DataWin: Elaboration Iteration I Report

Open Data Access Portal by Nidhi Patel & Niraj Patel

University of Windsor

Dr. Kobti

November 9, 2020

Shortcut Link:

Code Repository: https://github.com/nidhipatell/WinData

The above link will take you to the main code repository where you can access everything else.

Bug Reporting: https://github.com/nidhipatell/WinData/issues

The above link will take you to the issues form where you can view all the tickets created for the issues the team encountered.

Project Management: https://github.com/nidhipatell/WinData/projects/1

The above link will take you to the GitHub project management for WinData.

Documentation Repository: https://github.com/nidhipatell/WinData/wiki

The above link will take you to the documentation repository where you will find all the wiki.

Testing Strategy: https://github.com/nidhipatell/WinData/tree/main/Iteration%20I/testing

The above link will take you to the testing strategy used by the team. Please read ahead, the report talks about why and how we chose our testing strategy.

Elaboration Iteration I

This is the Elaboration Iteration I report, the main purpose of this document is to provide an overview of what happened in Iteration I. Previously an Inception Report has been completed to understand the problem and the requirements from this project. This iteration was mainly focused on the object-oriented analysis part and a little bit of the object-oriented design as well. Numerous different types of UML diagrams have been created to provide a nice overview of the design of this project.

Revised Phase Plan

Iteration I had to be modified due to the complexity of the requirements. The top-level requirements are not broken down in more subparts, so they are easier to approach and tackle. The original plan was to get the front-end done with data visualization, as well the ability to download data. Due to the complexity of those task, they were modified to be accomplished in the first iteration. One of the main reasons why the team had to a setback, is 2-4 days were spent learning GitHub and its project management tools, along side this the team also had to learn version control. By the end of the phasing planning the technology stack was revised as well.
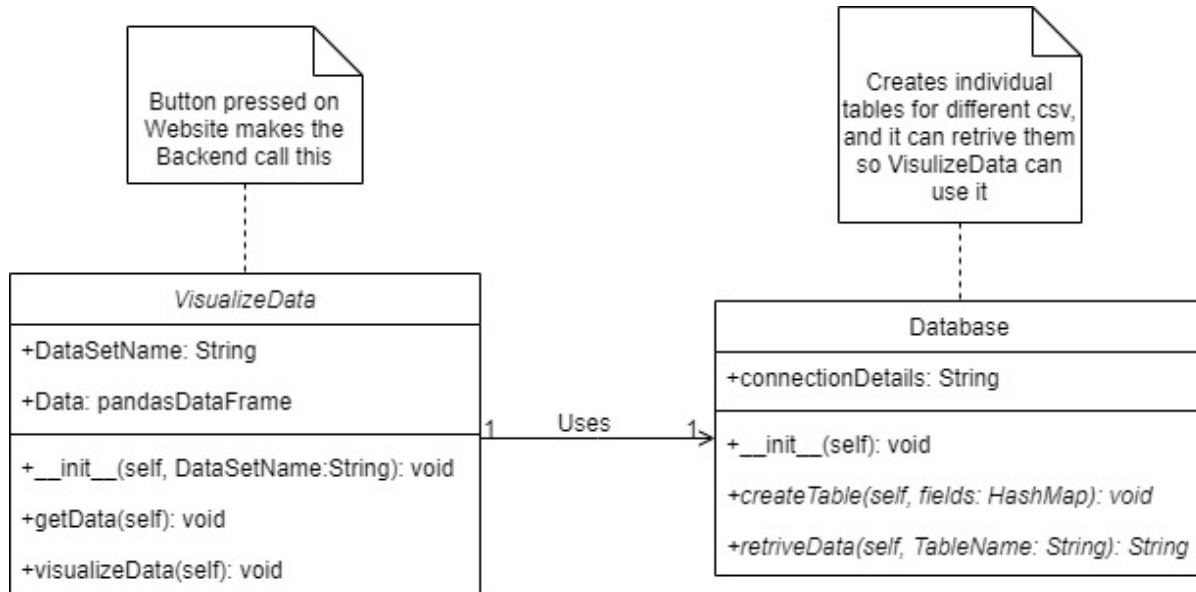
- Back-end: Flask
- Front-end: HTML/CSS
- Data visualization: Matplotlib, Pandas
- Database: SQLite/MySQL

UML Diagrams

The following are representation of the system done using UML class, sequence, and use-case diagrams.
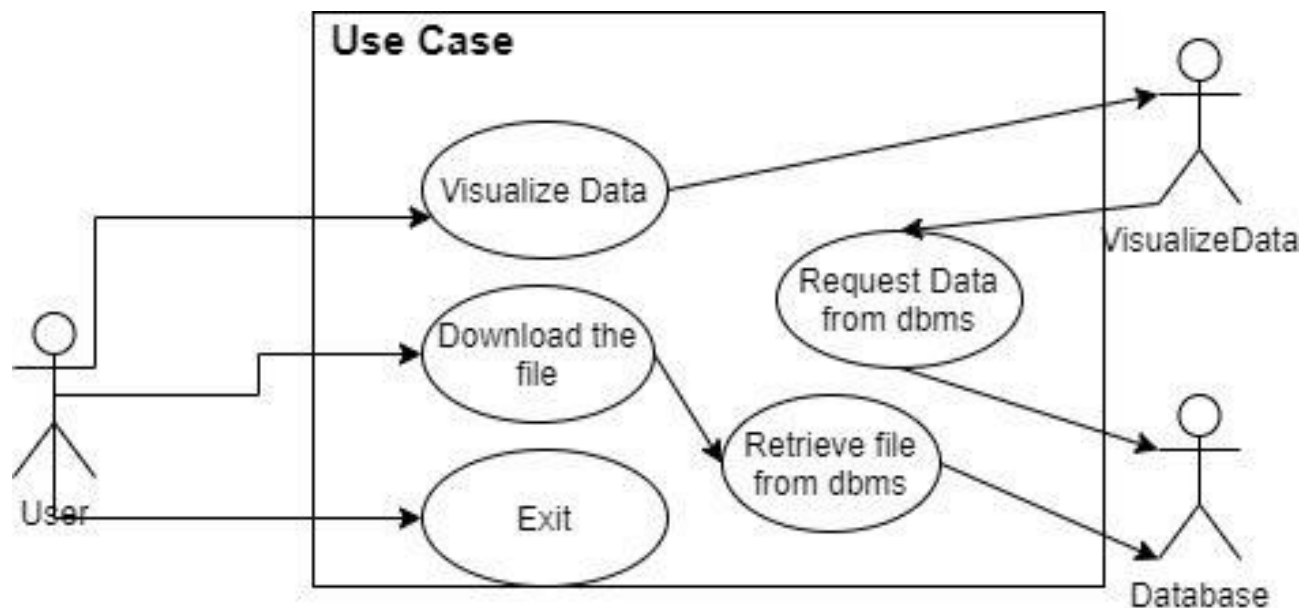
NOTE: The class diagram is made for Python hence why there is no constructor like the ones in Java but rather there is the __init__ method which acts as the constructor in Python OOP.

This is the representation of the backend with 2 different classes. When the user uses the website and clicks the button (which allows them to display data). Flask will automatically call VisualizeData and it will send the filename for the dataset that is going to be displayed. VisualizeData will then create an instance of the Database and retrieve the data and then display it using Matplotlib. A sequence diagram of the following will represent the above scenario.
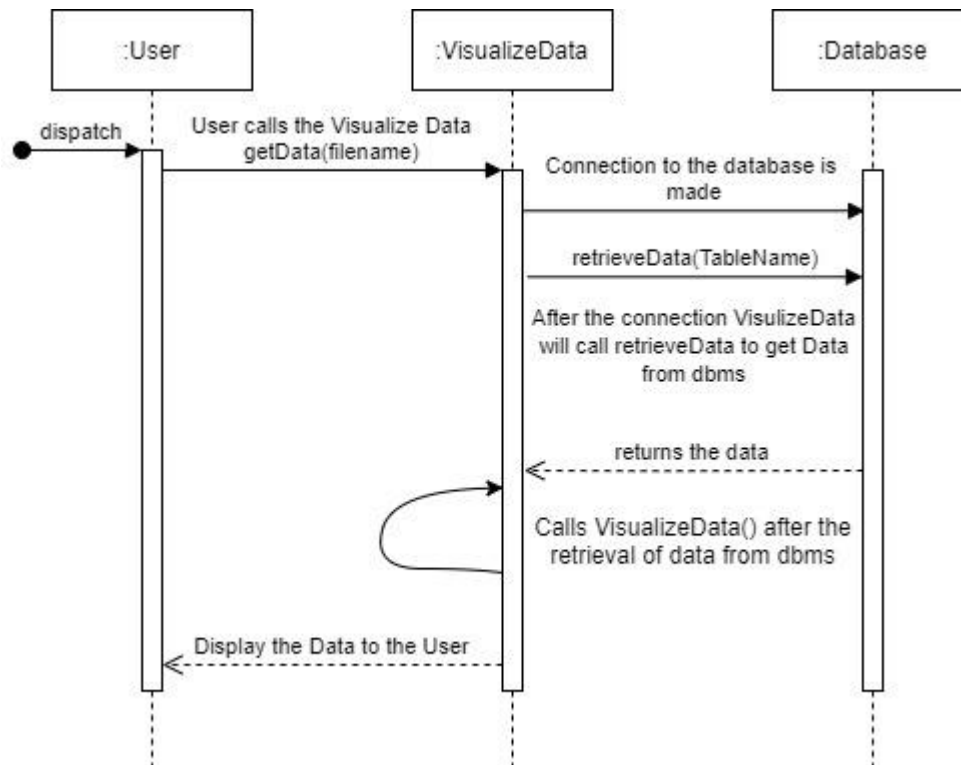
The above sequence diagram describes the classes and methods that will be used to complete the above scenario.

Goals and Constraints

- Implement the frontend, the title page at least.
- Create a sample data visualization as an example of what will be implemented in future iterations
- Upload data to SQLite or MySQL

Outcome

The initial front-end of the website is complete, and progress is made continually. For a prototype, a sample data set was chosen from https://opendata.citywindsor.ca/. For the testing the team decided to withhold on any unit test for at the moment and decided to test the DataVisualization.py with the sample dataset and that can be found in the testing folder, link is attached on 2nd page.
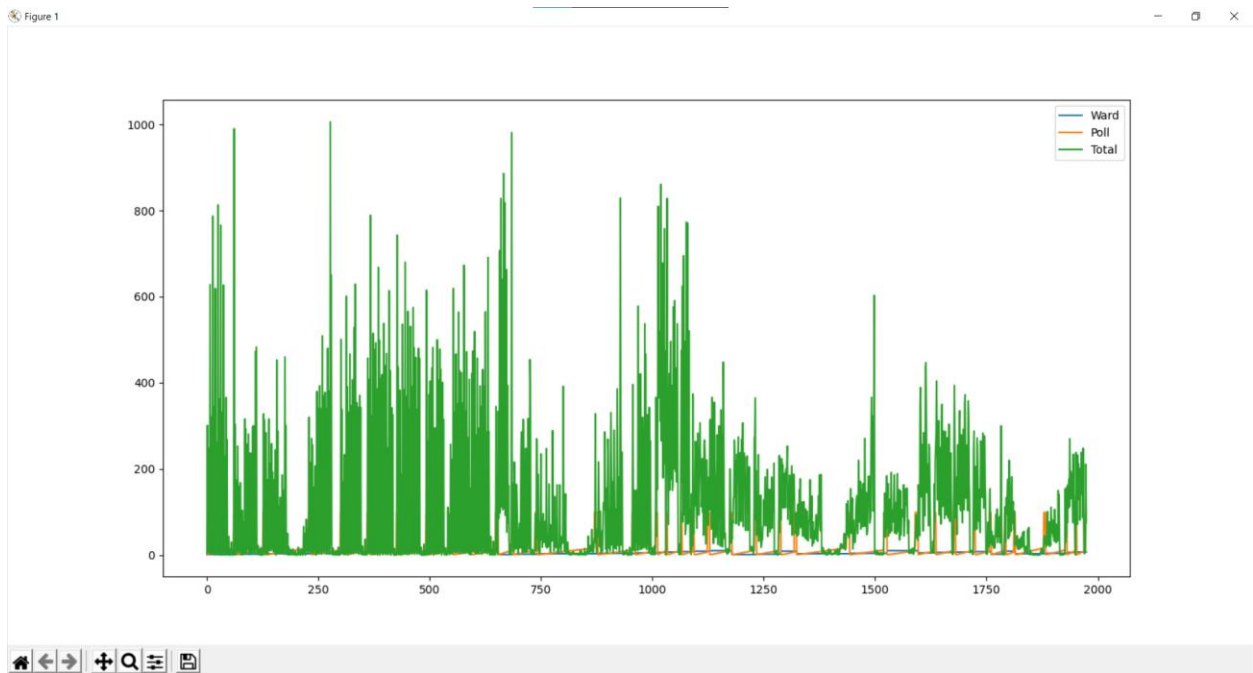
Prototypes

Below is the attached image of what has been accomplished in the project so far. This is the iteration I prototypes, and the next iterations' systems will be developed using these prototypes.

Website: A basic layout this is the title page that will be presented to the users

Here is an example of a dataset that is graphed

Other types of graphs will also be provided depending on the dataset.

Next Iteration's Plan

The plan for the next iteration is to further the process of the project but implementing the remaining features. As of now the system takes the data from a testing/sample file and it shows a graph for it. Since our overall requirement is to have the graph displayed on the webpage when the user presses the button for it. Another major plan is to update the website to allow the users to be shown all the datasets available to them to use.

Risk

- The datasets on the website will need to be updated if the original source updates theirs
- Using a database will be good in the long run but due to the project timeline it might be easier to store csv files and use them for the visualization