

Interactive Museum Timeline

4P02 Progress Report 1

COSC 4P02
Software Engineering II
Winter 2023
March 5, 2023

Instructor: Nasser Ezzati-Jivan
Brock University

Alex Duclos
ad19ua@brocku.ca
6738884

Goktug Cirag
gc19cy@brocku.ca
6776678

Eduardo Saldana
es18za@brocku.ca
6612626

Jashan Pannu
jp18jj@brocku.ca
6505861

Rishabh Rai
rr19pa@brocku.ca
6847156

1. Sprint Review

During this sprint, we intended to target the most critical requirements of the program and program a base to build off of. In this sprint we developed a user interface in the front end and a structure for the timeline in the backend. We intended to finish this sprint with all the working features needed for a basic and deployable program, however we ended it with a much more bare bones product due to challenges that were not solved during this sprint. Due to challenges later discussed in this report, we opted not to link the front and backends until issues are resolved.

For the first sprint our plan was to implement the following:

1.1 Timeline and Nodes

This is the most crucial feature of the backend since it is the structure of the timeline and requires the most consideration. All other features in the program are built off of or attached to the timeline in some way, so its implementation must be built to accommodate any expected features and be straightforward to change for feature ideas that arise in the future.

We used a linked list structure to represent the timeline and created a node class for it to contain. A planned feature is the ability to branch off into subnodes, we considered this and plan to address it by having nodes themselves contain a linked list of their subnodes to form a graph like structure.

1.2 Node data

The textual data contained within nodes is another critical feature due to the primary purpose of the program being its ability to convey and share information to the user. Nodes are able to contain a variety of data and be completely editable for the sake of timeline planning. The timeline structure itself is also capable of adding and managing nodes in alternative ways so that the overall structure is adequate for future features.

1.3 Node media

Handling media in this project is important to this project as images, videos and audio files are essential for user experience and immersion. The front end requires a reliable and usable method of interacting with media linked to nodes in order to create our implementation of a digital exhibit. This feature has not been completely addressed in this sprint since we failed to find a method of handling media which would be most suitable for this project. Our current partial solution for this sprint involves linking to media over the web and storing references within a csv file.

1.4 Basic User Interface

A user interface is a basic requirement for this program to serve its purpose. We've implemented a basic interface with features it must be capable of handling in the future such as traversing the timeline, viewing information and media related to a given node, and the management of information in nodes.

1.5 Digital Timeline Tour

This feature involves a guided and interactive traversal of the timeline for the user which serves as a digital tour. The timeline and nodes have been implemented in a way to make this feature simple to implement since it's simply a linear traversal of linked lists which progresses with user input. At this stage we have not started the implementation of this feature since we have not completely addressed the final state of how node media will be handled. Since a large portion of this feature will be presenting media and information to the user we've pushed the priority of node media to make a prerequisite of this feature.

2. Next Sprint Planning

For the sprint our overall goal is to finalize the features we were unable to complete within the first sprint as well as work on the secondary layer of features which required what we have finished as a prerequisite. By the end of the second sprint we hope to have a deployable application with all the necessary features to call it a working interactive timeline. To do this, we'll be looking at backlog items at the next 2 priority levels. The first sprint looked to complete everything at the highest priority, which was every feature at the base of the project and required done in order to build off of for developmental progression. As we previously took every item in our log at a priority of 5, on our one-to-five scale, we will now be adding features with a priority of 4 and 3.

Our next sprint will also follow up on what we've learned from addressing the issues in our first sprint, discussed in the next section.

3. Sprint Retrospective: Issues Encountered

During this sprint the main issue we encountered revolved around components which many different features interact with. For example, when deciding on the method of media management across the system we need to consider both how the front end will receive it and the back end will work with it. By developing separately without this consideration we ended the sprint with a partial placeholder solution, ultimately stunting our progress and having led to a waste of development time and effort. We've addressed this by bringing anyone involved with working on such components into discussions regarding implementation methods and ensuring such implementations are consistent across the whole system.

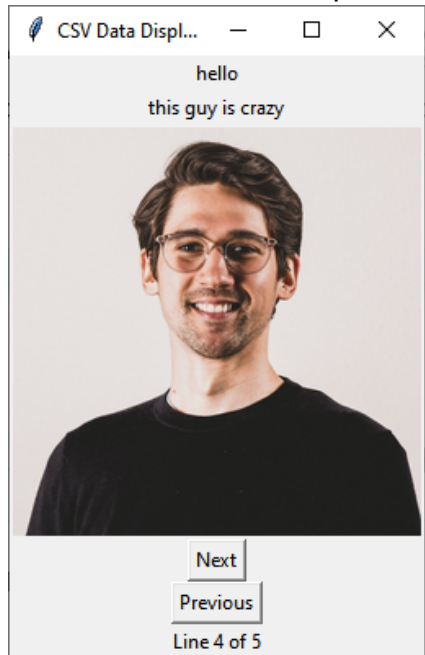
Another issue we encountered early was a lack of overall testing. Due to this we were not able to complete a proper linking of the front and backend and attempts to do so led to finding errors preventing the process. We resolved this issue by accounting for dedicated testing time and put efforts into creating more comprehensive sets of test cases to catch issues. We've come to the understanding that it is much easier to test an individual component for what's expected than to troubleshoot an error found in the interaction of multiple.

4. System Overview

3.1 Main

Main is the Tkinter window that launches the timelines, selecting a timeline and displaying the required data and images in Tkinter. It also allows the user to continue down the timeline going left and right.

Here is an example screenshot from running this program.



3.2 CSV_edit.py

In earlier implementations, we read from a CSV to output to tkinter. This file has all the required functions for that. Selecting a file, deleting rows, and appending data are all nicely wrapped up to be used by an inexperienced user with tkinter.

Here are example screenshots from running this program

✎ □ ×

Select a CSV file:
No CSV file selected

Title:

Description:

Image URL:

✎ CSV Data Appen... — □ ×


Select a CSV file:
C:/Users/Alex/Desktop/4p02/COSC4P02/data.csv

Title:

Description:

Image URL:

✎ Success ×

 Data appended to the CSV file.

And the new 6th slide of main.py



3.3 Timeline Structure

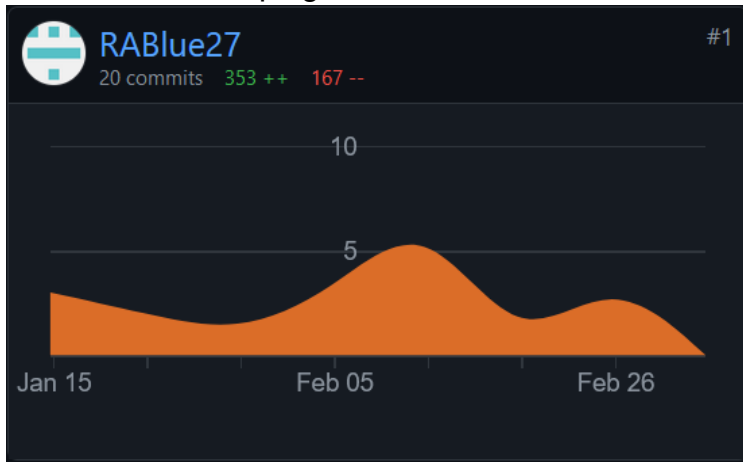
The timeline structure consists of the use of linked lists and nodes. Every function and method created is intended for use in future features. Considerations for different use situations added the need for multiple functions with identical results. For example, we have two `addNode` and `addNewNode` functions. One allows for adding an existing node passed as a parameter and the other takes the necessary data as parameters to create a new node to be added.

The lists handle adding nodes to a specified position, removing specified nodes, moving a specified node to another position in the list, retrieving a node at a specified position, displaying the contents of the list, and functions for efficiently adding or removing to the end of a list. Nodes are implemented to be data oriented and contain all the necessary data for each digital exhibit as well as having the functions needed to edit the data a node stores.

5. Team Contributions

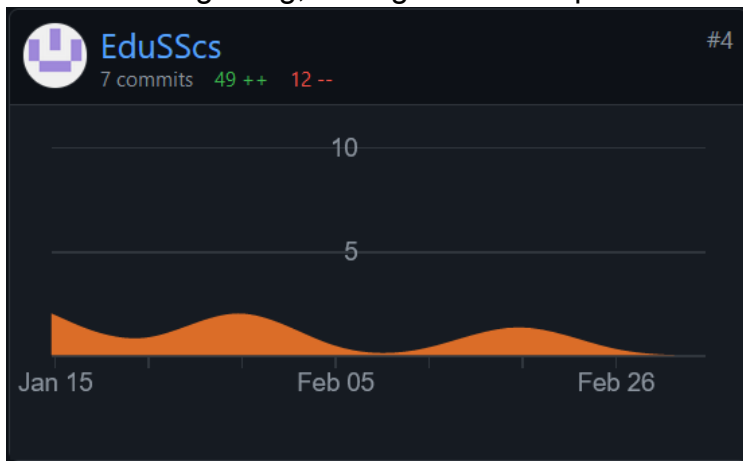
Alex Duclose - 6738884

Worked on developing the main user interface and related csv file management



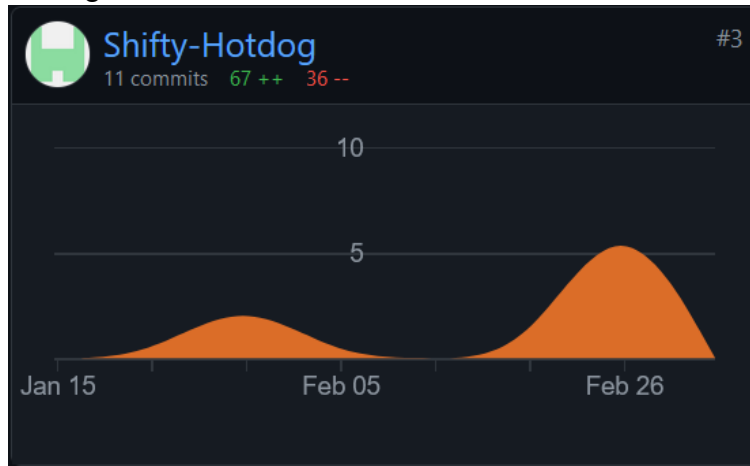
Eduardo Saldana - 6612626

Performed bug fixing, testing and developed additions to user interface.



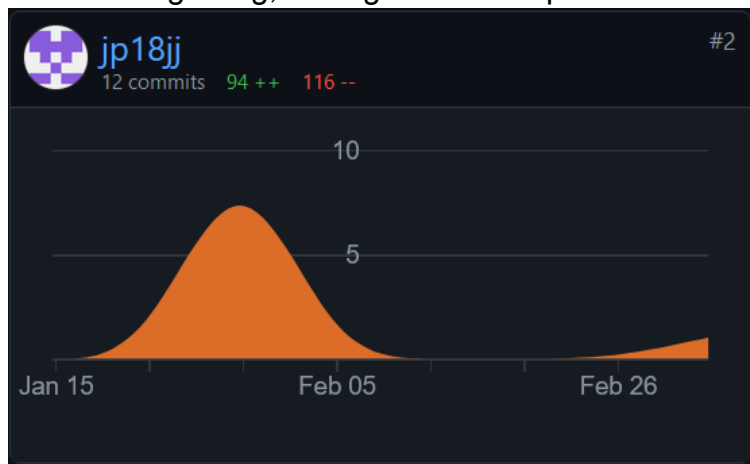
Goktug Cirag - 6776678

Handled bug fixing, testing and developed additions to user interface and csv file management.



Jashan Pannu - 6505861

Handled bug fixing, testing and developed additions to timeline and node structure.



Rishabh Rai - 6847156

Developed the timeline and node structure.

