EMPRESA LOGITRACK S.A.

Carnet 1 – Nombre completo del estudiante
201709431 – JORGE IVAN CHIC LAJUJ

Resumen

El proyecto consiste en desarrollar una plataforma web para la simulación y gestión de operaciones logísticas. La plataforma está compuesta por un backend en Spring Boot (Java) que expone una API REST y un frontend en React. El sistema carga una configuración inicial desde un archivo XML que define centros de distribución, rutas, mensajeros, paquetes y solicitudes. A partir de esta carga, el sistema permite gestionar estos elementos, procesar solicitudes de envío mediante una cola de prioridad, asignar mensajeros, y actualizar estados de los envíos. Toda la información se mantiene en memoria, sin uso de bases de datos tradicionales. Al final, el sistema puede generar un archivo XML de salida con el estado final y estadísticas de la simulación.

Palabras clave

1. Logística
2. Simulación
3. API REST
4. Spring Boot
5. React
6. XML
7. Cola de prioridad
8. Gestión de envíos
9. Centros de distribución
10. Mensajeros

Introducción

En el ámbito de la logística moderna, la capacidad de simular y gestionar operaciones de distribución es esencial para optimizar recursos y mejorar la eficiencia. Este proyecto tiene como objetivo desarrollar una plataforma web que permita simular un sistema logístico completo, desde la carga inicial de datos hasta la gestión dinámica de envíos. La solución combina un backend robusto en Spring Boot con un frontend intuitivo en React, ofreciendo una experiencia interactiva para el usuario. Mediante la carga de un archivo XML de configuración, el sistema inicializa todos los elementos necesarios (centros, rutas, mensajeros, paquetes y solicitudes) y permite su

administración a través de una API REST. La implementación sigue principios de POO y mantiene la información en memoria, garantizando coherencia y rapidez en las operaciones.

Desarrollo del tema

Arquitectura del Sistema

El sistema sigue una arquitectura cliente-servidor. El backend, desarrollado en Spring Boot, es el núcleo lógico que maneja la carga de XML, la gestión de datos en memoria y la exposición de endpoints REST. El frontend, en React, consume esta API para proporcionar una interfaz gráfica interactiva.

Componentes principales del sistema

El sistema está dividido en dos componentes principales:

Backend (Spring Boot):

Implementa la lógica de negocio, expone una API REST, valida y procesa el XML de entrada, gestiona las estructuras en memoria y aplica reglas de negocio (asignación de mensajeros, validación de rutas, gestión de estados, cola de prioridad).

Frontend (React):

Proporciona una interfaz gráfica para interactuar con el backend. Permite cargar el archivo XML, visualizar recursos, crear solicitudes, procesar envíos y observar cambios en tiempo real.

Funcionalidades Principales

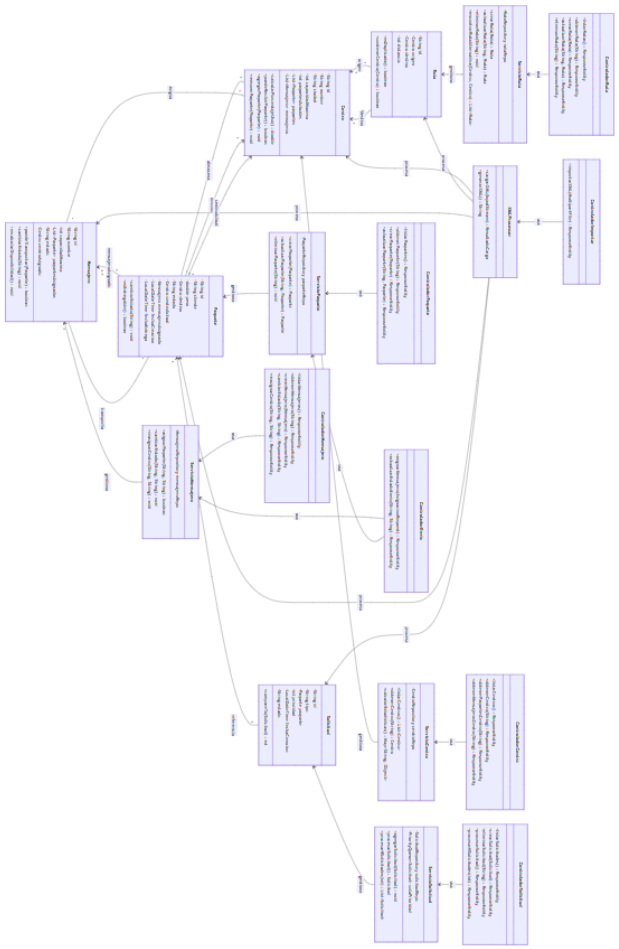
1. **Carga Inicial del Sistema:**
   - Endpoint: POST /api/importar
   - Procesa un archivo XML con la configuración inicial.
   - Valida estructura, detecta duplicados y omite elementos erróneos sin interrumpir el proceso.
   - Sobrescribe cualquier información previa en memoria.
2. **Gestión de Centros de Distribución:**
   - Endpoints: GET /api/centros, GET /api/centros/{id}, etc.
   - Consulta de capacidad, paquetes almacenados y mensajeros asignados.
   - No se permiten modificaciones desde la API.
3. **Gestión de Rutas:**
   - CRUD completo de rutas entre centros.
   - Validación de duplicados y manejo de impacto en envíos pendientes.
4. **Gestión de Mensajeros:**
   - CRUD y cambios de estado (DISPONIBLE, EN_TRANSITO).
   - Reasignación de centros con actualización de lists internas.
5. **Gestión de Paquetes:**
   - CRUD con validación de peso, existencia del centro destino y estados válidos (PENDIENTE, EN_TRANSITO, ENTREGADO).
6. **Gestión de Solicitudes (Cola de Prioridad):**
   - Procesamiento de solicitudes por prioridad (valor más alto primero).

- Validación de paquete, mensajero disponible, capacidad y ruta existente.
- Posibilidad de procesar múltiples solicitudes a la vez.

7. **Asignación Directa de Mensajero:**
   - Endpoint: PUT /api/envios/asignar
   - Asignación manual para pruebas o casos especiales.
8. **Gestión del Estado del Envío**:
   - Actualización de estados de paquete (PENDIENTE → EN_TRANSITO → ENTREGADO).
   - Registro de timestamp y liberación automática del mensajero al entregar.
9. **Generación de XML de Salida:**

- Estructura que incluye estadísticas y estado final de todos los elementos.

DIAGRAMA DE CLASES



Universidad de San Carlos de Guatemala
Escuela de Ingeniería en Ciencias y Sistemas, Facultad de Ingeniería
Introducción a la programación y computación 2, 2do. Semestre 2025

DIAGRAMA DE CARGA INICIAL DE XML

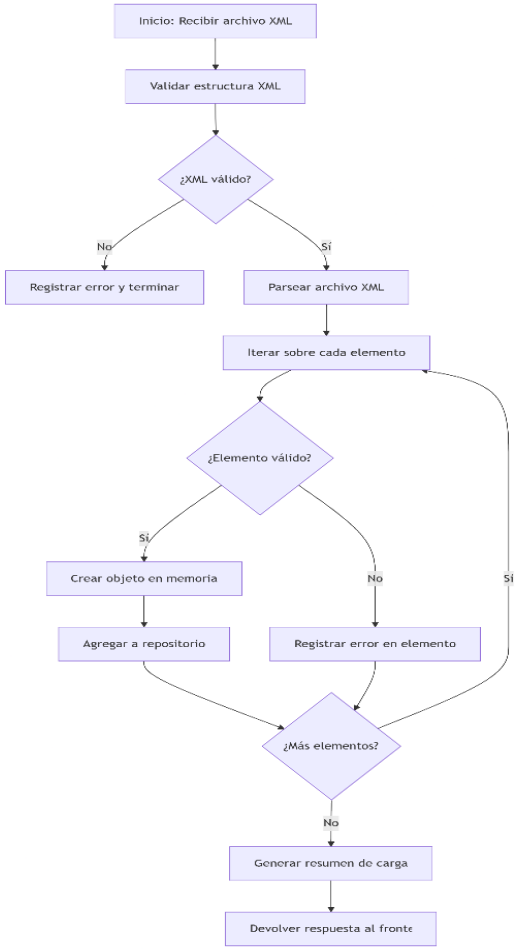
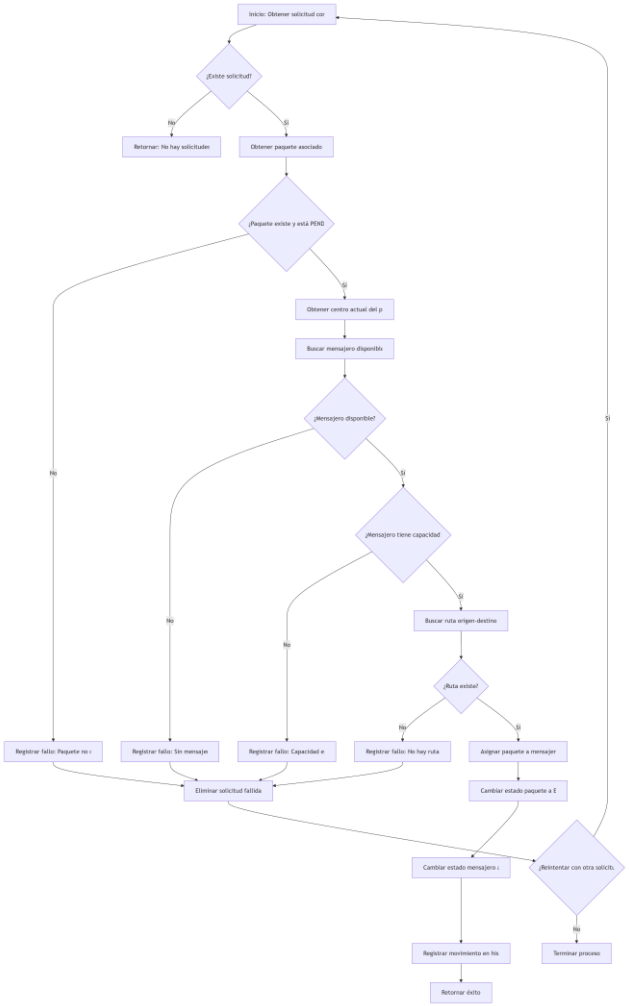


DIAGRAMA DE PROCESAMIENTO DE SOLICITUD DE ENVIO



Universidad de San Carlos de Guatemala
Escuela de Ingeniería en Ciencias y Sistemas, Facultad de Ingeniería
Introducción a la programación y computación 2, 2do. Semestre 2025

DIAGRAMA DE ASIGNACION DIRECTA DE MENSAJERO.

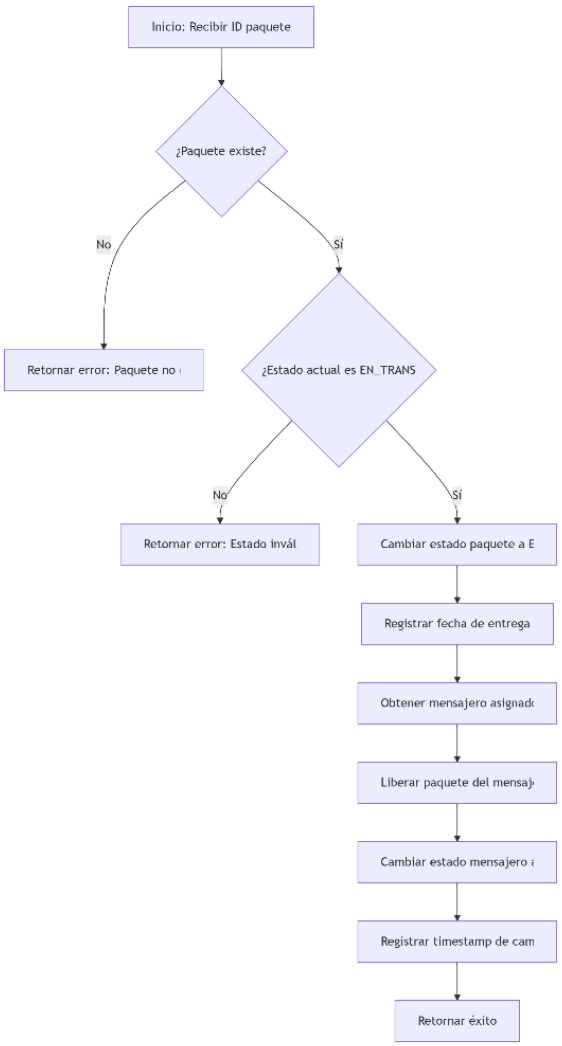
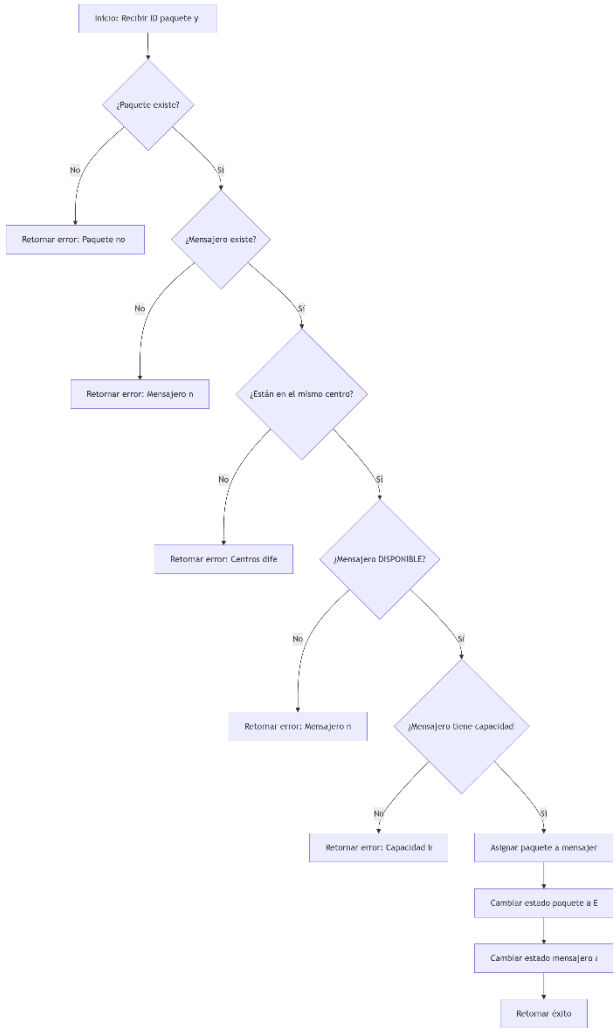


DIAGRAMA DE ACTUALIZACION DE ESTADO ENTREGADO



CONCLUSIÓN

Este proyecto integra tecnologías modernas de desarrollo web (Spring Boot, React) para construir una plataforma de simulación logística completa y funcional. La separación clara entre backend y frontend, junto con el uso de principios de POO y gestión en memoria, permite una solución escalable, mantenible y eficiente. La capacidad de procesar solicitudes por prioridad, gestionar recursos dinámicamente y generar reportes en XML hace de esta herramienta un sistema versátil para la simulación logística**.**

BIBLIOGRAGIA

Spring Boot y Java
Walls, C. (2023). Spring Boot in action (2ª ed.). Manning Publications.

Cosmina, I., Harrop, R., Schaefer, C., & Ho, C. (2020). *Pro Spring 5: An in-depth guide to the Spring Framework and its tools* (5ª ed.). Apress.

Team, S. (2024). Spring Boot reference documentation. https://docs.spring.io/spring-boot/docs/current/reference/html/

React y Frontend
Banks, A., & Porcello, E. (2023). Learning React: Modern patterns for developing React apps (3ª ed.). O'Reilly Media.