Final Project Submission Guidelines

For project requirements, please carefully review the **Final Project Overview.** Details regarding expectations for “level of challenge” for data source access and volume are specified in the Overview.

As your final set of deliverables for the Final Project, you will prepare and submit a 2-4 page document and a 1-3 minute video. The document should be submitted in PDF format and the video must be available as a web link and configured to allow access to all members of the SI 507 teaching team.

Several elements of your final Project Document are the same as elements of your Data Checkpoint. If the information has not changed you are free to re-use those components of your Checkpoint for your final Document.

Project Document

Project code (1/4 - 1/2 page)

- Link to github repo for your final project code
- A README containing any special instructions for running your code (e.g., how to supply API keys) as well as a brief description of how to interact with your program.
- Any required Python packages for your project to work (e.g., requests, flask). You do not need to name built-in packages (e.g., random, json)

Data sources (1/2 - 1 page)

- For each
  - origin, including URLs for data and documentation
  - format(s) (e.g., JSON, CSV, HTML)
  - brief description of how you accessed the data, and whether caching was used
  - summary of data
    - # records available (OK to estimate if precise number not known)
    - # records retrieved (OK to estimate if precise number not known)
    - description of records, including important fields/attributes of each record for the purpose of your project and what they represent

Data Structure (1/2 - 1 page)

- README describing your Data Structure (graphs or trees)
- A python file that constructs your graphs or trees from your stored data using classes (or some other method)
- JSON file with your graphs or trees

- A stand alone python file that reads the json of your graphs or trees.
- Screenshots showing your data and data structures

Interaction and Presentation Options (1/2 - 1 page)

- High-level, plain-English description of the user-facing capabilities of your project—what options does the user have for selecting and displaying data?
- Interactive and presentation technologies used (e.g., Flask, Plotly, command line prompts)
- Brief instructions for how a user would interact with your program

Demo Link (1 line)

- Provide link to demo video

Demo Video

- Show how a user would interact with your program, demonstrating at least 4 different options that can be chosen)
- Either provide narration or in-video text descriptions to explain what is happening

Rubric

Component	Requirement	Points
Project code	Valid GitHub repo link is provided	5
	README is included in repo containing special requirements and required packages (including “None” if appropriate), as well as brief instructions for how to interact with your program.	10
Data sources	Origin and documentation for each data source is provided accurately	5
	Access techniques are clearly described	5
	Caching is used where appropriate	5
	Data summary is provided and relevant data fields are described	10
	Level of challenge for data access (including access methods and number of records) is appropriate	10
Data Structure	README explains how data is organized into data structure	10
	Required .py and .json files are provided that demonstrate organization of data into data structures	5

	Screenshots of data & data structures are provided and match description	5
Interaction/ Presentation	Application capabilities and interactive/presentation technologies are described clearly	5
	Instructions for use are provided and easy to understand	5
	Level of challenge for interaction and presentation is appropriate	10
Demo	Link to demo video is provided and works (including correct sharing permissions)	5
	Application capabilities are described and demonstrated clearly	5
	Four or more different user selections are demonstrated	10
	Data presentations look good and make sense	10
	Total	**120**