**Android Application**

**Classes:**

**1 .WelcomeActivity class**

1.1 This class is reprsents the first screen showed to the user when he enters the app.The activity created by this class gives the user option to create a new tour by pressing a button.

1.2 This class extends Activity.

1.3 Methods in this class :

1.3.1 startTour();

This method is called by the android system when the user presses the button responsible for creating new tours. It starts new activity of StartTourActivity class.

1.4 This class’ functionality satisfies the FR1 requirement from SE.QA.RS document.

**2. StartTourActivity class**

2.1 This class provides an interface to the user to type a title and short description of the walk. The activity has two text fields for the user to choose a title and a short description for the tour . It also has a button to start the tour itself.

2.2 This class extends Activity. This activity is child of WelcomeActivity.

2.3 Methods in this class :

2.3.1 startTour();

This method starts another activity of MenuActivity class.

2.4 The class complies with the FR2 and FR7 Requirements from SE.QA.RS doc.

**3. MenuActivity class**

3.1 This class is responsible for giving the user an interface through which he can create new points in the tour, edit already stored points or to finish the tour. The menu interface is represented by three buttons, namely “Add new location”, “Edit locations” and “Finish”.The user is able to cancel the walk at any given time while in this activity,for which there is a button named “Quit”. A warning message will apear if the user decides to stop the walk. This class has underlying ,hidden from the user, function of writing the begginig latitue ,longitute and innitial timestamp to the local memory.

3.2 This class extends Activity, and makes use of a GPS service. This activity is child of StartTourActivity.

3.3 Methods.

*3.3.1 addLocation();* This method starts a new activity from AddLocationActivity class.

3.3.2 editLocations() This method starts a new activity from EditLocationsActivity class.

3.3.3 finishWalk() This method starts a new activity from the FinishWalkActivity class.

3.4 This class satisfies FR5 ,FR6 and FR7 requirements from the SE.QA.RS document.

**4. AddLocationActivity class**

4.1 This activity is where the user describes the location he wants to add to the tour and is given a chance to add a photo related to this location. The description part is represented by two text fields, one for the title of the location , and the other for a short description. The adding photo is represented by two buttons.The first creates an activity that leads the user to the gallery of pictures on the local memory storage. The second one creates activity that makes use of the phone’s camera.This class has underlying ,hidden from the user, function of writing the current latitue ,longitute and timestamp to the local memory. The user will be able to cancel the tour at any given time,through a button named “Quit” .The user will be warned that all the data saved so far will be lost.

4.2 This class extends Activity, and makes use of a GPS service. This activity is a child of MenuActivity.

4.3 Methods in this class

4.3.1 photoCamera(); Starts activity for result , a picture captured from the camera.The photo is linked to the location.

4.3.2 photoMemory(); Starts activity for result, a picture from the local memory. The photo is linked to the location.

4.3.3 addThisLocation(); Adds the location to the tour.If successfull goes back to MenuActivity.

4.4 This class satisfies the FR3 ,FR4,FR5 and FR7 requirements from the SE.QA.RS doc.

**5. EditLocationsActivity class**

5.1 This activity provides interface to the user through which he can edit or delete already stored locations from the tour. The interface will provide a listing of currently added locations and a link to each of them. Selecting the link will start new activity from which an activity is edited or deleted.

5.2 This class extends Activity. This activity is a child of MenuActivity.

5.3 Methods in this class .

5.3.1 editThisLocation(); Invoked when a location is selected. It starts new LocationEditorActivitythat edits the location.

5.4 The functionality of this class complies with the FR5 and FR7 requirements from the SE.QA.RS document.

**6. LocationEditorActivity class**

6.1 This activity provides basic interface through which the user can manipulate already stored location. He can edit it’s title , descrpiton, add additional photos or remove photos already linked to the location.

6.2 This class extends Activity.Activity from this class is child of EditLocationsActivity.

6.3 Methods in this class –

6.3.1 photoCamera(); Starts activity for result , a picture captured from the camera.The photo is linked to the location.

6.3.2 photoMemory(); Starts activity for result, a picture from the local memory. The photo is linked to the location.

6.3.3 changeTitle(); Changes the title of the location.

6.3.4 changeDescription(); Changes the description of the location.

6.3.5 unlinkPhoto(); Removes the link between the selected photo(s) and the current location.

6.4 The functionality of this class complies with the FR5 and FR7 requirements from the SE.QA.RS document.

**7. FinishWalkActivity class**

7.1 This activity finalizes the walk. The user is prompted with a textbox to write a long description for the Tour. A button is given , which when pressed uploads the entire tour to a server via HTTP POST request.

7.2 This class extends Activity. Activity from this class is child of MenuActivity.

7.3 Methods in this class –

7.3.1 uploadWalk(); invoked when the button “Upload” is pressed.

7.4 The functionality of this class stafisfies the requirements FR6 and FR7 as described in the SE.QA.RS document.