



School of Information Technologies
Faculty of Engineering & IT

ASSIGNMENT/PROJECT COVERSHEET - GROUP ASSESSMENT

Unit of Study: COMP5703 IT Capstone Project

Assignment name: Project Progress Report

Tutorial time: Wednesday 16.00-18.00 Tutor name: Hamid Samani

DECLARATION

We the undersigned declare that we have read and understood the [University of Sydney Academic Dishonesty and Plagiarism in Coursework Policy](#), and except where specifically acknowledged, the work contained in this assignment/project is our own work, and has not been copied from other sources or been previously submitted for award or assessment.

We understand that failure to comply with the *Academic Dishonesty and Plagiarism in Coursework Policy* can lead to severe penalties as outlined under Chapter 8 of the *University of Sydney By-Law 1999* (as amended). These penalties may be imposed in cases where any significant portion of my submitted work has been copied without proper acknowledgement from other sources, including published works, the internet, existing programs, the work of other students, or work previously submitted for other awards or assessments.

We realise that we may be asked to identify those portions of the work contributed by each of us and required to demonstrate our individual knowledge of the relevant material by answering oral questions or by undertaking supplementary work, either written or in the laboratory, in order to arrive at the final assessment mark.

| Project team | | | | |
|------------------|------------|--------------|----------------|-----------|
| Student name | Student ID | Participated | Agree to share | Signature |
| 1. Yuming Jiang | 460444981 | Yes✓ / No | Yes/No | |
| 2. Xinyi Liu | 450456633 | Yes✓ / No | Yes / No | |
| 3. Zhengyang Liu | 460161413 | Yes✓ / No | Yes / No | |
| 4. | | Yes / No | Yes / No | |
| 5. | | Yes / No | Yes / No | |
| 6. | | Yes / No | Yes / No | |
| 7. | | Yes / No | Yes / No | |
| 8. | | Yes / No | Yes / No | |
| 9. | | Yes / No | Yes / No | |
| 10. | | Yes / No | Yes / No | |

SIT Building, J12
The University of Sydney
NSW 2006 Australia

T +61 2 9351 3423
F +61 2 9351 3838
E sit.info@sydney.edu.au
sydney.edu.au/it

ABN 15 211 513 464
CRICOS 00026A

A Reference Model of Applying IoT (Internet of Things) and Blockchain to Enterprise System

Yuming Jiang, Xinyi Liu, Zhengyang Liu

Progress / Achievements

1. API Server

According to the user stories explained in our proposal, two main functions, user authentication API and blockchain system adapter, should be implemented. In the current stage, we finished the first version of our API server. In this version, the basic structure of the user authentication API has been established. Users can use the web application or the mobile application to call the user authentication API to register as a user, log in with the username and password and log out. Our API server can also remember the user authentication status so that users do not need to log in again to a different page. We use a node package called “Passport” to build the whole user authentication API which means if our users want to log in with their Google or Facebook account, it is easy to add that function and does not need to modify a lot of code. In the current stage, we have not finished the blockchain system adapter. As blockchain is a new topic, a lot of blockchain systems are also work-in-progress. In this situation, the usage documents of these blockchain systems are also not detailed enough. We need more time to assess and test these blockchain systems to choose the best one to implement our system.

2. Mobile Application

The first achievement on the mobile app is to determine the developing environment of the app. Before developing a mobile app, we need to learn the platform, based on which we will build our app. Since two members of our group are Windows users, iOS seemed unrealistic for our requirements. We decided to develop the Android version mobile app. Meanwhile, according to our knowledge to React-Native, it is a highly duplicable language that we can replicate the most code to use in the iOS environment, although some adjustments are needed. Another achievement is the essential functions of the mobile app. The login function has been realized with the assistance of firebase, which will be updated once the BigChainDB API is available. As shown in figure 1, the initial page of the app is the login page, which consists three main parts: the email field, the password field, and a “Login” button which is used to log a user into the system. When the user inputs the identical email and password value pair which have been stored in the database, the app will authorize the access to the user. The user will get into the main page. If the login information is not approved, an error message in red color will be returned and showed on the screen, between the password field and login button, as shown in figure 2. When the login is approved, the main page shown in figure 3 will pop up. Currently, the main page is almost blank, when it can connect the API and fetch data from the database, it will return the transaction data based on the user.

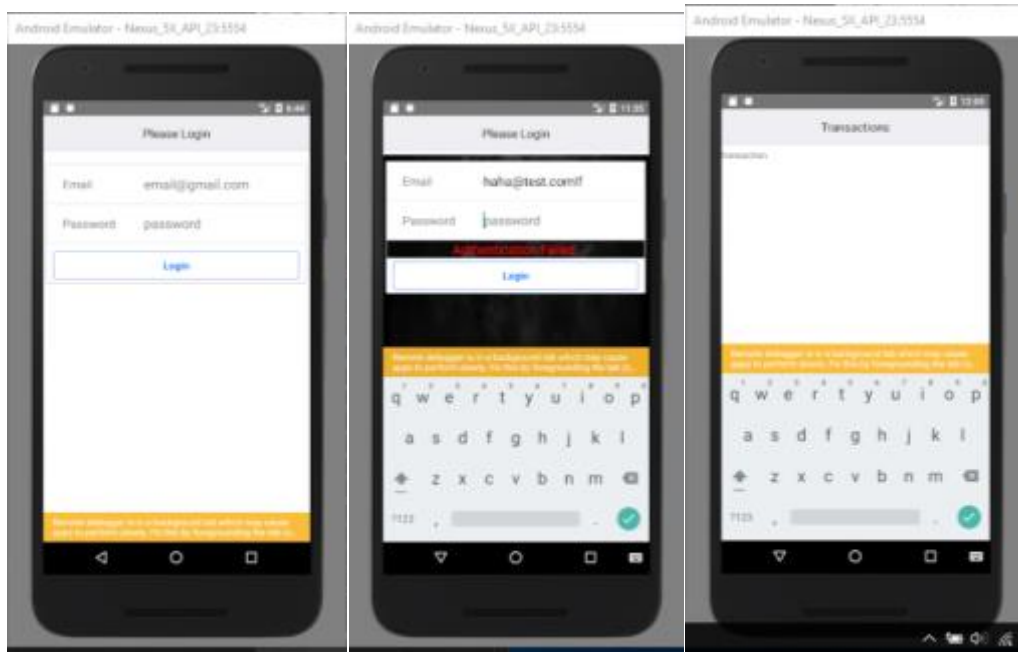


Figure 1

Figure2

Figure3

3. Web Application

According to the requirements of our system, there are four essential functions should be built: the login and register function, logistic information searching function and adding new tracking information function.

We plan to build five web pages that include the four essential functions. the main pages would show the company information and notification board about the system and user also can login or register the account on the main page. The tracking information list page that can list all the information that related to the user account. If the user clicks the tracking information of the list that will jump to a new page called tracking information checking page and the tracking information details would show on this page. Besides, the user also can add new tracking information on the website in a new page called tracking information adding page.

In the current stage, we achieve the first version of this web portal. The user can add the tracking information to the database and can check all the tracking information on the tracking information list page. In addition, the user can also check the details of every logistic information if the user clicks the specific tracking record in the list.

In conclusion, we have already completed the 1/4 essential functions of the web application. In the next stage, we will achieve all the essential functions of the web application and optimize the UI of every web pages.

Experienced problems

1. API Server

1.1. Environment Variable Configuration

API Server needs some environment variables which should be defined before the running of the server such as the port number and the URL of MongoDB. These environment variables cannot be hard-coded into API Server because that they may change in different environments. To solve this problem, we store configuration in one file called “.env” which is separated from code. We use a node package called “dotenv” to load environment variables from “.env” file into “process.env” variable at the runtime of API Server.

1.2. User Authentication

In the current stage, our API Server can only authenticate users whose username and password are stored in the local MongoDB database. However, in the future, we may need to provide other authentication methods for our users such as authentication with Google account or Facebook account. It is hard to write the code again and again to implement the user authentication function. We use a powerful and famous node package called “Passport” to handle all the functions related to authentication which reduces the future work and tidies the code.

1.3. Blockchain System Adapter

The blockchain is a new topic. Most of the blockchain system projects are also work-in-progress which means there may exist some unknown bugs and the documentation about the usage of the system is not detailed enough. In this situation, it is hard to understand the whole blockchain system and write an adapter to connect our API Server to the blockchain system. In the current stage, we are still assessing all the blockchain systems to choose the most useful one. We are trying our best to solve this problem as soon as possible.

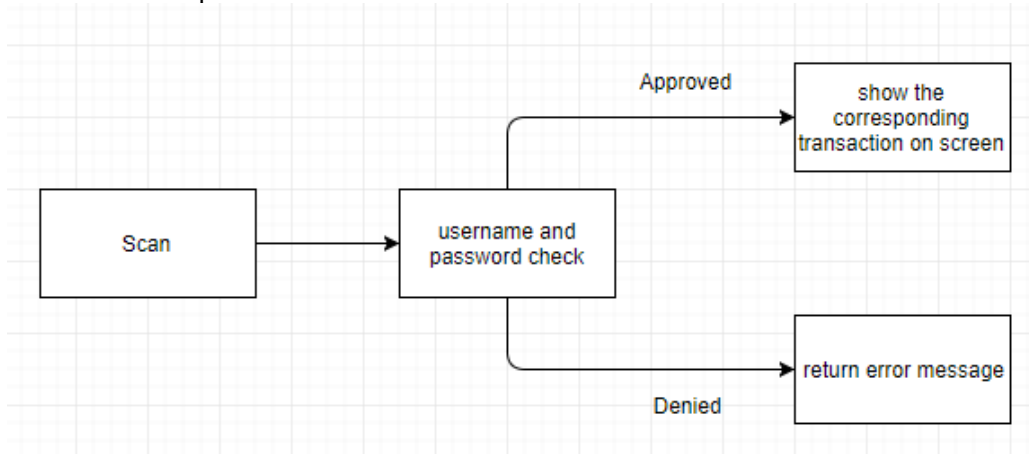
1.4. Deploy API Server on Linux

Before this project, we only focus on writing the code of API Server. We do not have any experience on using Linux. To solve this problem, we read a lot of tutorials online to learn how to install Node.js and MongoDB on Linux and how to deploy our API Server. In the current stage, the basic API Server has been successfully deployed and can be accessed by the web application and the mobile application.

2. Mobile App

2.1. The Scan Function Development

The scan function is still in the air. It is a little annoying to implement. As planned, the scan function should process as showed as follow.



When the user scans the QR code through the camera, the app will carry the username and password to the system database to check their validity. If the username and password are valid, the app will jump to the main page and show the transaction information related to this user. Otherwise, an error message will be returned to indicate the user does not have the authority to access the system.

2.2. Learning and Using Android Studio

Android Studio is not stable. Sometimes, when I tried to reload the page after some updates, it would stop working. Thus, I have to restart the Android Studio to continue.

2.3. Learning React-Native

Understanding some principle of react-native is difficult since it is different from another

language, including the framework, logic, and the method to call functions, such as the `mapStateToProps`, `connect`, and the reducer. They are not so easy to use. It took me a lot of time to digest.

3. Web application

3.1. Learning JavaScript

We had met some technical problems when we developed the web application. In the process of learning JavaScript, there are some concepts are challenging to understand for a beginner, such as the selector for the ID and date, the concept of the relative position and the usage of the `unload` method, etc.

3.2. Learning React.js

During the studying process of React.js, some concepts are not easy to understand, as what the props and state use for, how the callback function is used and how the payload works. Besides, we also met the incompatibility problem that the errors would be reported when the different versions of react.js, web pack, etc. work together, which took me some time to fix.

3.3. Learning and Using New software

We also learn to use some software like postman and so on, which also spent some time to learn how to use them.

Updated Timeline

| Week No | Task | Students in charge | status |
|---------|---|--|----------|
| Week 1 | 1. Determined objectives and goals of the project 2. Find out what should we learn 3. Learn something about react | Yuming Jiang Xinyi Liu Zhengyang Liu | complete |
| Week 2 | 1. Learning redux concepts and models | Yuming Jiang Xinyi Liu Zhengyang Liu | complete |
| | 2. Learn React-redux framework | | |
| | 3. Background research on the existing system | Yuming Jiang | complete |
| | 4. Literature review on related fields | Xinyi Liu Zhengyang Liu | complete |
| Week 3 | 1. Function designing | Yuming Jiang; Xinyi Liu; Zhengyang Liu | complete |
| | 1.1 Web app function design | Xinyi Liu | complete |
| | 1.2 Mobile app function design | Zhengyang Liu | complete |
| | 1.3 API_server design and construction | Yuming Jiang | complete |
| | 2. Drafted proposal presentation | Yuming Jiang; Xinyi Liu; Zhengyang Liu | complete |
| Week 4 | 1. Literature review for proposal and proposal draft 2. Component design and confirmation | Yuming Jiang Xinyi Liu Zhengyang Liu | complete |
| Week 5 | 1. Complete and submit proposal | Yuming Jiang; Xinyi Liu; Zhengyang Liu | complete |

| | | | |
|--------|--|--|------------|
| | 2. react-native study | | |
| Week 6 | 1. Blockchain-system review 2. Basic structure for API server | Yuming Jiang | complete |
| | 3. Configure the environment for react-native and build the basic framework for mobile app | Zhengyang Liu | complete |
| | 4. Build the basic structure of the web app | Xinyi Liu | complete |
| Week 7 | 1. Executable basic web app | Xinyi Liu | complete |
| | 2. Executable basic mobile app | Zhengyang Liu; Yuming Jiang | complete |
| Week 8 | 1. First version of web app | Xinyi Liu | Under work |
| | 2. First version of mobile app | Zhengyang Liu; Yuming Jiang | Under work |
| | 3. Well-function api_server and bigChainDB | Yuming Jiang | Under work |
| | 4. Test and integrate the whole basic system | Yuming Jiang; Xinyi Liu; Zhengyang Liu | Under work |

Evidence of collaboration

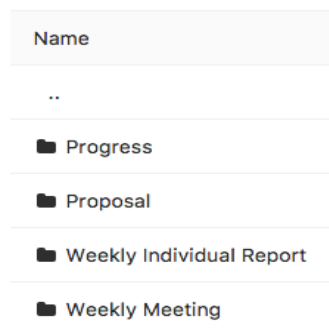


Figure 4. Repository



Figure 5. Slack

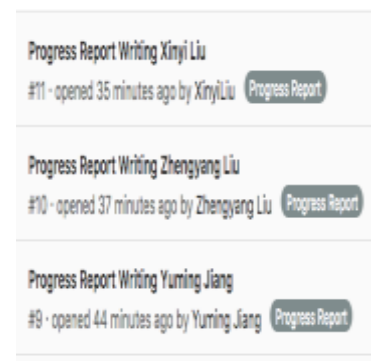


Figure 6. Issues

As shown in Figure 4, our weekly study note is uploaded to GitLab in folder Weekly Meeting, which also sent to tutor as weekly project status report via email. Besides, we also upload all the references and related document about the proposal and progress report on the Gitlab in corresponding folders.

As shown in Figure 5, we communicate via Slack to arrange our weekly meeting and communicated with our client and tutor.

As shown in Figure 6, Yuming Jiang wrote the contents related to API Server in the section Progress / Achievements and Experienced problem., which is recorded in issue #9. Zhengyang Liu was responsible for the mobile app achievements, experienced problems on mobile app development, and the updated timeline, which is recorded in issue #10. Xinyi Liu wrote the web application current achievements and the experienced problems that we met during the development of web application, which is recorded in issue#11.