Building an Easy-to-use Ride-sharing App for HK

Threeriders

Supervisor: Dr. Zhiyi Huang
Second Examiner: Dr. Hubert T.H. Chan

FYP18028
Group members: Leung Hon Man, Lau Chun Yin, Lau Chi Ho

THE UNIVERSITY OF HONG KONG
DEPARTMENT OF COMPUTER SCIENCE

Abstract

In recent years, ride-sharing has been an alternative to public transportation or private car rental for commuters in countries around the world. It is believed that ride-sharing is a way to alleviate traffic congestion and reduce air pollution caused by vehicles. However, Hong Kong, as a world-class city which is suffering from traffic congestion, has not exploited the benefits of ride-sharing. One possible reason for this is the absence of an extensively used ride-sharing mobile application in Hong Kong up to this date.

In an effort to increase the popularity of ride-sharing in Hong Kong, this project aimed to design and implement a fully-featured handy ride sharing mobile application for Android and IOS platform and related web services which adapted matching algorithms from cutting-edge research. Two algorithms were implemented and evaluated using real-world test cases and a virtual grid world simulator in this project.

Technology Used

Frontend:

React Native Expo

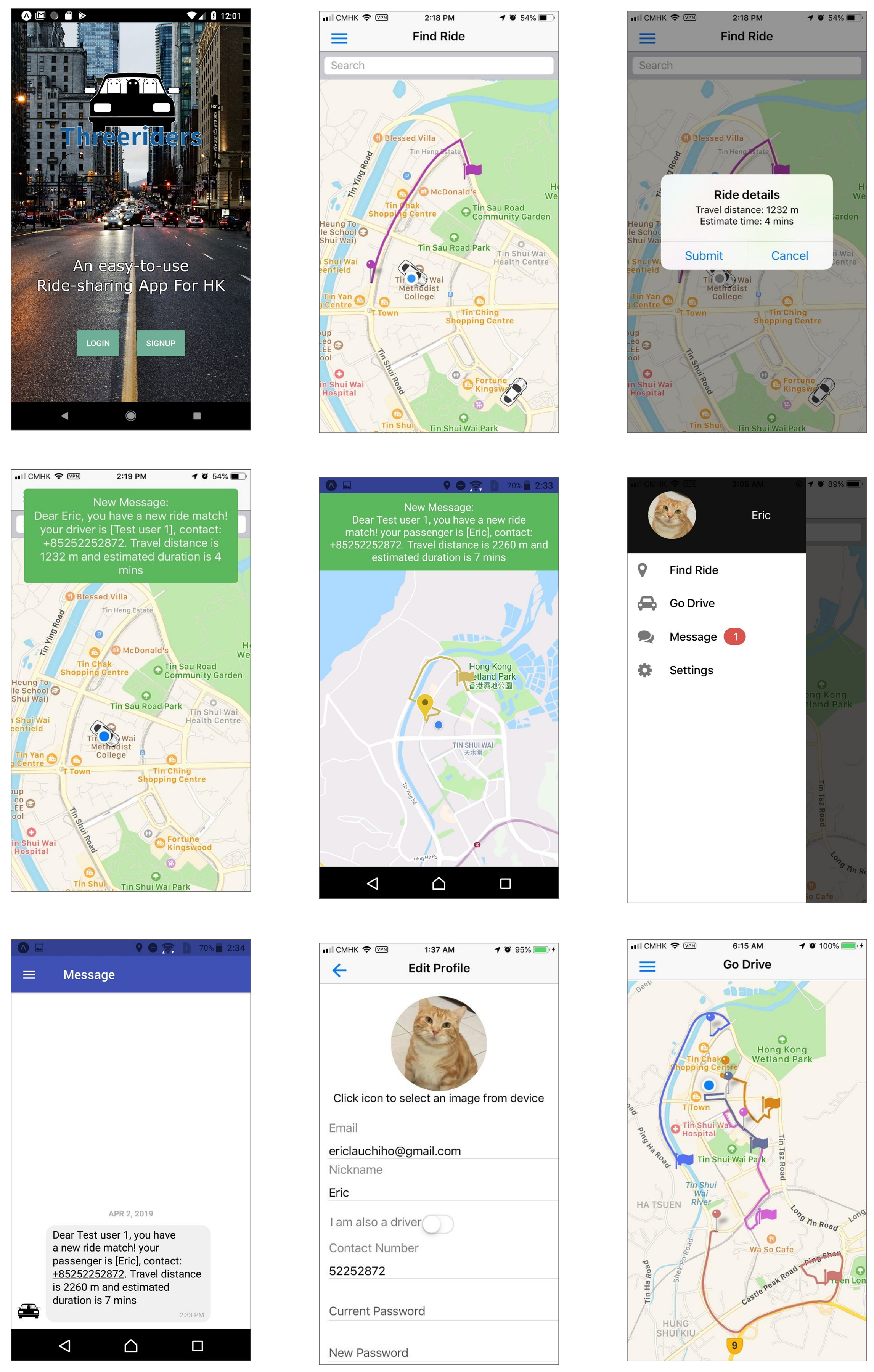
Backend:

node JS Express JS python

Database:

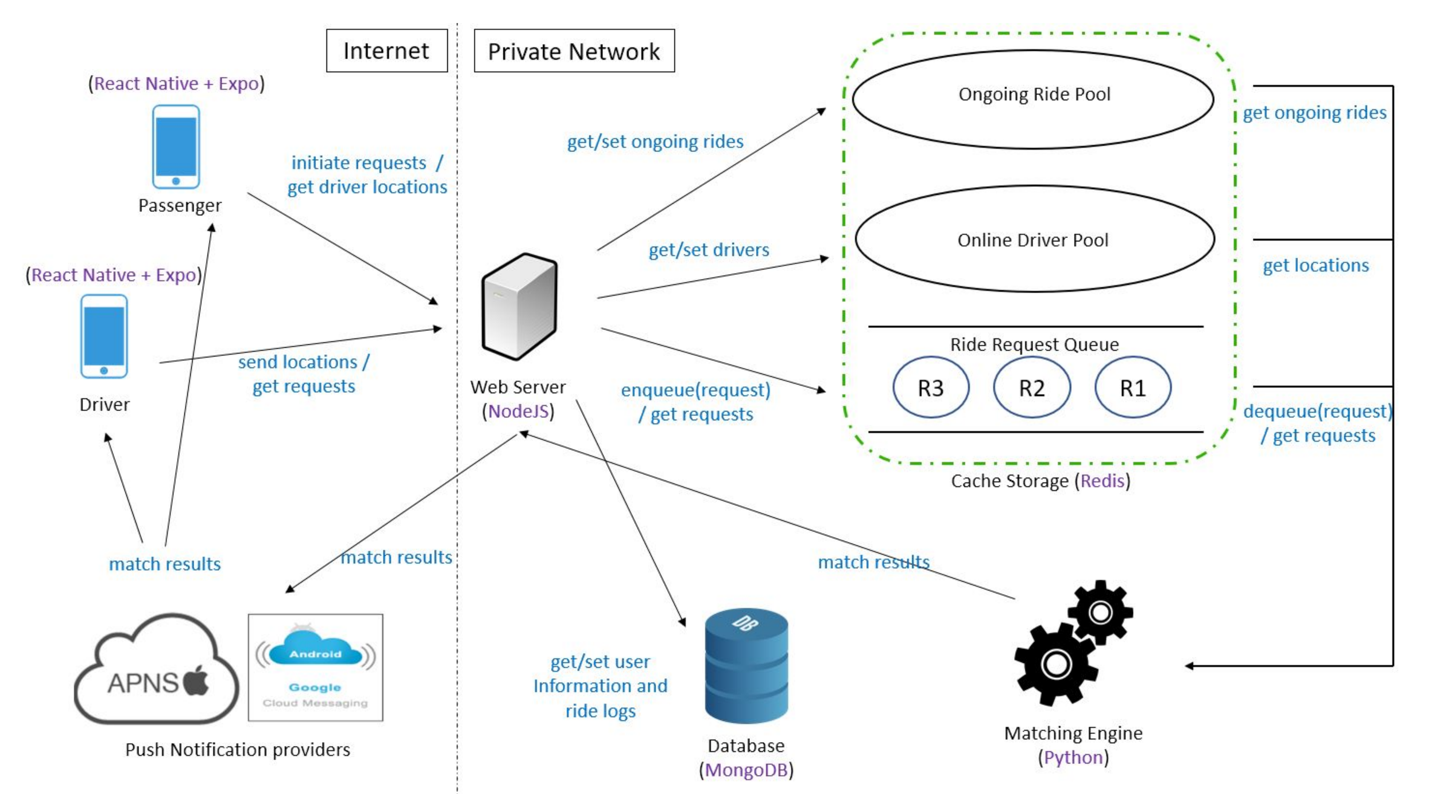
redis mongoDB

Screenshots of Our Mobile App



Methodology

System Architecture



Algorithm 1: Simple Greedy Algorithm

Input:

Requests R

Vehicles V

Contraints Z

Output:

Requests R' that cannot be matched where $R' \subseteq R$

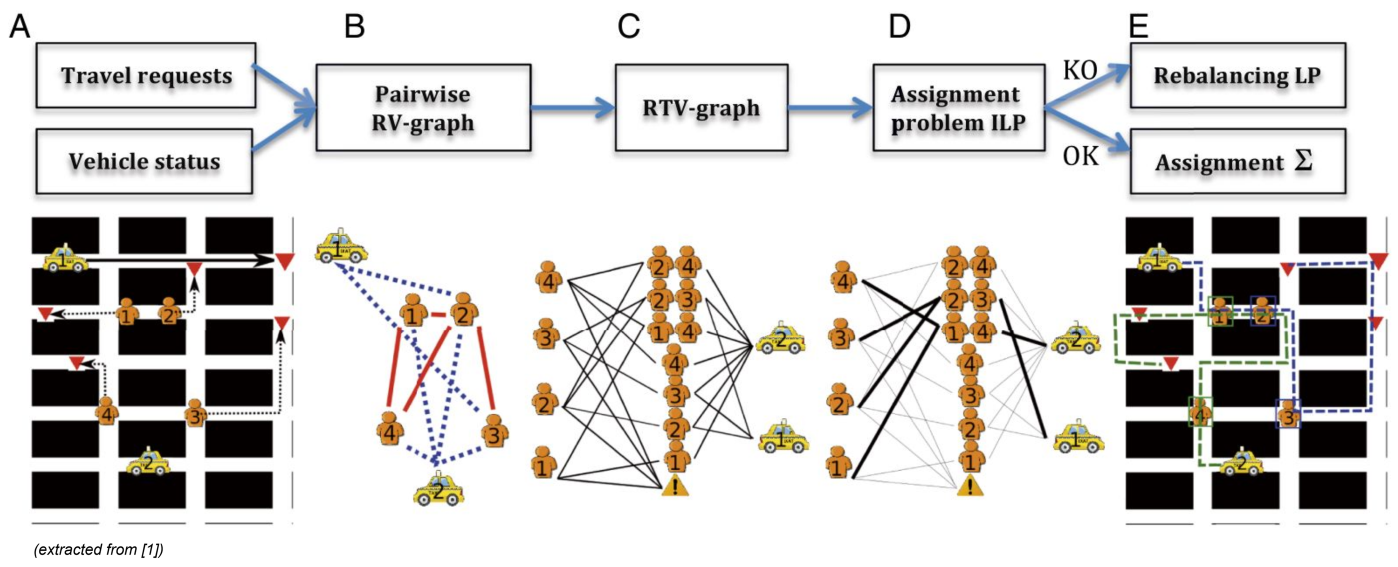
Match results $M = \{ m_1, m_2, \ldots, m_n \}$ where $m = (r, v)$

Steps:

1. $C \leftarrow \{ (r, v) \mid r \in R \wedge v \in V \}$
2. Sort C in asc order by $distance(r.origin, v.location)$
3. $M \leftarrow \{ c : c \in C \wedge r \text{ and } v \text{ satisfy } Z \text{ where } c = (r, v) \}$
4. $R' \leftarrow \{ r : r \in R \wedge r \notin \{ r' : (r', v') \in M \} \}$
5. Return M, R'

$$Z = \{ r \text{ is sharable with } v,\ v \text{ is not full}, \text{ straight line distance of } r.origin \text{ and } v.location < 1\ km \}$$

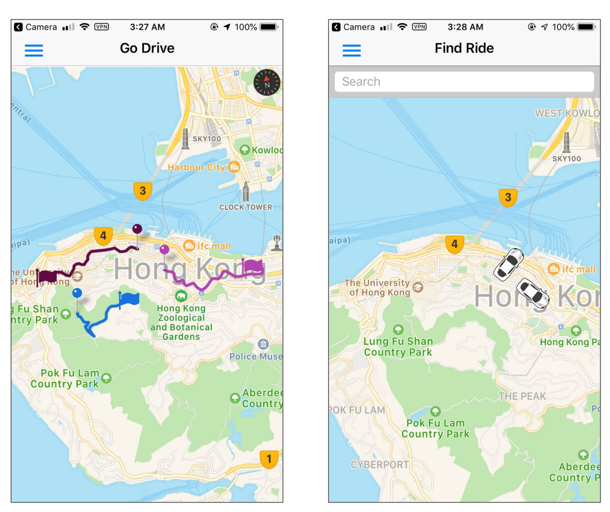
Algorithm 2: Dynamic Algorithm[1]



(extracted from [1])

Experiments and Results

A Real World Case Study

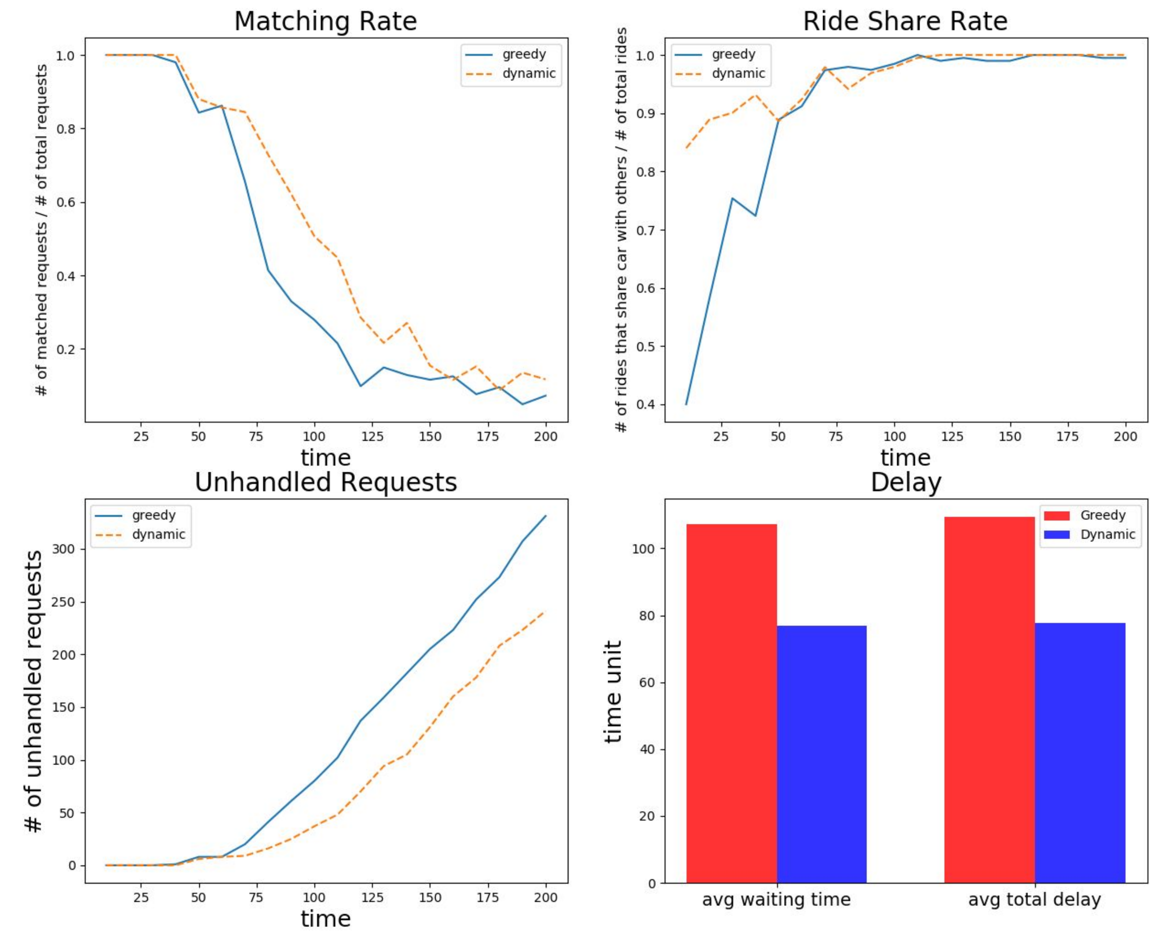


Results

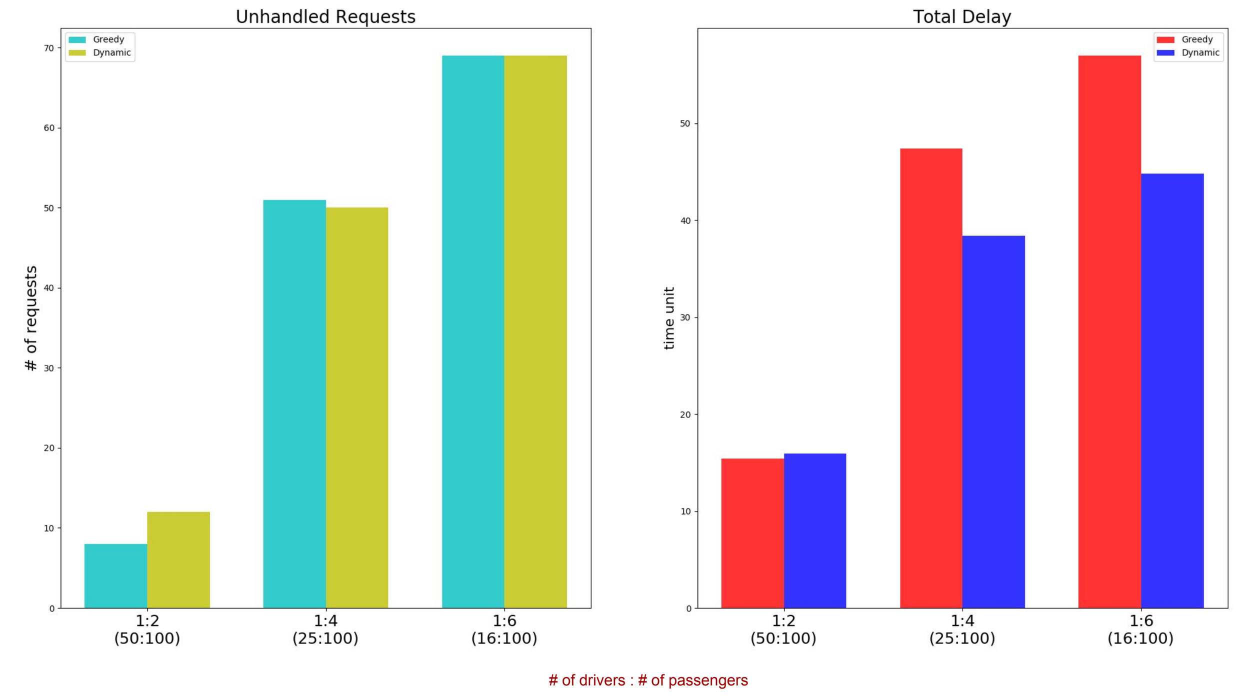
Greedy Algorithm:
R1->D1

Dynamic Algorithm:
R1->D2
R3->D1

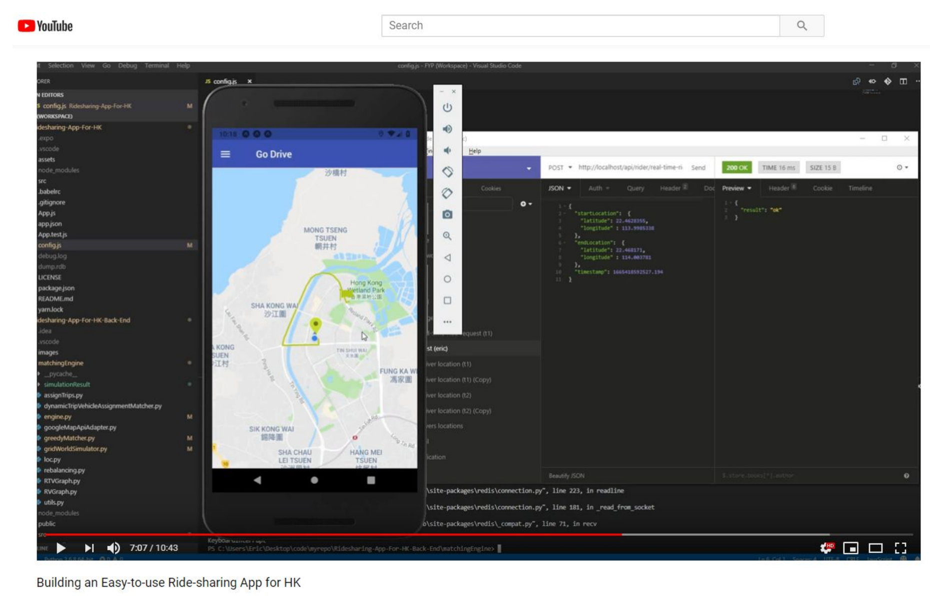
Grid World Simulator: Peak Traffic Time



Grid World Simulator: Benchmarking



Demonstration Video



Conclusion and Future Works

The results show that the dynamic algorithm has a better performance in general compared to the greedy algorithm. This project has created a free-of-change and usable ride-sharing app.

In the future, the mobile application can be fine-tuned to production ready and be published to iOS App Store and Android Play Store. In addition, the UI/UX of the mobile application and the time efficiency of the matching algorithm can also be improved.

References

[1] J. Alonso-Mora et al. (2017). On-demand high-capacity ride-sharing via dynamic trip-vehicle assignment. Proc Natl Acad Sci USA 114(3):462–467.
[2] Santi P, et al. (2014) Quantifying the benefits of vehicle pooling with shareability networks. Proc Natl Acad Sci USA 111(37):13290–13294.
[3] Our Mobile App GitHub Repository: https://github.com/eric19960304/Ridesharing-App-For-HK
[4] Our Back-end GitHub Repository: https://github.com/eric19960304/Ridesharing-App-For-HK-Back-End