**Front**

**App Usuario**

Debe permitir al usuario elegir la ubicación del servicio, la cual debe ser traducida en coordenadas. Cuando sepan en que formato enviaran las coordenadas por favor comunicarlo al equipo de David, Lacho y Felipe. El aplicativo debe quedar en espera de notificación o en consulta constante al servicio de “Solicitud de Servicio”. Por favor coordinar con Juan Pastrana y Gonzalo a su debido momento.

**App Conductor**

La aplicación debe permitir conocer la ubicación del servicio, de ser posible usar el mismo componente de mapas de la aplicación de usuario o en su defecto simplemente mostrar las coordenadas donde se debe realizar el servicio

**Servicios**

**Generar factura**

El servicio debe generar un texto plano formateado el cual represente la factura que se le va a enviar al usuario. La generación de esta factura debe ser asíncrona, por lo tanto debe existir un mecanismo de encolamiento de peticiones. Este servicio debe tener conexión a una cola, o storage escuchando peticiones y debe guardar las facturas en algún storage para archivos. Usar un servicio de blob storage de algún proveedor cloud y alguna cola. Sería interesante si usan algún framework para programación reactive.

**Enviar correo**

Este es un servicio genérico que envía correos electrónicos. El servicio debe funcionar de manera asíncrona, por lo tanto debe existir un mecanismo de encolamiento de peticiones.

Este servicio debe estar conectado a una cola de mensajería escuchando mensajes recomiendo hacerlo en nodejs y con alguna cola de mensajería simple como la de Azure o amazon.

**Pagar**

Este servicio representa, realizar un pago con una pasarela de pago se debe simular que el pago se realiza, este servicio debe ser síncrono y debe almacenar información del pago realizado en cualquier tipo de storage. Recomiendo un storage NoSQL como Azure Table, Firebase, Dynamo DB,Document Db etc.

**Calcular costo**

Este servicio recibe una distancio y(o) tiempo y calcula el costo de la tarifa basado en parametrización de la base de datos. Es posible que deban realizar un Servicio adicional para calcular primero la distancia o delegar esa responsabilidad al front (versión 1). Usen una bd NoSQL (Firebase), donde se almacena la información de tarifas. Prefiero que este servicio sea síncrono.

**Calcular tiempo**

Recibe coordenadas y usando algún api externa o lógica de uds calcula el tiempo, Se debe revisar si este servicio funcionará síncrono o asíncrono, todo depende del comportamiento de la aplicación web, Hablar con el equipo de App Usuario. Prefiero que sea Asincrono, usar un lenguaje que sea muy veloz como GO o Python.

**Solicitar Servicio**

Es el servicio core de la aplicación, se encarga de realizar las solicitudes de servicio, almacenarlas e informarle al servicio de gestión de conductor que el conductor ya no se encuentra disponible, este servicio recibe peticiones de forma asíncrona y luego le informa a los clientes o lo clientes están preguntándole todo el tiempo acerca de su petición. Trabajar en esto con los equipos front.

Recomiendo usar cola de mensajería, storage NoSQL(firebase o document db) para la data.

**Gestionar conductor**

Este servicio se encarga de ubicar conductores disponibles , o de marcar los conductores como no disponibles, usar una bd NoSql con datos precargados ejm (Google cloud Firebase, Document DB, Table Storage Dynamo) .

**Autenticación**

Esta será la entrada a la aplicación y el encargado de almacenar los usuarios y validar su ingreso al sistema. Por el momento solo enfoquémonos en un login y almacenamiento de usuarios sin tener en cuenta la seguridad para todos los servicios. Recomiendo usar Active Directory B2C y permitir usar al menos un conector genérico como Gmail.

**Notificaciones clientes Moviles o Front**

Bueno esta parte no estoy my seguro sería super que el app funcione en móviles pero eso depende de los equipos de front, hablar con ellos, en caso de que sí se soporte sería bueno enviar notificaciones a los celulares, para esto se pueden usar el servicio de Azure Notification Hub. https://azure.microsoft.com/en-us/services/notification-hubs/ , En caso de que no sea viable entonces ralizar un servicio que sea capaz de hacer push a clientes web como SignalR.