IM3080 Design and Innovation Project (AY2021/22 Semester 1)
Individual Report

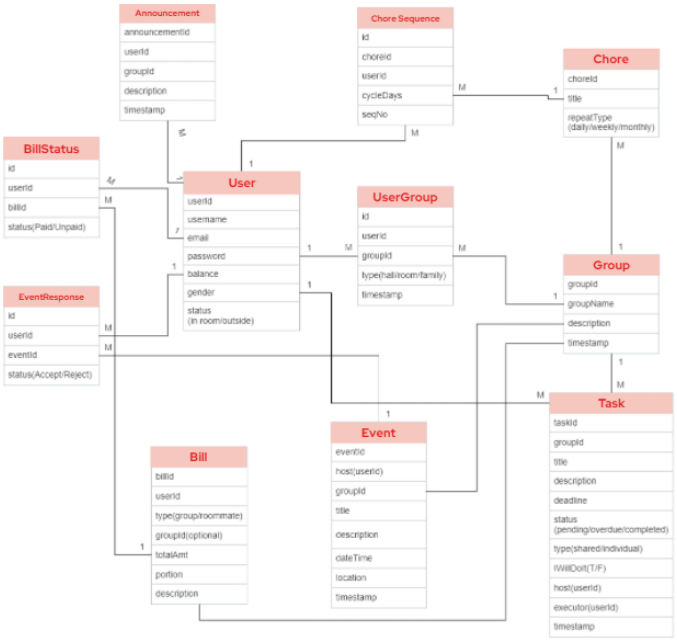
Name: **Zhu Ziyi**

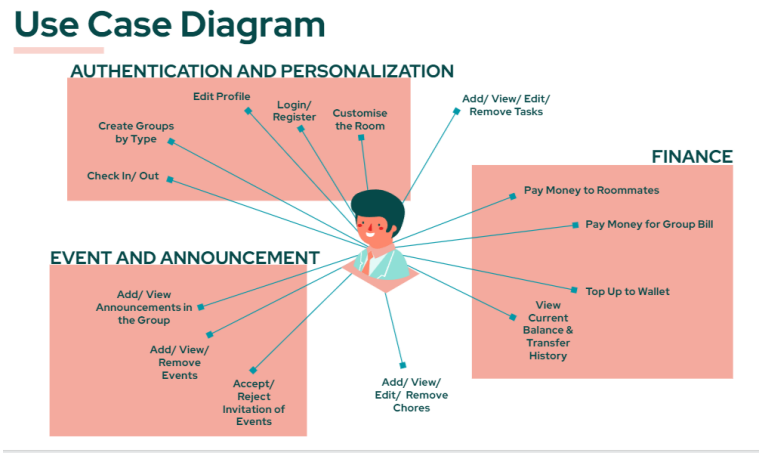
Group No: **6**

Project Title: **COHAB**

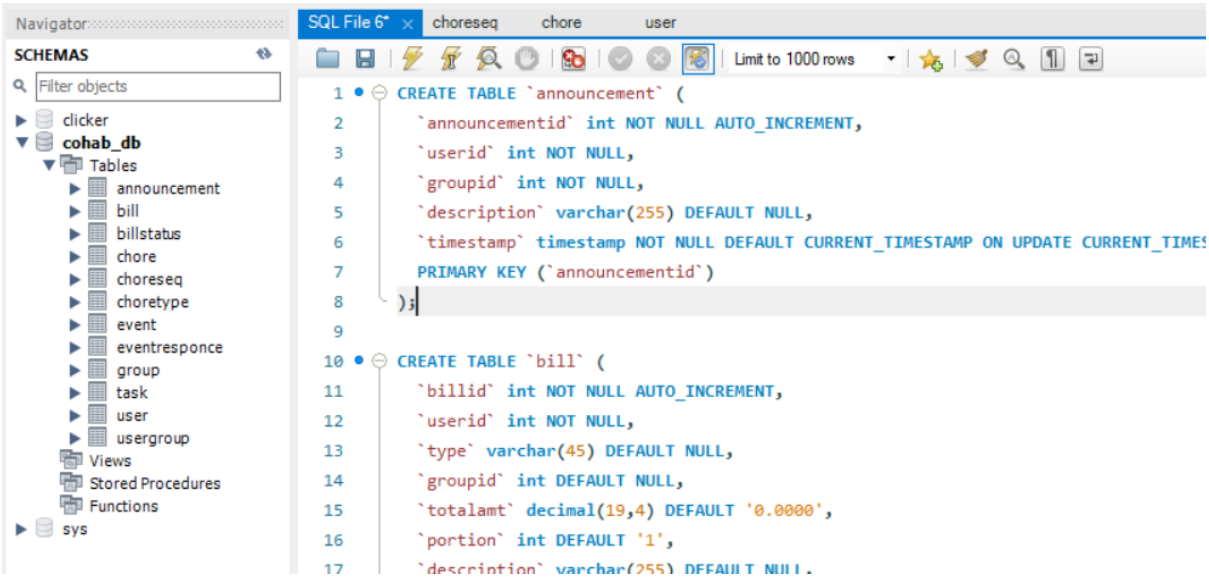
Contributions to the Project (1-2 page)

1. Use Case Diagram and Entity Diagram (Solo)

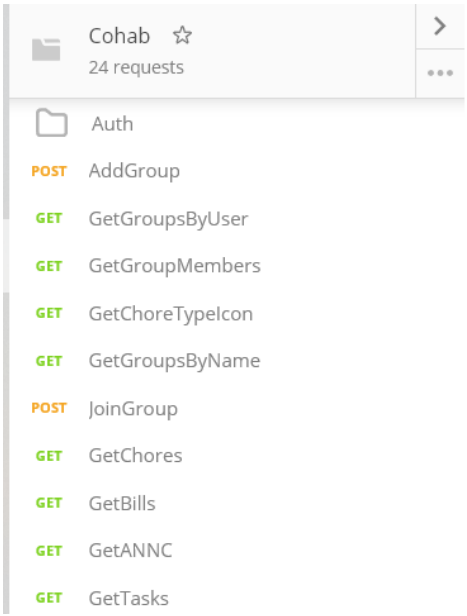


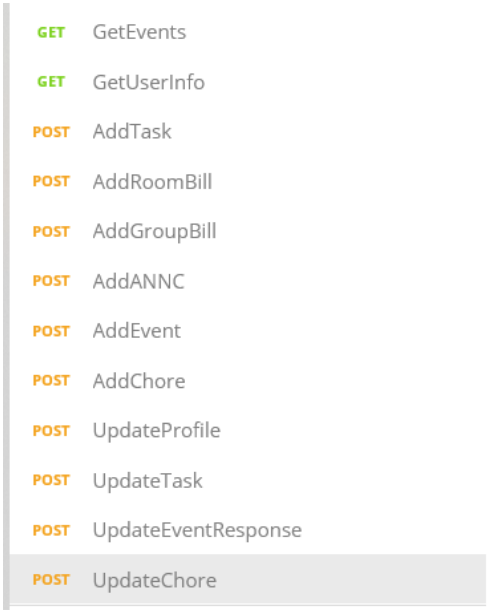


2. Database Design and Establishment (Solo)



3. Backend (Solo)
   Include: 24 APIs
   (Use Tomcat Server and MySQL)
   (Tools: MySQL Workbench, Postman, ngrok proxy)





4. Integration with react native app (front end)
   (Front end call API to do POST/GET… request for CRUD)

1>APIs example

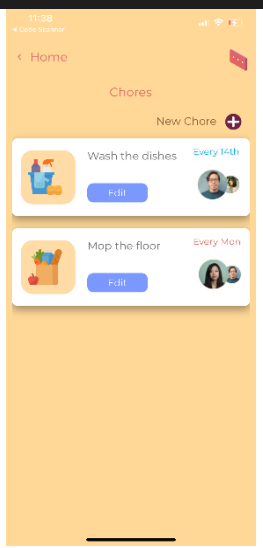
```
GROUP:
        POST:http://<ID>/cohab/addGroup
        dataPost:{
           "userId":12,
           "groupName":"Hall",
           "description":"This is for all hall 3 students",
        }
        POST:http://<ID>/cohab/joinGroup
        dataPost:
        {
           "userId":15,
           "groupId":8
        }
        GET:http://<ID>/cohab/getGroupsByUser?userId=15
        response:
        {
                "groups": [
                        {
                                "description": "This is for all NTU EEE students",
                                "id": 1,
                                "groupname": "EEE"
                        },
                        {
                                "description": "This is for all hall 3 students",
                                "id": 8,
                                "groupname": "Hall"
                        }
                ]
        }
CHORES:
```

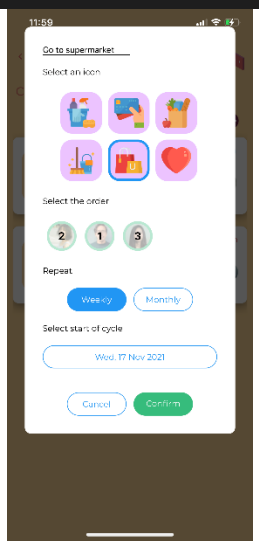
```
function getGroups(userId) {
  const getGroupsURL = ConstantHelper.CONNECTION + `getGroupsByUser?userId=${userId}`;

  const init = {
    method: "GET",
    headers: {
      Accept: "application/json",
      "Content-Type": "application/json",
    },
  };

  function updateGroups(json) {···
  }

  (async () => {
    try {
      const response = await fetch(getGroupsURL, init);
      const json = await response.json();
      updateGroups(json);
    } catch (error) {
      console.log(error);
    }
  })();
}
```





5. React native UI debug and adjustment
   - Room Screen
   - Chore Screen
   - Login Screen

6. Week8/Final Presentation (Demo)
   Part of final report

Reflection on Learning Outcome Attainment

Reflect on your experience during your project and the achievements you have relating to at least two of the points below:

(a) Engineering knowledge
(b) Problem Analysis
(c) Investigation
(d) Design/development of Solutions
(e) Modern Tool Usage
(f) The Engineer and Society
(g) Environment and Sustainability
(h) Ethics
(i) Individual and Teamwork
(j) Communication
(k) Project Management and Finance
(l) Lifelong Learning

Point 1: (a) Engineering knowledge (e) Modern Tool Usage (I) Lifelong Learning

This project enables all the team members to be familiar with **figma**, **React Native** and **GitHub**. So, we're all learning as we go along. At beginning, we design each screen UI by using **figma.** And since I was working on backend development individually. Whenever I write an **API** in **Java Servlet** on **Tomcat Server**, I implement the API and test functionality in **Postman** which is an API platform for building and using APIs. Postman simplifies each step of the API lifecycle and streamlines collaboration so you can create better APIs—faster, and it allows me to develop and test APIs without frontend part, so it provides flexible timeline for backend and frontend. We use **MySQL** to query the data from **database** and I choose **Workbench MySQL** which is a visual database design tool to mange our app database. And when we do the integration, I use **Ngrok proxy** to make us easy to access the server remotely. When a client makes a request, the application queries the database and produces a response in JSON format. This JSON can be parsed with most of the programming languages. Therefore, on frontend we use HTTP methods for **RESTful** Services (GET/POST/PUT/DELETE) to call the APIs which require **JSON** as the post data and each request will return a JSON object as response.

Point 2: (b) Problem Analysis

Handle exceptions by using try catch or console.log () to print errors or variable value for checking the logic mistakes or misplace of the elements.

Point 3: (c) Investigation

Research for a proxy, GitHub and react native (useEffect, useState, model, networking, app lifecycle…)

Point 4: (d) Design/development of Solutions

By using use case diagram, database design and entity diagram.

Point 5: (i) Individual and Teamwork (j) Communication

This project made me realized more that being part of the development team requires both the ability to work independently and communicate effectively. Once I finished the backend code, I started working with teammate for each screen integration and help teammate for UI design and debug. For the convenience of my teammates, I used proxy to make the server accessible to them. When I was doing integration, I wasn't clear about the logic of some of the front-end code. I am not quite familiar with React.js and I did not have much practical hands-on experience with the technology. However, we are lucky to have members who are more experienced in developing React native app. There were many setbacks during the process of coding certain features or functions. I was able to overcome these challenges by effectively using online resources like Stack Overflow, React Docs and other documentations to solve the issues. I understand that other members will be occupied with their own deadlines and school curriculum, and I put in extra effort to deliver desirable outcomes by deadlines and try to resolve all issues first with my own technical capabilities. Even though most of the problems encountered could be answered with online resources, there are also many instances that the problem actually lies in the emulator, IDE, or databases. In addition, we could have overlooked some minor issues that last for prolonged period, and I think it is helpful to reach out for help in order to avoid delay of delivery. Most of the time we actually need to help each other because some bugs actually need to be fixed in each other’s part.