Sprint 2 Submission

CS5500 Project

Group Name: Fire Emblem

Group Members: Xinyi Zhang, Zeyu Shen

***GitHub Project Board*: https://github.com/users/xinyisherryz/projects/1**

***Git Repo*: https://github.com/xinyisherryz/cs5500-fire-emblem**

***Trello Board*: https://trello.com/b/Rk1GKh9W/cs5500-fire-emblem**

*** Delivery of stories:**

1. Understand daily activities and insights

   We developed features in `ActivityService` class to retrieve daily activity durations for a specific type of activity and a summary of all activity types with their summed-up durations. These methods process the activity and present it in a structured format, allowing users to gain insights into their daily activities.

2. Identify the most visited locations

   We designed a method in `ActivityService` class to pull the top 10 most visited locations by checking each valid record and ranking the visited frequency. They can also access a list of all visited locations. This information helps users understand their preferences and daily routines.

3. View activities breakdown and trends

   In the `ActivityService` class, users can use a method to view the breakdown of their activities by type and also understand the trends by aggregating the total duration for each category. Based on this information, users can set their practice goals and manage their exercise time.

4. User account management

We created the `UserService` class which provides methods for users to manage accounts, including creating, retrieving, updating, and deleting their personal information. It empowers users with greater control over their account details.

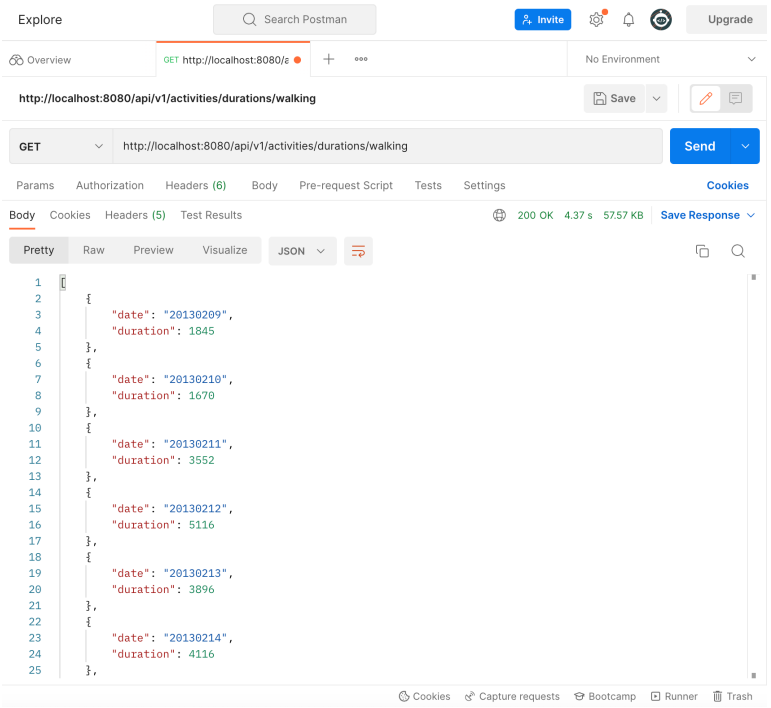
5. Simplify data operations via APIs

   Both activity and user related classes offer a suite of methods that simplify the data operations, making it easier for developers to upgrade the platform and for users to interact with the application.

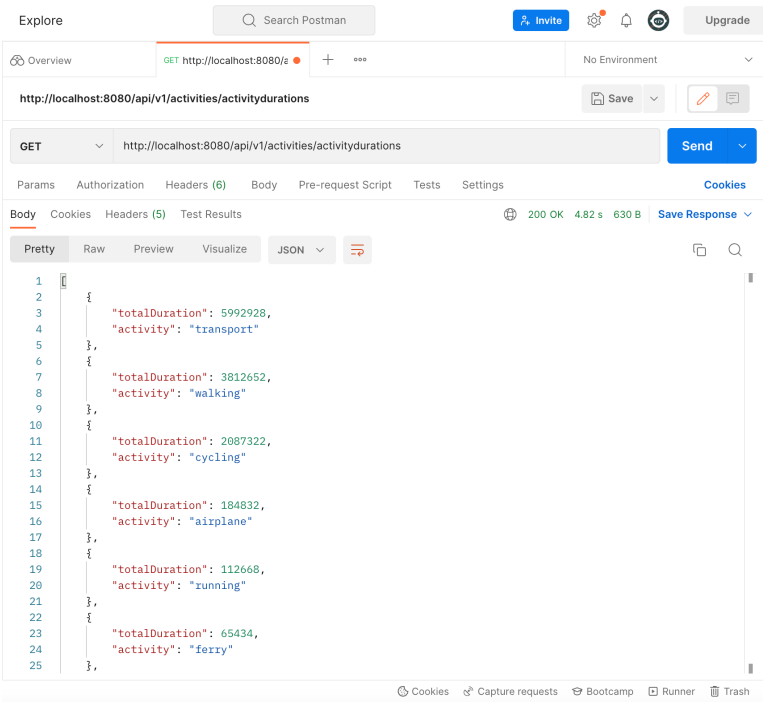
*** Functionality of Deliverable:**

The system has achieved its intended functionalities as we desired. To prove that we have successfully delivered the stories, we choose to use Postman to test each functionality. Since we already provided the screenshots of testing the basic CRUD operations for the activity class in the previous submission, we will only attach the proof for the advanced methods and user related operations.

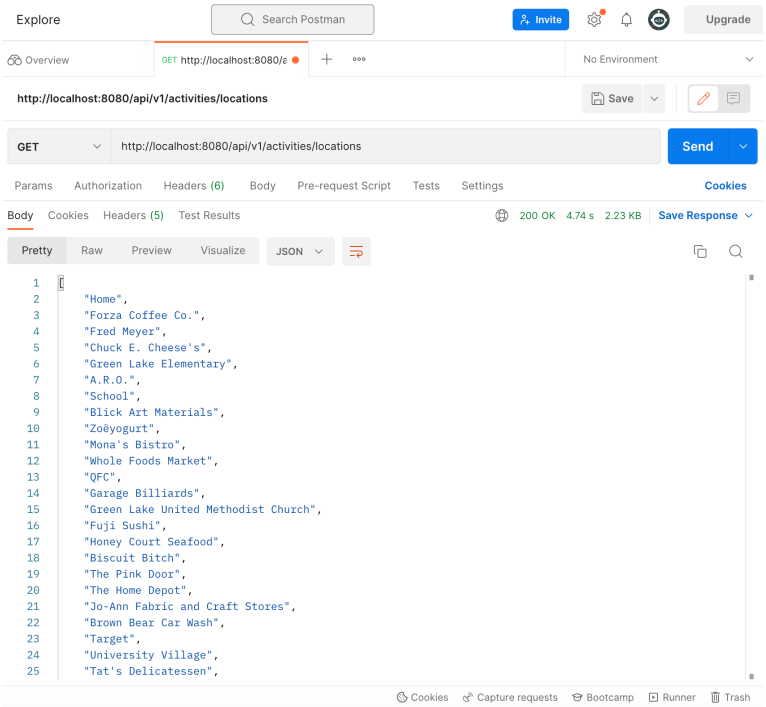
1. Retrieve daily activity duration for a specific type of activity



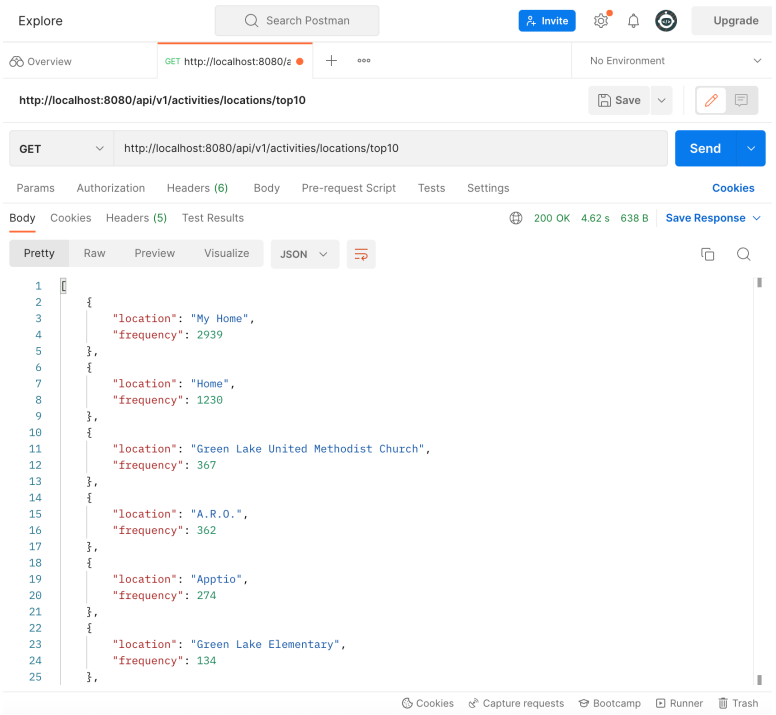
2. Get a summary of all activity types with their summed-up durations



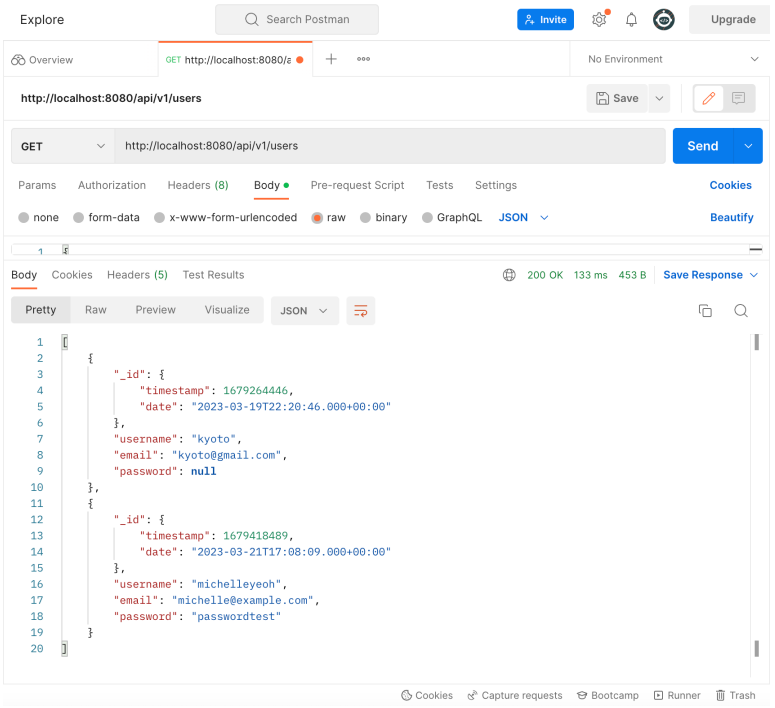
3. Return a list of all locations the user has visited



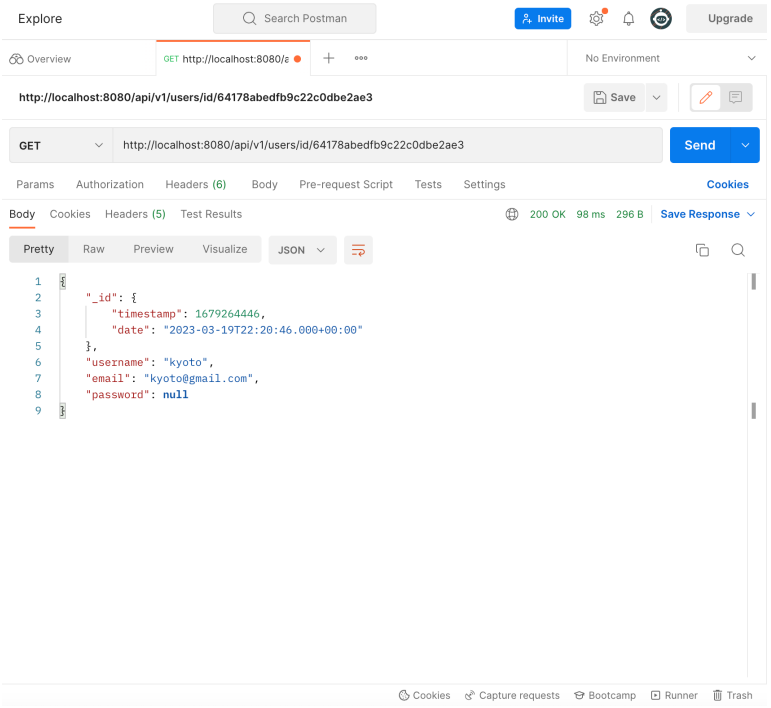
4. Provide the top 10 most visited locations with the frequency



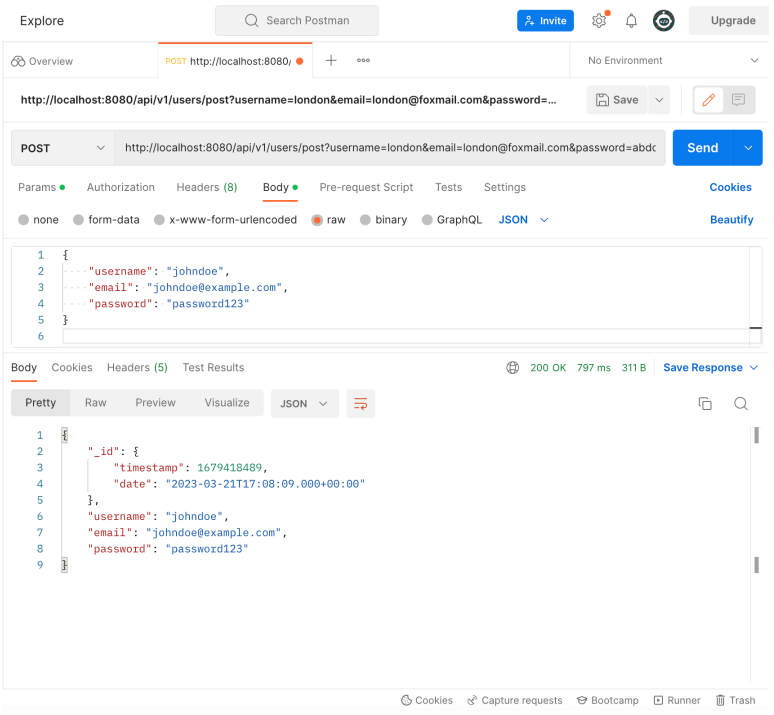
5. Fetch a list of users in the database

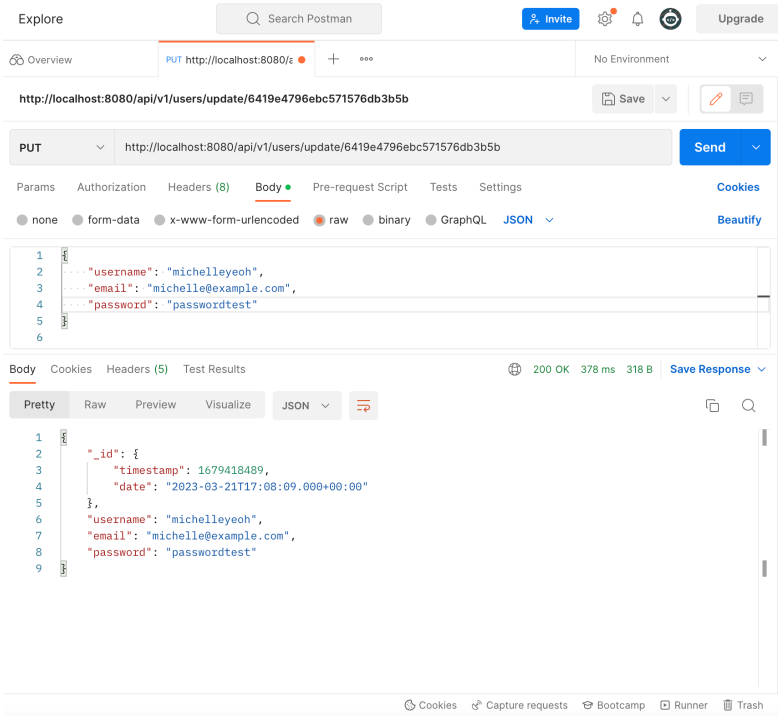


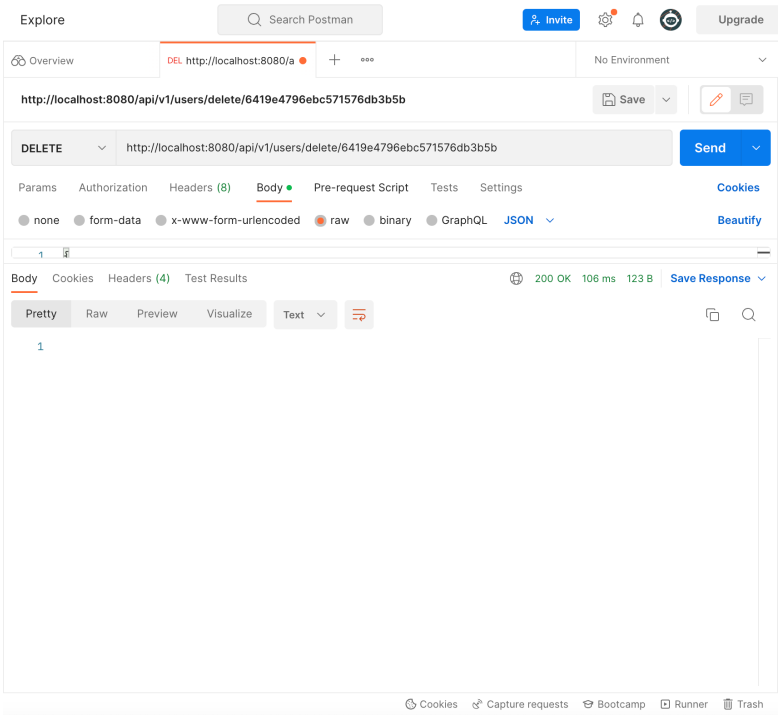
6. Check a specific user account given the id



7. Create, update, and delete user information







* Code Quality

We have tested our code quality using CodeMR and confirmed that the codebase has positive and robust attributes.

1. Coupling (100% Low): The modules are well-encapsulated and do not rely on each other, making them maintainable for developers to modify and reuse.
2. Complexity (49% Low, 51% Low-Medium): Given its level, we can assume that any developers that have previously not worked on our project can also easily make changes to the code.
3. Lack of Cohesion (100% Low): Regarding high cohesion, our modules and classes successfully contain all the relevant functions together, instead of spreading through the codebase.
4. Size (49% Low, 51% Low-Medium): A codebase with such size is convenient for developers to manage and work with.