Aileen Novero

CS 8674 – Independent Master’s Project

Summer Term 2016

Review of Project

Supervisor: Dr. Ian Gorton, Professor and Director of Computer Science, Seattle Campus

*Create, Host, Deploy in Python-ese*

*Creating a mySQL database web application*

*Hosted by Flask & Publicly Deployed on Amazon Web Services*

***Introduction***

As a neophyte utilizing the numerous packages available to the ever-increasing industry used Python language, I pursued the following project in order to have the opportunity to explore and implement tools and skills prized by the Computer Science industry, in particular Data Science.

**Objective:**

To build a Web-based data capture, storage and analysis system for academic conference papers. The initial data set will be based on conference paper titles, authors, abstracts and keywords from 10 years of software architecture conferences. The aim is to provide a user with a collection of analysis methods supported by Web-based visualization to explore the abstracts in interesting ways.

**Learning Objectives:**

The project was originally designed to include three main learning objectives. Two can be coalesced into the second topic, with the third, the inclusion of the public deployment added.

* Design a database for storing textual data and associated metadata :
  + Python Relational Database
    - Python, Python Pandas, sqlite3
    - Cleaning and parsing complex text files
    - Creation of the Database
    - Using Python visualization packages : matplotlib, Seaborn
* Implement a Web-based system that enables a user to submit queries based on specific attributes of the data set (eg year(s) of publication, author, institution, topics)
* Visualize the results using Web-based visualization
  + Flask
    - Python, Flask, jQuery, apache
    - Hosting the database in a web-application useable on local browsers
* Implement and Deploy a public server for the web-based application.
  + Amazon Web Services
    - AWS, Ubuntu, apache, mod-wsgi
    - Deploying the Flask application useable to customers

**Weeks 13-14 (10%)**

         Write a short (~3 pages) that briefly describes the design, how it can be expanded to eg store data about different conferences and/or add new visualizations, and any known limitations that could be addressed in the future.

*The Database*

The database was created implementing the packages for Python utilizing Python Pandas DataFrames and sqlite3, a syntax similar to MySql. Sqlite was chosen due to its inherent compatibility with python and pandas.

The database schema:

/Users/aileennovero1/Downloads/database_sciabstracts.pdf

Improvements:

Pandas dataframe, memory/speed.

Ability to parse the files, add to the database,

Succintness of the code

Hardcoded elements

*The Flask Web Application*

*Queries and Visualizations*

Utilizing Flask, jQuery, and the Python packages matplotlib and Seaborn. The following queries and visualizations are currently available

Extras:

* About the site author: Aileen Novero
  + Links to contact information, GitHub account, Linked In Profile, and Clickable email link
* User page, contact information of site author and academic supervisor for the project
* Welcome page
* Index Page

Authors:

* View all authored papers (searchable)
  + Author Name, Paper Title, Conference, Year Published, Author’s Total Publication Count
* View published authors given a conference and year (searchable)
  + Author Name, Paper ID number, Paper Tile, Author’s Count in total papers published per that given year
* Search for and view author’s metadata of queried author’s name (searchable)
  + Author Name, Paper ID, Paper Tile, Conference, Year Published
* Visualize the top overall 20 Authors broken down by count per conference per year:
  + Area Plot
  + Bubble Chart
* Visualize all Authors as a frequency based WordCloud

Conferences:

* Search for and view metadata a queried conference and year
  + Paper ID, Paper Tile, Abstract
* View each conference and year with links to the published papers, authors, and top keywords
* View paper’s published of given conference and year (searchable)
  + Paper ID, Paper Tile, Abstract
* Visualize each conference broken down by year
  + Table of the publication count, Pie Chart breakdown, and BarPlot breakdown

Papers/Affilations:

* View metadata of a queried Paper Id
  + Paper Title, Conference, Year Published, Abstract
* View paper metadata from queried affiliation term
  + Paper Id, Affilations, link to the paperID metadata above
* View Country by Count table
* Visualize Country Affilation by Paper Counts:
  + Bar Chart
  + Area Plot
  + ?

Keywords:

* View keyword metadata
  + Paper ID, Paper Title, Conference, Year Published
* Search for and view metadata of queried keyword
  + Table of above metadata, Table of Keyword count per conference/year, link to A HeatMap rendering of breakdown
* View table of Top 20 keywords with metadata and link to HeatMap rendering
* Visualize overall Top 20 keywords broken down by year and conference in HeatMap
* Visualize keywords as a frequency based WordCloud

Tables, Relational Database Returns (all searchable)

* Total Table
  + Abstract, Author affiliation, Authors (nonparsed), Conference Name, Paper Title, Keywords (nonparsed), Publication Year
* Abstracts
  + Paper Title, Year Published, Conference Name, Abstract
* Affiliations
  + Affiliation ID, Affiliation Text
* Authors
  + Author ID, Author Name
* Conferences
  + Conference ID, Conference Name
* Keywords
  + Keyword ID, Keyword
* Papers
  + Paper ID, Paper Title, Publication Year, Conference Name, Authors, Affiliation, Abstract, Keywords
* Publications
  + Publication ID, Conference Name, Publication Year
* Composite Tables
  + AffiliationPaper
    - Paper ID, Affiliation ID
  + PaperKey
    - Paper ID, Keyword ID
  + PaperAuthor
    - Paper ID, Author ID

Improvements:

Pandas dataframe, memory/speed

Succinctness of the code, refactoring

Hardcoded elements

Extendibility

Blueprints?

Aesthetics

Collision potential

*Amazon Web Services*

*Public Deployment with Ubuntu Server*

Ugh

Mod-wsgi, apache2, ssh

Fun times