Interactive Museum Timeline 4P02 Progress Report 2

COSC 4P02
Software Engineering II
April 2, 2023

Instructor: Nasser Ezzati-Jivan
Brock University

Alex Duclos
ad19ua@brocku.ca
6738884

Eduardo Saldana
es18za@brocku.ca
6612626

Goktug Cirag
gc19cy@brocku.ca
6776678

Jashan Pannu
jp18jj@brocku.ca
6505861

Rishabh Rai
rr19pa@brocku.ca
6847156

1. Sprint Review

During this pair of sprints, we intended to finish up the most critical requirements of the system which were incomplete from the previous sprints. This involved finishing major work on both the frontend UI and the timeline structure in the backend. On top of this, we also intended to finish the major subfeatures of the system as per the original plan laid out in January.

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.

A complete implementation of the timeline and nodes has been completed within this sprint.

1.2 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 file management has been implemented and completed during these sprints.

1.3 Digital Timeline Tour

This feature involves a guided and interactive traversal of the timeline for the user which serves as a digital tour. Due to this feature depending greatly on a completed timeline and media system, we decided to make Node Media a prerequisite on top of a finished timeline and node structure.

Considerations were made for this feature for this sprint but we've decided to push its completion to a later sprint when the overall system is near its finalized state.

1.4 Timeline Saving and Loading

This feature is important as a core functionality of the system where the state of a timeline can be edited, saved and loaded as would be expected by admin users.

A temporary solution of saving and loading this information in excel files had been converted to the usage of JSON formatted files. We intend on finalizing the usage of JSON as our save file format across the whole system in later sprints.

1.5 User Interface

With a basic user interface having been completed within the first sprint, we planned on further fleshing out the existing UI with features and adding functionality for other program features as well as creating the possibility of deployment as a web application.

While progress on the overall UI has been made during this sprint, we intend on continuing major work on the UI alongside the implementation of more subfeatures. Solutions for deploying as a web application have been explored and considered.

1.6 Quizzes

The subfeature of having quizzes that can be created by admins and completed by users is an important aspect of the educational system we expect to create.

Major considerations of both backend and frontend representations of this feature have been completed this sprint but no partial attempt has been pushed aside from a skeleton placeholder. We intend on completing the entirety of this feature in a later sprint.

2. Sprint Retrospective: Issues Encountered

Prior to these sprints, we had decided to solve issues encountered in earlier sprints including a lack of overall testing and a lack of considerations for how individual code fit in place with the rest of the system. We've solved both these issues by having peer programming where we were assigned to test each other's code and lower the likelihood of later encountering issues using another's code which puts much greater strain on development efforts and time than bug discovery and fixing during testing.

However, a major issue we encountered in this sprint revolved around our use of peer programming. Due to discrepancies in individual availability, assigning testers for each other's code required greater attention than we initially expected. Waiting prolonged periods to wait for others to finish testing defeated the purpose of testing in order to avoid the hurdles of more complex bug fixing down the development path. Towards the second sprint in this pair, we decided in our meeting to pay more attention to availability and make better attempts at assigning testing tasks. While we remedied the main problem with this, we also realized unexpected changes in availability was a significant issue and we need to be much more flexible and ready to swap testing roles with each other accordingly.

3. Next Sprint Planning

For the next sprints our main goal is to solve the issue of considering availability when assigning roles and tasks in order to drastically improve our efficiency and result output. While our results so far in each set of sprints have not met up with our expectations we are confident that with our discussion and planning from meetings using what we've experienced and learned will help us finish the overall system in these next sprints.

The intention for the next sprint is to carry over incomplete or partially completed features from this sprint over the next and include the subfeatures from the product backlog which have not been added to any sprint backlogs yet. Along with this, we plan on having a final meeting to discuss additional sub features and current feature refinement using what we learn in the following sprint. We hope that in this meeting we'll be able to assess the completion status of our overall system and base new feature generation and feature refinement based on a more tangible and usable system.

Sub features confirmed as focus for the next sprints include the digital timeline tour, the quizzes subfeature, implementing subnode branches, improving on the usability of the interface, creating system client options for admin preferences, and a feedback system for users to provide to admins.

4. Sprint Overview

3.1 Eduardo Saldana Suarez

Implemented the addition of a cyclical navigation feature which allows the user to endlessly cycle from the last element back to the first one when reaching the end of the list and vice versa.

Adjusted the size of the images to ensure there was a sense of consistency across the webpage, this also gives it a more polished look.

Updated the HTML by incorporating semantic elements, this provides better context to the content in the file. Also improves readability and clarifies what the intention of the different parts of the HTML elements are.

The positioning of the light/dark mode toggle button was updated so it could maintain a static position instead of jumping around every time a new image is pulled. This helps the user have a consistent position to look at when wanting to switch between the different modes.

Incorporated the addition of a quiz section to the webpage which is currently a placeholder for when the questions and answers are formulated. The intention is that the questions and answers cycle as the user moves through the images.

3.2 Jashan Pannu

Handled discovery of bugs through extensive testing of the timeline structure and its use by creating a reusable test harness. Also handled solving these bugs and adding additions to the existing structure which would likely be required for later features.

Created the timeline class to serve as the only backend interface the frontend needs to interact with by creating simplified functions for all required functionality. The goal is to map the entirety of the backend through this interface so that a simple timeline object is all that’s needed by the front end team and no understanding of the backend’s inner workings is required.

Explored the usage of various file management solutions to primarily settle on JSON as a file format within the backend to eventually implement auto saving and loading of timeline states.

3.3 Goktug Hizir Cirag

Updated the layout of the website to be more user friendly. The old layout was unintuitive and while it could be understood it was confusing to look at. The new layout is a lot cleaner and uses better natural mapping in order to make the site easier to use at a glance. The website also had user convenience features such as a light/dark mode toggle. Many users like to use dark mode as it is less strain on the eyes and many users use light mode as it has a clean and easy to read style. A toggle was added for users who prefer one over the other. The website also has highlights when a user hovers over a button or textbox. This is meant to give feedback to the user and make the UI a lot easier to understand and use.

3.4 Alex Duclos

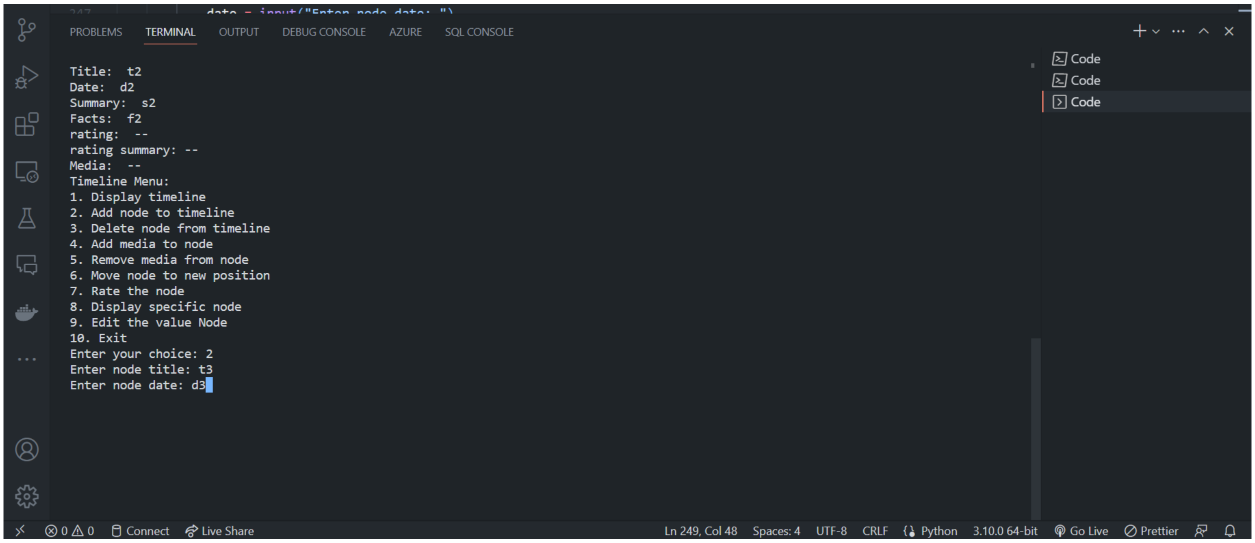
Converted the format of our data from CSV to JSON (Jashan's suggestion) which will allow easier expansion of data, such as additional information, embedded questions, etc. And is a more standardized format which can be easily parsed by any modern programming language. Also converted from Python & Tkinter to a web app using JS which can update without needing a refresh. Also introduced some CSS and the file structure for expansion later. Finally performed some cleanup on the git repo, including adding a .gitignore and fixing some bugs introduced.

Also got the website to work using github pages, for a live demo version.
https://htmlpreview.github.io/?https://github.com/Rishabh9742/COSC4P02/blob/main/Web/index.html

3.5 Rishabh Rai

Made an implement file management which handles all viewable media for each node. Create a function to add or remove media files from nodes. Implemented the system for interacting and traversing the timeline by traveling through nodes. Created input function for previous. Next and receiving a list of all nodes. Added a function for selecting the current node to view its contents. Also added a feature to rate and submit how interesting the user finds the node.

Example output screenshots:



```
date = input("Enter node date: ")
```

```
Title:  t2
Date:  d2
Summary:  s2
Facts:  f2
rating:  --
rating summary: --
Media:  --
Timeline Menu:
1. Display timeline
2. Add node to timeline
3. Delete node from timeline
4. Add media to node
5. Remove media from node
6. Move node to new position
7. Rate the node
8. Display specific node
9. Edit the value Node
10. Exit
Enter your choice: 2
Enter node title: t3
Enter node date: d3
```

5. Team Contributions

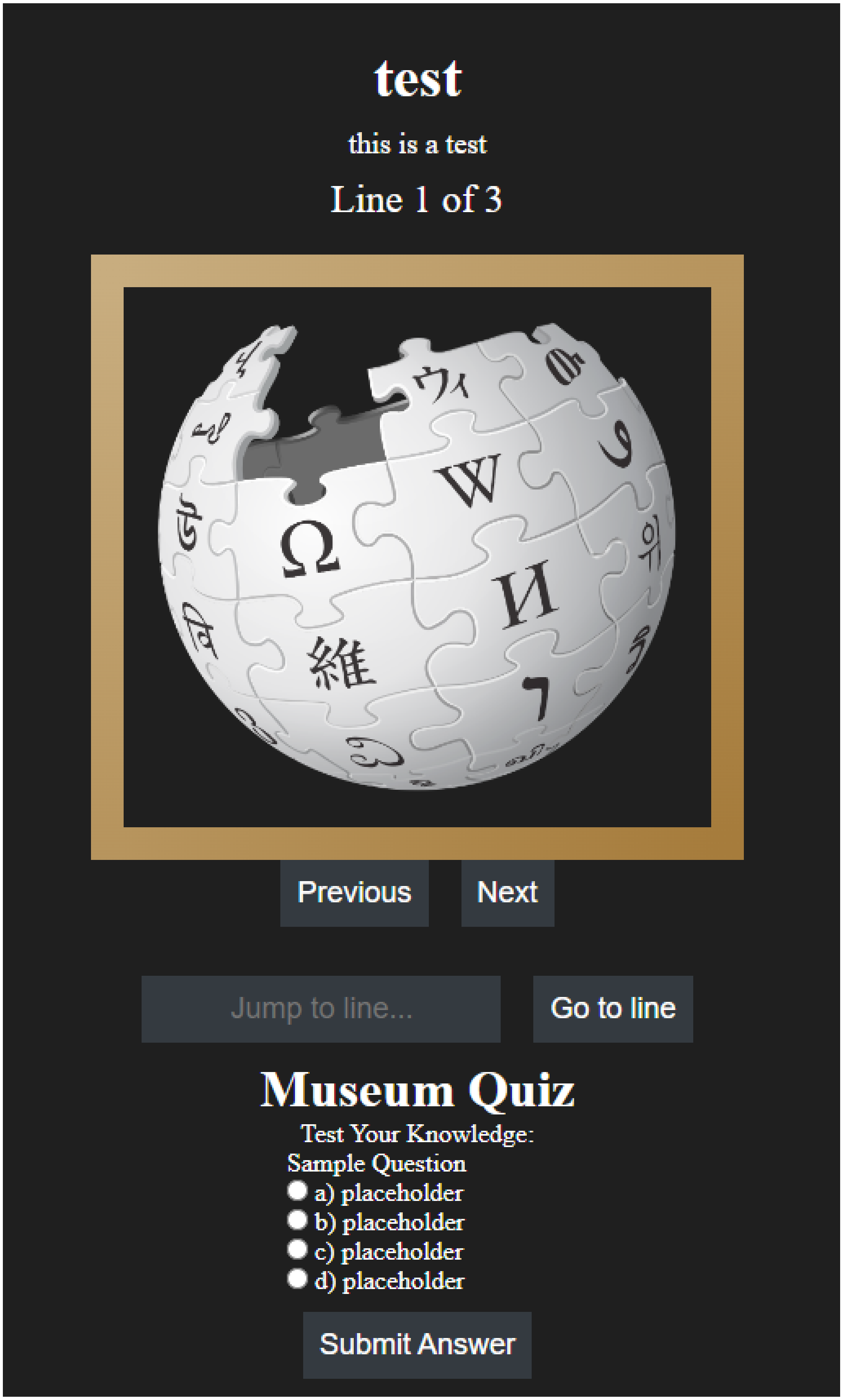
GITHUB LOGS

https://github.com/Rishabh9742/COSC4P02

Alex Duclose - 6738884

Converted the format of our data from CSV to JSON (Jashan's suggestion) which will allow easier expansion of data, and converted to a web app using JS which can update without needing a refresh,

Eduardo Saldana - 6612626
Added a cyclical navigation feature for seamless list traversal, adjusted image sizes for consistency and a polished look, and updated the HTML with semantic elements for improved context and readability. Additionally, the light/dark mode toggle button's position has been made static for user convenience. Finally, a quiz section has been incorporated as a placeholder, intended to offer an engaging and interactive learning experience once the questions and answers are added.



test

this is a test

Line 1 of 3

Museum Quiz

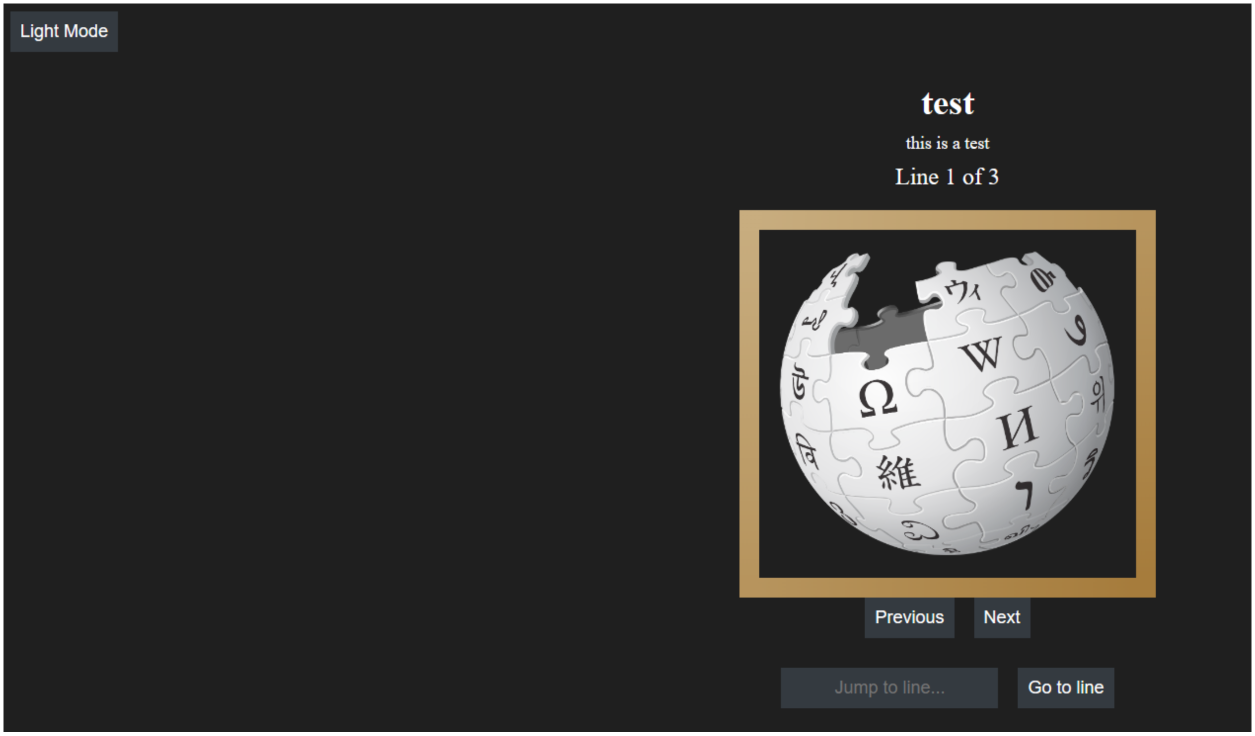
Test Your Knowledge:

Sample Question

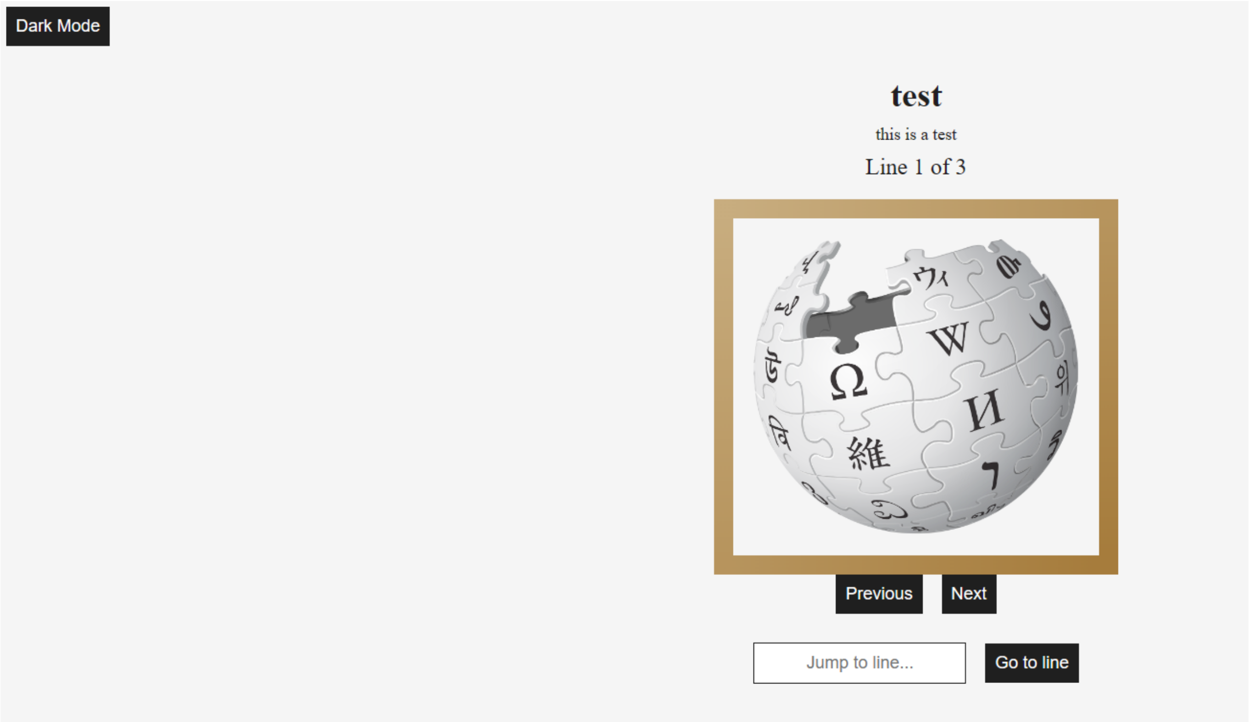
○ a) placeholder
○ b) placeholder
○ c) placeholder
○ d) placeholder

Goktug Cirag - 6776678

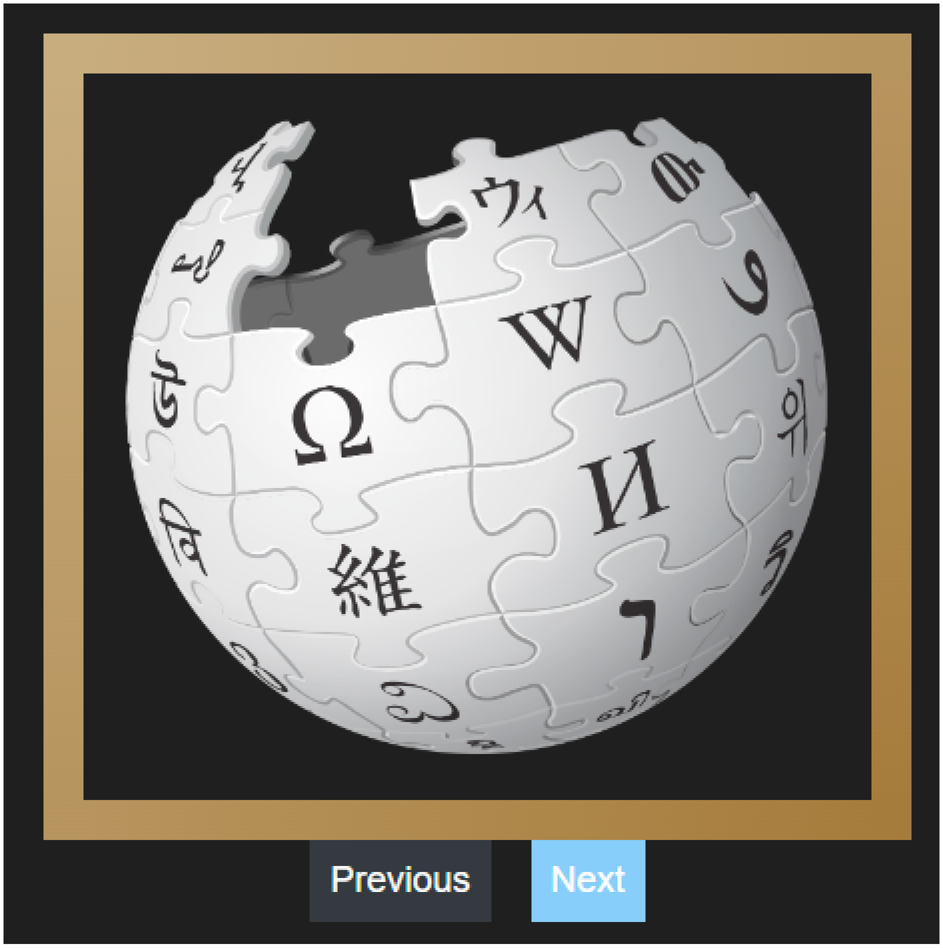
Fixed the layout of the website to be a lot more user friendly and easier to read.



Added a light mode for user convenience



Also added a button highlight



These changes are meant to make a User Friendly experience that is easy to understand. Improving the natural mapping of the site as well as adding features to aid users. Some users can choose light or dark mode to tailor their experience.

Jashan Pannu - 6505861
Performed testing and bug fixing on existing backend code and implemented the main timeline interface in the backend.

Rishabh Rai - 6847156
Made an implement file management which handles all viewable media for each node. Create a function to add or remove media files from nodes. Implemented the system for interacting and traversing the timeline by traveling through nodes. Created input function for previous. Next and receiving a list of all nodes. Added a function for selecting the current node to view its contents. Also added a feature to rate and submit how interesting the user finds the node.