Brock University
COSC 4P02

Software Requirements Specification
Interactive Timeline System

January 28, 2022

Group members:

Yathusan Thulasinathan - 6735955

Athavan Jesunesan (AJ) (Leader) - 6705271

Carl Paladino - 6850796

Yousaf Shah - 6704969

Steve John Abraham Jayaseelan - 6856694

Adil Bedri - 6820344

Fahad Khan - 6688998

Table of Contents:

1 - Introduction

This section will explain the purpose of the document and the overall scope of the project that this document will describe.

1.1 Purpose of the Document

The purpose of the document will be to provide a complete, thorough, and correct list of requirements for this software project. This document will detail the functionality of the system, the non-functional requirements and design diagrams like a sequence diagram, use case diagram etc. The document will also detail the user interface (UI) that will be created for the software project.

1.2 Product Scope

The objective of this project will be to provide accessibility to those who may lack the resources to physically view the museum and provide a more in-depth and personal experience. This will be done using an interactive timeline system that will be used to more fully cover topics that artifacts in the museum are related to. There are requirements that will be stated in later sections of this document that must be included in the project and there may be optional requirements that may be included after all the required features are completed. The final project should be delivered as a web app on the Internet.

Throughout this document, the scope will be checked along multiple different feature specifications to ensure a clear image of what the desired outcome is and to avoid inconsistent understanding across team members. The recommended scope must be respected in order to ensure that the project can be completed in a realistic and reasonable timeline.

2 - Overall Description

2.1 - Product vision

We want our system to be able to provide any potential user of a museum to be able to consume the knowledge of the museum's artifacts on a convenient web browser. The system should relay consistent data with the museum to the user, and the system should always have current data about the most recent artifacts in the museum to ensure 100% of artifacts are listed and show the user some relevant information. Adding information to the database should variably update the user interface, such as any new museum added will be populated, and any artifact linked to a certain museum will show up under that museum with all its information added automatically.

2.2 - Product functions

- Allow user to browse museums
- Allow user to browse museums artifacts
- Allow administrator to modify museums
- Allow administrator to modify artifacts

2.3 - Intended Audience

The intended audience for this system is anybody who would realistically attend a museum and have a mobile device. The nature of this application means that it will mostly be used by people browsing museum artifacts which could be any age group although generally from teenagers to adults. Another user group could consist of visualizing artifacts and their historical chronology, which may be students for history projects (ages 13-23). Thus the structure of the website should look modern for this age group and be efficient with its display of data.

2.4 - Operating Environment

This system is a web application, thus it will be hosted online using web technologies such as HTML, Javascript, JQuery, ASP/PHP, and styling packages such as Django, Bootstrap. The database will be hosted on a SQL server.

2.5 - User Documentation

Features and the interface of the system should be intuitive meaning that all functionality should be straightforward enough to not require a tutorial. Most of our user documentation will consist of meaningful names to direct the user to where they want to go, along with an intuitive interface which allows the user to navigate to where they want to be.

2.6 - Assumptions and Dependencies

- Due to this being a web system that interacts with a database the system requires access to the internet, which fortunately most museums nowadays provide wifi for their customers so this assumption will generally be true.
- Our web hosting will rely on Github page hosting to distribute our website.
- We depend on accurate information supplied from museums about their artifacts to allow us to model it in our system and display it all to the user.

3 - Functional Requirements

3.1 - Backlog Items

- 3.1.1 - Design the database architecture
- 3.1.2 - Ability for admins to add elements into the timeline
- 3.1.3 - Ability for the user to click on elements on the timeline to get more information
- 3.1.4 - Create a login/registration system for admins
- 3.1.5 - Embed videos into the information section
- 3.1.6 - Be able to zoom in and out for the timeline
- 3.1.7 - Add searching functionality
- 3.1.8 - After you click on one of the elements of the timeline, there should be back and forward button to browse over the timeline seamlessly
- 3.1.9 - Design themes for the website
- 3.1.10 - Add a location to be able to view artifacts in chronological order
- 3.1.11 - Make a queried artifact interactable to be able to see more information for it

3.2 - Prioritization order

- High priority items - Items that must be completed before being able to implement some correlated functionality
  3.1.1 - We need to design the database architecture in order to have a table structure that permits us to use the database in a way for the system structure to actually implement it's functionality
  3.1.4 - Must be able to create administrators to test the administrator functionality and develop functions for them
  3.1.9 - Must design the website structure and create physical functionality (buttons, forms, etc) to allow user interaction

- Medium priority items - Items that are critical to functionality but are part of the main functionality of the website such as implementing features for the user to interact with and populating the database.
  3.1.2 - Admins must be able to add items to the timeline. They will be able to access a add element page where they add a title, the year, a description and any multimedia elements
  3.1.3 - Ability for the user to click on elements on the timeline to get more information
  3.1.7 - The users should be able to search for any elements/artifacts they want to find more information about
  3.1.10 - Add a location to be able to view artifacts in chronological order

- Low priority items - These items augment the system and allow it to come together as a whole.

3.1.5 - Embed videos into the information section
3.1.6 - Be able to zoom in and out for the timeline
3.1.8 - After you click on one of the elements of the timeline, there should be back and forward button to browse over the timeline seamlessly
3.1.11 - Make a queried artifact interactable to be able to see more information for it

4 - Non-Functional Requirements

4.1 Performance

- System should be responsive within a justifiable amount of time (Processing time shouldn't take longer than 3 seconds for any query)

- Any requests that should not be executed will return an error to the user to demonstrate that their input command was invalid

4.2 Security

- Should ensure the database is secure and not susceptible to things such as SQL injections, or security breaches. This means ensuring the database information is secure and any database command that will be queried should be sanitized and filter out any malicious requests. Only admins should be allowed to access the administrator panel and a user should not be allowed to access it.

4.3 Supportability

- The system should be able to support an amount of people that might realistically attend a museum at any given time, this project we could assume maybe 300 concurrent users browsing and requesting data, along with 2 administrators of the system able to interact with the system.

- The system should be scalable, allowing us to add more museums, and artifacts as they come to exist. This also includes database size being able to support at least 15 different museums, with each museum containing 20 unique artifacts.

4.4 Reliability

- The webpage/app should have an initial help section for quick and easy navigation
- The downtime for software updates or fixing errors should be informed prior and should take the shortest time possible.

4.5 Usability

- The system has a good user interface design (support grouping related functions, good color schemes, appropriate located functions)

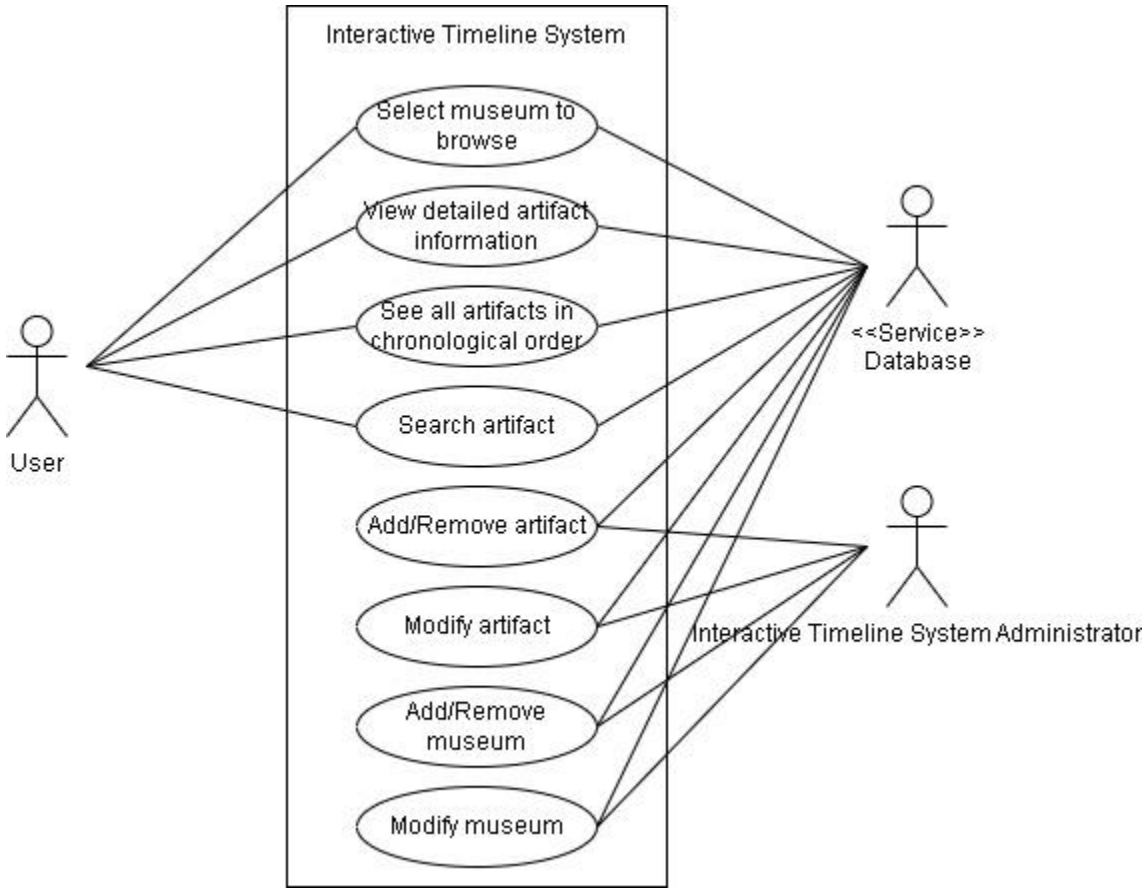
- The system should support the major web browsers (Any chromium browser, Firefox), Internet explorer is deprecated thus not requiring support.

- The system must ensure that only authorized administrators are the only people with access to make changes to museum information and or add/remove artifacts to database collection.

- Certain pages can be given the option to be localized.

5 - Use Case Diagram

5.1 Diagram



5.2 Explanation of Diagram

The user's purpose will be to strictly query relevant data from the database backend, the user's role will be to strictly view information and data.

The interactive timeline system administrator will be a user with elevated privileges where they are granted the access to be able to make changes to the data contained in the database, such as modifying artifacts and which museum they reside in, and modifying the museum information (location, open times, etc.).

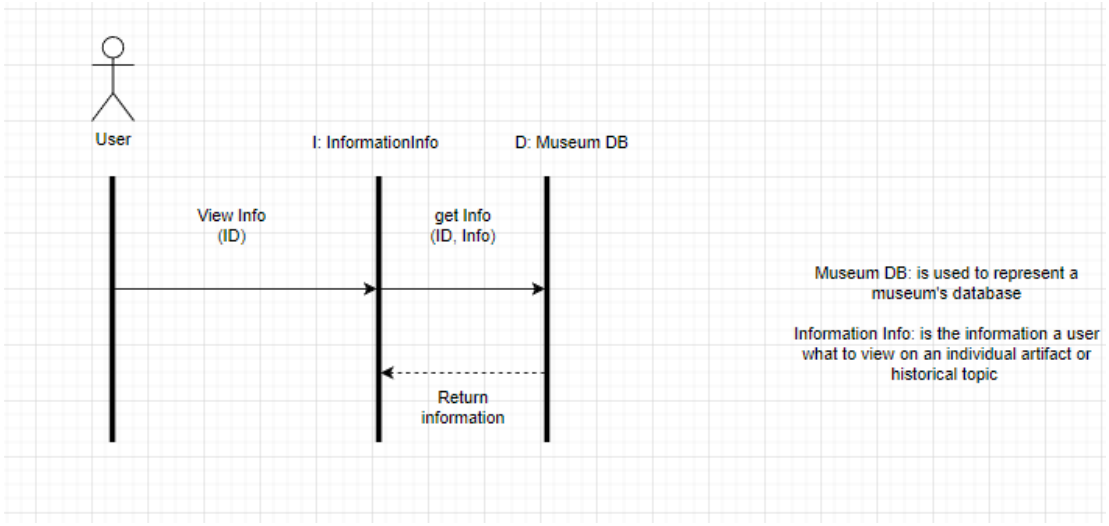
The database acts as a database, which responds to SQL queries such as select, update, insert, drop records within the tables.

The user is not authenticated unless the elevated user enters the administrative portal since the main functionality of this system is to relay any and all information to a general user for any museum due to the information contained in the database being intended as public information for ease of access to users.

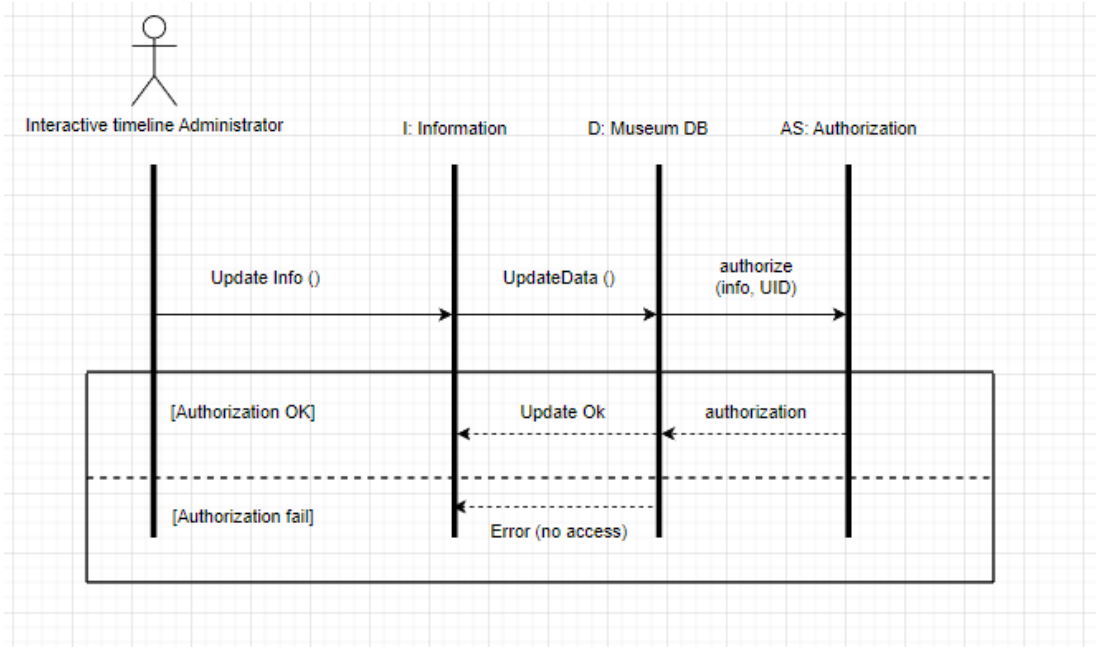
6 - Sequence Diagram

6.1 - Diagram

Sequence Diagram for User:



Sequence Diagram for Administrator:



6.2 - Explanation of Diagram

The purpose of these sequence diagrams is to help visualize and document the dynamic interactions between the various system components and the users of the interactive timeline system. This system includes users (visitors and museum staff) and system administrators. The diagrams also explain many actions performed by each actor and the flow of information between them.

The sequence diagram of User simply demonstrates how a user is able to view and retrieve artifact information. Once the user views and selects the information of desire, that information is retrieved from the museum's database for the user to obtain. Once the user has finished gaining the information, it will then be returned to the museum's database where it belongs.

The sequence diagram of the Interactive timeline Administrator demonstrates how an administrator accesses the back-end of the interactive timeline system. For the administrator to access the back-end, they must authorize themselves by logging in with a username and password. Once authorization is successful, the administrator is able to update any information (add, delete, edit). If changes have been made and the information authorization is successful, the information will then be uploaded onto the front-end of the interactive timeline system for users to view and interact.

7 - System Architecture Diagram

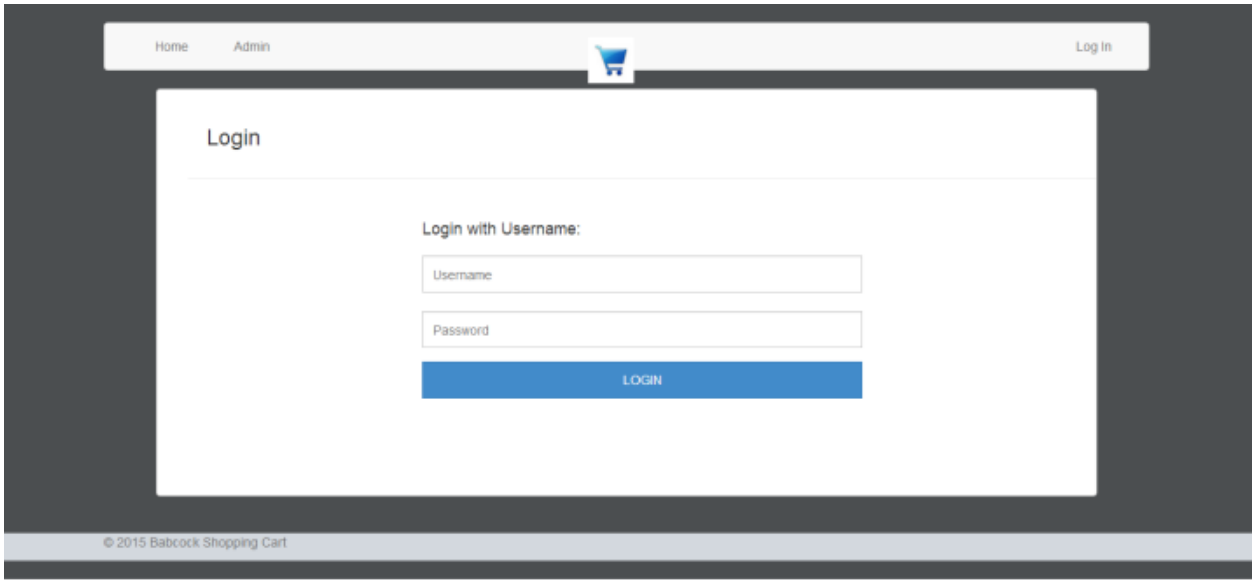
7. 1 - Diagram

7.2 - Explanation of Diagram

The **First Layer** of the System architecture is the Web Browser. This is the end goal and will be how the user interacts with the app.
The **Second Layer** is comprised of validation and administration. These crucial systems have the responsibility of making sure the appropriate person can access/edit what is broadcasted to the Web Browser.
The **Third Layer** of the system works in tandem with the second layer for the security but is handled elsewhere to promote security of the overall system.
Finally, the **Fourth Layer** is the request and calls for information from the database. This layer allows for the appropriate items to be sent to the administrator when changes are being made as well as, ensure the correct information is displayed on the Web Browser.

8 - UI Design

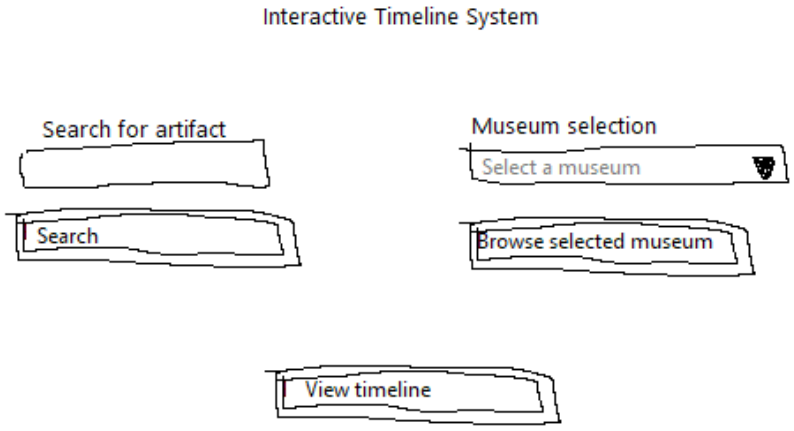
8.1 - Administrator Login portal



Source:
https://www.researchgate.net/publication/305426451/figure/fig3/AS:385593465950209@1468944054102/Administrator-Login-page-The-Administrator-Login-page-in-figure-7-provides-the.png

There is a separate page for the administrator to login to which doesn't complicate any of the general user portion of the website and allows the administrators a simple interface to be able to perform their tasks.

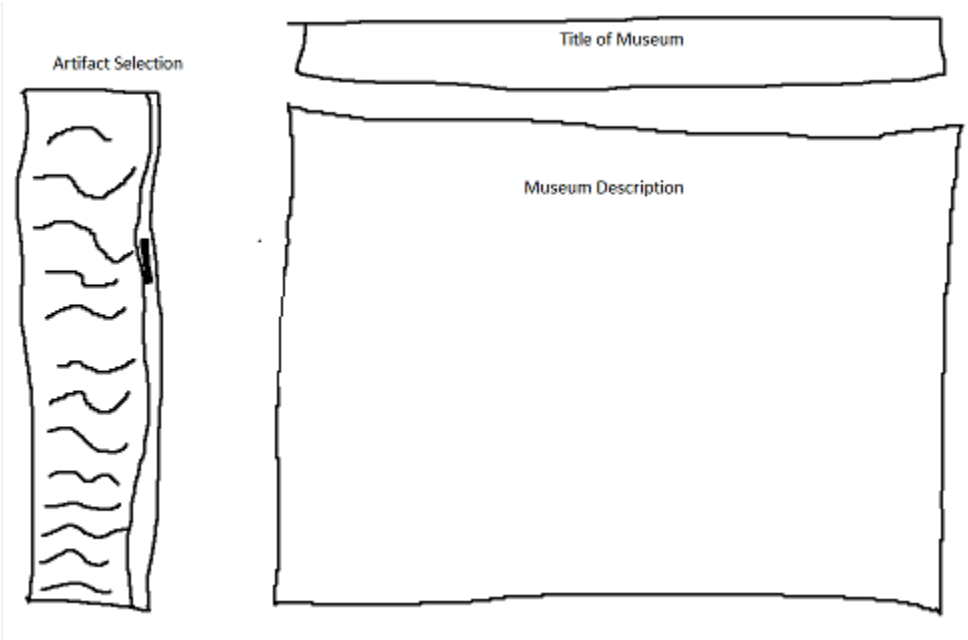
8.2 - User landing page



The user's landing page consists of:

- A search box where they can search for a specific artifact which then shows up possible options for matching artifacts to the searched term.
- Another input is a museum dropdown where the user can filter by a specific museum, then they click the "browse selected museum" button and it takes them to the museum landing page (8.3).
- The final user option on the landing page is the "view timeline" button where the user can view the artifacts in chronological order.

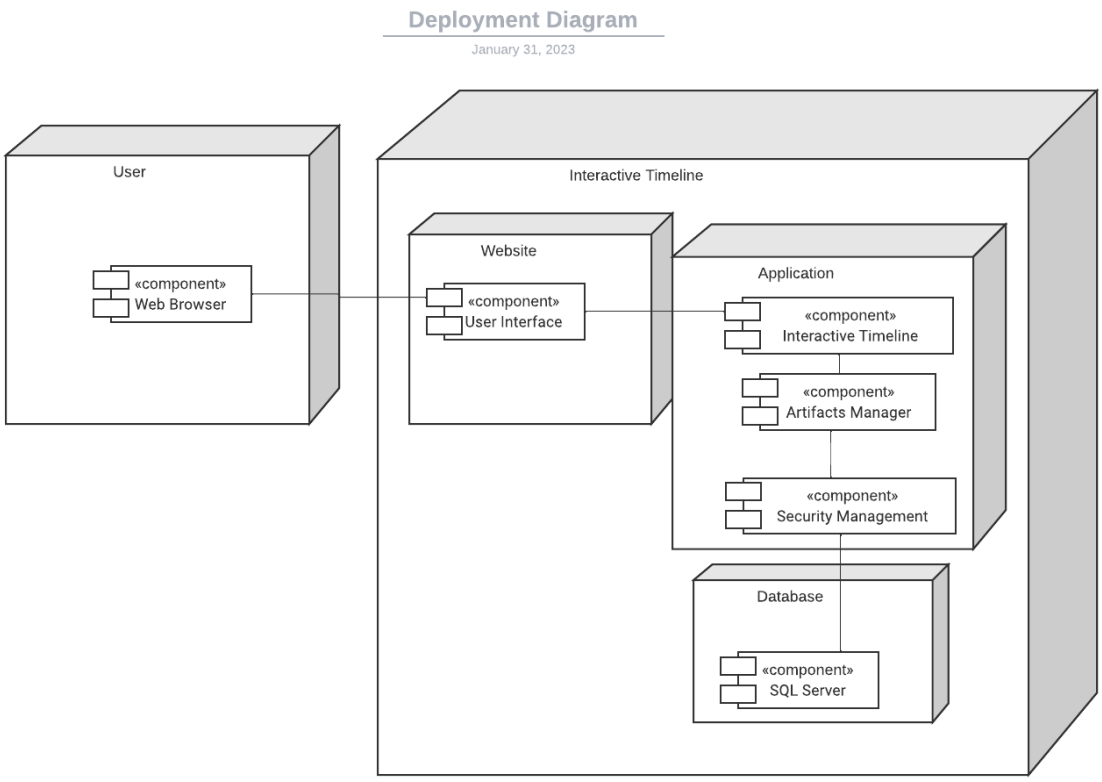
8.3 - Museum Landing Page



The museum landing page is populated with all the artifacts pertaining to the filtered museum with the default screen being information of the museum itself and all the artifacts located on the left side where the user can scroll and click any one which then populates the “museum description” box with information of that artifact such as images, description, any relevant links and other data.

9 - Deployment Diagram

9.1 - Diagram



9.2 - Explanation of Diagram

The purpose of this deployment diagram is to show the structure of the overall system that our website application will follow. It consists of the following nodes: “User”, “Website” “Application”, “Database”. Each of the nodes contains components that allow them to run properly. The user will access the Website using a Web Browser using the internet. The website will have access to the application which contains the interactive timeline itself. This also contains components for the admins to use themselves like the artifacts manager and updating the interactive timeline. The security management is also a part of the application. The application will have access to an external database.