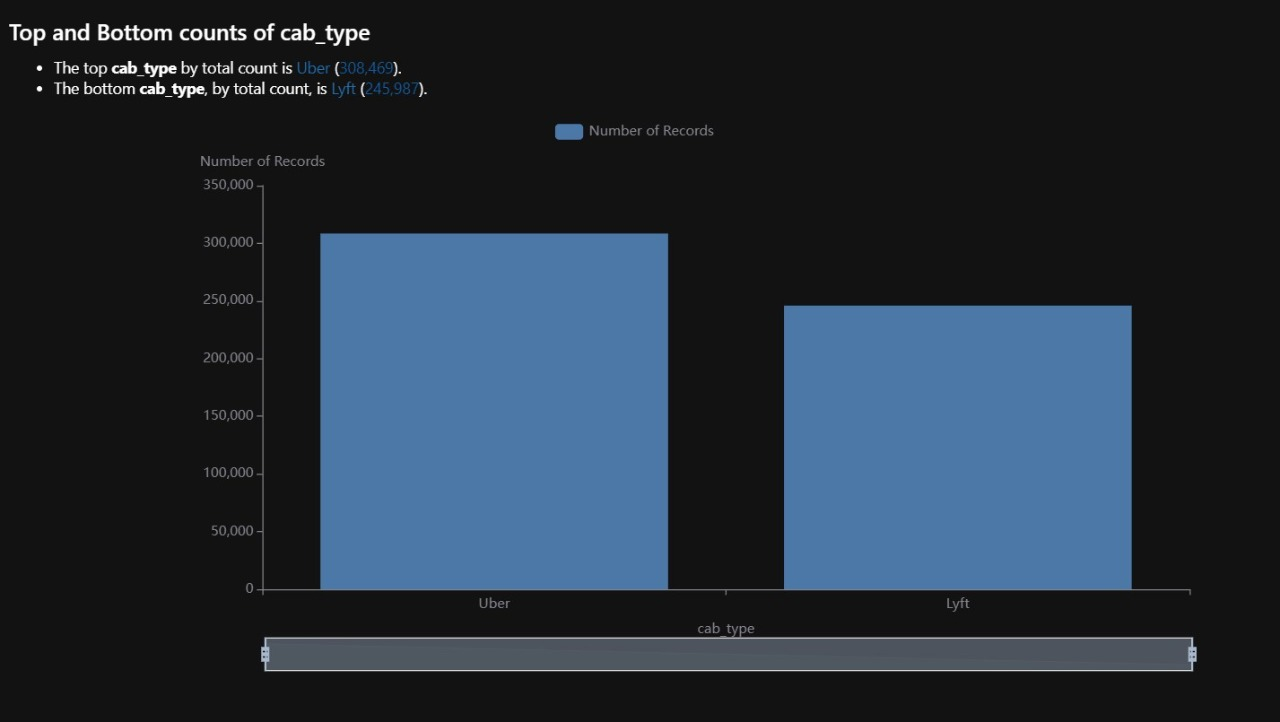
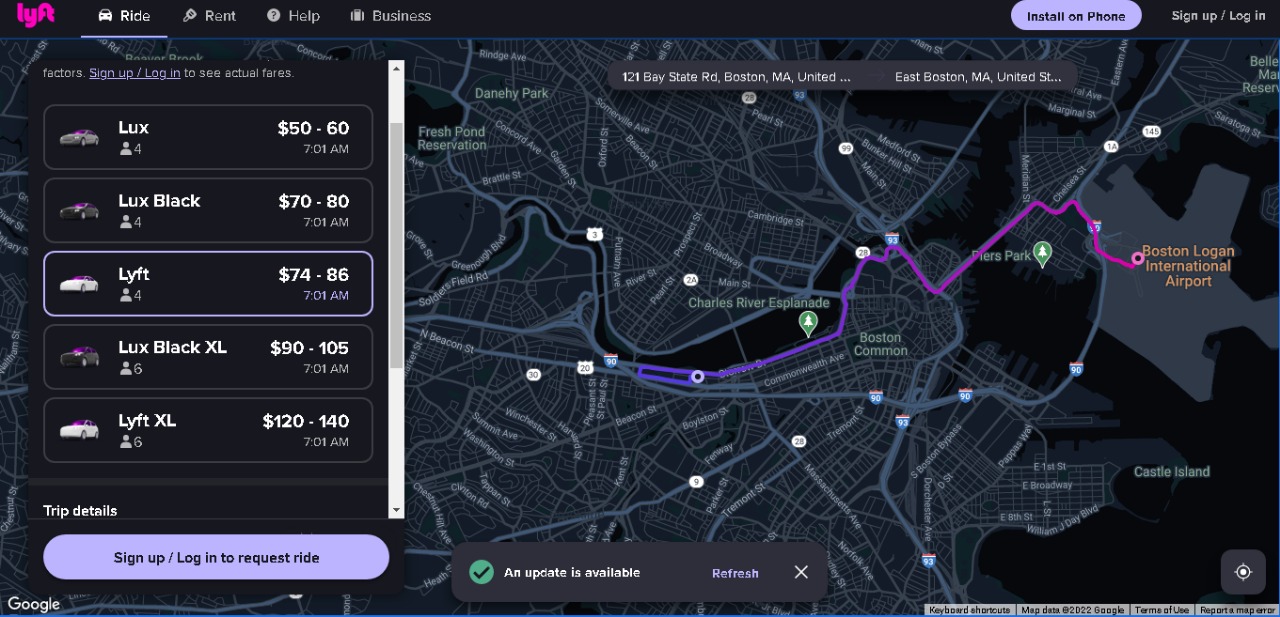
Phase 1

* **Pre-processing:**
  + **“taxi\_rides”:**

****

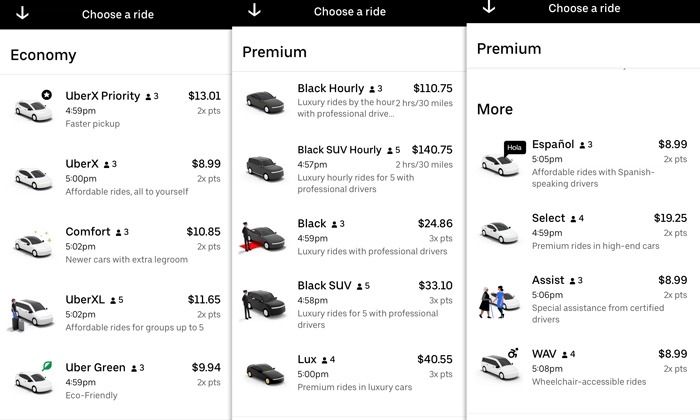
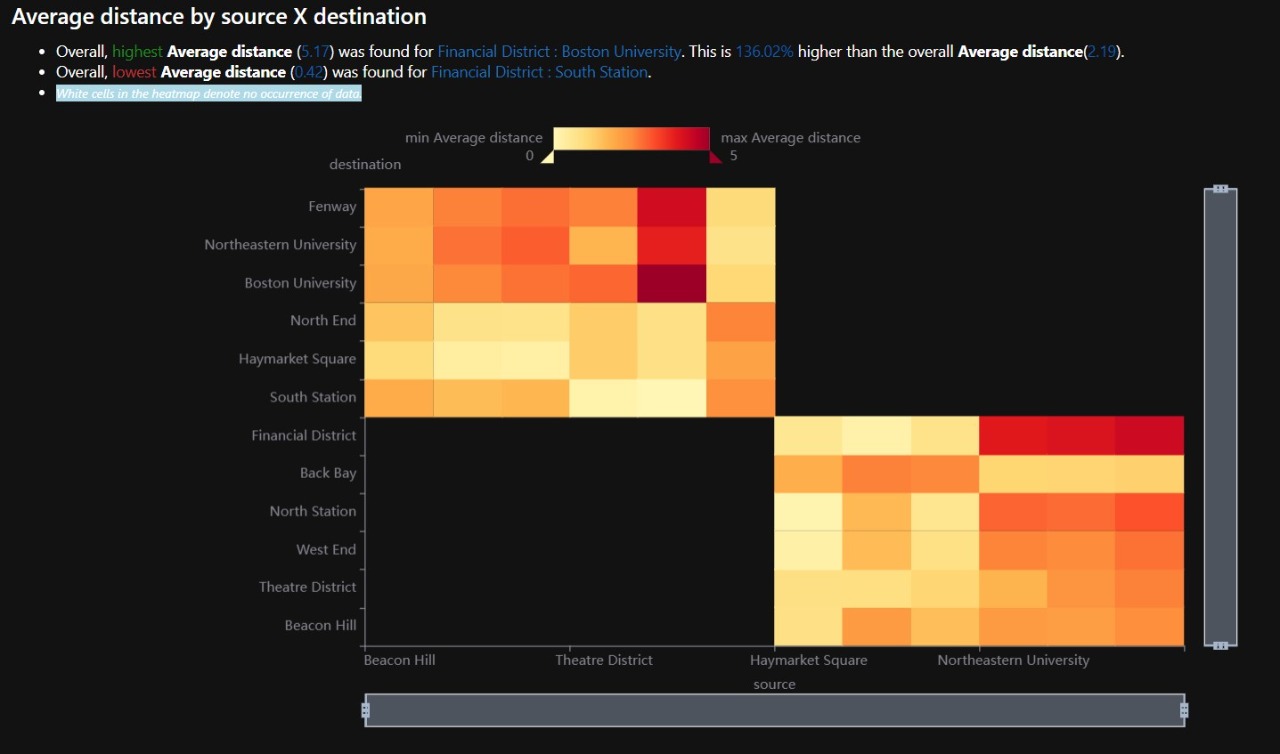
* + “cab\_type” is object, it’s either “uber” or “lyft”, so one hot
  + “time\_stamp” is float and it indicates time so it’s converted to datetime.
  + “price” contains null values in taxi, so it’s calculated by getting the price of the trips that are done by uberX which has the same source and destination.
  + “time\_stamp” was replaced by “date” and “hour” and was dropped later.
  + “id” values are unique, so it doesn’t help in prediction of the “price”. It will be dropped later after merge step.
  + “lyft” has 6 product IDs, uber doesn’t have any. Count of “name” is identical to count of “product\_id”, so a map was made to map “product\_id” to “name.
  + “product\_id” was dropped as “name” can replace it.
  + For “lyft” cabs, names were compared to see how they affect the price,we got the description from official company website:<https://help.lyft.com/hc/ru/articles/115012927427-Lyft-ride-modes-overview> and the order is:
    - Shared: Share a car with riders headed in the same direction at a discounted price.
    - Lyft: Standard Lyft car for up to 3\* riders
    - Lyft xl: SUV for up to 5\* riders
    - Lux: Luxury car for up to 3\* riders
    - Lux black: Premium black car service with leather seats for up to 3\* riders
    - Lux black xl: Premium black SUV with leather seats for up to 5\* riders
  + Ordinal encoding was done based on this order on “lyft\_types”
  + Price differences between the Uber ride types: The cost of your Uber ride is largely determined by the Uber service that you select.

The costs of the different services, from least expensive to most expensive: Uber Pool, Uber X, Uber Comfort, Uber XL, Uber Select, Uber Black, Uber SUV.

An example Uber from The Grove to the Century City Mall in Los Angeles

Note: This ride is 4.6 miles and 18 minutes. The price may change due to traffic, time of day, or discounts

Uber Ride Type Est. Ride Cost (4.6 miles, 18 minutes)

* + - Pool $9-$11
    - X $9-$12
    - Comfort $12-$16
    - XL $15-$20
    - Select $24-$30
    - Black $30-$40
    - Black SUV $42-$52
  + For “uber” cabs, names were compared to see how they affect the price, the order is:
    - UberPool
    - Taxi
    - UberX
    - UberXl
    - WAV
    - Black
    - Black SUV
  + Ordinal encoding was done based on this order on “uber\_types”
  + “cab\_type” for both “lyft” and “uber” were dropped as their values were encoded in “lyft\_type” and “uber\_type”
  + In “distance” there were outliers, so they were removed it .
  + **In “distance“: we got all street name from {data/Boston/street\_name website } using web scraping then we got the latitude and longitude of every street name using google.geoglocation api , so we can calculate distance easily between any source and destination.**
  + **“weather”:**
  + “time\_stamp” is float, so it’s converted to datetime.
  + For “rain” 85% of the data is null, K-Nearest Neighbors imputation method was used. Normalizing data was applied in order not to generate biased replacement for the missing values.
  + Regression techniques:
    - Polynomial regression
    - Multiple regression
  + For Multiple regression:
    - Mean Square Error 16.33271939415199
    - r2\_score: 75.16022240596294 %
    - Training time: 0.10199832916259766 seconds
    - For Polynomial regression: Degree : 6
    - Mean Square Error 3.203520780009517
    - r2\_score: 95.94640692535074 %
    - Training time 0.12099742889404297 seconds
  + Features used for regression are:
    - Distance
    - surge\_multiplier
    - Lyft\_Type
    - Uber\_Type
    - cab\_type\_Uber
* Training set size is 60% and testing set size is 40%

How data affect each other



* How data affect each other after feature selection



* Data visualization for “distance” outliers:

Chart, box and whisker chart

Description automatically generated

* Conclusion:
  + In this phase, we applied preprocessing for all features and feature selection and we have concluded that weather is not a good indicator for price and the product id (name) & distance was the main indicators for price. But for surge it was very effective in calculating the price for Lyft cabs but not for Uber cabs .

Phase 2

* **Pre-processing:**

**A picture containing graphical user interface

Description automatically generated**

**We do the same preprocessing as we did in phase 1 and we noticed that**

**Ridge Category in order of:**

**1: unknown**

**2: cheap**

**3: moderate**

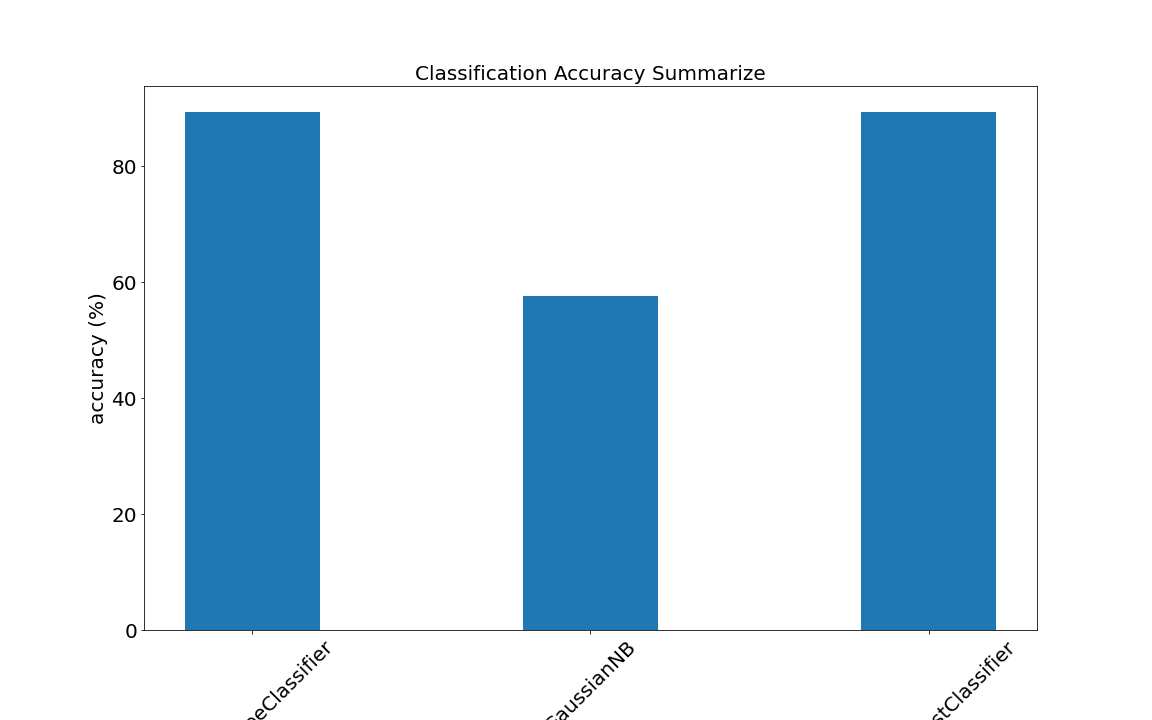
**4: expensive**

**5: very expensive**

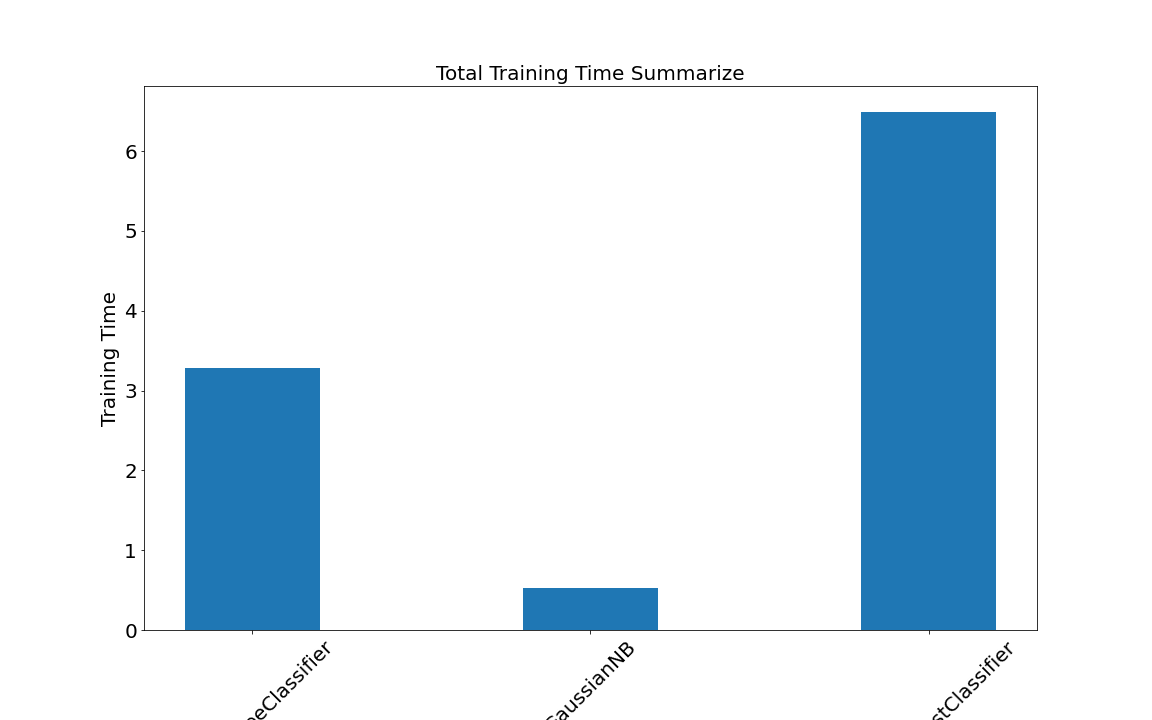
**Feature Selection:**

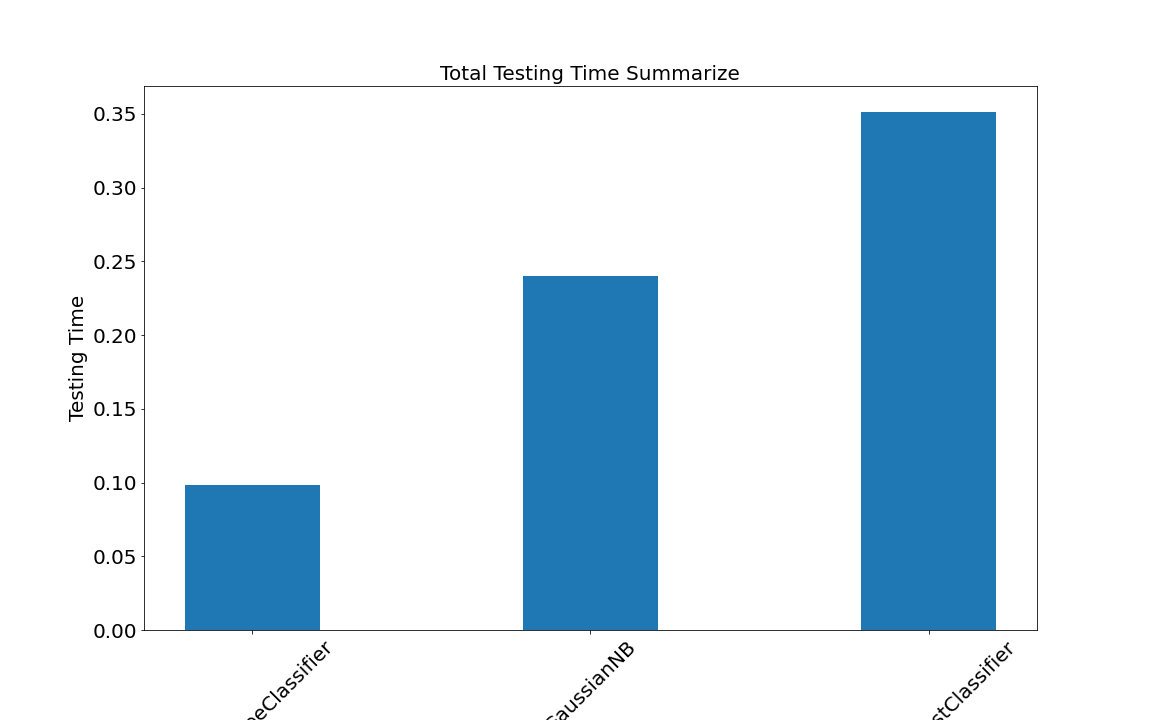
****

**Summarization:**

**1: classification accuracy**

**2: Total training time**

****

**3: Total test time**

**4: Mean Square Error**

****

Hyperparameter tuning:

## RandomForestClassifier's hyper parameters.

{'bootstrap': True, 'ccp\_alpha': 0.0, 'class\_weight': None, 'criterion': 'gini', 'max\_depth': None, 'max\_features': 4, 'max\_leaf\_nodes': None, 'max\_samples': None, 'min\_impurity\_decrease': 0.0, 'min\_impurity\_split': None, 'min\_samples\_leaf': 1, 'min\_samples\_split': 2, 'min\_weight\_fraction\_leaf': 0.0, 'n\_estimators': 3, 'n\_jobs': None, 'oob\_score': False, 'random\_state': None, 'verbose': 0, 'warm\_start': False}

## Try to Explain in detail how hyperparameter tuning affected RandomForestClassifier models’performance.

Android Application

Tools:

1: Heroku Cloud: we upload our model on Heroku cloud

So, we can use model anytime.

2: Flask Framework: to connect model with local server as framework.

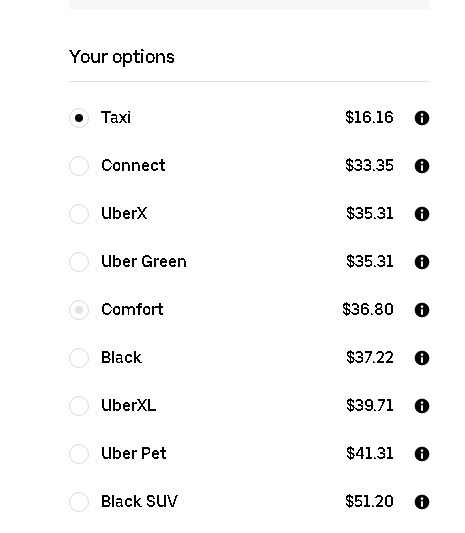
3: Android Studio: we use it to deploy our application.

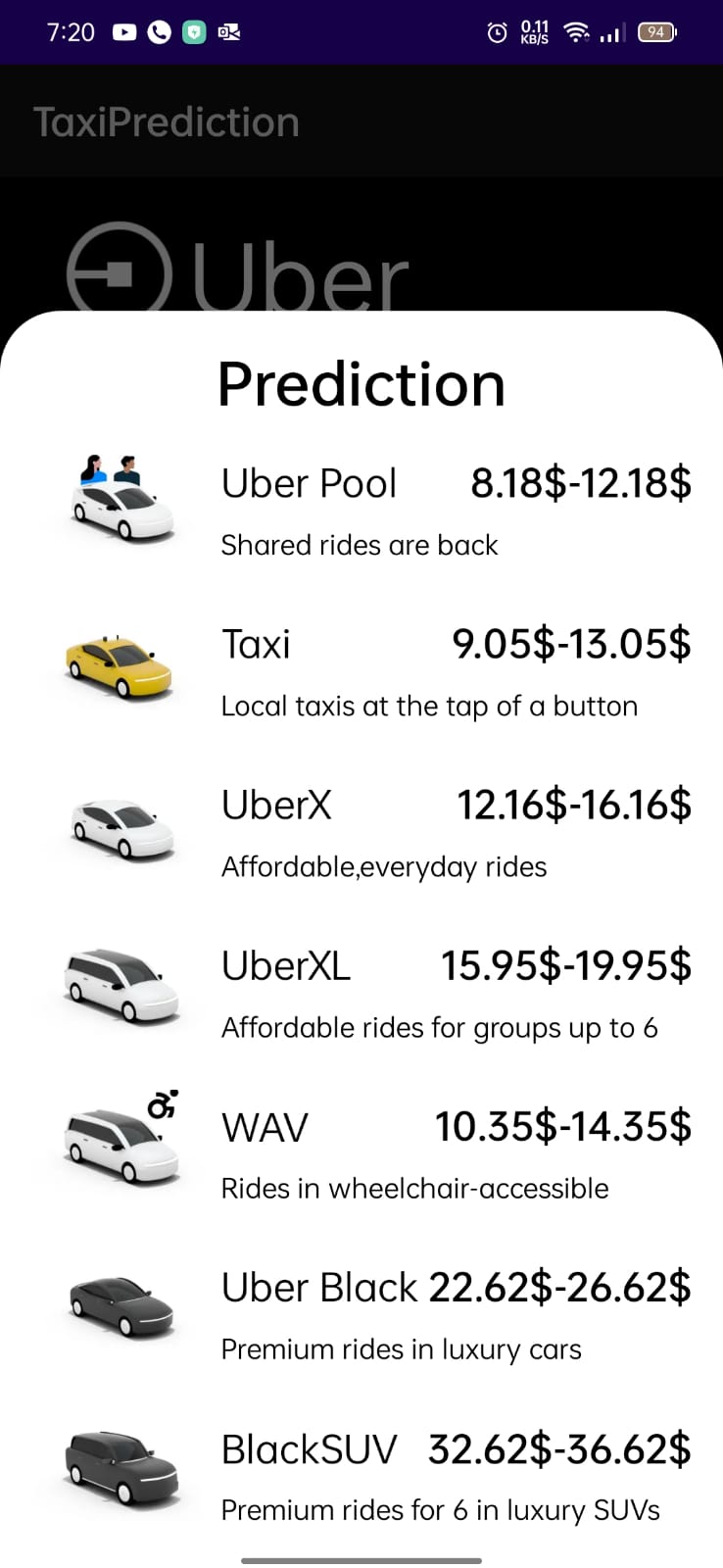
4: Google Places API: we use it to get any street name in all the world.

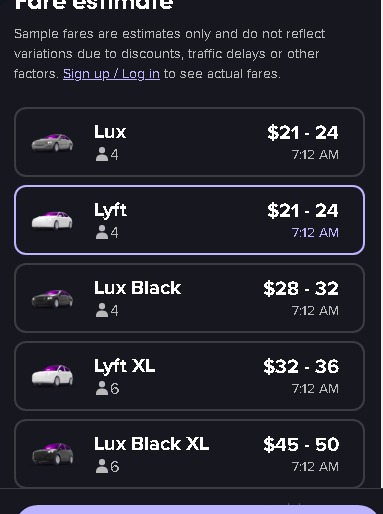
5: Google Map API: we use it to get maps from google

6: Google Distance API: to calculate distance between two points in map

Comparing Between our Application and the real application to predict the price between same distance

 Our uber app real uber app



 Our Lyft app Real app

