|  |  |  |
| --- | --- | --- |
| **ITMD 413-513** | ***Open Source Programming: Python*** | Final Project |

Submission of your final project must contain the following details/deliverables:

Description of your program and any resources deployable:

1. Abstract – Explain in some detail your app idea.

-use various data sources & resources to store grab & data

-make use of any machine learning capabilities

1. State any Resources used – system imports, modules/packages, kits, etc.

State any data sources (apis, etc.) & resources (packages) that were incorporated into your code work. List any relevant web link references.

3. \*Please include credentials for any user/admin logins you may have. \*\*Please include WHICH py file will be your main file to test.

Code work:

Note: Your application should contain separate module files of your code for organization and good modularity. Modules can include lists, functions, classes, etc.

1. Include at least one data source you are reading into your app for processing. Source examples: csv, xlsx, xml, json, txt, db/sqlite file formats, etc.
2. Include functionality to work your project idea in some UI.

A project idea maybe to pull data from some web source (api) to filter and analyze information from the source. Ex. -> google.finance, <https://coinmarketcap.com/api/>, <https://developer.github.com/v3/>, <https://hacker-news.firebaseio.com/v0/topstories.json>,

etc.

1. Present any data from your information gathering and learning into your console, some GUI layout or chart rendering. Data presenting can be graphic or analytical based (ex. output some key analytical findings, perform a comparative analysis of data, predictive analysis results, game stats, management reports, etc.).

Ex. **pygal** is an excellent module for the presenting of graphics, i.e., svg format, viewable easily in browsers such as FF.

Package installation instructions

**pip3 install requests**

**pip3 install pygal**

(Run the various py demo files contained in a zip file in BB under your final project folder, of pygal and git repo api calls, etc. Make sure to hover over bar columns with the generated svg file for any of the repo py files for nice *tooltip* like results of data)

Ex. demo file:

Python file:python\_repos.py

Genetated file:python\_repos4.svg

Awesome reference links:

<http://www.pygal.org/en/stable/index.html>

<http://www.pygal.org/en/stable/documentation/configuration/rendering.html>

1. Package all your py files, any source files and any auxillary files (db, xml,json files, etc.) you may have that need to be included to run your app. Include as a zip file name **PYFP\_YOURCOURSE#\_YOURNAME.zip**. If you are turning in a project as a team, you may submit your project files by a single member. MAKE SURE to include your team members in your title page in your doc file. Include your abstract, statement of resources and any relevant snapshots depicting your app in action, labeled properly (including any possible Extra Credit), into a doc file and submit them all into BB.

Any projects submitted by a team will involve more grading scruntiny. Your project will thus be graded on your app’s robustness, functionality, greater detail in work (i.e., use of more modules, solid UI (multiple controls, very functional UX) and/or displays of information).

Grad expectations:

-Must perform work in some GUI- ex. tkinter, Django/Flask

-Include a db file (mysql, sqlite, postgreSQL, etc) that will hold all pertinent data for any of your app IO needs.

-Error traps

-Thorough commenting & program descriptions in each py file

-Heirarchy chart of your modules for each py file

Extra credit options

-Unit testing

-Machine learning / NLP

-Use of mulitple packages for deployments

Grading Rubric

|  |  |
| --- | --- |
| Proper documentation | 25 points |
| Source file included (db, json, csv, etc.) | 25 points |
| Program Functionality | 100 points |
| Output | 50 points |
| Total | **200 points** |