

WTC app maintenance guide

Program Description

The Walking Tour Creator app, WTC for short, is used to create walking tours of towns, cities or smaller areas of interest. A user will enter the details of the area they are making a tour for. These details include a single word name for the area they are touring, a short description for the area they are touring and a long description for the area they are touring. Whilst making the tour, the user will be able to add key points of interest to the tour. As they do this, they will be prompted for the name of this point of interest and a description for the point of interest. The user is able to add photos to the points of interest and also remove those photos. The user is also able to remove key points of interest they have added. The user is able to cancel a walking tour they are in the process of making, this will erase all the locations they have added to the tour. Finally the user is able to save their walking tour to an online database and view the tour they made.

Program Structure

The app is separated into three activities:

- MainActivity - this is the launcher activity for the app, when the app first starts this is the screen the user is presented with.
- WalkActivity - this is the main part of the app, this where the user builds their tour. From here they can:
 - Add / Remove points of interest.
 - Add / Remove photos to / from a point of interest.
 - Cancel their walk.
 - Save their walk for later viewing.
 - They are also able to adjust the sample rate for non-key locations to save on how much data they transmit to the server.
- AboutActivity - this just a credits screen, it just says who is on the team, the app name and version number.

There are also some supporting UI classes for dialogs.

- NewWalkFragment - sets the details of the walk and starts WalkActivity with that information.
- EndWalkFragment - ends the walk if the user selects 'OK'.
- FinishWalkFragment - does the same as EndWalkFragment but saves the walk to the server before quitting.
- NoNetworkFragment - alerts the user that there is no network available.
- LocationDetailsFragment - gets the details for a key location from the user.

Main Data Areas

The main data area of the app is under the package **com.wtc.grp5.model**. This package contains the classes which contain the data for a tour.

WTCTour - This class represents the walking tour the user will create.

Field type	Field name	Description
String	tourName	The name of this tour.
String	shortDesc	The short description of this.
String	longDesc	The long description of this tour.
LinkedList<WTCLocation>	locations	The list of locations in the tour (includes key locations by inheritance).

WTCLocation - This class represents non-key locations in the tour (used for tracing lines between points on the website map)

Field type	Field name	Description
double	longitude	The longitude of this location.
double	latitude	The latitude of the location.
Calendar	oldTime	The original time stamp in a date/time format.
long	timeStamp	The timestamp for this location as minutes since the start of the walk.

WTCKeyLocation - This class represents key points of interest in the tour.

Field type	Field name	Description
List<String>	photos	The list of file paths for photos the user adds to this key location.
String	locName	The name of this key location.
String	locDesc	The description of this key location.

Algorithms

Saving to the server - main thread

Check network connectivity

If connected

Set up async task to save to server with the server URL and Tour data.

Execute async task.

Else tell the user they have no network connectivity.

Saving to the server - async task

Adjust the timestamps of the locations in the tour minutes since the tour start.

Set up the HTTP Client.

Set up the POST message with the URL from WalkActivity.

Set up the Multi-part MIME.

Add Name/Value pair for the JSON data of the tour.

For each key location **in** the tour

For each photo file path **in** key location photo list.

Add photo file path to Name/Value pair list.

For each Name/Value pair

Add Name/Value pair to the Multi-part MIME.

Send the POST data to the server.

Adding key locations to the tour

User presses the 'Add Key Location' button.

User is prompted for the name and description of the key location they're making.

User enters the details in presses 'OK'.

A map marker is added to map at the user's current location.

The marker title is given the name of the location the user entered.

The marker description is given the description of the location user entered.

The user's current longitude and latitude are put in a key location object and added to the tour list.

Removing key locations from the tour

User selects the marker for the key location.

User presses the 'Remove Key Location' button.

For each location **in** the tour

If location's longitude and latitude match marker's longitude and latitude.

 Remove location from tour.

Adding a photo to a location

User selects the marker for the location they want to add a photo to.

User presses 'Add Photo' button.

Image file is created.

Camera app is started.

User takes a photo.

Image saved to the image file.

For each Location **in** Tour

If currently selected marker's longitude and latitude match Location's longitude and latitude

 add photo file path to Location.

Files

The app uses a serialised data file, which is made via the TourSave class, to store the tour when the user navigates away from WalkActivity or even the entire app. The app also produces JPEG files for the pictures the user takes. The facilities for saving the pictures are provided by the Android OS, however, the naming for the picture files is handled by the app and the names of the pictures is based on the date and time it was taken to ensure name uniqueness.

Physical Limitations

The physical limitations of this program are the need for network communication (mobile data Wi-Fi) and GPS signal. The GPS signal, obviously is for recording the user's location into the tour and the network is for saving the tour to the server. Without either of these prerequisites the app can't function properly.

Suggestions for Improvements

The first improvement to make would be to add the ability to save a walk temporarily when the user selects 'Finish Walk' and there is no network. Another improvement that is needed is the ability for the user to be able to see the photos they have taken for the locations in the walk. The user should also be able to add a photo to locations from their device's photo gallery.

Rebuilding and Testing

To build/rebuild and test this app you will need:

- The Eclipse IDE with the ADT plugin.
- A copy of the Google Play Services Library project to put in the same workspace as the app project.
- The Google Play Services SDK.
- An Android device with at least Android 4 installed on it.

For actual system testing you need to install the app on your Android device. To do this just plug it into the computer you developing on and press “Build/Run” in Eclipse. You may be prompted for a device to build the app to. Just select your phone and press “OK”. The app should start up on its own.