



Software Engineering: CSE 416

Presented by: Amelia Orodio, Jacob Niyazov , Jesse Wang , Juan Ospina

## Table of Contents

1. Use Case Model
2. User Interface Model
3. Data Model
4. System/Domain Analysis Model
5. Software Model
6. Progress Reviews
7. Post Mortem
8. Appendix A: All Meeting Minutes
9. Appendix B: Requirements Presentation Slides
10. Appendix C: (N/A)
11. Appendix D: Detailed Description of Team Member Contributions

# 1. Use Case Model

Use Case #	UI Context	Use Case Name
2.1	Login Screen	Registered User Login
2.2	Login Screen	Guest User Login
2.3	Login Screen	Password Recovery
2.4	Register Screen	User Account Registration
2.5	Personal Profile Page	Logout User
2.6	Personal Profile Page	Create New Map
2.7	Personal Profile Page	Upload Map
2.8	Personal Profile Page	Import Map
2.9	Account Management Page	Edit Account Information
2.10	Application Support Screen	View FAQ
2.11	Map Feed Screen	Search
2.12	Map Feed Screen	Fork Map
2.13	Map Feed Screen	React to Map
2.14	Public Map View Screen	Navigating in a Map
2.15	Public Map View Screen	Comment on Map
2.16	Public Map View Screen	Export Public Map
2.17	Map Customization Page	Export Private Map
2.18	Map Customization Page	Editing Region Data (Heat Map)
2.19	Map Customization Page	Editing Region Data (Dot Distribution Map)
2.20	Map Customization Page	Editing Region Data (Choropleth Map)
2.21	Map Customization Page	Editing Region Data (Voronoi)

		Map)
2.22	Map Customization Page	Editing Region Data (Spike Map)
2.23	Map Customization Page	Delete Map
2.24	Map Customization Page	Publish Map
2.25	Map Customization Page	Undo/Redo

Use Case Number	2.1
Use Case Name	Registered User Login
Actors	Registered User
Story	<ol style="list-style-type: none"> <li>1. On the Login page, Click on the “Email Address” text field and fill out the user’s email</li> <li>2. Click on the “Password” text field and fill out the user’s password</li> <li>3. Press “Sign In,” which will forward them to the Personal Profile Page</li> </ol>
Scenario	When initially getting onto the site, bring the cursor to the “Email Address” box and type “joe.shmo@gmail.com.” Then after pressing enter, bring the cursor to the “Password” box and type “123123123.” Press “Sign In.” If the credentials are correct, then it will lead to the Personal Profile Page, however, if it does not, it will stay on the Login Screen and display an error message.
Exceptions	If the user enters the wrong email or password, the user will be shown a pop-up dialog with an error message

Use Case Number	2.2
Use Case Name	Guest User Login
Actors	Guest User
Story	1. Click on the Guest Login Button
Scenario	When initially getting onto the site, bring the cursor to the “Guest” button and press it. This will lead to the Public Map View Screen.
Exceptions	N/A

Use Case Number	2.3
Use Case Name	Password Recovery
Actors	Registered User
Story	<ol style="list-style-type: none"> <li>1. Under the Login section, enter the email of the account in the Email field</li> <li>2. Navigate to where it says “Forgot your password?”</li> <li>3. Press this text and it will send an email to the specified email with a link to reset the password</li> <li>4. Open the unique link in the sent email</li> <li>5. On the page that opens, two text fields named “New Password” and “Retype New Password” will appear</li> <li>6. Type the same new password into these text fields.</li> <li>7. Press enter and it will set the new password</li> </ol>
Scenario	Joe Shmo has forgotten the password to his account. He is on the Login Screen, enters “joeshmo@gmail.com” into the Email field, brings his cursor to where it says “Forgot your password?” and presses this. If the email exists, he will receive an email with a link to a password reset. When he opens the link it will show the two text fields “New Password” and “Retype New Password.”. He types the new password in twice and presses enter. A confirmation modal appears to tell of his successful password update.
Exceptions	If the user inputs an email that does not exist, an error message will pop up. If the user inputs two different new passwords, an error message will pop up.

Use Case Number	2.4
Use Case Name	User Account Registration
Actors	Unregistered User, Guest User
Story	<ol style="list-style-type: none"> <li>1. On the Register side, navigate to the Email Address text field and input your email</li> <li>2. Go to the Username text field and type in your username</li> <li>3. Underneath that will be the Password text field and type in your password</li> <li>4. Retype this password in the field under listed as “Retype Password”</li> <li>5. Press the button that says “Register” and this leads into the Personal Profile Page</li> </ol>
Scenario	Jane Doe is creating an account. She leads her cursor towards the Register side of the Login screen. She inputs “jane@gmail.com” into the Email field, “Jane Doe” into Username field, “JD123” into the Password field, and retyped the password into the Confirm Password field. She presses the button that states “Register” and logs in, however, if the email or username are taken or if the passwords don’t match, an error message will occur.
Exceptions	If the user inputs an email that is already in use then an error message will show up. If the user inputs a username that already exists then an error message will show up. If the password and the retyped password don’t match then an error message will pop up.

Use Case Number	2.5
Use Case Name	Log out User
Actors	Registered User
Story	<ol style="list-style-type: none"> <li>1. Navigate to the Accounts Menu</li> <li>2. Click on the logout button</li> </ol>
Scenario	Bring the mouse to the top-right corner of the interface where the accounts menu is and click on the menu so that it opens a drop-down Menu. Click on the log-out Button. This will log the user out and forward them to the Login Screen.
Exceptions	This assumes the user is logged in currently.

Use Case Number	2.6
Use Case Name	Create New Map
Actors	Registered User
Story	<ol style="list-style-type: none"> <li>1. Navigate to the Accounts Menu</li> <li>2. Click on the My Profile button</li> <li>3. Click on the Create Map button.</li> </ol>
Scenario	Joe Shmo would like to create a new map to begin editing. He brings the mouse to the top-right corner of the interface where the account menu is and he clicks on the menu so that it opens a drop-down menu. He then clicks on the “My Profile” button. This will open the personal profile page unique to him. He clicks on the new map button which will generate a map with a default base map for the user to make edits.
Exceptions	This assumes the user is logged in currently.

Use Case Number	2.7
Use Case Name	Upload Map
Actors	Registered User
Story	<ol style="list-style-type: none"> <li>1. Navigate to the Accounts Menu</li> <li>2. Click on the My Profile button</li> <li>3. Click on the Import Map button</li> <li>4. Select a file which uses the applications native file type in the user's file system and click open</li> <li>5. If successful, the user will be directed to the "Edit Map" page with the information read from the map</li> <li>6. If unsuccessful, there will be an error modal popup</li> </ol>
Scenario	<p>Joe Shmo would like to create a new map to begin editing. He brings the mouse to the top-right corner of the interface where the accounts menu is and he clicks on the menu so that it opens a drop-down menu. He then clicks on the "My Profile" button. This will open the personal profile page unique to him. He clicks on the import map button which opens the user's file system. Joe has found a map that he previously exported from our software and wants to make edits to it. He selects that map and clicks "Open". This map will be a replica of the map that the user has exported previously, including all the data and edits from before and Joe is free to begin editing.</p>
Exceptions	If the user uploads a map in the wrong format, the user will be shown a pop-up dialog with an error message

Use Case Number	2.8
Use Case Name	Import Map
Actors	Registered User
Story	<ol style="list-style-type: none"> <li>1. Navigate to the Accounts Menu</li> <li>2. Click on the My Profile button</li> <li>3. Click on the Upload Map button</li> <li>4. Select a vector map format (geoJSON, shape file, or KML) in the user's file system and click open.</li> <li>5. If successful, the user will be directed to the "Edit Map" page with the information read from the map</li> <li>6. If unsuccessful, there will be an error modal popup</li> </ol>
Scenario	Joe Shmo would like to create a new map to begin editing. He brings the mouse to the top-right corner of the interface where the accounts menu is and he clicks on the menu so that it opens a drop-down menu. He then clicks on the "My Profile" button. This will open the personal profile page unique to each user. He clicks on the import map button which opens the user's file system. Joe has found a geoJSON file online that he would like to use. He selects that file and hits open. The map loads and is now ready for editing.
Exceptions	If the user uploads a map in the wrong format, the user will be shown a pop-up dialog with an error message

Use Case Number	2.9
Use Case Name	Edit Account Information
Actors	Registered User
Story	<ol style="list-style-type: none"> <li>1. Navigate to the Account Menu</li> <li>2. Click on the Manage Account button</li> <li>3. Change the user's name, username, and/or email by typing into the respective fields, or leaving the fields blank to leave them as is</li> <li>4. Change the password by entering it into the "New Password" field and re-entering it into the "Confirm New Password" field, or leave the fields blank to leave it as is</li> <li>5. Press the Submit button</li> <li>6. A confirmation dialog appears to notify the user of successful changes</li> </ol>
Scenario	<p>John Doe wants to change some of his account details. He brings the mouse to the top-right corner of the interface where the accounts menu is and clicks on the menu so that it opens a drop-down menu. He clicks on the "Manage Account" button. This will open the Account Management page unique to each user. He leaves the "Name" field empty to leave it as is, enters a "CoolDude" into the "Username" field, enters "cooldude@gmail.com" into the "Email" field, and enters the same new password (i.e. "thecoollestduke123") into both the "New Password" and "Confirm New Password" fields. He presses the Submit button and this updates his account details. A confirmation dialog appears and says "Your username, email, and password have been updated successfully".</p>
Exceptions	<p>If the user doesn't enter the same password into both the "New Password" and "Confirm New Password" fields, a dialog will pop up to warn them to enter the same password twice. If the new password doesn't match the password requirements a dialog will pop up to warn them to try a different password.</p>

Use Case Number	2.10
Use Case Name	View FAQ
Actors	Registered User, Guest User
Story	<ol style="list-style-type: none"> <li>1. Navigate to the Account Menu</li> <li>2. Click on the Help button</li> <li>3. Expand any of the questions on the screen that one is curious about and read the custom-written response by the developers on how to use this aspect of the application</li> </ol>
Scenario	John Doe is not sure how this application works. He brings the mouse to the top-right corner of the interface where the accounts menu is and clicks on the menu so that it opens a drop-down menu. He clicks on the “Help” button. This will open the Application Support Screen. He sees the question “How can I import a Shapefile?” and wonders how to do this, so he clicks on both surrounding the question and it expands the step-by-step answer on how to import a Shapefile.
Exceptions	N/A

Use Case Number	2.11
Use Case Name	Search
Actors	Registered User, Guest User
Story	<ol style="list-style-type: none"> <li>1. While on the Map Feed Screen, bring the cursor to the search bar at the top of the page and click it</li> <li>2. Type in a map name or username and press the Enter key on the keyboard or the Search button</li> <li>3. View the maps with the specified map name or maps created by the user with the specified username</li> </ol>
Scenario	John Doe is looking for a cool map about cow populations. While on the Map Feed Screen, he brings the mouse to the top of the interface where the search bar is and clicks it. He types into the search bar “Cow Map” and presses the Enter key on his keyboard. A map titled “Cow Map” appears which displays a heat map of cow populations by administrative level.
Exceptions	If the user enters a map name or user name that does not exist, nothing will show up on the search.

Use Case Number	2.12
Use Case Name	Fork Map
Actors	Registered User
Story	<ol style="list-style-type: none"> <li>1. Click on a public map to reach the Public Map View Screen of this particular public map</li> <li>2. Click the “Fork Map” button in the bottom right corner of the interface</li> <li>3. Enter a name for the map you will fork into the dialog that appeared</li> <li>4. Click the “Fork Now” button</li> <li>5. A confirmation dialog appears to confirm the creation of the forked map</li> </ol>
Scenario	John Doe is looking at a map that he really likes on the Map Feed Screen. He clicks on the map and reaches its Public Map View Screen. From there he clicks the “Fork Map” button in the bottom right corner. A dialog opens asking him to enter a map name for his newly forked map. He enters “My COOLEST map ever” and clicks “Fork Now”. A confirmation dialog appears to say “You have successfully forked ‘My COOLEST map ever’”. The forked map appears on his Personal Profile Page.
Exceptions	If the user enters a map name for the map they want to fork that is not unique (i.e. a map with this name already exists) a dialog will show up warning the user of this and telling them to try again.

Use Case Number	2.13
Use Case Name	React to Map
Actors	Registered User
Story	<ol style="list-style-type: none"> <li>1. While on the Map Feed Screen, navigate the cursor to the thumbs up/down button under any map</li> <li>2. Click the thumbs up/down button to react to the map</li> </ol>
Scenario	John Doe is looking at a map that he really likes on the Map Feed Screen. He clicks on the thumbs-up icon under this map to like the map. He later realizes he doesn't like the map and clicks on the thumbs-down icon under this map to dislike the map and his reaction changes from thumbs up to thumbs down.
Exceptions	A user may only like (thumbs up) or dislike (thumbs down) a map, not both. Only public maps can have reactions. A user also cannot like or dislike a map multiple times.

Use Case Number	2.14
Use Case Name	Navigating in a Map
Actors	Registered User, Guest User
Story	<ol style="list-style-type: none"> <li>1. While on the Map Feed Screen, click on a public map to reach this particular map's Public Map View Screen.</li> <li>2. Place the cursor inside the map.</li> <li>3. Hold down the left mouse button and drag the mouse in any direction to pan the map in that direction.</li> <li>4. Hover the cursor over a section of the map and use the mouse wheel to zoom in and out into that section.</li> </ol>
Scenario	John Doe is looking at a dot distribution map on the Map Feed Screen and wants a closer look at a cluster of dots. He clicks on the map and reaches its Public Map View Screen. John hovers his cursor over the cluster and scrolls the mouse wheel downwards. He notices another smaller cluster nearby and decides to hold down the left mouse button and drag the mouse toward the smaller cluster. To zoom back out and view the entire map, John will scroll the mouse wheel downward.
Exceptions	N/A

Use Case Number	2.15
Use Case Name	Comment on a Map
Actors	Registered User
Story	<ol style="list-style-type: none"> <li>1. While on the Map Feed Screen, click on a public map to reach this particular map's Public Map View Screen.</li> <li>2. Click on the "Add a Comment" text field and fill it out with a comment.</li> <li>3. Click on the "Post" text field to post the comment on the map.</li> </ol>
Scenario	John Doe is looking for a cool map about cow populations. While on the Map Feed Screen, he clicks on the map and reaches its Public Map View Screen. He brings his mouse to the "Add a Comment" and types "Cow maps rock!". He clicks on the "Post" text field and shares his thoughts on cow maps with the world.
Exceptions	This assumes the user is currently logged in. If a user posts an empty comment a dialog will show up warning the user of this and telling them to try again.

Use Case Number	2.16
Use Case Name	Export Public Map
Actors	Registered User, Guest User
Story	<ol style="list-style-type: none"> <li>1. While on the Map Feed Screen, click on a public map to reach this particular map's Public Map View Screen</li> <li>2. Click on the "Export Map" text field which will download the map file in the application's native file type onto the local computer</li> </ol>
Scenario	John Doe is looking at a map that he likes on the Map Feed Screen. He clicks on the map and reaches its Public Map View Screen. He brings his mouse to the "Export Map" and the map downloads onto his computer.
Exceptions	N/A

Use Case Number	2.17
Use Case Name	Export Private Map
Actors	Registered User
Story	<ol style="list-style-type: none"> <li>1. Navigate to the Accounts Menu</li> <li>2. Click on the My Profile button</li> <li>3. Click on the edit symbol on a map that you would like to export to reach the Map Customization Page</li> <li>4. Click on the “Export Map” text field which will download the map file in the application’s native file type onto the local computer</li> </ol>
Scenario	John Doe wants to download a map he has been working on. He goes to his profile, clicks on the edit symbol for the corresponding map and reaches its Map Customization Page. He brings his mouse to the “Export Map” and the map downloads onto his computer.
Exceptions	This assumes the user is currently logged in.

Use Case Number	2.18
Use Case Name	Editing Region Data (Heat Map)
Actors	Registered User
Story	<ol style="list-style-type: none"> <li>1. Click on a map to enter the Map Customization Page</li> <li>2. Type into the “Name” field to change the name of the map</li> <li>3. Toggle/untoggle the base map feature</li> <li>4. Change text color using the color selector</li> <li>5. Change text font by selecting a font from the dropdown</li> <li>6. Change text size using the text size dropdown</li> <li>7. Edit which fields are in the legend and what each of them shows</li> <li>8. Change the color gradient of the heat map by selecting the two most extreme colors from the two heat map color selectors</li> <li>9. Click on a region and type a new value into the dialog box to edit the quantity/data being represented in that region</li> <li>10. Click the Finish Edits button to save changes</li> </ol>
Scenario	John Doe wants to edit his favorite heatmap. While on the Personal Profile Page, he clicks on the map and the heatmap Map Customization Page appears. He renames the map to “Best Heatmap” by typing into the name field. He changes the text size to 12pt by selecting it from the respective drop-down. Finally, he selects pink and yellow as the two extremes of the heatmap gradient. He clicks Finish Edits and a confirmation modal appears to confirm that he wishes to save the changes.
Exceptions	Assuming the user is logged in and the map clicked on is a heat map. If data entered into a region doesn't map the data used in other regions of the map, an error modal will appear warning the user to try again.

Use Case Number	2.19
Use Case Name	Editing Region Data (Dot Distribution Map)
Actors	Registered User
Story	<ol style="list-style-type: none"> <li>1. Click on a map to enter the Map Customization Page</li> <li>2. Type into the “Name” field to change the name of the map</li> <li>3. Toggle/untoggle the base map feature</li> <li>4. Change the text color, border color, and region color by selecting a box and using the color selector</li> <li>5. Change text size using the text size dropdown</li> <li>6. Change text font by selecting a font from the dropdown</li> <li>7. Change dot attributes (color, data, size) by selecting a dot and customizing the information in the properties display.</li> <li>8. Change features for all dots by clicking the “Select All” button and editing the information in the properties display</li> <li>9. Edit which fields are in the legend and what each of them shows</li> <li>10. Click the Finish Edits button to save changes</li> </ol>
Scenario	<p>John Doe wants to edit one of his unpublished Dot Distribution Maps. He goes to his profile, clicks on the map, and reaches its Map Customization Page. From there he clicks on the “View” button which takes him to a dot distribution map that he has uploaded previously. He doesn’t like the name of the map and clicks on the title and changes it. He decides to add more dots and selects the Dot icon. Once the dot icon is highlighted, he clicks on the map which drops a dot down, and unselects the dot icon by clicking it again. He wants the size of this particular dot to be larger so he clicks on that dot to bring up a dot customization menu. Here, he slides the slider option for customizing size to the maximum. He then realizes he wants to change the color of all the dots. He clicks the “Select All” button next to the Dot Icon and customizes the color in the customization menu which changes the color for all of the Dots on the map. John is happy with his changes so he clicks Finish Editing and confirms via the confirmation modal.</p>
Exceptions	This assumes the user is currently logged in and has selected a Dot Distribution Map.

Use Case Number	2.20
Use Case Name	Editing Region Data (Choropleth Map)
Actors	Registered User
Story	<ol style="list-style-type: none"> <li>1. Click on a map to enter the Map Customization Page</li> <li>2. Type into the “Name” field to change the name of the map</li> <li>3. Toggle/untoggle the base map feature</li> <li>4. Change text color using the color selector</li> <li>5. Change text size using the text size dropdown</li> <li>6. Change the font using the text font dropdown</li> <li>7. Edit which fields are in the legend and what each of them shows</li> <li>8. Change the color of the background or the borders using its respective color selector</li> </ol>
Scenario	<p>Jane Doe wants to edit an unpublished choropleth map they currently have on their Personal Profile Page. When they press on the map that they want to edit, it leads to the Map Customization Page for the selected map. When the Map Customization Page is open, they are able to change various aspects of the Choropleth Map. They change the name of the map by bringing their cursor to the “Name” field to “this map is awesome!!!” They are able to toggle the base map on and off to their desire. They are able to change the text color using the color selector. They are able to change the text size and font using the drop-downs for those respective features. To change the data of the legend, there will be various fields they can input such as the name of the legend and its contents. Lastly, they are able to change the color of the background or border of a region with their respective color-selectors and they change their color scheme to a pink-orange-white scheme. After completing all the edits they decide to do, they have a choice to select save or publish and will return them to the Personal Profile Page.</p>
Exceptions	This assumes the user is currently logged in and has selected a Choropleth Map.

Use Case Number	2.21
Use Case Name	Editing Region Data (Voronoi Map)
Actors	Registered User
Story	<ol style="list-style-type: none"> <li>1. Click on a map to enter the Map Customization Page</li> <li>2. Type into the “Name” field to change the name of the map</li> <li>3. Toggle/untoggle the base map feature</li> <li>4. Change text color using the color selector</li> <li>5. Change text size using the text size dropdown</li> <li>6. Change the font using the text font dropdown</li> <li>7. Edit which fields are in the legend and what each of them shows</li> <li>8. Change the color of the background, map borders, and Voronoi borders and regions using its respective color selector</li> <li>9. Change dot attributes (color, data, size) by selecting a dot and customizing the information in the properties display.</li> <li>10. Change features for all dots by clicking the “Select All” button and editing the information in the properties display</li> <li>11. Edit which fields are in the legend and what each of them shows</li> <li>12. Click the Finish Edits button to save changes</li> </ol>
Scenario	<p>John Doe wants to download a map he has been working on. He goes to his profile clicks on the map and reaches its Map Customization Page. From there he clicks on the “View” button which takes him to a dot distribution map that he has uploaded previously. He doesn’t like the name of the map and clicks on the title and changes it. He decides to add more dots and selects the Dot icon. Once the dot icon is highlighted, he clicks on the map in multiple places to drop multiple dots down. Once he is complete, he selects the dot icon again to deselect the dot option. He then selects the ”Draw Voronoi Diagram” option which constructs the proper Voronoi diagram using the dots John has placed. He then chooses to use the Voronoi borders instead to color the map. He selects the option to “Color with Voronoi” which redraws the map and colors to be based on the Voronoi diagram border. John can then click on a region and change the color to green as well as other properties that he would like. John is happy with his changes so he clicks Finish Editing and confirms via the confirmation modal.</p>
Exceptions	This assumes the user is currently logged in and has selected a Voronoi Map

Use Case Number	2.22
Use Case Name	Editing Region Data (Spike Map)
Actors	Registered User
Story	<ol style="list-style-type: none"> <li>1. Click on a map to enter the Map Customization Page</li> <li>2. Type into the “Name” field to change the name of the map</li> <li>3. Toggle/untoggle the base map feature</li> <li>4. Change text color using the color selector</li> <li>5. Change text size using the text size dropdown</li> <li>6. Change text font by selecting a font from the dropdown</li> <li>7. Edit which fields are in the legend and what each of them shows.</li> <li>8. Click on a spot in the map to add spikes, and specify the size of the spike through a slider in the pop-up.</li> <li>9. Change spike attributes (color, data, size) by selecting a spike and customizing the information in the properties display.</li> <li>10. Change features for all spikes by clicking the “Select All” button and editing the information in the properties display.</li> <li>11. Click the Finish Edits button to save changes</li> </ol>
Scenario	<p>John Doe wants to download a map he has been working on. He goes to his profile, clicks on the map, and reaches its Map Customization Page. From there he clicks on the “View” button which takes him to a spike distribution map that he has uploaded previously. He doesn’t like the name of the map and clicks on the title and changes it. He decides to add more spikes and selects the Spike icon. Once the spike icon is highlighted, he clicks on the map which places a spike, and unselects the spike icon by clicking it again. He wants the size of this particular spike to be larger so he clicks on that spike to bring up a spike customization menu. Here, he slides the slider option for customizing size to the maximum. He then realizes he wants to change the color of all the spikes. He clicks the “Select All” button next to the Spike Icon and customizes the color in the customization menu which changes the color for all of the spikes on the map. John is happy with his changes so he clicks Finish Editing and confirms via the confirmation modal.</p>
Exceptions	This assumes the user is currently logged in and has selected a Spike map.

Use Case Number	2.23
Use Case Name	Delete Map
Actors	Registered User
Story	<ol style="list-style-type: none"> <li>1. Navigate to the Accounts Menu</li> <li>2. Click on the My Profile button</li> <li>3. Click on the trash symbol on a map that you would like to delete.</li> <li>4. A confirmation modal will appear asking to confirm your deletion.</li> <li>5. Select “Confirm” to confirm the deletion of your map</li> </ol>
Scenario	Jane Doe is publishing a map. She navigates to the top-right corner of the screen to press the account icon. When pressing the account icon, a drop-down appears, and “My Profile.” She presses on this text and it leads her to the Personal Profile Page. She hovers over the map she wants to delete and sees the different options pop up. She selects the trash icon which opens a confirmation modal. She selects “Confirm” to delete the map.
Exceptions	This assumes the user is currently logged in.

Use Case Number	2.24
Use Case Name	Publish Map
Actors	Registered User
Story	<ol style="list-style-type: none"> <li>1. Navigate to the Accounts Menu</li> <li>2. Click on the My Profile Button</li> <li>3. Click on an unpublished map</li> <li>4. Under all of the editing tools, navigate to the button that states “Publish”</li> <li>5. A confirmation modal will appear to confirm it’s publishing</li> <li>6. Select “Confirm” to publish the Map</li> </ol>
Scenario	Jane Doe is publishing a map. She navigates to the top-right corner of the screen to press the account icon. When pressing the account icon, a drop-down appears, and “My Profile.” She presses on this text and it leads her to the Personal Profile Page. She selects the map that she wants to publish and it opens the editing features. With all the editing features, she navigates to the “Publish” button. She presses this button and a confirmation modal is displayed. She presses confirm and now her map is viewable by other users.
Exceptions	This assumes the user is currently logged in and the map is currently private.

Use Case Number	2.25
Use Case Name	Undo/Redo
Actors	Registered User
Story	<ol style="list-style-type: none"> <li>1. Navigate to the Accounts Menu</li> <li>2. Click on the My Profile button</li> <li>3. Click on a map.</li> <li>4. Edit the map in any way.</li> <li>5. Click on the undo button to undo the changes.</li> <li>6. Click on the redo to bring back changes that were undone.</li> </ol>
Scenario	John Doe has been working on a map but he is unsure if a change he made looks better. He is already on the Map Customization Page which he reached by going to the Accounts Menu, clicking on My Profile, and selecting the map he wanted to edit. John undoes the change he made to see if the previous version looked better and decides the newer version is better. He then clicks the redo to reach the previous state.
Exceptions	This assumes the user is currently logged in. If no edits have been made in the current session, undo and redo will be unavailable. Redo will not be available if nothing has been undone and is limited to the number of preceding undos.

## 2. User Interface Model

<b>View #</b>	<b>View Name</b>	<b>View Description</b>
1	Splash	View that welcomes users to the site and allows users to log in, register or continue as guests
2	Register	View to allow users to register
3	Reset Password	View to allow users to reset password
4	Map Feed	View which displays a scrollable list of all public maps
5	Map Feed Account Menu (Logged In)	View which shows the extended account menu while on the map feed page when a user is logged in
6	Map Feed Account Menu (Guest)	View which shows the extended account menu while on the map feed page when a guest is viewing
7	Search by User Results	View which depicts results of a search by username
8	Search By Map Results	View which depicts results of a search by map name
9	Public Map (Logged in)	View which depicts a single public map while logged in
10	Public Map Account Menu (Logged In)	View which shows the account menu while on a public map view and logged in
11	Public Map (Guest)	View which depicts a single public map while browsing as a guest
12	Public Map Account Menu (Guest)	View which shows the account menu while on a public map view and browsing as a guest
13	Personal Public Map	View which depicts a single public map owned by the logged

		in user
14	Profile (Posts)	View of the logged in user's profile published posts
15	Profile (Drafts)	View of the logged in user's profile draft maps
16	Profile Account Menu	View which shows the extended account menu while on the profile page
17	Heat Map Editing	View showing all options for editing a heat map
18	Dot Distribution Map Editing	View showing all options for editing a dot distribution map
19	Spike Map Editing	View showing all options for editing a spike map
20	Voronoi Map Editing	View showing all options for editing a Voronoi map
21	Choropleth Map Editing	View showing all options for editing a choropleth map
22	Map Editing Account Menu	View which shows the account menu while on an editing view
23	Upload File	View for allowing users to upload a native ExploreX file to create a map
24	Import File	View for allowing users to import a vector file and select which map type to create with it
25	Account Manager	View for allowing user to manage their account details
26	Account Manager Account Menu	View which shows the account menu while on the account management view
27	Help	View for displaying the FAQ
28	Help Account Menu	View which shows the account menu while on the help view
29	Finish Editing Map Modal	Confirmation modal for when the user wants to finish editing

		their map
30	Publish Map Modal	Confirmation modal for when the user wants to publish their map
31	Delete Map Modal	Confirmation modal for when the user wants to delete their map
32	Delete Draft Map Modal	Confirmation modal for when the user wants to delete their draft map
33	Upload File Error Modal	Error modal for when the user uploads an unsupported file type
34	Import File Error Modal	Error modal for when the user imports an unsupported file type
35	Login Error Modal	Error modal for when the user provides the wrong password or username
36	Reset Password Successful Modal	Modal to indicate password recovery email has been sent
37	Reset Password Failed Modal	Error modal for when the user does not exist
38	Register Username Error Modal	Error modal for when the user already exists
39	Register Password Match Modal	Error modal for when the passwords provided do not match
40	Register Password Weak Modal	Error modal for when the password is too weak
41	Save Changes to Account Modal	Error modal for when the user already exists
42	Account Manager Password Match Modal	Error modal for when the passwords provided do not match
43	Account Manager Password Weak Modal	Error modal for when the password is too weak
44	Account Manager Username	Username already exists

	Error Modal	
45	Account Manager Email Error Modal	Error modal for when the email already exists

Splash

 ExploreX

[Don't have an account?](#)

# Welcome, Map Lovers

Username or Email

Password [Forgot password?](#)

 Sign in

 Log in as Guest

## Mapping Your World, Your Way



Created by the pink team

Register

[Login](#)

## Let's get started

Username or Email

 dash@ui8.net

Password

 Password

Confirm Password

 Password[Register](#)

## Mapping Your World, Your Way



Created by the pink team

[Reset Password](#)[Don't have an account?](#)

## Forgot your password?

No worries, it isn't the end of the world.

Username or Email

 dash@ui8.net

[Recover](#)

## Mapping Your World, Your Way



Created by the pink team

Map Feed

SEARCH

Profile icon

**U.S. Choropleth**



By: Yolo Girl 102  17 

**Awesome pink map!!!!**



By: Team Pink 12  2 

©ExploreX

Map Feed Account Menu (Logged In)



ExploreX

SEARCH



## U.S. Choropleth



By: Yolo Girl

102 17

Awesome pink map!!!!



By: Team Pink

12 2

My Profile

Edit Account

FAQ

Log Out

©ExploreX

Map Feed Account Menu (Guest)



SEARCH



Log In

Register

FAQ

U.S. Choropleth



By: Yolo Girl

102



17



Awesome pink map!!!!



By: Team Pink

12



2



©ExploreX

Search by User Results



ExploreX

Yolo Girl

User



U.S. Choropleth



By: Yolo Girl

102



17



Shmoney



By: Yolo Girl

3k



2



©ExploreX



[Search by Map Results](#)



X

Map



By: hater456

35



1



©ExploreX

Public Map (Logged in)



SEARCH

Awesome pink map!!!!  
Team Pink

Hi guys I wanted to share this map because it is awesome and pink

12 2

Amelia123123

| Hey Team Pink, this map is so cool and awesome omg

hater456

| This map stinks so much

comment...

©ExploreX

Public Map Account Menu (Logged in)



SEARCH

Awesome pink map!!!!  
Team Pink

Hi guys I wanted to share this map because it is awesome and pink

12 2

Amelia123123

| Hey Team Pink, this map is so cool and awesome omg

hater456

| This map stinks so much

comment...

My Profile

Edit Account

Map Feed

FAQ

Log Out

©ExploreX

Public Map (Guest)



SEARCH



Awesome pink map!!!!  
Team Pink



Hi guys I wanted to share this map because it is awesome and pink

 12  2

Amelia123123

| Hey Team Pink, this map is so cool and awesome omg

hater456

| This map stinks so much

©ExploreX

Public Map Account Menu (Guest)



SEARCH

Awesome pink map!!!!  
Team Pink

Log In

Map Feed

FAQ

Hi guys I wanted to share this map because it is awesome and pink

12 2

Amelia123123

| Hey Team Pink, this map is so cool and awesome omg

hater456

| This map stinks so much

©ExploreX

Personal Public Map



SEARCH



Awesome pink map!!!!  
Team Pink



≡  
unpublish  
delete

Hi guys I wanted to share this map because it is awesome and pink

12 2

Amelia123123

| Hey Team Pink, this map is so cool and awesome omg

hater456

| This map stinks so much

comment...

©ExploreX

Profile (posts)

 ExploreX 

## Team Pink

Hi everyone! we are team pink and we love <333 maps!! \*yas \*slay \*pink

Posts Drafts

Awesome pink map!!!!   
35 likes 12 dislikes

less awesome pink map   
1 like 35 dislikes

yay!!! my first map!!!   
5 likes 1 dislike

©ExploreX

Profile (drafts)



## Team Pink

[Upload](#)[Import](#)

Hi everyone! we are team pink and we love <333 maps!! \*yas \*slay \*pink

[Posts](#)[Drafts](#)

Non-pink map!!!!



©ExploreX

Profile Account Menu



## Team Pink

Hi everyone! we are team pink and we love <333 maps!! \*yay \*slap

Posts

Drafts

Awesome pink map!!!!



12 1 2

less awesome pink map



35 1 1

yay!!! my



1 1 5

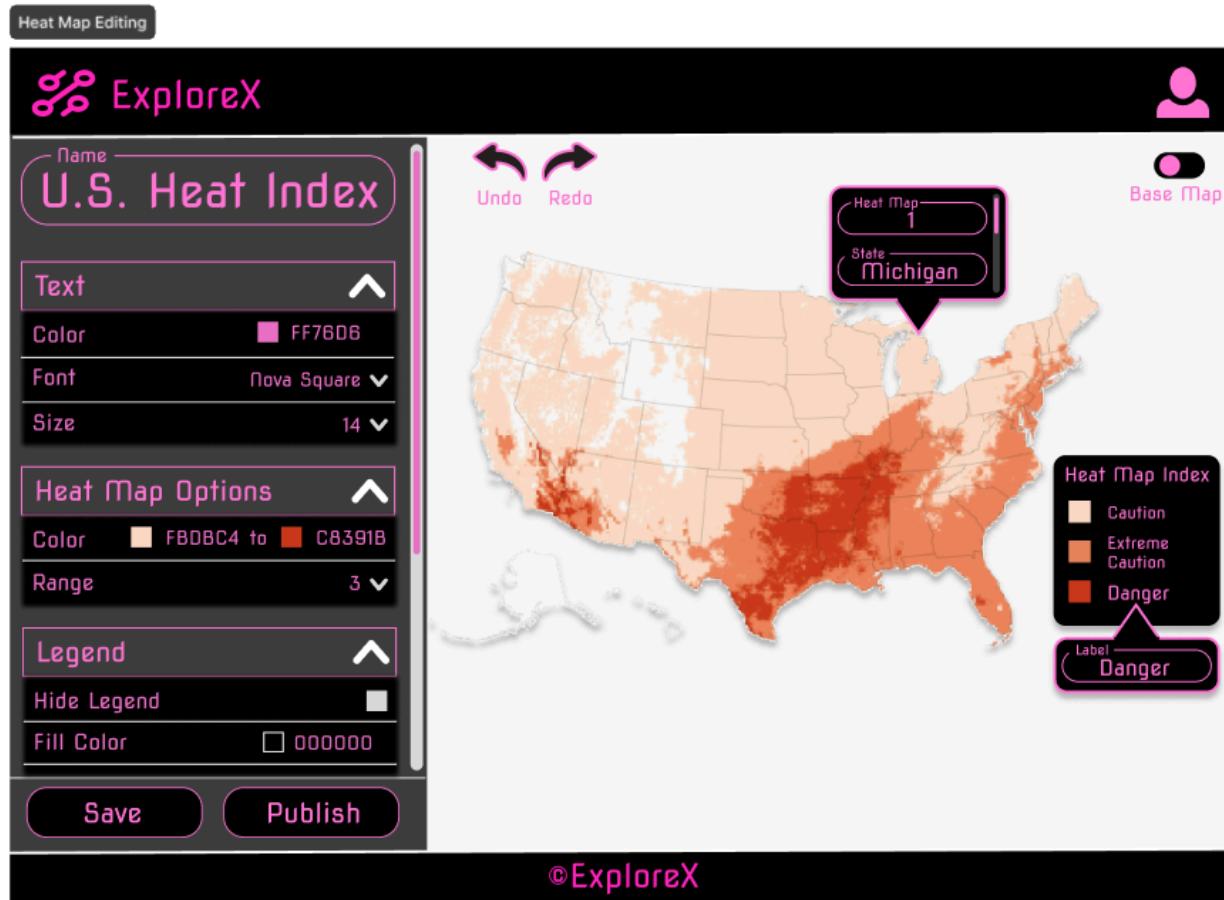
Map Feed

Edit Account

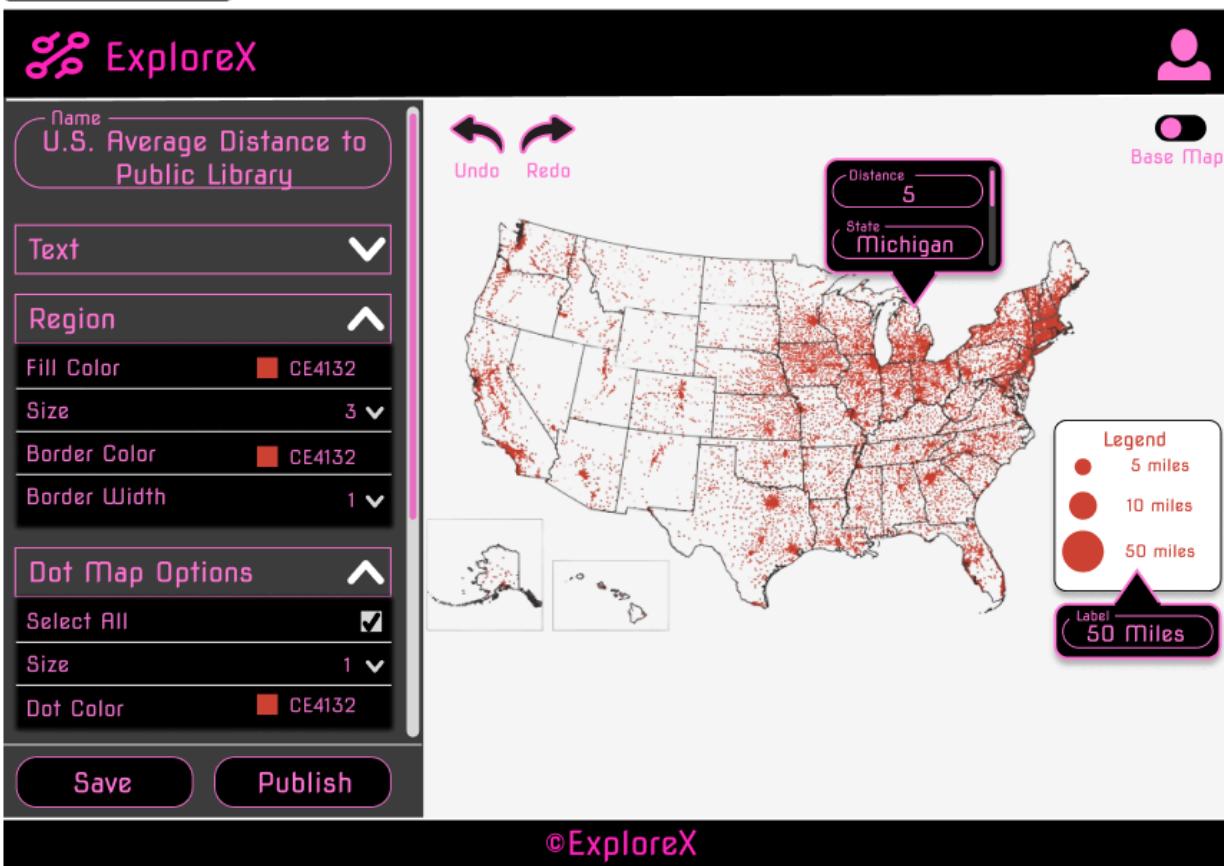
FAQ

Log Out

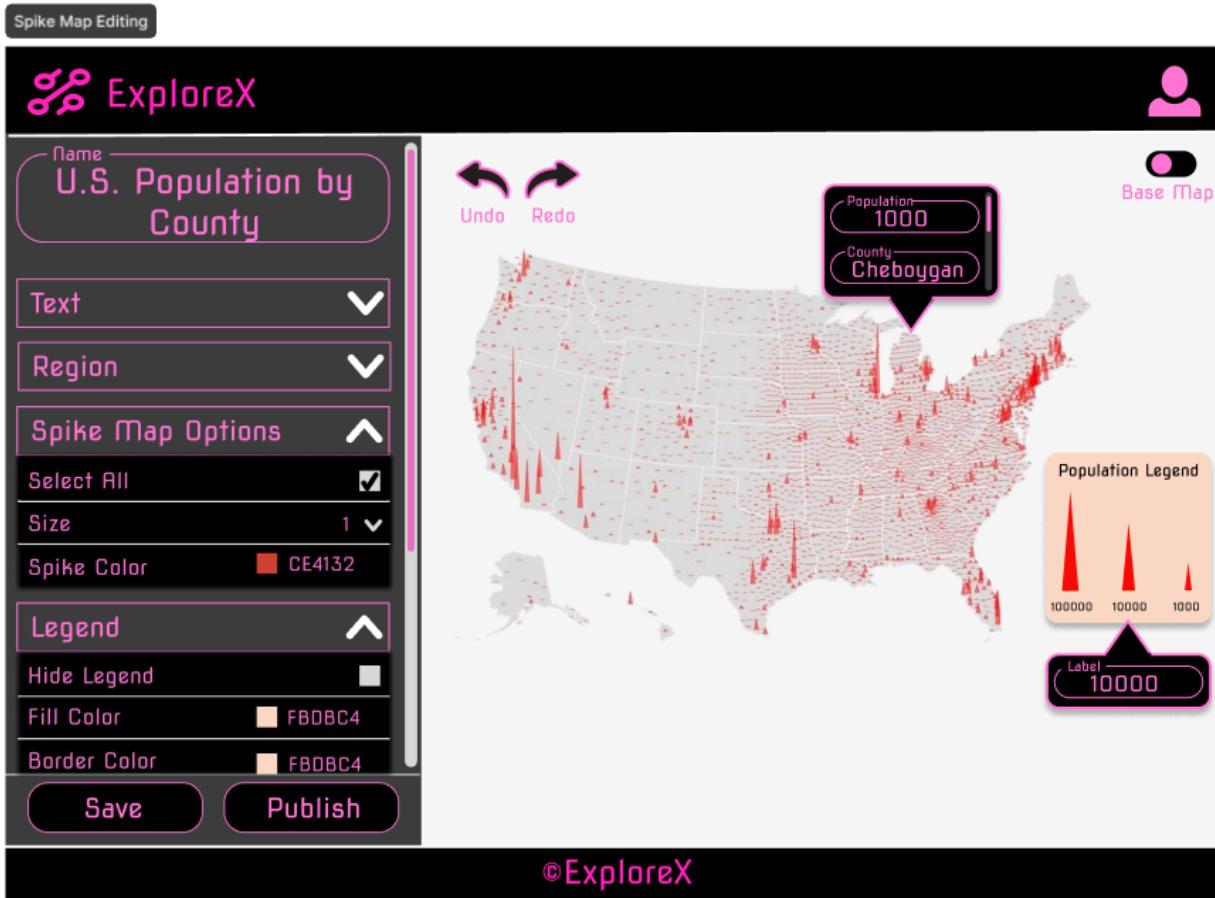
©ExploreX



Dot Distribution Map Editing



©ExploreX



Voronoi Map Editing

# ExploreX

Name

Text

Region

Fill Color

Size

Border Color

Border Width

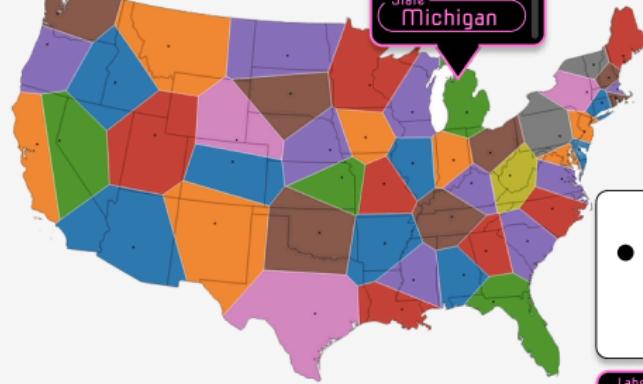
Voronoi Map Options

Select All

Size

Dot Color

Undo Redo



Address

State

Label

©ExploreX

Choropleth Map Editing

# ExploreX

Name: Political Map

Text

Color: FF76D6

Font: Nova Square

Size: 14

Region

Fill Color: 6B70D9

Size: 3

Border Color: FFFFFF

Border Width: 1

Legend

Save Publish

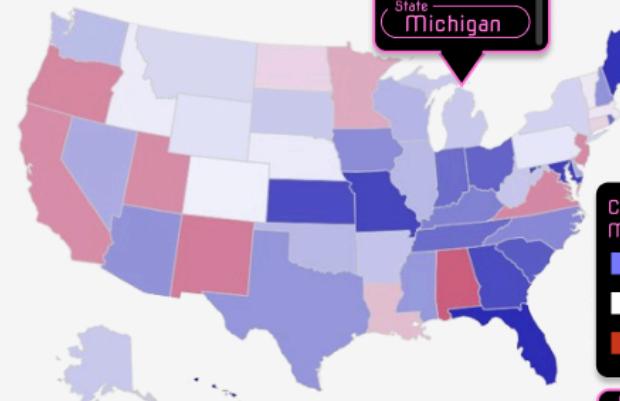
Undo Redo

Party Democrat  
State Michigan

Choropleth Map Index

- Democrat (Blue)
- Mixed (White)
- Republican (Red)

Label Republican



©ExploreX

Map Editing Account Menu

# ExploreX

## Political Map

Name: Political Map

Text:

Color: FF76D6

Font: Nova Square

Size: 14

Region:

Fill Color: 6B7DD9

Size: 3

Border Color: FFFFFF

Border Width: 1

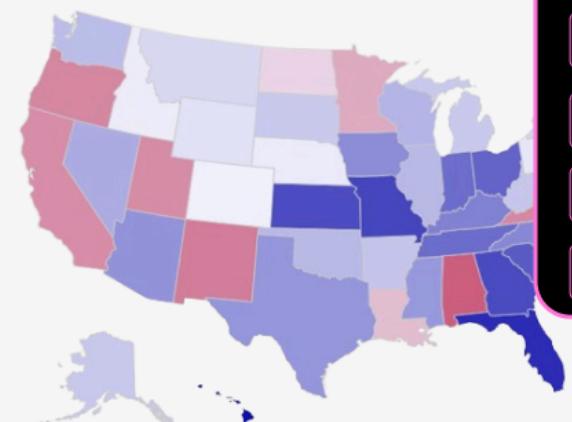
Legend:

Save

Publish

Undo

Redo



Republican

My Profile

Edit Account

Map Feed

FAQ

Log Out

©ExploreX

Upload File

# ExploreX

User icon

## Team Pink

Hi everyone! we are team pink and we love <333 maps!! #yas #slay #pink

Posts

Upload an ExploreX map file:

Awesome pink  my\_map.json  my first map!!!



12  2 

35  1 

1  5 

©ExploreX

Import File

# ExploreX

Team Pink

Hi everyone! we are team pink and we love <333 maps!! #yas #slay #pink

Posts

Awesome pink map!!!

Import a vector file below!

Accepted formats:

- Shapefile/Dbf
- KML
- GeoJson

 my\_map.json

Select a map type:

- Heat Map
- Dot Distribution Map
- Choropleth Map
- Voronoi Map
- Spike Map

Create Map

12  2 

35  1 

1  5 

©ExploreX

Account Manager



Edit your Profile



Let's make some changes

Username: hater456

Bio

I just hate people,  
especially that  
yolo girl.

Email: hatelife@gmail.com



Password: \*\*\*\*\*

Confirm Password: \*\*\*\*\*



©ExploreX

Account Manager Account Menu



Edit your Profile



Let's make some changes

Username: hater456

Bio

I just hate people,  
especially that  
yolo girl.

Email: haterlife@gmail.com

Password: \*\*\*\*\*

Confirm Password: \*\*\*\*\*

[My Profile](#)[Map Feed](#)[FAQ](#)[Log Out](#)

Submit

©ExploreX

Help



Frequently Asked Questions



Don't know where to start?

Question 1 ▼

Question 4 ▼

Question 2 ▼

Question 5 ▼

Question 3 ▼

Question 6 ▼

©ExploreX

Help Account Menu



Frequently Asked Questions



Don't know where to start?

Question 1 ▼

Question 4 ▼

Question 2 ▼

Question 5 ▼

Question 3 ▼

Question 6 ▼

My Profile

Edit Account

Map Feed

Log Out

©ExploreX







### Delete Map Modal



Delete Draft Map Modal



## Team Pink

Upload Import

Posts

Drafts



Are you sure you want to delete this map?  
This action is **PERMANENT**.

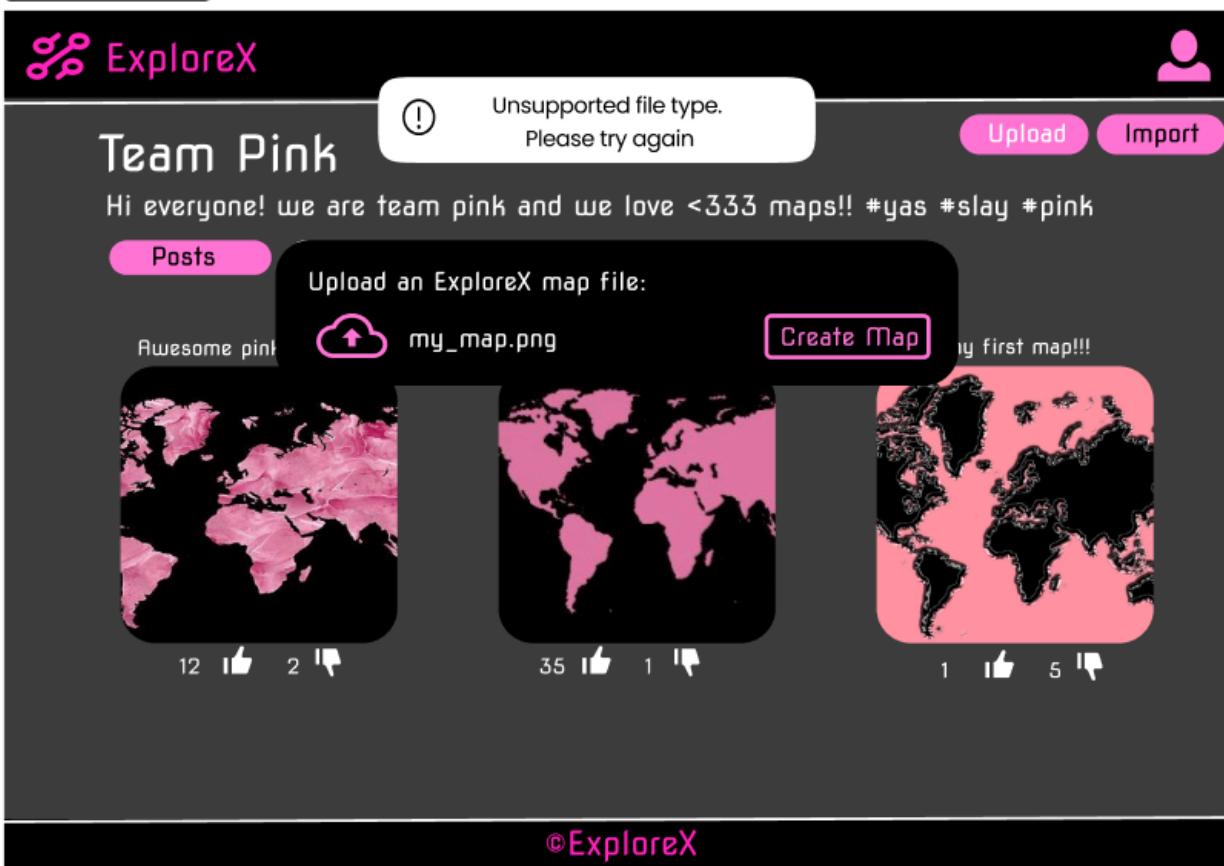


CONFIRM



©ExploreX

Upload File Error Modal



The screenshot shows the ExploreX mobile application interface. At the top, there is a navigation bar with the ExploreX logo, a user profile icon, and two buttons: "Upload" and "Import". A modal window titled "Upload File Error Modal" is displayed, containing the message "Unsupported file type. Please try again" with an exclamation mark icon. Below the modal, the main feed shows a post from "Team Pink" with the caption "Hi everyone! we are team pink and we love <333 maps!! #yas #slay #pink". The post includes a "Posts" tab, an "Upload an ExploreX map file:" section with a cloud icon and the file name "my\_map.png", and a "Create Map" button. Three world maps are displayed in cards: the first map is titled "Awesome pink" and has 12 likes and 2 dislikes; the second map has 35 likes and 1 dislike; the third map has 1 like and 5 dislikes.

©ExploreX

Import File Error Modal

# ExploreX

## Team Pink

Hi everyone! we are team pink and we love <333 maps!! \*yas \*slay \*pink

Posts

Awesome pink map!!!

Import a vector file below!

Accepted formats:

- Shapefile/Dbf
- KML
- GeoJson

my\_map.py

Create Map

Select a map type:

- Heat Map
- Dot Distribution Map
- Choropleth Map
- Voronoi Map
- Spike Map

12 14 2 1

35 14 1 1

1 14 5 1

©ExploreX

Login Error Modal

# ExploreX

[Don't have an account?](#)

## Welcome, Map Lovers

Mapping Your World, Your Way

Username or Email

dash@ui8.net

Password [Forgot password?](#)

Password

Sign in

Log in as Guest

Created by the pink team



Reset Password Successful Modal

[Don't have an account?](#)

# Welcome, Map Lovers

Username or Email

 Username

Password

[Forgot password?](#) Password

Button

[Sign in](#)

Button

[Log in as Guest](#)

! Check your email to recover your account

## Mapping Your World, Your Way



Created by the pink team

Reset Password Failed Modal

[Don't have an account?](#)

Account does not exist

## Forgot your password?

No worries, it isn't the end of the world.

Username or Email

 dash@ui8.netRecover

## Mapping Your World, Your Way



Created by the pink team

Register Username Error Modal

 ExploreX

[Login](#)



User already exists

## Let's get started

### Mapping Your World, Your Way



Username or Email

Password

Confirm Password

[Register](#)

Created by the pink team

Register Password Match Modal

 ExploreX

[Login](#)



Passwords do not match

# Let's get started

## Mapping Your World, Your Way



Username or Email

 dash@ui8.net

Password

 \*\*\*

Confirm Password

 \*\*\*\*

[Register](#)

Created by the pink team

Register Password Weak Modal

[Login](#)

# Let's get started

Mapping Your World, Your Way

! Password is too weak

Username or Email

Password

Confirm Password

[Register](#)



Created by the pink team

Save Changes to Account Modal



Edit your Profile

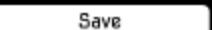


Let's make some changes

Username: ~~hate458~~Bio

I just hate people especially that yolo girl.

Save changes to Account?



Confirm Password: \*\*\*\*\*



©ExploreX

Account Manager Password Match Modal



The screenshot shows a modal window titled "Edit your Profile" with a user icon in the top right corner. A message bubble indicates "Passwords do not match". The main content area contains fields for "Username" (hater456), "Email" (hatelife@gmail.com), and "Password" (\*\*\*\*\*). Below the password field is a "Confirm Password" field with three asterisks. A large button labeled "Submit" is at the bottom right. On the left, there is a "Bio" section containing the text: "I just hate people, especially that yolo girl."

ExploreX

Edit your Profile

>Passwords do not match

Let's make some changes

Username: hater456

Email: hatelife@gmail.com

>Password: \*\*\*\*\*

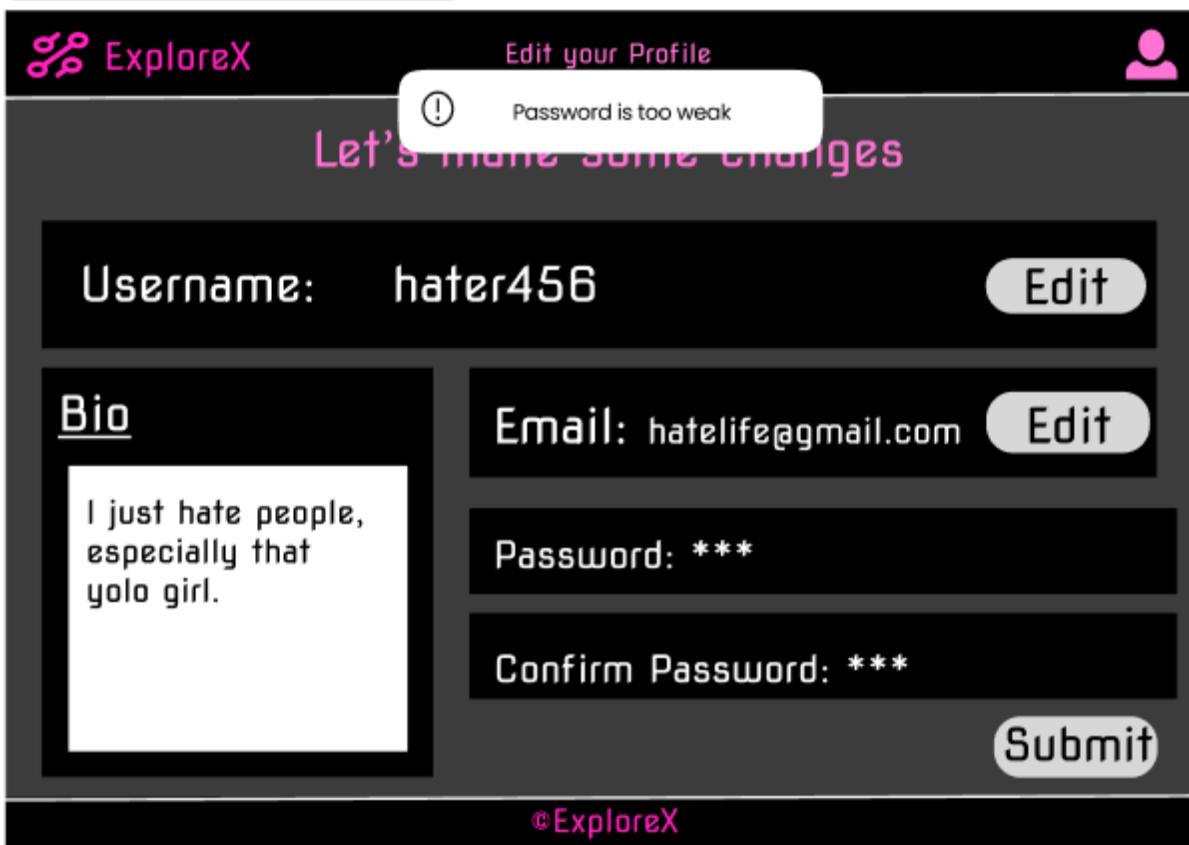
Confirm Password: \*\*\*

Submit

Bio

I just hate people, especially that yolo girl.

Account Manager Password Weak Modal



ExploreX Edit your Profile 

Let's make some changes

Username: hater456 

Bio

I just hate people,  
especially that  
yolo girl.

Email: hatelife@gmail.com 

Password: \*\*\*

Confirm Password: \*\*\*



©ExploreX

Account Manager Username Error Modal



ExploreX Edit your Profile 

Let's make some changes

Username: hater456 

Bio

I just hate people, especially that yolo girl.

Email: hatelife@gmail.com 

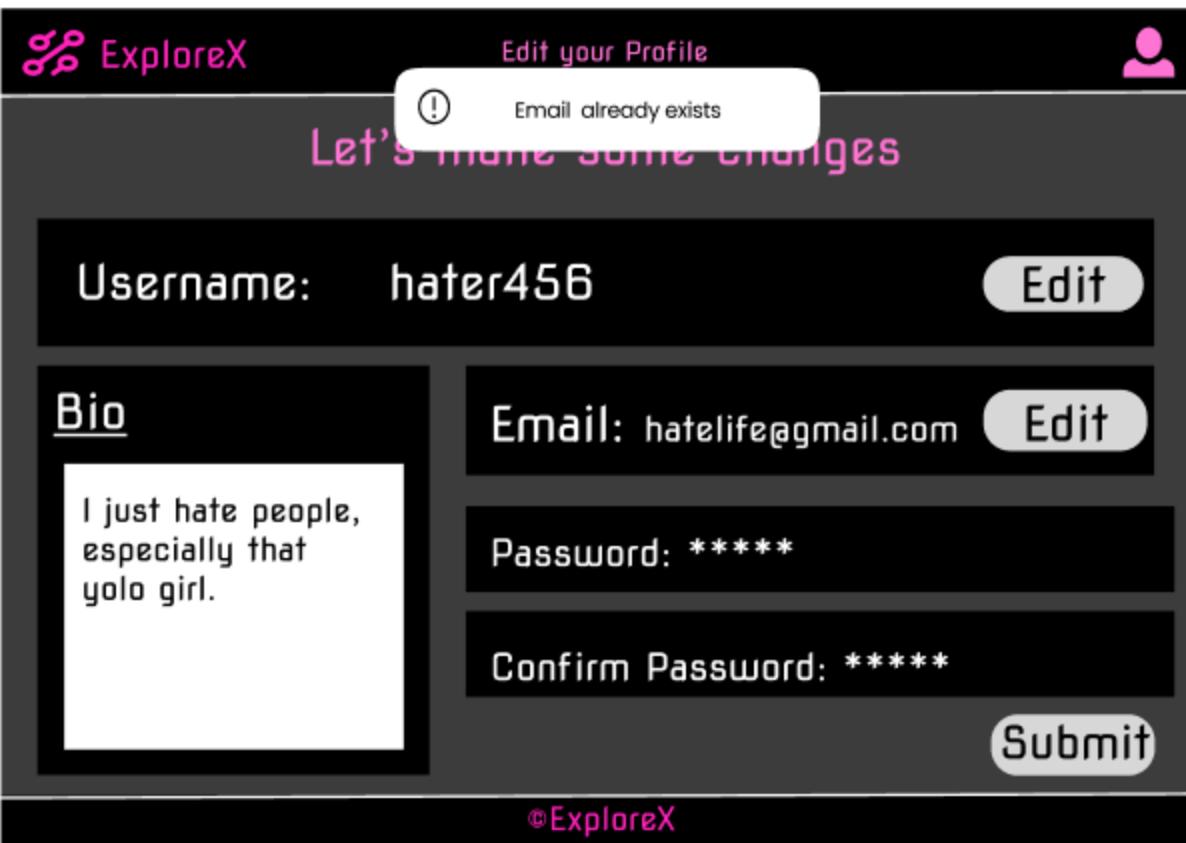
Password: \*\*\*\*\*

Confirm Password: \*\*\*\*\*



©ExploreX

Account Manager Email Error Modal



ExploreX Edit your Profile 

Let's make some changes

Username: hater456 

Bio

I just hate people,  
especially that  
yolo girl.

Email: hatelife@gmail.com 

Password: \*\*\*\*\*

Confirm Password: \*\*\*\*\* 

©ExploreX

### 3. Data Model

```
// Mongoose Schema for Users Collection
```

- username: { type: String, required: true, unique: true },
- email: { type: String, required: true, unique: true },
- password: { type: String, required: true },
- name: { type: String, required: true },
- mapsOwned: [{ type: mongoose.Schema.Types.ObjectId, ref: 'Map' }],
- creationDate: { type: Date, default: Date.now }

```
// Mongoose Schema for Maps Collection
```

- name: { type: String, required: true },
- type: { type: String, required: true },
- graphics: Object,
- description: { type: String },
- owner: { type: mongoose.Schema.Types.ObjectId, ref: 'User', required: true },
- isPublic: { type: Boolean, required: true },
- reactions: [{ type: mongoose.Schema.Types.ObjectId, ref: Reaction}],
- creationDate: { type: Date, default: Date.now }

```
// Mongoose Schema for Reactions Collection
```

- user: { type: mongoose.Schema.Types.ObjectId, ref: 'User', required: true },
- map: { type: mongoose.Schema.Types.ObjectId, ref: 'Map', required: true },
- isComment: { type: Boolean },
- isLike: { type: Boolean },
- isDislike: { type: Boolean },
- data: { type: String },
- interactionDate: { type: Date, default: Date.now }

## Data Dictionary (Sorted Alphabetically by Entity and Attribute)

Entity	Attribute	Type	Relationship	Constraint	Description
Maps Collection	_id	ObjectId			
	creationDate	Date			This will contain the date and time a map was created
	description	String			This will be the optional description given to a map
	graphics	Object			This is an object which contains each of the accessible graphic types of the specific map type (i.e. the attributes of this object depend on the map type)
	isPublic	Boolean		Required	This will be true when this map is public, false otherwise
	name	String		Required	This will be the name given to the map
	owner	ObjectId	References to User	Required	This will be a reference to the user object of the user who created the map
	reactions	Array of ObjectIds	References to Reactions		This will be an array of reaction objects that this map has
	type	String		Required	This will specify if the map is a heat map, dot distribution map, spike map, choropleth map, or voronoi map
Reactions Collection	_id	ObjectId			
	data	String			This will contain the string associated with a comment if isComment is true
	interactionDate	Date		Default: Current Date	This is the date the user interacted
	isComment	Boolean			This is a boolean representing whether or not a user commented
	isDownVote	Boolean			This is a boolean representing whether or not a user downvoted

	isUpVote	Boolean			This is a boolean representing whether or not a user upvoted
	map	ObjectId	References to Map	Required	This is the map the reaction is on
	user	ObjectId	References to User	Required	This is the user associated with the reaction that is made
Users Collection	_id	ObjectId			
	creationDate	Date		Default: Current Date	This is the date the user was created
	email	String		Required, Unique	This is a string representation of the users email
	mapsOwned	Array of ObjectIds	References to Map		This is an array of Map objects that the user owns
	name	String		Required	This is a string representation of the user's full name
	password	String		Required	This is a string representation of the user's unique password
	username	String		Required, Unique	This is a string representation of the user's unique username

## 4. System/Domain Analysis Model

### Executive Summary:

This project is a mapping application that addresses the growing need for customizable, data-rich maps. Our product seeks to revolutionize the way users interact with geographic data, setting itself apart from existing software by offering an intuitive user experience, advanced visualization options, and a wide range of map graphics. While there are several mapping tools available in the market, our vision is to provide a platform that empowers map enthusiasts to create highly personalized and informative maps easily. Companies such as Flourish and Google My Maps offer competing services that focus on the creation of custom maps but fail to recognize the value of community. We will focus on creating a community surrounding map graphics where users can build their own maps and build on top of other maps in the community. By providing value to the user in a way that allows them to express their data with advanced map customization, we aim to be the go-to choice for individuals seeking to visualize and analyze geographical data as well as join a community of fellow map-lovers.

### Objectives

1. Develop a user-friendly mapping application that enables users to create 5 distinct map graphic types (e.g., heat maps, choropleth maps, dot distribution maps) with ease.
2. Provide advanced customization options, allowing users to tailor map graphics to their specific data and needs.
3. Grant users the ability to export, share, and re-import maps.
4. Ensure seamless integration with different data sources and vector graphic formats.
5. Implement robust security measures to protect user data and map configurations.
6. Optimize performance and scalability to handle large datasets and user loads.
7. Provide descriptive and comprehensive explanations of the functionalities within our application.

### Strategies/Philosophies

1. **User-Centric Design:** We prioritize user needs and preferences in the development process to ensure a seamless and intuitive user experience.
2. **Material Design:** Create an immersive user interface by making it responsive, adaptive, clearly focused, graphic, etc.
3. **Agile Development:** We will follow an iterative and flexible approach to adapt to changing requirements and feedback from our client as well as new feature ideas during development.

4. **Open Data Integration:** Emphasis will be placed on allowing users to easily integrate open data sources and import/export maps specific to our application, enhancing the application's versatility.
5. **Modular Architecture:** We will design the application in a modular fashion to enable scalability and future feature additions.

## Constraints

1. Our application will only support vector graphics files and not raster graphics files.
2. Users will not be able to edit the geometry of maps.
3. Users won't be able to collaboratively edit maps.
4. Imported files can only be of administrative divisions (i.e. not roads, canals, etc.).
5. Imported files must adhere to specific structures using proper naming conventions, must not be corrupt, and must be below the file size threshold.

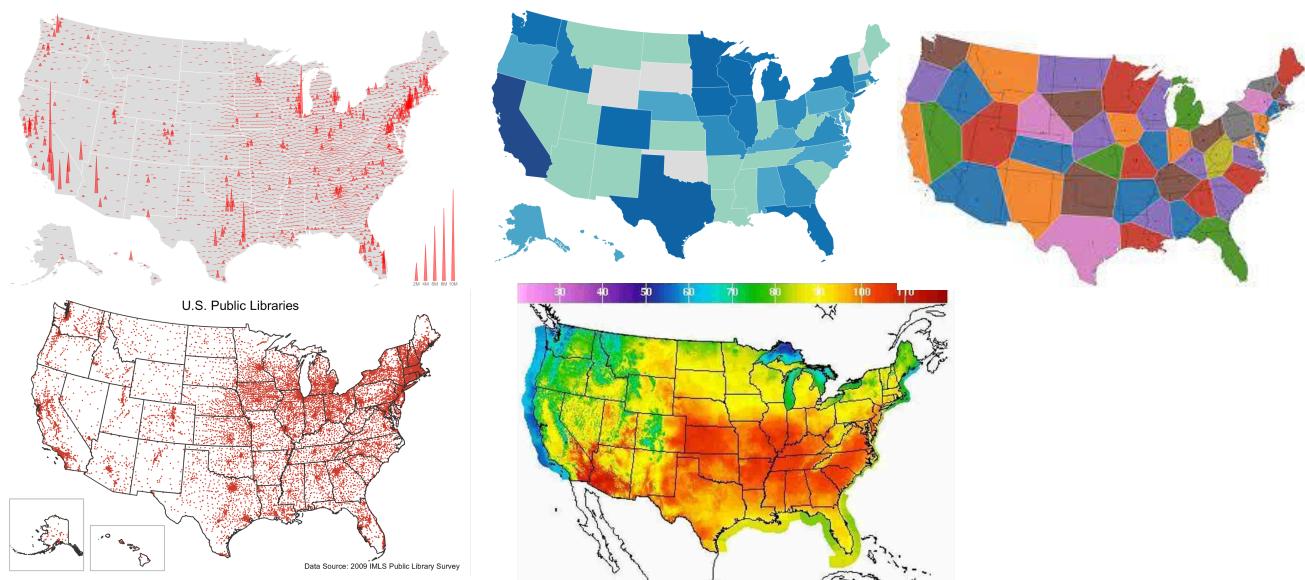


Figure 1: A spike map, choropleth map, Voronoi map, dot distribution map, and heat map

### Heat Maps:

Our heat maps will represent data with colors that correspond to a specific value or density. They could be used to visualize patterns, concentrations, or variations in data.

### Choropleth Maps:

Our choropleth maps will use colors or shading to represent data values for different geographic areas, such as countries, states, or regions. They could be used for showing election results, income levels, or disease prevalence by area.

### **Dot Distribution Maps:**

Our dot distribution maps will use dots or symbols to represent individual data points or events at their specific geographic locations.

### **Voronoi Maps:**

Our voronoi maps involve calculating regions based on proximity to specified points or features and then overlaying them onto a geographic area for spatial analysis and decision-making, offering insights into resource allocation and facility optimization.

### **Spike Maps:**

Our spike map represents geographic data by placing spikes on a map to display specific points of interest or data values at particular locations. It is useful for highlighting localized data trends, anomalies, or concentrations within a geographical context.

## **Actors**

1. **Admins:** Full access to all features, user management, and system settings.
2. **Registered Users:** Ability to create, customize, and share maps. Access to collaboration and community interaction features.
3. **Guest Users:** Limited access to view public maps and basic features.

## **Services**

1. **Map Creation:** Users can create, customize, and save maps using accepted data formats and application custom formats.
2. **Map Classification and Search:** Users can find and search for other public maps uploaded by users using properties and keywords.
3. **Map Editing:** Users can edit their own maps through various controls such as naming, coloring, etc.
4. **Map Forking:** Users can fork public maps on the application and use/customize it as their own map.
5. **Collaboration:** Tools for users to share maps and export maps in a custom file format that can be reuploaded for editing.
6. **Account Management:** Users can manage their account information such as username, and password, and recover their accounts in the case of a forgotten password.
7. **Community Interactions:** Users can communicate with other users through actions such as liking/disliking and commenting on public maps.
8. **Application Support:** Provide comprehensive explanations of how to use the main functionalities of the application.

## 5. Software Model

### Similar Problems

Similar problems that the team has worked on and may utilize as an aid in our development process include:

- The Playlister (application developed in CSE 316)
- The Top 5 Lister (MERN application developed by TheMckillaGorilla on GitHub)
- MERN stack application tutorials by Javascript Mastery Youtube channel
- ObservableHQ tutorials on developing/designing maps with D3.js
- MongoDB Authentication tutorials for identifying users

### Complete Technology Set

- Javascript, HTML, CSS for developing the code of the application
- MongoDB/Mongoose for storing and retrieving data that will be used on our application and interacting with the database
- Express as a Middleware
- React to develop the frontend of the application
- Node.js to create the run time environment
- react-router-dom for handling react routing in the application
- jsonwebtoken for assigning JWTs to users
- Leaflet API for loading vector files into JS maps
- L.KML.js as a leaflet KML layer plugin for rendering KML files
- Shp.js and Catiline.js APIs for rendering shapefiles
- leaflet-choropleth plugin for generating choropleth maps
- heatmap.js plugin for generating heatmaps in leaflet
- D3.js library for generating spike maps, voronoi maps, and dot distribution maps
- OpenStreet API for rendering the base maps
- Nodemailer for sending emails in JS (for password reset)
- Handlebars.js to create semantic HTML templates (to send in the password reset email)
- Material UI library for creating frontend components that follow the material design pattern
- Bcrypt.js for encrypting passwords (security)
- Pusher API for creating a live commenting system

### Training Verification

Jacob: I have completed a course on software development at Stony Brook University (CSE 316). In this course, I developed a MERN based web application over the course of several iterations. Starting from vanilla javascript and building up to a full scale application using

MongoDB, Express, React, and Node.js. Before creating the final application, I practiced designing the entire application using UML diagrams before building it. In addition, I've developed a full stack web application with a team as part of a startup which connected university students to local landlords. We utilized Python (Django) for the back end and Javascript (React) for the frontend. These experiences/training has prepared me to complete the project at hand with ExploreX.

Amelia: During the 2022 Fall semester, I took CSE 316 which gave me the basis for MERN stack development and will be heavily used throughout our project. Throughout the semester, we were asked to create different variations of the Playlister program, starting with vanilla javascript and ending with a full stack application using the MERN stack, which was a web application to where users could create their own playlists of songs on Youtube and share them to the other users of the application. The final product took a month and a half to finish, from the beginning of November to the end of the semester and has strengthened my skills with these technologies.

Juan: I have a solid foundation in the technology set required for this project. I took a class, CSE 316, which covered the MERN stack extensively, focusing on MongoDB, Express, React, and Node.js. In addition during this class we also learned about the software development lifecycle, design patterns, and creating UML charts. This course provided me with a comprehensive understanding of these technologies and their integration. During my internship, I worked with a team to create components for a web application using Salesforce and React. This experience allowed me to apply my React skills in a real-world setting. Moreover, I have personal projects to my credit, where I've developed web applications using React, further enhancing my expertise in the framework.

Jesse: The main training I want to highlight to validate my skills is CSE 316, a course that instructs the usage of MERN stack through the creation of a web application. As a class, we each developed our own implementation of Playlister, a web application that allows users to create playlists, share playlists, and interact with the community. The main tools that were utilized were MongoDB, Express, React, and Node.js. This course also instructed us through the software development life cycle, including the creation of UML diagrams and schemas for our data. With this knowledge in mind, I am well prepared to handle the creation of ExploreX and be a contributing member to the team.

### **Front-End Component Composition**

([https://lucid.app/lucidchart/de33a114-434b-45c7-bf00-39ccb2aa5740/edit?view\\_items=y0loiuK0ibhU&invitationId=inv\\_a8a87804-1356-4062-bc34-e24bbceeb995](https://lucid.app/lucidchart/de33a114-434b-45c7-bf00-39ccb2aa5740/edit?view_items=y0loiuK0ibhU&invitationId=inv_a8a87804-1356-4062-bc34-e24bbceeb995) )

### **Front-End Component Design**

([https://lucid.app/lucidchart/63e10eff-0aec-44f7-9471-e09546987487/edit?viewport\\_loc=-2890%2C-2265%2C8467%2C3994%2C0\\_0&invitationId=inv\\_dd85774f-4f36-4671-a25b-260f58615d69](https://lucid.app/lucidchart/63e10eff-0aec-44f7-9471-e09546987487/edit?viewport_loc=-2890%2C-2265%2C8467%2C3994%2C0_0&invitationId=inv_dd85774f-4f36-4671-a25b-260f58615d69))

## Back-End API

### Auth

([https://lucid.app/lucidchart/11c6abe7-696c-463b-a809-372d157b8335/edit?viewport\\_loc=2082%2C-459%2C4992%2C2355%2CHWEp-vi-RSFO&invitationId=inv\\_33b905b3-bfa2-4968-9c6c-08cb9ad7c0de](https://lucid.app/lucidchart/11c6abe7-696c-463b-a809-372d157b8335/edit?viewport_loc=2082%2C-459%2C4992%2C2355%2CHWEp-vi-RSFO&invitationId=inv_33b905b3-bfa2-4968-9c6c-08cb9ad7c0de))

### Store

([https://lucid.app/lucidchart/6fc76615-6258-4f58-8c36-6ff818a35dc5/edit?view\\_items=bLwmjqnA8j9&invitationId=inv\\_13c5d763-0848-4aa8-a6c4-d5f81995031b](https://lucid.app/lucidchart/6fc76615-6258-4f58-8c36-6ff818a35dc5/edit?view_items=bLwmjqnA8j9&invitationId=inv_13c5d763-0848-4aa8-a6c4-d5f81995031b))

## 6. Progress Reviews

### Build 1: Infrastructure Establishment and Deployment

In our initial build, we focused on laying the foundation for our mapping application. The critical tasks included choosing render.com for our deployment needs, deploying a bare-bones full-stack project, and implementing thorough front-end and back-end tests to ensure the application's stability. Simultaneously, we set up a CI/CD pipeline to facilitate seamless collaboration within the team using GitHub actions. The culmination of this build was the creation of a detailed schedule, providing a roadmap for the subsequent phases of development.

### Build 2: User Interaction and Core Screens

Build 2 centered on enhancing user interaction and introducing core screen functionalities. We implemented features such as Splash/Login, Register, Forgot Password, and FAQ screens, subjecting each to rigorous testing. Additionally, we mapped out the Feed, Public Map View, and App Bar functionalities, integrating client tests for comprehensive validation. Editing screens for Choropleth, Dot Dist, Spike, Voronoi, and Heat maps were introduced, alongside the development of User Profile and Account Management screens. Cypress testing was used for each front-end feature introduced.

### Build 3: User Management and File Handling

Build 3 marked a shift towards user management and efficient file handling. We established a robust user authentication system, including user schema, registration, login verification, and logout functionalities. Account Management received focused attention, with tasks covering username, bio, email, and password editing. Profile Page Management was refined, ensuring accurate reactions display and allowing users to delete or edit draft maps. Commenting functionalities were introduced, enabling users to interact with maps through comments. Importantly, a native file type converter was implemented, allowing seamless integration of geoJSON, KML, and shape files into our system. Testing using Jest and Supertest for our server-side features was introduced. Cypress, Jest, and Supertest testing continued for each new feature in the application.

### Build 4: Dynamic Map Generation and Display

Build 4 marked a significant milestone in our project, focusing on generating and displaying dynamic maps. Leveraging the turf and leaflet libraries, we successfully implemented

Choropleth, Dot Dist/Spike, Heat, and Voronoi maps. The maps were dynamically generated, complete with legends for corresponding values. Display features were enhanced to include crucial information such as the title and username of the map creator, along with captions. Routing and password reset functionalities were addressed to ensure a smooth user experience. Importantly, we implemented an abstract import system, enabling users to accept and process KML, JSON, and SHP/DBF files seamlessly. Continued testing for each new feature.

## Build 5: Map Interaction and Management

In Build 5, our focus shifted to enhancing user interaction and refining map management functionalities. We reworked reactions, connecting them seamlessly with the backend, and reintroduced likes/dislikes to the map feed cards. The ability to delete both public and private maps was implemented, ensuring proper removal from the backend and associated feeds. We introduced map publishing and forking, allowing users to convert maps to published or duplicate them with different users. Furthermore, we improved the display of maps on the public feed and profile pages, implemented search functionality, implemented map export, and warned users of potential file corruption issues in import. Testing was generated for each new feature.

## Build 6: Feature Refinements and Finalization

Our final build focused on miscellaneous features and refinements to ensure a polished mapping application. Specific rendering options were tailored for each map type, providing users with unique editing options. We introduced the Save and Publish functionality, along with dynamic legend rendering, enhancing user experience. Editing for Choropleth, Dot Dist/Spike, Heat, and Voronoi maps were meticulously addressed, allowing users to modify values, legends, colors, and sizes with ease. The Undo/Redo feature was implemented using JsTPS Transactions, ensuring the application's robustness. This build also incorporated specific editing details for each map type, such as size adjustments, dynamic legend generation, and adding/removing points in Voronoi maps. The culmination of these tasks resulted in a comprehensive and user-friendly mapping application, offering versatile functionalities for users' mapping needs.

## 7. Post Mortem

### Project Overview

**Objective:** Our goal was to develop a web application that allows users to create, edit, and share map-based graphics using vector files. The application aimed to serve as a community platform for map enthusiasts, researchers, and educators, facilitating the creation and sharing of custom map graphics.

**Technologies Used:** MERN stack (MongoDB, Express.js, React, Node.js), with additional libraries for handling vector files like SHP/DBF, GeoJSON, and KML.

### Achievements

**1. Robust Account Management:** Implemented a secure account system, integrating content creation with public/private settings and ownership details. Added functionalities for password recovery and user authentication.

**2. Map Graphic Creation and Editing:**

- Enabled users to upload SHP/DBF, GeoJSON, and KML files, or fork existing maps on the site.
- Developed an advanced map editing interface, allowing users to name regions, attach custom data properties (strings, numbers, colors), and modify map graphics.
- Implemented a graphics editor for decorating maps with texts, colors, and legends, essential for detailed map customization.

**3. Map Graphics Exporting:** The application supports exporting maps in PNG, JPG, and a custom native JSON format, catering to diverse external usage scenarios.

**4. Map Classification and Search:** Integrated a search functionality to enable users to easily find and utilize public maps, based on various properties and metadata.

**5. Community Interaction Features:** Facilitated user interactions through forums for map discussions, error reporting, and collaboration opportunities.

### Challenges and Solutions

**1. Technical Complexity in Map Editing:**

- **Challenge:** The map editing feature, being the core of the application, presented significant technical hurdles, especially in handling various vector file formats and customizing map properties.

- **Solution:** We utilized specialized libraries for vector file processing and invested in iterative UI/UX designs to make the map editing user-friendly and efficient.

## 2. Performance Optimization:

- **Challenge:** Handling large map files and ensuring smooth interaction in the map editor was challenging.

- **Solution:** Implemented lazy loading and efficient data handling techniques to optimize application performance as well as setting an upper bound file upload size.

## 3. User Engagement and Feedback:

- **Challenge:** Initially, user engagement was lower than expected, with feedback indicating a need for more intuitive navigation and features.

- **Solution:** Conducted user testing sessions to gather feedback, leading to several UI/UX improvements and the introduction of tutorial guides within the application.

## Lessons Learned

- **User-Centric Design:** The importance of continuous user feedback in shaping the application cannot be overstated. User testing should be an ongoing process throughout the development cycle.

- **Scalability Planning:** Anticipating the need for scalability, especially when dealing with large data sets like map files, is crucial for long-term success.

- **Team Collaboration:** Effective communication and collaboration among team members were key to overcoming technical challenges and meeting project deadlines.

## Future Directions

- **Enhanced File Processing:** Improving the application's performance and usability with larger datasets/files through compression algorithms.

- **AI-Driven Map Analysis:** Integrating AI to offer advanced analytics and insights based on map data.

- **Community Building:** Focusing on strategies to grow and engage the user community, including organizing mapping challenges and a ranking system.

## Conclusion

This project was a significant learning experience for our team, blending complex technical challenges with the need for a deep understanding of user requirements. The application has successfully established a platform for map enthusiasts and professionals to create, share, and collaborate on map-based projects. Moving forward, we aim to enhance the application based on user feedback and emerging technological trends in the field of map graphics and community-driven platforms.

## Appendix A: All Meeting Minutes

Sep 12, 2023 2:30 PM

Attendees: All team members

### Notes

- Discussed GIS File Formats (Vector vs. Raster).
- Explored Shapefiles, GeoJSON, and KML formats.
- Practical Experimentation:
  - Used DIVA-GIS to download and experiment with shapefiles.
  - Utilized Mapshaper for shapefile editing.
  - Generated GeoJSON via GeoJSON Maps; used geojson.io.
- Explored Keyhole Markup Language (KML) and its uses.
- Reviewed Project Requirements: local file loading, map rendering, navigation.
- Agreed on hands-on approach for file format learning.
  - Work on your own time to get an understanding of file formats.
- Assigned individual research/experimentation tasks.
- Emphasized timely progress and meeting deadlines.
- Scheduled Next Meeting for progress updates.
  - Agreed to just meet Tuesday after class every week

### Action items

- All Team Members
  - Research further into GIS file formats and gather resources for reference.
  - Experiment and mess around with file formats.
  - Think about web application architecture and functionality

Sep 19, 2023 2:30 PM

Attendees: All team members

### Notes

- User Base:
  - Potential users: map enthusiasts, researchers, educators, and developers.
  - Use cases: reports, social media, and publications for geographic data.
- Map File Constraints:
  - File extension constraints
  - Think about conversions.

- Required Services:
  - Customer's broad requirements: accounts, map creation/editing, navigation, exporting, and community interactions.
    - Liking, disliking, commenting
- Map Graphics Understanding:
  - Chose 5 map types: Choropleth, Heat, Voronoi, Dot and Spike Maps
  - Make sure these are feasible
- Set up:
  - ExploreX is our name
  - Assigned document and who's doing what section.
- Executive Summary - **Jesse**:
  - Some existing products we found are Google Earth and Google Maps
  - Think about what separates us.
    - Emphasize **community**
- Objectives - **Amelia**:
  - Refer to the homework document for functionality
- Strategies/Philosophies - **Jacob**:
  - Modularity, look at the class notes for more examples
- Constraints - **Jacob**:
  - Identified project limitations, emphasizing map graphic types.
  - Main one is file limitations and editing limitations
    - We cannot change the map (regions and borders) other than styling
- Actors & Services Definition - **Juan**:
  - Started listing user categories and their privileges.
    - Most likely just Guest and Logged In users
  - Began defining services for better clarity.
    - Authentication, Exporting, refer to homework document for list
- Hosting services for app
  - Render?

#### Action items

- All Team Members
  - Check feasibility of your assigned map
    - Jesse: Choropleth
    - Amelia: Heat
    - Juan: Voronoi
    - Jacob: Dot and Spike
  - Work on your assigned part
  - Explore hosting services
- Jesse
  - Add Github Discord bot to track progress

Sep 26, 2023 2:30 PM

Attendees: All team members

## Notes

- Use Case Model:
  - Separate based on different sections.
  - Review 316 Use Cases also for examples
- Authentication - **Jesse**
- Profile Page and Editing - **Jacob**
- Map Feed and Public Map View - **Amelia**
- Map Editing - **Juan**
- Also, work together to help each other with miscellaneous things that may have been forgotten.
  - Fork should be in the public map view
  - Upload/Import should go in the Profile Page
- Finish screens and use case names by **tomorrow**
- Then finish defining each use case.
- For the next homework, use Figma

## Action items

- All Team Members**
  - Work on your respective part
- Amelia:**
  - Make a figma and share it with everyone for the next homework.

Oct 3, 2023 2:30 PM

Attendees: All team members

## Notes

- Reiterated application's vision and service alignment with UI design.
- Discussed human perception's role in UI design, especially in element proximity and meaning association.
- Emphasized the importance of balance, symmetry, regularity, predictability, sequentiality, economy, unity, proportion, simplicity, and groupings in UI composition.
- Reviewed key UI design principles: alignment, balance, clarity, compatibility, comprehensibility, configurability, consistency, control, directness, efficiency, familiarity, flexibility, forgiveness, predictability, recovery, responsiveness, simplicity, and uniformity.

- Explored options for UI mockup tools, considering Figma, Moqups, Adobe XD, Cacoo, and The Wireframer. Decided on Figma.
- Discussed high expectations for UI mockup diagrams and the need for a visually pleasing, well-organized design.
- Suggested consulting Google's Visual Style Guide and considering Material Design philosophy for effective UI design.
- Planned to add a 'User Interface Model' directory on Google Drive with a Google Doc containing all UI views.
- Agreed to create a table listing all application views with columns for view number, name, and a brief description.
- Decided to provide mockup renderings filled with example data for each listed view, accommodating all user interactions.

#### Action items

- Split up design into different categories of the site to work on in parallel.
- Decide on a theme and styling framework.
- Creating Figma mockups for each use case.

Oct 10, 2023 2:30 PM

Attendees: All team members

#### Notes

- Start with an introduction to the product vision from the System Analysis, setting the stage for the presentation's story arc.
- Outline the main problem the product aims to solve, establishing a connection with the audience (the customer).
- Transition into a discussion of key product use cases, explaining how they informed the UI model creation.
- Avoid detailed Use Case diagrams; instead, opt for a more narrative explanation of significant Use Cases.
- Integrate content from the User Interface model, using visuals to demonstrate how the UI addresses user needs.
- Highlight how the proposed solution aligns with the customer's needs and addresses the identified problem.
- Include visuals and examples from the UI model to illustrate key points and enhance audience understanding.
- Conclude with a summary that reiterates the product vision and how the proposed solution fulfills it.

- Prepare for potential questions from the audience, focusing on areas like user experience, technical feasibility, and project timeline.
- Ensure the presentation is well-rehearsed, concise, and within the 20-25 minute timeframe.

#### Action items

- Create a pink themed slide deck (matching our app theme color). - **Jacob**
- Decide on each member's talking points and presentation slides needed to complete.
- Complete presentation slides in parallel. - **Everyone**
- Rehearse the presentation a couple times to ensure smooth transitions and completion timing. - **Everyone**

Oct 12, 2023 2:30 PM

Attendees: All team members

#### Notes

- Finalize and document Mongoose schemas for all necessary collections ensuring all fields are properly defined with types and constraints.
- We will be using three schemas (Map, User, Reaction).
- Review and validate the relationships between different collections, especially the references among Users, Maps, and Reactions.
- Ensure that all schema fields are accurately reflected in the Data Dictionary, with clear descriptions and constraints for each attribute.
- Cross-check the Data Dictionary for consistency, making sure all entries are alphabetized and correctly linked to their respective entities.

#### Action items

- Decide on how each model is connected to each other (one-to-many, one-to-one, etc.) - **Jesse, Jacob**
- Specify the type of each field, if it will be required, and any limitations/restrictions for it. - **Amelia, Juan**
- Schedule a review session to collectively go over the Data Model and Data Dictionary for completeness, accuracy, and adherence to project requirements. - **Everyone**

Oct 24, 2023 2:30 PM

Attendees: All team members

## Notes

- Decide on our extensive technology stack (i.e. which libraries will be used for what, front-end app, back-end app, etc.). We'll be using a base MERN stack with create-react-app amongst a list of libraries mentioned in the software model.
- We each have similar experience levels with the technologies we will be using which allows for everyone to be on the same page.
- We will utilize a similar software design to a previous project we'd all completed at Stony Brook University as our template and work from there.
- We will use Lucid charts for collaborating on UML diagrams.
- Starting with the front-end composition diagram, then go to auth diagram, then store diagram, and end with the back-end design diagram.

## Action items

- Decide on component structure for front-end and model it in the UML diagram.
- Finalize routing between front-end and back-end. - **Juan, Amelia**
- Finalize component state, props, and functions for each component in the application. - **Jacob, Jesse**
- Create UML diagrams for all requirements. - **Everyone**

Oct 31, 2023 2:30 PM

Attendees: All team members

## Notes:

- Discussed various server deployment options including AWS, Render.com, and Digital Ocean.
- Explored the process of deploying a bare-bones full-stack project, emphasizing the need for scalability.
- Planned front-end tests using Cypress; discussed coverage and integration testing.
- Explored back-end testing strategies, considering frameworks like Supertest and Jest.
- Initiated the setup of a CI/CD pipeline for automated testing and deployment, with a focus on automating tests on GitHub Actions.

## Action items

- Server Deployment: **Jacob**
- Full-stack Project Deployment: **Jesse**
- Front-end Tests: **Amelia**
- Back-end Tests: **Jacob**

- CI/CD Setup: **Juan**

Nov 7, 2023 2:30 PM

Attendees: All team members

Notes:

- Provided progress updates on individual tasks from Build 1.
- Discussed the implementation details of Splash/Login, Register, Forgot Password, FAQ screens.
  - Reviewed Figma UI models and discussed what kind of components in Material UI we can take advantage of to create these pages.
  - Attaching states to texts and setting up a store reducer to generalize our states.
- Outlined a general edit map screen that would suit the needs of each specific map our app would provide.
  - Agreed that we should add states to all the dynamic components inside the side panel to simplify future builds.
- Emphasized the importance of client tests, specifically focusing on user interactions and error handling.

Action Items

- Client Tests: **Everyone**
- Splash/Login, Register, Forgot Password, FAQ: **Jesse**
- Map Feed, Public map view, App bar: **Jacob**
- Edit Screens: **Juan**
- User Profile, Account Management: **Amelia**

Nov 14, 2023 2:30 PM

Attendees: All team members

Notes:

- Reviewed progress on Build 2 tasks.
- Discussed User Authentication tasks, including potentially updating the user schema, login verification, and guest login.
  - Thought of ideas on how to implement recovering your password. Using Google Cloud to implement the email sending.

- Explored Account Management tasks, focusing on editing username, bio, email, and password.
- Delved into Profile Page Management, covering post display, draft management, and error handling.
- Initiated discussion on Commenting tasks and the creation of a native file type.
  - For native file type, we agreed that having one type of map would be beneficial, therefore we would need to decide on the file type for our maps.
  - Geojson's ease of use with leaflets makes it the likely candidate.

Action Items:

- User Auth Tasks: **Jesse**
- Account Management Tasks: **Jacob**
- Profile Page Management Tasks: **Amelia**
- Commenting Tasks: **Amelia**
- Create Native File Type: **Juan**

Nov 21, 2023 2:30 PM

Attendees: All team members

Notes:

- Reviewed progress from Build 3.
- Discussed the intricacies of generating Choropleth, Dot Dist / Spike, Heat, and Voronoi maps.
  - Looked through different libraries that could be of use for creating such maps.
  - Libraries that came up: d3, turf.js, and leaflet plugins.
- Explored the Display Maps functionality, including the title, username, and caption display.
- Initiated discussions on routing, password reset, and abstract import/upload features.
  - Looked at different npm libraries to convert shapefiles and KML files into geojson.
  - Libraries we found: @togeojson, shapefile
    - Discussed testing the libraries to make sure they can convert properly.
- Agreed to complete display map, routing, and import earlier in the week to facilitate map generation.

Action Items:

- Generate Choropleth Map: **Jesse**
- Generate Dot Dist / Spike Map: **Jacob**
- Generate Heat Map: **Amelia**
- Generate Voronoi Map: **Juan**
- Display Maps: **Amelia and Juan**
- Routing / Pass Reset: **Jesse**
- Abstract Import/Upload: **Juan and Amelia**

Nov 28, 2023 2:30 PM

Attendees: All team members

Notes:

- Provided updates on progress from Build 4.
- Discussed Rework Reactions, Delete Map, Publish Map, Fork Map functionalities.
  - With backend setup these functionalities would be easier to implement, we agreed to have more tasks to complete this build.
- Explored showing maps on the public feed, on profile pages, and implementing search functionality.
  - Complete community interactions within the app so that we can focus on map editing in the next build.
- Initiated discussions on exporting maps and handling corrupt files.

Action Items:

- Rework Reactions: **Amelia**
- Delete Map (Public and Private): **Juan**
- Publish Map: **Jacob**
- Fork Map: **Jesse**
- Show Maps on Public Feed: **Jacob and Jesse**
- Show Maps on Profile Page: **Amelia and Juan**
- Search Map: **Jacob**
- Export Map, Corrupt Files: **Juan**

Dec 5, 2023 2:30 PM

Attendees: All team members

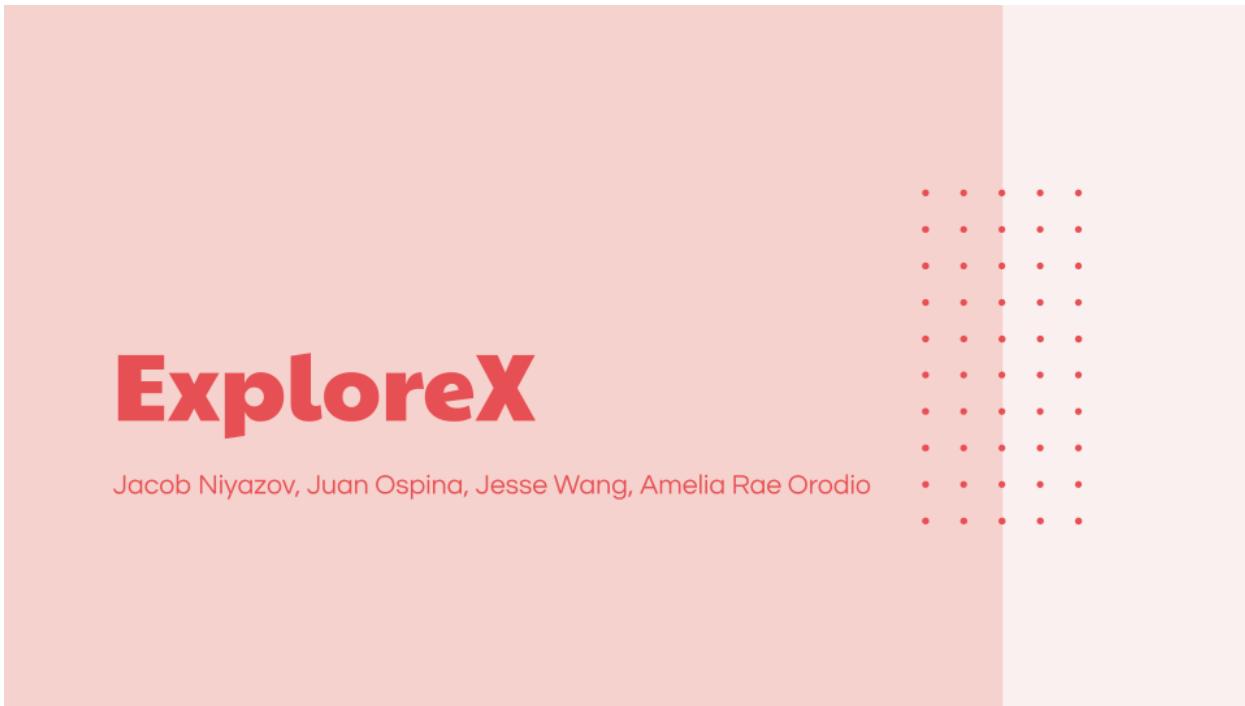
## Notes:

- Reviewed progress from Build 5.
- Discussed miscellaneous tasks, including map rendering, saving, and legend rendering.
  - Discussed bugs that have shown up and how we can address them.
  - Decided that we do not need to deal with maps bigger than 10 MB and it will reduce leaflet performance significantly rendering it useless.
- Explored editing Choropleth, Dot Dist / Spike, Heat, Voronoi maps with specific rendering options.
  - Each member continues to work on the map type previously worked on.
- General editing for colors and borders is to be implemented first then we can dive into type-specific features.
- Initiated discussions on Undo/Redo functionality.
  - Use the previous 316 as a base implementation for our undo redo functionalities.

## Action Items:

- Miscellaneous: **Everyone**
- Editing Choropleth: **Jesse**
- Editing Dot Dist / Spike: **Jacob**
- Editing Heat: **Amelia**
- Editing Voronoi: **Juan**
- Undo/Redo: **Amelia**

## Appendix B: Requirements Presentation Slides



The contents slide has a light beige background with a red vertical bar on the right side containing a grid of red dots. At the top center, the word 'Contents' is written in a large, bold, red sans-serif font. Below it, a numbered list of 11 items is presented in a smaller red font:

1. Meet The Team
2. The Problem
3. Product Vision
4. Our Objectives
5. Strategies and Philosophies
6. Constraints
7. Custom Map Types
8. Simplicity of Use
9. Advanced Features
10. Finished Product
11. Conclusion

## Meet the Team

### 01. Jacob Niyazov

Software Engineer

### 03. Jesse Wang

Software Engineer

### 02. Juan Ospina

Software Engineer

### 04. Amelia Rae Orodio

Software Engineer

## The Problem

Map enthusiasts lack user-friendly software for creating visually appealing and data-rich maps. Existing options are insufficient, and we need an intuitive solution that enables users to easily design, share, and enhance maps without complexity. This software will serve to meet the unique demands of map enthusiasts, enabling them to express their creativity and deliver engaging, data-driven map experiences.

## Product Vision



### User-Centric

We aim to provide a safe and loving community for map-lovers and enthusiasts



### Advanced Visuals

Revolutionary interactions with maps that have never been seen before



### The Difference

Competitors exist, but do not stress and value the idea of community

## Our objectives



Develop a user-friendly mapping application



Provide advanced customization options, allowing users to tailor maps to their needs



Grant users the ability to export, share and re-import maps.



Ensure seamless integration with different data sources and vector graphic formats



Provide descriptive and comprehensive explanations of the functionalities within ExploreX



Optimize performance and scalability to handle large datasets and user load

## Strategies and Philosophies

### User Centric Design

Prioritize user needs in the development process to ensure a intuitive user experience.

### Material Design

Create an immersive user interface that is responsive, adaptive, clearly focused, graphic, etc.

### Agile Development

Follow an iterative and flexible approach to adapt to changing requirements and feedback from our clients.

### Open Data Integration

Emphasis on allowing users to integrate open data sources and import/export maps specific to our application.

### Modular Architecture

Design the application in a modular fashion to enable scalability and future feature additions.



## Constraints

### 01. Map Graphics

Only vector graphics are supported

### 03. Map Geometry

Map geometry is fixed and not editable



### 02. Collaboration

Users will not be able to edit maps together in real time

### 04. File Support

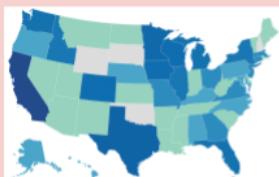
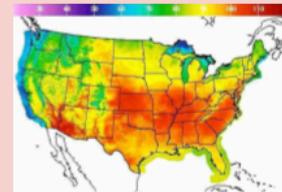
Only KML, geoJSON, shape files, and our custom map files will be supported

## Custom Map Types

01.

### Heat Maps

Our heat maps will represent data with colors that correspond to a specific value or density.



• • • •  
• • • •

### Choropleth Maps

Our choropleth maps will use colors or shading to represent data values for different geographic areas, such as countries, states, or regions.

02.

03.

## Custom Map Types

### Dot Distribution Maps

Our dot distribution maps will use dots or symbols to represent individual data points or events at their specific geographic locations.



• • • •  
• • • •

### Voronoi Maps

Our voronoi maps involve calculating regions based on proximity to specified points or features and then overlaying them onto a geographic area for spatial analysis and decision-making.

04.

## Custom Map Types

05.

### Spike Maps

Our spike map represents geographic data by placing spikes on a map to display specific points of interest or data values at particular locations.



And more to come...

06.



## Simplicity of Use

Select 1 of 5  
Map Types

01.

Upload a Map

02.

Customize

03.

Publish

04.

Save For Later

05.

Enjoy

## 1. Upload a Map

Pick from the variety of accepted file formats

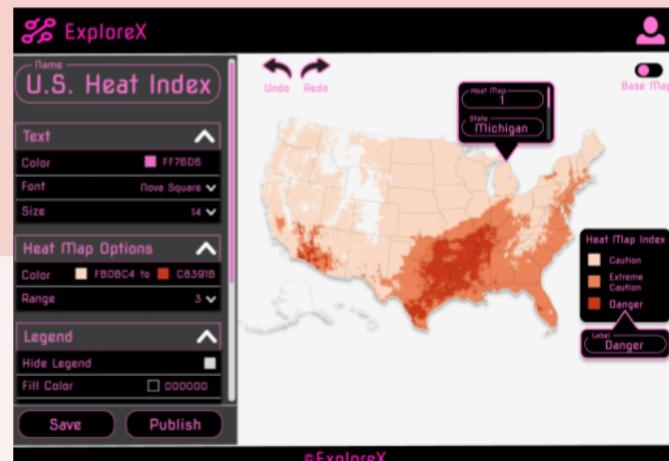


## 2. Select 1 of 5 Map Types

We support Heat, Dot Distribution, Spike, Voronoi, and Choropleth Maps

## 3. Customize

With each map comes different editing features. Freely customize your map the way you'd like



## 4. Save For Later

Save your edits to work on them later



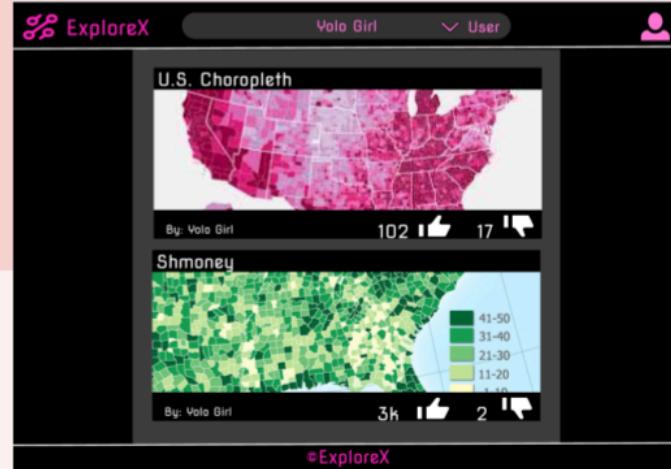
## 5. Publish

Publish a map on ExploreX with ease



## 6. Enjoy !

This platform is open to browse other user's published maps



## Advanced Features

ExploreX  
Native  
Sharing

01. Map forking
- 02.
03. Community Interaction

04. Account Customization

## 1. Map forking

Start off where another user left off. Fork any public map in the community.



## 2. ExploreX Native Sharing

Utilize our native file type to import/export your maps and even share maps with friends



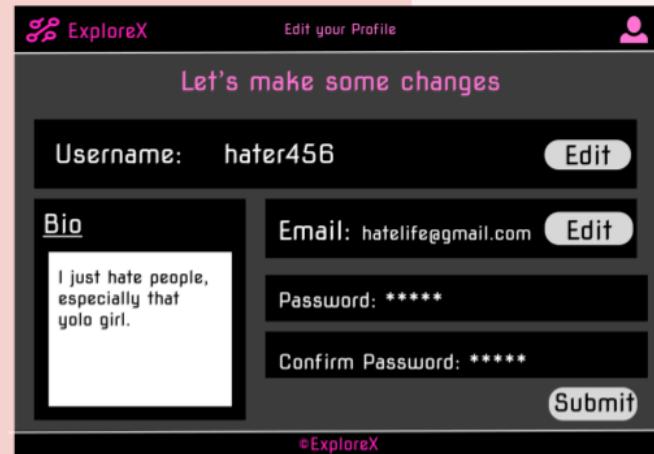
## 3. Community Interaction

Liking, disliking, and commenting on other maps is in the palm of your hands



## 4. Account Customization

Customize your username, email, bio, and change your password all on one screen.



Let's make some changes

Username: hater456 [Edit](#)

**Bio**  
I just hate people, especially that yolo girl.

Email: hatelife@gmail.com [Edit](#)

Password: \*\*\*\*\*

Confirm Password: \*\*\*\*\*

[Submit](#)

## Finished Product

### User-Friendly

Our application will be accessible to both novice users and experienced map enthusiasts.



### Visualization Tools

Create captivating, data-rich maps that engage and inform your audience.



### Effortless Sharing

Easily share your maps with a global audience and showcase your work and insights to a wider community.



### Community of Enthusiasts

Share your creations and insights while fostering collaboration among like-minded individuals.



## Next Steps



### Develop

Once we've completed designing our product, we will begin developing the application (~7 weeks)

### Test

Extensive testing will be conducted on the product throughout the development lifecycle

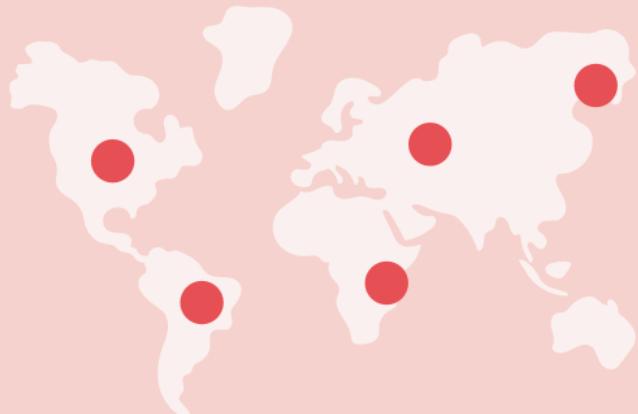
### Deploy

Finally, we will deploy the functional application to the web for the world to use

## Conclusion

As a team from  
Explore X,

We thank you for your  
time and hope you join us  
in revolutionizing the  
mapping community



# Thanks



Do you have any questions?

[Support@ExploreX.com](mailto:Support@ExploreX.com)

+1 010 101 0101

[ExploreX.com](http://ExploreX.com)

Please don't actually contact us

## Appendix D: Detailed Description of Team Member Contributions

### Jacob Niyazov

In my role on the project, I focused on front-end and back-end development, creating key features that significantly enhanced user interaction and experience. My notable contributions included developing the map feed, which displays public maps as interactive cards, allowing users to navigate and filter maps based on specific properties. I also built a comprehensive user account portal for personalizing account details, and in collaboration, streamlined the process for creating general maps with legends. My individual expertise was further showcased in developing dot distribution and spike maps, adding depth to our application's mapping capabilities. Additionally, I implemented a feature for publishing maps from drafts to the map feed, encouraging community engagement and showcasing our application's versatility.

Moreover, I was responsible for the deployment and continuous integration of our web application, ensuring seamless updates from our GitHub repository to our hosting platform. This role required a thorough understanding of deployment technologies and a meticulous approach to maintaining the application's integrity. I also led the initiative to migrate both the front end and backend to custom domains, addressing a critical cookie-transfer issue. This move not only enhanced the security and reliability of user sessions but also demonstrated my capability in managing complex aspects of web application deployment. My efforts in both development and deployment were instrumental in ensuring that our application was not only functional but also robust and user-friendly, directly impacting the project's success.

### Amelia Rae Orodio

In my role on the project, I served as a key contributor to the development team, with a primary focus on user interface design and functionality implementation. A significant accomplishment was the creation of a sophisticated profile screen that seamlessly integrates various elements. This screen adeptly displays a user's essential information, including their username and bio, along with a comprehensive showcase of their posts and drafts. I took charge of designing and implementing personal map cards. This effort aimed to enhance the user interface by providing users with a visually appealing snapshot of their map-related activities. Additionally, I worked on improving the functionality, including the delete and edit buttons on those cards, enabling users to efficiently manage their content.

Moreover, I played a pivotal role in the implementation of crucial features related to map functionality. I actively participated in the development of reaction handling mechanisms on both the map feed and public map view screens, fostering user engagement. Collaborating closely with other team members, I contributed significantly to the creation and refinement of the "create

map" modal, ensuring a smooth and intuitive experience for users. Beyond that, my involvement extended to the intricate tasks of heat map creation and editing, incorporating functionalities such as undo/redo capabilities, heat legend editing, and information presentation for heat maps. Furthermore, my versatile contributions encompassed extensive support on both the front and back end, showcasing my commitment to delivering a comprehensive and polished project.

Jesse Wang

In my role within the project, I took the responsibility of architecting and implementing the user authentication system, creating seamless user interactions within our application. I had to make secure and user-friendly login/logout functionalities, ensuring a smooth and reliable authentication process using libraries such as Bcrypt and Crypto. Leveraging industry best practices, I also created the register and password recovery process via emails which allowed users a fail safe system in case of forgotten passwords.

Furthermore, my role extended to augmenting the project's functionality by integrating advanced features. One main one was forking maps that existed. This allowed users to create a copy of their own maps for them to edit. The main map editing I worked on was Choropleth maps. I took the responsibility of the entire choropleth editing working on various features within such as color editing, property fields management, and legend responsiveness. I streamlined the experience by implementing an efficient saving mechanism, empowering users to save their work and come back to it. Moreover, I spearheaded the integration of a robust routing system, facilitating efficient navigation across various sections. This allowed the user to visit various links through the URL instead of being forced to click through our application. Additionally, I contributed to enhancing user experience by creating a comprehensive FAQ section, answering common questions a user may have.. Through a blend of strategic planning, technical expertise, and attention to user-centric details, these accomplishments collectively ensured a robust, user-centric, and feature-rich application experience.

Juan Ospina

In my capacity on the project, I spearheaded the implementation of a robust CI/CD pipeline using GitHub Actions, significantly enhancing our development workflow. This involved automating our test cases to ensure that only code passing rigorous testing would be deployed, guaranteeing the stability and reliability of our application. My meticulous approach to continuous integration and deployment played a pivotal role in maintaining the project's integrity and facilitating seamless updates from our GitHub repository to our hosting platform.

Additionally, I contributed to the overall efficiency of our deployment process, aligning with best practices in the industry.

Furthermore, I played a crucial role in designing and developing the foundation of the edit map screen, a feature that greatly improved user interaction with our application. The seamless integration of the side panel allowed users to effortlessly interact with maps, loading properties from the GeoJSON onto the side panel for easy editing, creation, and deletion. My work extended to the backend, where I implemented core functionalities for maps, including creation, updating, and retrieval. I also played a pivotal role in enabling users to export in various formats, such as native files and images and import maps ensuring compatibility with different file types like SHP/DBF, KML, GeoJSON, and the Native File Type. Additionally, my expertise was crucial in the implementation of Voronoi maps, where I added features like point addition and deletion, contributing to the diverse and feature-rich nature of our application.