

Final Project Details

CIS 192

November 1, 2017

1 Overview

The final project is your chance to explore something you're interested in using Python. We want to see what you've learned here and we're excited to see the projects you develop. In the past, we've had interesting websites, games, and ML projects.

2 Requirements

2.1 Project Rules

Your project should take each group member roughly 10 hours of work. Successful projects can be completed in less time with good effort, and ambitious students may find themselves working slightly longer.

You are welcome to work alone or with a partner, but no groups any larger than two people will be permitted.

2.2 Deadlines

- Your proposal is due Wednesday, November 8th at **1:29 PM**. Your proposal should be between 100-400 words and will include the names of everyone in the group. The proposal will be a rough outline of your intended project and should make note of how you attempt to meet the rest of the requirements listed below. Submit this proposal by email and note that **all proposals must be approved by Harry to be valid project choices**.
- Final code is due December 11 at 11:59 PM. This will be submitted as a .zip to Canvas and must include a README file explaining your project structure and how it meets the requirements.
- Project Demos (December 12 or 13, TBD) will be conducted during reading days. If you will be away, please let Harry know to schedule a make-up time before or after to do your demonstration. The demo will take no more than 10 minutes.

2.3 Sample Ideas

- An app that analyzes live Tweet sentiment about a topic.
- Create a python library that does something useful. Check <http://pypi-ranking.info/alltime> for ideas.
- Use a WEB REST API for Dropbox, Facebook, etc. to do something cool
- Build a game.
- Scrape the web for betting data
- Create a visualization library for a particular field (random graphs)
- Make a phoneme-checker
- Chatbot
- Implement Lambda Calculus in Python

2.4 Coding Requirements

- Define and use a custom class that features a magic method (other than `init!`). It's okay if this class inherits from something other than just `object`.
- Use at least three of these modules: `collections`, `copy`, `functools`, `os`, `itertools`, `random`, `re`, `sys`, `math`, `datetime`, `random`, `heapq`, `requests`, `BeautifulSoup`, `flask`, `json`, `pickle`, `numpy`, `scipy`, `matplotlib`, `scikit-learn`, `pandas`, `nltk`.
- Use at least one of a decorator or a generator function.
- A `readme` file explaining how you meet the requirements, how to use your app, and the names of the group members.

3 Grading

Rubric to be hosted on the course website under "project".