## Android Fundamentals Project Self-Evaluation

**Instructions:** Once you’ve completed your Final Project, please respond to the questions below. This is a chance for you to briefly explain to the grader your thought-process during development. Once you are done, include this with the source code and accompanying files you are submitting. Then, give yourself a pat on the back for making a great app!

# Questions about Required Components

## Permissions

**Please elaborate on why you chose the permissions in your app.**

|  |
| --- |
| <uses-permission android:name="android.permission.INTERNET" />  to work with API  <uses-permission android:name="android.permission.ACCESS\_NETWORK\_STATE" />  to check if there is internet connection available  <uses-permission android:name="android.permission.READ\_SYNC\_SETTINGS" />  <uses-permission android:name="android.permission.WRITE\_SYNC\_SETTINGS" />  <uses-permission android:name="android.permission.AUTHENTICATE\_ACCOUNTS" />  for SyncManager  <uses-permission android:name="android.permission.ACCESS\_FINE\_LOCATION" />  <uses-permission android:name="android.permission.ACCESS\_COARSE\_LOCATION" />  for Location updates |

## Content Provider

**What is the name of your Content Provider, and how is it backed? (For example, Sunshine’s Content Provider is named WeatherProvider backed by an SQLite database, with two tables: weather and location.)**

|  |
| --- |
| AirProvider  SQLite database, with three tables: locations, objects, cities |

**What backend does it talk to? (For example, Sunshine talks to the OpenWeatherMap API.)**

|  |
| --- |
| Talks to carma.org API. It provides an access to a database that contains information about the carbon emissions of over 60,000 power plants and 20,000 power companies worldwide. |

**If your app uses a SyncAdapter, what is it called? What mechanism is used to actually talk over the network? (For example, Sunshine uses HttpURLConnection to talk to the network, but your app may use a third-party library to do the talking.)**

|  |
| --- |
| AirSyncAdapter  uses simple HttpURLConnection |

**What loaders/adapters are used?**

|  |
| --- |
| CursorLoaders for current and future summary fragments, objects list for location and object details.  ArrayAdapter for autocomplete text view  PagerAdapter for viewpager  CursorAdapter for objects list |

## User/App State

**Please elaborate on how/where your app correctly preserves and restores user or app state. (See rubric for examples on this question)**

|  |
| --- |
| When user goes back from settings activity, location is reloaded automatically.  When user goes back from details view, the list is on the same spot. |

# Questions about Optional Components

Answer the questions that are applicable to your final project

## Notifications

**Please elaborate on how/where you implemented Notifications in your app:**

|  |
| --- |
| Notifications are triggered on location updates. |

## ShareActionProvider

**Please elaborate on how/where you implemented ShareActionProvider:**

|  |
| --- |
| ShareActionProvider is used to share polution info for current location. |

## Broadcast Events

**Please elaborate on how/where you implemented Broadcast Events:**

|  |
| --- |
| LocationChangedReceiver is a BroadcastReceiver that handles location updates  StartupBroadcastReceiver is used to initialize the location library. |

## Custom Views

**Please elaborate on how/where you implemented Custom Views:**

|  |
| --- |
| I only used a third-party library for a custom view. |