

4.1 Programs in system

The Walking Tours application will consist of two main components. The two applications are the Android application and the web processing application. The web application is dependent on the Android application, and in some sense the Android application depends on parts of the web application.

The Android application:

The Android application will be the main application in this project. The Android application will consist of several Activities and Classes which will be used in conjunction with user interactions and Google API's to record a users walk.

This application evolves around the process of initially creating a walk with a given name and descriptions. This information is then stored within the application (in an easily accessible location) and the users current GPS location is retrieved. This GPS location can then be used to provide the user with a localised map of the surrounding area, which can then be used to display the users movements (requires the ability to constantly record GPS location) and also allow for the addition of flagging up "Points of Interest" along the current route.

These "Points of Interest" will also prompt the user for a name, description and the option of taking a photo/selecting an image to be associated with said "Point of Interest". These "Points of Interest" will also be displayed on the current Map and their location on this map will be determined by the GPS location where the user creates the point of interest, at this point the coordinates of the users location will be assigned to the "Point of Interest" which allows for placement upon the Map.

Once the user is satisfied with the current walks situation (e.g. all "Points of Interest" added and walk finished) then they can upload the results to the sever. The user will be prompted to make sure they do in fact want to upload the walk and miss-clicks often happen especially on touch screen devices.

The uploading to the sever will require a device (running Android) with network connectivity of some variation in order to access the server, the information will be sent to the server as an HTTP POST request. This is where the Application is reliant on the web application, if the sever is unavailable then the application cannot meet the functional requirements (FR 6).

This part of the application will run on any Android enabled device running Android version 2.3+ (Gingerbread onwards) and a minimum API level of 10.

The web application:

The web application refers to the website and the processing server required in order to produce a fully functional product.

On it's own the website will be able to retrieve information form a database, this information will be directly related to the number/details of currently saved walks sent from the android application. The information retrieved from the database will then be displayed initially as a list of all the stored walks (listed by title) then a visitor to the website can click a walk and be shown the walks route, the information, the walks duration as well as all points of interest associated with said route.

The web application is heavily dependent on the Android Application. If no information (e.g. walks) was being sent to the server from the application then the database will contain little or no entries resulting is none or few walks being displayed on the website. The main dependency is the web application with the Android application as saved walks or needed to the web application to produce effective results.

This part of the application will on the web server, and will on any modern up to date browser.

4.2 Significant Classes:

FrmHome (FR 1):

The FrmHome Class represents the initial screen presented to the user. This screen will be key in the running of the application as it houses a Button which calls for the creation of the walk, until this button is clicked the user will ot progress from this initial state, where only the FrmHome screen is displayed.

FrmCreate (FR 2):

The FrmCreate Class represents the screen where the user will be prompted for the information related to the walk's information. This information will include the walks name, short description and its longer more detailed description. This Class also calls the creation of the FrmView which will be the Map/route display to the user.

FrmView (FR 3):

The FrmView Class is used to model the current route using graphical representation. The map will display the users current position and the current time elapsed along the route, as well as displaying all (if any) current Points of Interests along the current walk's path. The user can also create new Points of interest and add them to the walk when viewing this screen.

FrmReview (FR3, FR 4, FR 5,FR 6, FR 9):

The FrmReview Class is used to display a summary of the walk to the user. This review is displayed after th user chooses to upload the walk to the server. The result of this summary can then be used to either accept the upload or decline and carry on recording.

FrmNewPOI(FR 3, FR 4:

The FrmNewPOI allows the user to specify the information required in order to create a new Point of Interest which can then be assigned to the current walk and stored locally ready for the server upload. This Class is vital if the application is to achieve maximum functionality.

POI (FR 3, FR4):

The POI Class is used to specify/hold the information related to each Point of Interest. As the user can create multiple Points of Interests during one walk recording it is essential to modulate the system, having a separate Class to store/model the Points of Interest allows this functionality.

FrmCancel (FR 5):

This Class provides the user with the ability to cancel the upload to the server. This prevents the user accidentally uploading a walk that is unfinished or just generally do not want to upload said walk. The user will be prompted making sure they understand that the walk will not be uploaded.

FrmUpload (FR 6):

This Class provides the user with the ability to upload the walk to server. This allows the user to view the walk they have just recorded/uploaded via the web application. This is essential part of the applications functionality.

4.3 Mapping Function Requirements:

Functional Requirement:	Classes providing requirement:
FR 1	FrmHome
FR 2	FrmCreate
FR 3	FrmView ,FrmNewPOI, POI
FR 4	FrmView, FrmNewPOI, POI
FR 5	FrmReview, FrmCancel
FR 6	FrmReview, FrmUpload
FR 7	FrmCreate, FrmView, FrmNewPOI, FrmConfirm, FrmCancel, FrmUpload, FrmReview
FR 8	Handled by the web application
FR 9	FrmView, FrmReview, FrmUpload, Handled by the web application