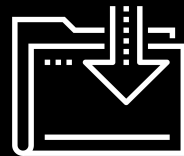




Project #2: Visualize Me, Captain!

Data Boot Camp



Project Requirements

Project Description

01

Your task is to **tell a story** with data visualizations.

02

Focus on providing users an **interactive means** to explore data themselves.

03

Prepare a **10-minute presentation** that lays out your theme, coding approach, data munging techniques, and final visualization.

04

You may choose a project of any theme, but we encourage you to **think broadly**.

05

You will have **ample time in class** to work with your group, but expect to put in **hours outside of class** as well.

Detailed requirements

Technical:

- Create a flask application. The flask application will be used for the following:
 - a. Queries data from a SQL database using SQLAlchemy
 - b. Returns data to users via REST API endpoints
 - c. Returns web pages that contains your visualisations
- Store your data in a SQL database. It is recommended that you use SQLite so as to reduce complexity of hosting a database on the cloud. You will need to write code to do a once-off write in order to store the data into the SQL database.
- Create REST API endpoints using flask. Each REST API endpoint should retrieve data from a certain table in your database, and return the data to the user in JSON format. The API endpoints you create should be used by your JavaScript web visualisations.
- Create web pages to provide users with a user interface for interacting with the visualisations. You have full creative reign on how the website should look and feel. You should use at least HTML, CSS and JavaScript when building your web pages. Use of CSS frameworks like bootstrap or materialize is encouraged to give your web page an impressive design!

Detailed requirements

Technical:

- Create at least 5 visualisations using at least one of the following visualisation libraries (or something we haven't covered in this course - please check with the instructional team first):
 - a. Leaflet.js, Plotly.js, D3.js
 - b. Your visualisations will fetch data from the REST API endpoints you have created in the earlier step.
 - c. Your visualisations will be scored based on:
 - i. Function: does the chosen visualisation provide people with answers to their questions? See:
https://legaltechdesign.com/LegalDesignToolbox/wp-content/uploads/2015/09/choosing_a_good_chart.png
 - ii. Style: does the visualisation make use of good design choices e.g. font sizes, markers, tooltips, x and y axis, labels, etc. See tips here:
<https://www.tableau.com/learn/articles/data-visualization-tips>
- Deploy your web application to a web hosting service. We recommend deploying to Heroku. However, you are welcome to deploy it to other web hosting services.
- Provide clear documentation in the form of a README.md file at the root of your repository.
- Evidence of git branching, regular commits and pull requests.

Detailed requirements

Technical - bonus:

- Incorporate API documentation using Flask RestX: <https://flask-restx.readthedocs.io/en/latest/>
- Include user-driven interaction to filter data in the visualisations. Use different buttons/dropdowns/etc and JavaScript event listeners.
- Use of at least one JS library or plugin that we did not cover.

Presentation Requirements

Your team will also be responsible for preparing a formal 10-minute presentation that covers:

- Describe the questions you and your group found interesting, and what motivated you to answer them?
- Describe the process and steps taken to create your solutions i.e. loading data into a database, creating an API, creating your visualisations.
- Present your visualisations and demonstrate how they can be used to answer your group's questions.
- Retrospective - talk about the wins and challenges that your group faced when building the solution.

Project teams

Group 1	Group 2	Group 3	Group 4
Monika Sidhu	Oksana Ayres	Claudia Valenti	Trushil Patel
Amy Zhang	Laurent Ah-Sue	Carolina Dias Do Vale	Shruti Hindocha
Asfia Khatoon	Brianne Ng	Daniel Bourke	Laura Trentino
Pablo Crespo	Walton Tan		

Rubric - technical

SECTION	WEIGHT	REQUIREMENT
Technical	5.0%	Create a flask application. The flask application will be used for the following: - Queries data from a SQL database using SQLAlchemy - Returns data to users via REST API endpoints - Returns web pages that contains your visualisations
	5.0%	Store your data in a SQL database. It is recommended that you use SQLite so as to reduce complexity of hosting a database on the cloud. You will need to write code to do a once-off write in order to store the data into the SQL database.
	15.0%	Create REST API endpoints using flask. Each REST API endpoint should retrieve data from a certain table in your database, and return the data to the user in JSON format. The API endpoints you create should be used by your JavaScript web visualisations.
	10.0%	Create web pages to provide users with a user interface for interacting with the visualisations. You have full creative reign on how the website should look and feel. You should use at least HTML, CSS and JavaScript when building your web pages. Use of CSS frameworks like bootstrap or materialize is encouraged to give your web page an impressive design!
	30.0%	Create at least 5 visualisations using at least one of the following visualisation libraries (or something we haven't covered in this course - please check with the instructional team first): - Leaflet.js - Plotly.js - D3.js Your visualisations will fetch data from the REST API endpoints you have created in the earlier step. Your visualisations will be scored based on: - Function: does the chosen visualisation provide people with answers to their questions? See: https://legaltechdesign.com/legalDesignToolbox/wp-content/uploads/2015/09/choosing_a_good_chart.png - Style: does the visualisation make use of good design choices e.g. font sizes, markers, tooltips, x and y axis, labels, etc. See tips here: https://www.tableau.com/learn/articles/data-visualization-tips
	10.0%	Deploy your web application to a web hosting service. We recommend deploying to Heroku. However, you are welcome to deploy it to other web hosting services.
	2.5%	Provide clear documentation in the form of a README.md file at the root of your repository. The README.md file should explain the following: - Purpose and motivation - Solution - explain how your solution works. - Usage - How to install Python requirements / dependencies - How to run the code locally - How to deploy the code to your web hosting service
	2.5%	Evidence of git branching, regular commits and pull requests.
Technical - bonus	5.0%	Incorporate API documentation using Flask RestX: https://flask-restx.readthedocs.io/en/latest/
	10.0%	Include user-driven interaction to filter data in the visualisations. Use different buttons/dropdowns/etc and JavaScript event listeners.
	5.0%	Use of at least one JS library or plugin that we did not cover.

Rubric - presentation

SECTION	WEIGHT	REQUIREMENT
Presentation	2.5%	Describe the questions you and your group found interesting, and what motivated you to answer them
	2.5%	Describe the process and steps taken to create your solutions i.e. loading data into a database, creating an API, creating your visualisations.
	12.5%	Present your visualisations and demonstrate how they can be used to answer your group's questions.
	2.5%	Retrospective - talk about the wins and challenges that your group faced when building the solution.

Suggested data sources

Description	URL
Australian Government Open Source Datasets	https://data.gov.au/
Kaggle Open Source Datasets	https://www.kaggle.com/datasets
Public APIs	https://github.com/public-apis/public-apis
Australian Bureau of Statistics	https://www.abs.gov.au/
World Bank Open Data	https://data.worldbank.org/
World Health Organization Data	https://www.who.int/data/gho/
FiveThirtyEight (news agency open data)	https://data.fivethirtyeight.com/
Google dataset search	https://toolbox.google.com/datasetsearch

Schedule

Day 1 (Today):

Start brainstorming topics with your group and researching potential data sets. Your focus should center around:

- Selecting a topic
- Finding a data set
- Finding inspiration
- “Sketching” your ideal visuals
- Starting a 1-page proposal

Day 2:

- You will need to finalise a 1-page proposal that includes:
 - A brief articulation of your chosen topic and rationale
 - A link to your data set(s) and a screenshot of the metadata if it exists.
 - 3 or 4 screenshots of relevant, “inspiring” visualizations that frame your creative fodder
 - A sketch of the design
 - A link to the primary GitHub repository you’ll be housing your work in
- Project work
- Deploying web applications to Heroku (demo)

Day 3:

- Project Work
- R lesson (optional and recorded)

Day 4:

- Project Work
- R lesson (optional and recorded)

Day 5:

- Project Work

Day 6:

- Project Work

Day 7:

- Group Presentations
- Project Submissions by Tuesday, January 25, 2022 11:59PM



Questions?