

CSE416

Final Document

<MAPBOOK>

Team black

Yun Ah Kim, Ratanak Rin (Jass), Samuel Han, Haneul Lee

Table of content

I. System Analysis.....	3
II. Use Case Model.....	6
III. User Interface Model.....	39
IV. Requirement Presentation(Attached to the end of this PDF).....	40
V. Data Model.....	41
VI. Software Model.....	48
VII. Code Presentation(Attached to the end of this PDF).....	52
VIII. Meeting Log.....	52
IX. Post Mortem (Next Page).....	67

I. System Analysis

Executive Summary: Revolutionizing Mapping with MapBook

1. Overview:

MapBook is an innovative map editing application designed to transform the way individuals and communities approach map services, including map displaying, customizing, managing, uploading, exporting, and collaborating (forking and real-time editing).

2. The Need:

In a rapidly evolving world, where geographical information is the basis of decision making, there exists a pressing need for an innovative map editing web application. It is essential to manage geospatial information effectively, and the ability to customize maps to specific needs, and engaging communities for the map enthusiast is required. MapBook empowers users to contribute to map accuracy and fosters collaboration in various map types.

3. Existing Solutions vs. MapBook

While there are existing mapping tools in the industry, they often fall short in terms of friendly usability (simplicity), feasibility (design/UI), collaboration and integration. The following are some of the mapping software that exist in the present marketplace.

- QGIS: Steep learning curve, interface customization, lack of community support
- Mapbox Studio: Complex for beginners, rendering speed and pricey service
- OpenStreetMap (OSM) Editor: Complex for beginners, limited offline editing

Objective

MapBook aspires to offer a user-friendly interface for creating, customizing and managing geographical data. It aims to support collaborative editing, various files compatibility, data import/export, and customization of map styles. Version control feature is specially added to facilitate tracking of changes as well as the real-time collaboration feature.

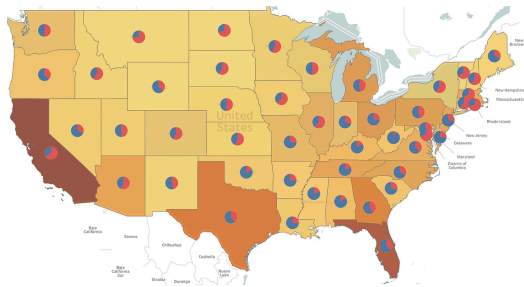
Strategies/Philosophies

We plan to implement agile development. We believe the utmost focus is to make the web application easy to use and easy to see, therefore we also plan to approach the creation of our application by making it user-focused by implementing an efficient user interface. We will accomplish this by testing our product consistently in order for us to secure an efficient and effective UI/UX.

We will accomplish this by following specific orders: first we will research the market and figure out which products include features that would accomplish our goals. Then, we will find and plan out what features we would like to implement in addition to the basic required features. Then we will implement the product. Finally, we will consistently edit and revise our code to make appropriate corrections.

Constraints

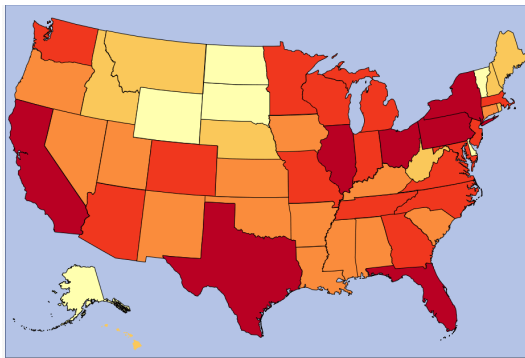
We will help users express various data through the MapBook application. Below is the map graphic type we provide.



1. Pie chart map

→ Good for representing various comparable data.

ex) Comparing the number of users of IOS vs Android.

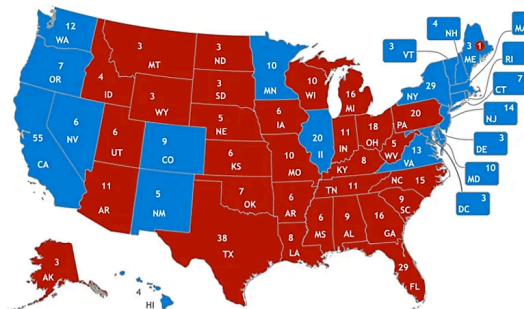


2. Heat map

→ Suitable for representing data with figures.

→ Possible to display positive/negative indicators to users through color.

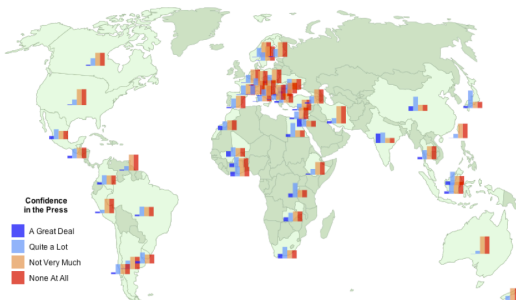
ex) Average annual temperature (Will help users understand the data at a glance)



3. Thematic map

→ Thematic map makes it easy to see conflicting data at a glance.

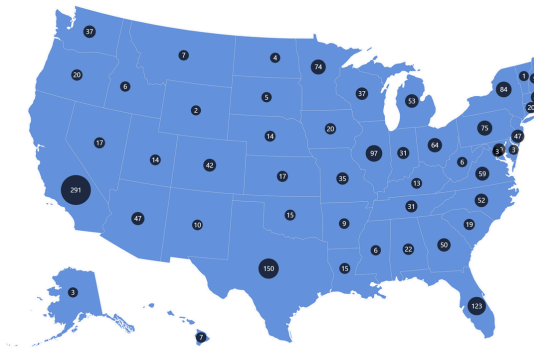
ex) The approval rating data of Republicans and Democrats in the United States.



4. Bar chart map

→ Easy to compare data, similar to a pie chart map.

→ However, unlike pie charts, it can represent different data and provide a clearer picture of numerical differences between the data.



5. Circle chart map

→ The circle chart map allows visual representation of figures through the size of the circle.

ex) The number of people in each state in the United States. (users will be able to see the data at a glance.)

Actors

1. Guest user

- a. Can only search and view maps

2. Basic user

- a. Can search/view and comment on others' maps
- b. Create/edit maps

3. Admin

- a. Can do everything a basic user can do
- b. Can view all accounts and delete them
- c. Can view all posts and delete them

Services

1. Account Management

- a. User registration and authentication
- b. Admin account management
- c. Password recovery

2. Map Graphic **Creation/Editing/Viewing**(Upload : SHP/DBF, GeoJSON, KML files)

- a. Create map from existing templates or fork other users map
- b. Viewing and Navigation - interactive map display + zooming and panning
- c. Edit map geometry, add custom map properties, customize map properties(labels, colors, legends), undo redo functionalities

3. Map Graphics **Exporting**

- a. Export map as PNG JPG and JSON formats

4. Map **Classification** and **Search**

- a. Add tags to map with keywords and categories(topics) for searchability
- b. Search filters should be applied for finding specific map graphics

5. **Community Interaction**

- a. User discussion comments and feedback on maps
- b. Collaboration on map graphics

II. Use Case Model

1. Use Case Listing

Use Case #	UI Context	Use Case Name
1.1	Landing Page	Get started in Intro
2.1	Login Page	Login Information Form
2.2	Login Page	Forgot Password button
2.3	Login Page	Continue as a Guest button
2.4	Login Page	Go to Registration button
3.1	Get started pop-up	Go to user registration
3.2	Get started pop-up	Continue as guest
4.1	Registration Page	User Registration
5.1	Mapbook Main Page	App Logo OnClick
5.2	Mapbook Main Page	Add Maps
5.3	Mapbook Main Page	Go to My Maps
5.4	Mapbook Main Page	Go to Social
5.5	Mapbook Main Page	Go to User Profile
5.6	Mapbook Main Page	View Map Detail
5.7	Mapbook Main Page	Open Options (Fork maps,Share maps,Export maps)
5.8	Mapbook Main Page	Map Search
6.1	Map details Page	Map Navigation
6.2	Map details Page - user	Open options: Viewing others' map (Fork maps,Share maps,Export maps)
6.3	Map details Page - user	Comment (Write new comment, reply)

7.1	My Map Details Page	Navigate map
7.2	My Map Details Page	Share map
7.3	My Map Details Page	Export map
8.1	Edit Map Page	Edit maps
8.2	Edit Map Page	Edit maps' details
8.3	Edit Map Page	Edit map visibility
8.4	Edit Map Page	Redo
8.5	Edit Map Page	Undo
8.6	Edit Map Page	Save changes
8.7	Edit Map Page	Delete map
9.1	Create New Map Page	Create a new map
9.2	Create New Map Page	Add data to each region of a map
9.3	Create New Map Page	Import map
10.1	Social Page	Search for topic
10.2	Social Page	View discussion posts
10.3	Discussion Post Page - user	Comment on discussion posts
10.4	Social Page	Go to next/previous pages of the social page
11.1	User Profile Page - user	Edit User Information
11.2	User Profile Page - user	Edit User Profile Picture
11.3	User Profile Page - user	Logout
11.4	User Profile Page - user	Remove account
12.1	My Maps Page - user	Create more maps
12.2	My Maps Page - user	Delete created maps
12.3	My Maps Page - user	View created maps

13	Admin Home Page	Manage as Admin
14	Manage Menus Page	Choose Category
15	Manage Maps Page	Manage maps
16	Manage Users Page	Manage users
17	Manage Discussion Page	Manage discussion posts
18	Password Recovery Page	Password recovery

2. Use Case Descriptions

Note) User refers to "logged in user"/ Guest refers otherwise

CASE 1

Number	1.1
Name	Get started in Intro
Actors	Guest, User, Admin
Story	<ol style="list-style-type: none"> 1. Actors(User, Guest, Admin) will scroll the landing page for the available features/content part of the web app. 2. On the end of the screen, will be the GetStarted button 3. Left Click on the button -> GetStarted dialogue will popup
Scenario	N/A
Exceptions	N/A

CASE 2

Number	2.1
Name	Login Information Form
Actors	User, Admin

Story	<ol style="list-style-type: none"> 1. On load, the user will be prompted to fill in their username and password to login. 2. Once the user enters proper information, they will click on the 'login' button to login and be redirected to the 'Main page'.
Scenario	Username: user0 Password: password123
Exceptions	Entering information that does not exist for either or both username or password will cause an error message to show that says 'incorrect username or password'.

Number	2.2
Name	Forgot Password button
Actors	User, Admin
Story	<ol style="list-style-type: none"> 1. Left click on 'forgot password' button to go to the 'password recovery page'
Scenario	N/A
Exceptions	N/A

Number	2.3
Name	Continue as a Guest button
Actors	Guest (anyone accessing the application)
Story	<ol style="list-style-type: none"> 1. Left click on the 'continue as a guest' button to skip logging in and continue as a guest user.
Scenario	N/A

Exceptions	N/A
-------------------	-----

Number	2.4
Name	Go to Registration button
Actors	Guest
Story	1. Left click on 'register' to go to the 'registration page'.
Scenario	N/A
Exceptions	N/A

CASE 3

Number	3.1
Name	Go to user registration
Actors	User, Guest, Admin
Story	1. Left Click on 'register' to go to new account registration page
Scenario	N/A
Exceptions	N/A

Number	3.2
Name	Continue as guest
Actors	User, Guest, Admin
Story	1. Left Click on 'continue as guest' to go to main page

Scenario	N/A
Exceptions	N/A

CASE 4

Number	4.1
Name	User registration
Actors	User, Guests, Admin
Story	<ol style="list-style-type: none"> 1. Enter a username 2. Enter an email address 3. Enter a password, where password should be at least 8 characters long, including both letters and numbers, and at least one special character 4. Enter a confirmed password 5. Upload a user profile picture by clicking the “Upload” button, select an image from the user’s local drive and click the “OK” button. 6. Click “Sign up” button 7. After successfully signing up, it will take users to the login page 8. If a user already has an account, they can click on “Already have an account? Sign in” button or link and go to the login page
Scenario	<ol style="list-style-type: none"> 1. User launches the MapBook application and lands on the app landing page. 2. On the landing page, the user taps the “Get Started” button and a modal pops up. 3. On the pop up modal, the user taps the “Sign up” button. 4. User is then directed to the registration page and enters the following information:

	<ol style="list-style-type: none"> a. Username: Simply Underlying b. Email address: simply.underlying@gmail.com c. Password: ***** d. Confirm password: ***** e. Profile picture: [Upload an image] <ol style="list-style-type: none"> 5. User clicks on the “Sign up” button on the registration page to submit the registration. 6. The server validates the email format, checks if the provided email is unique and ensures the password meets security requirements. 7. If all data is valid, the server creates a new user account. 8. The user then receives a verification email. 9. The user will be redirected to the login page with a successful message.
Exceptions	<ol style="list-style-type: none"> 1. If the email address provided is already registered, display an error message: “Email address is already in use. Please use a different email or proceed to login.” 2. If the email format is incorrect (e.g., missing “@” symbol), display an error message: “Invalid email format. Please enter a valid email address.” 3. If the password does not meet security requirements (e.g., too short), display an error message with password requirements: “Password must contain at least 8 characters long and contain a combination of letters, numbers, and special characters.” 4. If the “Password” and the “Confirm Password” do not match, display an error message: “Passwords do not match. Please ensure both fields have the same password.” 5. If a general issue occurs during the registration process (e.g., database connection issue), display an error message: “Registration failed due to a generic error. Please try again later.”

	6. If the uploaded profile picture exceeds a certain size limit, display an error message: “Profile picture size exceeds the limit. Please upload a smaller size image.”
--	--

CASE 5

Number	5.1
Name	App Logo OnClick
Actors	Guest, User, Admin
Story	1. Left Click on AppLogo on the Header of the MainPage
Scenario	1. If User is already on MainPage → User will stay on MainPage 2. If User is on a different page (ex : Social, Create(Add) Map Page) → User will be redirected to the MainPage
Exceptions	N/A

Number	5.2
Name	Add Maps
Actors	Guest, User, Admin
Story	1. Guest <ul style="list-style-type: none"> a. A guest will left Click on AddMap link on the Header of the Page b. Popup will show :” Guest unavailable to access, login/register to use this features” 2. User and Admin <ul style="list-style-type: none"> a. Left Click on AddMap Link on the Header of the Page

Scenario	N/A
Exceptions	N/A

Number	5.3
Name	Go to My Maps
Actors	Guest, User, Admin
Story	<ol style="list-style-type: none"> 1. Guest <ol style="list-style-type: none"> a. A guest will leftClick on the MyMaps header title b. Popup will show :” Guest unavailable to access, login/register to use this features” 2. Logged in user and Admin <ol style="list-style-type: none"> a. Left Click on MyMapsPage on the Header of the Page
Scenario	N/A
Exceptions	N/A

Number	5.4
Name	Go to Social
Actors	Guest, User
Story	<ol style="list-style-type: none"> 1. Left Click on SocialPage on the Header of the Page
Scenario	N/A
Exceptions	N/A

Number	5.5
Name	Go to User Profile
Actors	Guest, User, Admin
Story	<ol style="list-style-type: none"> 1. Guest <ol style="list-style-type: none"> a. A guest will leftClick on the UserProfile header image b. Popup will show : ”Guest unavailable to access, login/register to use this features” 2. Logged in user and Admin <ol style="list-style-type: none"> a. Left Click on UserProfile on the Header of the Page
Scenario	N/A
Exceptions	N/A

Number	5.6
Name	View Map Detail
Actors	Guest, User, Admin
Story	<ol style="list-style-type: none"> 1. Choose Map that user wants to see 2. Left Click on the Map thumbnail to see details
Scenario	N/A
Exceptions	N/A

Number	5.7
Name	Open Options
Actors	Guest, User, Admin

Story	<ol style="list-style-type: none"> 1. Click on the option menu on the Map Image 2. Choose option(Fork maps,Share maps, Export maps)
Scenario	N/A
Exceptions	N/A

Number	5.8
Name	Map Search
Actors	Guest, User, Admin
Story	<ol style="list-style-type: none"> 1. Guest <ol style="list-style-type: none"> a. Left Click on the search bar b. Type search keyword c. Click on search button 2. User and Admin <ol style="list-style-type: none"> a. Left Click on the search bar b. Type search keyword c. Click on search button d. Search bar would be cleared and a tag will show up on the bottom of the search bar to show that the result is filtered by that specific keyword e. Left click on the tag's x button to remove the search filter f. Search filter will be gone and all the maps will be rendered on the screen

Scenario	<ol style="list-style-type: none"> 1. User would like to search for a map that shows the birth rate of a specific region. The user would type “BirthRate” in the search bar and click the search button 2. User will have a tag with the text “BirthRate” below the search bar. User would like to look at all the maps again, so he/she will click the tag and see the entire maps
Exceptions	<ol style="list-style-type: none"> 1. When user keyboard input is invalid 2. User clicks somewhere near the tag but not the tag’s x(close) button

CASE 6

Number	6.1
Name	Map Navigation
Actors	Admin, User, Guest
Story	<ol style="list-style-type: none"> 1. Left click and drag to move within the map 2. Scroll up to zoom in 3. Scroll down to zoom out 4. Left click regions to get more information
Scenario	N/A
Exceptions	N/A

Number	6.2
Name	Open Options: Viewing Others’ Map
Actors	Admin, User, Guest

Story	<ol style="list-style-type: none"> 1. Left click on the three-dot menu to open a dropdown menu of actions 2. Choose to either fork, share maps, or export map <ol style="list-style-type: none"> a. 'Fork' will redirect user to the 'MapEdit' page (<i>only for admin and users</i>) b. 'Share' will generate a link to that map that users can copy c. 'Export' will download the current map as a GeoJSON file
Scenario	N/A
Exceptions	N/A

Number	6.3
Name	Comment
Actors	Admin, User
Story	<ol style="list-style-type: none"> 1. Fill out the input field and left click on the 'comment' button to post comment on that map 2. Left click on 'reply' to reply to others' comments
Scenario	<ol style="list-style-type: none"> 1. E.g "You made a typo for the information in Japan. You should fix it!" <p>Once the comment is made, it will be visible to anyone.</p>
Exceptions	<ol style="list-style-type: none"> 1. If the comment has no title or body and the user clicks on 'comment', then an error message will show up and the post will not be made.

CASE 7

Number	7.1
---------------	-----

Name	Navigate map
Actors	User, Admin
Story	<ol style="list-style-type: none"> 1. Left click and drag to move within the map 2. Scroll up to zoom in 3. Scroll down to zoom out 4. Left click regions to get more information
Scenario	<ol style="list-style-type: none"> 1. The user would like to move the points of view on the map on the 'My maps details pages'. The user clicks the map and drag it to move. 2. The user would like to zoom in on the map on the 'My maps details pages'. The user scrolls up on the map to zoom in. 3. The user would like to zoom out on the map on the 'My maps details pages'. The user scrolls down on the map to zoom in. 4. The user would like to get information of a specific region on the map on the 'My maps details pages'. The user left click on the region which user wants to know
Exceptions	N/A

Number	7.2
Name	Share map
Actors	User, Admin
Story	<ol style="list-style-type: none"> 1. Left click the share map button. 2. The map will be shared on Social page
Scenario	N/A
Exceptions	N/A

Number	7.3
Name	Export map
Actors	User, Admin
Story	<ol style="list-style-type: none"> 1. Left click the export button 2. Select the file type (PNG, JPG, and your own JSON file format) 3. The file will be downloaded
Scenario	N/A
Exceptions	N/A

CASE 8

Number	8.1
Name	Edit maps
Actors	User, Admin
Story	<ol style="list-style-type: none"> 1. <i>Refer to 9.1 for editing a map</i> 2. Click on “save changes” to save the changes made
Scenario	<ol style="list-style-type: none"> 1. The user would like to delete the region on the map on the 'EditMapPage'. The user right clicks the map region and deletes the data.
Exceptions	N/A

Number	8.2
Name	Edit maps' details

Actors	User, Admin
Story	<ol style="list-style-type: none"> 1. Right click of the map region 2. Modify the map data(name, visibility, additional information) for the region. 3. Click on “save changes” to save the changes made
Scenario	<ol style="list-style-type: none"> 1. The user would like to modify the region on the map on the 'EditMapPage'. The user right clicks the map region and changes the region’s detail data such as name, visibility.
Exceptions	N/A

Number	8.3
Name	Edit map visibility
Actors	User, Admin
Story	<ol style="list-style-type: none"> 1. Click the private toggle button to prevent other users from seeing it 2. Click on “save changes” to save the changes made
Scenario	N/A
Exceptions	N/A

Number	8.4
Name	Redo
Actors	User, Admin
Story	<ol style="list-style-type: none"> 1. Press the redo button to cancel to undo modification

Scenario	N/A
Exceptions	If there is no transaction to redo, the button will be disabled

Number	8.5
Name	Undo
Actors	User, Admin
Story	1. Press the undo button to restore the modification
Scenario	N/A
Exceptions	If there is no transaction to undo, the button will be disabled

Number	8.6
Name	Save changes
Actors	User, Admin
Story	1. Press the save button to save changes.
Scenario	N/A
Exceptions	1. If required entries are not entered (such as map name), the save button is not activated.

Number	8.7
Name	Delete map
Actors	User, Admin

Story	<ol style="list-style-type: none"> 1. Left click the delete map button 2. The map file which the user is creating will be deleted
Scenario	N/A
Exceptions	N/A

CASE 9

Number	9.1
Name	Create map
Actors	User, Admin
Story	<ol style="list-style-type: none"> 1. Login user clicks the “Create new map” button on the “MapbookMainPage” 2. On the “CreateNewMapPage”, users have to fill in input data 3. Enter a name for the created map 4. Enter a topic for the created map 5. Select a map template (Optional) 6. Select map’s visibility 7. Click on different region(s) on map graphics to form a new created map 8. After left-clicking on each region, there will be a modal pop up prompting users to enter data for each selected region 9. After finishing adding data to each region, click the “Add data” button to submit input data for each region, the modal pop up will be closed and redirected users back to the “CreateNewMapPage” 10. Login users also have an option to import an external map to create their own map 11. Click “Create map” button

	<p>12. After successfully creating the map, it will take users to the “MyMapDetailsPage.”</p>
Scenario	<ol style="list-style-type: none"> 1. User login and land on the “MapbookMainPage” 2. User taps the “Create new map” button 3. User is then directed to the “CreateNewMapPage” and enters the following information: <ol style="list-style-type: none"> a. Map Name: 2023 IOS vs. Android Usage Comparison b. Topic: “Technology” c. Templates (Optional): Pie chart map d. Visibility: Public e. User ignores the “Import Map” button since he/she wants to create a map from scratch 4. User then select regions to form their map by left clicking on the desired regions 5. User selected all southeast asian countries 6. After selecting each country, a modal pops up and prompting user to type in data for that country 7. User enter the usage percentage of each devices in each country 8. User clicks the “Create map” button on the “CreateNewMapPage” to submit the created map’s data. 9. The server validates input data and checks if each provided data meets the requirement (e.g., type checking). 10. If all data is valid, the server creates a new map. 11. The user will be redirected to the “MyMapDetailsPage” with a successful message.
Exceptions	<ol style="list-style-type: none"> 1. If the provided name of the map already exists in the user’s map inventory, display an error message: “The name of the map already exists in your map inventory. Please provide a unique name for your map.”

	<ol style="list-style-type: none"> If the user forgot to complete or leave the required input fields empty (e.g., Map Name, Topic, Visibility), display an error message: “Please fill in all the required fields.” If a general issue occurs during the registration process (e.g., database connection issue), display an error message: “Creating map failed due to an error. Please try again later.”
--	---

Number	9.2
Name	Add data to each region of a map
Actors	User, Admin
Story	<ol style="list-style-type: none"> On the “CreateNewMapPage”, after a user fills out map’s information like name, topic, visibility, and template, they can select the desired regions to form their map Each region on a map serves as a button to open the “Add data to a region” modal. After a user left-clicks on any regions they want to add data to, a “Add data to a region” modal will pop up Then a user adds input data that they want that clicked region to have. After inputting the data for the region, a user has to click the “Add data” button on that modal to submit the input information and close the modal After closing the modal, the user will be redirected back to the “CreateNewMapPage”.
Scenario	<ol style="list-style-type: none"> A user fills in the following information for their map: <ol style="list-style-type: none"> Map Name: 2023 IOS vs. Android Usage Comparison Topic: “Technology”

	<ul style="list-style-type: none"> c. Templates (Optional): Pie chart map d. Visibility: Public e. User ignores the “Import Map” button since he/she wants to create a map from scratch <ol style="list-style-type: none"> 2. User then select regions to form their map by left clicking on the desired regions 3. User selected all southeast asian countries 4. After selecting each country, a modal pops up and prompting user to type in data for that country 5. User enter the usage percentage of each devices in each country 6. User clicks the “Create map” button on the “CreateNewMapPage” to submit the created map’s data. 7. The server validates input data and checks if each provided data meets the requirement (e.g., type checking). 8. If all data is valid, the server creates a new map. 9. The user will be redirected to the “MyMapDetailsPage” with a successful message.
Exceptions	<ol style="list-style-type: none"> 1. If a general issue occurs during the registration process (e.g., database connection issue), display an error message: “Adding data to a region failed due to a database connection error. Please try again later.”

Number	9.3
Name	Import map
Actors	User, Admin

Story	<ol style="list-style-type: none"> 1. After clicking the “Create New Map” button and redirecting to the “CreateNewMapPage”, login users have an option to import an external map to their new map creation 2. Login users click on the “Import” button 3. They will then redirect to their local drive and will be able to select three types of files to upload (e.g., SHP/DBF, GeoJSON, KML) 4. After importing a file from the user's local drive, they can fill in the rest of the map’s information (e.g., Name, Visibility) and click the “Save” button to save changes. 5. They will be able to add data to each region of the imported map 6. After successfully creating the map, it will take users to the “MyMapDetailsPage.”
Scenario	<ol style="list-style-type: none"> 1. A user clicks on the “Create new map” button on the “MapbookMainPage” 2. A user clicks the “Import” button on the “CreateNewMapPage” 3. A user selects “ios_vs_andriod.geojson” from their own computer 4. After importing the file, a user enters the following information: <ol style="list-style-type: none"> a. Map Name: 2023 IOS vs. Android Usage Comparison b. Topic: “Technology” c. Templates (Optional): Pie chart map d. Visibility: Public e. User add data to each region of the imported map 5. User clicks the “Create map” button on the “CreateNewMapPage” to submit the created map’s data. 6. The server validates input data and checks if each provided data meets the requirement (e.g., type checking). 7. If all data is valid, the server creates a new map. 8. The user will be redirected to the “MyMapDetailsPage” with a successful message.

Exceptions	<ol style="list-style-type: none"> 1. If a user uploads an invalid type of file, display an error message: “Please upload a valid file.” 2. If a general issue occurs during the registration process (e.g., connection issue), display an error message: “Importing a map failed due to a generic error. Please try again later.”
-------------------	--

CASE 10

Number	10.1
Name	Filter map’s topic
Actors	Guests, User, Admin
Story	<ol style="list-style-type: none"> 1. The user types in the search box 2. The user clicks the “Search” button or presses the “Enter” key to initiate the search 3. Then the page will show only the maps that match the same searched topic
Scenario	<ol style="list-style-type: none"> 1. In the search box, the user enters a search query “Technology” 2. The user clicks the “Search” button or presses the “Enter” key to initiate the search 3. The application processes the search query and displays a list of maps that have “Technology” as their topics 4. The page displays a text: “12 results found” below the search box and above the search results
Exceptions	<ol style="list-style-type: none"> 1. If the user attempts to initiate a search with an empty query, it will be considered as no filter and the page shows maps normally. 2. If the search query does not yield any results, display a text: “No results found for your search. Please try a different search term.”

	3. If a general issue occurs during the registration process (e.g., connection issue), display an error message: “Searching maps failed due to a generic error. Please try again later.”
--	--

Number	10.2
Name	View discussion posts
Actors	Guests, User, Admin
Story	<ol style="list-style-type: none"> 1. The user left clicks on the discussion post they want to view 2. Then the user will be redirected to the “DiscussionPostPage” to view the post in more detail
Scenario	<ol style="list-style-type: none"> 1. There are a list of discussion posts created by all users on the “SocialPage” 2. The user left clicks on one of the discussion posts, titled “2023 IOS vs. Android Usage Comparison” 3. The user will be redirected to the “Discussion post” page to view the post in more detail
Exceptions	N/A

Number	10.3
Name	Comment on discussion posts
Actors	User, Admin
Story	<ol style="list-style-type: none"> 1. From the “Discussion post” page, there will be 3 sections: Discussion Post’s description, comments, and comment box 2. The user left clicks on the comment box at the bottom of the page 3. The user types in their comment

	<ol style="list-style-type: none"> After typing the comment, the user clicks “Post” button to post their comment on that discussion post The comment will show up in the comment section right above the comment box and below the post description
Scenario	<ol style="list-style-type: none"> The user lands on the “Social page” There are a list of discussion posts created by all users The user left clicks on one of the discussion posts, titled “2023 IOS vs. Android Usage Comparison” The user will be redirected to the “Discussion post” page to view the post in more detail The user comments “Keep up the good work!” and post the comment The comment will show up in the comment section right above the comment box and below the post description
Exceptions	<ol style="list-style-type: none"> If a general issue occurs during the registration process (e.g., database connection issue), display an error message: “Adding a comment failed due to a database connection error. Please try again later.”

Number	10.4
Name	Go to the next/previous page on the Social page
Actors	Guests, User, Admin
Story	<ol style="list-style-type: none"> After land on the “MapbookMainPage”, login user clicks the “Social” button The “SocialPage” will list all of the map posts created by all of the users

	<ol style="list-style-type: none"> 3. On the “SocialPage”, there are “<<” and “>>” buttons at the bottom of the page 4. Click on the “>>” button will take the user to next pages of the “SocialPage” 5. Click on the “<<” button will take the user to previous pages of the “SocialPage”
Scenario	<ol style="list-style-type: none"> 1. User land on the “Mapbook Main page” 2. User clicks on the “Social” button 3. User is then directed to the “SocialPage” 4. User will then see a list of maps created by all other users 5. At the bottom of the page, a user clicks the “>>” button to go to the next page of the “SocialPage” 6. At the bottom of the page, a user can also clicks the “<<” button to go to the previous page of the “SocialPage”
Exceptions	<ol style="list-style-type: none"> 1. Immediately after clicking the “Social” button on the “MapbookMainPage” and land on the “Social page”, the users will not be able to see the “<<” button and they can not go to the previous page of “SocialPage” since it already is the first page of the “SocialPage”. 2. When users navigate to the last page of the “SocialPage”, users will not be able to see the “>>” button and they can not go to the further pages of the “Social page” since it already is the end of the “SocialPage”. 3. If a general issue occurs during the registration process (e.g., database connection issue), display an error message: “Going to the next/previous page failed due to a generic error. Please try again later.”

CASE 11

Number	11.1
Name	Edit User Information
Actors	Admin, User
Story	<ol style="list-style-type: none"> 1. Click on the 'edit profile' button to enter profile editing mode 2. The profile editing mode will change the user's username and password to an input field 3. The user will make desired changes 4. Click 'save' button to save changes made
Scenario	<ol style="list-style-type: none"> 1. JohnDoe123 -> JohnDoe321: → This change will update the user's username and will be used to login to their account from then on.
Exceptions	<ol style="list-style-type: none"> 1. Invalid input for username will show an error (e.g empty, exceeding the character limit)

Number	11.2
Name	Edit User Profile Picture
Actors	Admin, User
Story	<ol style="list-style-type: none"> 1. Click on 'upload/change image' to upload or change profile picture 2. Select an image file from local machine and it will be uploaded as a profile picture
Scenario	N/A
Exceptions	<ol style="list-style-type: none"> 1. If a file type that is not an image file is uploaded or the image file exceeds the size limit, an appropriate error message will be displayed.

Number	11.3
Name	Logout
Actors	Admin, User
Story	1. Left click on the ‘logout’ button to logout of the currently logged in account and redirect to the “main page”
Scenario	N/A
Exceptions	N/A

Number	11.4
Name	Remove Account
Actors	User
Story	<ol style="list-style-type: none"> 1. Click on ‘remove account’ to delete currently logged in account 2. A confirmation popup will show up– click ‘yes’ to continue or ‘no’ to cancel 3. If the user clicks ‘yes’ their account will be removed and will be redirected to the landing page
Scenario	N/A
Exceptions	N/A

CASE 12

Number	12.1
Name	Create more maps
Actors	User, Admin

Story	<ol style="list-style-type: none"> 1. Left click the create button. 2. It will deliver user to Add map page
Scenario	<ol style="list-style-type: none"> 1. The user would like to create a new map on the 'My maps pages'. The user clicks the create button after the map cards.
Exceptions	N/A

Number	12.2
Name	Delete created maps
Actors	User, Admin
Story	<ol style="list-style-type: none"> 1. Left click the delete button on the map. 2. The map will be deleted
Scenario	<ol style="list-style-type: none"> 1. The user would like to remove the created map on the 'My maps pages'. The user clicks the delete button which is located right bottom of the map cards.
Exceptions	N/A

Number	12.3
Name	View created maps
Actors	User, Admin
Story	<ol style="list-style-type: none"> 1. left click on the map to view users created map 2. It will deliver user to My Map Details page
Scenario	<ol style="list-style-type: none"> 1. The user would like to see the created map on the 'My maps pages'. The user clicks the map cards to see it.

Exceptions	N/A
-------------------	-----

CASE 13

Number	13
Name	Manage as Admin
Actors	Admin
Story	<ol style="list-style-type: none"> 1. Admin will land to the MainPage of the WebApplication after login 2. On the header of the page, left click “Admin” label to go to AdminManagePage for managing accounts, maps, discussion etc
Scenario	N/A
Exceptions	N/A

CASE 14

Number	14
Name	Choose Category
Actors	Admin
Story	<ol style="list-style-type: none"> 1. Admin will be able to choose from the three menus <ol style="list-style-type: none"> a. Manage Maps b. Manage Users c. Manage Social Posts
Scenario	<ol style="list-style-type: none"> 1. Admin left click Manage Maps <ol style="list-style-type: none"> a. Land to Manage Maps Page 2. Admin left click Manage Users <ol style="list-style-type: none"> a. Land to Manage Users Page

	3. Admin left click Manage Social Posts a. Land to Manage Social Posts Page
Exceptions	N/A

CASE 15

Number	15
Name	Manage Maps
Actors	Admin
Story	<ol style="list-style-type: none"> 1. The admin will be able to view all the maps that were created 2. On right click on a map (where all the maps are visible) a ‘delete’ button will show up, and the map will be deleted if the admin left clicks it 3. When viewing a map (after clicking on a map), they can left click a delete button to delete the map they are viewing
Scenario	N/A
Exceptions	N/A

CASE 16

Number	16
Name	Manage users
Actors	Admin
Story	<ol style="list-style-type: none"> 1. Admin will be able to see a list of users on the “ManageUsersPage” 2. Right click on each user will display a “Delete user” button

	3. After clicking on the “Delete User” button, that user account will be deleted from the database
Scenario	<ol style="list-style-type: none"> 1. Admin right click on a user named “John” 2. Admin clicks on the “Delete User” button 3. User “John” has been deleted from the database
Exceptions	N/A

CASE 17

Number	17
Name	Manage Social Posts
Actors	Admin
Story	<ol style="list-style-type: none"> 1. The admin will be able to view all the social posts and comments of the post that were created. 2. On right click on the posts, a ‘delete’ button shows up, and the post will be deleted by clicking the delete button. 3. When they view the posts, they can delete the post by clicking the delete button. 4. They also delete specific comments on the post by clicking the delete button on the comments.
Scenario	N/A
Exceptions	N/A

CASE 18

Number	18
Name	Password Recovery

Actors	Guest, User, Admin
Story	<ol style="list-style-type: none"> 1. The user will be prompted to fill in their email for password recovery 2. Once the user enters proper information, they will click on the “reset password” button to get a reset email. 3. An alert prompt will show for alerting that the email is sent and the user will be redirected to the ‘login page’.
Scenario	<ol style="list-style-type: none"> 1. User email : test@gmail.com 2. User will enter user credentials(if needed) and get a recovery email to reset the password
Exceptions	<ol style="list-style-type: none"> 1. When a user/admin inputs a invalid email address

III. User Interface Model

1. User Interface View Listing

View Number	Name	Description
1	Landing page	Initially, Landing Page will popup. This page will show a description of what the web will provide
2	Get started page (popup)	Users could choose either to continue as a guest or to login and register.
3	Login page	Users will enter credentials(:email and password) to login.
4	Register page	User who does not have a account will create one with valid credentials
4-1	Register page (success)	After creating a account, user will be sent to the login page to login with the created data
5	Password recovery page	User who forgot their password will get a password reset link, via entered email address
5-1	Password recovery page(success)	After the password gets successfully reseted, user ill have to login with the updated data
6	Main Page	<p>After login, the user will land on this page</p> <p>The MapBook main page will display a list of created maps created by all of the users and a user can search for a specific map post via a search box.</p> <p>From the list, a login user can also share, fork or export maps by clicking on the vertical dots option button on each map.</p>
7	Map Details Page	<p>Display each map post in details including navigating map, commenting (for login users only), liking, editing (if one views their own map), forking maps (for login users only), sharing and exporting map.</p> <p>There will be different sub screens displayed when a user clicks on the Fork, Export, Share and Options buttons.</p>
8	Edit Map Page	<p>Displays a whole map where users can make changes to a map by adding, editing and deleting data of a region, changing the map's visibility, and importing a map from their local drive.</p> <p>There will be different sub screens displayed when a user clicks on the Import, Visibility, Save, Fork and Add/Edit data buttons.</p>

9	Add map page	Displays a whole map where users can make changes to a map by adding, editing and deleting data of a region, changing the map's visibility, and importing a map from their local drive.
10	Social page	Displays all postings on the MapBook. Users can see the title of the posting and by clicking it, they can see the details of the post.
11	Discussion post page	Displays the details of the social posting. It will also show comments of the post.
12	My maps page	Users can view all of the pages they have created on the My maps page. On this page, users can check out the map in the draft. And they can create a new map, delete the map they created, and see details of the map by clicking on it.
13	User Details page	Consists of currently logged in user's personal information including their profile picture, username, email, and phone number, and few other buttons such as editing personal information and logging out.
14	Admin home page	This will be the same as the main page, but will additionally include another option in the header called "admin".
15	Admin manage maps page	Displays all the maps created by all users and can view, edit, or delete any of them.
16	Admin manage users page	Displays all the accounts that are registered and can view, edit, or delete any of them.
17	Admin manage discussion page	Displays all the discussion posts created by users and can view, edit, or delete any of them.

2. User Interface Mockup Diagrams

-  User Interface Mockup Diagram.pdf

IV. Requirement Presentation(Attached to the end of this PDF)

-  416 Requirements Presentation (Team Black)

V. Data Model

1. Persistent User Model

※ N in the null column means that it cannot have a value of NULL ※

Table Name		User	Table Description		Contains User's information	
Note						
NO	PK	physical name	Type	NULL	default value	description
1	PK	_id		N		
2		name	String	N		
3	PK	user_name	String	N		
4		pw	String	N		have to be serialized
5		email	String	N		fit with email format
6		is_admin	boolean	N	False	
7		profile_img	String	N		
8		maps_created	Array	N		array of Map_id

Table Name		Map	Table Description		Contains map data	
Note						
NO	PK	physical name	Type	NULL	default value	description
1	PK	_id		N		
2		map_name	String	N		
3		topic	String	N		
4		is_visible	boolean	N	True	True: public / False:private
5		file_path	String	N		Contains the path of the geojson file
6		user_id	ObjectId	N		User_id(only) will be stored
7		template	enum	N	["HEAT","THEME","BAR","PIE","CIRCLE"]	
8		colors	Array			Array of Strings(hex values) will be stored
9		data_value	Map			Key:data_name Value: num(user input)
10		data_names	Array			Array of Strings
11		heat_range	Array		[0,0,0,0,0]	Only when enum ="HEAT"

12		map_users_like d	Map	N		
----	--	---------------------	-----	---	--	--

Table Name		Map Comment	Table Description		Comments for each map that has been created	
Note						
NO	PK	physical name	Type	NULL	default value	description
1	PK	_id		N		
2		map_comment_c ontent	String	N		The text that users commented on a map
3		map_comment_o wner	ObjectId	N		Store the comment’s owner ObjectId
4		map_id	ObjectId	N		Store the ObjectId of the post which the comment has been commented on

Table Name		Map Reply	Table Description		Replies of each comment of the map	
Note						
NO	PK	physical name	Type	NULL	default value	description
1	PK	_id		N		
2		map_reply_conte nts	String	N		The text that users reply on a map comment
3		map_reply_owne r	ObjectId	N		Store the reply’s owner ObjectId
4		map_comment_i d	ObjectId	N		Store the ObjectId of the user which the reply has been commented on

Table Name		Discussion Post	Table Description		Contains data for a discussion post	
Note						
NO	PK	physical name	Type	NULL	default value	description
1	PK	_id		N		
2		title	String			

3		post_content	String	N		
4		post_images	Array	N		
5		post_owner	ObjectId			
6		view_count	int		0	
7		discussion_users_liked	Map	N		

Table Name		Discussion Post Comment	Table Description		Comments for each discussion post on the discussion page	
Note						
NO	PK	physical name	Type	NULL	default value	description
1	PK	_id		N		
2		discussion_comment_content	String	N		The text that users commented on a discussion post
3		discussion_comment_owner	ObjectId	N		Store the comment’s owner ObjectId
4		discussion_post_id	ObjectId	N		Store the ObjectId of the discussion post which the comment has been commented on

Table Name		Discussion Post Reply	Table Description		Replies of each comment of the discussion post on the discussion page	
Note						
NO	PK	physical name	Type	NULL	default value	description
1	PK	_id		N		
2		discussion_reply_contents	String	N		The text that users reply on a discussion comment
3		discussion_reply_owner	ObjectId	N		Store the reply's owner ObjectId
4		discussion_comment_id	ObjectId	N		Store the ObjectId of the user which the reply has been commented on

2. Data Dictionary

_id

- Id of Objects
- These Objects could be either : User, Map, Post, Comment, Replies
- Will be the primary key for every table

colors

- Selection of colors that will be used in the Map
- Type : Array of Strings

data_names

- Type of data that will be compared in each region
- Type : Array of Strings

data_value

- Key value pair of data that will store data_name as a key and take user input value, number type, as the value
- Type : Dictionary

discussion_comment_contents

- The text that users commented on a discussion post.
- Type: String

discussion_comment_owner

- Stores the discussion post comment's owner ObjectId
- Type: ObjectId

discussion_comment_id

- Stores the discussion comment's ObjectId which reply is hanging on
- Type: ObjectId

discussion_post_id

- Stores the ObjectId of the discussion post which the comment has been commented on

discussion_reply_contents

- Stores the text that users reply to comment
- Type : String

discussion_reply_owner

- Stores the discussion reply's owner ObjectId
- Type: ObjectId

discussion_users_liked

- Array of user ObjectIds of users that liked a discussion post

- Type: Array

email

- Stores the email address of user
- Type : String

heat_range

- Array of number that user will be able to modify, to select the range of colors, to make a gradient on the map for easy visualization for comparing data
- Type : Array of numbers

is_admin

- Indicates whether or not the user is an administrator
- Type : Boolean
- True: the user is admin
- False: the user is not admin

json_path

- path of the geo json
- Type : String

map_comment_contents

- The text that users commented on a map.
- Type: String

map_comment_id

- Stores the map comment's ObjectId which reply is hanging on
- Type: ObjectId

map_comment_owner

- Stores the map comment's owner ObjectId
- Type: ObjectId

map_id

- Stores the ObjectId of the map which the comment has been commented on

map_name

- Name of the map
- Type : String

map_reply_contents

- Stores the text that users reply to comment

- Type : String

map_reply_owner

- Stores the map reply's owner ObjectId
- Type: ObjectId

map_users_liked

- Array of user ObjectIds that liked the current map
- Type: Array

maps_created

- Stores ObjectIds of map that the user created before
- Type : Array of ObjectId

name

- Stores the name of user
- Type : String

owner

- Creator of the Map
- Type : User._id

post_comments

- Array of post_comment objectIds
- Type: Array

post_content

- Text body of a discussion post
- This field may be left empty
- Type: String

post_images

- Array of paths to images the user wishes to post
- This field can be left empty
- Type: Array

post_writer

- ObjectId of the user who created the current discussion post
- Type: ObjectId

profile_img

- Stores the path of profile image of user
- Type : String

pw

- Stores the password of user
- Type : String
- Should be encrypted

template

- Map templates that user will be able to select
- Type : ENUM
- Possible ENUM strings will be : ["HEAT", "THEME", "BAR", "PIE", "CIRCLE"]

title

- Title of a discussion post
- Type: String

topic

- Category/topic of the map (ex : Economics, Environment etc)
- Type : String

user_name

- Stores the user name (ID or account)
- Type : String

view_count

- View count of a discussion post
- Type: Integer

visibility

- Whether the map is visible or not
- Type : boolean

VI. Software Model

1. Similar Problems

- Social-Media-Web-Application
 - Social media web application created using React and mongoDB. The repository is similar to our discussion page and comment we want to create, and the technology stack is similar. Therefore, referring to this repository, it will help us develop our social/post page and comment function.
<https://github.com/tristingreenstein/Social-Media-Web-Application/tree/main>
 - Similar tasks: managing posts, commenting on posts, sharing posts, user authentication
 - Technologies that will be used from this project: MERN, JWT, REST API, BCrypt
- Day logger
 - Similar tasks:
 - Creating a personal account system (register, login, logout, personal profile information)
 - Uploading information to the public as a specific user
 - Creating an admin account that has access to all other users' information
 - Technologies that will be used from this project: MERN, MySQL, Heroku, Netlify, Bcrypt, Cloudinary, Fetch API, Material UI

2. Complete Technology Set

- Front-End (Based on React)
 - React.js: user interface rendering, create UI components
 - React Router: Handling client-side routing and navigation with in SPA
 - Mapbox GL: Advanced map features and customization
 - Material UI + Bootstrap CSS: styling

- Redux: State Management in React Application (Since we might have complex state requirements)
 - Redux tutorial - <https://www.youtube.com/watch?v=zrs7u6bdbUw>
 - React Redux Login, Logout, Registration example with Hooks - <https://www.bezkoder.com/react-hooks-redux-login-registration-example/>
- Axios / built in FetchAPI: Making HTTP requests to backend API
- Back-End (Node.js and Express.js)
 - Node.js: Server-side runtime environment
 - Express.js: routing, handling HTTP requests, managing server, user authentication, file uploads
 - MySQL2: store map data, user profiles, configurations
 - Firebase/ Cloudinary : storing and serving file upload
 - Multer: Uploading and handling map-related files such as images
 - AUTH (May use these libraries)
 - Passport.js
 - JWT
 - Bcrypt: hashing and salting passwords
 - Body-parser
 - Security (lib/middleware)
 - Bcrypt: hashing and salting passwords
 - CORS: handling cross-origin requests from front-end
 - OAuth2: securing API endpoints, user authentication and authorization
- Development
 - Git and Github: version control, collaboration
 - VS Code: Code development
 - Postman: Test API endpoints and requests
- Deployment
 - Heroku: server application
 - Netlify: client application
 - Might use techs:
 - Render

- Firebase
- AWS

3. Training Verification

a. Yunah

i. React

- CSE316 (NoteApp/DayLogger)

:Used Mern Stack to implement Web Application that interacts with users. Mainly used React, and node, express as backend while using mongo as the main backend server. Also used Heroku and Netlify for deployment.

- ExpenseTracker

: Expense Tracking App that allows users to enter the category of their transactions and the amount. The app is based on React and uses Redux for state management and passport.js for authentication.

- Other : node, express, mongo

o Haneul

i. Project

- CSE219 (Fall 2017): prof: Richard McKenna

: Developed Metro Map Maker. At this time, the front-end design was done using Javafx without using react.

- CSE305 (Spring 2018) prof: Art Lee

: developed a web app , an Angry Book that imitates Facebook by using php and mysql. I connected the php to the database and optimized the database

ii. React

- Global Bridge Co., Ltd (Songdo, Incheon) Jan 2020 ~ Aug 2023

: Participated in the creation of the initial prototype of the company's web app. Developing a working-mockup using React and AntDesign. Application of api in collaboration with backend developers.

iii. DataBase

- Global Bridge Co., Ltd (Songdo, Incheon) Jan 2020 ~ Aug 2023

: Database schema design and optimization. Query optimization.
Connect database with other clients (c++/python/node.js)

- Ratanak Rin

- i. Day logger: allows users to set up a series of questions (similar to Google Forms) that they can then log data for those questions over multiple days, and a basic user profile, login/registration, and admin account features.

- 1. Made in CSE 316, Spring 2022
 - 2. MySQL, Express.js, React, Node.js
 - 3. Github links:

- <https://github.com/Ratanak-Rin-Sbu/Day-Logger-client> &

- <https://github.com/Ratanak-Rin-Sbu/Day-Logger-server>

- ii. Camnect: a full stack responsive social media app consists of authentication and authorization, likes, comments, add friends, posting feature, dark/light modes.

- 1. Made in Spring 2023
 - 2. MongoDB, Express.js, React, Node.js

- 3. Github link: <https://github.com/Ratanak-Rin-Sbu/Camnect>

- iii. Assign ME: an app consists of three main features, task management, schedule maker, and note taking app aiming to increase productivity for users with ease. Used a fully featured API includes authentication, CRUD operations, schema validation, serialization/deserialization, stacks integration.

- 1. Made in Spring 2023
 - 2. FastAPI, React, MongoDB

- 3. Github link: <https://github.com/Ratanak-Rin-Sbu/AssignMe>

- Samuel Han

- i. CSE 316 - MERN, Heroku, Cloudinary, Postman, Bcrypt: **Notes App/Day logger** (same as the notes app/day logger written in Yunah and Ratanak's section)

- 1. Made in Spring 2022

2. Additional tutorial links:

<https://www.youtube.com/watch?v=7CqJlxBYj-M&t=953s>

ii. MERN, Material UI : **Community Chat** - web app that allows users to chat with others based on their interests.

1. Made in June 2023

2. Front end link: <https://github.com/samuelhan713/samcord-client>

3. Back end link: <https://github.com/samuelhan713/samcord-server>

4. Tutorial link: <https://socket.io/how-to/use-with-react>

5. Tutorial link 2:

<https://dev.to/novu/building-a-chat-app-with-socketio-and-react-2edj>

4. Front-End Component Composition(Attached)

-  Design Diagram - Page 1.pdf

5. Front-End Component Design and Back-End API(Attached)

-  Front-end Component Composition.pdf

VII. Code Presentation(Attached to the end of this PDF)

-  CSE416 CODE PRESENTATION.pdf

VIII. Meeting Log

<1st meeting (09/01/2023)> → Project Discovery

- Members present: Yunah Haneul Jass Sam
- Achieved
 - Setup github repo for first assignment (load and view map files)
 - Created remote communication space (Slack)
 - Exchange information (emails...)
 - Understand the first assignment requirement
- Goals before next meeting
 - Get the first assignment done and ready to submit

- Individually understand the requirements of the second assignment and ready to discuss with the team
- Action item

Tasks	Assigned to	Status	Priority	Due Date
Setting up React application and Mapbox GL, finding resources	Yunah, Sam	Done	High	09/05/2023
Loading files, Mapbox GL config	Jass, Sam, Yunah	Done	High	09/08/2023
Converting KML, shape files to geoJson	Haneul	Done	High	09/15/2023

- Total duration: 2 hours
- Overview
 - Set up working environment
 - Study vector geographical map
 - Study the 3 main files GeoJson, KML, and Shape file

<2nd meeting (09/15/2023)> → Finalized Project Discovery & Start System Analysis

- Members Present : Yunah Hanuel Jass Sam
- Achieved
 - Submit GP 1
 - Understand and discuss GP 2
 - Goals and overview of our application
 - Added document to shared Google Drive
- Goals before next meeting
 - Write your own part of the assignment
 - Briefly brainstorm about features, design, scalability, database design (these are stuffs that we can think of ahead)
- Action item

Tasks	Assigned to	Status	Priority	Due Date
Executive Summary	Jass	Done	High	09/22/2023
Objectives	Jass	Done	High	09/22/2023

Strategies and Philosophies	Haneul	Done	High	09/22/2023
Constraints	Yunah	Done	High	09/22/2023
Actors	Sam	Done	High	09/22/2023
Services	Sam	Done	High	09/22/2023

- Total duration: 2 hours
- Overview:
 1. Load and view external map of the site
 2. Map editing (edit/add labels)
 3. Save edited map
 4. Upload map
 5. Fork map from other users

<3rd meeting (09/22/2023)> → Use cases

- Members Present : Yunah Hanuel Jass Sam
- Achieved
 - Revise SystemAnalysis Report
 - Submit Group Project 2
 - Understand and discuss GP 3 (Use Cases)
 - Create the structure/ components of the web application to figure out what kind of use cases are going to exist
 - Added document to shared Google Drive
- Goals before next meeting
 - Make notes on what kind of use cases will exist on the web application
 - Make modifications to the UI Components if necessary(e.g. features, design etc)
- Action item

Tasks	Assigned to	Status	Priority	Due Date
Landing page, main page, password recovery	Yunah	Done	High	09/29/2023
Login, map details page, profile page	Sam	Done	High	09/29/2023
Get started page, my maps	Haneul	Done	High	09/29/2023

details page, my maps page				
Registration page, create map page, social page, admin page	Jass	Done	High	09/29/2023

- Total duration: 2 hours
- Overview:
 - There will be a lot of use cases while interacting with the web, therefore creating a sketch of the web app and working, coming up with all the possible interactions that can appear in the application.

<4th meeting (09/25/2023) - zoom> → Use cases & pages

- Members Present: Yunah Hanuel Jass Sam
- Achieved
 - More discussion regarding the use case assignment
 - Finalized stories, scenarios and exceptions for use cases
- Goals before next meeting
 - Finished each assigned task individually
 - Pre-group meeting reading for UI Model assignment
 - Added document to shared Google Drive
- Action item

Tasks	Assigned to	Status	Priority	Due Date
Landing page, main page, password recovery	Yunah	Done	High	09/29/2023
Login, map details page, profile page	Sam	Done	High	09/29/2023
Get started page, my maps details page, my maps page	Haneul	Done	High	09/29/2023
Registration page, create map page, social page, admin page	Jass	Done	High	09/29/2023

- Total duration: 2 hours
- Overview: finalized formats, shared opinions regarding each use case's stories, scenarios and exceptions.

<5th meeting (10/1/2023)> → UI Model

- Members Present : Yunah Hanuel Jass Sam
- Achieved
 - Understand and discuss group UI Model assignment
 - Created shared Figma UI design space
 - Assigned tasks for each member
 - Finished app design prototype by sketch
 - Added document to shared Google Drive
- Goals before next meeting
 - Get UI View Listing done
 - Get UI mockup design done (follow the sketch)
 - Notify each other the work progress
- Action item

Tasks	Assigned to	Status	Priority	Due Date
Design prototype (sketch)	Everyone	Done	High	10/06/2023
Landing page, header, get started page, login, register, password recovery	Yunah	Done	High	10/06/2023
Main page, map details page, Edit map page	Jass	Done	High	10/06/2023
Create map page, social page, social post page, my maps page	Haneul	Done	High	10/06/2023
Profile page, user update page, admin manage map page, admin manage users page, admin manage social posts	Sam	Done	High	10/06/2023

- Total duration: 3 hrs 30 mins
- Overview:
 - Followed sketch prototype for the design
 - Learn Figma more
 - Shared some Figma resources

<6th meeting (10/1/2023)> → Requirement Presentation

- Members Present : Yunah Hanuel Jass Sam

- Achieved
 - Understand and discuss Requirement Presentation assignment
 - Assigned tasks for each member to present
 - Received feedbacks from UI model
 - Added document to shared Google Drive
- Goals before next meeting
 - Prepare for the presentation
 - Make presentation slides
 - Add missing parts of the UI Model assignment to the Requirement Presentation
- Action item

Tasks	Assigned to	Status	Priority	Due Date
Landing page, get started page, login, register, password recovery	Yunah	Done	High	10/13/2023
Main page, map details page, Edit map page	Jass	Done	High	10/13/2023
Create map page, social page, social post page, my maps page	Haneul	Done	High	10/13/2023
Profile page, user update page, admin manage map, admin manage users, admin manage social posts	Sam	Done	High	10/13/2023

- Total duration: 2 hrs
- Overview:
 - Presentation slides
 - User interface review
 - UI Model feedbacks

<7th meeting (10/12/2023)> → Requirement Presentation Recording

- Members Present : Yunah Hanuel Jass Sam
- Achieved
 - Practiced and recorded for the Requirements Presentation
 - Added parts (add map data in detail) that we missed from UI Model
 - Added document to shared Google Drive

- Created Data Model document and added to Drive
- Goals before next meeting
 - Finish the Data Model structures
 - Persistent data model
 - Finish data dictionary
- Action item

Tasks	Assigned to	Status	Priority	Due Date
Recorded presentation	Everyone	Done	High	10/13/2023
Edit recording	Sam	Done	High	10/13/2023
User, Map, Map comment, Map reply	Jass, Haneul	Done	High	10/16/2023
Social post, social post comment, social post reply	Sam, Yunah	Done	High	10/16/2023
Data dictionary	Everyone	Done	High	10/16/2023

- Total duration: 2 hrs 30 mins
- Overview:
 - Filmed Requirements Presentation & Submitted
 - Data Model will be Finished based mainly on the components created
 - Will create a table for each component and input all the necessary fields for it

<8th meeting (10/16/2023)> → Software Model

- Members Present : Yunah Hanuel Jass Sam
- Achieved
 - Added document to shared Google Drive
 - Understanding the assignment
 - Assigned tasks
- Goals before next meeting
 - Prepare software model notes
 - Work on components
 - Domain analysis
 - Complete technology set
 - Training verification

- Action items

Tasks	Assigned to	Status	Priority	Due Date
Domain analysis, technology set, training verification	Everyone	Done	High	10/23/2023
Frontend composition diagram (individually)	Everyone	Done	Medium	10/30/2023
Frontend component design (individually)	Everyone	Done	Medium	10/30/2023
Backend API	Everyone	Done	Medium	10/30/2023

- Total duration: 2 hrs

- Overview:

- Individually complete the first part of the assignment (training verification, complete technology set, domain analysis)
- Adding each other own notes

<9th meeting (10/23/2023)> → Software model notes and components

- Members Present : Yunah Hanuel Jass Sam
- Achieved
 - Complete technology set notes
 - Training verification notes
 - Domain analysis
 - Divided Frontend component design, frontend composition diagram, backend API
- Goals before next meeting
 - Get software model done and ready to submit
- Action item

Tasks	Assigned to	Status	Priority	Due Date
Frontend composition diagram (individually)	Everyone	Done	High	10/30/2023
Frontend component design (individually)	Everyone	Done	High	10/30/2023
Backend API	Everyone	Done	High	10/30/2023

- Total duration: 1.5 hrs

- Overview: Frontend composition diagram, frontend component design, and backend API

<10th meeting (10/27/2023)> → Finalized Software Model & Build 1

- Members Present : Yunah Hanuel Jass Sam
- Achieved :
 - Finalized the documentation for group assignment “Software model”
 - Divided Lucid chart, one for the FrontEnd Composition and the other for Backend Api.
 - Modified the Documentation for assignment submission, based on requirements.
- Goals before next meeting : X
- Action item : Read Build 1 assignment if posted
- Total duration: 3 hrs
- Overview:
 1. Submit SW Model GP
 2. Work on Build1 if posted

<11th meeting (11/01/2023)> → Continue Build 1

- Members Present : Yunah Hanuel Jass Sam
- Achieved :
 - Plan the development process (Divide FE/BE)
 - Choose deployment platform
 - Monday : Use Dev platform for planning build
- Goals before next meeting :
 - Plan each Build Stages
 - Based on role (FE/BE) begin composing
- Action item

Tasks	Assigned to	Status	Priority	Due Date
React components (landing page, main page)	Yunah	Done	High	11/06/2023
React components (login, register)	Sam	Done	High	11/06/2023
Frontend testing with Cypress	Sam, Yunah	Done		

Heroku front & back deployment	Haneul	Done	High	11/06/2023
Server setup, DB integration, config	Jass	Done	High	11/06/2023
Backend testing with Jest	Jass, Haneul	Done	High	11/06/2023

- Total duration: 1 hr
- Overview:
 1. Conversation on what deployment server that is going to be used
 2. Plan build
 3. Divide how the development process is going to be continued

<12th meeting (11/03/2023)> → Finalized Build 1 & Start Build 2

- Members Present : Yunah Hanuel Jass Sam
- Achieved :
 - Create platforms that are going to be used together and how the platforms are going to be used
 - Divide the components that are going to be covered(FE/BE)
 - BE: DB & server setup - Jass
 - Haneul: Heroku front and back deployment
 - FE: Divide by components, and by test cases
- Goals before next meeting :
 - Specify build Stages 3-6
 - Add more test cases
 - Back : Add test Api's
 - Front : Add Components
- Action item
 - Yunah - Components (Landing, Main)
 - Sam - Components (Login, Register)
 - Hanuel - Deployment
 - Jass - DB Setup,

→ All due on submission by Sunday Evening

Tasks	Assigned to	Status	Priority	Due Date
-------	-------------	--------	----------	----------

React components (landing page, main page)	Yunah	Done	High	11/06/2023
React components (login, register)	Sam	Done	High	11/06/2023
Frontend testing with Cypress	Sam, Yunah	Done		
Heroku front & back deployment	Haneul	Done	High	11/06/2023
Server setup, DB integration, config	Jass	Done	High	11/06/2023
Backend testing with Jest	Jass, Haneul	Done	High	11/06/2023

- Total duration: 9AM-5PM
- Overview:
 1. Use the platforms to set the web application
 2. Create the scaffold of the web application
 3. Created basic components and basic server, ready for the deployment

<13th meeting (11/10/2023)> → Continue Build 2

- Members Present : Yunah Hanuel Jass Sam
 - Achieved :
 - Rescheduled assigning components.
 - Goals before next meeting :
 - Finish Build #2 : All FE Components
 - Prepare Presentation
 - Action item
 - Yunah - Components (Specified on Monday build plan)| FE testing Presentation
 - Sam - Components (Specified on Monday build plan)| WorkFlow Presentation
 - Hanuel - Fix FE test, BEroutes | Deployment Presentation
 - Jass - FE Components, BEroutes| BE Test presentation
- 1st checkup (Monday 11/13) / 2nd (Tuesday 11/14 - Before build meeting)
- Total duration: 2H

<14th meeting (11/14/2023)> → Build 3

- Members Present : Yunah Hanuel Jass Sam
- Achieved :

- Build3 meeting → Schedule the build plan : Divide individual parts for build 3(Specification on monday)
 - Handle Heroku deployment error
 - Goals before next meeting :
 - Finish Build #3:
 - BE API routes
 - FE-BE functionality integration
 - Action item
 - Yunah - Fix CSS | Add functionality (FE-BE integration)
 - Sam - Fix CSS | Add functionality (FE-BE integration)
 - Hanuel - BE routes
 - Jass - BE routes
- Due Monday(11/21)
- Total duration: 1.5H

<15th meeting (11/21/2023)> → Build 4

- Members Present : Yunah Hanuel Jass Sam
- Achieved :
 - Build4 meeting
 - Schedule the build plan : Divide individual parts for build 4(Specification on monday)
 - Handle Profile update error
- Goals before next meeting :
 - Begin Build 4
 - Creating onClick handling for map
 - Continue FE-BE functionality integration
 - Clean Code(More readable)
 - Admin page handling
- Action item
 - Yunah ,Sam, Hanuel, Jass : Handling Click and importing map data on the map
 - Due Friday(11/24)
- Total duration: 1.5H

<16th meeting (11/24/2023)> → Continue Build 4

- Members Present : Yunah Hanuel Jass Sam
- Achieved :
 - Continue with Build4 meeting
 - Map loading issue(Create and Edit Map)
 - Divide other minor parts
 - Clean Code(More readable)
 - Admin page handling
- Goals before next meeting :
 - Continue Build 4
 - Creating onClick handling for map
 - Edit data/render data on map page
 - Continue FE-BE functionality integration
- Action item
 - Yunah ,Sam, Hanuel, Jass : Handling Click and importing map data on the map
 - Due Before next build meeting(11/27)
- Total duration: 1H

<17th meeting (11/28/2023)> → Build 5

- Members Present : Yunah Hanuel Jasson Sam
- Achieved :
 - Build5 meeting
 - Implement Create and Edit Map
 - Divide other minor parts
 - Clean Code(More readable)
 - Remove map dummy and render real map data
- Goals before next meeting :
 - Continue Build 5
 - Creating/Editing map functionality
 - Render data on map page
 - Continue FE-BE functionality integration

- Action item
 - Yunah ,Sam, Hanuel, Jass : Handling Click and importing map data on the map
→ Due Before next meeting(12/1)
- Total duration: 1H

<18th meeting (12/1/2023)> → Finalized Build 5

- Members Present : Yunah Hanuel Jass Sam
- Achieved :
 - Continue with Build4 meeting
→ Map loading issue(Create and Edit Map)
 - Divide other minor parts
 - Clean Code(More readable)
 - Admin page handling
- Goals before next meeting :
 - Continue Build 4
 - Creating onClick handling for map
 - Edit data/render data on map page
 - Continue FE-BE functionality integration
- Action item
 - Yunah : Handling Click and importing map data on the map
 - Sam : Main Page render/ My Maps rendering/ Edit Map
 - Hanuel : Img/undo redo/fork export and other options
 - Jass : Map Details
 - Common work: Cypress Test for each component
→ Due Before next Progress meeting
- Total duration: 1 Hrs

<19th meeting (12/4/2023)> → Final Build

- Members Present : Yunah Hanuel Jass Sam
- Achieved :
 - Build5 meeting

- Render MapData & Graphics on Map
- Divide other minor parts
 - Clean Code(More readable)
 - Divide render by Template
- Goals before next meeting :
 - Continue Build 4
 - Creating onClick handling for map
 - Edit data/render data on map page
 - Continue FE-BE functionality integration
- Action item
 - Yunah : UI/Frontend functionality + HeatMap Data rendering
 - Sam : UI/Frontend functionality + Circle Data rendering
 - Hanuel : Backend functionality + Thematic Data rendering
 - Jass : Map Details, Backend functionality + PieBar Data rendering
 - Common work: Test adding for done parts
 - Due Before next Progress meeting
- Total duration: 1.5 Hrs

<20th meeting (12/11/2023)> → Final Build Cont.

- Members Present : Yunah Hanuel Jass Sam
- Achieved :
 - Found errors needed to fix before system testing
 - Worked on display data for each type of maps
 - Edit map page
- Goals before next meeting: get the above tasks done
- Action items

Tasks	Assigned to	Status	Priority	Due Date
Go over the app and find all the errors	Everyone	Done	High	12/17/2023
Continue on editing map and display data on map	Everyone	Done	High	12/17/2023

- Total duration: 2 Hrs

<21st meeting (12/17/2023)> Final meeting → Finalized Final Build

- Members Present: Yunah Haneul Jass Sam
- Achieved:
 - Finished products
 - Use cases testing and go over
 - Merge all assignment documents
 - Prepared for final presentation
- Goals before next meeting - N/A
- Action items:

Tasks	Assigned to	Status	Priority	Due Date
Coding & fixing errors	Everyone	Done	High	12/17/2023
Final presentation prep	Everyone	Done	High	12/17/2023
Frontend testing with Cypress	Sam, Yunah	Done	High	12/17/2023
Merge assignment docs	Everyone	Done	High	12/17/2023
Fixed user profile & validate email during registration	Jass	Done	High	12/17/2023
Fixed dataNames, apply changes to map details	Haneul	Done	High	12/17/2023

Total duration: 1PM - 6PM

IX. Post Mortem (Next Page)

Web Development Project Postmortem

Map Book - A Web Application for editing map data.

(Duration: 8/31/2023 to 12/18/2023)

Achievements

1. Successful Launch

The project was successfully launched on December 18th, meeting the initial project timeline, on the domain <https://mapbook-f381d1faf354.herokuapp.com/>.

What Went Well & Lessons Learned

- Creating the social page where people can create and share images or their thoughts went smoothly because we effectively planned the component structures well (e.g. creating a comment section component and a separate reply comment section)
 - Choosing the appropriate Map API(Map Box) that provides plentiful features that allows editing and displaying map graphics.
 - Established the need for regular reviews to identify and address potential issues early
 - Thorough/appropriate documentation for future reference and knowledge transfer.
-

Challenges Faced

1. Front-end
 - Choosing a map api that directly supports map templates that we need
 - States that needed to be managed all over components.
 - Creating specific class names for tags so that the CSS won't affect unwanted tags.
 - Displaying map data on the map graphics.
 - Recognized the need for structural changes during the project as the components grew.
 - Facing unexpected fail cases while testing, due to the lack of considering edge cases.
 2. Back-end
 - Saving image and geojson files in external storage (firebase/cloudinary) with Multer library
 - Testing failed due to data generated during the previous test (flaky tests)
 - Implementing password recovery from scratch presented a substantial challenge, especially constructing a secure process of sending resetting password emails to users
-

What Can be Improved

1. Planning stage
 - a. Software Model
 - The format of Front-End Composition Diagram could be more precise
 - Most application state management details and transaction management for Undo/Redo should be mentioned in the Software Model
2. Implementation stage
 - a. Front-end
 - Creating a unified naming criteria for class names so that all team members can create/edit parts of the application without conflict.
 - Using redux for more data other than the user data for more efficient state management.
 - Highlighting the importance of comprehensive planning in our large project. This includes creating detailed plans for the use case model, UI model, data model, and software model.
 - Considering all edge cases in our planning to facilitate thorough testing and prevent unexpected errors in the application.
 - b. Back-end
 - Through the unification of data transfer formats, file transfer methods to complex data storage can be combined into one.
 - Create a database for testing, preventing the test from affecting the deployment server.

Attachments