Jaldi.io Project

CP476: Internet Computing

Winter 2021

Project Proposal

Kaish Panjwani – 183168050

Ravish Virani – 173084290

Sahil Lalani – 183145550

Introduction

As we know the life of most people in today's world revolves around technology. One of the biggest contributors to this technological involvement in our lives is social media. Big parts of the population of many countries use social media at least 1 hour a day. We have become so socially connected online that we have started to lose personal interaction times with closely related people.

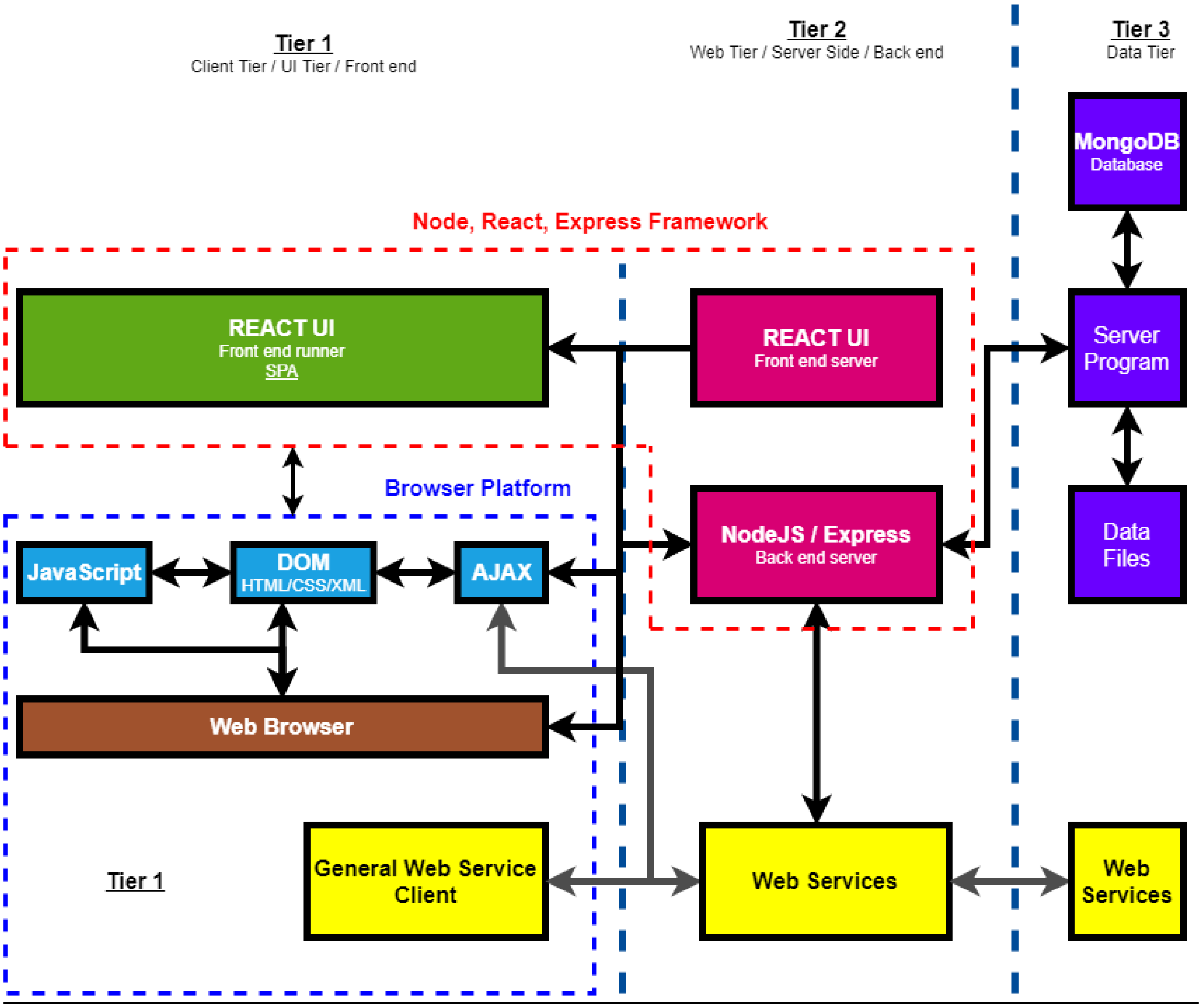
Therefore, a problem of social interaction arises due to less quality time with people around you. We as a society are connected to people on the other side of the globe but still sometimes do not know what is happening on the other side of the room. For example, the families are growing apart. Another issue is that during the ongoing Pandemic of COVID-19, rather than spending more time with people near them and improving their relationship, people are spending more and more time on social media, movies, and gaming. Thus, this is the most important problem identified making families the main target audience.

This project aims to develop the web feature which encourages this form of family engagement. This app is designed to be a highly elaborate Family Connection Web-Application which would encourage family interactions. The application would aim to boost interaction by including elements like image sharing, chat rooms and Recipe zone (our most ambitious attempt). Having a different approach from a regular messenger tool, the project hopes to categorize zones so that each family member can contribute to share or receive information from the platform.

The web app will allow the users to create accounts and categorize them in a database based on what family they are part of, is the user an admin or normal user, etc. Admin users are the users who have created a family and have control over who can be the part of the family. All the users including admins, will be able to participate in a closed environment chat room and share media in an organized manner. We plan to include a recipe zone where the users can share recipes with the family, and which will also allow the users to calculate the cost estimates of the recipe.

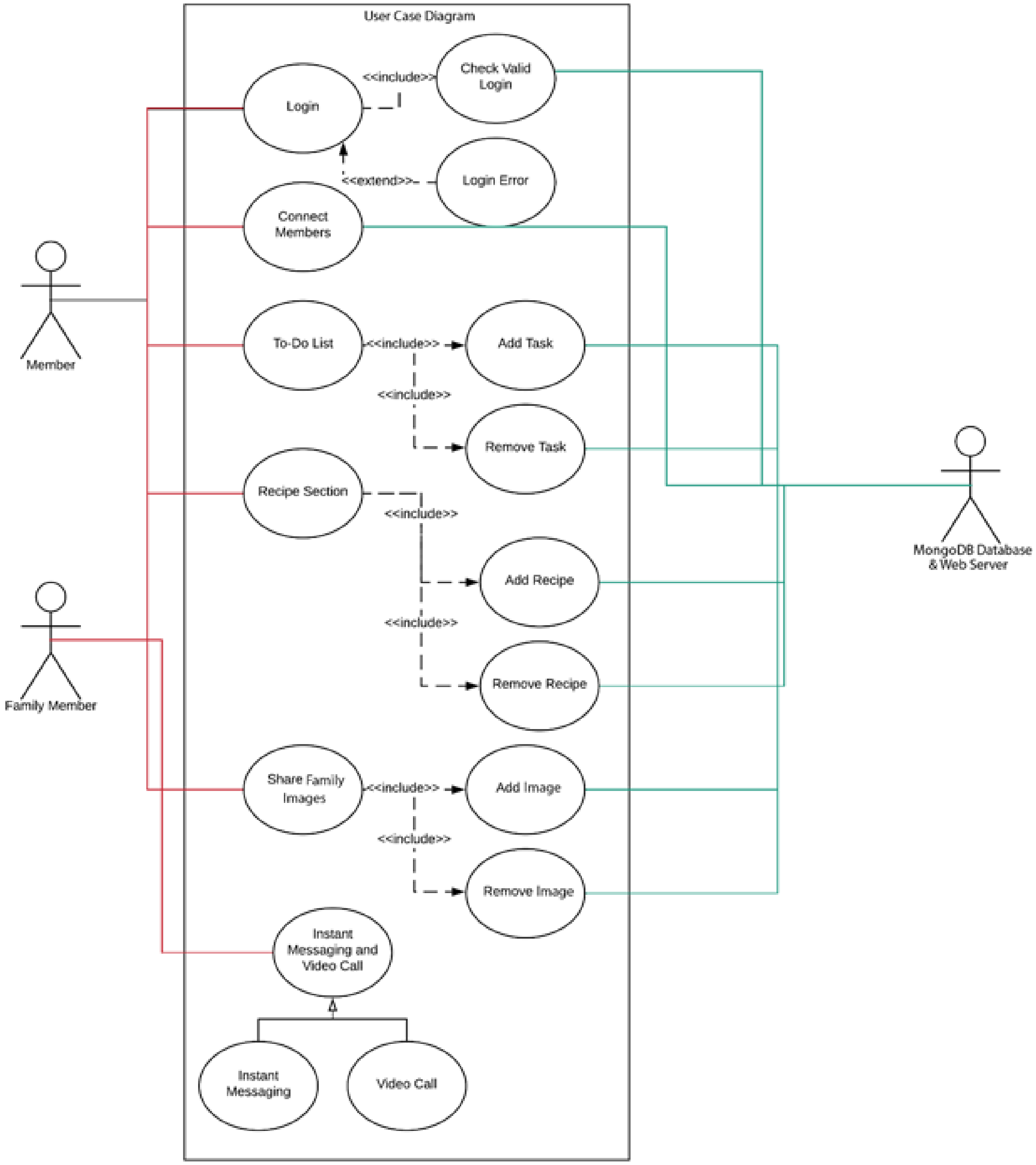
System Design

1. Web Application Architecture



The web application will have the following architecture. It is divided into three tiers. The first tier will handle the front end and will be rendered using ReactJS with the help of AJAX, DOM and JavaScript. The second tier would be made up of the front-end server and the back end server. This section would be handled using NodeJS. The third tier would be the data tier which would take care of the data files and database. We will be using MongoDB as the web application database. And the server program will connect this tier to the second tier. At the bottom is the Web Services platform which would use the Restful API to provide a better interaction with the client.

2. Use Case Diagram



The diagram above displays how the users/clients will interact with the web application. Here the three interacting parties would be a Member/Client, a Family Member (part of a family connection) and the MongoDB database and Web Server will store all the information and present it to the user and provide functionality to the web application.

3. Tools

The list below displays the list languages, libraries, frameworks and tools that will be used for this project.

The project will be using the following Languages:

- JavaScript (web application functionality)
- JSON/XML (for storage of object data)
- HTML/CSS (basic structure)
- PHP (server side programming)

The project will be using the following Libraries and Frameworks:

- DOM (modifying page structure)
- React JS (front end / UI)
- React Redux
- React Modal
- React Big-Calendar
- Restful API (better interaction with client)

This project will be using the following Tools:

- Notepad++
- MongoDB Atlas (storing data)
- Photoshop
- NodeJS (back end functionality)
- Visual Studio Code (project development)
- Web Browser (web application running platform)

Problem Solving & Algorithms

One of the main problems constructing a web application for public use is privacy of every individual using the application. To counter this problem, we would be developing a Login page which would enable users to Login or sign up for the application. To keep it secure the login would include inserting a password which would enable only the owner of the account to access the application. This would be done by implementing the SSL library the user forms data to ensure security and robustness.

The application includes a recipe sharing room, which allows a user to upload or view any recipes posted on the application. The recipes could include pictures of the recipe, description and procedure and items used in the recipe along with their cost. The individual cost would then be added along to give the approximate amount to make a particular recipe determined by the amount of each item used.

The web application will also have an event Calendar where clients can upload their events. When a client uploads an event, they have the chance to add details to the event. The calendar will use the date picker to choose what days would the event use.

These all will be using SHA1 Encryption Algorithm to store passwords for security, Routing Algorithm to route through different pages including redirecting users to login page if they haven’t logged in, and Storing Algorithm to fetch and store items from and to Database.

Milestones & Schedule

Task ID	Description	Due Date	Lead
1.	Project research & team up	10th March	Ravish Virani
2.	Project proposal	22st March	Sahil Lalani
3.	Login Page set up	25th March	Kaish Panjwani
4.	Calendar	28th March	Ravish Virani
5.	Client-Side Components UI	31th March	Sahil Lalani
6.	Server-Side Component	2nd April	Kaish Panjwani
7.	Problem Solving Algorithms	6th April	Ravish Virani
8.	Project demonstration	8th April	Sahil Lalani
9.	Project submission	10th April	Kaish Panjwani

References

- https://reactjs.org/docs/getting-started.html
- https://www.php.net/docs.php
- https://developer.mozilla.org/en-US/docs/Web/JavaScript
- https://nodejs.org/en/docs/
- https://docs.mongodb.com/
- https://api.jquery.com/category/ajax/
- https://developer.mozilla.org/en-US/docs/Web/API/Document_Object_Model
- https://developers.google.com/web/fundamentals/security/encrypt-in-transit/enable-https
- https://www.openssl.org/docs/

Appendices

Task ID	Description	Done By
P1.1	Creativity & originality	Everyone
P1.2	Core problems and algorithms	Ravish and Sahil
P1.3	Requirement specifications	Kaish and Sahil
P1.4	Architecture design	Kaish
P1.5	Milestones and schedule	Sahil
P1.6	Writing of the proposal	Everyone

Task ID	Description	Implementation
P2.1	Client-side component & UI	Using ReactJS
P2.2	Server-side CGI components	Using NodeJS and Express
P2.3	Database tier design, data, usage	Using MongoDB
P2.4	New features and tools	Recipe, Calendar
P2.5	Problem solving algorithms	Explained in above section
P2.6	Efficiency and robustness	Implementation of SHA1 for security