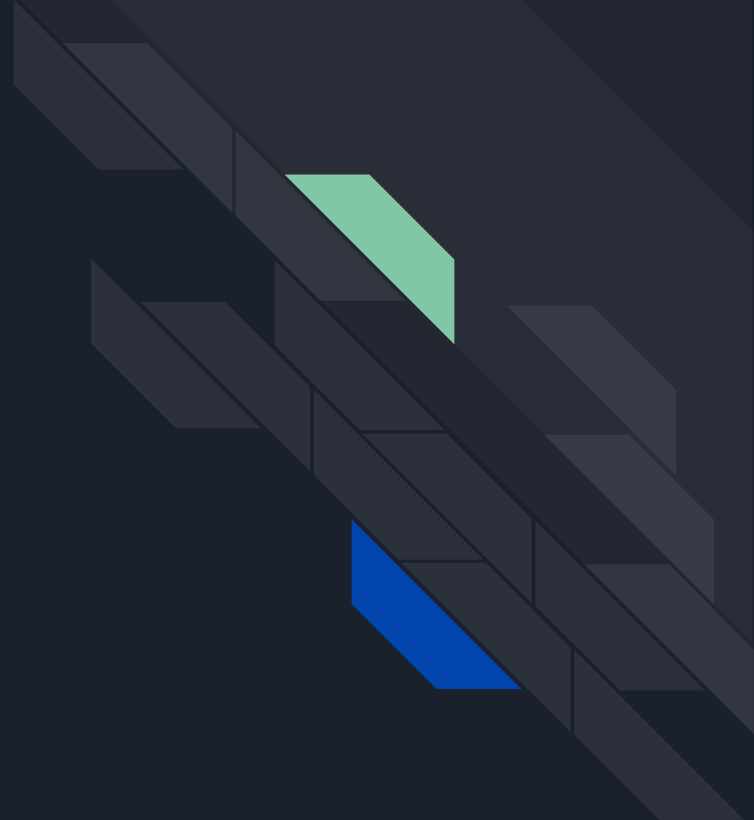# COSC 4P02 - Project Interactive Map

William McCormick - (6652762) - Scrum Master
Riely Benson - (6360507) - Product Owner
Kevin Olenic - (6814974) - Development Team
Salman Saeed - (6479018)- Development Team
Connor Vrieswyk - (6669345) - Development Team

Overview

The project chosen for development was a fully functional interactive museum mapping application capable of providing all users, regardless of their level of technical skill, extensive information and resources for their visit.

Along with providing important information, such as exhibit descriptions/locations, museum information, and locations of interest for visitors, this app allows museum personnel to personalize their museum's map layout/functionalities.

By providing an intuitive and user-friendly interface, our interactive mapping application can help museum personnel streamline their work while providing an enhanced experience for visitors.

Demo

We will now take the time to run through a demonstration of our application and its functionalities.
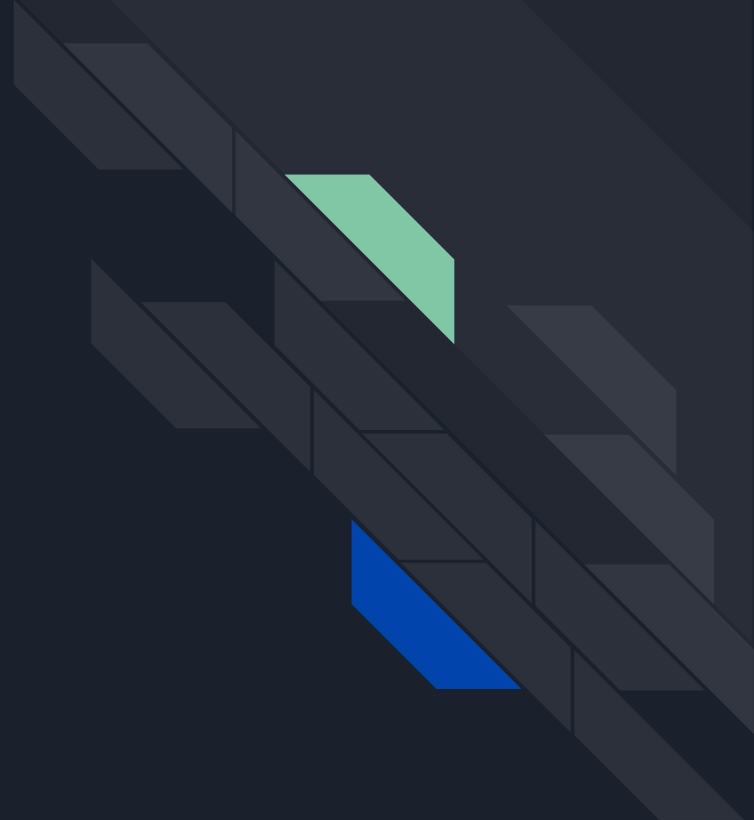
Key Decisions

- Scrum Process

- Software License

- Tools

- Mapping Interface

- Hosting

Scrum Process / Progress Monitoring

- Divided up into roles (Website & App)
- Sprint Backlog defined early (user stories, key wants/needs, etc.) using ScrumDesk
- Biweekly meetings (Weekly for the final month)
- Daily Scrum Messages (mostly Kevin)

- GitHub, Discord, Scrumdesk to track
- Extra progress updates/side meetings for updates/questions/assistance

Sprint Backlog/User Stories

Title	Actors	Description	Data	Pre-condition	Post-Condition	Steps	Exception Flows
Create Map	Admin, Individual chosen by museum to upload Map	An employee or admin may wish to load a new museum to the program	Museum layout, exhibit information, images showing museum or exhibits	User-command issued by individual to load map data	Confirmation that map has been saved to database and ability to access map data on main page	1.User logs into system 2.User navigates to Map uploading function on system 3. User fills map data 4.reviews map and makes edits if necessary 5.User accepts map	May try to set name of map to one that already exists
Edit/update Map	Admins, individual chosen by museum to update map info	An employee or admin may wish to update or edit the info in pre-existing map	Museum layout, exhibit info, images	User-command issued by individual to edit/update map info	Confirmation that the museums map info has been updated	1.User logs into system 2. User navigates to map editing function 4. User makes changes to map 5.User reviews changes to map 6.User accepts updated map	Map not updated successfully
Delete Map	Admins, authorized individuals	An individual may wish to delete an existing map	Museum being deleted	User-command issued by individual to delete amp	Confirmation map has been deleted	1.User logs into system 2.User navigates to map delete function 3.User clicks button to delete map 4.Map is deleted	Map was not deleted successfully
View Map	Admin, user, museum employees	An individual may wish to view a map that has been uploaded to the system	Museum being viewed	User-command to view designated museum	Show museum wishing to be viewed	1.Individual access the system (could be anyone) 2.User navigates to map they wish to view 3.User clicks on button to view designated map 4.Designated map is shown	Map is unavailable, Error loading map

Add Image	Admin, employees	An individual may wish to upload an image to be viewed with the map	Image being added, map image is being added tp	User-command to select an Image	Alert user image has been added	1.User selected an image 2.They are shown the image and validate it is correct 3.They upload the image and add it to the map	Did not select a compatible image, image did not load successfully, image was not retrieved properly
Remove Image	Admin, employee	An individual may wish to remove an image from a map	Image being removed, map image is being removed from	User-command to remove image	Alert user that image has been removed from map	1.User selects image being removed	Removal failed

BackLog:
- Add Map
  - Enter: Museum name & Number of floors
  - Enter: Floor layout
  - Enter: Floor/exhibit Info
  - Upload: Floor/exhibit Images
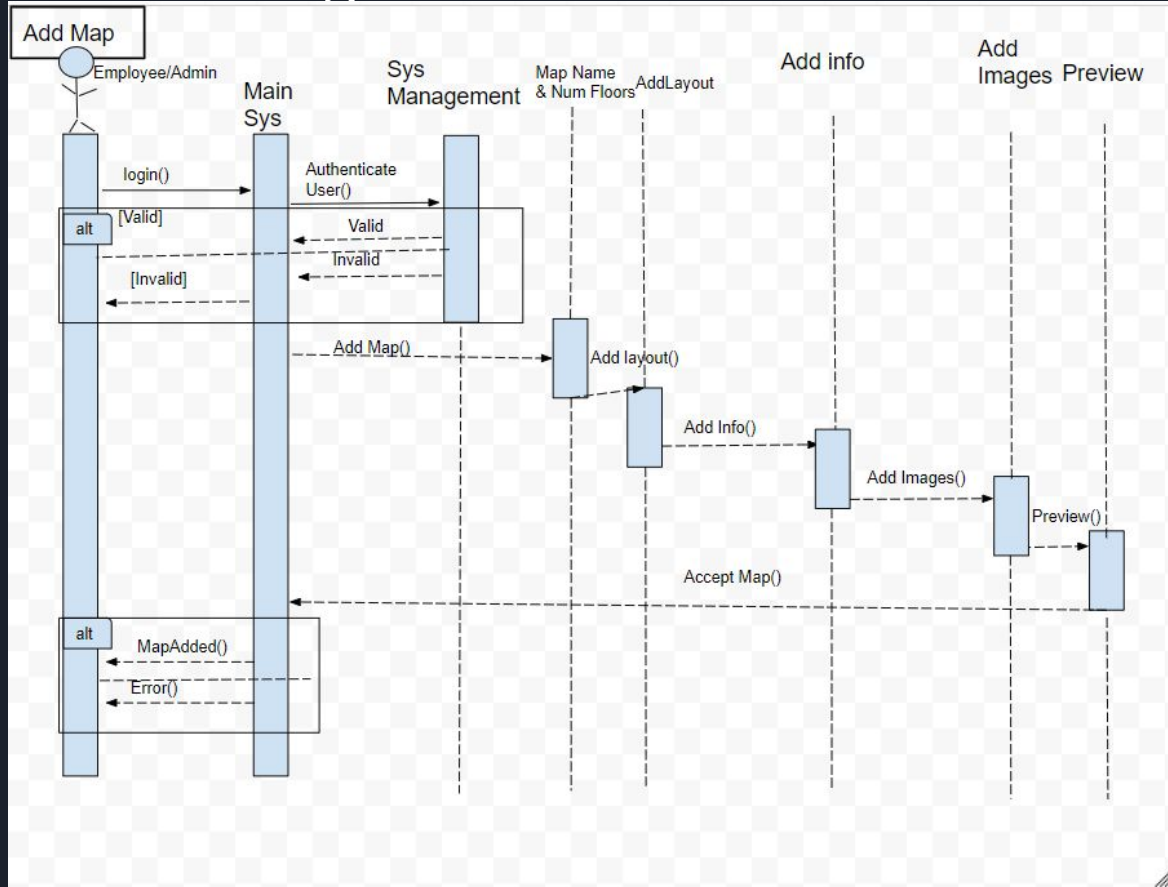 - Preview Map
- Edit Map
  - Edit layout
  - Edit Info
  - Edit Images (add/remove)
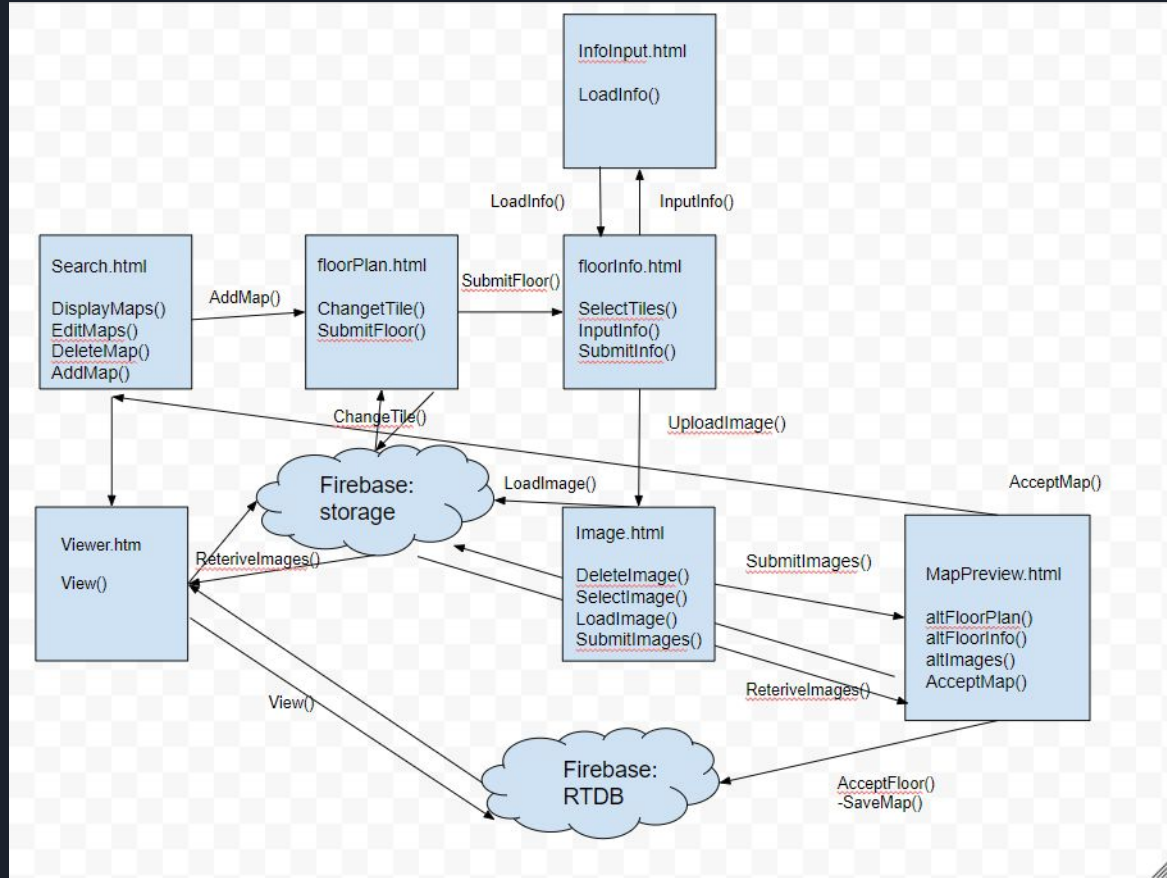 - Preview Changes
- Implement DB functionality
  - Store Museum data
  - Store Image data

- Delete Map
- View Map
- Login/logout

Sequence Diagrams

Interactions Between Components

Software License

-BSD 2-Clause "Simplified" License

- Permissive, no copyleft provision, flexible, compatible, minimal legal risk, offers access to Brock University

Tools

-Teams

-RJ TextEd

-Discord

-Android Studio

-Figma

-Firebase

-Github Pages

-ScrumDesk

Mapping Interface

Hosting

Initial Plan

-   **Upload blueprint/base museum layout to image map and hotspot interactive pieces**

Decision

-   **Utilizing TileMap, we could create museum layouts by grouping graphical tiles together, and allow for more efficient memory management + faster loading times**
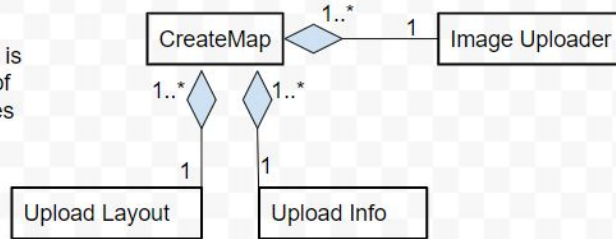
-   **Com**bination of Github Pages and Firebase
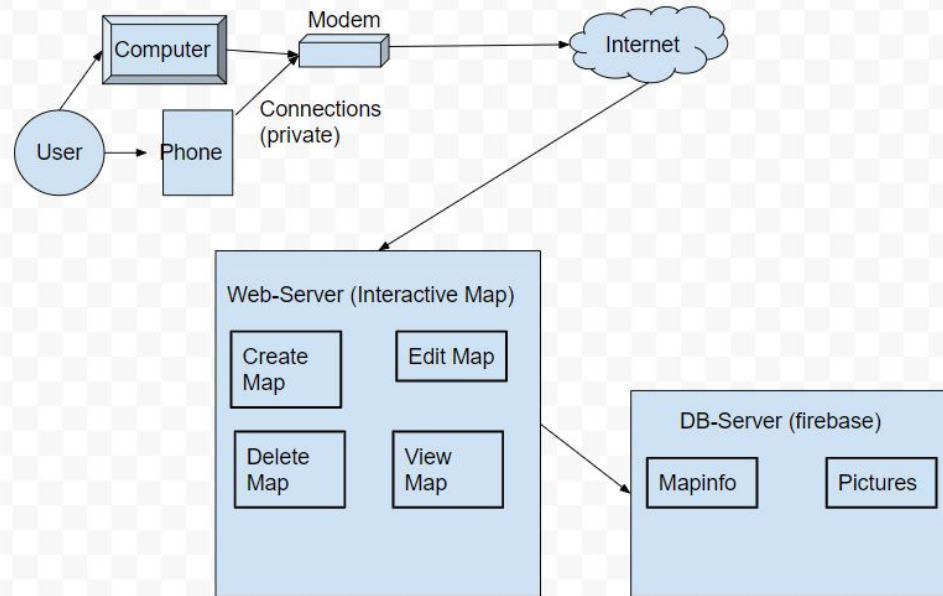
Relationships
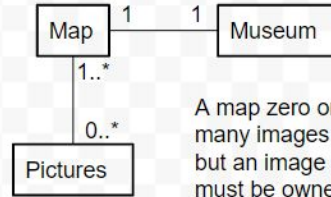
Physical Deployment

Testing

- Unit Testing
- Integration Testing
- Functional Testing
- Regression Testing
- Random Testing

Debugging Techniques

- Statistical Debugging
  - Collecting Execution Data (random/preset cases)
  - Analyze Execution Data
  - Prioritize the potential sources of the bugs
- Print Statements/Logging Statements
- Built-in debuggers
- Code refactoring

Challenges

Foreseen

- Data Accuracy
- Optimal UI
- Compatibility
- Time Management

Unforeseen

- Technical Issues
- Maintenance
- Communication
- Effective Collaborative Work

Business Value/Roadmap

- Application Value
    - Museums and Customers
- Scalability
- ROI
- Brand Image
- Data Insights
- Cost Savings
- Future Plans for Deployment