidenciando mejoras potenciales en eficiencia operativa y capacidad de respuesta. En conjunto, los resultados de las validaciones demostraron que CargaSafe responde adecuadamente a los criterios de éxito propuestos, alineando investigación, diseño UX y arquitectura técnica en una solución viable y de alto valor para los usuarios.

Recomendaciones

Se recomienda priorizar en el roadmap el desarrollo de las funcionalidades que demostraron mayor impacto en las validaciones: el sistema de alertas en tiempo real, la trazabilidad completa por viaje y la generación automática de reportes. Además, se sugiere incorporar la funcionalidad adicional detectada en campo, el enlace público de seguimiento, por su alto valor percibido para empresas y clientes. Finalmente, se aconseja realizar pilotos con empresas de alimentos y farmacéutica para validar métricas reales de reducción de incidentes y ajustar el producto antes de su escalamiento, asegurando que CargaSafe evolucione de forma consistente con las necesidades del mercado y los aprendizajes del proceso Lean UX.

Video About-the-Team

CargaSafe

LOS PARKERS

Video en Youtube: <https://www.youtube.com/watch?v=gPWvu15rknA>

Bibliografía

- Bogdanov, V. (2024, 23 octubre). *Real-Time Supply Chain Visibility: a Game-Changer*. rinf.tech. <https://www.rinf.tech/real-time-supply-chain-visibility-a-game-changer/>
- Flock Freight | 2023 F&B Research Study. (s. f.). <https://www.flockfreight.com/2023-food-beverage-research-study>
- Perfectplanner. (2025, 30 enero). *Enhancing Supply Chain Visibility through Real-Time Tracking Technologies*. Perfect Planner. <https://perfectplanner.io/enhancing-supply-chain-visibility/>
- Technavio. (2024, 27 marzo). Cold Chain Logistics Market For Pharmaceuticals Industry size is set to grow by USD 12.81 bn from 2024-2028, Agility Public Warehousing Co. K.S.C.P, Air Canada & AVINEX, and more to emerge as Some of the Key Vendors, Technavio. PR Newswire. <https://www.prnewswire.com/news-releases/cold-chain-logistics-market-for-pharmaceuticals-industry-size-is-set-to-grow-by-usd-12-81-bn-from-2024-2028--agility-public-warehousing-co-kscp-air-canada--avinex-and-more-to-emerge-as-some-of-the-key-vendors-technavio-302099252.html>

Anexos

Anexo A Artefactos visuales

Lean UX Canvas

Lean UX Canvas

<h2>1. BUSINESS PROBLEM</h2> <p>Las empresas de transporte de productos sensibles necesitan soluciones accesibles para monitorear múltiples parámetros de la carga (temperatura, humedad, vibración, etc.) y evitar pérdidas económicas por incumplimiento de parámetros.</p>	<h2>5. SOLUTION IDEAS</h2> <p>“</p> <ul style="list-style-type: none"> • Plataforma IoT de monitoreo en tiempo real (múltiples sensores) • Sistema de alertas automáticas (vibración, volcado, energía, etc.) • Dashboard intuitivo de gestión de flota • Reportes automatizados de trazabilidad completa • Modelo de suscripción escalable <p>”</p>	<h2>2. BUSINESS OUTCOMES</h2> <ul style="list-style-type: none"> • 150 empresas suscritas en año 1 • Reducción 30% pérdidas de clientes • 80% satisfacción de usuarios • Revenue recurrente de \$180K anuales • Expansión a 3 países LATAM
<h2>3. USERS</h2> <p>Primarios:</p> <ul style="list-style-type: none"> • Gerentes de operaciones de transporte • Conductores de vehículos <p>Secundarios:</p> <ul style="list-style-type: none"> • Clientes finales de servicios de transporte • Administradores de empresas de transporte 		<h2>4. USER OUTCOMES & BENEFITS</h2> <p>Empresas de Transporte:</p> <ul style="list-style-type: none"> • Reducción de pérdidas por múltiples factores (temperatura, vibración, etc.) • Cumplimiento regulatorio automatizado • Optimización de rutas y recursos <p>Clientes Finales:</p> <ul style="list-style-type: none"> • Visibilidad completa del estado de sus productos • Reportes de trazabilidad en tiempo real • Reducción de riesgos operativos
<h2>6. Hypothesis</h2> <p>Creemos que las PYMEs de transporte adoptarán una solución de monitoreo multiparamétrico si es accesible y fácil de usar.</p> <p>Para validar desarrollaremos un MVP con funcionalidades core (temp, humedad, ubicación) y mediremos adoption rate.</p>	<h2>7- WHAT'S THE MOST IMPORTANT THING WE NEED TO LEARN FIRST?</h2> <p>¿Están las empresas de transporte dispuestas a pagar por una solución de monitoreo multiparamétrico IoT y cuál es el precio óptimo que maximiza adopción y rentabilidad?</p>	<h2>8. WHAT'S THE LEAST AMOUNT OF WORK WE NEED TO DO TO LEARN THE NEXT MOST IMPORTANT THING?</h2> <p>“</p> <ul style="list-style-type: none"> • Desarrollar landing page con pricing y funcionalidades • Realizar 20 entrevistas con empresas target • Crear prototipo de dashboard con múltiples datos • Validar willingness to pay con pre-orders ? <p>”</p>

UXPRESSIA

This persona was built in upressoia.com

Design-Level EventStorming

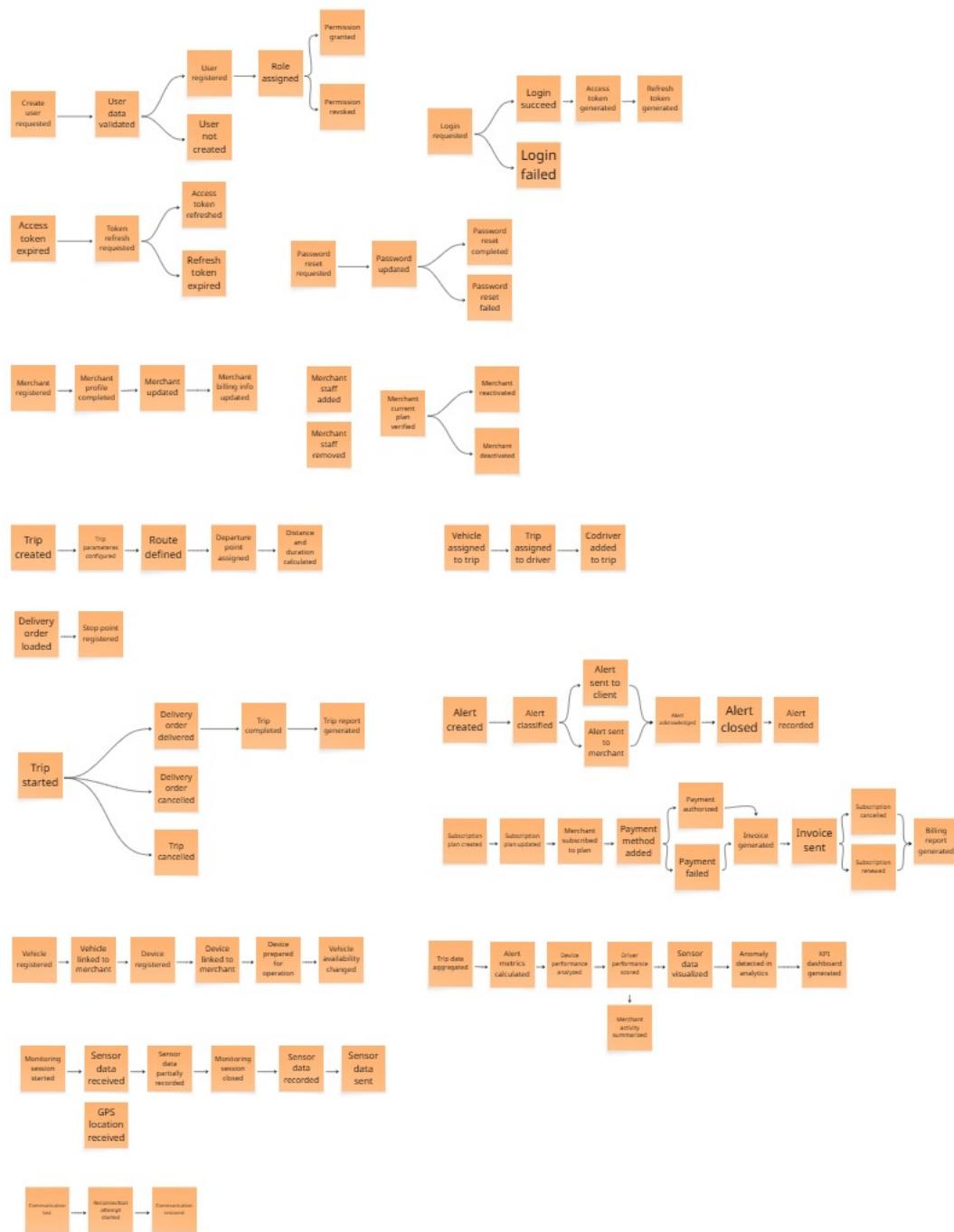
Step 1: Unstructured Exploration

STEP 1: Unstructured Exploration

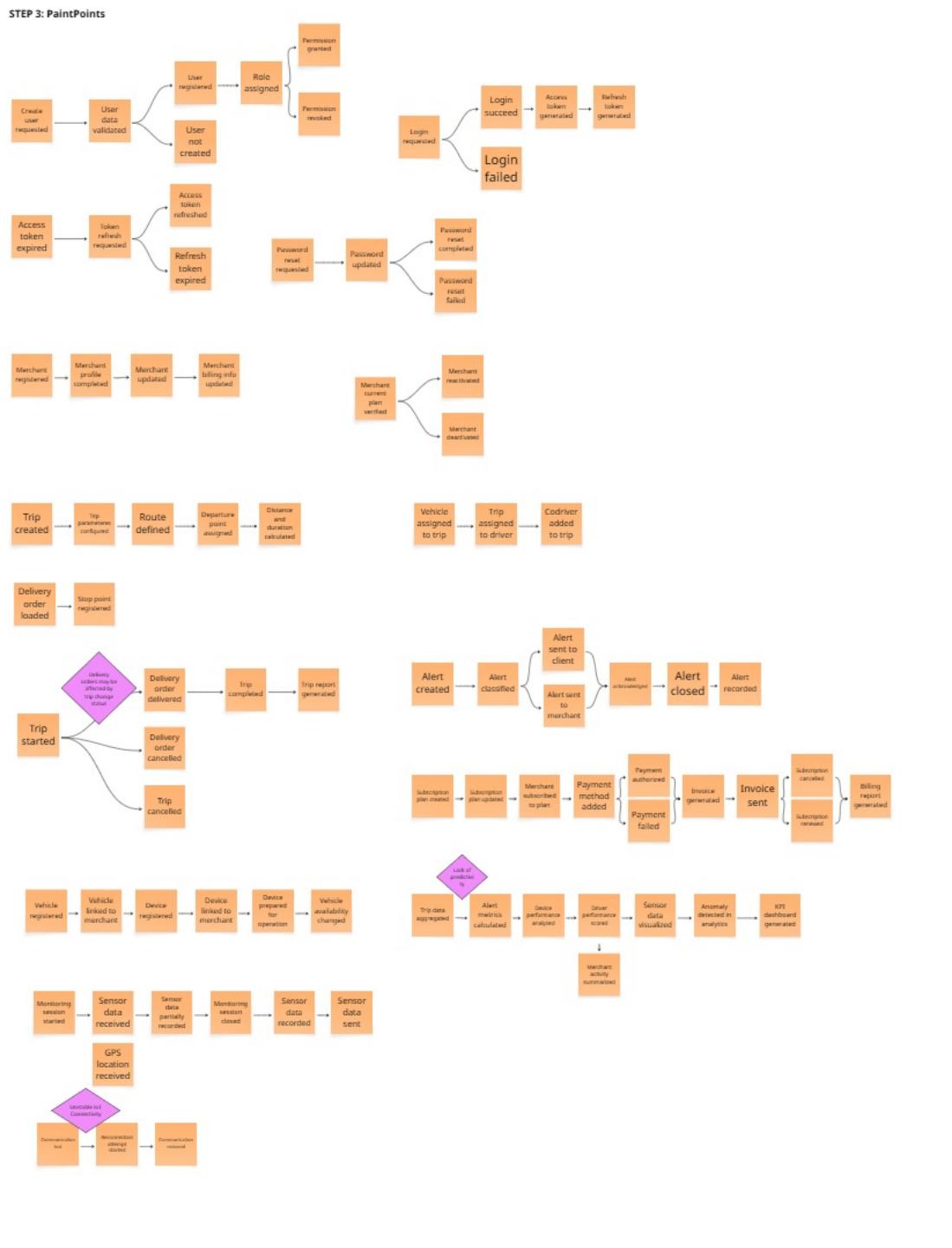


Step 2: Timeline

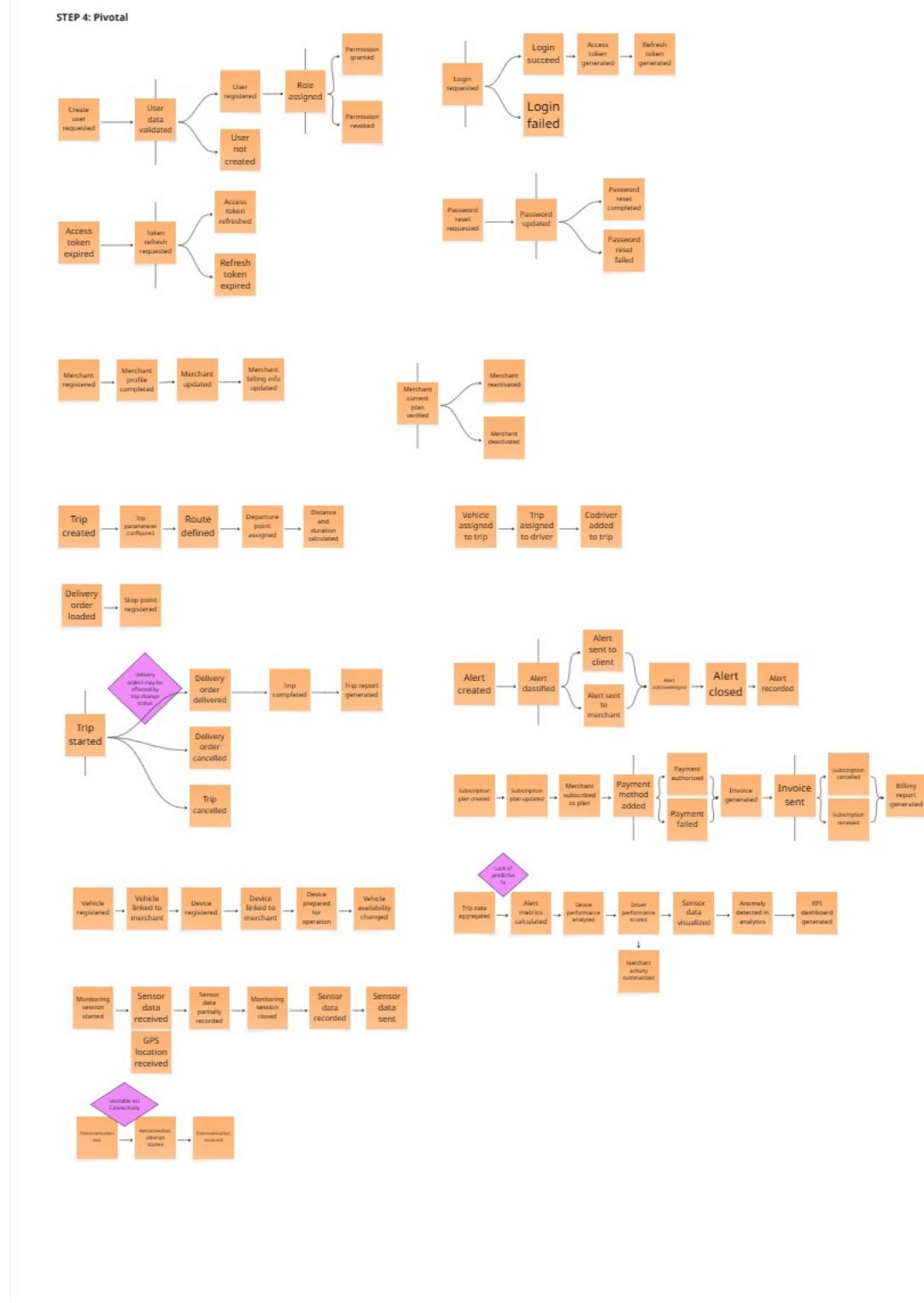
STEP 2: Timeline

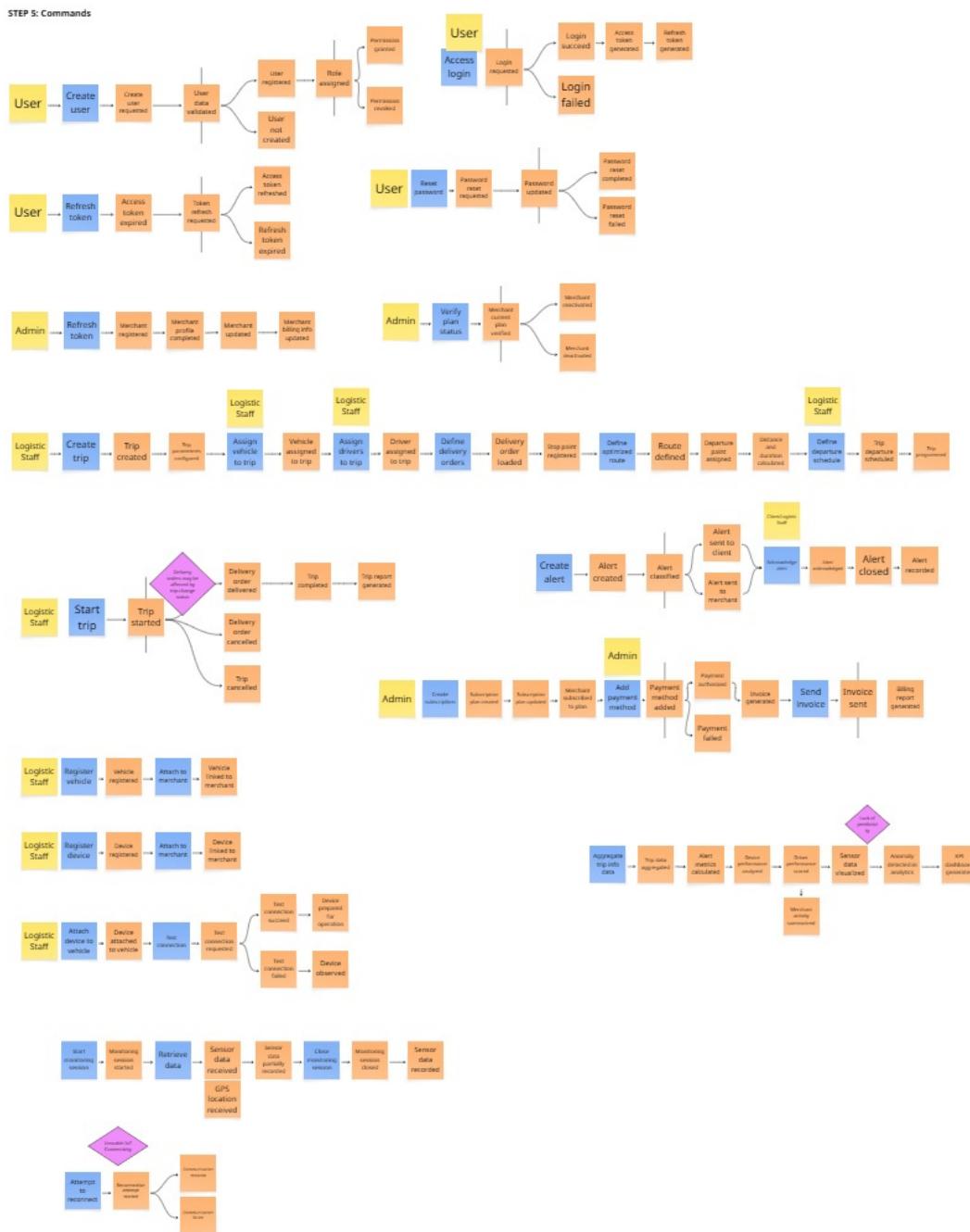


Step 3: Pain Points

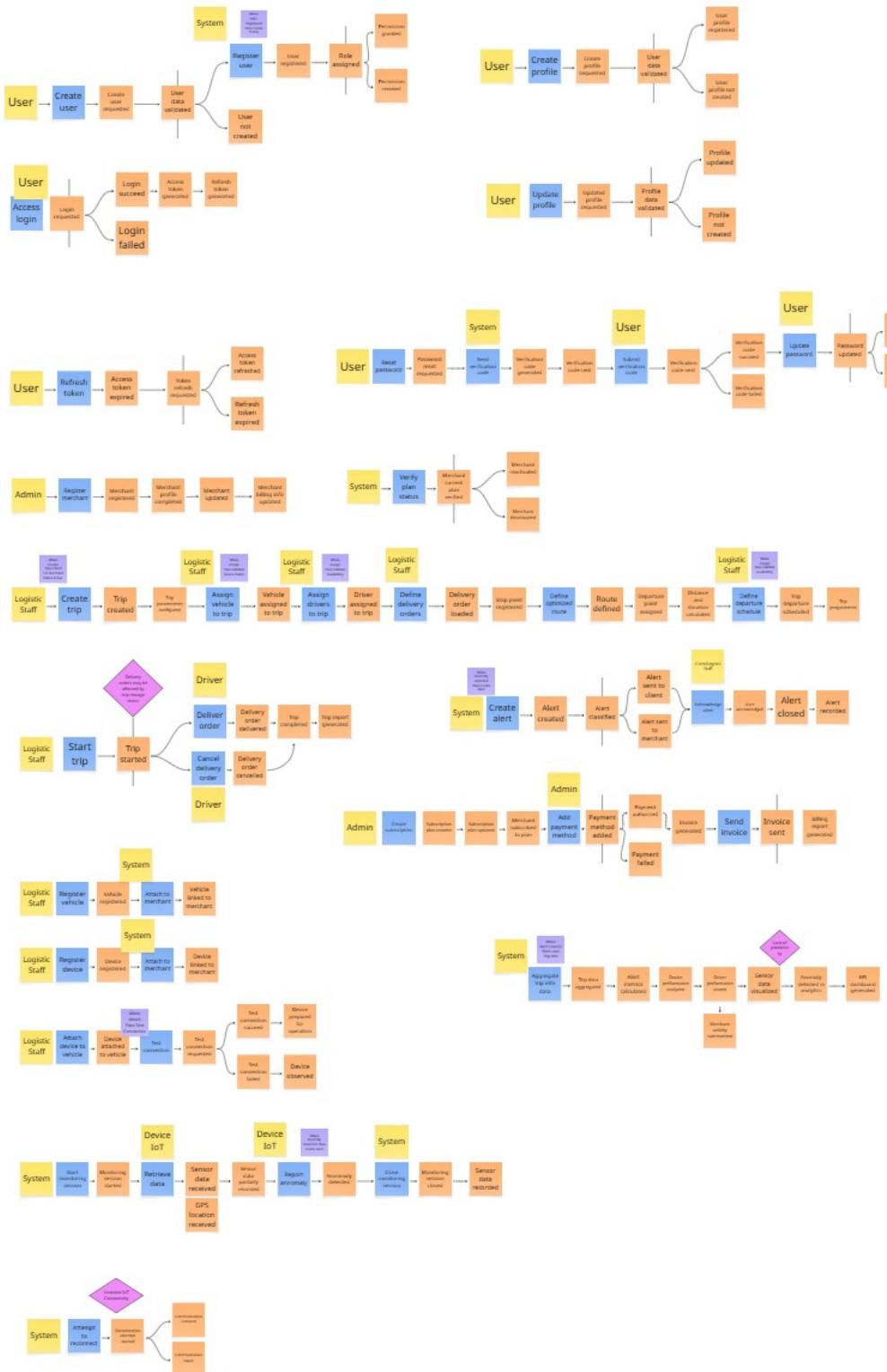


Step 4: Pivotal Points

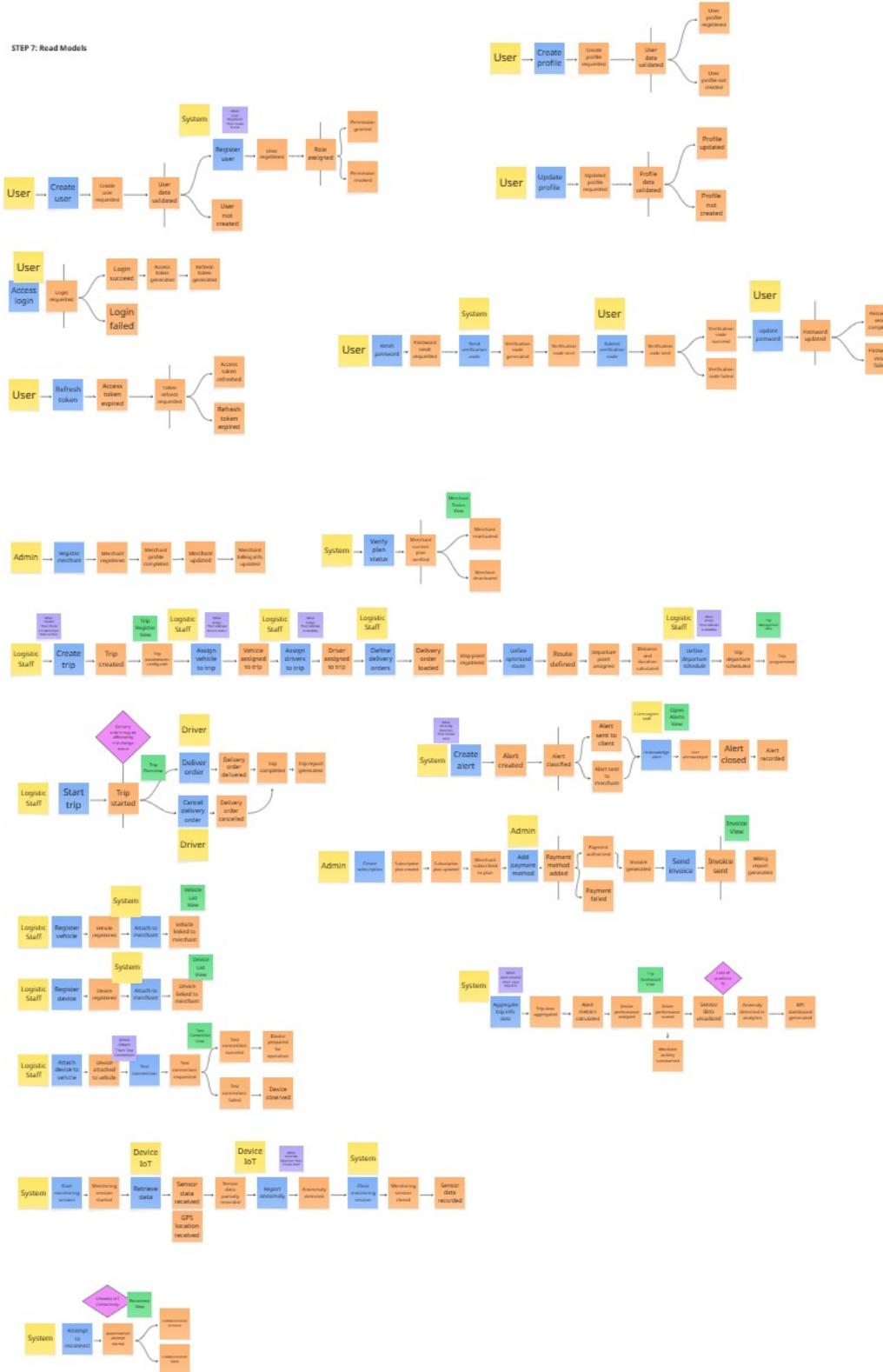


**Step 5: Commands**

STEP 6: Policies

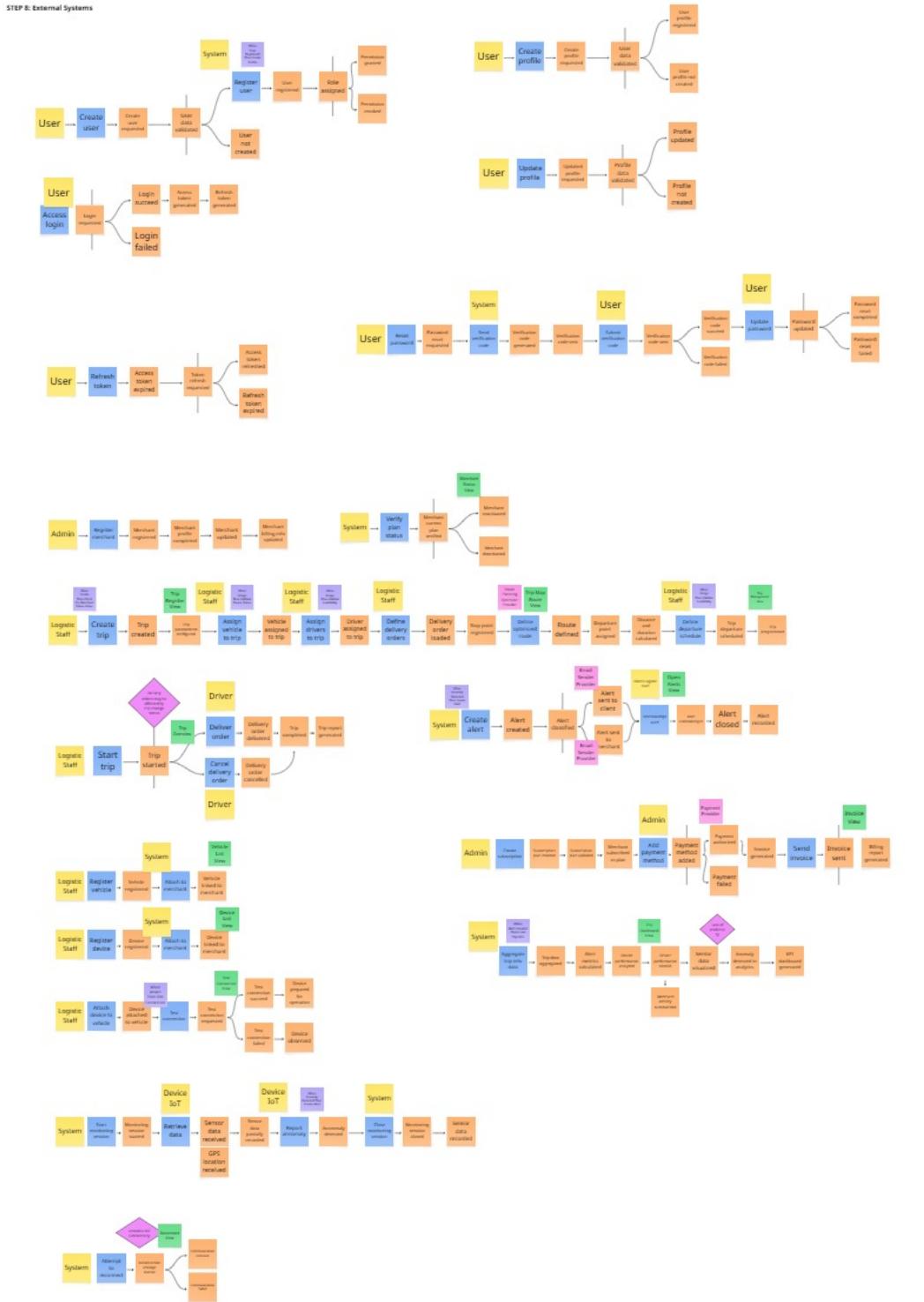


Step 6: Policies

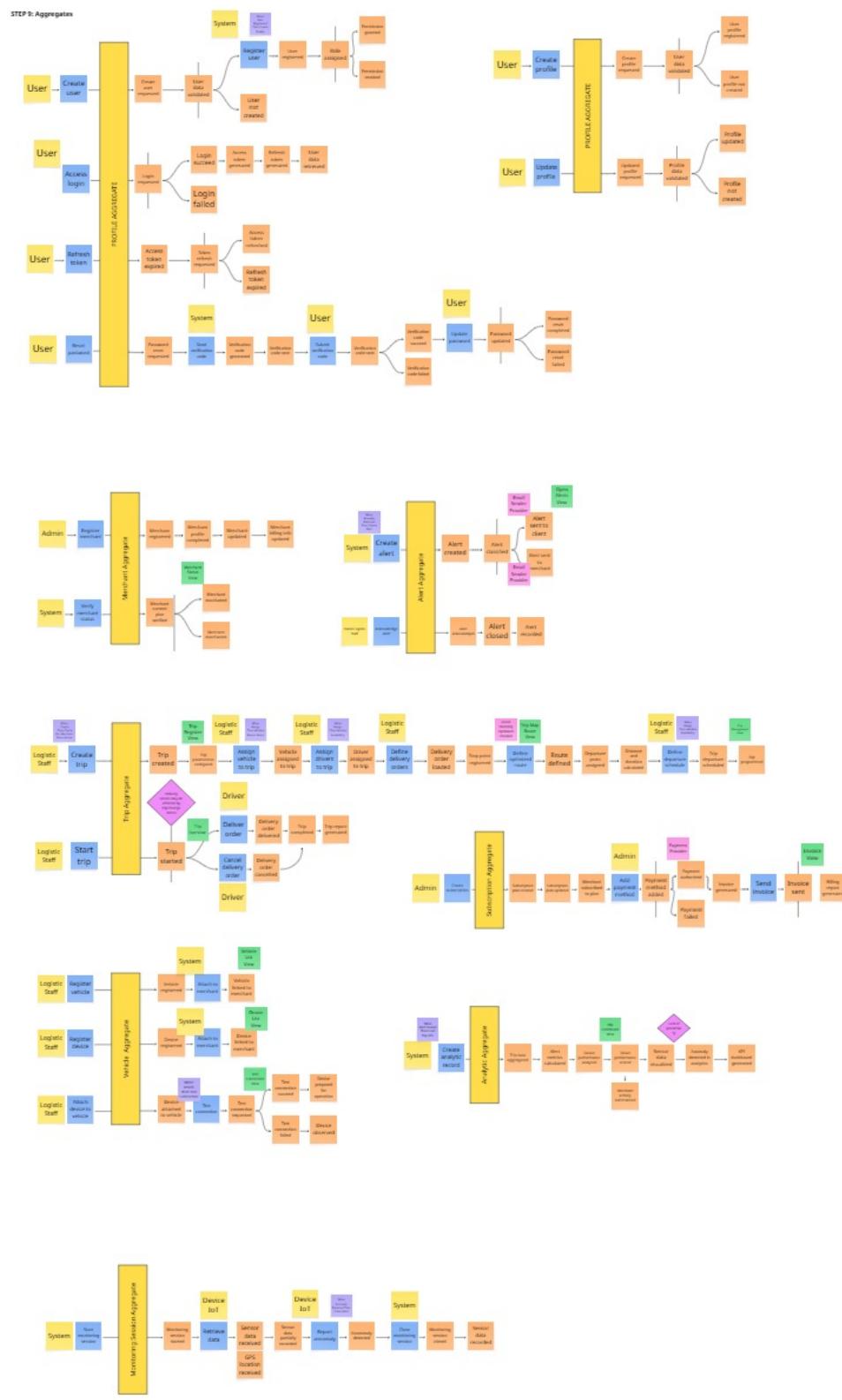


Step 7: Read Models

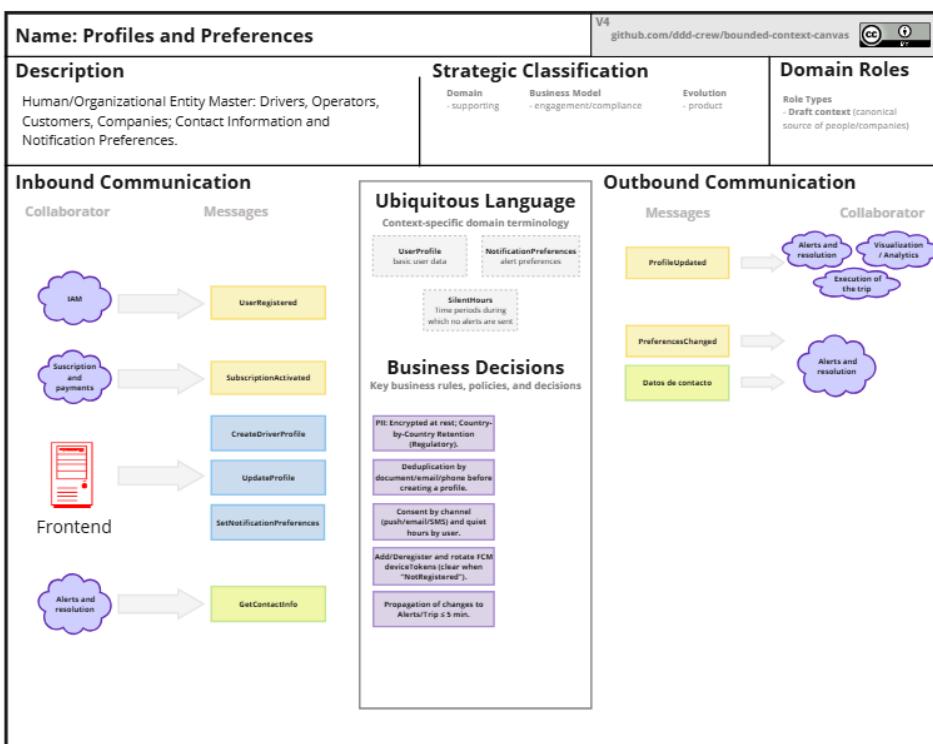
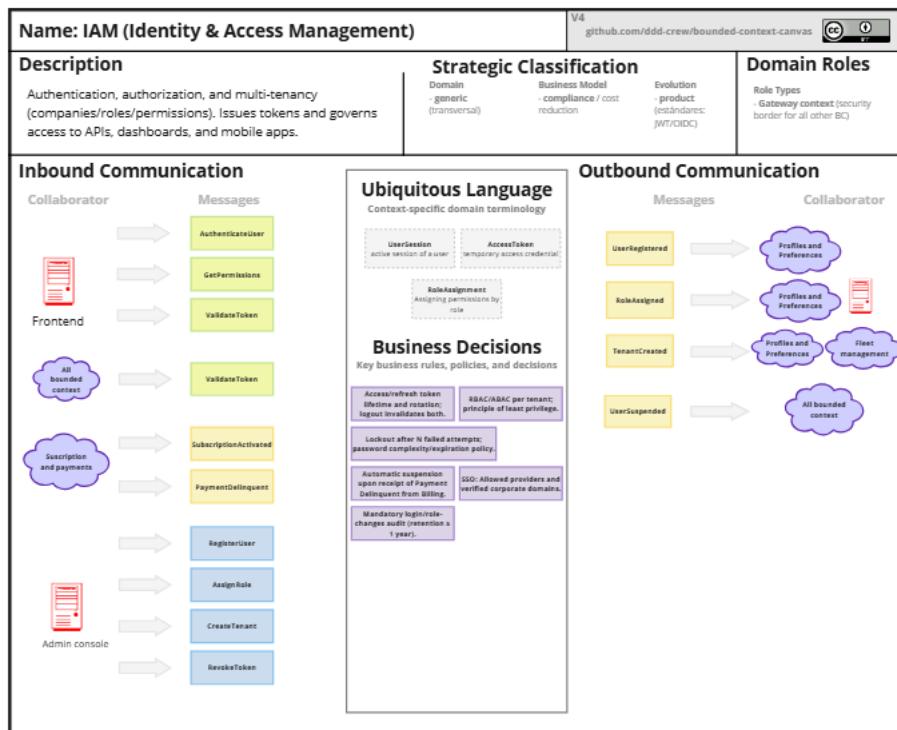
Step 8: External Systems

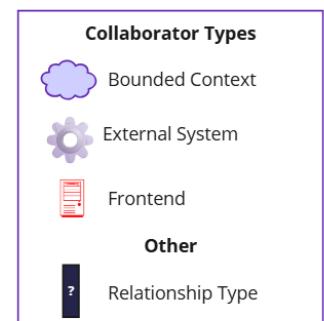
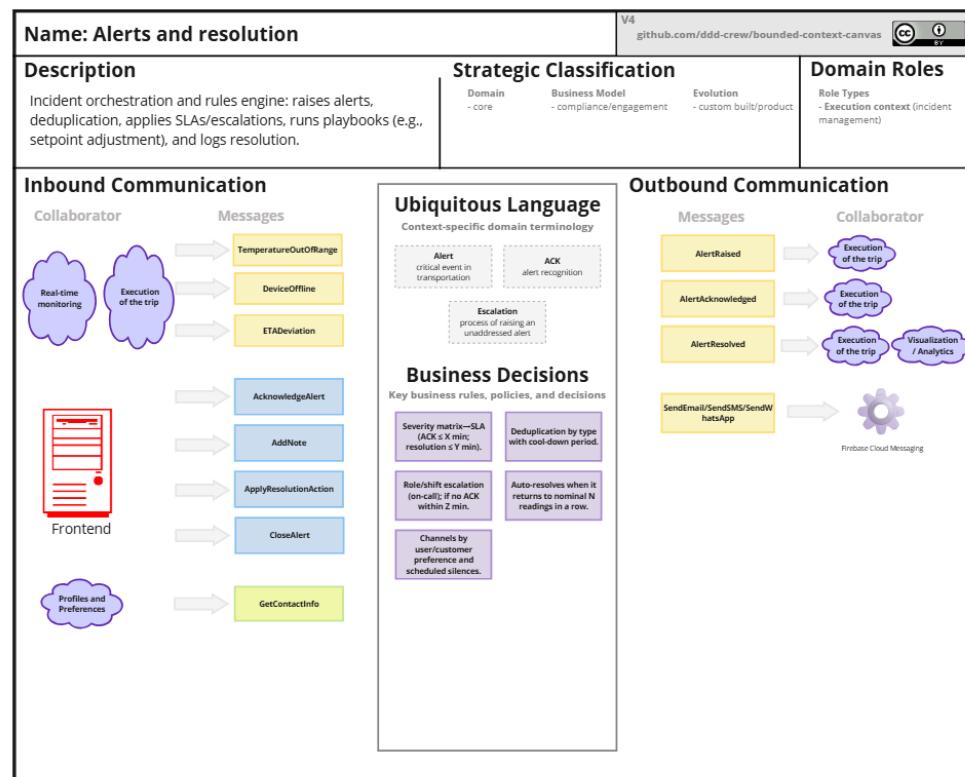
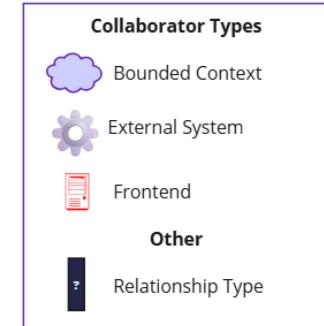
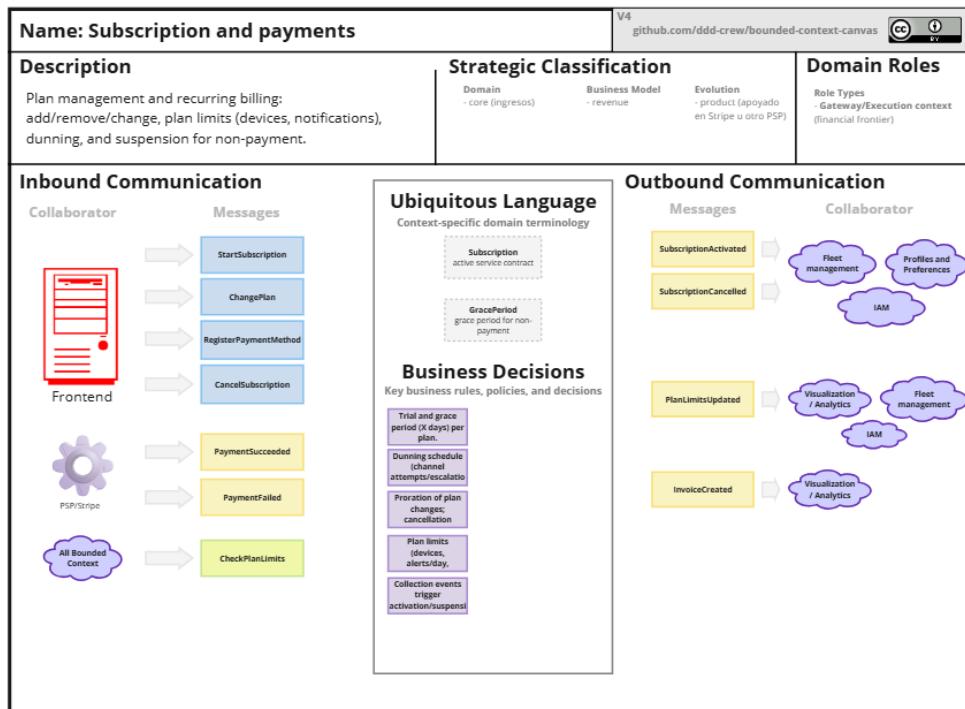


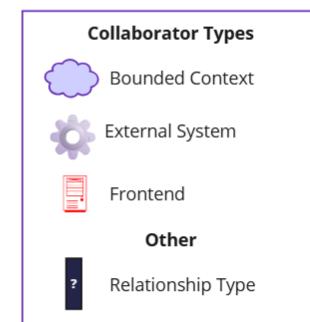
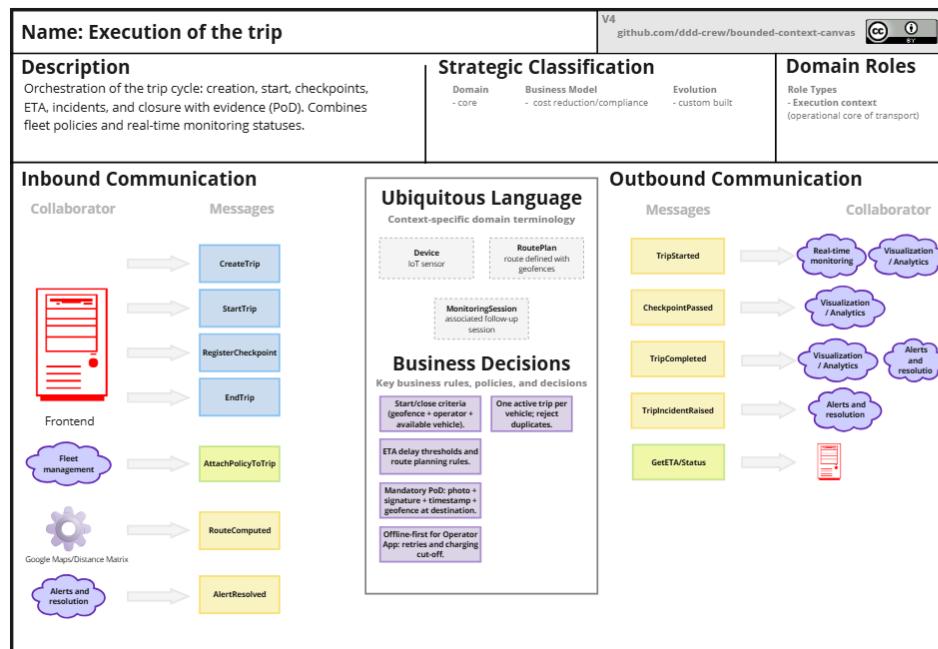
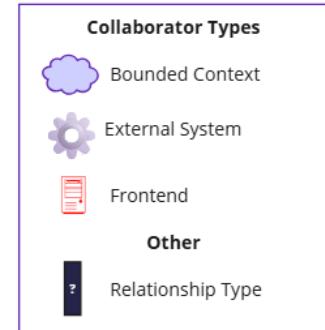
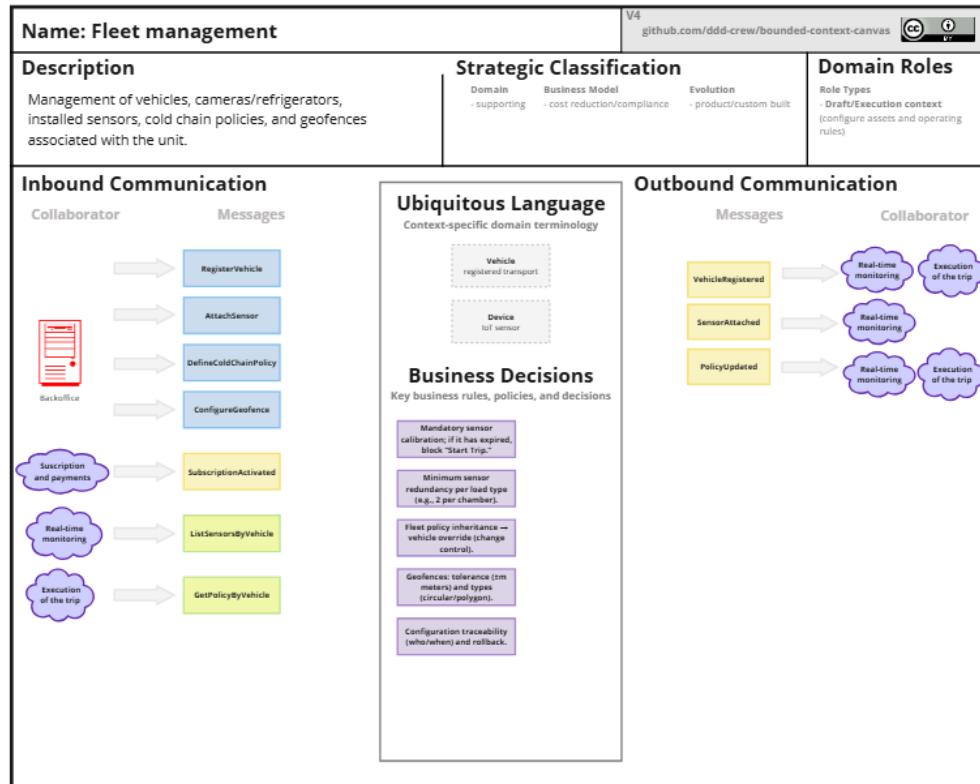
Step 9: Aggregates

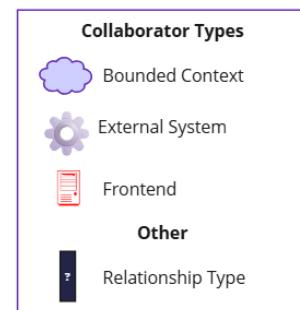
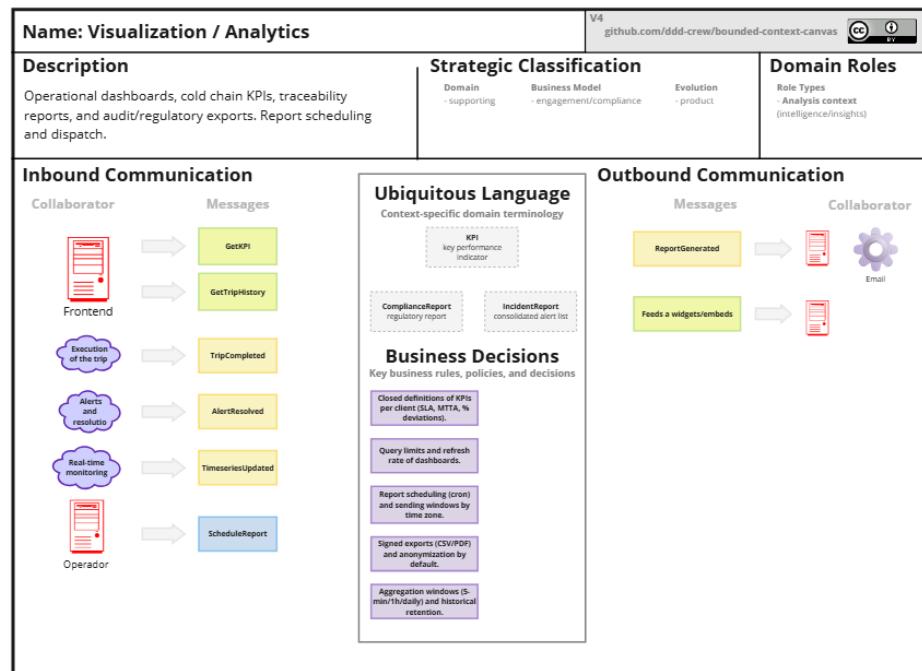
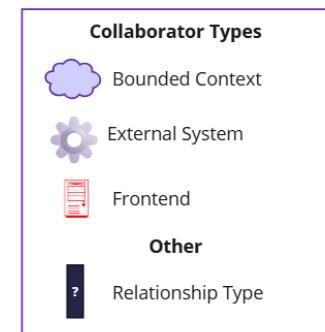
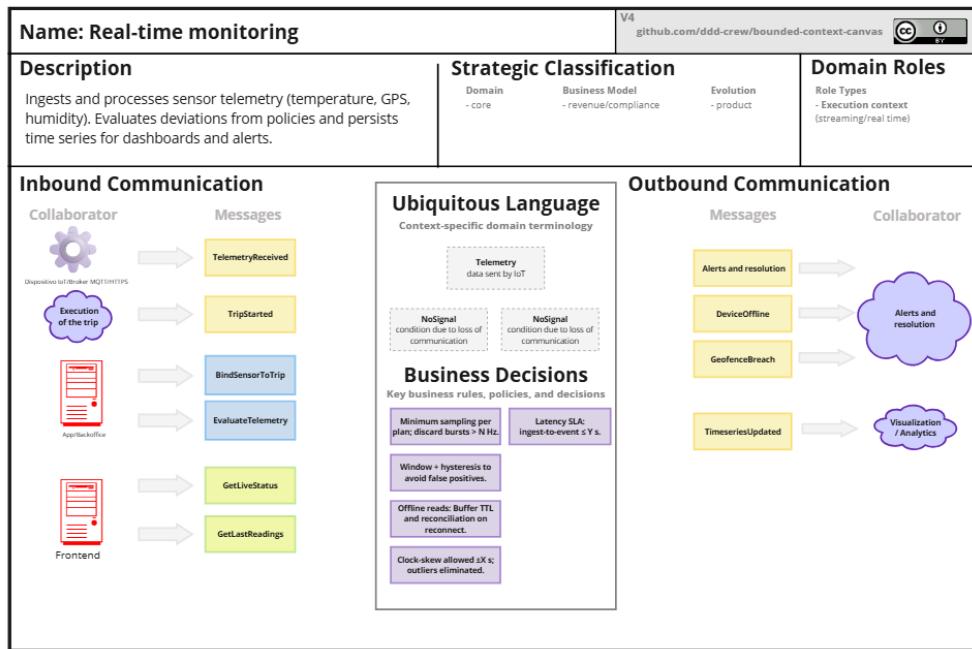


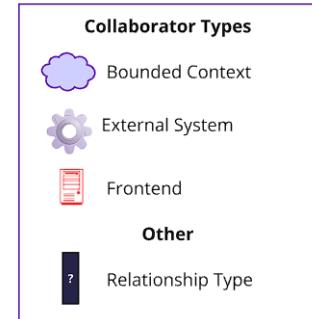
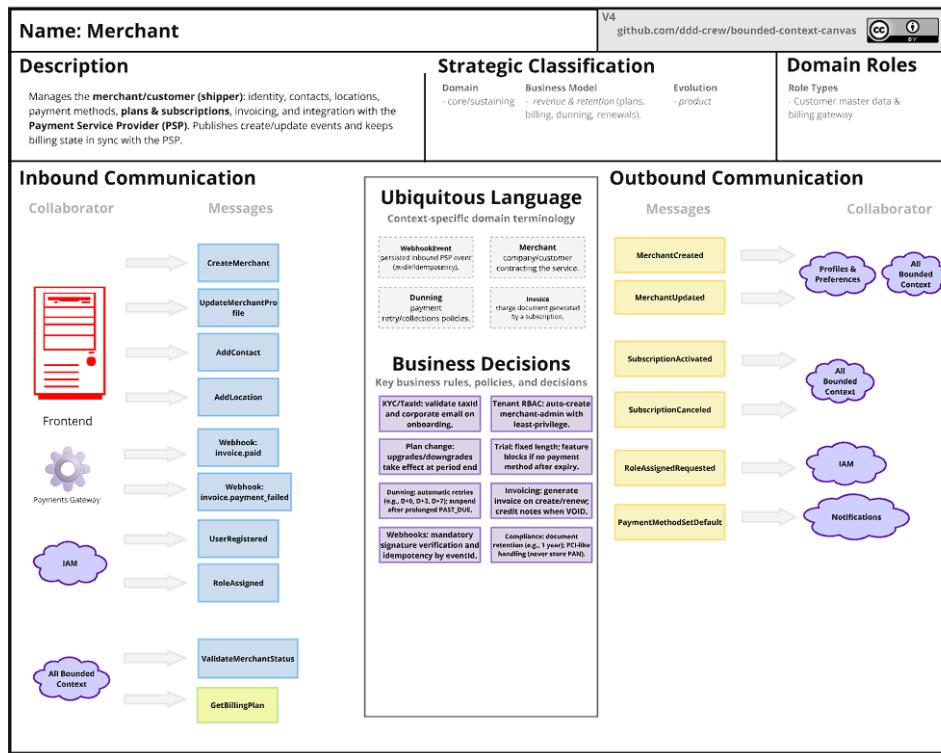
Bounded context canvases











Activar Windows
[Ve a Configuración para activar.](#)

Anexo B Evidencias Técnicas de Implementación

Vista inicial del encabezado principal y navegación

The screenshot shows the CargaSafe landing page. At the top, there is a navigation bar with links to 'Funcionalidades', 'Beneficios', 'Testimonios', 'Planes', and 'Contáctanos'. Below the navigation is a large orange button labeled 'Acceder vía web' and another labeled 'Descargar app'. A language selector 'ES' is also present. The main content area features a large image of a semi-truck driving on a road under a cloudy sky. Overlaid on the image is the CargaSafe logo (a truck icon) and the text 'Carga Safe' followed by the tagline 'Monitoreo inteligente para un transporte confiable.' A call-to-action button '¡Cotízanos Ya!' is visible. Below the main image, there is a section titled 'FUNCIONALIDADES' with the heading '¿Qué hace CargaSafe?' and three small thumbnail images.

Sección de planes de suscripción con precios y descripción de beneficios

The screenshot shows the 'SUSCRIPCIONES' (Subscription) section of the CargaSafe website. The title 'Elige tu plan' is displayed. Three subscription plans are listed in boxes:

- PLAN BASICO**
Hasta 2 vehículos
s/. 29/mes
✓ Monitoreo en tiempo real
✓ Alertas inteligentes
✓ Gestión centralizada
[Elegir plan](#)
- PLAN ESTANDAR**
Hasta 5 vehículos
s/. 59/mes
✓ Monitoreo en tiempo real
✓ Alertas inteligentes
✓ Gestión centralizada
[Elegir plan](#)
- PLAN PREMIUM**
Vehículos ilimitados
s/. 99/mes
✓ Monitoreo en tiempo real
✓ Alertas inteligentes
✓ Gestión centralizada
[Elegir plan](#)

Dashboard principal con métricas de viajes, incidencias y alertas activas

<https://carga-safe.web.app/dashboard>

Universidad Redes sociales Herramientas par... Programación Todos los favoritos

CargaSafe

- Dashboard
- Vehicles
- Sensors
- Trips
- Alerts
- Subscriptions

Dashboard - Monitoreo de Cargas

Visualización y análisis de viajes, incidencias y alertas del sistema CargaSafe

TOTAL DE VIAJES
7

VIAJES ACTIVOS
6

ALERTAS TOTALES
7

ALERTAS PENDIENTES
3

Gráfico de Incidencias por Mes

Incidencias por Mes

Fecha Inicio: dd/mm/aaaa Fecha Fin: dd/mm/aaaa Todos los tipos

Mes	Incidentes Abiertos	Incidentes Resueltos
Septiembre	5	1
Octubre	5	1
Noviembre	8	1
Diciembre	7	1
Enero	9	1
Febrero	7	1
Marzo	5	1
Abril	7	1
Mayo	5	1
Junio	4	1
Julio	6	1
Agosto	5	1
Septiembre	5	1

Vista del módulo de alertas con listado, estado y acciones de resolución

<https://carga-safe.web.app/alerts>

Universidad Redes sociales Herramientas par... Programación Todos los favoritos

CargaSafe

- Dashboard
- Vehicles
- Sensors
- Trips
- Alerts
- Subscriptions

Alerts

2
 Alerts today

3
 Resolved

Search alerts Status

ID	Type	Delivery Order ID	Status	Created	Closed	Actions	Details
1	High Temperature	T342	Closed	Wed Oct 08 2025 16:05:14 GMT-0500 (hora estándar de Perú)	Wed Oct 08 2025 16:05:26 GMT-0500 (hora estándar de Perú)	Resolved	
2	Excessive vibration	T343	Closed	Wed Oct 08 2025 16:41:00 GMT-0500 (hora estándar de Perú)	Wed Oct 08 2025 16:41:04 GMT-0500 (hora estándar de Perú)	Resolved	
3	Energy Loss	T344	Closed	Wed Oct 08 2025 16:41:04 GMT-0500 (hora estándar de Perú)	Wed Oct 08 2025 16:41:04 GMT-0500 (hora estándar de Perú)	Resolved	
4	Trip Loss	T345	Active	Wed Oct 08 2025 20:08:42 GMT-0500 (hora estándar de Perú)	---		
5	Gate Loss	T346	Active	Wed Oct 08 2025 20:08:42 GMT-0500 (hora estándar de Perú)	---		

Items per page: 1 – 5 of 5

Panel principal de Firebase mostrando la compilación e implementación exitosa del proyecto

The screenshot shows the Firebase Hosting dashboard for the project "carga-safe". The main header says "Administrar sitio" (Manage site) with a note "Preguntarle a Gemini cómo configurar Hosting para tu app" (Ask Gemini how to configure Hosting for your app). The left sidebar has sections for Hosting (selected), App Check, Novedades, AI Logic (NUEVO), Categorías de producto, Compilación, Ejecución, Analytics, and IA. The right side shows the "Versión actual" (Current version) with a star icon, the email "github-action-1062335524@carga-safe.iam.gserviceaccount.com", the date "8/10/25, 7:24 p.m.", and the hash "4ba51d". Below it is a list of "Versiones anteriores" (Previous versions) with four entries, each with a star icon, the same email, dates, and hashes: "4ba51d" (8/10/25, 7:24 p.m.), "098a22" (8/10/25, 7:19 p.m.), "cb96d4" (8/10/25, 7:00 p.m.), and "0942cb" (8/10/25, 6:57 p.m.). To the right are sections for "Dominios" (Domains) showing "carga-safe.web.app" (Predeterminado) and "carga-safe.firebaseioapp.com" (Predeterminado), and buttons for "Configuración de almacenamiento de actualizaciones" (Storage update configuration), "Agregar un dominio personalizado" (Add a custom domain), and "Ver los 2 dominios" (View both domains).

Historial de implementaciones y dominio activo del proyecto

The screenshot shows the main Firebase project overview for "carga-safe". The top bar includes "Descripción gene..." (General description), "Hosting" (selected), "App Check", "Novedades", "AI Logic (NUEVO)", "Categorías de producto", and "Spark" (Sin costo (USD 0 al momento)). The main area shows the "carga-safe" project with a "Plan Spark" button. Below it is a "Compilación" (Compilation) section with a "Hosting" card showing "Downloads" (No hay datos) and a "Historial de implementaciones" (Deployment history) card for "Implementado" (Implemented) on "8 oct 2025 7:24 p.m." by "github-action-1062335524@carga-safe.iam.gserviceaccount.com". At the bottom are "Próximos pasos con Gemini" (Next steps with Gemini) cards for "Cuéntanos acerca de tu aplicación" (Tell us about your app) and "Agrega estadísticas y supervisión" (Add statistics and monitoring), and a "Ver los documentos" (View documentation) link.

Configuración y despliegue de la landing mediante GitHub Pages

GitHub Pages

GitHub Pages is designed to host your personal, organization, or project pages from a GitHub repository.

Your site is live at <https://los-parkers-iot.github.io/iot-solutions-development-cargasafe-landing/>

Last deployed by [SgtGarro](#) 2 days ago

[Visit site](#) [Unpublish site](#)

Build and deployment

Source

Deploy from a branch

Branch

Your GitHub Pages site is currently being built from the `develop` branch. [Learn more about configuring the publishing source for your site.](#)

`develop` / (root) Save

Learn how to [add a Jekyll theme](#) to your site.

Your site was last deployed to the `github-pages` environment by the [pages build and deployment workflow](#). [Learn more about deploying to GitHub Pages using custom workflows](#)

Custom domain

Custom domains allow you to serve your site from a domain other than `los-parkers-iot.github.io`. [Learn more about configuring custom domains](#).

GitHub Actions

All workflows

Showing runs from all workflows

5 workflow runs

	Event	Status	Branch	Actor
chore(merge): merge branch 'develop' into feat/visualization-and-an...	main	Today at 9:34 PM	1m 49s	...
Merge branch 'hotfix/subscriptions-api-url'	main	Today at 7:23 PM	1m 31s	...
fix(subscriptions): replace hardcoded API URLs with environment var...	main	Today at 7:17 PM	1m 44s	...
fix(ci): update deployment branch to main in Firebase hosting workfl...	main	Today at 6:59 PM	1m 43s	...
build(ci):: add Firebase configuration and update environment settings	release/v0.0.1	Today at 6:55 PM	1m 40s	...

The screenshot shows the 'Vehicles' section of the CargaSafe application. On the left is a vertical navigation bar with icons for Dashboard, Vehicles (selected), Sensors, Trips, Alerts, and Subscriptions. At the bottom of the bar is a 'Logout' link. The main area has a header 'Vehicles' with a '+ Add Vehicle' button. Below it are two boxes: 'Vehicles Available' (1 of 2 total) and 'In Service' (1 active assignments). A search bar 'Search by plate, type...' is followed by two dropdown filters: 'All states - All states' and 'All capability - All capability'. A table lists vehicle details: ID (1), Plate (ABC-123), Type (Truck), Capabilities (REFRIGERATED), Status (In Service), IoT (IMEI-123456789), and Actions (edit, delete). Another row shows ID (2), Plate (ASADASD), Type (Van), Capabilities (REFRIGERATED, HEAVY_LOAD), Status (Out Of Service), IoT (-), and Actions (edit, delete). At the bottom are pagination controls: 'Items per page: 5', '1 - 2 of 2', and navigation arrows.

Vista del módulo Sensors con listado y detalles

The screenshot shows the 'IoT Devices' section of the CargaSafe application. The left navigation bar is identical to the previous screenshot. The main area has a header 'IoT Devices' with a '+ Register Device' button. A search bar 'Search by IMEI or vehicle...' and a dropdown filter 'All states - All states' are present. A table lists device details: ID (1), IMEI (IMEI-123456789), Connectivity (Online), Vehicle (ABC-123), and Actions (edit, delete). At the bottom are pagination controls: 'Items per page: 5', '1 - 1 of 1', and navigation arrows.

Pantalla del módulo Trips mostrando historial y estados de viajes

The screenshot shows the 'Trips' section of the CargaSafe application. On the left is a vertical navigation bar with icons for Dashboard, Vehicles, Sensors, Trips (selected), Alerts, and Subscriptions. A 'Logout' button is at the bottom. The main area has a header 'Trips' with a '+ Create' button. Below is a search bar and filters for Status (All, Recent) and Created Between dates. A table lists four trips:

ID	Status	Driver	Total Deliveries	Completed deliveries	Created at	Started at	Actions
3	CREATED	1	0	0	11/15/25, 5:21 AM	-	
4	CREATED	1	0	0	11/15/25, 5:21 AM	-	
2	CREATED	1	1	1	11/14/25, 4:01 PM	-	
1	COMPLETED	1	1	1	11/14/25, 4:01 PM	11/14/25, 4:02 PM	

At the bottom are buttons for 'Items per page' (5), '0 of 0', and navigation arrows.

Módulo Alerts con alertas activas, estados y acciones disponibles

The screenshot shows the 'Alerts' section of the CargaSafe application. The left navigation bar includes 'Alerts' (selected). The main area displays two summary boxes: one red for 'Active Alerts' (0) and one green for 'Closed' (1). Below is a table of alerts with columns for ID, Type, Delivery Order ID, Status, Created, Closed, Actions, and Details.

ID	Type	Delivery Order ID	Status	Created	Closed	Actions	Details
1	TEMPERATURE	1	ACKNOWLEDGED	November 14, 2025 at 4:03:31 PM GMT-5	---		
2	TEMPERATURE	1	ACKNOWLEDGED	November 14, 2025 at 4:03:40 PM GMT-5	---		
3	TEMPERATURE	1	CLOSED	November 14, 2025 at 4:03:45 PM GMT-5	November 15, 2025 at 3:33:41 AM GMT-5		
4	TEMPERATURE	1	ACKNOWLEDGED	November 14, 2025 at 4:03:50 PM GMT-5	---		

At the bottom are buttons for 'Items per page' (5), '1 - 4 of 4', and navigation arrows.

GET /api/v1/fleet/vehicles Get all vehicles

Parameters

No parameters

Execute Clear

Responses

Curl

```
curl -X 'GET' \
'https://iot-solutions-development-cargasafe.onrender.com/api/v1/fleet/vehicles' \
-H 'accept: */*' 
```

Request URL

```
https://iot-solutions-development-cargasafe.onrender.com/api/v1/fleet/vehicles
```

Server response

Code	Details
200	Response body

```
[  
  {  
    "id": 3,  
    "plate": "ABC-123",  
    "type": "TRUCK",  
    "capabilities": [  
      "REFRIGERATED"  
    ],  
    "status": "IN SERVICE",  
    "odometerKm": 234,  
    "deviceImeis": [  
      "IMEI-123456789"  
    ],  
    {  
      "id": 2,  
      "plate": "ASADASD",  
      "type": "TRAILER",  
      "capabilities": [  
        "REFRIGERATED",  
        "HEAVY LOAD"  
      ],  
      "status": "OUT_OF_SERVICE",  
      "odometerKm": 0,  
      "deviceImeis": []  
    }  
]
```

Response headers

```
alt-svc: h3=":443"; ma=86400  
cf-cache-status: DYNAMIC  
cf-ray: 99ecf19f6da5cdca-LIM  
content-encoding: br  
content-type: application/json  
date: Sat, 15 Nov 2025 07:13:04 GMT  
priority: u1,i  
rnr-id: 1869f25e-848d-4da2  
server: cloudflare  
server-timing: cfExtPri
```

Sensores

GET /api/v1/fleet/devices Get all devices

Parameters

No parameters

Cancel

Execute Clear

Responses

Curl

```
curl -X 'GET' \
'https://iot-solutions-development-cargasafe.onrender.com/api/v1/fleet/devices' \
-H 'accept: */*' 
```

Request URL

<https://iot-solutions-development-cargasafe.onrender.com/api/v1/fleet/devices>

Server response

Code	Details	Links
200	Response body	 
	[{ "id": 1, "imei": "IMEI-123456789", "firmware": "v1.0.0", "online": true, "vehiclePlate": "ABC-123" }]	
	Response headers	
	alt-svc: h3=":443"; ma=86400 cf-cache-status: DYNAMIC cf-ray: 99ecf3083905a9ea-1TM content-encoding: br content-type: application/json date: Sat,15 Nov 2025 07:14:02 GMT priority: u1,i rnrnd-id: 0061d814-20dd-436f server: cloudflare server-timing: cfTxtPri vary: Origin,Access-Control-Request-Method,Access-Control-Request-Headers,Accept-Encoding x-render-origin-server: Render	

Responses

Code	Description	Links
200	Devices returned	No links
	Media type	
	<input type="text" value=" */*"/>	

Viajes

Trips Endpoint for managing trips sources

GET /api/v1/trips List all trips

Parameters

No parameters

Responses

Curl

```
curl -X 'GET' \
  'https://iot-solutions-development-cargasafe.onrender.com/api/v1/trips' \
  -H 'accept: */*
```

Request URL

<https://iot-solutions-development-cargasafe.onrender.com/api/v1/trips>

Server response

Code **Details**

200 Response body

```
[{"id": 1, "merchantId": 1, "driverId": 1, "deviceId": 1, "status": "CREATED", "createdAt": "2025-11-14T16:01:45.508312", "startedAt": null, "completedAt": null, "order": null, "id": 1, "name": "Warehouse 1", "address": "Plaza Mayor de Lima, Cercado de Lima", "latitude": -12.046374, "longitude": -77.042795}, {"deliveryOrders": [{"id": 2, "tripId": 1, "clientEmail": "Daniel@hotmail.com", "sequenceOrder": 1, "address": "Av Brasil, Jesús María, Perú", "serviceLat": null, "status": "PENDING", "minHumidity": null, "maxHumidity": null}]}]
```

Response headers

```
alt-svc: h3="443"; ma=86400
cache-control: DYNAMIC
cf-ray: 70000000-0001-4f82-988e-4cad
content-encoding: br
content-type: application/json
date: Sat, 15 Nov 2025 00:47:55 GMT
pragma: no-cache
x-edge-id: S1f6sc0-938e-4cad
server: cloudflare
server-timing: cfExtPri
vary: Origin,Access-Control-Request-Method,Access-Control-Request-Headers,Accept-Encoding
x-render-origin-server: Render
```

Responses

Code **Description**

Links

Alertas

Alerts Endpoint for managing alerts sources

GET /api/v1/alerts

Parameters

No parameters

Execute **Cancel** **Clear**

Responses

Curl

```
curl -X 'GET' \
'https://iot-solutions-development-cargasafe.onrender.com/api/v1/alerts' \
-H 'accept: */*'
```

Request URL

<https://iot-solutions-development-cargasafe.onrender.com/api/v1/alerts>

Server response

Code	Details
200	Response body <pre>[{ "id": 1, "deliveryOrderId": 1, "alertType": "TEMPERATURE", "alertStatus": "ACKNOWLEDGED", "incidents": [{ "id": 1, "alertId": 1, "description": "Threshold Exceeded", "createdAt": "2025-11-14T16:03:31.436556", "acknowledgedAt": "2025-11-14T16:07:21.972417", "closedAt": null }], "notification": [{ "id": 1, "alertId": 1, "notificationChannel": "EMAIL", "message": "The TEMPERATURE threshold has been exceeded.", "sentAt": "2025-11-14T16:03:31.43659" }] }, { "id": 2, "deliveryOrderId": 1, "alertType": "TEMPERATURE" }]</pre> <p>Download</p> Response headers <pre>alt-svc: h3=":443"; ma=86400 cf-cache-status: DYNAMIC cf-ray: 98ecff9b6c58cdcd-LIM content-encoding: br content-type: application/json date: Sat, 15 Nov 2025 07:15:47 GMT priority: u=1,i rnrdr-id: Se75dc4d-e5ad-4b03 server: cloudflare x-content-type-options: cf-nosniff</pre>

Swagger UI

Para lograr un acceso rápido, coloca tus marcadores aquí en la barra de marcadores: [Importar marcadores ahora...](#)

Swagger [/docs](#) [Explore](#)

IoTParkers [0.0.1-SNAPSHOT](#) [OAS 3.0](#)

[/docs](#)

IoTParkers

Apache 2.0

CargaSafe Safe wiki Documentation

Servers

[https://iot-solutions-development-cargasafe.onrender.com - Generated server url](https://iot-solutions-development-cargasafe.onrender.com)

Devices Endpoints for managing IoT devices in fleet management

GET /api/v1/fleet/devices/{id} Get device by ID

PUT /api/v1/fleet/devices/{id} Update device

DELETE /api/v1/fleet/devices/{id} Delete device

GET /api/v1/fleet/devices Get all devices

POST /api/v1/fleet/devices Create device

POST /api/v1/fleet/devices/{id}/firmware Update device firmware

PATCH /api/v1/fleet/devices/{id}/online Change online status from device

GET /api/v1/fleet/devices/bv-online/{online} Get devices by online status

Buscar

23:33
14/12/2025

Anexo C Videos de Entrevistas

Tipo	Nombre del entrevistado - Segmento	Enlace
Recolección de datos	Miguel Ruiz - Empresa	URL del video de la entrevista
Recolección de datos	Mario Hinostroza - Empresa	URL del video de la entrevista
Recolección de datos	Luis Torres Arevalo - Empresa	URL del video de la entrevista
Recolección de datos	Adrián Zapata - Cliente Final	URL del video de la entrevista
Recolección de datos	Brenda Calderon - Cliente Final	URL del video de la entrevista
Recolección de datos	Gabriel Díaz - Cliente Final	URL del video de la entrevista
Validación	Henry Centurion - Empresa	URL del video de la entrevista
Validación	Miguel Ruiz - Empresa	URL del video de la entrevista
Validación	Adrián Zapata - Cliente Final	URL del video de la entrevista
Validación	Brenda Calderon - Cliente Final	URL del video de la entrevista
Validación	Gabriel Díaz - Cliente Final	URL del video de la entrevista

Anexo D Videos de Exposiciones

Entrega	Título del Video	Enlace
TB1	Exposición TB1 – Los Parkers	https://upcedupe-my.sharepoint.com/:g/personal/u202019577_upc_edu_pe/IQB_9JdDMR3Sp5hkh8r3xaVAXQKwrhKGvuz_-3lt_hqA0?nav=eyJyZWZlcnJhbEluZm8iOnsicmVmZXJyYWxBcHAiOijPbmVEcmI2ZUZvckJ1c2luZXNzliwicmVmZXJyYWxBcHBQbGF0Zm9ybSI6IlldlYiIsInJlZmVyc
TP	Exposición Trabajo Parcial – Los Parkers	https://upcedupe-my.sharepoint.com/:g/personal/u202019577_upc_edu_pe/IQAUWedvxLfdQpAtGeg8TGAoAdfKGt9f-XKRmEtmlZIQRc?nav=eyJyZWZlcnJhbEluZm8iOnsicmVmZXJyYWxBcHAiOijPbmVEcmI2ZUZvckJ1c2luZXNzliwicmVmZXJyYWxBcHBQbGF0Zm9ybSI6IlldlYiIsInJlZmVyc
TB2	Exposición TB2 – Los Parkers	https://upcedupe-my.sharepoint.com/:g/personal/u202019577_upc_edu_pe/IQDxVSyUZZKuR7Xlnqth2LifAZf5JGs8kxoifj2Dvxsm6a8?nav=eyJyZWZlcnJhbEluZm8iOnsicmVmZXJyYWxBcHAiOijPbmVEcmI2ZUZvckJ1c2luZXNzliwicmVmZXJyYWxBcHBQbGF0Zm9ybSI6IlldlYiIsInJlZmVyc
TF	Exposición Final del Proyecto – Los Parkers	https://upcedupe-my.sharepoint.com/:g/personal/u202019577_upc_edu_pe/IQA9CR29jg6nRLZT9BJ3ulVsAVgmMweq3MQe9YvznFcMt0A?e=a