Customizable Analysis and Visualization Tool for COVID Cases

Milestone 2

Team Members

- Calvin Burns, cburns2017@my.fit.edu (Team Lead)
- Sam Hartle, shartle2017@my.fit.edu
- Nicole Wright, nwright2017@my.fit.edu
- Stian Olsen, shagboeolsen2017@my.fit.edu

Faculty Advisor/Client

- Dr. Philip Chan, pkc@cs.fit.edu

Progress Matrix

Task	Completion %	Stian	Sam	Nicole	CJ	To do
1. Setup Django environments	100%	20%	20%	20%	40%	none
2. Create database models	100%	25%	25%	25%	25%	none
3. Import data to newly created database via CSVs	100%	25%	25%	25%	25%	none
4. Implement Feature 4.1 (Customizable Operations on Variables)	25%	0%	10%	0%	15%	UI and supporting backend code

Task 1 Summary - Set up Django Environments

- Different Operating Systems
  - Creating unique scripts for macOS and Windows
- Allowing each team member to create admin users, manage migrations, and carry out features built into Django.
- Challenges:
  - Running Postgres database with non-default credentials on Windows
  - Understanding Django workflow

Task 2 Summary - Create Django Models

- 8 models in total
  - User - Representing a basic account
  - Location - Representing a single location
  - Metric - Representing a statistic that is not time related
  - Datapoint - Represents a single data point associated with a dataset
  - Dataset - Collection of *Datapoints*
  - Operation - Represents an abstract, editable math operation or formula
  - Plot - Represents a saved plot
  - Dashboard - Saved configurations of plots, tables, and maps

Task 3 Summary - Unique Datasets

- Each of us wrote scripts to import a unique CSV file
  - FDOH case data
  - Lockdown Dates by country
  - Lab Testing throughout Florida
  - Mask Mandate Dates
- The data is stored using the models created in Task 2

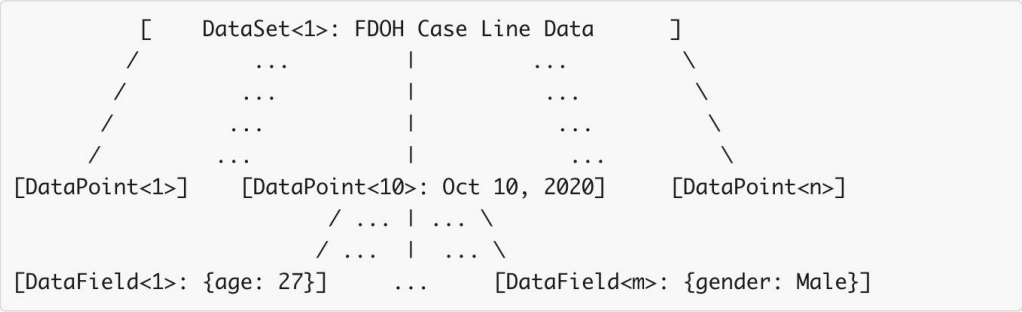
Task 4 Summary - Feature 4.1

- 25% completed
- Models created and can be used for operations
  - Model utilizes SymPyCharField
  - Allow a user to input operations supported by the SymPy library
- To do:
  - Implement a UI so the users can interact with the dashboard.
  - Write backend code to process the custom operations

Demo - Male vs. Female Pie Chart

Challenges

- Figuring out a good design for importing and running queries
  - Large amounts of Covid data
    - Files with more than 750,000 lines



The diagram above gives a visual representation of a `DataSet` with *__n__* number of `DataPoints` and each `DataPoint` has *__m__* `DataFields`.

- Setting up Django environment on Linux and Windows

Task Matrix for Milestone 3

Task	Stian	Sam	Nicole	CJ
1. Continue Feature 4.1 (Customizable Operations on Variables)	Design a frontend operation selection UI based off mock-up in Design Doc	Work on majority of the backend code for processing selections from fronted for custom operations	Assist Stian with UI for selecting custom operations on the fronted	Further investigation of SymPy library to see how it will integrate with Sam's writing of backend operations code
2. Small GUI demo that integrates lockdown and mask mandate data	Create a line graph demo which displays a timeline using the dates in the lockdown and mask mandate datasets	Support Stian and Nicole with specific operations they need for their demo	Create a line graph demo which displays a timeline using the dates in the lockdown and mask mandate datasets	Oversee development and assist as needed since demo will be similar to Male/Female pie chart demo
3. Consider different options for saving plots	Look into a solution which separates frequently used datasets from datasets that are not accessed as often	Look at past datasets for how often occurrences of newly added or changed previous data happens	Assist other team members with research options	Look into appending newly added data

Questions?