

# Sistemas Distribuidos 1 TP N° 1: Bike Rides Analyzer



1° Cuatrimestre, 2023

Apellido y Nombre	Email
Sabatino, Gonzalo	gsabatino@fi.uba.ar

## Índice

<b>1. Objetivos y Estructura del Documento</b>	<b>2</b>
<b>2. Diagramas</b>	<b>2</b>
2.1. Supuestos . . . . .	2
2.2. Diagrama de Robustez . . . . .	2
2.3. Diagrama de Despliegue . . . . .	3
2.4. Diagramas de Actividades . . . . .	4

## 1. Objetivos y Estructura del Documento

El objetivo del presente documento es realizar una documentación inicial de la aplicación, presentando los primeros diagramas.

No se presentan un diagrama forma (Vistas 4+1 o C4) puesto que estos diagramas servirán para validar las ideas iniciales, y así proceder a esquemas formales (que serán presentados en el próximo informe).

En las siguientes secciones se analizan sobre los diagramas:

- Supuestos utilizados.
- Diagrama de Robustez.
- Diagrama de Despliegue.
- Diagramas de Actividades.

## 2. Diagramas

### 2.1. Supuestos

- El cliente envía los datos en orden de la siguiente manera: Stations, Weather, Trips. Es decir, envía primero la información considerada estática de las tres ciudades, y luego ingesta los datos sobre los viajes realizados.
- El cliente podrá empezar a consultar el estado de las queries luego de cargar los datos estáticos.
- El servidor puede tener cargado en memoria los datos estáticos (como una side table) para poder unir con los datos de los viajes cuando lo crea oportuno.

### 2.2. Diagrama de Robustez

En la Figura 1 se puede observar el diagrama.

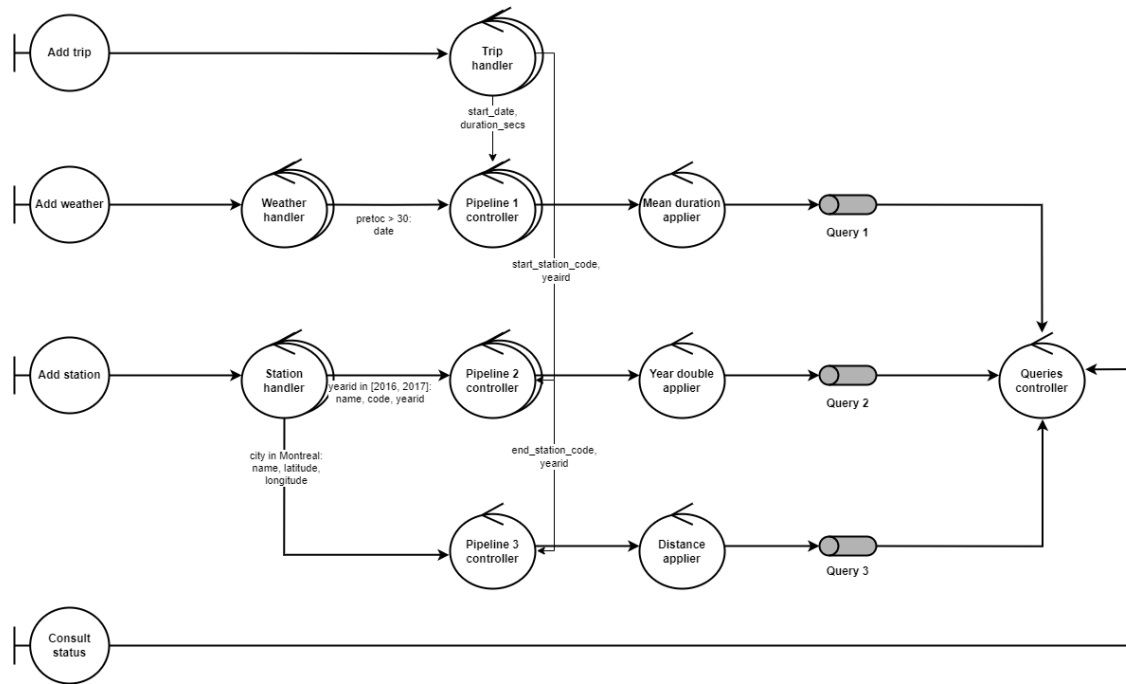


Figura 1: Diagrama de Robustez de la aplicación

### Consideraciones

- *Trips Handler*, *Weather Handler* y *Stations Handler* formatean los datos: Pasan las fechas a un tamaño correcto, eliminan las columnas innecesarias y redirigen los datos al pipeline correcto.
- Si se carga la información estática dividida por ciudades (para optimizar consultas), entonces *Pipeline 1 Controller* y *Pipeline 2 Controller* pueden contar con paralelismo. Cabe destacar que en este primer diseño no se tiene la certeza de si se utilizará o no dicho modelo.
- La consulta que realiza el cliente (*Consult Status*) devuelve un mensaje de error si no se tienen resueltas las tres queries. En caso de que sí se hayan resuelto, se devuelven dichas queries.

### 2.3. Diagrama de Despliegue

En la Figura 2 se puede observar el diagrama.

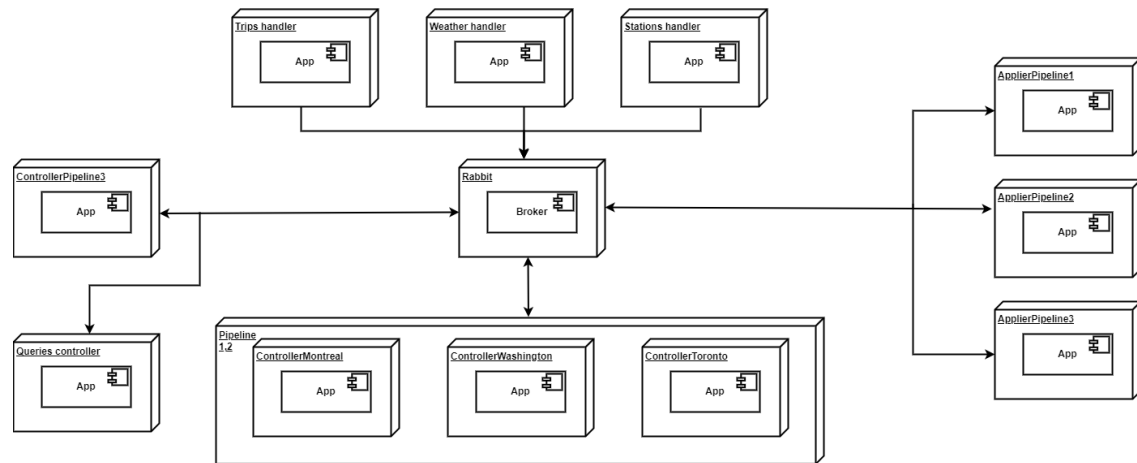


Figura 2: Diagrama de Despliegue de la aplicación

### Consideraciones

- Se aprecia cómo los nodos desplegados del servidor interactúan a través del broker RabbitMQ.
- Los nodos desplegados por ciudad de los controladores pueden ser desplegados en la misma máquina física o no, esto se determinará más adelante.
- Los manejadores (*Handlers*) se muestran separados, aunque se analizará si conviene tenerlos desplegados en distintas máquinas o en la misma máquina pero en distintos procesos.

### 2.4. Diagramas de Actividades

En las dos Figuras 3 y 4 se muestran los diagramas de actividades propuestos.

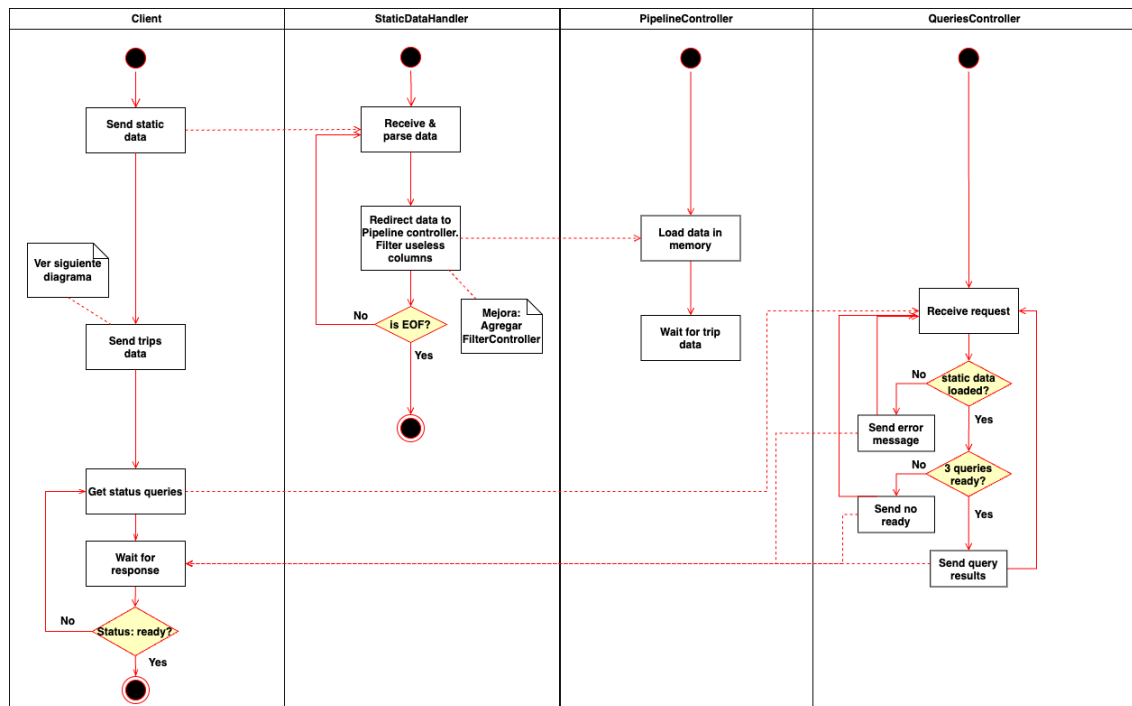


Figura 3: Diagrama de Actividades: Flujo de envío de datos estáticos y recibo de queries

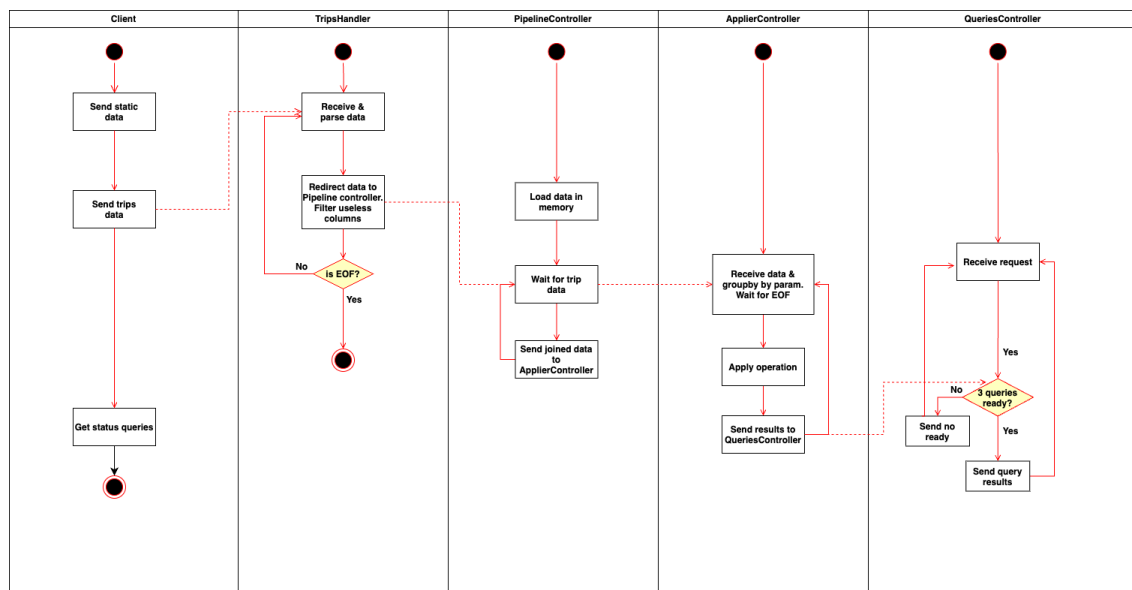


Figura 4: Diagrama de Actividades: Flujo de envío de viajes y procesamiento de queries