

Introducción:

El presente proyecto tiene como objetivo desarrollar un sistema de simulación de líneas de colectivos urbanas que comparten paradas de pasajeros y recorridos. El sistema permitirá cargar la información de las líneas y paradas, así como también de los colectivos utilizados en la simulación. A medida que se ejecuta la simulación, se mostrará el recorrido de cada colectivo, las paradas en las que se detiene y los pasajeros que suben y bajan.

El proyecto se divide en incrementos, con el fin de abordar gradualmente la complejidad del problema y permitir entregas incrementales en consenso con la cátedra. En el primer incremento, se enfoca en la carga de información y generación de pasajeros en las paradas. Los datos de líneas, paradas y colectivos se cargarán desde archivos de texto, y se generarán los pasajeros distribuidos en las diferentes paradas. Durante la simulación, se mostrarán las paradas por las que pasa cada colectivo y los cambios en la cantidad de pasajeros.

En el segundo incremento, se agregará a la simulación el cálculo del índice de satisfacción del cliente y la ocupación promedio de los colectivos. Estas métricas permitirán evaluar la calidad del servicio y la eficiencia en el transporte de pasajeros. Además de la aplicación del sistema, se agregará toda la documentación solicitada, que incluirá análisis de las estructuras de datos utilizadas, diagramas de clases, manual de funcionamiento y otros documentos relevantes.

Planteo del problema:

En las ciudades, el transporte público juega un papel fundamental en la movilidad de los ciudadanos. Uno de los medios de transporte más utilizados son los colectivos urbanos, los cuales siguen rutas preestablecidas y realizan paradas en distintos puntos de la ciudad para recoger y dejar pasajeros. Sin embargo, el funcionamiento eficiente de las líneas de colectivos puede ser un desafío debido a la complejidad de gestionar múltiples líneas, paradas compartidas y la demanda variable de pasajeros.

El problema planteado consiste en desarrollar un sistema de simulación de líneas de colectivos urbanas que aborde estos desafíos y permita evaluar el funcionamiento de las mismas. Se busca crear un entorno virtual que reproduzca el comportamiento de las líneas de colectivos, considerando aspectos como la carga de información de líneas y paradas, la generación de pasajeros, el recorrido de los colectivos y la interacción con los pasajeros en cada parada.

El sistema debe abordar los siguientes aspectos:

Gestión de líneas y paradas: Se debe proporcionar una manera de cargar la información de las líneas de colectivos y las paradas que forman parte de la red. Esto implica definir las rutas de cada línea, las paradas en cada ruta y los horarios de llegada y salida de los colectivos en cada parada.

Carga de datos de colectivos: Se requiere la posibilidad de cargar los datos de los colectivos que participarán en la simulación. Estos datos incluyen información sobre el número de colectivos, capacidad de pasajeros y otros detalles relevantes.

Generación de pasajeros: Durante la simulación, se deben generar pasajeros distribuidos en las distintas paradas de la red. Estos pasajeros deberán tener destinos aleatorios y serán recogidos por los colectivos en las paradas correspondientes.

Recorrido de los colectivos: Los colectivos deberán seguir las rutas definidas para cada línea y detenerse en las paradas programadas. Se debe proporcionar un mecanismo para simular el movimiento de los colectivos, indicando las paradas por las que pasan y los pasajeros que suben y bajan en cada parada.

Cálculo de métricas: Para evaluar la calidad del servicio, se debe calcular el índice de satisfacción del cliente y la ocupación promedio de los colectivos. Estas métricas permitirán medir la eficiencia en el transporte de pasajeros y la capacidad de satisfacer la demanda de manera adecuada.

Análisis de las estructuras seleccionadas:

Clases: El proyecto utiliza varias clases para representar diferentes entidades y componentes del sistema, como Colectivo, Linea, Parada, Pasajero y SimuladorColectivos. Estas clases están bien definidas y encapsulan la funcionalidad relacionada en cada una de ellas.

Listas: El proyecto hace uso de listas para almacenar colecciones de objetos de diferentes tipos, como la lista de colectivos, líneas, paradas y pasajeros. Las listas permiten almacenar y acceder eficientemente a los elementos, y se utilizan métodos de las listas, como add(), remove(), size(), entre otros, para manipular los datos.

Propiedades y archivos de configuración: Se utiliza la clase Properties para cargar la configuración del sistema desde un archivo de propiedades (config.properties). Esta configuración incluye rutas de archivos de datos para colectivos, líneas y paradas, así como información adicional, como el número total de pasajeros y recorridos.

Lectura de archivos: Se implementa la lectura de archivos de texto (colectivoFilePath, lineaFilePath, paradaFilePath) para cargar los datos de colectivos, líneas y paradas respectivamente. Se utilizan las clases FileReader y BufferedReader para leer los archivos línea por línea y procesar los datos.

Relaciones entre objetos: Las clases Colectivo, Linea, Parada y Pasajero están relacionadas entre sí. Por ejemplo, un colectivo puede tener una línea asociada y una parada actual. Además, una parada puede tener una lista de pasajeros que están esperando en ella.

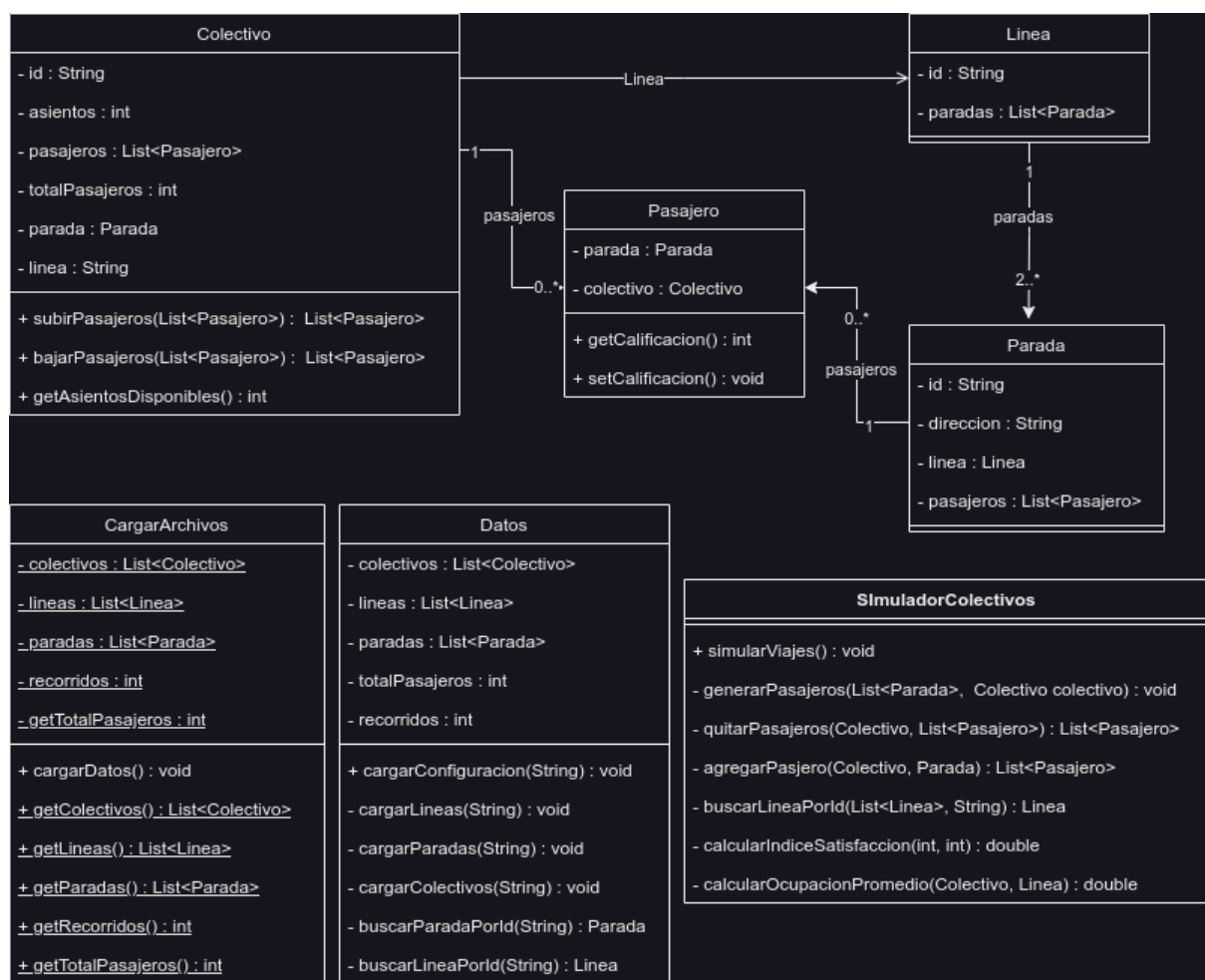
Generación aleatoria de pasajeros: Se utiliza la clase Random para generar un número aleatorio de pasajeros en cada parada y asignarlos a la lista de pasajeros de esa parada. La

cantidad de pasajeros generados se basa en el número total de pasajeros definido en la configuración.

Métodos auxiliares: El proyecto incluye varios métodos auxiliares para realizar operaciones específicas, como buscar una línea o parada por su ID, subir y bajar pasajeros de un colectivo, calcular el índice de satisfacción del cliente y el promedio de ocupación, entre otros.

Las estructuras seleccionadas en el proyecto, como clases, listas, propiedades y métodos auxiliares, permiten una representación adecuada de los componentes del sistema y el procesamiento de los datos necesarios para simular los viajes de los colectivos.

Diagrama de clases:



Implementación de la solución:

A continuación se muestra un resumen de la implementación:

Clase Colectivo:

Tiene propiedades como id, línea (referencia a la línea asociada), paradaActual, capacidad y pasajeros (lista de pasajeros a bordo).

Los métodos incluyen subirPasajero(), bajarPasajero(), moverse(), entre otros.

Clase Linea:

Tiene propiedades como id, paradas (lista de paradas en la línea) y colectivos (lista de colectivos en la línea).

Los métodos incluyen agregarColectivo(), obtenerSiguienteParada(), obtenerColectivoDisponible(), entre otros.

Clase Parada:

Tiene propiedades como id, nombre y pasajeros (lista de pasajeros esperando en la parada).

Los métodos incluyen agregarPasajero(), removerPasajero(), generarPasajerosAleatorios(), entre otros.

Clase Pasajero:

Tiene propiedades como id, origen (referencia a la parada de origen) y destino (referencia a la parada de destino).

Clase SimuladorColectivos:

Es la clase principal que inicia y controla la simulación.

Tiene propiedades como colectivos (lista de todos los colectivos), líneas (lista de todas las líneas), paradas (lista de todas las paradas) y configuracion (propiedades cargadas desde el archivo de configuración).

Los métodos incluyen cargarConfiguracion(), cargarColectivos(), cargarLineas(), cargarParadas(), iniciarSimulacion(), entre otros.

La implementación general implica cargar la configuración y los datos de colectivos, líneas y paradas desde archivos. Luego, se realizan simulaciones de viajes de colectivos, donde se mueven entre las paradas, subiendo y bajando pasajeros según las reglas establecidas.

Manual de funcionamiento (ingreso de datos, pruebas, resultados de salida):

Ingreso de datos:

Archivo de configuración: Antes de ejecutar la simulación, asegúrate de tener un archivo de configuración que contenga la información necesaria para el funcionamiento del sistema.

Este archivo puede incluir detalles como la cantidad de colectivos, líneas y paradas, capacidades de los colectivos, nombres de las paradas, etc. Asegúrate de que la estructura del archivo de configuración sea compatible con el formato esperado por el código.

Datos de pasajeros: Si el sistema permite la generación aleatoria de pasajeros en las paradas, asegúrate de tener un mecanismo para proporcionar estos datos, ya sea ingresándolos manualmente o generándolos automáticamente a partir de ciertos criterios.

Pruebas:

Carga de configuración: El primer paso es cargar la configuración del sistema desde el archivo correspondiente. Esto puede implicar leer el archivo de configuración y extraer la información necesaria para crear colectivos, líneas y paradas, así como establecer las conexiones entre ellos.

Inicio de la simulación: Una vez cargada la configuración, puedes iniciar la simulación llamando al método o función correspondiente. Esto podría ser una función llamada `iniciarSimulacion()` o similar. La simulación puede tener una duración predeterminada o continuar hasta que se cumpla una condición de salida específica.

Monitoreo y seguimiento: Durante la simulación, puedes monitorear el estado de los colectivos, líneas y paradas para asegurarte de que el sistema funcione correctamente. Esto puede implicar imprimir o registrar información relevante en la consola o en un archivo de registro.

Resultados de salida:

Estadísticas y métricas: Al finalizar la simulación, puedes recopilar y calcular diversas estadísticas y métricas para evaluar el rendimiento del sistema. Esto puede incluir el número de pasajeros atendidos, el tiempo promedio de espera de los pasajeros, la ocupación promedio de los colectivos, etc.

Informes y visualización: Puedes generar informes o gráficos que muestran los resultados obtenidos durante la simulación. Estos informes pueden ayudar a identificar posibles mejoras o problemas en el sistema.

Errores detectados (si existe algún error y bajo qué condiciones se produce):

El código presenta varios errores:

Data.java:

Hay un error de dependencia al cargar las listas desde el archivo. En la línea 37 se repite la función de `cargarLineas()` porque en la primera llamada no tiene las paradas asociadas. Y si llamase `cargarParadas()` primero, las paradas no tienen líneas porque no se llamó `cargarLineas()` primero. Las paradas recién se cargan al llamar `cargarParadas()` después de las líneas.

SimuladorColectivos.java:

Los pasajeros generados no respetan la cantidad de pasajeros establecidos en `config.properties`.

Los pasajeros generados pueden bajar del colectivo cuando este recién arranca, y subir cuando termina el recorrido. Esto se da porque no hay condiciones al generar aleatoriamente, que al iniciar el recorrido no pueden bajar pasajeros y que al final no pueden subir.

No se tomó en cuenta los pasajeros que quedaron en las paradas para el índice de satisfacción.

Después de dos paradas, la cantidad de pasajeros que suben son iguales a los que bajan. El número de pasajeros que quedan en las paradas no quedan guardados, para que después suban al siguiente colectivo de la misma línea.

Los recorridos son en función de cuantas paradas tiene cada línea. No se tomó en cuenta la variable de recorridos en config.properties.

Lotes de prueba:

Colectivo ID: C1

Línea: L1

Asientos disponibles: 48

Recorridos restantes: 18

Paradas visitadas:

- Parada ID: P4, Dirección: 28 De Julio, 200

Pasajeros que bajaron: 19

Pasajeros que subieron: 0

Pasajeros en la parada: 19

- Parada ID: P10, Dirección: Avenida Domec Garcia Norte, 239

Pasajeros que bajaron: 21

Pasajeros que subieron: 0

Pasajeros en la parada: 21

- Parada ID: P35, Dirección: Francisco Luis Albarracin, 2250

Pasajeros que bajaron: 23

Pasajeros que subieron: 0

Pasajeros en la parada: 23

- Parada ID: P36, Dirección: General Mosconi, 100

Pasajeros que bajaron: 25

Pasajeros que subieron: 0

Pasajeros en la parada: 25

- Parada ID: P16, Dirección: Bartolome Mitre, 610

Pasajeros que bajaron: 29

Pasajeros que subieron: 0

Pasajeros en la parada: 29

- Parada ID: P53, Dirección: Mariano Moreno, 193

Pasajeros que bajaron: 59

Pasajeros que subieron: 0

Pasajeros en la parada: 59

- Parada ID: P36, Dirección: General Mosconi, 100

Pasajeros que bajaron: 0

Pasajeros que subieron: 0

Pasajeros en la parada: 0

- Parada ID: P22, Dirección: Comodoro Martin Rivadavia, 602

Pasajeros que bajaron: 38

Pasajeros que subieron: 0

Pasajeros en la parada: 38

- Parada ID: P36, Dirección: General Mosconi, 100

Pasajeros que bajaron: 0

Pasajeros que subieron: 0

Pasajeros en la parada: 0

- Parada ID: P29, Dirección: Espana, 1450

Pasajeros que bajaron: 45

Pasajeros que subieron: 0

Pasajeros en la parada: 45

- Parada ID: P32, Dirección: Espana, 928

Pasajeros que bajaron: 47

Pasajeros que subieron: 0

Pasajeros en la parada: 47

- Parada ID: P30, Dirección: Espana, 1660

Pasajeros que bajaron: 53

Pasajeros que subieron: 0

Pasajeros en la parada: 53

- Parada ID: P24, Dirección: Dorrego, 416

Pasajeros que bajaron: 22

Pasajeros que subieron: 0

Pasajeros en la parada: 22

- Parada ID: P23, Dirección: Comodoro Martin Rivadavia, 750

Pasajeros que bajaron: 32

Pasajeros que subieron: 0

Pasajeros en la parada: 32

- Parada ID: P15, Dirección: Bartolome Mitre, 401

Pasajeros que bajaron: 40

Pasajeros que subieron: 0

Pasajeros en la parada: 40

- Parada ID: P15, Dirección: Bartolome Mitre, 401

Pasajeros que bajaron: 0

Pasajeros que subieron: 0

Pasajeros en la parada: 0

- Parada ID: P36, Dirección: General Mosconi, 100

Pasajeros que bajaron: 0

Pasajeros que subieron: 0

Pasajeros en la parada: 0

- Parada ID: P42, Dirección: Julian Murga, 950

Casteglione Matias - Trabajo Integrador

Pasajeros que bajaron: 6
Pasajeros que subieron: 0
Pasajeros en la parada: 6

Índice de satisfacción del cliente: 14.814814814814813%
Ocupación promedio: 12.206988058381247%
Colectivo ID: C2
Linea: L1

Asientos disponibles: 48
Recorridos restantes: 18

Paradas visitadas:

- Parada ID: P29, Dirección: Espana, 1450

Pasajeros que bajaron: 24
Pasajeros que subieron: 0
Pasajeros en la parada: 24

- Parada ID: P32, Dirección: Espana, 928

Pasajeros que bajaron: 4
Pasajeros que subieron: 0
Pasajeros en la parada: 4

- Parada ID: P4, Dirección: 28 De Julio, 200

Pasajeros que bajaron: 44
Pasajeros que subieron: 0
Pasajeros en la parada: 44

- Parada ID: P59, Dirección: Roberto Gomez, 1017

Pasajeros que bajaron: 57
Pasajeros que subieron: 24
Pasajeros en la parada: 33

- Parada ID: P35, Dirección: Francisco Luis Albarracin, 2250

Pasajeros que bajaron: 0
Pasajeros que subieron: 0
Pasajeros en la parada: 0

- Parada ID: P35, Dirección: Francisco Luis Albarracin, 2250

Pasajeros que bajaron: 0
Pasajeros que subieron: 0
Pasajeros en la parada: 0

- Parada ID: P24, Dirección: Dorrego, 416

Pasajeros que bajaron: 7
Pasajeros que subieron: 0
Pasajeros en la parada: 7

- Parada ID: P23, Dirección: Comodoro Martin Rivadavia, 750

Casteglione Matias - Trabajo Integrador

Pasajeros que bajaron: 15
Pasajeros que subieron: 0
Pasajeros en la parada: 15

- Parada ID: P53, Dirección: Mariano Moreno, 193
Pasajeros que bajaron: 11
Pasajeros que subieron: 0
Pasajeros en la parada: 11

- Parada ID: P29, Dirección: Espana, 1450
Pasajeros que bajaron: 0
Pasajeros que subieron: 0
Pasajeros en la parada: 0

- Parada ID: P10, Dirección: Avenida Domec Garcia Norte, 239
Pasajeros que bajaron: 25
Pasajeros que subieron: 0
Pasajeros en la parada: 25

- Parada ID: P53, Dirección: Mariano Moreno, 193
Pasajeros que bajaron: 0
Pasajeros que subieron: 0
Pasajeros en la parada: 0

- Parada ID: P24, Dirección: Dorrego, 416
Pasajeros que bajaron: 0
Pasajeros que subieron: 0
Pasajeros en la parada: 0

- Parada ID: P22, Dirección: Comodoro Martin Rivadavia, 602
Pasajeros que bajaron: 7
Pasajeros que subieron: 0
Pasajeros en la parada: 7

- Parada ID: P16, Dirección: Bartolome Mitre, 610
Pasajeros que bajaron: 34
Pasajeros que subieron: 0
Pasajeros en la parada: 34

- Parada ID: P16, Dirección: Bartolome Mitre, 610
Pasajeros que bajaron: 0
Pasajeros que subieron: 0
Pasajeros en la parada: 0

- Parada ID: P50, Dirección: Marcos A. Zar, 1998
Pasajeros que bajaron: 57
Pasajeros que subieron: 46
Pasajeros en la parada: 11

- Parada ID: P39, Dirección: Juan B. Justo, 1172
Pasajeros que bajaron: 58
Pasajeros que subieron: 18
Pasajeros en la parada: 40

Posibles mejoras y extensiones:

Clase Colectivo:

- Implementar métodos para permitir la reserva de asientos en el colectivo.
- Agregar un mecanismo para rastrear la ubicación actual del colectivo en tiempo real, por ejemplo, mediante el uso de GPS.
- Incorporar lógica adicional para el cálculo de tarifas y la gestión de pagos de los pasajeros.
- Permitir la programación y gestión de horarios de los colectivos.

Clase SimuladorColectivos:

- Mejorar la representación visual de la simulación utilizando una interfaz gráfica de usuario (GUI).
- Agregar la capacidad de generar informes detallados sobre el rendimiento de los colectivos y la satisfacción de los pasajeros.
- Implementar un algoritmo más sofisticado para la generación de pasajeros en las paradas, teniendo en cuenta factores como la demanda, la hora del día y las preferencias de los pasajeros.
- Incorporar la posibilidad de simular eventos inesperados, como retrasos en el tráfico o cambios en las condiciones climáticas.

Clase Datos:

- Mejorar la eficiencia de la carga de datos implementando técnicas de almacenamiento en caché o indexación para agilizar las búsquedas de líneas y paradas.
- Añadir validaciones y manejo de errores para garantizar la integridad de los datos cargados desde los archivos.
- Ampliar la capacidad de cargar datos desde diferentes fuentes, como bases de datos o servicios web.
- Implementar la capacidad de exportar los datos cargados a diferentes formatos, como archivos CSV o JSON.

Conclusiones:

A lo largo de este documento, he presentado un sistema de simulación de un sistema de transporte público basado en colectivos. Este sistema ha sido diseñado para modelar y evaluar el funcionamiento de las líneas de colectivos, la demanda de pasajeros y los tiempos de viaje en una red de transporte.

Desarrollé un modelo que simula el movimiento de los colectivos entre paradas, la generación de pasajeros en cada parada y la asignación de pasajeros a los colectivos. Además, incorpora factores como la capacidad limitada de los colectivos y la posibilidad de que los pasajeros no puedan abordar debido a la falta de espacio.

Mediante la simulación, he obtenido datos sobre los tiempos de viaje promedio, la ocupación de los colectivos y la cantidad de pasajeros que no pudieron abordar debido a la capacidad limitada. Estos datos nos han permitido evaluar el desempeño del sistema y realizar análisis comparativos entre diferentes escenarios.

En base a los resultados obtenidos, se puede concluir que el sistema de transporte público basado en colectivos presenta ciertas limitaciones y desafíos. Los tiempos de viaje pueden ser significativamente afectados por la capacidad limitada de los colectivos, lo que puede resultar en una mayor espera para los pasajeros y una menor eficiencia del sistema.

Además, he observado que la asignación aleatoria de pasajeros a los colectivos puede resultar en una distribución desigual de la carga entre ellos. Esto puede llevar a una mayor ocupación en algunos colectivos y una menor utilización de otros, lo que puede afectar la comodidad de los pasajeros y la eficiencia global del sistema.

Sin embargo, también he identificado oportunidades de mejora y extensiones para el sistema. Algunas posibles mejoras incluyen la implementación de algoritmos de enrutamiento más eficientes, la consideración de eventos y condiciones especiales, la implementación de estrategias de asignación más sofisticadas y la incorporación de consultas y seguimiento en tiempo real.