



# CUSTOMAP

**Juyeon Nam**

juyeon.nam@stonybrook.edu

**Seolhee Yun**

seolhee.yun@stonybrook.edu

**Shihao Wen**

shihao.wen@stonybrook.edu

**Weikang Yang**

weikang.yang@stonybrook.edu

# **Table of Contents**

|                                     |             |
|-------------------------------------|-------------|
| <b>1. System Analysis</b>           | <b>4pg</b>  |
| 1.1 Executive Summary               |             |
| 1.2 Objectives                      |             |
| 1.3 Strategies/Philosophies         |             |
| 1.4 Constraints                     |             |
| 1.5 Actors                          |             |
| 1.6 Services                        |             |
| <b>2. Use Case Model</b>            | <b>8pg</b>  |
| 2.1 Use Case Listing                |             |
| 2.2 Use Case Descriptions           |             |
| <b>3. Use Interface Model</b>       | <b>39pg</b> |
| 3.1 User Interface View Listing     |             |
| 3.2 User Interface Mockup Diagrams  |             |
| <b>4 Data Model</b>                 | <b>54pg</b> |
| 4.1 Persistent Data Model           |             |
| 4.2 Dictionary                      |             |
| <b>5 Software Model</b>             | <b>57pg</b> |
| 5.1 Similar Problems                |             |
| 5.2 Complete Technology Set         |             |
| 5.3 Training Verification           |             |
| 5.4 Front-End Component Composition |             |
| 5.5 Front-End Component Design      |             |
| 5.6 Back-End API                    |             |
| <b>6. Deployment</b>                | <b>65pg</b> |

**7. Code** **65pg**

7.1 File Structure

7.2 Code Convention

**8 Build Schedule** **67pg**

**9 Reflection** **72pg**

# **1 System Analysis**

## **1.1 Executive Summary**

1. This project is to develop a web application that can create, edit, share and export custom graphics of maps using real-world geographic data. This app serves the needs of map enthusiasts, researchers, educators, and application developers who need map graphics that can be used for reports, social media, publications, and research papers.
2. Existing products on the market often focus on static maps or require a lot of technical expertise. The differentiation of our product is that we provide a user-friendly platform that allows users to create maps using data in a variety of file formats, customize map graphics, and build a shared collaboration community. With a vast array of customizable features, users can tailor their map graphics to align with specific objectives. This helps increase utilization by customizing sources of map data.
3. Our application focuses on creating map graphics based on administrative divisions and attaching data to those divisions. It focuses on specific features such as data attachment, collaborative editing, and multiple export formats to meet the unique needs of the target users.

## **1.2 Objectives**

1. Provides an account management system with secure authentication and recovering forgotten password functionality.
2. Provides the function of creating map graphics through file upload or forking existing maps on the platform. These maps can be set to be either public or private by the user.
3. Provides options that user can choose map type among default, heat, point, bubble, thematic, and choropleth map

4. Provides functions for users to classify, search and use public maps based on properties.
5. Provides the function of customizing map graphics using text, color and legend. Provides tools for: map viewing, navigation, naming regions, attaching custom data properties to regions, and adding decorations (texts, fonts, text positions, fill colors, color gradients, border colors, background colors, legend details, etc.).
6. Provides map graphics upload options in SHP/DBF (ZIP), GeoJSON, and KML file formats
7. Provides map graphics export options in PNG, JPG and JSON formats
8. Provides a collaborative environment by promoting community interaction through discussing, getting feedback, editing collaboratively and sharing among users.

### **1.3 Strategies/Philosophies**

1. User-centered Design: Design a user-friendly interface, and design an easy-to-use map creation and editing workflow to satisfy both beginner and experienced users.
2. Agile and Adaptive Development: Use agile methodologies for development, iteratively improve applications based on user feedback, and adapt to changing user needs flexibly.
3. Scalability: Systems need to be designed with scalability in mind to handle a growing user base and increasing amounts of data.
4. Cooperation Environment: Promote collaboration, sharing and comments among users to create a vibrant map-making community. Through map search, users can find maps from other users they want and share opinions with each other. Through this, users can point other users in the right and better direction.

5. Data Security: Implement security measures by providing a secure account system and both public and private content to protect user data and privacy.

## **1.4 Constraints**

1. Map graphic types: The application should be able to create at least five different map graphic types (e.g., heat map, choropleth map, point map, bubble map, route map, thematic map). These map type templates should be predefined and each type can be customized.
2. Vector Maps: The focus of the application is vector maps and administrative divisions.
3. Data sources: The application will rely on user-supplied data or publicly available geospatial datasets to create maps; private data sources are not in scope.

## **1.5 Actors**

1. Administrator: Manage user accounts, review all content, and maintain the operation of the platform.
2. Registered users: Have a platform account, access all functions, create maps, edit maps, share maps, participate in discussions, personalize settings, and have map ownership.

## **1.6 Services**

1. Accounts Management: User registration, identity verification, password recovery, account management.

2. Map Graphic Creation: Upload SHP/DBF (ZIP), GeoJSON, KML files, or fork existing maps on the platform to create map graphics.
3. Map Graphic Selection: Choosing map graphic. Provides default, heat, point, bubble, thematic, and choropleth map type
4. Map Graphic Editing: Graphic editing (text, color, legend), map viewing and navigation, naming regions, attaching custom data properties, and adding decorations.
5. Map Graphics Exporting: Provides map export in PNG, JPG and JSON formats.
6. Map Classification and Search: Map classification and search functions based on properties.
7. Community Interactions: Users discuss maps, give feedback, and collaborate through map sharing.

## 2. Use Case Model

### 2.1 Use Case Listing

| Use Case # | UI Context         | Use Case Name      |
|------------|--------------------|--------------------|
| 2.1        | Registration/Login | User Registration  |
| 2.2        | Registration/Login | User Login         |
| 2.3        | Registration/Login | Forgot Password    |
| 2.4        | User Dashboard     | Create New Map     |
| 2.5        | User Dashboard     | Edit Existing Map  |
| 2.6        | User Dashboard     | Search for Maps    |
| 2.7        | User Dashboard     | Map Classification |
| 2.8        | User Dashboard     | View User Profile  |
| 2.9        | User Dashboard     | View Map           |
| 2.10       | Map View           | Map Navigation     |
| 2.11       | Map View           | Discuss Maps       |
| 2.12       | Map Edit           | Set Map Type       |
| 2.13       | Map Edit           | Fork Map           |
| 2.14       | Map Edit           | Import map         |
| 2.15       | Map Edit           | Export Map         |
| 2.16       | Map Edit           | Delete editing     |

|      |                    |                                     |
|------|--------------------|-------------------------------------|
| 2.17 | Map Edit           | Share Map                           |
| 2.18 | Map Edit           | Redo/Undo                           |
| 2.19 | Map Edit           | Customize Map Graphics              |
| 2.20 | Map Customize Tool | Text                                |
| 2.21 | Map Customize Tool | Color                               |
| 2.22 | Map Customize Tool | Legend                              |
| 2.23 | User Profile       | Edit User Profile                   |
| 2.24 | User Profile       | Change Password                     |
| 2.25 | Admin Dashboard    | Manage User Accounts                |
| 2.26 | Admin Dashboard    | Review User-Generated Content       |
| 2.27 | Map Customize      | Default Map with Custom Settings    |
| 2.28 | Map Customize      | Heat Map with Custom Settings       |
| 2.29 | Map Customize      | Point Map with Custom Settings      |
| 2.30 | Map Customize      | Bubble Map with Custom Settings     |
| 2.31 | Map Customize      | Thematic Map with Custom Settings   |
| 2.32 | Map Customize      | Choropleth Map with Custom Settings |

## 2.2 Use Case Descriptions

|                  |   |
|------------------|---|
| <b>Number:</b>   | 2.1   |
| <b>Name:</b>     | User Registration   |
| <b>Actor(s):</b> | Registered user, Administrator  |
| <b>Story:</b>    | <p>1. User accesses the registration page from the login page.</p> <p>2. User enter the information required for their registration, including username, email and password.</p> <p>3. User submits a registration form.</p> <p>4. System validates the information and checks for duplicate usernames and emails.</p> <p>5. If the information is valid and unique, the user will receive a confirmation email.</p> <p>6. When the user clicks the confirmation link in the email, the user account is created and redirected to the user's dashboard.</p> |
| <b>Scenario:</b> | <p>1. User input:</p> <p>    Username: cse416</p> <p>    Email: cse416@example.com</p> <p>    Password: *****</p> <p>    Confirm Password:*****</p> <p>2. System verification information.</p>  |

|                     |  |
|---------------------|--|
|                     | <p>3. User receives a confirmation email.</p> <p>4. User clicks the confirmation link in the email.</p> <p>5. Create a new account for "cse416".</p>   |
| <b>Exceptions :</b> | <p>1. If the username or email is already taken, the system should display an error message and prompt the user to use a different username or email.</p> <p>2. If the password does not meet the security requirements, the system will provide the security requirements for the password.</p> |

|                  |  |
|------------------|--|
| <b>Number:</b>   | 2.2  |
| <b>Name:</b>     | User Login   |
| <b>Actor(s):</b> | Registered user, Administrator   |
| <b>Story:</b>    | <p>1. When the user accesses the application and clicks the "Login" button, a login form appears.</p> <p>2. Users provide their username and password.</p> <p>3. System validates the entered information.</p> <p>4. If the information is correct, the user will be logged in and redirected to their dashboard.</p> <p>5. If the information is incorrect, the system displays an error message.</p> |
| <b>Scenario:</b> | <p>1. User input:</p> <p>Email: john@example.com</p>   |

|                     |  |
|---------------------|--|
|                     | <p>Password: *****</p> <p>2. System verification information.</p> <p>3. User is logged in and redirected to their dashboard.</p>                     |
| <b>Exceptions :</b> | 1. If the username entered does not exist or the password is incorrect, the system should display an error message and prompt the user to try again. |

|                  |  |
|------------------|--|
| <b>Number:</b>   | 2.3  |
| <b>Name:</b>     | Forgot Password  |
| <b>Actor(s):</b> | Registered user  |
| <b>Story:</b>    | <p>1. User clicks the "Forgot Password" link on the login page.</p> <p>2. Users enter their user name, email address, and phone number and submit a request.</p> <p>3. System verifies that the email address is correct and exists in the user database.</p> <p>4. If the user name, email address, and phone number are valid and registered, the user can change their password.</p> <p>5. User enters a new password and confirms.</p> <p>6. User receives a confirmation message and is redirected to the login page.</p> |
| <b>Scenario:</b> | 1. User clicks "Forgot Password".  |

|                     |  |
|---------------------|--|
|                     | <p>2. User enters user name, email address, and phone number associated with their account and submits the request.</p> <p>3. User clicks change password, sets a new password and logs in successfully.</p>   |
| <b>Exceptions :</b> | <p>1. If the email address entered is not registered, the user should be notified that the email is not associated with any account.</p> <p>2. If the user name does not match the user's email address, it will be notified.</p> <p>3. If the password does not meet the security requirements, the system will provide the security requirements for the password.</p> |

|                  |   |
|------------------|---|
| <b>Number:</b>   | 2.4   |
| <b>Name:</b>     | Create New Map  |
| <b>Actor(s):</b> | Registered user   |
| <b>Story:</b>    | <p>1. On the logged in user's dashboard, the user can click the "Create New Map" button.</p> <p>2. System provides map creation options including new maps, uploading geospatial data files (SHP/DBF(ZIP), GeoJSON, KML) or forking existing maps.</p> <p>3. System processes the file and creates a new map based on the uploaded data, and enters the Map Edit interface.</p> <p>4. User-defined maps include named areas, additional data properties, and added decorations.</p> |

|                     |   |
|---------------------|---|
|                     | 5. System automatically saves the map.  |
| <b>Scenario:</b>    | <p>1. User clicks the "Create New Map" button.</p> <p>2. User selects the "Upload File" option.</p> <p>3. User uploads a GeoJSON file.</p> <p>4. System processes the file and creates a new map.</p> <p>5. Users customize the map by adding a legend, changing region colors, and attaching data properties to regions.</p> <p>6. System automatically saves the map.</p> |
| <b>Exceptions :</b> | <p>1. If there is a problem with the uploaded file or customization, the system should provide an error message.</p> <p>2. If the user cancels the map creation process, the system will return to the dashboard.</p>   |

|                  |  |
|------------------|--|
| <b>Number:</b>   | 2.5  |
| <b>Name:</b>     | Edit Existing Map  |
| <b>Actor(s):</b> | Registered user  |
| <b>Story:</b>    | <p>1. Registered users access the dashboard and select an existing map for which they have editing permissions.</p> <p>2. Users click the "Edit" button for the selected map.</p> <p>3. The system opens the Map Edit interface.</p> |

|                     |   |
|---------------------|---|
|                     | <p>4. Users make changes to the map, including named areas, additional data properties, and added decorations.</p> <p>5. Users click the "Save" button to save and finish editing their maps.</p>   |
| <b>Scenario:</b>    | <p>1. User selects an existing map titled "North America GDP Heat Map".</p> <p>2. User clicks the "Edit" button.</p> <p>3. System opens the Map Edit interface.</p> <p>4. Users update data properties, change region colors, and add text labels.</p> <p>5. System automatically saves the edited map.</p> |
| <b>Exceptions :</b> | <p>1. If a user attempts to edit a map that they do not have permission to edit, access is denied with an error message.</p> <p>2. If the user cancels the editing process, the system discards the changes and returns to the dashboard.</p>   |

|                  |  |
|------------------|--|
| <b>Number:</b>   | 2.6  |
| <b>Name:</b>     | Search for Maps  |
| <b>Actor(s):</b> | Registered user,   |
| <b>Story:</b>    | <p>1. The user accesses the dashboard.</p> <p>2. The user identifies the search bar on the dashboard.</p> <p>3. The user enters search criteria into the search bar. This can include keywords, locations, or map types.</p> |

|                     |   |
|---------------------|---|
|                     | <p>4. System retrieves and displays relevant maps based on the search criteria.</p> <p>5. Users select maps from search results to view or collaborate on.</p>  |
| <b>Scenario:</b>    | <p>1. User enters "North America" into the search bar.</p> <p>2. System displays a list of maps related to North America.</p> <p>3. Users click on the map titled "North America GDP Heat Map" to view its details.</p> |
| <b>Exceptions :</b> | <p>1. If no map matches the search criteria, a message will appear indicating that no results were found.</p>   |

|                  |   |
|------------------|---|
| <b>Number:</b>   | 2.7   |
| <b>Name:</b>     | Map Classification  |
| <b>Actor(s):</b> | Registered user   |
| <b>Story:</b>    | <p>1. On the dashboard, there is an option for "Map Classification".</p> <p>2. User clicks on the "Map Classification" option.</p> <p>3. System provides users with the option to categorize maps based on keywords, location, map type, or data properties.</p> <p>4. User enters classification criteria.</p> <p>5. System retrieves and matches maps based on classification criteria.</p> |

|                     |   |
|---------------------|---|
|                     | <p>6. System displays a list of public maps that match the classification criteria, including map titles and descriptions.</p> <p>7. User can click on a map in the list to view more details or collaborate on it.</p> |
| <b>Scenario:</b>    | <p>1. User clicks on the "Map Classification" option.</p> <p>2. User input classification criteria.</p> <p>3. System retrieves and displays a list of maps related to United States population data.</p>                |
| <b>Exceptions :</b> | <p>1. If there are no maps matching the specified classification criteria, the system should notify the user that no results were found.</p>  |

|                     |   |
|---------------------|---|
| <b>Number:</b>      | 2.8   |
| <b>Name:</b>        | View User Profile   |
| <b>Actor(s):</b>    | Registered user   |
| <b>Story:</b>       | <p>1. Users access their own profile settings, or select a specific user's profile to view.</p> <p>2. System displays the selected user's profile information, including username, profile, and public map items.</p>   |
| <b>Scenario:</b>    | <p>1. Users view other users' profiles to learn more about their map contributions and interests.</p>   |
| <b>Exceptions :</b> | <p>1. Profiles should respect privacy settings and some information may be hidden based on user preference.</p> <p>2. If a selected user does not exist or if there's an error in fetching the user data, the system will display an appropriate error message.</p> |

|                     |   |
|---------------------|---|
| <b>Number:</b>      | 2.9   |
| <b>Name:</b>        | View Map  |
| <b>Actor(s):</b>    | Registered user   |
| <b>Story:</b>       | <p>1. Registered users can access the map on the website.</p> <p>2. The user selects maps to view and explore.</p> <p>3. System displays the selected map, including relevant geographical data and markers.</p> <p>4. Users can pan, zoom, and interact with the map to explore its details.</p> |
| <b>Scenario:</b>    | <p>1. Users view and explore the maps that they choose.</p>   |
| <b>Exceptions :</b> | <p>1. Some maps may have restricted access according to the map owner's settings</p> <p>2. If the system encounters an error while retrieving the map or if the map doesn't exist, an appropriate error message will be displayed to the user.</p>  |

|                  |  |
|------------------|--|
| <b>Number:</b>   | 2.10   |
| <b>Name:</b>     | Map Navigation   |
| <b>Actor(s):</b> | Registered user  |
| <b>Story:</b>    | <p>1. Registered users select the map on the website.</p> <p>2. Users can zoom in or out and move the map left/right/up/down</p> <p>4. System displays zoomed-in or out the map and moved left/right/up/down the map</p> |

|                     |  |
|---------------------|--|
| <b>Scenario:</b>    | <ol style="list-style-type: none"> <li>1. User selects a map to explore</li> <li>2. System shows the map that the user selects</li> <li>3. User scroll or drag the mouse over the map to zoom in or out and move left/right/up/down</li> <li>4. System displays zoomed-in or out map and moved left/right/up/down map</li> </ol> |
| <b>Exceptions :</b> | <ol style="list-style-type: none"> <li>1. To a certain extent, system cannot zoom in or out and move the map left/right/up/down more</li> </ol>  |

|                  |   |
|------------------|---|
| <b>Number:</b>   | 2.11  |
| <b>Name:</b>     | Discuss Maps  |
| <b>Actor(s):</b> | Registered user   |
| <b>Story:</b>    | <ol style="list-style-type: none"> <li>1. Users participate in a discussion related to a specific map</li> <li>2. Users can provide feedback and share insights.</li> </ol>   |
| <b>Scenario:</b> | <ol style="list-style-type: none"> <li>1. User navigates to the map contents of interest.</li> <li>2. System displays the map details along with an associated discussion section.</li> <li>3. User reads the ongoing discussion.</li> <li>4. User chooses to contribute by entering their comments or feedback into a text box.</li> <li>5. User submits their comment.</li> </ol> |

|                     |   |
|---------------------|---|
|                     | <p>6. System posts the user's comment under the map discussion, with their username.</p> <p>7. Other users can now view and respond to the new comment.</p> |
| <b>Exceptions :</b> | <p>1. If the user submits an empty comment, the comment is not sent.</p>  |

|                     |  |
|---------------------|--|
| <b>Number:</b>      | 2.12   |
| <b>Name:</b>        | Set map type   |
| <b>Actor(s):</b>    | Registered user  |
| <b>Story:</b>       | <p>1. User can choose map type they want</p>   |
| <b>Scenario:</b>    | <p>1. User clicks on map type drop down</p> <p>2. User can choose map type among default, heat, point, bubble, thematic, choropleth map</p> <p>3. Users named their name and submit.</p> <p>4. The system create new map based on user's input</p> |
| <b>Exceptions :</b> | <p>1. If the user does not submits map name or type, the system prompts the user to enter them.</p>  |

|                  |  |
|------------------|--|
| <b>Number:</b>   | 2.13   |
| <b>Name:</b>     | Fork Map   |
| <b>Actor(s):</b> | Registered user  |
| <b>Story:</b>    | <p>1. On the dashboard, there is an option for "Fork map".</p> |

|                     |   |
|---------------------|---|
|                     | <p>2. User chooses the map file that will be revised.</p> <p>3. User clicks on the "Fork map" option.</p> <p>4. System creates a copy of the map and indicates that it is a forked version.</p> <p>5. User changes some features to the forked map.</p> <p>6. User saves the changes.</p> <p>7. System gives confirmation about the success of the process.</p> |
| <b>Scenario:</b>    | <p>1. User chooses the map file that they want to revise.</p> <p>2. User clicks on the "Fork map" option.</p> <p>2. User add texts and change fonts to the copied map.</p> <p>3. System gives confirmation about the change.</p>  |
| <b>Exceptions :</b> | <p>1. If there are any error during forking, shows:<br/>ERROR: fork fail, please try again.</p> <p>2. Some maps may have a limit on forks</p>   |

|                  |  |
|------------------|--|
| <b>Number:</b>   | 2.14   |
| <b>Name:</b>     | Import map   |
| <b>Actor(s):</b> | Registered user  |
| <b>Story:</b>    | <p>1. On the dashboard, there is an option for "Import map".</p> |

|                     |  |
|---------------------|--|
|                     | <p>2. User clicks on the "Import map" option.</p> <p>3. User chooses the map file that they will upload and the file format should be SHP/DBF (ZIP), GeoJSON, and KML file formats</p> <p>4. System uploads the file to the website.</p> |
| <b>Scenario:</b>    | <p>1. User clicks on the "Import map" option.</p> <p>2. User chooses the file in SHP/DBF (ZIP), GeoJSON, and KML file formats that they will import.</p> <p>3. System upload the file to the website</p>                                 |
| <b>Exceptions :</b> | <p>1. If there are any error during uploading, shows:<br/>ERROR: upload fail, please try again.</p>  |

|                  |   |
|------------------|---|
| <b>Number:</b>   | 2.15  |
| <b>Name:</b>     | Export Map  |
| <b>Actor(s):</b> | Registered user   |
| <b>Story:</b>    | <p>1. On the dashboard, there is an option for "Exportmap".</p> <p>2. User clicks on the "Export map" option.</p> <p>3. System provides users with the option to export the file with the format ".png/.jps/.geojson"</p> <p>4. User chooses the version of the export map file</p> <p>5. The system generates the corresponding version of the map file.</p> |

|                     |   |
|---------------------|---|
|                     | 6. System automatically downloads the file to the user's device.  |
| <b>Scenario:</b>    | <p>1. User clicks on the "Exportmap" option.</p> <p>2. User chooses the export version of: ".png"</p> <p>3. System auto download the file to the device</p> |
| <b>Exceptions :</b> | <p>1. If there are any error during download, shows:<br/>ERROR: download fail, please try again.</p>  |

|                  |   |
|------------------|---|
| <b>Number:</b>   | 2.16  |
| <b>Name:</b>     | Delete editing  |
| <b>Actor(s):</b> | Registered user   |
| <b>Story:</b>    | <p>1. On the edit toolbar, there is an option for "Delete editing"</p> <p>2. User clicks on the "Delete map" option.</p> <p>3. The system pops the confirmation window with: "Are you sure to delete this editing?"</p> <p>4. User clicks on "Yes"</p> <p>5. System deletes the database of this map.</p> |
| <b>Scenario:</b> | <p>1. User clicks on the "Delete editing" option.</p> <p>2. System shows the window "Are you sure to delete this editing?"</p> <p>3.1 User clicks on "Yes"</p>  |

|                     |   |
|---------------------|---|
|                     | <p>3.1 System deletes the map, and brings the user back to the render map page</p> <p>3.2 User clicks on “No”</p> <p>3.2 System brings the user back to the edit page and continue editing.</p> |
| <b>Exceptions :</b> | Confirmation to the user to delete the map being edited.  |

|                  |  |
|------------------|--|
| <b>Number:</b>   | 2.17   |
| <b>Name:</b>     | Share Map  |
| <b>Actor(s):</b> | Registered user  |
| <b>Story:</b>    | <p>1. On the dashboard, there is an option for "Share map".</p> <p>2. User clicks on the "Share map" option.</p> <p>3. The system brings the user to the “New Post” page</p> <p>4. User enter the topic and text of the post and add the current map to the post as a.png file.</p> <p>5. User clicks “post” button</p> <p>6. System saves the post brings the user to the share page, and adds the new post to the post list.</p> |
| <b>Scenario:</b> | <p>1. User clicks the share button on the edit page</p> <p>2. System shows the “New post” page</p>   |

|                     |   |
|---------------------|---|
|                     | <p>3. User fills in the “title” and “text”</p> <p>4. User add “tag”</p> <p>5. User clicks the “post” button</p> <p>6. System brings the user to the share page and shows the post list, the user can view the new post in the list.</p> |
| <b>Exceptions :</b> | <p>1. If the user doesn't fill in the title, return a reminder:<br/>“Please add a title to your post”</p>   |

|                  |  |
|------------------|--|
| <b>Number:</b>   | 2.18   |
| <b>Name:</b>     | Redo & Undo  |
| <b>Actor(s):</b> | Registered user  |
| <b>Story:</b>    | <p>1. There are redo and undo buttons in the editing toolbar.</p> <p>2. The user clicks on the undo button</p> <p>3. The system returns to the previous editing state and updates the current map state.</p> <p>4. The user clicks the redo button.</p> <p>5. The system carries out the last undone operation and updates the current map status.</p> |
| <b>Scenario:</b> | <p>1. The user clicks the undo button in the editing screen.</p> <p>2. The system returns to the previous editing state and updates the current map state.</p>   |

|                     |  |
|---------------------|--|
|                     | <p>3. The user clicks the redo button on the edit screen.</p> <p>5. The system carries out the last undone operation and updates the current map status.</p>                 |
| <b>Exceptions :</b> | Note the number of operations in the work stack, disable the undo button if there are no previous operations, and disable the redo button if there are no undone operations. |

|                  |   |
|------------------|---|
| <b>Number:</b>   | 2.19  |
| <b>Name:</b>     | Customize Map Graphics  |
| <b>Actor(s):</b> | Registered user   |
| <b>Story:</b>    | <p>1. In the toolbar for editing maps, there is an option to "add graphics".</p> <p>2. The user clicks on "Add Graphics".</p> <p>3. A pop-up will appear with options to add shapes, such as squares, triangles, circles, and other regular geometric patterns.</p> <p>4. The user adds the selected graphic to the map and edits the shape of the graphic.</p> |
| <b>Scenario:</b> | <p>1. The user clicks on "Add Shape".</p> <p>2. The system pops up a list of shapes that can be added.</p> <p>3. The user selects "rectangle".</p> <p>4. The user customizes the dimensions and boundaries of the rectangle.</p>  |

|                     |      |
|---------------------|------|
| <b>Exceptions :</b> | None |
|---------------------|------|

|                     |  |
|---------------------|--|
| <b>Number:</b>      | 2.20   |
| <b>Name:</b>        | Text   |
| <b>Actor(s):</b>    | Registered user  |
| <b>Story:</b>       | <p>1. In the toolbar, there is an option to "Add Text".</p> <p>2. The user clicks on "Add Text".</p> <p>3. After selecting this function, you can click on the map to create a transparent text box.</p> <p>4. Users can click on the existing text box to edit the content of the text box.</p> <p>5. You can click save to add a new text box.</p> |
| <b>Scenario:</b>    | <p>1. User clicks on "Add Text".</p> <p>2. The user clicks on the map and a transparent text box is generated at the clicked spot.</p> <p>3. The user repeats step 2 to add a new text box.</p> <p>4. User clicks on one of the text boxes to edit the content.</p> <p>5. The user continues to add or edit the next text box.</p>                   |
| <b>Exceptions :</b> | <p>There is a limit to the length of the text, if the user enters a long string then returns an Error:</p> <p>"Too long to add as a text"</p>  |

|                     |  |
|---------------------|--|
| <b>Number:</b>      | 2.21   |
| <b>Name:</b>        | Color  |
| <b>Actor(s):</b>    | Registered user  |
| <b>Story:</b>       | <p>1. In the toolbar, there is a "Color" option".</p> <p>2. The user clicks on "Color".</p> <p>3. A color editor pops up.</p> <p>4. User selects an existing component in the map to change the color of that component.</p> |
| <b>Scenario:</b>    | <p>1. The user clicks on "Color".</p> <p>2. User selects an existing component in the map to change the color of that component.</p>   |
| <b>Exceptions :</b> | <p>1. If there's an error applying the color or if the color palette fails to load, an appropriate error message will be shown.</p>  |

|                  |   |
|------------------|---|
| <b>Number:</b>   | 2.22  |
| <b>Name:</b>     | Legend  |
| <b>Actor(s):</b> | Registered user   |
| <b>Story:</b>    | <p>1. In the toolbar, there is a "legend" option.</p> <p>2. User clicks on "legend".</p> <p>3. The user sorts the map by selecting an existing component or shape on the map.</p> |

|                     |   |
|---------------------|---|
|                     | 4. The system saves the user's edits  |
| <b>Scenario:</b>    | <p>1. The user clicks on "legend".</p> <p>2. The user categorizes the components or shapes by selecting them on the map.</p> <p>3. Save edits</p> |
| <b>Exceptions :</b> |   |

|                  |   |
|------------------|---|
| <b>Number:</b>   | 2.23  |
| <b>Name:</b>     | Edit User Profile   |
| <b>Actor(s):</b> | Registered user   |
| <b>Story:</b>    | <p>1. In the user profile interface, there is an "Edit" button, and the user clicks on the "Edit" button.</p> <p>2. Users edit in the pop-up editing window.</p> <p>3. User clicks "Save" button.</p> <p>4. The system pops up a window to confirm the changes to the user: "Do you confirm the changes to the profile?"</p> <p>5. The user clicks "Confirm":<br/>Return to the profile interface, the system will synchronize the edited attributes to the database.</p> <p>6. User clicks "Cancel":</p> |

|                     |  |
|---------------------|--|
|                     | The user clicks "Cancel", the system returns to the profile interface and makes the original content visible, and the system does not modify the database.   |
| <b>Scenario:</b>    | <ol style="list-style-type: none"> <li>1. User clicks the "Edit" button.</li> <li>2. Users edit "Name" in the pop-up editing window.</li> <li>3. User clicks "Save" button.</li> <li>4. The system pops up a window to confirm the changes to the user: "Do you confirm the changes to the profile?"</li> <li>5. User clicks "Confirm".</li> <li>6. Return to the profile page, the modified profile will be displayed.</li> </ol> |
| <b>Exceptions :</b> | Users can only edit their own profile, if they go to another person's homepage profile, they can't click the edit button.  |

|                  |  |
|------------------|--|
| <b>Number:</b>   | 2.24   |
| <b>Name:</b>     | Change Password  |
| <b>Actor(s):</b> | Registered user  |
| <b>Story:</b>    | <ol style="list-style-type: none"> <li>1. There is a "Change Password" button on the User Profile screen</li> <li>2. The user clicks on the "Edit Password" button.</li> <li>3. The system pops up a verification window, requiring the user to enter the current password.</li> </ol> |

|                     |   |
|---------------------|---|
|                     | <p>4. The system compares what the user enters with the corresponding password that exists in the current database</p> <p>4.1. If the verification comparison is successful:</p> <ul style="list-style-type: none"> <li>• The user enters the new password twice in the pop-up window</li> <li>• The user clicks the confirmation button</li> <li>• The system compares the two new passwords and if the comparison is the same then the password is updated in the database</li> </ul> <p>4.2 If the comparison result is wrong it returns an error report "The passwords entered twice are not the same, please check."</p> <p>5. If the comparison validation fails then it returns an error "Verification failed"</p> |
| <b>Scenario:</b>    | <p>1. The user clicks on the "Edit Password" button.</p> <p>2. The user enters the current password in the validation pop-up window.</p> <p>3. If the verification comparison is successful:</p> <p>4. The user enters the new password twice in the pop-up window</p> <p>5. The user clicks the Confirm button</p> <p>6. The system database updates the password in the database</p>  |
| <b>Exceptions :</b> | <p>1. If the new password and the confirmation do not match, the system notifies the user.</p> <p>2. If the new password doesn't meet security criteria (e.g., too short), the system prompts an error.</p>   |

|                  |   |
|------------------|---|
| <b>Number:</b>   | 2.25  |
| <b>Name:</b>     | Manage User Accounts  |
| <b>Actor(s):</b> | Admin User  |
| <b>Story:</b>    | <p>1. There is a "Manage Accounts" button on the administrator's profile page.</p> <p>2. The administrator clicks the "Manage Accounts" button.</p> <p>3. The system returns to the management page and displays the full list of users.</p> <p>4. The administrator can click the Delete button to delete a user.</p> <p>5. The system pops up a confirmation window:<br/>"Do you want to delete this user?"</p> <p>6. The administrator clicks "Confirm", and the system deletes the corresponding user's information in the database.</p> <p>7. The administrator clicks "Cancel" to return to the user list interface, the system does not change the database.</p> |
| <b>Scenario:</b> | <p>1. The administrator clicks the "Manage Accounts" button on the profile screen.</p> <p>2. The administrator clicks the Delete button on the user list screen to delete the user.</p> <p>3. The administrator clicks "Confirm" in the confirmation window, and the system deletes the corresponding user's information in the database.</p> <p>4. Return to the user list</p>   |

|                     |  |
|---------------------|--|
| <b>Exceptions :</b> | <ol style="list-style-type: none"> <li>1. Profiles should respect privacy settings and some information may be hidden based on user preference.</li> <li>2. If the admin tries to delete an account that has critical content or ongoing processes, the system prompts a warning.</li> </ol> |
|---------------------|--|

|                  |   |
|------------------|---|
| <b>Number:</b>   | 2.26  |
| <b>Name:</b>     | Review User-Generated Content   |
| <b>Actor(s):</b> | Admin User  |
| <b>Story:</b>    | <ol style="list-style-type: none"> <li>1. Administrator accesses the Administrator Dashboard.</li> <li>2. Administrators view user-generated content.</li> <li>3. The system presents a list of user-generated content (e.g., maps, discussions, comments) that needs review.</li> <li>4. The administrator selects specific content to view.</li> <li>5. The system displays detailed information of the selected content and user information related to this content.</li> <li>6. The administrator evaluates whether the content complies with community guidelines and policies.</li> <li>7. The administrator takes appropriate action based on the review, such as deleting or marking inappropriate content.</li> <li>8. Administrators may provide comments or feedback to users regarding content.</li> </ol> |
| <b>Scenario:</b> | <ol style="list-style-type: none"> <li>1. The administrator logs in to the administrator dashboard.</li> </ol>  |

|                     |  |
|---------------------|--|
|                     | <p>2. View a list of recently uploaded maps for review.</p> <p>3. The administrator selects the map titled "World" for review.</p> <p>4. Making sure it meets the guidelines.</p> <p>5. The administrator approves the release.</p>                                      |
| <b>Exceptions :</b> | <p>1. If administrators see content that violates platform policies, they should be able to take appropriate action, such as removing or marking the content and notifying users of the violation.</p> <p>2. The system should have a mechanism for content appeals.</p> |

|                  |   |
|------------------|---|
| <b>Number:</b>   | 2.27  |
| <b>Name:</b>     | Default Map with Custom Settings  |
| <b>Actor(s):</b> | User  |
| <b>Story:</b>    | <p>1. User selects the default map type.</p> <p>2. User sets colors and text for the map elements.</p> <p>3. User adds and removes legend entries with corresponding colors and labels.</p> <p>4. User saves the map configuration.</p>   |
| <b>Scenario:</b> | <p>1. User selects the default map.</p> <p>2. User sets the color blue for Region A, green for Region B, and adds texts accordingly.</p> <p>3. User clicks the plus sign to add more legend.</p> <p>4. User creates a legend with labels for Region A and Region B.</p> <p>5. User deletes a legend entry by clicking the minus sign.</p> <p>6. User saves the map configuration.</p> |

|                        |  |
|------------------------|--|
| <b>Exceptions</b><br>: | 1. User tries to save without setting coordinates or layer attributes. |
|------------------------|--|

|                        |  |
|------------------------|--|
| <b>Number:</b>         | 2.28   |
| <b>Name:</b>           | Heat Map with Custom Settings  |
| <b>Actor(s):</b>       | User   |
| <b>Story:</b>          | <ol style="list-style-type: none"> <li>1. User selects the heat map type.</li> <li>2. User inputs coordinates or clicks on the map to set coordinates.</li> <li>3. User assigns values and labels to the heat points.</li> <li>4. User customizes colors for the map.</li> <li>5. User saves the heat map configuration.</li> </ol>  |
| <b>Scenario:</b>       | <ol style="list-style-type: none"> <li>1. User enters coordinates (50, 30) and assigns it a number of 5000, labeling it as 'High'.</li> <li>2. User selects coordinates (40, 20) via map click and assigns it a number of 3000, labeling it as 'Medium'.</li> <li>3. User deletes a data point by clicking the delete button.</li> <li>4. User chooses different colors for map elements.</li> <li>5. User saves the configured heat map.</li> </ol> |
| <b>Exceptions</b><br>: | <ol style="list-style-type: none"> <li>1. Addressing errors in coordinate input.</li> <li>2. User tries to save without entering coordinates or values.</li> </ol>   |

|                  |  |
|------------------|--|
| <b>Number:</b>   | 2.29   |
| <b>Name:</b>     | Point Map with Custom Settings   |
| <b>Actor(s):</b> | User   |
| <b>Story:</b>    | <ol style="list-style-type: none"> <li>1. User selects the point map type.</li> <li>2. User sets coordinates manually or through map clicks.</li> <li>3. User adds meaningful labels to each point.</li> <li>4. User customizes colors for the map.</li> </ol> |

|                     |  |
|---------------------|--|
|                     | 5. User saves the point map configuration.   |
| <b>Scenario:</b>    | <ol style="list-style-type: none"> <li>1. User selects the point map.</li> <li>2. User manually inputs coordinates (60, 40) and labels it as 'Point A'.</li> <li>3. User clicks on the map to set coordinates (70, 50) and labels it as 'Point B'.</li> <li>4. User deletes a point by clicking the delete button.</li> <li>5. User chooses different colors for map elements.</li> <li>6. User saves the configured point map.</li> </ol> |
| <b>Exceptions :</b> | <ol style="list-style-type: none"> <li>1. User tries to save without setting coordinates or labels.</li> <li>2. Handling errors in coordinate input or label addition.</li> </ol>  |

|                  |  |
|------------------|--|
| <b>Number:</b>   | 2.30   |
| <b>Name:</b>     | Bubble Map with Custom Settings  |
| <b>Actor(s):</b> | User   |
| <b>Story:</b>    | <ol style="list-style-type: none"> <li>1. User selects the bubble map type.</li> <li>2. User inputs coordinates or clicks on the map to set coordinates.</li> <li>3. User specifies bubble color, radius, and popup data.</li> <li>4. User clicks on a bubble to view detailed information.</li> <li>5. User customizes colors for the map.</li> <li>6. User saves the bubble map configuration.</li> </ol>  |
| <b>Scenario:</b> | <ol style="list-style-type: none"> <li>1. User selects the bubble map.</li> <li>2. User inputs coordinates (45, 35) for a blue bubble with a radius of 100000 and set a popup data.</li> <li>3. User sets coordinates (55, 25) for a red bubble with a radius of 500000 and set a popup data.</li> <li>4. User clicks on a bubble to view its information.</li> <li>5. User deletes a bubble by clicking the delete button.</li> <li>6. User chooses different colors for map elements.</li> <li>7. User saves the configuration.</li> </ol> |

|                        |   |
|------------------------|---|
| <b>Exceptions</b><br>: | <ol style="list-style-type: none"> <li>1. User tries to save without setting coordinates or attributes.</li> <li>2. Handling errors in coordinate input or bubble configuration.</li> </ol> |
|------------------------|---|

|                        |  |
|------------------------|--|
| <b>Number:</b>         | 2.31   |
| <b>Name:</b>           | Thematic Map with Custom Settings  |
| <b>Actor(s):</b>       | User   |
| <b>Story:</b>          | <ol style="list-style-type: none"> <li>1. User selects the thematic map type.</li> <li>2. User clicks within map layers to set coordinates and sets the value.</li> <li>3. User clicks put button.</li> <li>4. User adjusts legend properties (value, color, opacity).</li> <li>5. User hides or displays specific legends on the map.</li> <li>6. User saves the thematic map configuration.</li> </ol>   |
| <b>Scenario:</b>       | <ol style="list-style-type: none"> <li>1. User selects the thematic map.</li> <li>2. User clicks within map layers to set coordinates and sets the value.</li> <li>3. User clicks put button.</li> <li>4. User defines values for color legend</li> <li>5. User changes the color of a specific legend.</li> <li>6. User adjusts the opacity of the legend and the map.</li> <li>7. User hides a legend to simplify the map view.</li> <li>8. User clicks update legends button.</li> <li>9. User deletes a layer by clicking the delete button.</li> <li>10. User saves the configuration.</li> </ol> |
| <b>Exceptions</b><br>: | <ol style="list-style-type: none"> <li>1. User tries to save without setting coordinates or values.</li> <li>2. Handling errors in legend setup or configuration.</li> </ol>   |

|                  |                                     |
|------------------|-------------------------------------|
| <b>Number:</b>   | 2.32                                |
| <b>Name:</b>     | Choropleth Map with Custom Settings |
| <b>Actor(s):</b> | User                                |

|                     |  |
|---------------------|--|
| <b>Story:</b>       | <ol style="list-style-type: none"> <li>1. User selects the choropleth map type.</li> <li>2. User clicks within map layers to set coordinates.</li> <li>3. User defines color, ID, and statistic.</li> <li>4. User clicks put button.</li> <li>5. User clicks on a layer to view detailed information.</li> <li>6. User saves the choropleth map configuration.</li> </ol>  |
| <b>Scenario:</b>    | <ol style="list-style-type: none"> <li>1. User selects the choropleth map.</li> <li>2. User clicks within map layers to set coordinates.</li> <li>3. User defines color, ID, and statistic.</li> <li>4. User clicks put button.</li> <li>5. User clicks on a layer to view its information.</li> <li>6. User deletes a layer by clicking the delete button.</li> <li>7. User saves the configuration.</li> </ol> |
| <b>Exceptions :</b> | <ol style="list-style-type: none"> <li>1. User tries to save without setting coordinates or layer attributes.</li> <li>2. Handling errors in choropleth map layer setup or configuration.</li> </ol>   |

### 3. Use Interface Model

#### 3.1 User Interface View Listing

| # | Name                        | Description   |
|---|-----------------------------|---|
| 1 | Main Page Screen            | The first screen displayed after opening this website.  |
| 2 | Registration/Login Screen   | <p>Display registration and login forms.</p> <ul style="list-style-type: none"> <li>1. User login form with email and password fields.</li> <li>2. Option to register a new account.</li> <li>3. Option to reset forgotten password.</li> </ul>   |
| 3 | User Dashboard Screen       | <p>User dashboard showing various options.</p> <ul style="list-style-type: none"> <li>1. Option to create a new map.</li> <li>2. Options for editing existing maps.</li> <li>3. Option to search maps based on criteria.</li> <li>4. Options to classify and sort maps based on properties.</li> <li>5. Option to view your own profile information.</li> <li>6. Small pictures and information of the map.</li> <li>7. Option to view the map.</li> <li>8. Option to delete the map.</li> <li>9. Display the map you participated in.</li> </ul> |
| 4 | User Profile Screen         | <p>Display the user's profile information.</p> <ul style="list-style-type: none"> <li>1. Option to edit profile information.</li> <li>2. Option to change password.</li> </ul>  |
| 5 | Map View Screen             | <p>Displays a detailed view of the map, including map graphics and information.</p> <ul style="list-style-type: none"> <li>1. Map viewer with zoom and pan controls.</li> <li>2. Display discussions related to the map.</li> </ul>   |
| 6 | Map Discussion Forum Screen | Users participate in map discussions.   |

|   |                           |   |
|---|---------------------------|---|
| 7 | Map Edit Screen           | <p>Displays the map edit page, a form for editing an existing map with custom options.</p> <ol style="list-style-type: none"> <li>1. Option to map fork is used to create a new map based on an existing map.</li> <li>2. Option to import map files into the system.</li> <li>3. Options for custom maps.</li> <li>4. Option to export the map in various formats (e.g. PNG, JPG, JSON).</li> <li>5. Options for sharing maps.</li> <li>6. Options for public/private maps.</li> </ol> |
| 8 | Map Customize Tool Screen | <p>Displays map customization tools, including text, colors, legends, and properties additions.</p> <ol style="list-style-type: none"> <li>1. Tools for adding and editing text on the map.</li> <li>2. Tool for color customization to modify area color.</li> <li>3. Tools for customizing map legends.</li> </ol>  |
| 9 | Admin Dashboard Screen    | <p>Admin dashboard for managing user accounts and content.</p> <ol style="list-style-type: none"> <li>1. Manage user accounts.</li> <li>2. Check for user-generated content.</li> </ol>   |

## 3.2 User Interface Diagrams

Diagrams in planning stage can be found:

<https://www.figma.com/file/09twyRxxG0J32sf1DNv55X/Green-team-draft?type=design&node-id=1602-2849&mode=design&t=dRUSC308bpjgdmyp-0>

### 1. Main Page Screen



# Welcome to CUSTOMAP, Discover and Customize Various Map!

Create custom maps according to your needs. With CUSTOMAP, you can adjust properties and add extra details easily. Try it out for a simple and enjoyable mapping experience!



## Why Choose CUSTOMAP?



### Map Graphic Creation

Create map by uploading SHP/DBF (ZIP), GeoJSON, KML files, or forking existing maps.



### Map Graphic Selection

Select various map type you want to create! We provide Default, Heat, Point, Bubble, Thematic, and Choropleth map.



### Map Graphic Editing

Edit map according to your needs. You can rename region, specify color of it or add legend.



### Map Graphics Exporting

Export your map in PNG, JPG and JSON formats.



### Map Search

Search the map that suits your needs.



### Community Interactions

Discuss maps with others, give feedback, and collaborate through map sharing.

Interact with maps with CUSTOMAP. Create, share, and explore various custom maps!

## 2. Registration/Login Screen

### 2-1 Register page

 **CUSTOMAP**

Home MyPage Sign in

# WELCOME BACK!

Login to your account

Email

Password

[Forgot Password?](#)

[Sign in](#) [Sign up](#)



 **CUSTOMAP**

Home MyPage Sign in

 **Make New Account**

ID

Password

Name

Email

Phone

[Create Account](#)

## 2-2 Forgot password page 1

CUSTOMAP

Home MyPage Sign in

Forget password

ID

Email

Phone

Veritification

WARNING  
Input information does not match the system  
Verification Failure

Try again

## 2-3 Forgot password page 2

CUSTOMAP

Home MyPage Sign in

Forget password

New Password

New Password again

Change Password

WARNING  
2 new passwords doesn't match  
Please try again

Try again

### 3. User Dashboard Screen

The screenshot shows the CUSTOMAP user dashboard. On the left, a sidebar displays the user's profile picture (a green bear icon) and name "Hello JuyeonNam". Below the name are links for "Dashboard", "My Profile", "Create Map", and "Search Map". The main area is titled "Dashboard" and contains a section titled "Maps you have participated in". It lists six maps with preview images and edit/delete buttons:

- Heat\_Geojson
- Point\_KML
- Bubble\_SHP
- Default\_NA
- Choropleth\_World
- Thematic

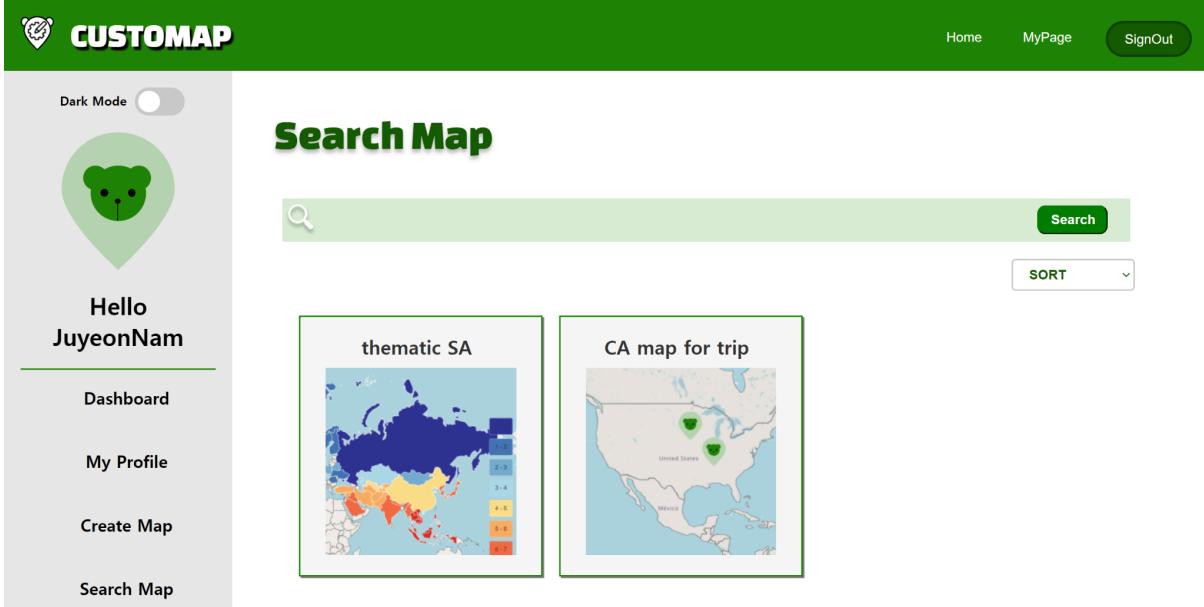
A "SORT" button is located in the top right corner of this section.

#### 3-1. User Dashboard Screen - Create Map

The screenshot shows the "Create Map" screen. The left sidebar is identical to the dashboard, showing "Hello JuyeonNam" and navigation links. The main area is titled "Create Map" and includes the following elements:

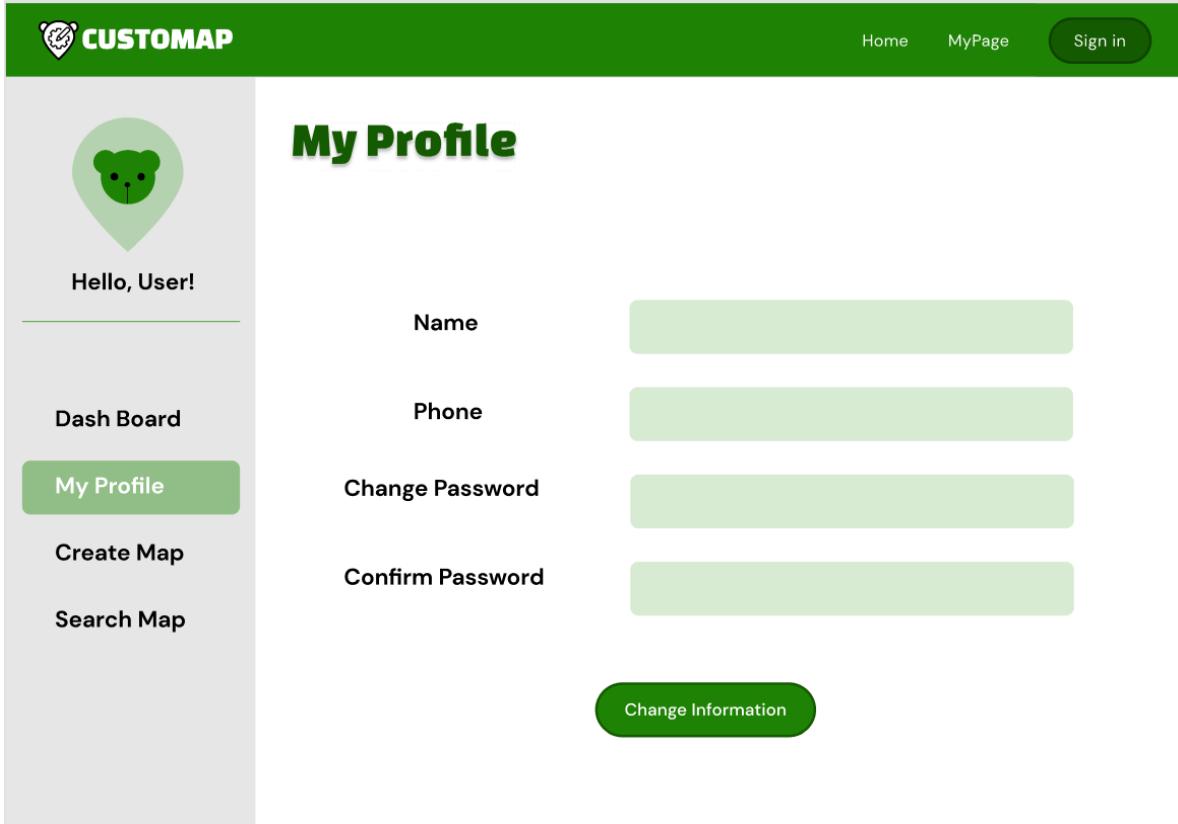
- A header with "Select Map Type", "Fork Map", and "Upload File" buttons, and a note "Available on SHP/DBF, GeoJSON, KML".
- A "Map Name:" input field with "Map Name" placeholder text.
- A large map canvas showing a coastal area with roads, water bodies, and green spaces. Labeled locations include "Port Jefferson", "Setauket", "Stony Brook", "Head of the Harbor", "Stony Brook University", "Terryville", and "South Setauket Park".
- A "Map Description:" input field with "Description" placeholder text.
- Buttons for "Cancel" and "Create Map" at the bottom right.

### 3- 2. User Dashboard Screen - Search Map



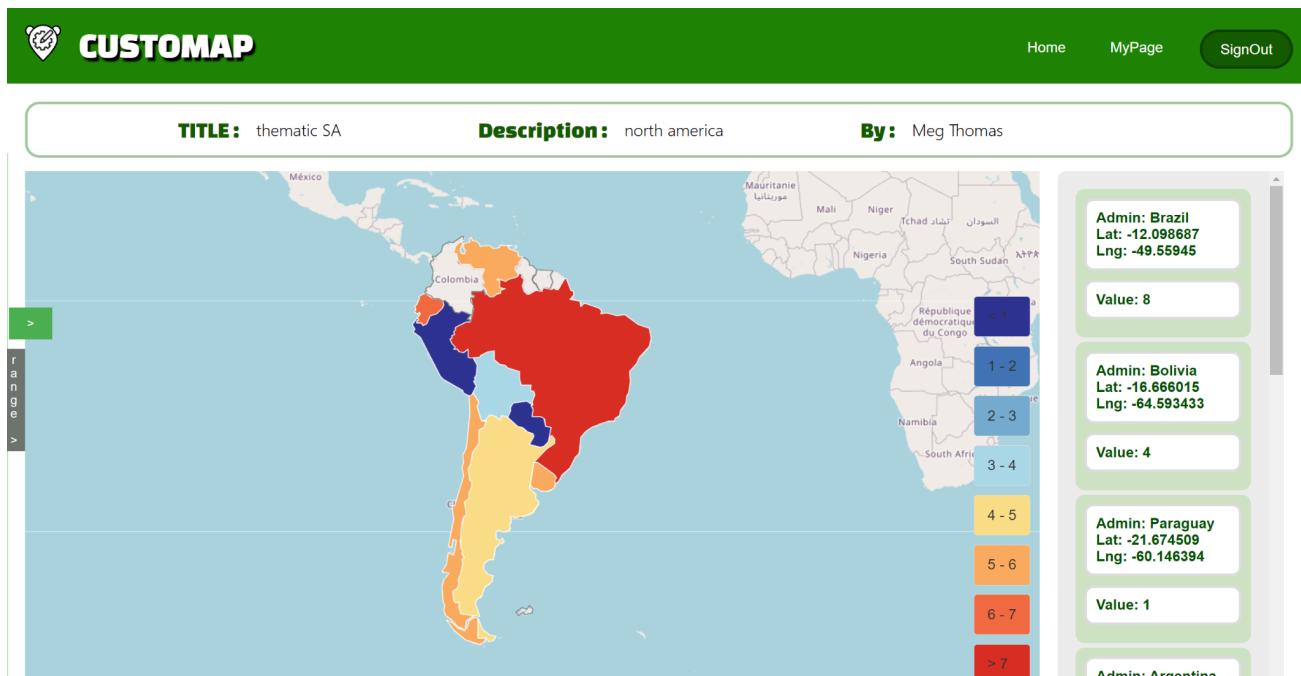
The screenshot shows the 'Search Map' section of the CUSTOMAP user dashboard. At the top, there is a green header bar with the CUSTOMAP logo, a 'Dark Mode' toggle switch, and navigation links for 'Home', 'MyPage', and 'SignOut'. On the left, a sidebar displays a profile picture of a green bear inside a green location pin, with the text 'Hello JuyeonNam'. Below this are menu items: 'Dashboard', 'My Profile', 'Create Map', and 'Search Map', with 'Search Map' highlighted. The main content area features a search bar with a magnifying glass icon and a 'Search' button. To the right of the search bar is a 'SORT' dropdown menu. Two map thumbnails are shown: 'thematic SA' (a world map with color-coded regions) and 'CA map for trip' (a map of North America with specific locations marked). A green 'Search' button is located at the bottom right of the main content area.

### 4. User Profile Screen



The screenshot shows the 'My Profile' screen of the CUSTOMAP user profile. At the top, there is a green header bar with the CUSTOMAP logo, a 'Sign in' button, and navigation links for 'Home' and 'MyPage'. On the left, a sidebar displays a profile picture of a green bear inside a green location pin, with the text 'Hello, User!'. Below this are menu items: 'Dash Board', 'My Profile' (which is highlighted), 'Create Map', and 'Search Map'. The main content area features a large title 'My Profile' in green. It includes input fields for 'Name' (a light green rectangular field), 'Phone' (a light green rectangular field), 'Change Password' (a light green rectangular field), and 'Confirm Password' (a light green rectangular field). At the bottom right is a green 'Change Information' button.

## 5. Map View Screen



## 6. Map Discussion Forum Screen

The screenshot shows a map of Eurasia with a comment overlay on the left. The comment area includes a text input field ('post comment here'), a user name ('JuyeonNam'), a message ('nice map!'), and a map background selection section ('Map Background: [dropdown] Lat: 56.57439935811001 Lng: -67.49987387965282'). Below the map, there is a text input field ('Type a message...') and a 'Send' button. The map itself shows various countries with their names in multiple languages.

## 7. Map Edit Screen

The screenshot shows the CUSTOMAP interface for editing a map. At the top, there's a green header bar with the CUSTOMAP logo, a dark mode toggle switch, and navigation links for Home, MyPage, and SignOut. On the left, a sidebar displays a user profile picture of a green bear, the name "Hello JuyeonNam", and menu options: Dashboard, My Profile, Create Map, and Search Map. The main area is titled "Edit Map" and contains a world map. A heatmap overlay is applied to the map, with a legend indicating "Heat\_Geojson". The map has labels in multiple languages. Below the map, there are input fields for "Map Name" (Heat\_Geojson) and "Map Description" (This is heat map), and buttons for "Cancel" and "Edit Map".

## 8. Map Customize Tool Screen

### 8-1. Map Customize Tool Screen - Default Map

The screenshot shows the CUSTOMAP Map Customize Tool screen. The top features a green header bar with the CUSTOMAP logo, Home, MyPage, and SignOut links. The main area is titled "Customize Tools" and includes a toolbar with "Font" dropdown, "Pick a Color" button, color swatches, "Applied Text Change" button, and "SAVE" button. To the right of the toolbar is a "Legend" panel with a placeholder for "Legend Label". The central part of the screen displays a world map with a light gray background. A small green button with a right arrow is located in the top-left corner of the map area.

## 8-2. Map Customize Tool Screen - Heat Map

The screenshot shows the CUSTOMAP website interface for creating a heat map. At the top, there's a green header bar with the CUSTOMAP logo, navigation links for Home, MyPage, and SignOut, and a search bar. Below the header is a section titled "Customize Tools" with a sub-instruction: "Enter latitude and longitude directly, or click on the map to set them. Enter a number between 500 and 5000 and press ADD to display it on the map." A form below the instruction includes fields for "Map Background" (with a preview icon), "Lat" (green input field), "Lng" (green input field), "Number" (green input field), "Label" (green input field), and buttons for "Add", "Edit", "Delete", and "SAVE". The main area features a world map where users can click to add data points. The map includes labels for countries and cities in multiple languages. A copyright notice at the bottom right of the map area reads "Leaflet | © OpenStreetMap contributors".

## 8-3. Map Customize Tool Screen - Point Map

This screenshot displays the CUSTOMAP website's point map customization tool. The layout is identical to the heat map screen, with a green header, "Customize Tools" section, and a world map for adding markers. The map is labeled in various languages, and the same copyright notice for Leaflet and OpenStreetMap is present. The "Point Map" screen also includes the "Map Background" selection and coordinate entry fields.

## 8-4. Map Customize Tool Screen - Bubble Map

 CUSTOMAP

Home MyPage SignOut

### Customize Tools

Enter latitude and longitude directly, or click on the map to set them. Set Color and Enter the radius(more than 10000). Press ADD to display it on the map.

Map Background:  Lat:  Lng:  Color:  Radius:  Popup:  Add



Leaflet | © OpenStreetMap contributors

## 8-5. Map Customize Tool Screen - Thematic Map

**CUSTOMAP**

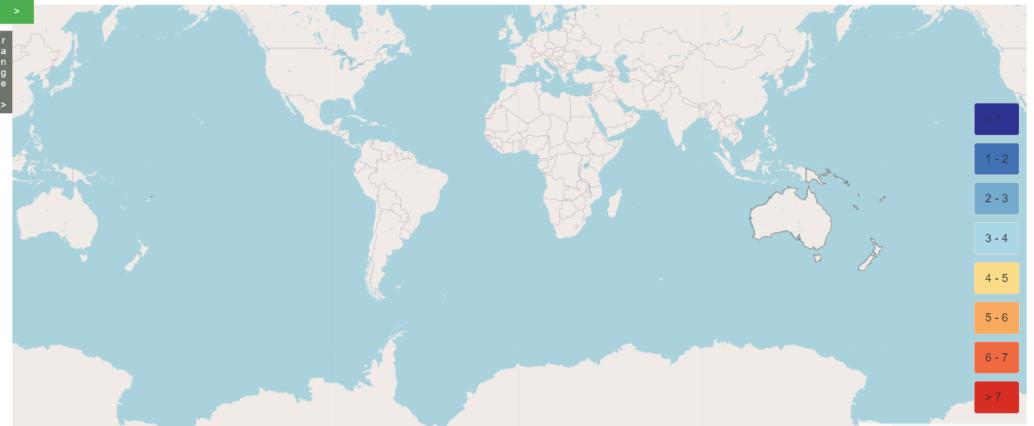
Home MyPage SignOut

### Customize Tools

Click on the map to set area. Set Value number OR Open "range" on left side and put display to set Color and Opacity. Press PUT to change the map.

Selected Area || Admin : ; Lat : ; Lng :

Value :  Put    SAVE

> 

< range >

Value: -inf Color:  Opacity: 1

Value: 1 Color:  Opacity: 1

Value: 2 Color:  Opacity: 1

Value: 3 Color:  Opacity: 1

Value: 4 Color:  Opacity: 1

Value: 5 Color:  Opacity: 1

Value: 6 Color:  Opacity: 1

Value: 7 Color:  Opacity: 1

display

on left side and put display to set Color and Opacity. Press PUT to change the map.

Home MyPage SignOut

### Customize Tools

Value: -inf Color:  Opacity: 1

Value: 1 Color:  Opacity: 1

Value: 2 Color:  Opacity: 1

Value: 3 Color:  Opacity: 1

Value: 4 Color:  Opacity: 1

Value: 5 Color:  Opacity: 1

Value: 6 Color:  Opacity: 1

Value: 7 Color:  Opacity: 1

display

Update Legends

< range >

< 1 Color:  Opacity: 1

1 - 2 Color:  Opacity: 1

2 - 3 Color:  Opacity: 1

3 - 4 Color:  Opacity: 1

> 7 Color:  Opacity: 1

on left side and put display to set Color and Opacity. Press PUT to change the map.

Home MyPage SignOut

< range >

< 1 Color:  Opacity: 1

1 - 2 Color:  Opacity: 1

2 - 3 Color:  Opacity: 1

3 - 4 Color:  Opacity: 1

> 7 Color:  Opacity: 1

on left side and put display to set Color and Opacity. Press PUT to change the map.

Home MyPage SignOut

## 8-6. Map Customize Tool Screen - Choropleth Map

 **CUSTOMMAP**

Home MyPage SignOut

### Customize Tools

Click on the map to set area. Set Color and enter ID and Statistic. Press PUT to change the map.

Selected Area || Admin : ; Lat : ; Lng : ;

Color :  ID :  Statistic :  Put    SAVE



## 9-1. Admin Dashboard Screen

### 9-1. Admin Dashboard Screen - User List

**Hello, Admin**

User List

Map List

## Admin Dashboard

|  |   | SORT |
|--|---|------|
|  | Meg Thomas<br>meg.thomas@gmail.com      |      |
|  | Shihao Wen<br>shihao.wen@stonybrook.edu |      |

**Map List**

Title: Europe Bubble  
Description: Each bubble represents the population in a specific location, and the size of the bubble corresponds to the population size.

Title: Oceania Point  
Description: label is the meaning of the coordinates of this point.

### 9-2. Admin Dashboard Screen - Map List

**Hello, Admin**

User List

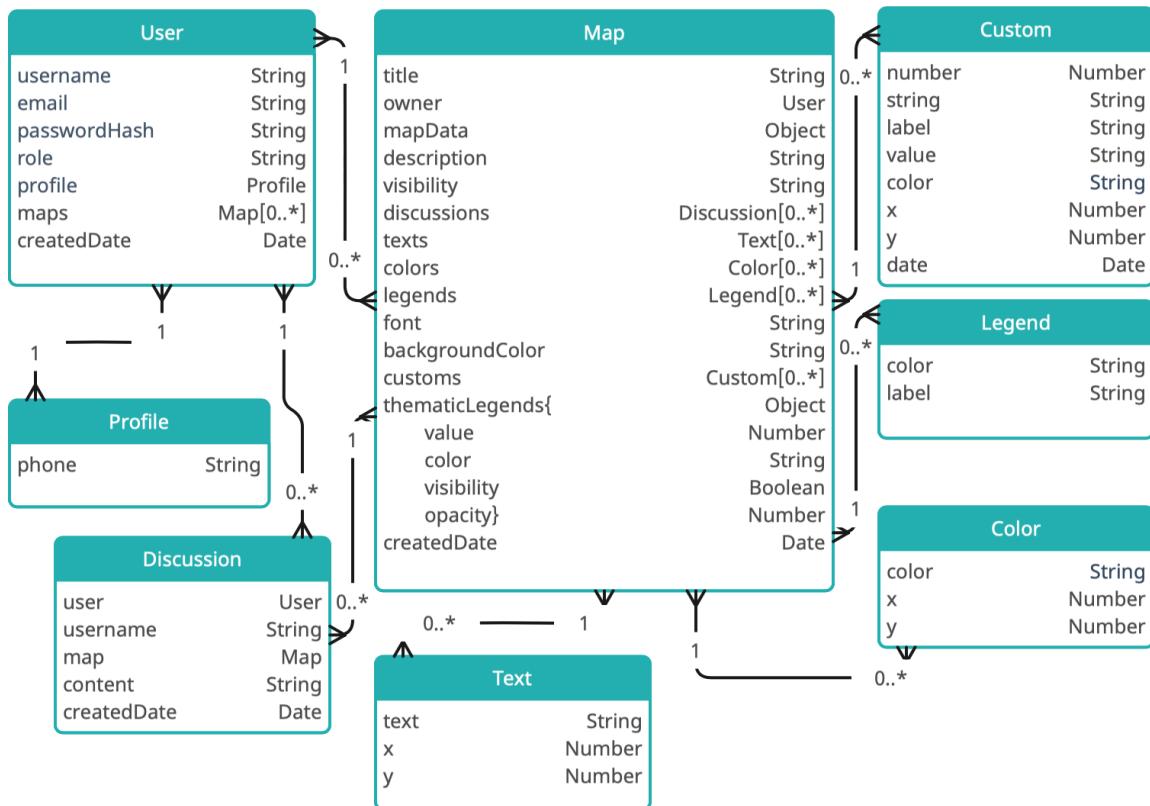
Map List

## Search Map

|  |  | Search |
|--|--|--------|
|  |  |        |
|  |  |        |

# 4 Data Model

## 4.1 Persistent Data Model



## 4.2 Dictionary

- User
  - username: User's username (String, required).
  - email: User's email address (String, required, unique).
  - passwordHash: User's password, hashed and salted (String, required).
  - role: User's role, either "user" or "admin" (String, required).
  - profile: Profile References (Profile).
  - maps: Array of Map References, user's created maps (Map[0..\*]).
  - createdDate: (Date, auto-generated).

- Profile
  - phone: User's phone number (String).
- Map
  - title: Title of the map (String, required).
  - owner: User Reference (Object, required).
  - mData: Map file (User, required).
  - description: (String).
  - visibility: Map's visibility, either 'public' or 'private' (String, default: 'private').
  - discussions: Array of Discussion Reference (Discussion[0.. \*]).
  - texts: Array of Text Reference (Text[0.. \*]).
  - colors: Array of Color Reference (Color[0.. \*]).
  - legends: (Legend[0.. \*]).
  - font: (String).
  - backgroundColor: (String).
  - customs:(Custom[0.. \*]).
  - thematicLegends: (Object[{
    - value: { type: Number },
    - color: { type: String },
    - visibility: { type: Boolean },
    - opacity: { type: Number },
 }]).
  - createdDate: (Date, auto-generated).
- Discussion
  - user: User Reference (User, required).
  - username: (String).
  - map: Map Reference (Map, required).
  - content: (String).
  - createdDate: (Date, auto-generated).

- Text
  - text: (String, required)
  - x: Map latitude (Number, required)
  - y: Map longitude (Number, required)
- Color
  - color: (String, required)
  - x: Map latitude (Number, required)
  - y: Map longitude (Number, required)
- Legend
  - color: (String, required)
  - label: (String, required)
- Custom
  - number: (Number)
  - string: (String)
  - label: (String)
  - value: (String)
  - color: (String)
  - x: Map latitude (Number, required)
  - y: Map longitude (Number, required)
  - date: (Date, auto-generated).

## 5 Software Model

### 5.1 Similar Problems:

#### **Front-end:**

1. Create user interface: React
2. Map rendering and creation of map components: Leaflet
3. Make HTTP requests from the front-end to the back-end: axios
4. Process file upload: toGeoJSON, shapefile, jszip
5. Process geospatial data formats: toGeoJSON, shapefile

#### **Back-end:**

1. Database: MongoDB and Mongoose
2. Hash and salt passwords: bcrypt
3. Receive HTTP requests from the back-end to the front-end: cors
4. Create RESTful API: Node.js and Express.js
5. Handle user login and session management: Express.js
6. Middleware for processing requests and responses: Express.js

#### **Version Control:**

1. Version control and team collaboration: Git and GitHub

### 5.2 Complete Technology Set:

#### **Front-End:**

1. React: The primary library for building user interfaces and managing front-end components.
2. leaflet: Used for map rendering and creating map components.
3. axios: Promise-based HTTP client, used to make HTTP requests from the front end to the back end.

4. `toGeoJSON`: Library for converting various geospatial data formats (e.g. Shapefile, KML) to GeoJSON.
5. `shapefile`: Library for parsing shapefiles.
6. `jszip`: Library for handling ZIP files, for handling shapefiles as ZIP archives.

#### **Back-End:**

1. `Node.js`: For server-side development.
2. `Express.js`: Middleware for creating RESTful APIs and handling requests and responses.
3. `MongoDB`: NoSQL database for storing and managing user and map data.
4. `Mongoose`: Used to use MongoDB.
5. `GridFS`: MongoDB's file storage system, used to store uploaded map files.
6. `bcrypt`: Hashes and salts user passwords.
7. `cors`: Middleware for handling cross-domain HTTP requests.
8. `express-session`: Middleware for handling user sessions and authentication.

#### **Version Control:**

1. `Git`: Version control for tracking changes to your code base.
2. `GitHub`: Platform for hosting Git repositories, team collaboration, issue tracking, and code reviews.

## **5.3 Training Verification:**

#### **Member name: Shihao Wen**

1. Training resources: CSE316
2. Completion date: Spring of 2022
3. Project: Build a web application similar to stackoverflow.com.
4. Technologies used: React, axios, Node.js, Express.js, MongoDB, Mongoose, bcrypt, cors, express-session

5. Project link:

<https://github.com/CSE-316-Software-Development/final-project-WENSHIHAOO>

1. Training resources: CSE416

2. Completion date: Fall of 2023

3. Project: Building a map web application (front-end only)

4. Technologies used: leaflet, toGeoJSON, shapefile, jszip

5. Project link: <https://github.com/CUSTOMAP-416/CUSTOMAP>

1. Training resources: MongoDB official documentation

2. Completion date: Fall of 2023

3. Project: None

4. Technology used: leaflet, toGeoJSON, shapefile, jszip

5. Project link: None

**Member name: Seolhee Yun**

1. Training resources: CSE316

2. Completion date: Spring of 2023

3. Project: Build a web application about Note and class management site

4. Technologies used: React, axios, Node.js, Express.js, MySQL, cors, express-session

5. Project link: [https://github.com/sinaetown/316\\_final\\_project](https://github.com/sinaetown/316_final_project)

1. Training resources: CSE416

2. Completion date: Fall of 2023

3. Project: Building a map web application (front-end only)

4. Technologies used: leaflet, toGeoJSON, shapefile, jszip

5. Project link: <https://github.com/CUSTOMAP-416/CUSTOMAP>

1. Training resources: MongoDB official documentation

2. Completion date: Fall of 2023

3. Project: None

4. Technology used: GridFS

5. Project link: None

**Member name: Juyeon Nam**

1. Training resources: CSE316
2. Completion date: Spring of 2023
3. Project: Build a web application about Note and class management site
4. Technologies used: React, Node.js, Express.js, MySQL, cors, express-session
5. Project link: [https://github.com/JuyeeMyeong/web\\_project\\_cse316](https://github.com/JuyeeMyeong/web_project_cse316)

1. Training resources: CSE416
2. Completion date: Fall of 2023
3. Project: Building a map web application (front-end only)
4. Technologies used: leaflet, toGeoJSON, shapefile, jszip
5. Project link: <https://github.com/CUSTOMAP-416/CUSTOMAP>

1. Training resources: MongoDB official documentation
2. Completion date: Fall of 2023
3. Project: None
4. Technology used: leaflet, toGeoJSON, shapefile, jszip
5. Project link: None

**Member name: Weikang Yang**

1. Training resources: CSE316
2. Completion date: Spring of 2023
3. Project: Build a web application about Note and class management site
4. Technologies used: React, Node.js, Express.js, MySQL, cors, express-session
5. Project link:  
<https://github.com/CSE-316-Software-Development/final-project-zuxiang-weikang>

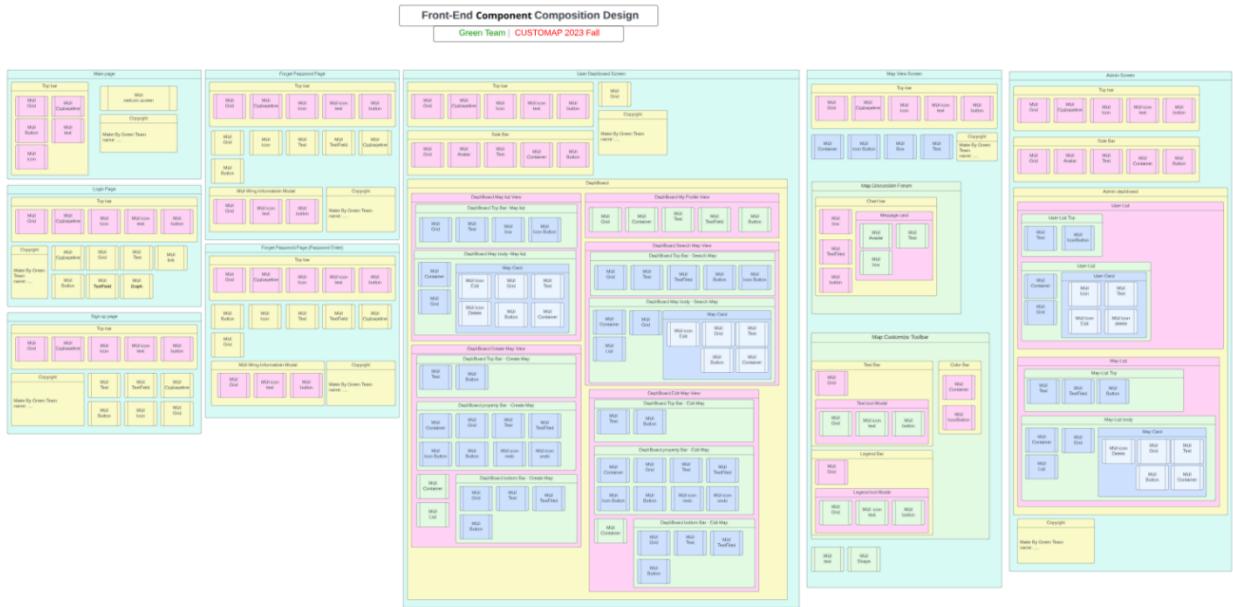
1. Training resources: CSE416
2. Completion date: Fall of 2023
3. Project: Building a map web application (front-end only)

4. Technologies used: leaflet, toGeoJSON, shapefile, jszip
5. Project link: <https://github.com/CUSTOMAP-416/CUSTOMAP>

1. Training resources: MongoDB official documentation
2. Completion date: Fall of 2023
3. Project: None
4. Technology used: leaflet, toGeoJSON, shapefile, jszip
5. Project link: None

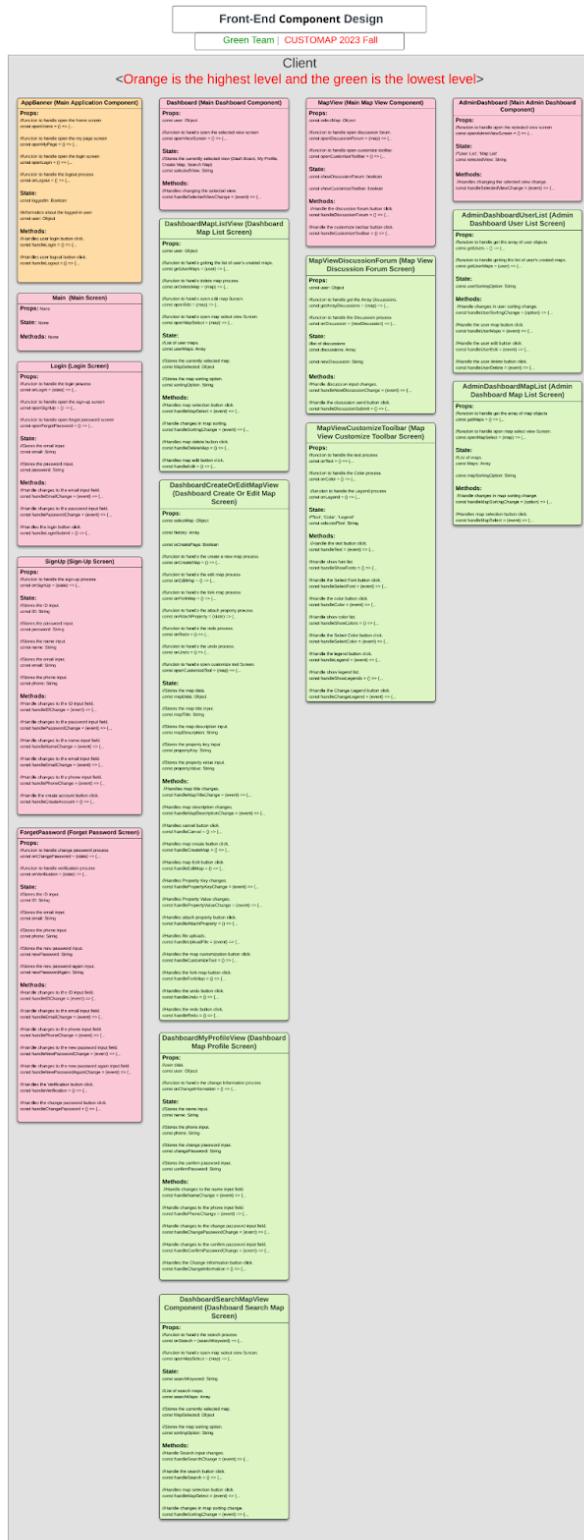
## 5.4 Front-End Component Composition :

[https://lucid.app/lucidchart/fe4dfa2f-e6ef-4030-85e6-8d80230533a3/edit?view\\_items=6\\_KFc5hs-VYZ&invitationId=inv\\_13211ccb-5524-4571-94e7-03f72c9bd844](https://lucid.app/lucidchart/fe4dfa2f-e6ef-4030-85e6-8d80230533a3/edit?view_items=6_KFc5hs-VYZ&invitationId=inv_13211ccb-5524-4571-94e7-03f72c9bd844)



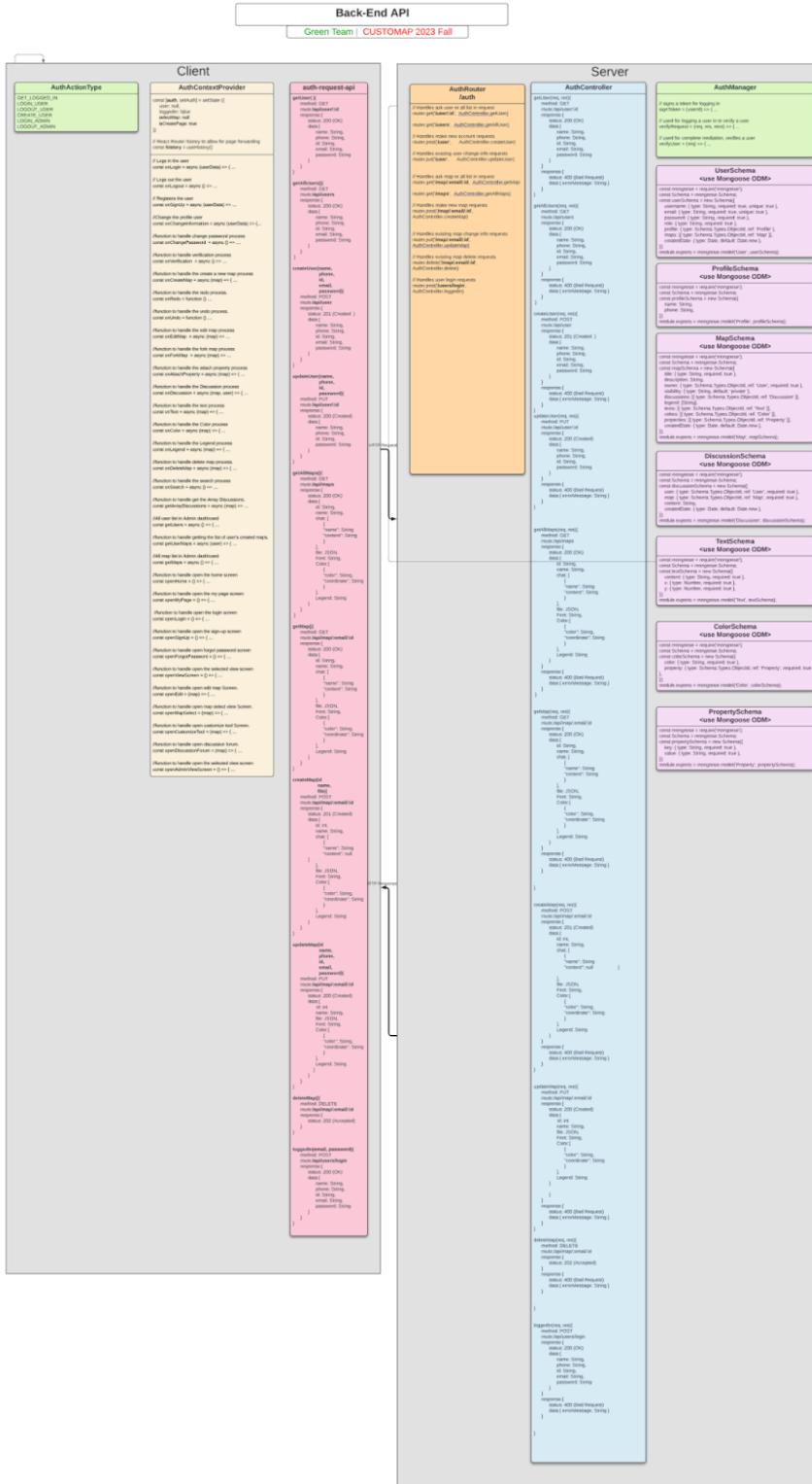
# 5.5 Front-End Component Design

[https://lucid.app/lucidchart/2642d963-fd9a-4df6-bab9-ca924dd676de/edit?viewport\\_loc=-203%2C-809%2C3136%2C1470%2C0\\_0&invitationId=inv\\_8549d50e-b4e6-4292-a784-9b662cf97971](https://lucid.app/lucidchart/2642d963-fd9a-4df6-bab9-ca924dd676de/edit?viewport_loc=-203%2C-809%2C3136%2C1470%2C0_0&invitationId=inv_8549d50e-b4e6-4292-a784-9b662cf97971)



## 5.6 Back-End API

[https://lucid.app/lucidchart/47d880ff-c5d1-459d-bf1b-aecbb41d3fac/edit?viewport\\_loc=270%2C-3086%2C2584%2C1211%2C0&invitationId=inv\\_ca2ddfb4-cd33-4bcc-843c-a344c462cd85](https://lucid.app/lucidchart/47d880ff-c5d1-459d-bf1b-aecbb41d3fac/edit?viewport_loc=270%2C-3086%2C2584%2C1211%2C0&invitationId=inv_ca2ddfb4-cd33-4bcc-843c-a344c462cd85)



## 6. Deployment

We deployed both our client and server side on Heroku, which is a cloud platform as a service (PaaS). The reason why we chose Heroku for our deployment is it provides easy setup for a MERN stack deployment.

We use MongoDB for our database. We chose MongoDB because it provides high performance (speed) and is scalable and flexible.

Our URL for deployed application is

<https://customap416client-3b33f67d5c86.herokuapp.com/>

## 7. Code

More detail for our code can be found:

<https://github.com/CUSTOMAP-416/CUSTOMAP>

### 7.1 File Structure

- client: code for client side
  - src
    - assets\_img: store img files that used in our website.
    - components: store .js file for all react component
    - mapTypes: store .jsx file for mapTypes(Default, Heat, Point, Bubble, Thematic, Choropleth)
    - styles: store all css files.
    - auth\_store
      - auth\_store\_store\_request\_api
        - index.js: all functions for request api

- index.js: files for types of updates to our auth and store state that can be processed
- server: code for server side
  - fork\_map: geojson files to provide users when they forked map
  - models: store schemas in our DB
  - auth\_store\_controller.js:
  - auth\_store\_router.js: store all routers.
  - auth.js: files for authorization
  - index.js: create server connection and setup the middleware

## 7.2 Code Convention

We use Camel Case for file naming and all codes except css file and Kebab Case for css file. We expect these conventions will provide organized and understandable code by offering consistency when writing code.

We follow REST API, which is flexible and readable. We want to use Hypertext Transfer Protocol (HTTP) requests to create, read, update, and delete (CRUD) data by REST API.

## 8 Build Schedule

More detail build schedule can be found:

<https://weikangyang1008s-team.monday.com/boards/5460835934>

### 8.1 Build 1

|              |   |
|--------------|---|
| Juyeon Nam   | Implement Some Back-End Tests.<br>Implement Bare-Bone Backend code                                |
| Seolhee Yun  | Implement Some Front -End Tests   |
| Shihao Wen   | Deploy a Bare-Bones Full-Stack Project.<br>Implement Bare-Bone Front&Backend code<br>Setup CI/CD. |
| Weikang Yang | Server Deployment.<br>Make Build Schedule.  |

### 8.2 Build 2

|             |   |
|-------------|---|
| Juyeon Nam  | layout and style:<br>-Dashboard<br>-DashboardMapView<br>-DashboardSearchMapView<br>-AdminDashboard<br>-AdminDashboardMapView<br>-AdminDashboardUserList |
| Seolhee Yun | layout and style:<br>-AppBanner<br>-Login   |

|              |   |
|--------------|---|
|              | <ul style="list-style-type: none"> <li>-SignUp</li> <li>-ForgotPassword</li> <li>-DashboardMyProfileView</li> </ul>   |
| Shihao Wen   | Client Deployment.  |
| Weikang Yang | <p>layout and style:</p> <ul style="list-style-type: none"> <li>-Main page</li> <li>-DashboardCreateOrEditMapView</li> <li>-MapView</li> <li>-MapViewCustomizeToolbar</li> <li>-MapViewDiscussionForum</li> </ul> |

## 8.3 Build 3

|              |  |
|--------------|--|
| Juyeon Nam   | User Registration functionality<br>User Login functionality<br>Edit User Profile functionality<br>Add Jest back-end test   |
| Seolhee Yun  | Forgot Password functionality<br>Change Password functionality<br>Basic UI for User Profile<br>Add Cypress front-end test  |
| Shihao Wen   | View User Profile functionality<br>Create New Map functionality<br>Edit Existing Map functionality<br>Map Classification functionality<br>View Map functionality |
| Weikang Yang | Basic UI for Registration/Login<br>Basic UI for User Dashboard   |

## 8.4 Build 4

|              |   |
|--------------|---|
| Juyeon Nam   | Map Navigation functionality<br>Add explanation part in Main page<br>Customize Map - Text functionality<br>Add Jest back-end test   |
| Seolhee Yun  | Map Navigation functionality<br>Customize Map - Color functionality<br>Add Cypress front-end test   |
| Shihao Wen   | Fork Map functionality<br>Import map functionality<br>Delete editing functionality<br>Share Map functionality<br>Search Map functionality<br>Customize Map - Color functionality<br>Customize Map - Text functionality<br>Customize Map - Legend functionality<br>Redo/Undo functionality |
| Weikang Yang | Map Navigation functionality<br>Export Map functionality  |

## 8.5 Build 5

|            |  |
|------------|--|
| Juyeon Nam | Admin User Dashboard functionality<br>User Delete functionality<br>Review User-Generated Content(Admin) functionality<br>Modify of some UI(drop down menu, button hover, customize page, map view page, profile page...)<br>Add Jest back-end test |
|------------|--|

|              |  |
|--------------|--|
| Seolhee Yun  | <p>Admin User Dashboard functionality</p> <p>Show maps according to user in Admin dashboard</p> <p>Review User-Generated Content(Admin) functionality</p> <p>Modify of some UI(drop down menu, button hover, customize page, map view page, profile page...)</p> <p>Add Cypress front-end test</p> |
| Shihao Wen   | <p>Discuss Maps functionality</p> <p>Participate in Map Discussion functionality</p> <p>Admin Map Dashboard functionality</p>  |
| Weikang Yang | <p>Discuss Maps functionality</p> <p>UI for Dark mode</p> <p>UI on the small screen</p>  |

## 8.6 Build 6

|             |   |
|-------------|---|
| Juyeon Nam  | <p>Write Final Documentation</p> <p>Set different img according to mapType</p> <p>Browser alert when successful after modifying the profile</p> <p>Main page css &amp; Website tab design</p> <p>Fix Error: When deleting the user, the map and profile are not deleted.</p> <p>Fix Error: Admin Map page - sidebar size(width is different between userlist&amp;maplist)</p> <p>Fix Error: Server timeout error when user delete</p> <p>Fix Error: Discussion bar css(position/text input&amp;button does not shown properly)</p> <p>Fix Error: Login&amp;sign in error message does not appear properly</p> <p>Add Jest back-end test</p> |
| Seolhee Yun | <p>Fix and create UI for customize(map type) + error message in custom page</p>   |

|              |  |
|--------------|--|
|              | <p>Browser alert when successful after modifying the profile</p> <p>Fix Error: If the user does not exist and only the map exists, clicking on the map will cause a browser alert due to the inability to read user information.</p> <p>Fix Error: Modify the UI if the user does not have a map</p> <p>Fix Error: Login status doesn't hold after page refresh</p> <p>Fix Error: Login&amp;sign in error message does not appear properly</p> <p>Fix Error: Discussion comment for empty string</p> <p>Add Cypress front-end test</p>           |
| Shihao Wen   | <p>Heat Map functionality</p> <p>Point Map functionality</p> <p>Bubble Map functionality</p> <p>Thematic Map functionality</p> <p>Choropleth Map functionality</p> <p>Fix Error: Not saved and showing saved front</p> <p>Fix Error: The legend is not being saved properly.</p> <p>Fix Error: Login status doesn't hold after page refresh</p> <p>Fix Error: Admin page only works after refresh</p> <p>Fix Error: Server timeout error when user delete</p> <p>Fix Error: if one map deleted, remain map does not appear properly in admin</p> |
| Weikang Yang | <p>Dark mode update &amp; error fix</p> <p>Fix and create UI for customize(map type) + error message in custom page</p> <p>Fix Error: After customizing, click view to display the custom tool.</p>  |

## 9 Reflection

Name: Weikang Yang

I learned the following skills in this class:

- I learned how to design and organize the component structure of a front-end application so that the code is easy to understand and maintain. Split the application's components into smaller reusable parts by iteratively reviewing and refactoring the components to improve the maintainability and scalability of the code.
- Understood the importance of teamwork in software engineering, including code version control, merge requests, and team communication. Used version control tools (Git) to manage code and resolve conflicts in a timely manner. Ensured that team members maintained good communication and collaboration with each other, and conducted regular code reviews to ensure quality. Communicate via discord group and build the work pipeline/schedule with Monday.com.
- Understood the importance of teamwork in software engineering, including code version control, merge requests, and team communication. Ensured styling and layout specifications between components and used global styling or CSS modularization to isolate component styles to minimize conflicts.
- Learned how to deploy a front-end application to a production environment for user access. Understood different deployment options (e.g. static hosting, cloud hosting, etc.) and chose the right deployment method based on the requirements. Learned server configuration and performance optimization techniques to ensure that the application runs stably in the production environment.

What struck me the most was that we wanted to provide a night mode toggle in the app. The night mode required changing the appearance of the app, such as background color, text color, etc. The problem was how to make sure the styles in different components responded correctly to the change. If the state of one component changes, other components need to be notified to update accordingly. The challenges we encountered in implementing this feature include: communicating effectively between different components, ensuring that the styles in different components respond correctly to changes in the night mode, and sharing and managing the state of this mode between different components is the more difficult part.

Name: Juyeon Nam

Through the project, I learned a lot of things and because of it, our product was satisfying. The product we made followed the initial and build plan well. Our final product also worked well without errors.

If we had more time to improve the project, we would have created more diverse pages. On our application, there were cases where two functions were on one page. For example, our user profile page has two functionalities users check their profile and edit their information. If there are clearly separate pages, it would be better.

It was my first time working on a project with more than two people and I was able to learn a lot. First of all, I was able to learn how to communicate with my team members. By communicating with each other when we were confused or stuck at some point, we were able to come up with an effective solution. The project seems to have improved by communicating and sharing opinions.

Furthermore, I realized the importance of the planning stage and CI/CD in a large project. It was a significant experience to know the various planning stages and CI/CD and learn how to make those. Through detailed plans made in advance, we were able to reduce confusion during the build stage and work efficiently because each person could work according to the plan.

Additionally, through this project, I can learn a lot from other people's code. Since it was a team project, I often had to understand other people's code. It was meaningful because I learned a lot of new things such as functions in the process of seeing and understanding functions that I do not use often. In particular, it was my first time writing code using useContext, which I learned was very effective in terms of managing and calling states. Also, it was a meaningful experience that allowed me to know how to useEffect and useState appropriately in various situations. Additionally, I can learn how others organize code structures and manage codes. The bare-bone code that one of our team members made was amazing, and that made our work less confusing and much easier.

Through this project, I was able to gain experience in teamwork and project management as well as technical skills. It was valuable to learn a lot through the project and thanks to my team members, the project was successfully finished.

Name: Shihao Wen

Throughout the course, I learned to use various web development tools, including BrowserRouter, Route, Routes, createContext, useContext, useEffect, useState, useNavigate, and Link. These technologies play a vital role in the functionality and user experience of the application. For example, the use of useEffect is crucial for managing side effects and helps improve program efficiency.

In addition to technical skills, the course also provides methods for writing system design, system analysis, use case models, user interface models, data models, and software models. These models serve as important guides, ensuring that the application is comprehensive and well-structured.

Assigning tasks to team members is a key aspect of our collaboration. We assign tasks based on factors such as individual strengths and workload distribution. However, challenges come with them, and effective coordination is critical to achieving our common goals.

One of the most challenging phases of the project was during Build #5, when TA highlighted the need to implement at least five different map types. This requirement resulted in a major reorganization of the program structure and data structures, which cost me several days of effort.

Looking back, I realize the importance of a robust initial design of program architecture and data models. If there is an opportunity for redesign, I would gain a deeper understanding of the program architecture and data model.

Name: Seolhee Yun

I experienced the entire process of organizing a project plan and completing it

through this class. At the beginning of this semester, I learned about planning projects and writing documents. This whole process was new to me, so I was very inexperienced. However, I think I made progress little by little by communicating with my team members and making plans.

In this project, I participated in all aspects of the website planning process, from role allocation to deployment. I collaborated with my team through Github and saw their updates in real-time. While planning the project, we created expected functions and scenarios through the Use Case Model and learned how to realize our plans through the Interface Model, Data Model, and Software Model.

In particular, I was in charge of front-end testing using Cypress, which involved testing the code completed by team members at the end of each week. I reviewed whether our program was working well and what errors were occurring and reported this to my team members. Sometimes I would fix the errors I found and test again. It was my first time testing through code, but I realized that testing is an important part of applying it to a large project.

As I worked on the project with my team, I realized again the importance of communication and filling in gaps. My team members were all very quick and responsive, exchanged ideas freely, and helped each other with problems they couldn't solve. Our team's method was to classify tasks and then assign them to individuals, which became more effective when we used Monday. This allowed team members to see whether they had started, worked on, or finished a task, and they also posted any changes to their plans right away on Monday. This worked best for our last build #6. My team members and I checked for errors and posted them on Monday to let all team members know that an error had occurred, and then they fixed their parts.

One semester was a very short time to plan and complete the project, and the actual implementation took only 6 weeks, but relying on each other with my team members to solve this problem was a precious experience.