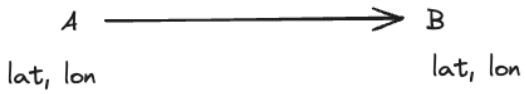
Architecture Of Uber

Uber Service - Ride Matching Service

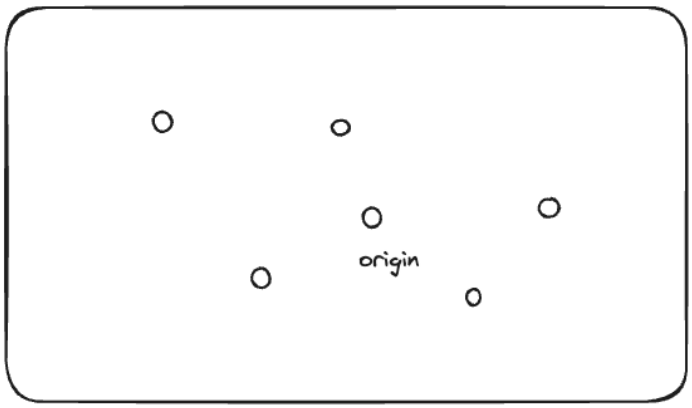
Features:

- User should be able to request a ride from A to B
- Nearby drivers should be prompted with a request and they can accept or reject it.
- Based on the ride, we should be able to calculate approximate fare
- May be some ETA related data we can show

Location is represented in the form of latitude and longitude



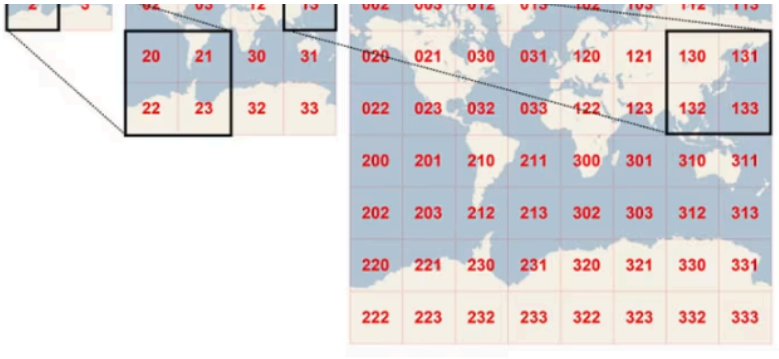
Finding K nearest drivers :

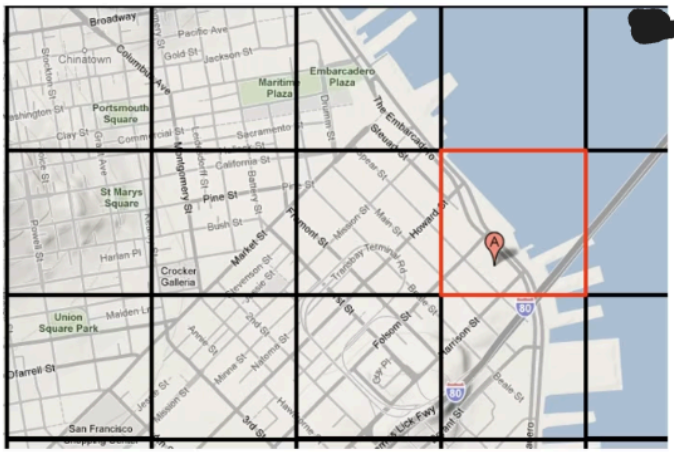


Earth -> 2D Plane

- we divide the plane into multiple squares, and these squares are hierarchal in nature.

GeoHashing -> Quadtrees

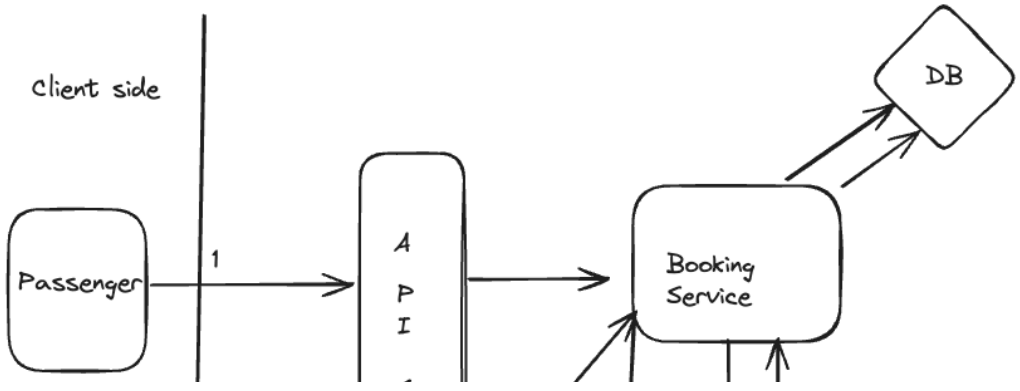


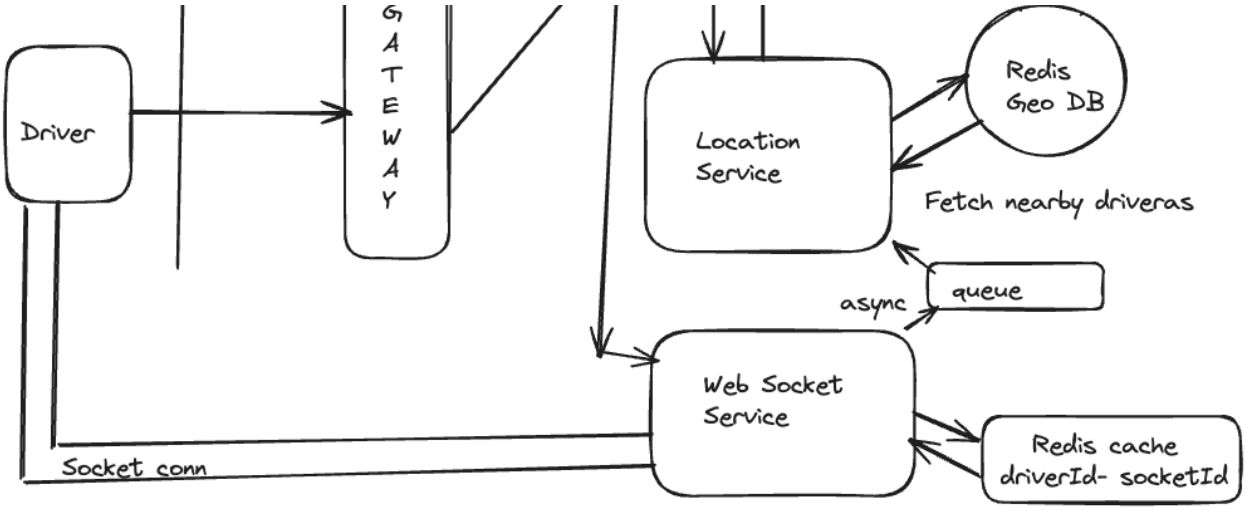


Redis Geospatial
POSTGis
MongoDB

Uber H3 - Hexagonal

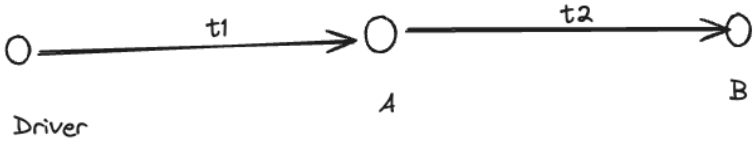
Initiated
Driver Assigned
RIde Started
Cancelled
COmpleted





Min Base Fare + Ride Fare

t1+t2 -> ETA



- Use 3rd party Map service like GMaps/ Ola Maps

Graph DB -> Dijkstra, A* ,, ... Shortest path