

ArtPI

Team: put_me_to_REST

Roster: Jesse “McCree” Chen (PM), Kelvin Ng, Eric “Morty” Lau, David Xiedeng

~~~~~

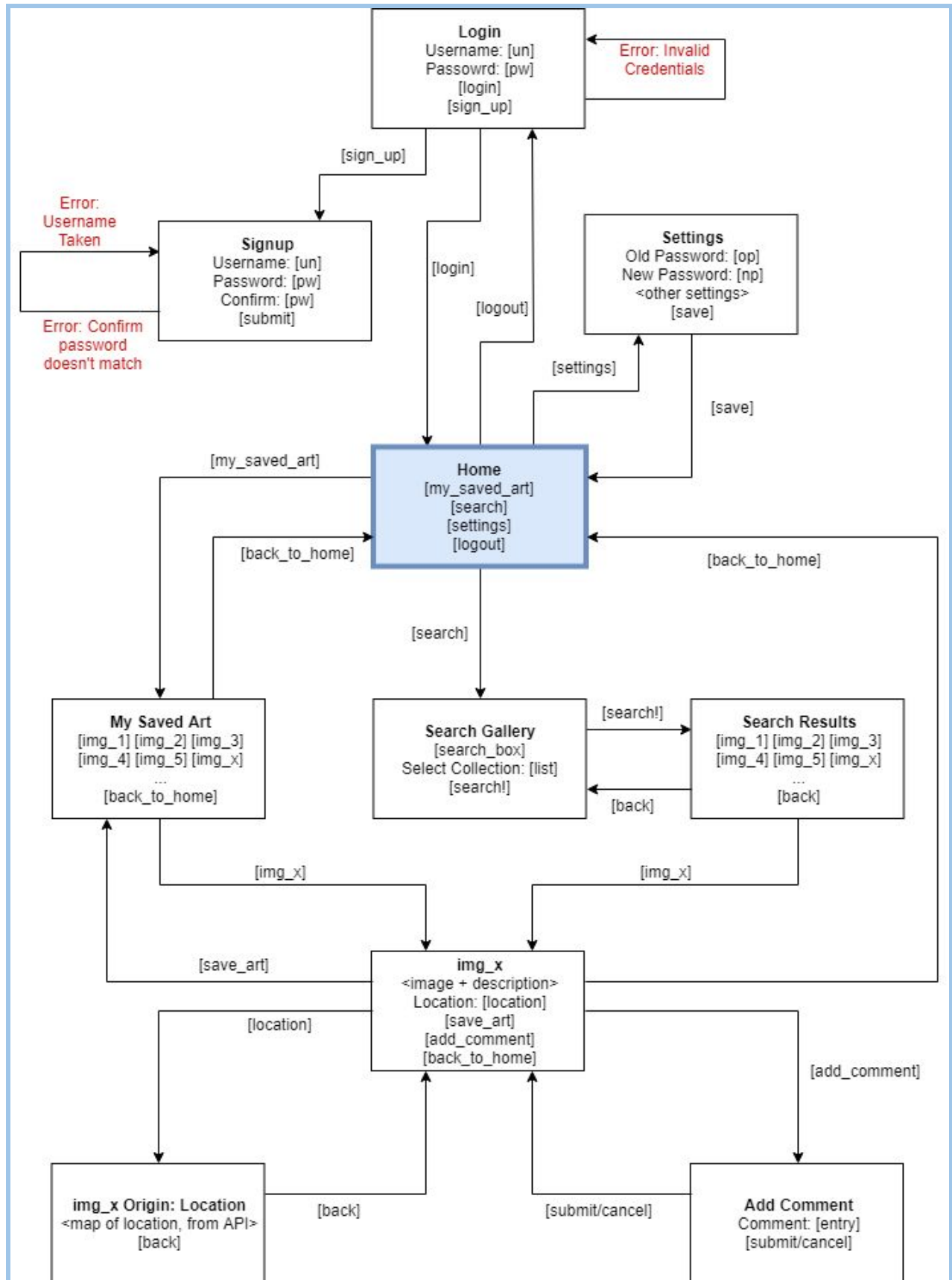
~

## **Overview**

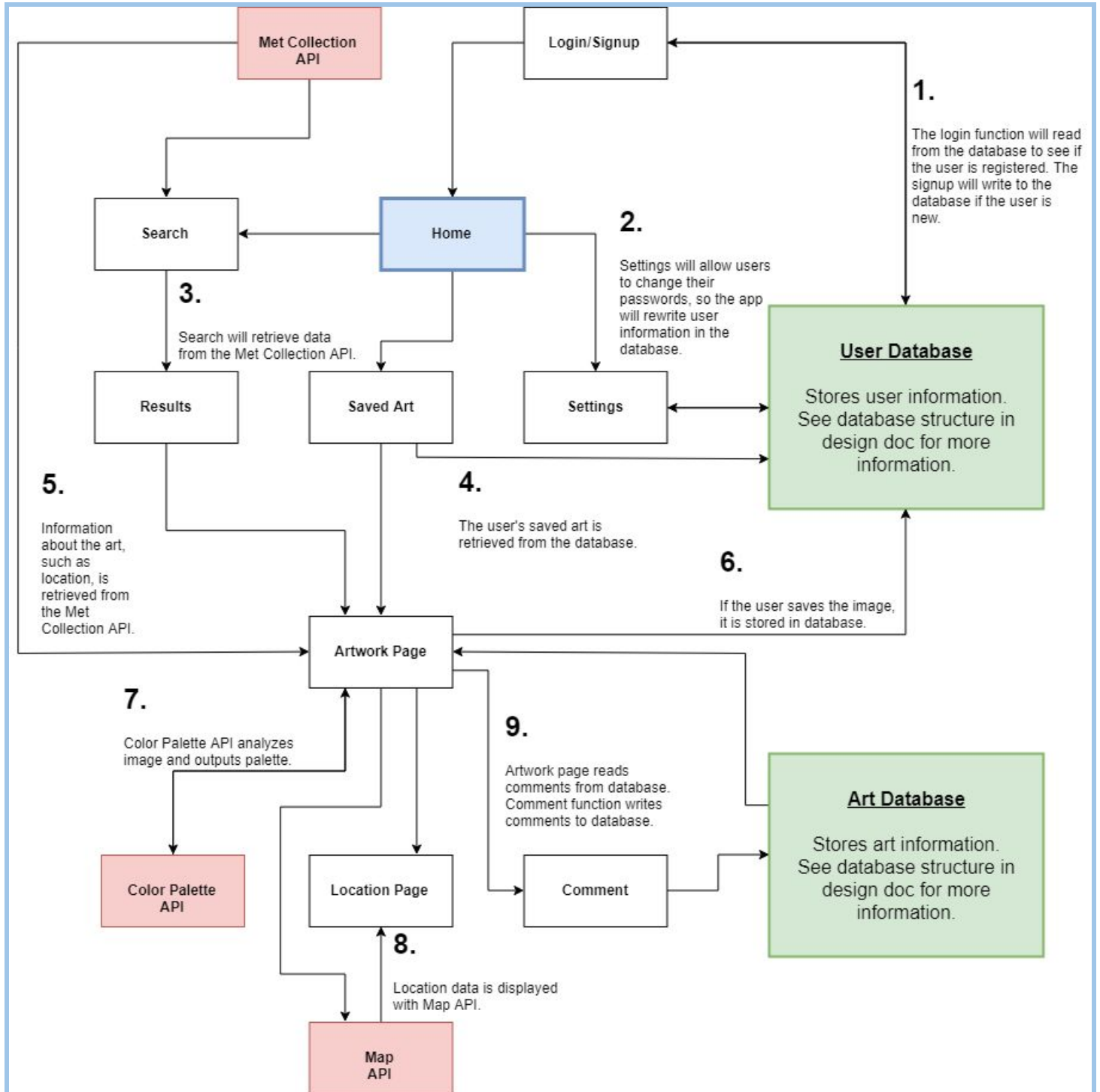
Our website provides a way for users to browse artworks from the Metropolitan Museum of Art’s collection. Below are some of the features of our website:

1. Create Account
  - a. Upon visiting the site, users can login or create an account to access the website’s services
2. Search Gallery
  - a. Type in keywords and the website will search through the Met’s collections and display the results. Clicking on a piece will take the user to the art’s description page.
3. View (Saved) Art
  - a. By clicking on a piece of art from search results, the user will be taken to a page with descriptions of the art. In that page, the user will also be able to:
    - i. Leave a comment
    - ii. Save the art
    - iii. Locate the “birthplace”, if available
    - iv. Determine the color palette
4. Comment
  - a. Under the artwork and descriptions, users can see other users’ comments and leave a comment.
5. Save Art
  - a. By saving the art, the user can quickly find it again by going to their homepage and viewing all their saved work.
6. Locate Origin
  - a. The Met’s API provides the creation place of their works, if known. By using this information and inputting it to our map API, we are able to generate a map of the location.
7. Color Palette
  - a. By inputting the image file into a color palette identifying API, we can provide information on the colors in the work. If the user is an artist or just studying art, it would be a helpful tool for them.
8. Settings
  - a. Users can change their passwords.

## Sitemap



## Component Map



## Database Structure

### 1. User Database

This stores information about user accounts and their private information.

(user TEXT, password TEXT, saved\_art BLOB)

| User         | Password         | Saved Art (objectID)* |
|--------------|------------------|-----------------------|
| exampleUser1 | examplePassword1 | [208218, 41293]       |
| ...          | ...              | ...                   |
| #string      | #string          | #list of integers     |

\* Conveniently, the Met database has an “objectID” for every item in their collection. So, instead of referring or saving them with their names, we can just a short number to get the item.

### 2. Art Database

This stores artwork, their information, and comments.

(artID INTEGER, comment TEXT, user TEXT, timestamp BLOB)

| objectID | Comment                                            | User         | Timestamp             |
|----------|----------------------------------------------------|--------------|-----------------------|
| 208218   | Such a nice vase!                                  | exampleUser1 | 11--16--2019:10:05:31 |
| 41293    | Hate the sunflower ;(                              | trollUser2   | 11--17--2019:21:43:02 |
| 208218   | @exampleUser1,<br>disagree. It looks too<br>murky. | exampleUser2 | 11-17-2019:22:00:17   |
| ...      | ...                                                | ...          | ...                   |
| #integer | #string                                            | #string      | #import datetime      |

## Role Assignments

Jesse “McCree” Chen (PM): Design document, project flow, coding here and there

Kelvin Ng: Back-end database and front-end implementation of Met API

Eric “Morty” Lau: Front-end, CSS, Bootstrap, Implementation of Imagga API

David Xiedeng: Back-end database and front-end implementation of Map Static API