Estilo *arquitectónico Cliente-servidor y microservicios* con tres ramas principales: Kafka REST API, HTTP REST, WebSockets.

***WAF***: Un Web Application Firewall (WAF) protege de múltiples ataques al servidor de aplicaciones web en el backend. La función del WAF es garantizar la seguridad del servidor web mediante el análisis de paquetes de petición HTTP / HTTPS y modelos de tráfico.

***LB***: Un balanceador de carga se encarga de regular la operación de los servicios y le da fiabilidad y disponibilidad.

***RDBMS***: Un sistema de gestión de bases de datos relacionales.

***KAFKA***: Permite que los servicios se comuniquen entre sí de forma eficiente, escalable y durable. Se utiliza para comunicar eventos entre los microservicios. Los datos de supply y demand se procesan en los siguientes servicios:

\*Hadoop Hive: Este servicio se utiliza para almacenar los datos de supply y demand de forma estructurada.

\*ML Fraud Detection Routing: Este servicio utiliza aprendizaje automático para detectar fraudes.

\*Maps Creation ETA Calculator: Este servicio crea mapas y calcula las ETAs("estimación de tiempo de llegada") para los viajes.

\*Pricing Surging: Este servicio ajusta los precios de los viajes en función de la demanda.

\*Spark and Storm: Estos servicios se utilizan para procesar datos en tiempo real.

\*Analytics using ELK and Jupyter: Estos servicios se utilizan para realizar análisis de datos.

***HTTP REST***: Esta rama se utiliza para proporcionar una interfaz RESTful a los datos de supply y demand.

***WEBSOCKETS***: Esta rama se utiliza para proporcionar notificaciones en tiempo real a los usuarios.

***BACKUP DATACENTER***: funciona como copias de seguridad en caso de errores y eso le entrega una mayor seguridad a los datos y tolerancia a fallos.

***LOGSTASH***: es una herramienta que se utiliza para recopilar y analizar logs.

***ELK***: es una plataforma de análisis de logs que se utiliza para visualizar los logs de Kafka.

***DISCO:*** es una base de datos escalable que almacena los datos de supply y demand y por ello hay más de un servidor que la respalda.

El conductor y el cliente mandan datos que pasan por WAF y LB para consumir los microservicios de Kafka api, http rest y web sockets. En Kafka se manipulan los datos para la toma de decisiones y en WebSocket se almacenan y gestionan los datos para las notificación e informes.