

PROYECTO CHILEANTRUCKS

Sábado 8 de
Noviembre



EQUIPO DE TRABAJO
BASTIAN MOYA
FELIPE HEREDIA

DOCENTE: FERNANDO GONZALO HERRERA FRANCESCONI

SECCIÓN: 707V

EQUIPO DE TRABAJO



BASTIAN MOYA

D. full stack
Security champion



FELIPE HEREDIA

D. full stack
Compliance officer

El Viaje del Proyecto (Hasta Hoy)

Sprints 2-3: Backend (MVP 1)

Construcción de la Base de Datos (PostgreSQL) y toda la API (Django/DRF) con su lógica de negocio (JWT, Postulación).

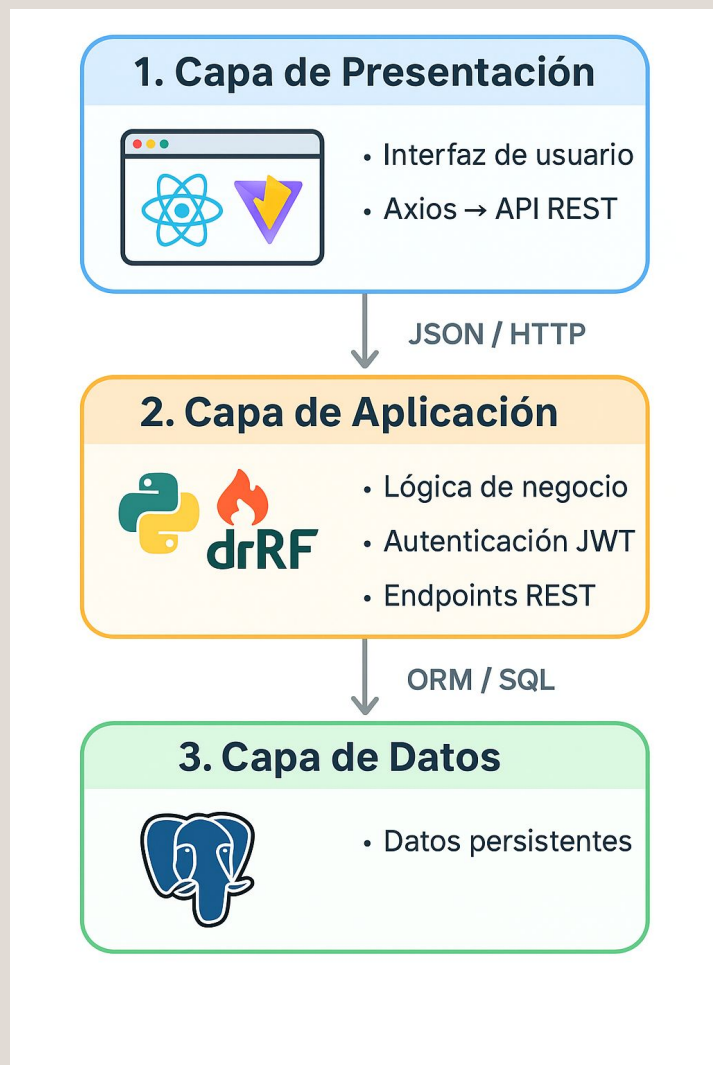
Sprints 0-1: Diseño

Planificación, Alcance, Historias de Usuario, MER, Diagramas de Flujo, Wireframes y Contrato de API.

Sprint 4: Frontend (MVP 2)

Construcción de la Interfaz de Usuario (React) y conexión completa con la API, resultando en una aplicación funcional.

Arquitectura Actual (3 Capas)



1. Frontend (Cliente)

El cliente (React.js) se ejecuta en el navegador. Es el único responsable de la UI/UX y consume la API.

2. Backend (Servidor)

El servidor (Django/DRF) procesa toda la lógica de negocio. Recibe peticiones de React y es el único que puede hablar con la BD.

3. Base de Datos (Almacén)

La base de datos (PostgreSQL) almacena de forma persistente todos los datos. Nunca es contactada directamente por el Frontend.

Objetivos del MVP 2:



Construir la UI: Desarrollar la interfaz de usuario con React para los 3 roles: Empresa, Transportista y Administrador.

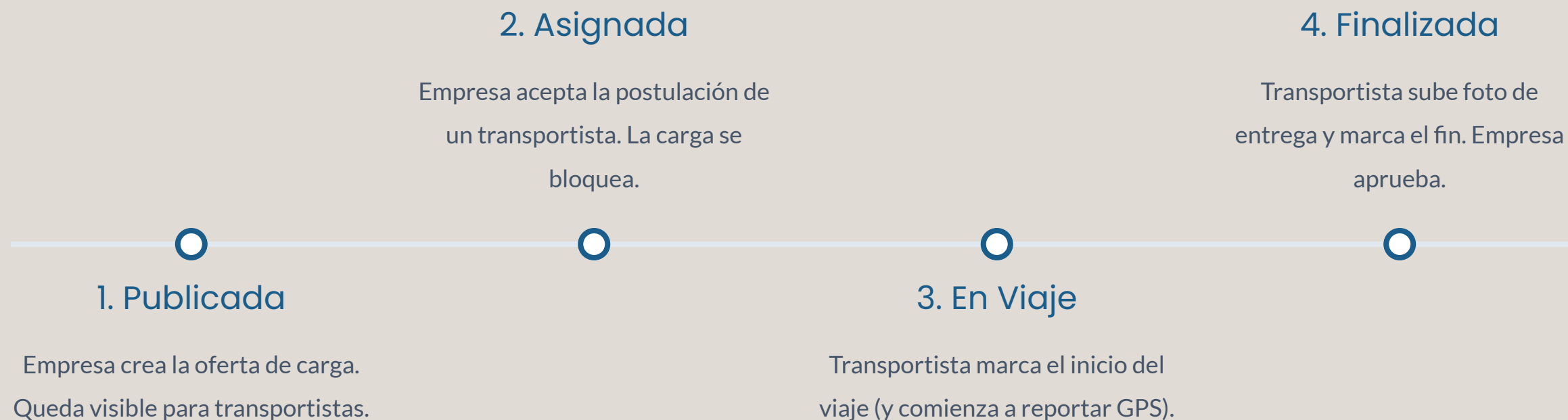


Conectar el Ecosistema: Conectar 100% el Frontend (React) con el Backend (API de Django) para crear un flujo de usuario funcional.



Integrar Servicios: Implementar las integraciones clave de UI: Mapas (Leaflet), Gráficos (Chart.js) y Geocodificación (Nominatim).

Funcionalidad Core: Ciclo de Vida de la Carga



Funcionalidades: Flujo de Empresa

Crear Carga (con Mapa)

Formulario para crear cargas conectado a la API. Integra `react-leaflet` para seleccionar origen/destino con un clic.

Geocodificación (Nominatim)

El formulario de "Crear Carga" incluye un buscador que consume la API de Nominatim para centrar el mapa en una dirección.

Dashboard de Gestión

La Empresa puede ver sus cargas publicadas y las postulaciones recibidas, con botones para Aceptar o Rechazar transportistas.

Nueva publicación

Título

Ej: Pallets de frutas

Descripción

Detalles de la carga, temperatura, fechas, etc.


Buscar dirección de Origen

Ej: Alameda 123, Santiago


Buscar dirección de Destino

Ej: Av. del Mar 500, La Serena

O haz clic en el mapa (Origen)



O haz clic en el mapa (Destino)



Comuna (detectada por el mapa)

Comuna (detectada por el mapa)

Peso (kg)

Ej: 6.000

Monto a Pagar (CLP)

Ej: 120.000

Tipo de Camión (Opcional)

Cualquiera

Fecha estimada de viaje

dd-mm-aaaa

Hora estimada de viaje

--:--

Publicar carga

Funcionalidades: Flujo de Transportista

CRUD de Camiones

El transportista puede registrar, ver, editar y eliminar sus camiones desde su perfil, conectando al CRUD de la API.

Ver y Postularse a Cargas

Dashboard para ver cargas disponibles. El botón "Postular" consume la API de "Postulación Inteligente" (validando peso/tipo).

Trazabilidad GPS (Mock)

El transportista puede ver su viaje asignado en un mapa (`MapaRuta`) y un botón para "Iniciar Viaje" (que reporta ubicación al backend).

des (Empresa: Agrosuper)

 **Origen:** No especificada
(Cerrillos)

 **Destino:** No especificada
(Ñuñoa)

 **Fecha Planificada:** 20-11-2025 (20:04)

Distancia: 13.08 km | **Peso:** 7000.00 kg **Camión Requerido:** Cualquiera

 **Iniciar Viaje**

 **En Curso**

 No tienes cargas en curso.

 **En Revisión (Esperando aprobación)**

flete (Empresa: Agrosuper)

Origen: Maipu  **Destino:** Ñuñoa

Finalizado: 07-11-2025 17:25

Esperando que la empresa confirme la entrega para liberar el pago.

Funcionalidades: Flujo Admin y Social

Dashboard Admin

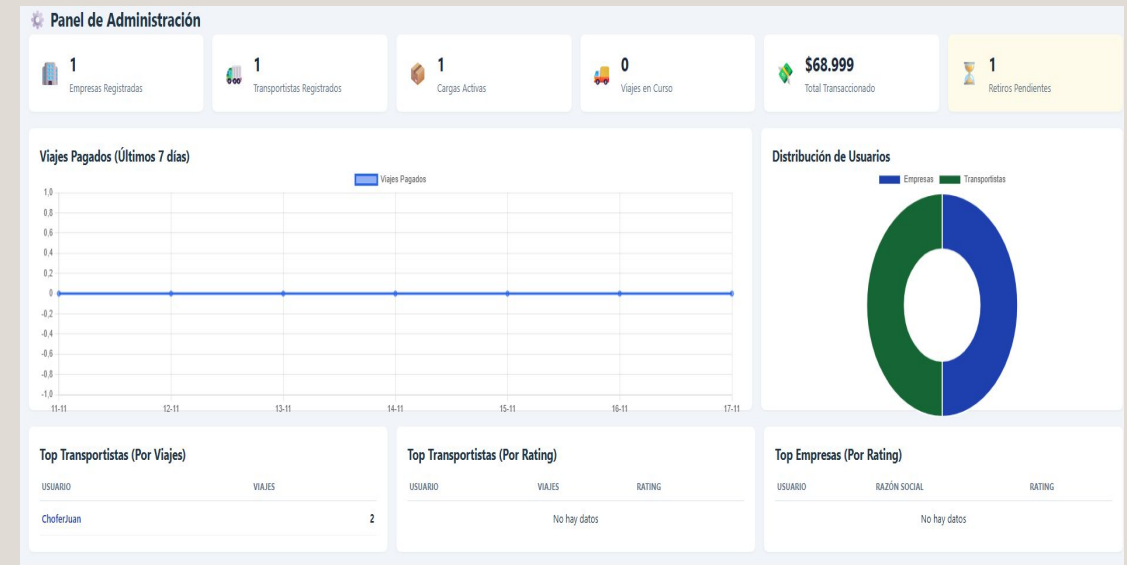
El Admin tiene un dashboard que consume el endpoint de KPIs y los muestra usando `Chart.js` (Gráficos de Usuarios y Viajes).

Perfil Público y Reseñas

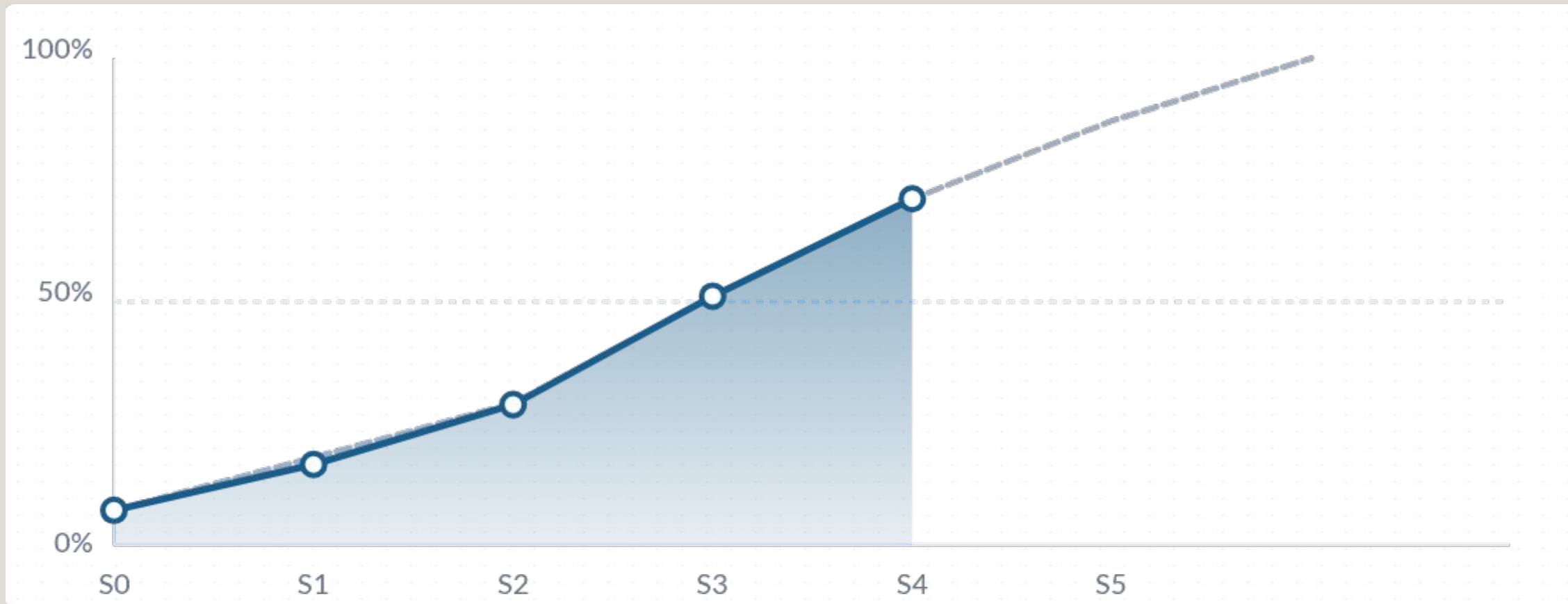
Implementada la UI para ver perfiles públicos (con foto) y el sistema de reseñas con estrellas (visual).

UI de Chat Interno

Interfaz de Chat (sin real-time) que consume la API de Mensajes para enviar y leer chats entre Empresa y Transportista.



Métricas: Cierre Sprint 4



Avance Sprint 4 (Frontend): 100% completado (9 de 9 tareas). Avance Total del Proyecto: 71.1% (32 de 45 tareas).

Próximos Pasos: Sprint 5 (QA y Seguridad)

QA y Pruebas E2E

Ejecutar pruebas End-to-End formales para los 3 flujos principales y validar casos de borde (ej. postular con camión inválido).




Seguridad Adicional

Implementar más flujos de seguridad" (Backend + Frontend) y el flujo de "Verificación de Transportista" (Admin).

Bug Fixing

Sesión de corrección de errores críticos encontrados durante las pruebas E2E antes del refinamiento final.

Conclusión de Hito MVP 2

-  **Aplicación Funcional:** El sistema ya no es solo una API; es una aplicación web funcional que conecta los 3 roles.
-  **Integraciones Clave Listas:** Mapas (Leaflet), Gráficos (Chart.js) y Geocodificación (Nominatim) están implementados y operativos.
-  **Listos para QA:** El MVP 2 está completo y listo para entrar en la fase de Pruebas de Calidad (Sprint 5).