# INSTALLATION AND SETUP

Below are the tools used during our development process of History Explorer. Using other tools or simply the command line is your choice. The latest version of each tool was used during our installation and is labeled so the same version can be used; attempting to use another version may cause side-effects.

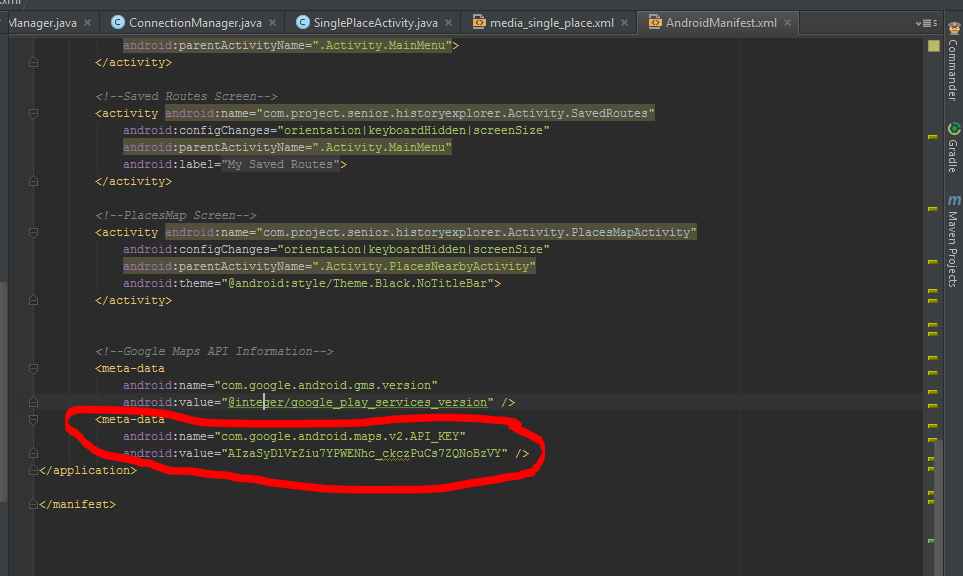
1. Download a telnet/ssh client to remote into the server: (optional) <http://www.putty.org/>
2. Download a FTP client: (optional) <https://filezilla-project.org/>
3. Install Github GUI: <https://windows.github.com/>
4. Install Git Bash: (optional) http://git-scm.com/downloads

## 1.1 Android Setup

* Install NodeJS v0.10.33: <http://nodejs.org/>
* Install Android Studio v0.8.14: <https://developer.android.com/sdk/installing/studio.html>
* Follow the instructions on Android Studio to setup the environment.
* Open a command prompt/git bash/terminal and clone the github directory into your preferred directory. All production was done under the Dev branch. ‘git clone –b Dev <https://github.com/FIU-SCIS-Senior-Project-2014-Fall/History-App.git>’
* Start Android Studio and open the following directory Code/Source Files/Android app/History Explorer
* Select Tools/Android/SDK Manager and install all contents of the following SDK Packages:
  + Tools
  + API 17-20
  + Extra
* On your Android Device, make sure you have Developer Options in the Settings with USB Debug enabled. For more info, <http://developer.android.com/tools/device.html>
* In git bash/command prompt/ terminal, you are able to see which devices are connecting but typing ‘adb devices’ this starts the adb server and will show UDID and Availability of the devices (Offline, Connected, etc). For more info, <http://developer.android.com/tools/help/adb.html>
* Retrieve a Google Maps API Key. FIU email address was used as the account for the API Key

<https://developers.google.com/maps/documentation/android/start#get_an_android_certificate_and_the_google_maps_api_key>

* Site to enable API Key. Make sure you are logged on first: <https://console.developers.google.com/project/395015670207/apiui/api?authuser=0>
* Create an OAuth 2.0 API Key. <https://console.developers.google.com/project/395015670207/apiui/credential?authuser=0>
* Once you receive the API Key, replace the key within the Android Solution. This can be found in “HistoryExplore/app/src/AndroidManifest.xml” in the highlighted area shown below



* If the API Key is not changed, there will be an issue trying to connect to the Google Maps on the Android Application.

## 1.2 Web Application Setup

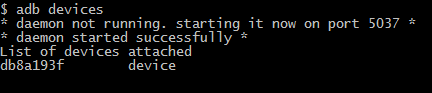
# GETTING STARTED

In this chapter, we were show how to run the application and give two specific use case scenarios as examples. Any installation or environment configuration will be discussed as well. This chapter is divided into each individual system since they both require a different setup.

## 2.1 Android Setup

To run the Android application there are a few pre-conditions that need to be satisfied before installing the apk.

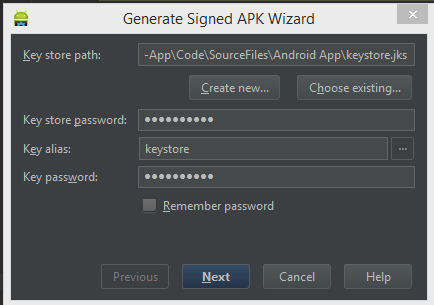
1. Make sure your Android device has Android OS 4.3 or higher
2. Make sure your Android Device is connected via USB to the main computer.
3. Open Android Studio with the History Explorer project open.
4. Open a terminal/command prompt/git bash, and run ‘adb devices’, you should see the following, but with a different UDID. This means your device is connected properly and ready for commands.



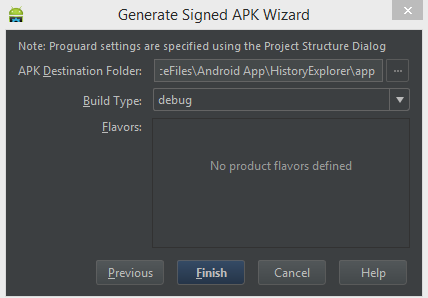
### *2.1.1 How to run the application*

Due to Google Map being used on the application, the Emulator cannot be used to test the Google Map portion of the application. Therefore, an apk must be generated for use. Below are instructions on how to generate the apk and install it on your Android Device.

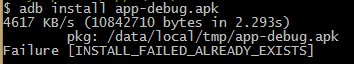
1. In Android Studio, select Build and ‘Generate signed APK’
2. Go to the keystory directory, ‘../Code/Source Files/Android App/keystory.jks’
3. Both key store password and key password are historyapp
4. Key alias is keystore



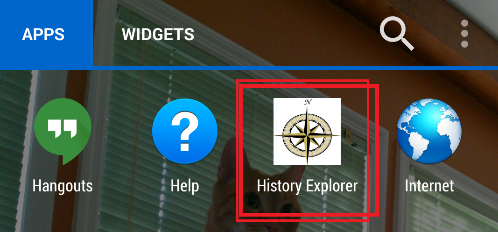
1. Select next. Build type should be debug, APK Destination Folder should be ‘../Android App/HistoryExplorer/app’. Click Finish.



1. In your command prompt/terminal/ git bash, cd to the APK Destination folder
2. Type ‘adb install app-debug.apk’ and enter. The application will install onto your Android Device and return the statement Success.
3. If you receive the following error, make sure to delete the application from your device before reinstalling. There is a command for removing before installing but I had less issues if I just removed it manually.



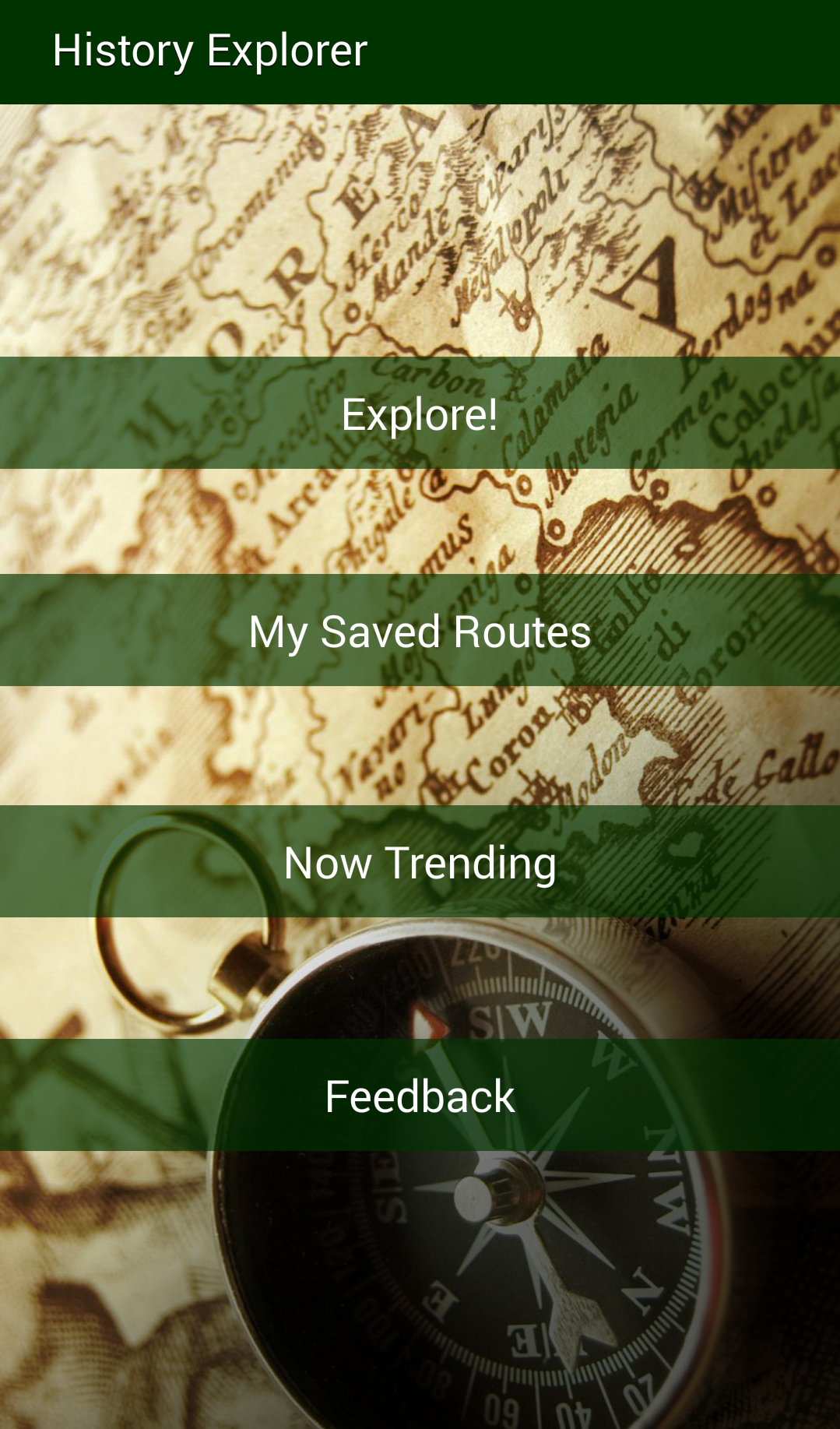
1. You should now see the application on your android device as below.



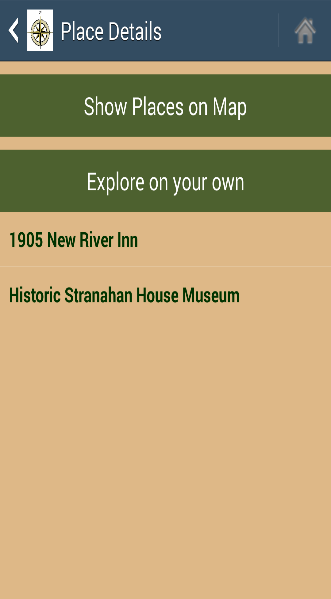
### *2.1.2 Use Case 1: AccessPlaceDetails*

Before being able to run the following use case, section 4.1.1 must be completed. Use Case HEA004: AccessPlaceDetails is performed the following way.

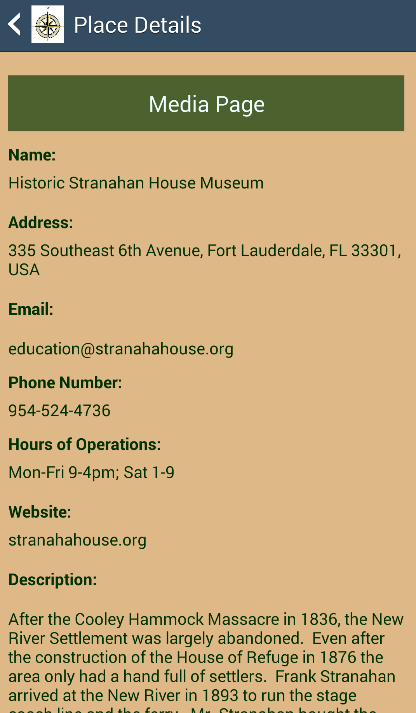
1. Start open History Explorer app by tabbing on the icon.
2. Splash screen will show up and then you will be transferred to the Main Screen.
3. Within the main screen, select “Explore”



1. You will then to transition to the PlacesNearbyActivity and will see all the places within the database.
2. Select a location. For this example, I will select “Historic Stranahan House Museum”



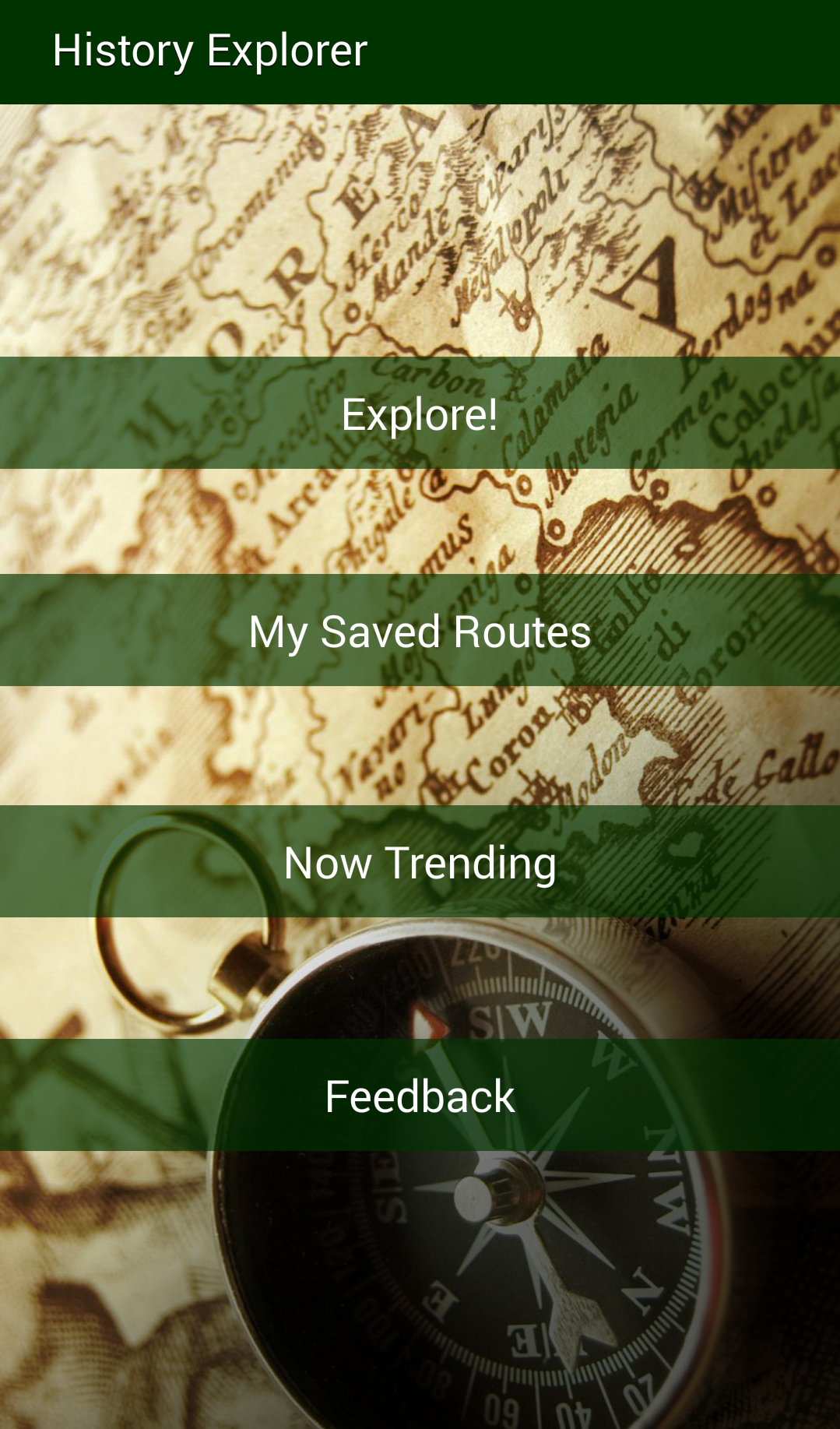
1. This will take me to the SinglePlaceActivity screen and I will be able to see all the details about this site on one screen.



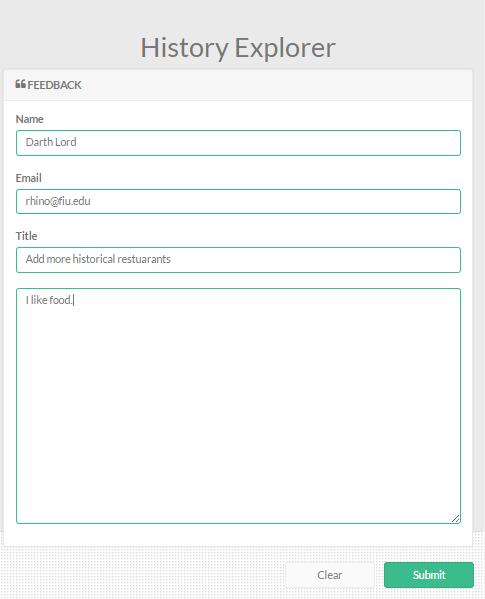
### *2.1.3 Use Case 2: AndroidSubmitFeedback*

Before being able to run the following use case, section 4.1.1 must be completed. Use Case HEA006: AndroidSubmitFeedback is performed the following way.

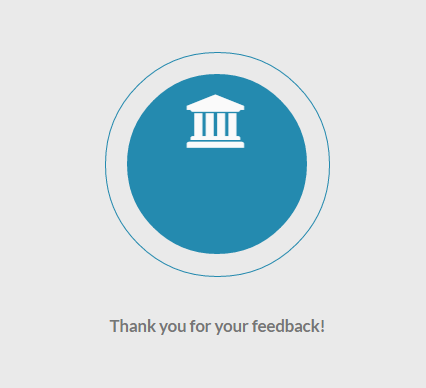
1. Start open History Explorer app by tabbing on the icon.
2. Splash screen will show up and then you will be transferred to the Main Screen.
3. Within the main screen, select “Explore”



1. Your default browser will open to “http://ha-dev.cis.fiu.edu/WebApp/#/layout/feedback
2. Fill in the following details, as such.



1. Click Submit
2. You will be redirected to the Thank page.



## 2.2 Web Application Setup

### *2.2.1 How to run the application*

### *2.2.2 Use Case 1*

### *2.2.3 Use Case 2*