Sri Lanka Institute of Information Technology

Programming Applications and Frameworks (IT3030)

Continuous Assignment – 2023, Semester 1

Initial Document

A picture containing text, clipart, vector graphics

Description automatically generatedGROUP ID:Cyan\_145

IT 20708276 Navaratne U.D.K

IT 21054686 Hewage R.P

IT 21035876 Kiriella K.G.A.K

IT 21007538 Maddumage P.W

# Contents

This document should give a brief introduction to the assessment panel on how you are planning to implement the assignment. This document should **not** exceed seven (7) pages in length. The following details are expected.

1. A very brief project description
2. The functional requirements for the REST API and the client web application separately.
3. The non-functional requirements for the REST API and the client web application separately.
4. An overall architecture diagram for the entire system you are contracted to design and implement (you can ignore the mobile applications).
5. A detailed architecture diagram(s) for the REST API.
6. A detailed architecture diagram(s) for the client web application. You should decide upon a suitable frontend architecture.
7. References
8. Brief Project Description

A web application called "CYAN" enables users to write food reviews and distribute them across other platforms. Today, many people have taken up the interest of capturing images of their food and posting them online. This platform is very important in bringing together all food lover. On the other side, when customers publish a feedback about their experiences, this platform promotes in the marketing of food outlets.

This app platform was created through using spring boot and react technologies. The platform will be established with both technical and non-technical users in mind, so all users will be able to easily use the user interface. Users may register on the platform, log in, and then start posting their reviews along with photographs and captions. The privacy of user accounts will be well-created with effective security. The owner of the post has the ability to edit and delete published food reviews. Users can see other users' posts and leave likes and comments on them, which promotes a sense of community on the platform. Also, there are options to follow other people whose viewpoints they respect and sign up for notifications when new reviews are published. The platform discusses a user's past reviews and ratings to recommend new places they might like while also allowing them to search. Having the option to follow other users whose opinions they value also makes it easier for consumers to find new places and keep up with the newest trends in food and drink. Encourages users to explore new things and supports them in finding new locations that match their likes and tastes. The option to post images of the food and drinks they sampled increases the review's visual placement and lets other readers know what to plan. System architecture improves loading and posting times for food images. In a food and beverage review platform, you can also leave comments about the staff's knowledge, attentiveness, speed of service, and overall impression of the service. Overall, the social networking site for food and drink reviews created with React and Spring Boot is an expanded platform for food lovers to explore, share, and discuss different foods.

1. Function requirements of the REST API:

*[Main Functions]*

***User Management Service***

The rest API implemented for this web application should get the information about users from the database, it should have the facility to insert **[POST]** users into the database, it should have the capability of **[GET]** displaying user details as well as update**[PATCH]** and delete **[DELETE]** information on a certain user.

***Post Management Service***

A post consists of a description, images of the client’s experience, location details, hashtags and a rating for the experience. A client can view their own posts **[GET]** through their account as well as other user’s publications **[GET/Allusers].** A user can modify their own food reviews **[PUT/PATCH]**.A user could also delete their posts **[DELETE]**

***Follow/Unfollows, Like/Dislike Management (Interaction Management Service)***

In order to increase the interaction users should be able to manage follows and unfollows .The details of a followers of a particular user and the following for that user should be accessible**[GET]** through the rest api.

It should also have the capability to decrease **[PATCH]** once a user unfollows another user and increment once a user follows another user. Similarly like and dislikes are managed

***Comment Management (Interaction Management Service)***

A user should be able to add a comment under a certain **[POST].**A comment can be added and well as modified according to the user needs **[PUT].** A user can also view comments posted by other users **[GET/Allcomments]**

*[Additional Functions]*

***Search Function:***

A user could be able to search posts using locations mentioned under each post. A user should also have the capability to search another user via username **[GET]**

Functional Requirements for the client Web application:

***User Management Service:*** The web application should provide user details in a convenient and easy to understand manner. The user profile should be

***Interaction Management:*** The services contributing towards increasing client interactions such as likes/dislikes, comments, follows/ unfollows should presented in a user friendly manner for an example when a user likes a certain post it is applicable to show a thumbs up/ number of thumbs up’s .The number of likes should be incremented and displayed

Likewise the number of followers a particular user has should also be included along with the number of users they are following in a comprehensible manner

***Authentication:*** Once the user is verified there should be a success message displayed along with the facility for user to navigate to their profile

***Notification Services:*** The system should indicate the notifications to the user via a bell icon within their profile

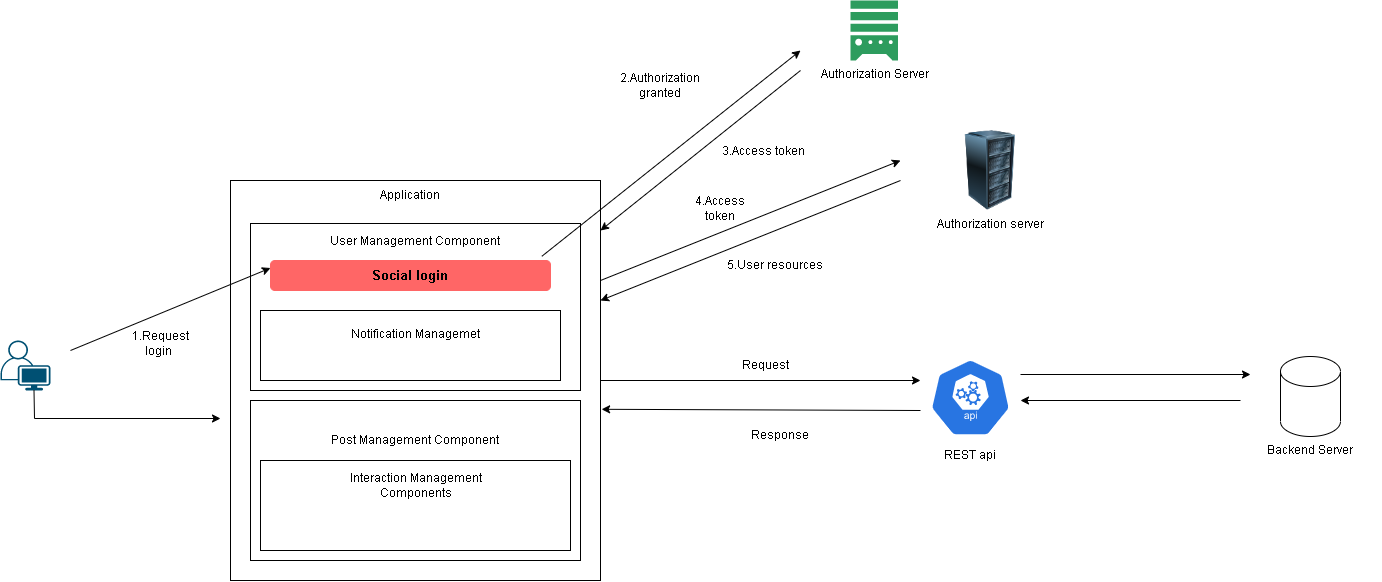
1. Nonfunctional Requirements of the REST API:

* Performance: The REST API should efficiently access details from the database of users, posts and interactions.
* Speed: The details fetched by the REST API should be accessed fast so that it reduces response time
* Security: The client should be authenticated properly so that user’s do not face personal details being exposed
* Maintainability: The rest api should be easily maintainable and capable of fetching, inserting and modifying a large number of records
* Scalability : The REST API should be able to adapt to new technologies without affecting the client goals of the website
* Compatibility :Since the web application also consists of a mobile application information to this application should be able to share details between the two system

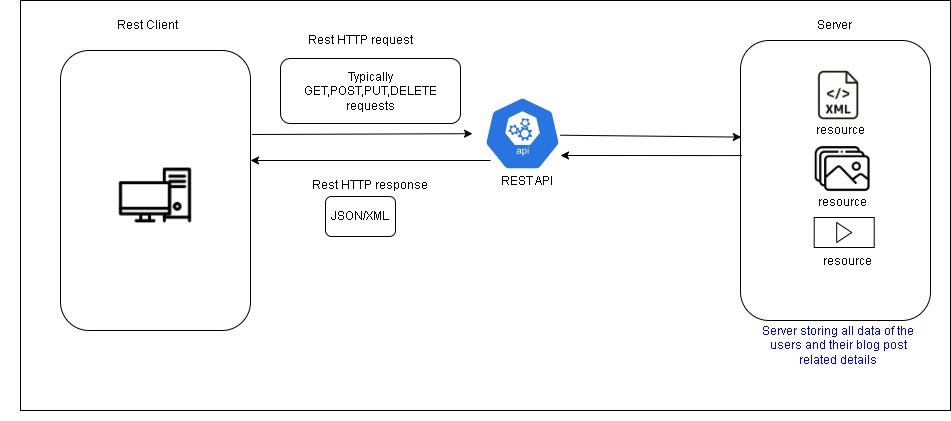
Nonfunctional Requirements for the client Web application:

* Usability: The web application should be efficient for user to achieve their intentions that they expect from the application
* Performance: The web application should efficiently navigate between components reducing response time
* Reliability: The correct client components which are requested should be accessible for the client. Furthermore, the flow of the component should meet the client requirements

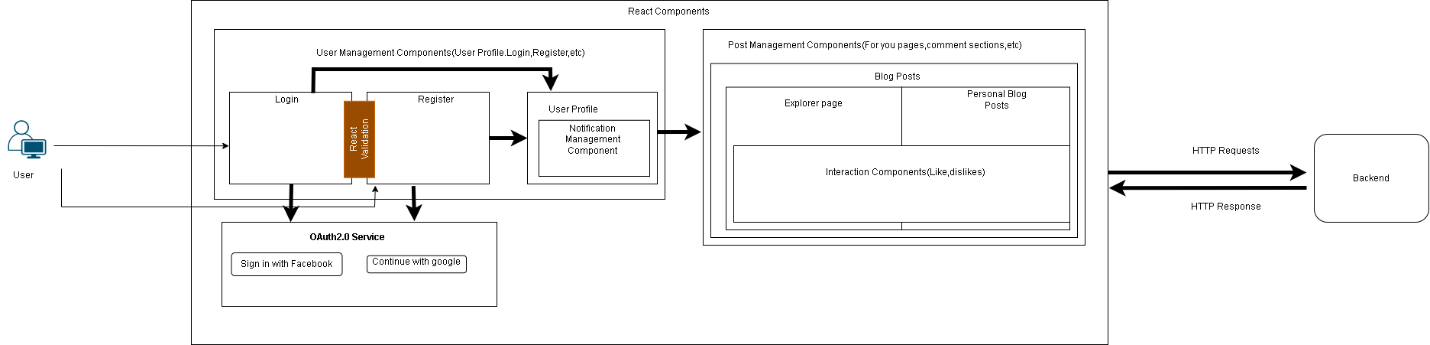
1. Overall Architecture Diagram



1. Architecture diagram for REST API



1. Detailed diagram of the front end architecture



1. Resources

* [AWS Amazon.com-What is a REST API](mailto:https://aws.amazon.com/what-is/restful-api/)
* [React developer tools](mailto:https://react.dev/blog/2023/03/16/introducing-react-dev)
* [The Java API for Restful web services](mailto:https://react.dev/blog/2023/03/16/introducing-react-dev)