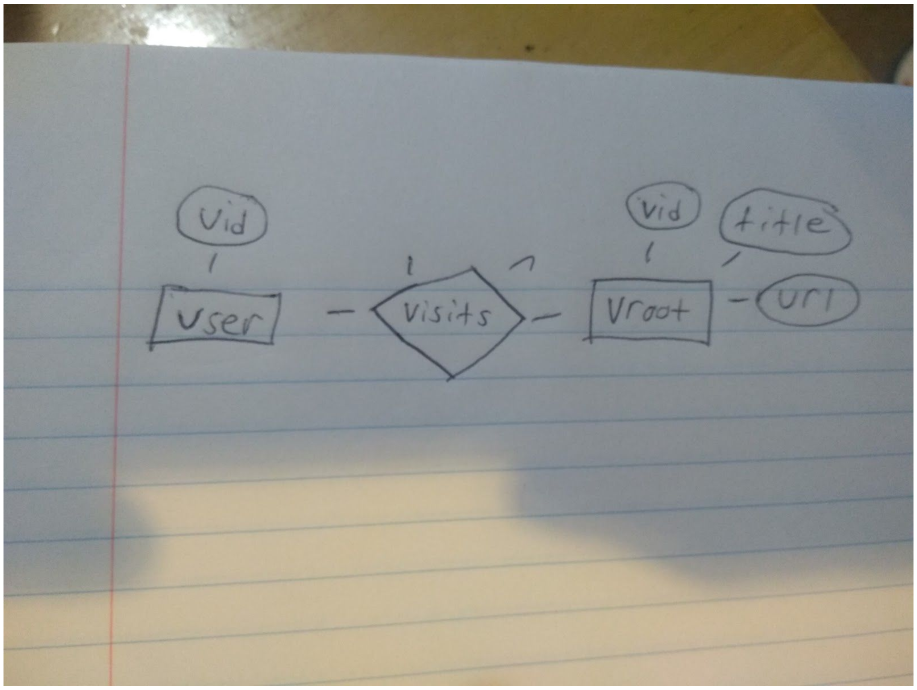
Kevin Gomes
Dr. Peckham
CSC 436
1 May 2018

Final Project Documentation

The project I have worked on was to create a PostgreSQL database and relations by gathering data from a CSV dataset using python. The dataset used was an anonymous collection of visits to specific portions of Microsoft's website. This dataset can be found here: http://mlr.cs.umass.edu/ml/. The first step was to use python to create 2 CSV files from the main dataset. One for a user, one for a vroot. A vroot is made up of an ID, title, and URL that is relative to www.microsoft.com. For example, /library would be going to the page www.microsoft.com/library. Being an old dataset from a machine learning website, many of the pages may not exist anymore. Afterwards, a PostgreSQL database was created, and 2 tables were created using SQL. They were then populated using the COPY query, and the source was the CSV files we created previously.

A simple ER diagram was followed as a base for the project. The table structure was designed with this in mind.



Base ER diagram of user and vroot relations

Many test queries were performed to make sure the data was working correctly. These include ordering my a column, using multiple conditions in the “WHERE” clause of SQL, and using different comparators. Each result was successful, and there were no problems here.

The only thing unfinished is linking the two tables together. For example, in its current state it is impossible to get a specific user that went to a specific vroot, as they are not linked by another relation. In order to complete this project, one would need to write a 3rd python script to go through the original data, and for each user in there, get all of the vroots they went to, and store both the UID and the VID into a table. This proves difficult, as for each user there are a variable amount of vroots, repeating for each user.

In conclusion, the project is 90% done and only this small hiccup remains. All data is fully imported from CSV files, and it is with this method that nearly any dataset can quickly be transformed into the relational model by altering the python scripts.