Lyfter is an app-based personal vehicle system (i.e., ride-hailing), and they approach your team for help in designing a system that allows them to efficiently match riders to drivers. They want your team to help them claim “the wait time is never more than 5 minutes”, and thus in determining how many drivers they should recruit if they anticipate 100 riders at any instant of time.

Initial Questions -

* Where are the drivers deployed from
  + Central location, or current location
* Area of service for users
  + Anywhere and everywhere (hardest way)
  + A certain city
    - In this case it would probably be best to determine the largest city in the US and find the distance from one end to the other so that we can get the maximum travel distance for a city which will help set up the validation process (choosing the 100 hypothetical passenger locations within that distance)
  + Only places within 5 minutes of the central location (cheap way)

Subsystems

* Passenger location database
  + We can populate this with 100 random locations from the criteria above initially then graduate to making a system to add users to the database at runtime
* Driver location database
  + If this is based on a central location this will not be needed but I assume we wont go that route
* Matching Algorithm
  + Match each user with a driver such that travel time for the distance between the driver and passenger is no greater than 5 minutes if possible, if not possible minimize all travel times.
    - If the amount of passengers outweighs the number of drivers then the driver who will be closest to that person with the shortest current travel time should be assigned to them.
    - We could start timers automatically when matches to view the behavior at run time
* User Interface (would make if we have time)
  + Simple
    - Login as passenger or driver
      * cache a name[ip or other] and location
      * Input info into appropriate database as passenger or driver
    - Output to screen the mapping
      * Perhaps show location of the driver and passenger like uber, but that is it
* System Interface/Database
  + Holds the mapping of drivers to passengers and any other relevant information and displays it currently