MISWA
TEAM PIONEERS

Developer Documentation (Android Application)

Authors

Mariam Mahboob

Elena Fredenbrink

Design

Factors

- The platform used to develop the application is **KITKAT (4.4).**
  which required the minimum sdk of 19.
- The programming language used in backend coding of the application is Java.
- The data is retrieved from phpmyadmin database , which uses mySQL as a language.The connection build between the application and database is supported with several web api's.

Use of Web API's:

Since, it is not possible to connect mySql database with Android directly.The web api's are written in this respect.
There are a total of 6 api's:

	NAME	**FUNCTIONALITY**
1.	getIds.php	get ids of the employer and the company on the basis of username and password entered, in a JSON array named "data"
2.	login.php	get status (true/false) from the url , to check is username and password entered by the user are correct.
3.	ChangePassword.php	get status (true/false) from the url , to check if new password entered is updated in the database.
4.	missions.php	get list of the mission **(**name of mission and address to that mission**)** assigned to a particular employer in a JSON array named "missions"
5.	ContactCustomer.php	for getting contact of the customer in a JSON array "contacts".
6.	ContactCompany.php	for getting contact of the company in a JSON array "contacts"

These api's are uploaded in a server at **FileZilla,** and you could get response to any api by entering a url address.

http://apps.dev01.luqon.com/appui/api/+"**APINAME"+?+"PARAMETERS**"

Software build process

The "MISWA" android app is programmed in eight java classes, and six Xml files.

1. MainActivity.java
2. ChangePassword.java
3. UserActivity.java
4. MissionList.java
5. ListViewAdapterM.java
6. MapsActivity.java
7. JSONfunctions.java
8. DirectionsJSONParser.java

These classes are supported by two external jar files

- commons-io-2.4.jar
- maps.jar

1. Main Activity

The **MainActivity** handles the display of login screen through the xml file **activity_main.xml** .This class uses two Web api's , **login.php** and **getIds.php** to track the value of employee id and company id ,and to check if the user has login with correct username and password respectively.
When the company add any employee they assign them the the similar username and password.
So when the username and password is entered correctly and is equal to each other, the system will figure out it is the first time login and will prompt to change the password.

2. Change password

ChangePassword uses the **activity_change_password.xml** to display screen for entering the new password.
It will have two password fields for typing the new password and for confirming that password.The password in both fields need to be similar, after which it would be updated in the database.
The new entered password needs to be different from the previous password.
Web api **ChangePassword.php** is used for chech is password is updated without any technical issues.
Employee id and company id will be passed between all activities.

3. User Activity

When the new password is updated or when the user enter correct username and password other than the first time. The system enters the **UserActivity.java** which is used to display the welcome sceen through **activity_user** file.

4. Mission List , 5. List View Adapter

When the “Mission button” in **activity_user** is clicked the next activity **MissionList.java** will show the missions in a list through **activity_mission_list.** This activity is linked to another java class **ListViewAdapterM.java** which is linked to **listview_itemm** for displaying the individual item on the list by getting all the item from JSON array through **missions.php** api. If the JSON array is empty the list view will be shown empty with the default text field displaying about empty missions at the moment. Any one item from the list would be selected and the value of that item would be passed to the next activity.

The system at this point needs to be connected to the AGA simulator to function.
There are two values used from the AGA simulator database are,

- Wheel speed
- Parking Brake

6. Maps Activity

The system would do nothing when the driver is not moving but when the speed of car is raised from 0 to any other value the screen would change to **Maps.Activity** The **activity_maps** UI would show the minimal distance from the driver current location to the final destination.

The final destination would be found from the address taken from the text field in **activity_mission_list.xml.**
The Api key is taken from google maps when is unique for every system that uses google maps.So if the project is ran on any system and is not showing the maps, it would more likely to be because of api key written in **android manifest file,** that would be considered duplicate.

So in order to get new api key,go to link below and create a new one,
https://console.developers.google.com/flows/enableapi?apiid=maps_android_backend&keyType=CLIENT_SIDE_ANDROID&r=FD:DB:40:12:7F:72:E9:F4:C4:EE:30:1E:82:80:FC:61:B3:63:CB:BF%3Bcom.example.mariam.miswa

You can also add your credentials to an existing key, using this line:

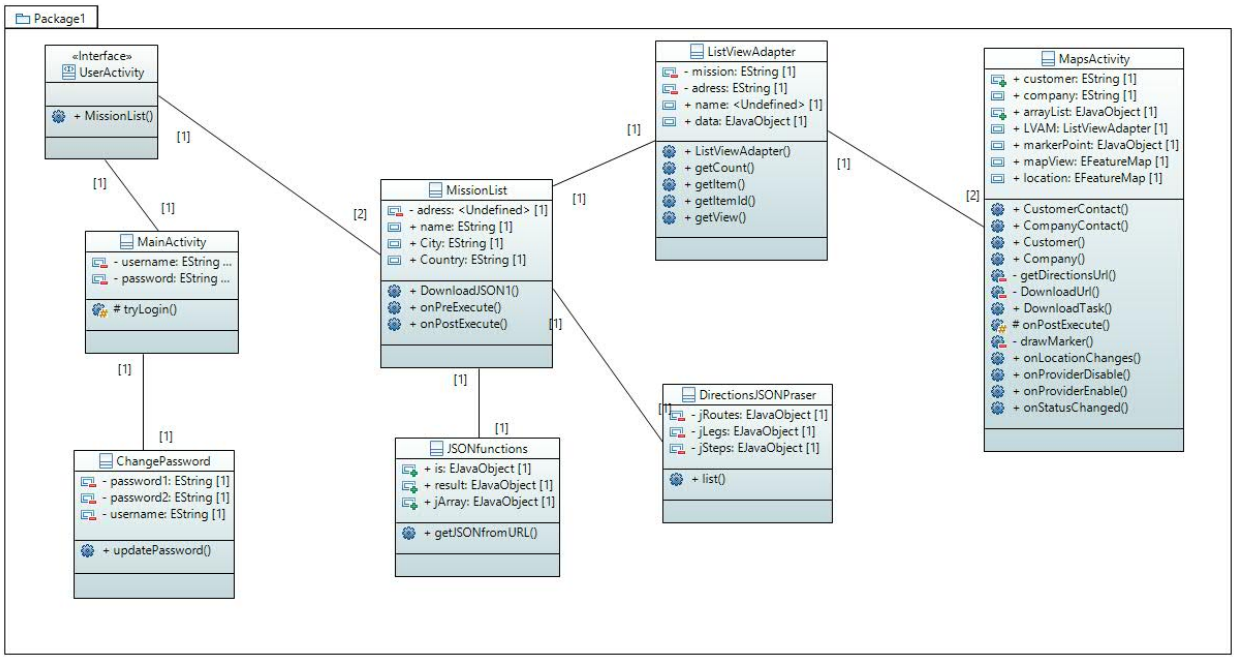
FD:DB:40:12:7F:72:E9:F4:C4:EE:30:1E:82:80:FC:61:B3:63:CB:BF;com.example.mariam.miswa

The maps would detect driver current location from the last known Location. so in order to get the direction the longitude and latitude for the current location will be figured out on the basis of location detected from the google maps in an other app from the mobile. If no other app has used location or the GPS in the mobile is turned off , the system would get an error finding current location.

MapsActivity also stores the functionality of calling the customer and company.
This uses two web api **ContactCustomer.php** and **ContactComapny.php** which get phone numbers from the json array. Company phone number will be figured on the basis of company id associated with the driver's employee id. On the other hand Customer phone number will be detected on the basis of name of selected mission.

The button click will put the value of phone numbers in the keypad that would be called from there.

Class Diagram:



Sequence diagrams: