**Confluence Apps**

**A Rutgers 2019 Data Engineering and Visualization ETL Project**

**Krunal Panchal, Brown Varghese, Radha Mahalingam**

**Quick Summary**

The ever-changing mobile landscape is a challenging space to navigate. The percentage of mobile over desktop is only increasing. Android holds about 53.2% of the smartphone market, while iOS is 43%. To get more people to download various app, we need to make sure the consumers can easily find the app that they need. We offer an ETL pipeline that assists young Adults and Parents to make educated, informed choice to buy/download free/paid mobile/tablet apps from Apple store and Google Play Store. The source data obtained two different data sources in Kaggle. At the end of the ETL process, we provide our data in a NoSQL database (Mongo DB) which exposes, for different categories/genres, type and content ratings, the KPIs to determine which app store for various categories will deliver the most value to the Mobile apps consumers. This current version is focused towards data from Google and Apple App stores but can be extended to include other app stores in future. This ETL pipeline will be the backbone to build a mobile app analytics platform in future – an intuitive and user-friendly front end to help consumers as well as producers

**Steps to run the Pipeline**

1. Make sure the downloaded files - googleplaystore.csv and appleStore.csv are stored in the ‘Resources’ folder in the *pwd (current working directory).*
2. From pwd, run Jupyter Notebook.
3. Select Apps\_Data\_Extract.ipynb and run the script.
4. Select Apps\_Data\_transformation.ipynb and run the script
5. To load the CSV generated from programs above we will need to run following python script. This will require following steps
   1. Make sure MongoDB Compass is installed
   2. Launch mongod on Gitbash.
   3. Launch Terminal or command line to run the python file mentioned below:
      1. Confluence\_Apps.py
   4. Once script has run to success launch MongoDB Compass
   5. Now you should be able to see following DB and collections:

* **Database:**

Confluence\_Apps

* **Collections:**

Android\_Data.segment

Apple\_Data.segment

Apple\_Google\_Summary.segment

**Narrative / Motivation**

We are providing a Database for the Mobile apps consumers – especially young adults and parents to make informed decisions about what apps to buy / download for various categories of apps from Google Play Store and Apple store, based on apps price, user rating and number of users (installs).

**Final Schema / Data Model / How to use the data**

Explain what the final data model in your database is. Why did you make that decision and how do you expect people to use it. Entity-Relation diagrams would be great (<https://dbdiagram.io/home> or other online tools)

The process of data extraction and data transformation resulted in the following final data model, which was loaded on NoSQL database – Mongo DB. The details are provided later in the report.

**Google Play Store App Details(**Android\_Data.segment**):** This collection in Mongo DB represents the Google Play Store App Details, after data cleaning and data wrangling.

**Apple Store App Details(**Apple\_Data.segment**):** This collection in Mongo DB represents the Apple Store App Details, after data cleaning and data wrangling.

Data Table Structure for Android and Google Data Collection:

|  |  |
| --- | --- |
| Serial\_number | Unique Numbers |
| App\_Name | Application Name |
| App\_Size | Size of the Applications |
| currency | Currency Code |
| App\_Price | Price relative to Currency Code |
| Type | Type of App |
| Review\_count | Total Reviews for the App |
| User\_Rating | User Ratings for given App |
| Content\_Rating | Application Content Rating |
| Category | Category Application falls under |

**Apps Summary Details(**Apple\_Google\_Summary.segment**):** This collection in Mongo DB represents the aggregated summary level details of apps, grouped at the level of Category / Type / Content Rating. This provides valuable analytics like average user ratings, average price and Average users at an aggregate level.

Data Table Structure for Android and Google Summary Data collection:

|  |  |
| --- | --- |
| App\_Store | Application Provider |
| Category | Application Category |
| Type | Application Type |
| Content\_Rating | Application Content Rating |
| Avg\_App\_Price | Avg App Price |
| Avg\_review\_count | Avg Review Count |
| Avg\_User\_Rating | Avg User Ratings |

**Data Sources**

Write a paragraph explaining where you got the data and why you decided to use it. How does it help you solve the problem in your narrative? Dig into (1-2 sentences) into specifics about the data - eg: are there assumptions made in the data you are collecting?

We considered two datasets in .csv form from Kaggle on mobile apps from Google Play Store and Apple Store, each of which captured top 7000 to 9000 mobile apps, covering various categories. In the Kaggle site, these data datasets are infrequently updated, the latest update happened in March 2019. These data sets allowed us to build KPIs around average user ratings and average number of users (KPIs representing popularity of apps) and average price (KPI representing cost effective apps) aggregated at various categories/content rating level.

The structure of the input source datasets are as follows:



**Transformation Step**

Explain how you got your raw data into the final model. What were the specific steps you had to take to get the data into the final data model.

In order to build our final collections/database, we had to perform the following data cleaning and data transformation activities :

**Data Cleaning steps performed:**

*Brown to contribute*

**Data transformation steps performed:**

*Radha to contribute*

**Project Plan**

****