



AIRPORT GUIDER ROBOT

CS590BD



Project group: 9

Team members:

Lavanya Kumar Somu

Sivaji Ganesh Kandimalla

Kogila Vani Kannan

Likitha Janga

AUGUST 1, 2015

UMKC
CS590BD

Motivation:

Since my childhood I heard a lot about robots and their capabilities but I never really got a chance to play and work with a robot which can be coded and customized according to my needs and requirements. It is very much exciting while going through the features and capabilities of Robome on internet. The simple idea that we are going to do project using a robot is a great motivational factor to take up and successfully implement and execute the project.

After careful consideration of features and analytic comparison between Robome and Romo **we have chosen Robome for our project**. Robome provides us the ability to customize the remote control triggers, we can record our own voice and use them as a customized voice commands for Robome. We would like to use all these features in successful implementation of an ***Airport Guider Robot***.

Objectives:

- To improve the travellers experience by providing them certain services.
- Airport Guider Robot can also answer questions and for entertainment factors like climatic conditions in and around the airport, flight status etc.
- Airport Guider Robot can guide the passenger to various places in airports like ATMs, Shopping areas, Washrooms, Currency Exchange centres, Baggage claim areas, Terminal routes etc.

Related Work:

Customer service robots are now becoming a reality in the airports. A mobile customer service robot is recently been trailed at Geneva airport and it was well received by the travellers. This robot is helpful in guiding the passengers to ATMs, Shopping areas, Washrooms, Currency Exchange, Baggage claim areas, Terminal routes etc. The robot not just offers directions to the passengers for various facilities but it also physically take them to the destination.

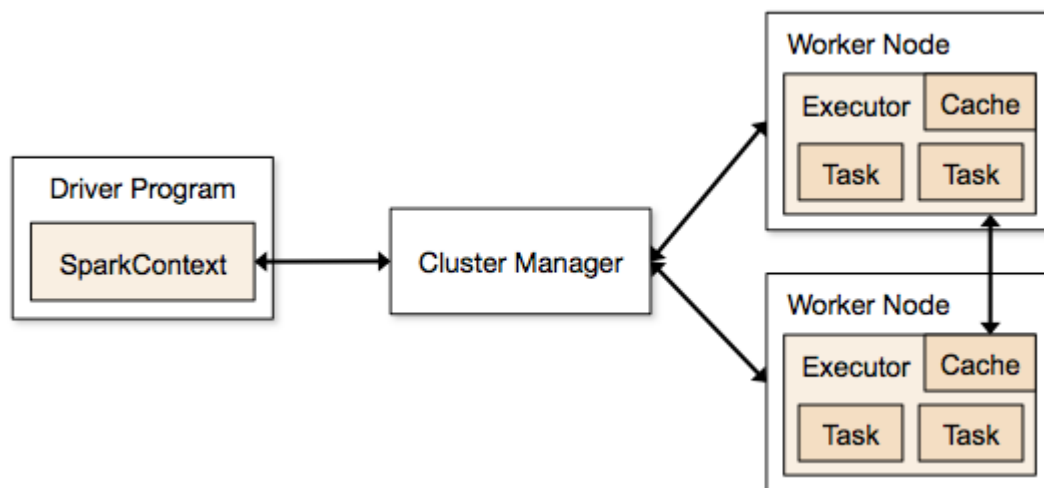
The passengers have to physically touch the screen and have to choose the facility for which they need directions that's it the robot will start direct them in to the destination and the passengers had to just follow it.

Architecture:

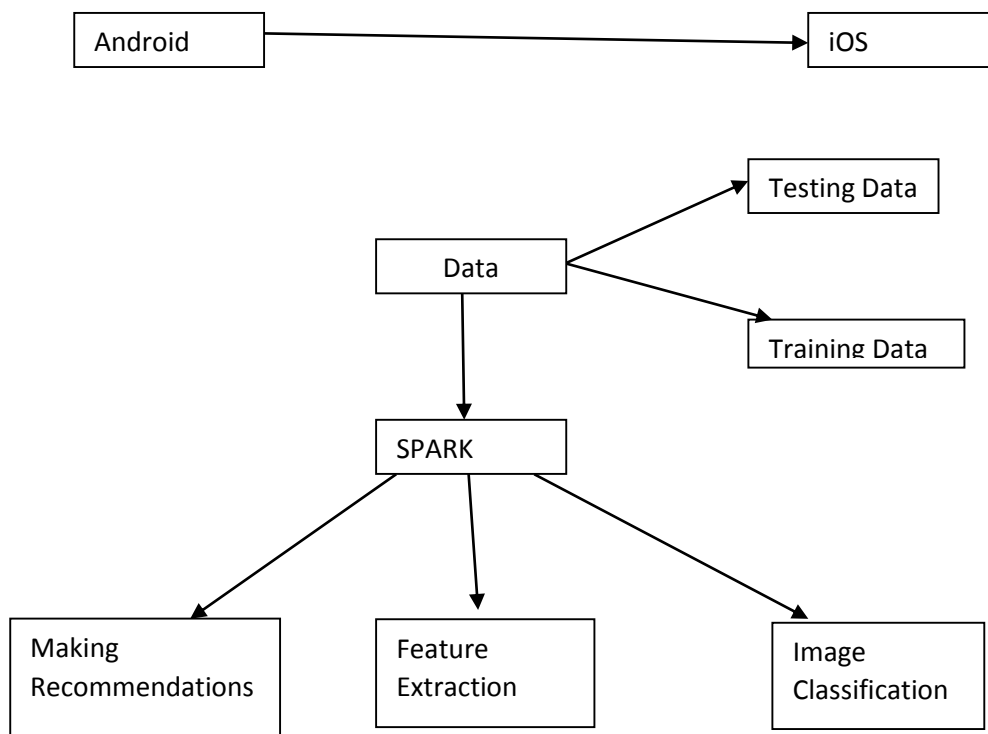
Spark Architecture:

Spark mainly consists of two key components RDDs (Resilient distributed data sets) and DAG (Directed acyclic graph). When data sets which are created in our project we have used RDDs. They are parallelized collections which are based on scala datasets. Spark application

run independently as processes and these are coordinated by the driver program.



System Architecture:



System Features:

- Face Recognition with greeting
- Play Music
- Make Call
- Question and Answers
- Weather information
- Add Reminder feature

- Gesture Recognition
- Voice command the flight number and get the Flight Status
- Recommends passenger the top 5 shopping areas in the airport
- Image Classification
- Play movie

Feature Details:

Nearby locations API:

We have used nearby location finder API using speech recognition. We have used **YELP API** to find the nearby locations.

In our project the travelers at the airport can search the nearby available restaurants by cuisine. They can also search other places like hotels in case of any flight delay or cancellation.

It basically takes the input through voice format and recognizes it and converts it back to text and prompts the user for its correctness. Once after user confirmation, the nearby restaurants at any airport will be displayed in the 'search result area' based on the cuisine inputted by the user.

This application will list down all the restaurants, ATMs etc as per the given voice input.

Movement Control and Expression using socket communication:

The following actions and expressions are implemented: Go Stop, Left, Right, Back, Front and Smile. Robo-Me is configured for performing all the actions mentioned above by both voice communication and button click events.

It was also designed in a way to respond to the touch and voice commands. Socket programming is used to connect both Android and iOS devices. Speech recognition is the translation of spoken words into text.

RoboMe is designed to recognize the speech automatically and act accordingly. Based on the user input, RoboMe can either move left or right, front or back even stop. The commands are passed to RoboMe using Android device by configuring the IP address and socket port in both the devices.

Weather API:

Description:

We have used a free weather API to display the temperature information of a given location in both degrees and Celsius. This is helpful to our increment 1 of the project as we are required

to display the temperature information requested by the passenger to help him suggest the temperature in the area so that he could be able to adjust to the surroundings.

Weather API Provider: wunderground

In this API we have three plans such as stratus, cumulus and anvil. Stratus tells about the local weather features. Cumulus tells about weather forecast and also severe alerts. In Anvil we have access to all the above levels.

The following applications use this API : Living Earth, Banjo, US home town locator etc.,

Base-URL Structure:

http://api.wunderground.com/api/00e83aeb00e951db/conditions/q/MO/Kansas_city.json

Image Capturing:

We have implemented an image capture coding to capture the picture from using either of the cameras of the mobile and save the image to the gallery.

Sing a Song:

We have implemented the feature to play an audio file given as input to the application. This can help in entertaining the people as we all have some delays during our journeys.

Make a call and send SMS:

We have implemented this feature using the Twilio API which is available. The robo can call or send message to a person in this application which helps the traveler in case of any emergency.

Add Reminder:

This feature is implemented in order help the person to remind him about the flight timing when to check-in or it also reminds him to meet the respective person at some time.

Gesture Recognition:

It recognizes the static position and dynamic positions of a user. It takes the accelerometer readings and calculates the orientation and then depending on the movement it moves the robot. It helps the traveler in saying hi and hello and also helps in movement.

Face Recognition:

We have implemented it in our application in which the user has to register his face at the very first time when he meets the robome and then from then onwards it detects the face of a user and greets him (like hi likitha). We have used Kairos SDK to implement this feature.

Recommendation:

We have implemented this with the help of spark in order to recommend the top shopping areas in the airport. It also recommends the traveler in finding the best restaurants in the airport.

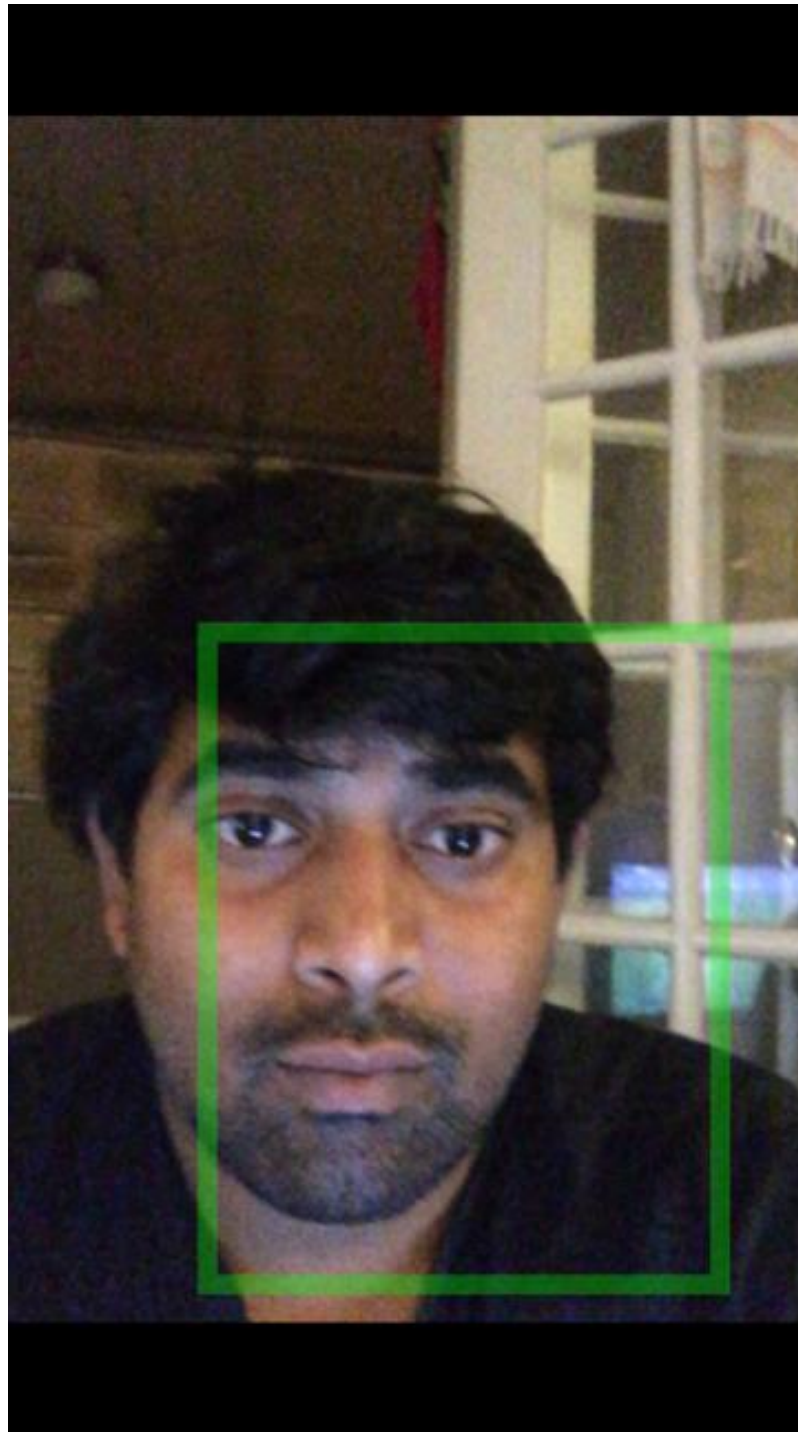
Voice command:

We can give a voice command to the robome and get the flight status in case of any delay. The traveler gives his respective flight number through voice based commands and then gets his status on the robot which displays complete flight information.

Results and Evaluations:

Screen shot for Face Recognition:





Screenshot for Weather:

We must give the state and city then it gives temperature of the particular location.

iOS Simulator - iPhone 6 - iPhone 6 / iOS 8.3 (12F69)

Carrier 6:13 PM

State Code

City Name

Get Temperature

Temperature Result

iOS Simulator - iPhone 6 - iPhone 6 / iOS 8.3 (12F69)

Carrier 6:14 PM

State Code

City Name

Get Temperature

Temperature 87.1 F (30.6 C)

Color Detection:

T-Mobile 11:34 PM
Red Blue Green

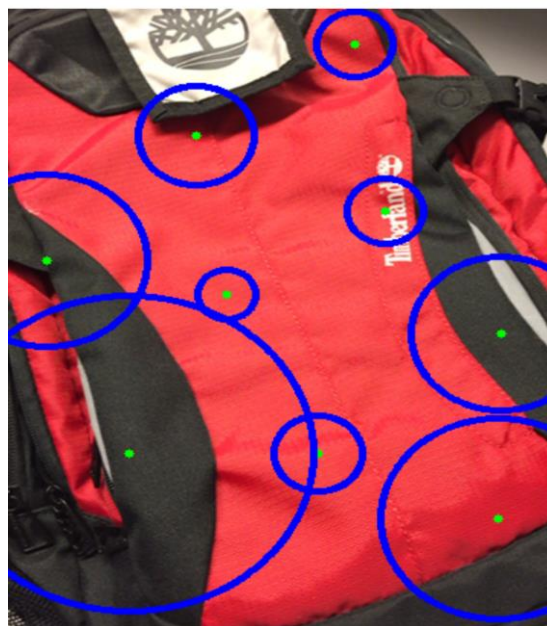


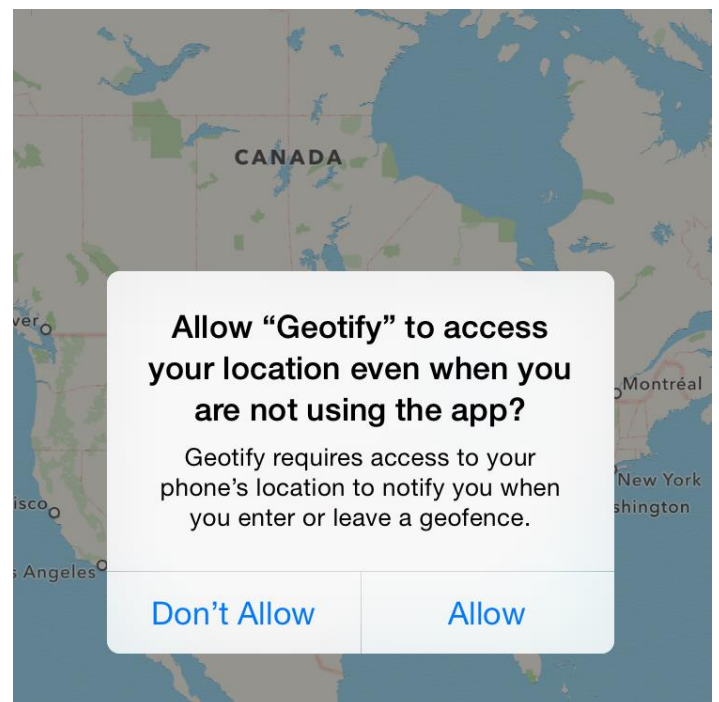
Image Capturing:



Sing a Song :

Playme

Geo- locations:

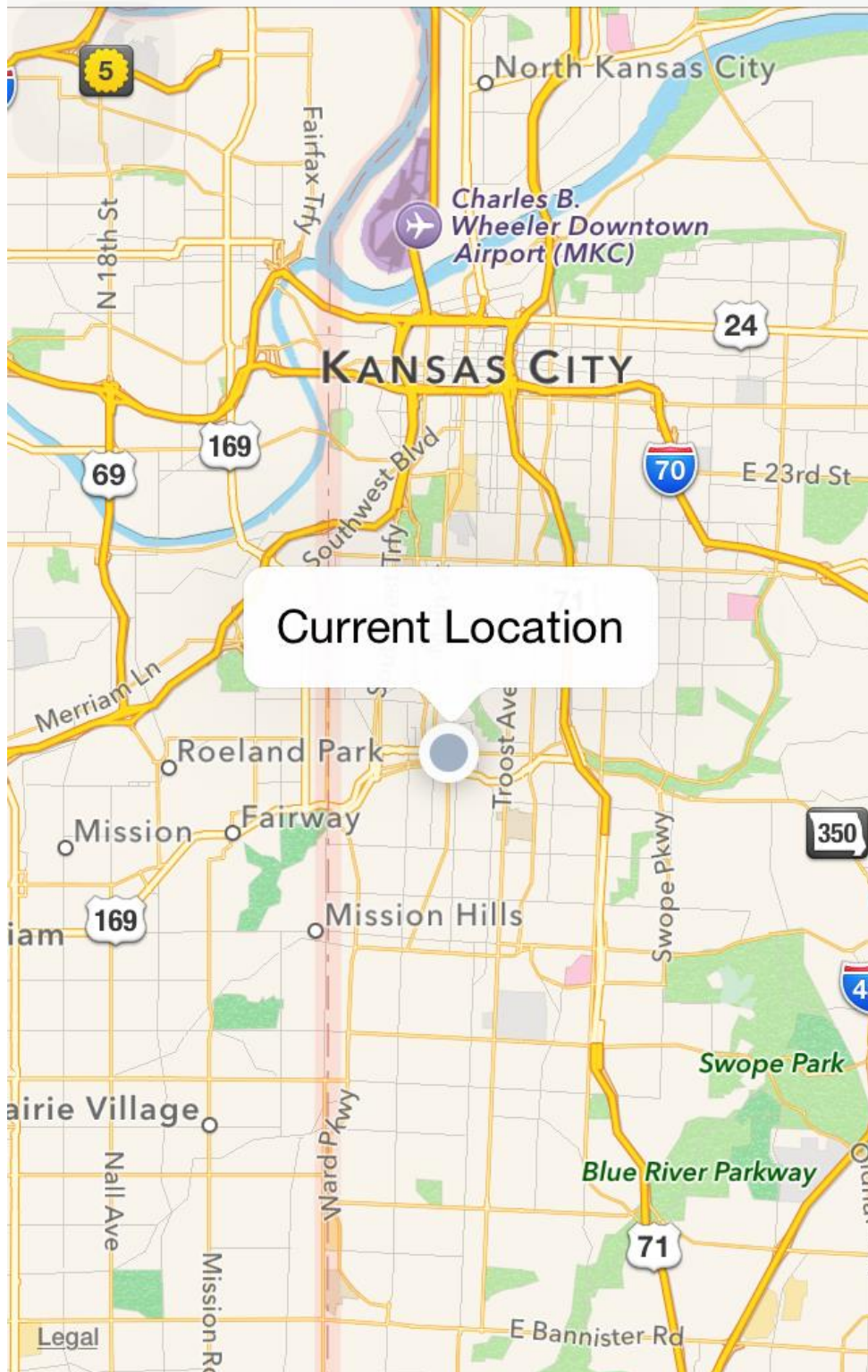


iPod

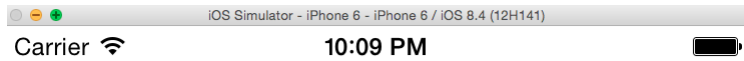
10:16 PM



Geotifications (0)

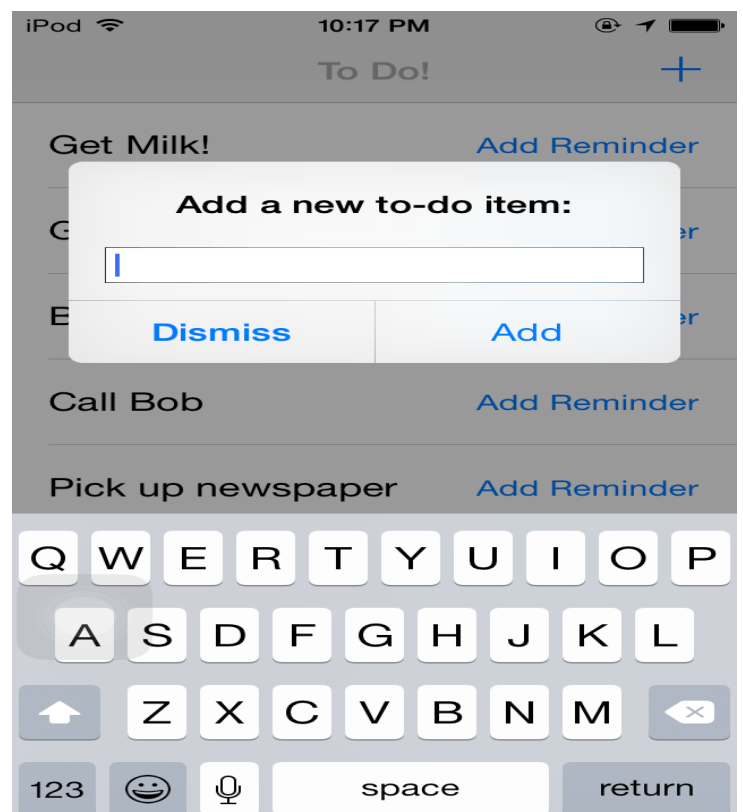
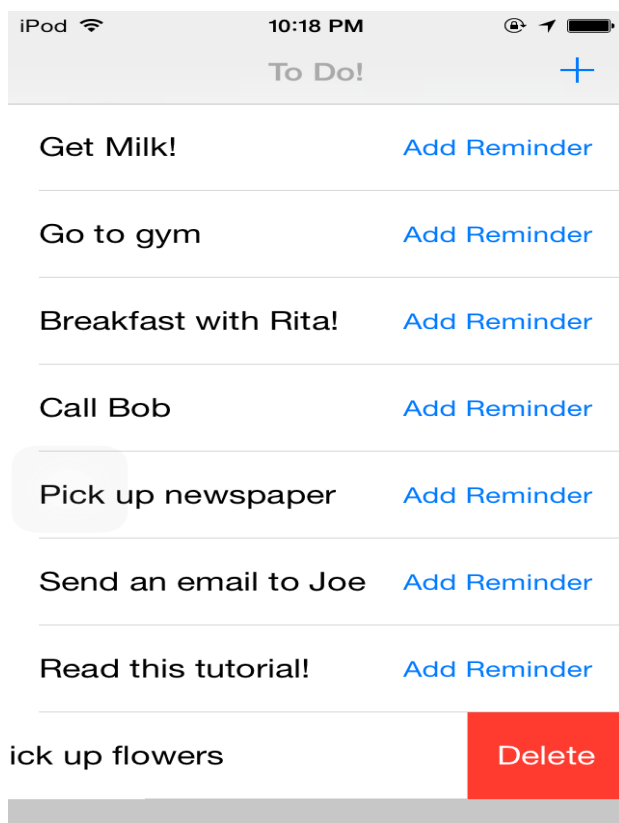


Time Display:

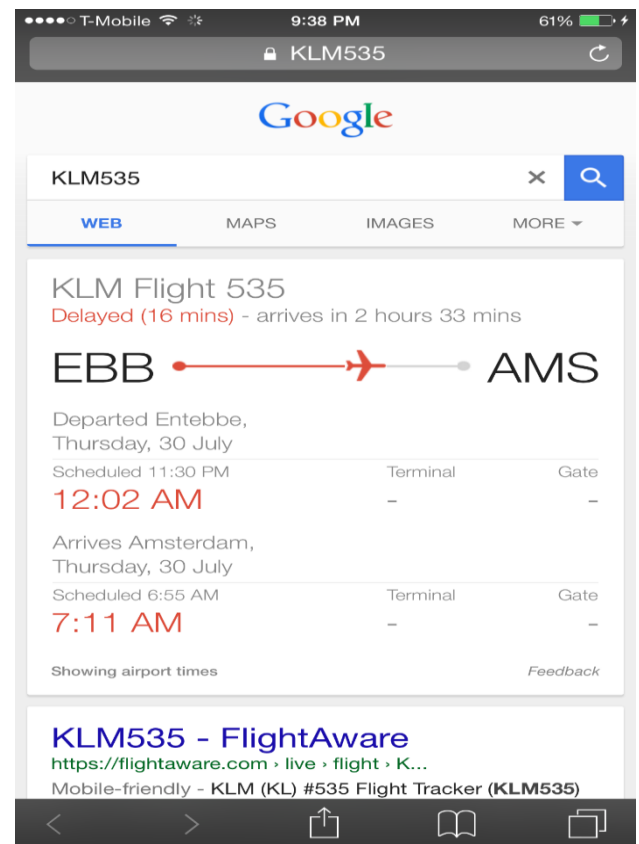
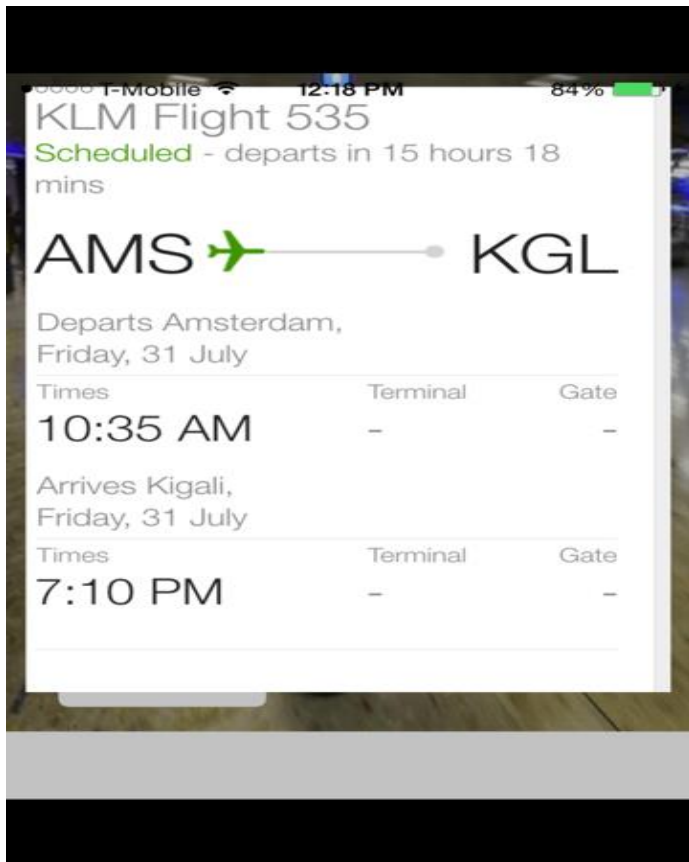


10:09:25

Event Reminder:



Flight Status: once a passenger sends his flight number as voice command. The current status of that flight will be displayed in web view. To go back to the application screen he has to give **thank you** as a voice command to close the webview.



Recommendation:

- ✓ We have a data set of 100 Shopping Stores in airport of various categories like Cloths shopping, Jewellery shopping, Entertainment centres, duty free shopping etc.
- ✓ We have a total of 20 Customers
- ✓ These customers gave their ratings for 71 stores out of 100.

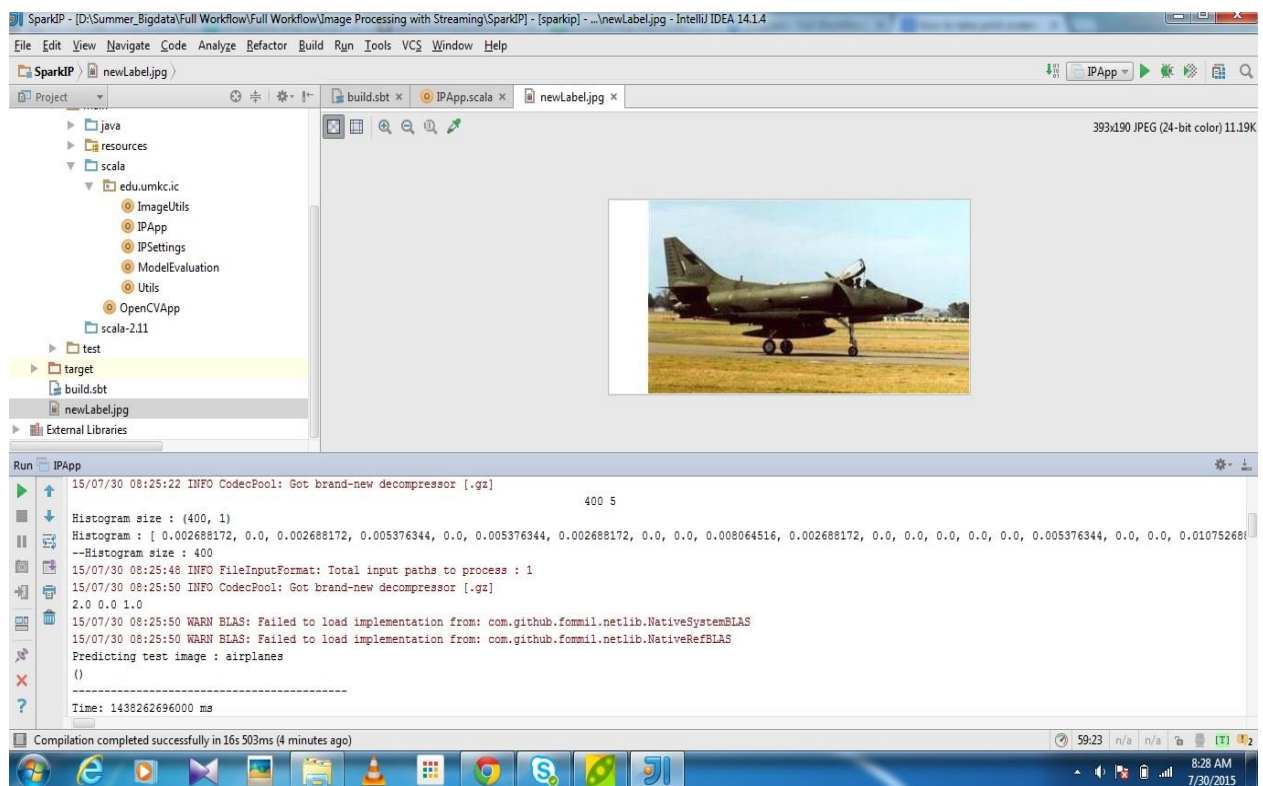
```
15/07/30 04:38:25 INFO deprecation: mapred.job.id is deprecated. Instead, use mapreduce.job.id
15/07/30 04:38:25 INFO FileInputFormat: Total input paths to process : 1
Got 200 ratings from 20 users on 71 Shopping Stores.
Training: 138, validation: 28, test: 45
15/07/30 04:38:29 WARN BLAS: Failed to load implementation from: com.github.fommil.netlib.NativeSystemBLAS
[Stage 25:>] (0 + 4) / 4 15/07/30 04:38:31 INFO UniLoader: successfully loaded C:\Users\LAVANY~1\AppData\Local\Temp\jnlloader91
[Stage 27:>] (0 + 4) / 4 15/07/30 04:38:33 WARN LAPACK: Failed to load implementation from: com.github.fommil.netlib.NativeSystem
15/07/30 04:38:33 INFO UniLoader: already loaded netlib-native_ref-win-x86_64.dll
RMSE (validation) = 0.9467975394413778 for the model trained with rank = 8, lambda = 0.1, and numIter =
RMSE (validation) = 0.8954500307997715 for the model trained with rank = 8, lambda = 0.1, and numIter = 20.
RMSE (validation) = 2.847304489713536 for the model trained with rank = 8, lambda = 10.0, and numIter = 10.
RMSE (validation) = 2.847304489713536 for the model trained with rank = 8, lambda = 10.0, and numIter =
RMSE (validation) = 0.9612371214713711 for the model trained with rank = 12, lambda = 0.1, and numIter = 10.
RMSE (validation) = 0.8638704981693146 for the model trained with rank = 12, lambda = 0.1, and numIter
RMSE (validation) = 2.847304489713536 for the model trained with rank = 12, lambda = 10.0, and numIter
[Stage 1615:=====] (3 + 1) / 4 ]
```

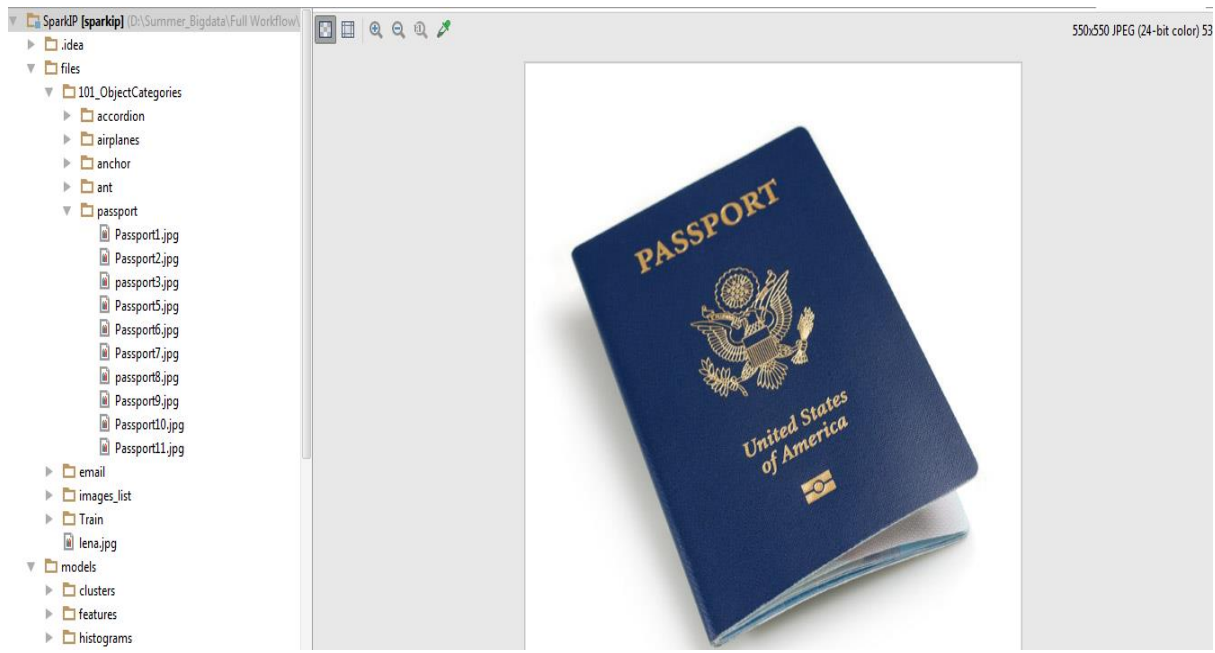

The below screen shot recommends the customer about the top 5 shopping stores located in the airport based on the customers ratings.

```
The best model was trained with rank = 8 and lambda = 0.1, and numIter = 20, and its RMSE on the test set is 1.3313222655306796.
The best model improves the baseline by -37.50%.
Shopping Stores recommended for you:
1: GUCCI --> SHOPPING CATEGORY
2: PLAY STATION --> GAMES CATEGORY
3: ALLURE --> SPA TO RELAX
4: OLAY --> COSMETICS SHOPPING
5: AMERICAN HONEY --> HOT DRINKS SHOPPING
15/07/30 04:54:25 INFO RemoteActorRefProvider$RemotingTerminator: Shutting down remote daemon.
15/07/30 04:54:25 INFO RemoteActorRefProvider$RemotingTerminator: Remote daemon shut down; proceeding with flushing remote transports.
15/07/30 04:54:25 INFO RemoteActorRefProvider$RemotingTerminator: Remoting shut down.
```

Image classification:

We have successfully deployed the given image classification project. We are working on passport image classification. Whenever a customer lost his passport our airport guider robot will come to the assist him and help him in finding the passenger.





Conclusion:

- ✓ Thus we implemented client server socket programming to connect both android and ios device.
- ✓ We also trained the data samples and tested using various data classification techniques.
- ✓ Our main aim was to build a robome project for helping passengers in an airport by assisting/providing them various services.
- ✓ We implemented many imminent features such as flight status display, suggesting passengers top 5 shopping areas in and around the airport.
- ✓ There is a lot of scope for the improvement of this airport guider robot.

Future Work:

We are trying to display up the currency exchange rate in our application to the traveler and also we are trying to help the traveler in **finding the lost passport through image classification**.

References:

- <http://www.bluebotics.com/robot-guide-airport/>
- <http://www.spencer.eu/papers/joosseCTIT15.pdf>
- <https://www.youtube.com/watch?v=L5BdMtbQMA>
- <http://www.wunderground.com/>

