## 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.**

|  |
| --- |
| The two permissions that we make use of are Internet and Network Access State. As far as choosing Internet, we wanted to be able to connect to the internet without a problem, this will allow the user to connect to our API where all of the JSON data is located. The Access Network State allows Parse to check the database that we have online. However, we have decided that we wish to stay with a local storage currently for our user’s created list. For future developments we want to have this state available that way we can allow user’s to share their list with others. If we didn’t have access to the Internet the main view that the app opens to would not be populated. |

## 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.)**

|  |
| --- |
| Currently, the app does not implement a ContentProvider. The app makes use of an external library in its place called Parse, this takes the job of the ContentProvider and we add in information by our other methods. |

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

|  |
| --- |
| The backend only talks to a JSON file that happens to be hosted through Github by Github Pages. The AsyncTask deciphers the text and puts it onto the view while Parse takes in the user’s list and stores it locally for us to serve back whenever the user loads back up the app. |

**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.)**

|  |
| --- |
| Well our app made use of an AsyncTask which we named DownloadDataTask. It makes use of the HttpURLConnection as well and connects to the network through that mode. It finds itself |

**What loaders/adapters are used?**

|  |
| --- |
| There were several adapters used for the RecyclerViews that were implemented in the App. Actually, every list we made was a RecyclerView. We took advantage of using |

## 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)**

|  |
| --- |
| We were able to keep our Fragment that shows both the TotalList and the CardList to maintain themselves during rotation by just preserving the location and then just reloading the view. Whenever it wakes up you can still see where you last left off and you can also see what items still exist on the grocerylist. When you close the app and reload, we use Parse to maintain the GroceryList for our user. We decided that having it locally stored is more important than having it online since we don’t want to make it dependent on the internet for them to access this information. However, in later versions we will want the online database to allow for lists to be shared. |

# 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:**

|  |
| --- |
| N/A |

## ShareActionProvider

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

|  |
| --- |
| N/A |

## Broadcast Events

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

|  |
| --- |
| N/A |

## Custom Views

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

|  |
| --- |
| N/A |