COMP9900 Project Proposal
Group AlphaGo

04 Oct, 2020

[Scrum Master]			
Wanchao Huang	Front-end	z5192415	z5192415@ad.unsw.edu.au
[Group Member]			
Raymond Lu	Front-end	z5277884	raymondluk1995@gmail.com
Tao Xue	Back-end	z5219280	z5219280@ad.unsw.edu.au
Xiaohan Zhu	Back-end	z5187021	z5187021@ad.unsw.edu.au
Ziqi Liang	Front-end	z5202297	z5202297@ad.unsw.edu.au

Contents

1 Background

1.1 Problem statement

With the development of online transaction systems, more and more transactions can be carried out on the Internet. Due to the portability of the network platform, this not only brings convenience to users, but also provides sellers with more potential customers, thereby improving commodity exposure.

This project is dedicated to developing a real estate sales page in the way of auction. The user group of the website can be roughly divided into potential buyers, registered auction bidders, and real estate sellers.

For potential users, what they need is a convenient bidding platform that can correctly match their needs, provide sufficient and detailed housing information, and promptly give feedback after a transaction.

To save user's searching time and boost up sales, a property recommendation system is going to be introduced. Moreover, for better property administration experience, property auction history display will be included in the program development.

For sellers, this website is designed to provide a reliable real estate sales platform. First of all, we will ensure the authenticity of users. Every registered user will have a unique token in the background. Secondly, we provide a real estate management back-end system so that sellers can update real estate information at any time and manage auctions. Thirdly, we will also be responsible for notifying all associated users of the auction results after the auction is over.

As a system involving transactions, we want to provide not only basic functionalities, but also a practical, reliable, and user-interactive system. We will carry out follow-up detailed development for the above aspects, improve the system during program development, and keep refining details during the project.

1.2 Existing systems analysis

For reference learning, we looked at the existing online housing transaction sites *Openn Negotiation* (https://www.openn.com.au/) and *AnywhereActions* (https://anywhereauctions.com.au/) and summarized the pros and cons of the two sites.

For the user guide module, both two sites provide introductions and Q&A to each identity (buyers and sellers), which give users instructions to use this website.

In the search module, both of these sites could find houses from satellite maps and can filter houses by min land size, as well as the number of bedrooms or bathrooms. *Openn* can filter results by type and features, while *AnywhereActions* can filter by max or min price. *AnywhereActions* can also sort actions by price or time. For each searched result tag on *Openn*, the photo of the house's appearance, location, sale status and current highest price of each house are clearly given. The number of bedrooms, bathrooms, and garages, and the number of current bidders and observers can also found too. Besides, users can collect houses they are interested in at any time. On *AnywhereActions*, the differences are the auction end time is provided, and users can see the video of the house.

On the page of the property introduction, both of these websites have multiple pictures of the house. Users can find the size and the detailed description of this house. The contact information of the Agent is easy to be found. Both of them have the text description, and a mini-map showing the house location. The contrast between these two websites is the price of a property is only available on *Openn*, while, *AnywhereActions* has the auction end time displayed.

Both of these websites sell the property through an agent. We aim to build an auction site that does not require an agent. All transactions can be carried out by users themselves.

2 User Stories

All user stories of Alphago online property sales are given below. The stories are designed based on the project objectives and analysis of existing systems.

2.1 User Authentication

Users do not need to login to view the default index page. They can also see what properties are under auction and search properties based on location or filters on requirements. However, a user without logging in cannot bid or post a property to sell.

1. User Registration Page.

   If a customer user has not registered yet, a registration process needs to be gone through first. A new customer user needs to mark down username, his/her first name, last name, email address, password, avatar and phone number. Also, during the registration, a confirmation of password consistency will be required. After the front end passes the registration information to the back end, the system will check that there will be no duplicate username, email addresses or phone numbers existing in the database.

2. Login Page

   Registered users can login in at this page by inputting the username and password. After the system confirms that the inputted information is correct, the user will be directed to the page that he/she was browsing right before going to the login page.

   If a user forgets his/her password, he/she can apply to reset the password simply by clicking the 'forget password' button, this will trigger a validation email sent to this user, after verifying the identification, they can proceed to reset the password.

3. Reset password Page

   If a user is permitted to reset the password, he/she can input the new password at this page and confirm the new password, then submit to the backend to update their new password.

This epic has an estimated difficulty score of 4/10 with a time estimate of 2 weeks.

2.2 User Profile and Notification

After the user logged in, his/her avatar will show on the top of the website. when the user clicks the avatar there will be a drop-down menu which contains five buttons: 'My Profile', 'My Auctions'(served as Property Administration Page), 'Notifications', 'Log out', the first four of which link to respective pages.

1. User Profile Page

   A logged in user can view his/her profile on this page. If the user wants to update his/her profile, there is a **update** button on this page. After clicking it, the profile information will turns into a form with previous saved values set to be default values in the form. With targeted values altered, the user can click the **submit** button to update his/her profile.

2. Notification

   This page is a notification center which receives system notice from time to time. If an auction is successful, the seller and the highest bidder will be notified with the outcome, and the opposite side's contact details. If an auction has passed in, the seller will be informed of the passing in result, the bidding history and the highest bid.

   When a new unread message comes, there will be red dot on top of the **Notification** button in the drop down menu as a reminder.

2.3 Property

1. Property Registration Page

   When a logged in user wants to sell a property, he/she needs to register the new property for sale. The following information will be required to provide for registration:

   - address of property
   - postcode
   - suburb name
   - number of bedrooms
   - number of bathrooms
   - number of parking space in garage
   - property photos
   - property area
   - auction start date
   - auction end date
   - keywords

   After inputting the address, postcode and suburb, these information will be validated by the Google Map API. To build up the recommendation system and improve the user searching accuracy, the seller is also required to choose some

keywords that fit the property in a given keyword checkbox list. With all validations passed, the property photos will be uploaded to a cloud server (i.e. AWS S3), and the system gets the path of where the photos locate. After combining the photo path and the input text information together and transforming them into JSON format, the data is then passed to the back end by POST request and the back end saves the data in database (i.e. MySQL).

In this story, the property registration part has an estimated difficulty score of 4/10 with a time estimate of 2 weeks. This part could be done simultaneously with the User Authentication part.

By accomplishing this part, sellers are able to advertise their complete property details.

2. Property Administration Page

For a user who has registered or purchased any properties, there should be a page revealing the properties associated with the user and the properties' current status, and enabling sellers to update the information of their properties. User can find the link to this page simply by clicking the avatar after they logged in.

For a buyer user, he/she should be able to :

- View the latest highest bid of an on-going auction and his/her previous bid in this auction.
- View purchased properties in the past, and the final bids in the transactions.

When it comes to a seller user, he/she should have accesses to:

- View the latest highest bid of an on-going auction of an owned property.
- View the sold properties and their final price in transactions.
- View the bid history of both sold properties and on sale properties.
- Change the information of owned properties, and alter the reserve price of properties under auction.

This story has an estimated difficulty score of 3/10 with a time estimate of 1 week.

2.4 Buyer Related Functionalities

For potential buyers, one core functionality is to find an ideal property easily. This epic is separated into three parts: search box, search filter, and recommendation system. And another core functionality is to create new bids. This epic could be realized in the property bid page, which includes the property's information and the latest bid price etc.

1. Property Search

   Potential buyers can search properties with the elements below.

   (a) Search Box

   The search box will be displayed on the home page and the property finding page. It only accepts three kinds of input, which are:

   - Postcode
   - Suburb Name
   - Address

   A potential buyer should only type in one of these three kinds of inputs. After clicking the search button, matched properties should be returned with their thumbnail images from the back end database. If a user wants to change the location, he/she needs to type in the new address/postcode/suburb in the search box again.

   (b) Search Filter

   The user can improve the precision of results by adding filters on the following aspects:

   - A drop-down menu to choose how many bathrooms are wanted.
   - A drop-down menu to choose how many bedrooms are wanted.
   - A drop-down menu to choose how many garages are wanted.
   - A date picker to choose the auction start date and closing date.
   - Two input boxes for minimum area and maximum area.

   (c) Recommendation System

   The recommendation system records a registered user's browsing history as well as bidding history, and based on it, the system suggests unseen properties which have high similarities to the browsed ones in terms of keywords. Every time the recommendation system shows three properties that have the highest similarities with the browsed ones. The similarity calculation is decided to be the cosine similarity among keywords. If there are n keywords in total, for each property, it has a vector with n elements. For the n^{th} keyword, if a property matches, then in the property's vector the n^{th} element would be 1 and be 0 if does not. If multiple unseen properties have the same similarity to one browsed or bid property, they will be ranked in the closest distance to the recorded property.

   This epic has an estimated difficulty score of 8/10 with a time estimate of 3 weeks.

2. Property Bid

   In the property page, there are the following elements:

- A carousel of property photos.
- Property information including number of bathrooms, address, suburb etc.
- A map revealing the surroundings of the property. This could be realized by getting connected to Google Map API.
- Property current highest bid display.
- Bid history of the current property.
- An input box for the user to type in a new bid and a button for bid submission. This is only available for users who have logged in.
- Display of auction start time and end time.
- Display of user's last bid price if available.

If a buyer is interested in one property, he/she can create a new bid which must be greater than the current highest bid. An invalid bid which is lower or equal to the current highest bid will get a warning on the page and the bid will not be passed to the back end. Once the user clicks the submit button, there will be a pop-up window alert the user to double-check their decision. After the new bid submission is confirmed, the current highest bid price will get updated. Furthermore, if a bid is passed to the back end within 5 minutes right before the auction closing time, the auction closing time will be extended by 2 minutes in the database first, and then the closing time on the front end will consecutively update. Besides, once the auction closing time reaches, the bidding submission button will be disabled, and the property will not show up in the searching results.

This epic has an estimated difficulty score of 5/10 with a time estimate of 1 weeks.

2.5 Novelty

1. Computer Vision Model for Photo Classification

   If time permits, computer vision technology could be introduced to identify the amenities in a uploaded property photo to classify which section should this photo belongs to, for example, kitchen or bathroom. There are two pre-trained model available already: *ssd_mobilenet_v2* and *faster_rcnn_inception_resnet_v2*, and we can try to integrate these two models into our system with some modifications. This part could be tried if other fundamental functionalities could be realized before week 8.

   With the assistance of deep learning model, when the seller uploads the property photos, the time of tagging each photo could be saved. And in the existing systems, this practice has not been applied yet.

   This story has an estimated difficulty score of 10/10 with a time estimate of 3 weeks.

2. Recommendation System Based On Keywords

   There are no such a system in real estate auction that can recommend a user with properties in high similarity to his/her favor. In normal systems, properties which are close to your current search region are recommended. This practice is simple and fast, but not efficient enough for users because they need to apply their preference filter again.

   Combined with keyword description, recommendation is made in a higher granularity as a user's preference is projected to finer details. This can improve the user interaction with online property auction.

   This story has an estimated difficulty score of 8/10 with a time estimate of 1 weeks.

3. Dynamically update pages

   As a bidding website, it is important to update web pages on time. If a user A makes a bidding on a property when another user B is browsing that property's bidding page, updating the highest bid on the page that the user B is viewing is conducive to maintaining a good user experience.

   However, in order to achieve this, we may need to continuously extract data from the background, which may increase the burden on the background. In the later development, we are going to continue to explore optimization solutions.

2.6 Jira Backlog

All of the user stories have been put into Jira, along with an estimated number of story points. Below is a screenshot showing all issues in the Jira product backlog.

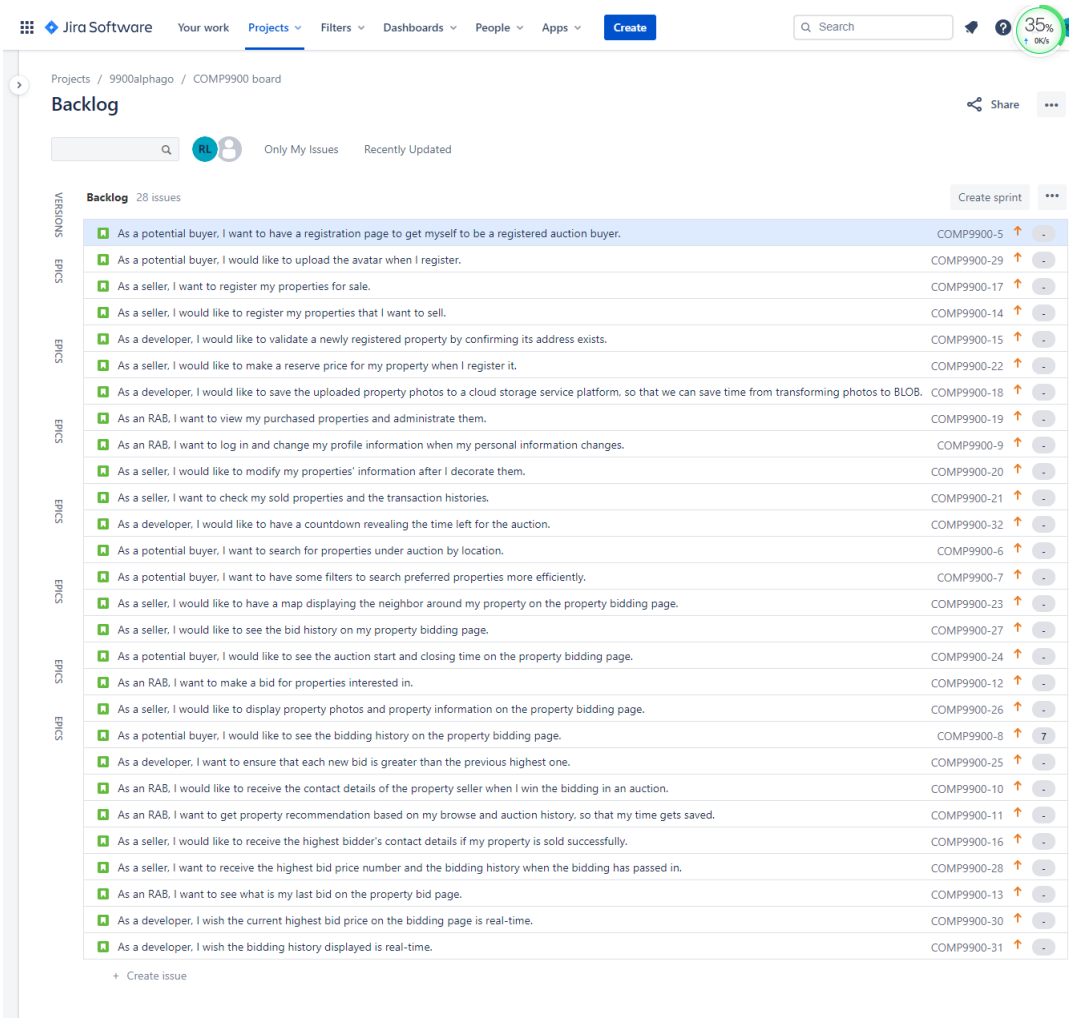


Figure 2.6.1 Product Backlog

Below is a screenshot showing one of the user stories in Jira with a detailed description and assigned story points.

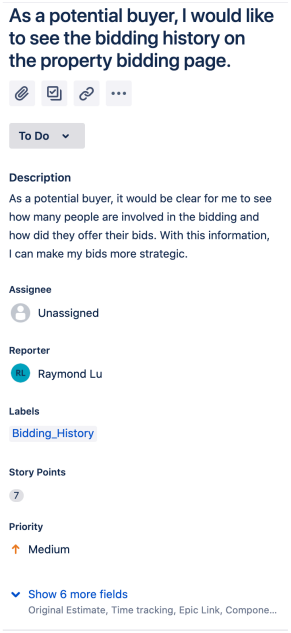


Figure 2.6.2 Detailed Story Sample

3 Project Sprints

3.1 First Sprint

Sprint	Dates	Objectives	Jira Backlog ID
1	Start: Monday, Sep 28, Week 3 Due: Thursday, Oct 15, Week 5	Basic layout of the index page should be accomplished. Also, the functionalities of user login page, user registration page, user profile page, property registration page, property administration page should be finished. Address validation should be considered, in case a non-existing address is registered. Progress Demo A is scheduled on Thursday in Week 5.	COMP9900-5 COMP9900-14 COMP9900-15 COMP9900-17 COMP9900-18 COMP9900-22 COMP9900-29

The screen shot below reveals the creation of the Sprint 1.

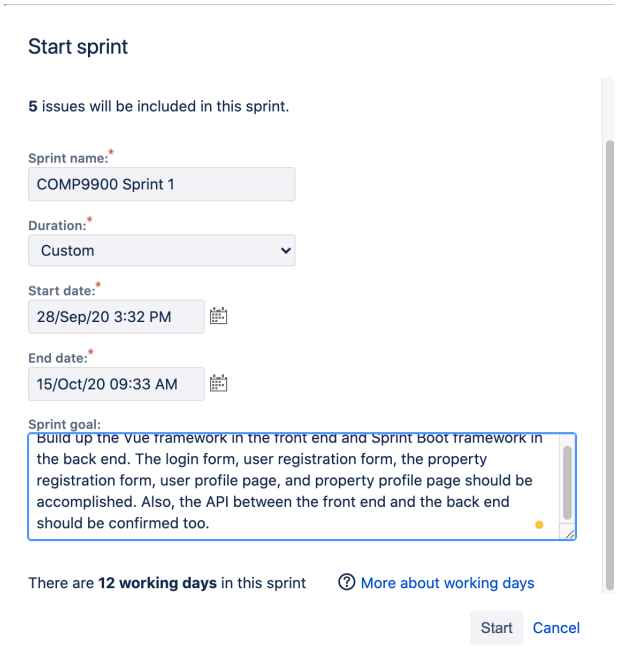


Figure 3.1.1 Sprint 1 Creation

The screen shot below reveals the commencement of the Sprint 1.

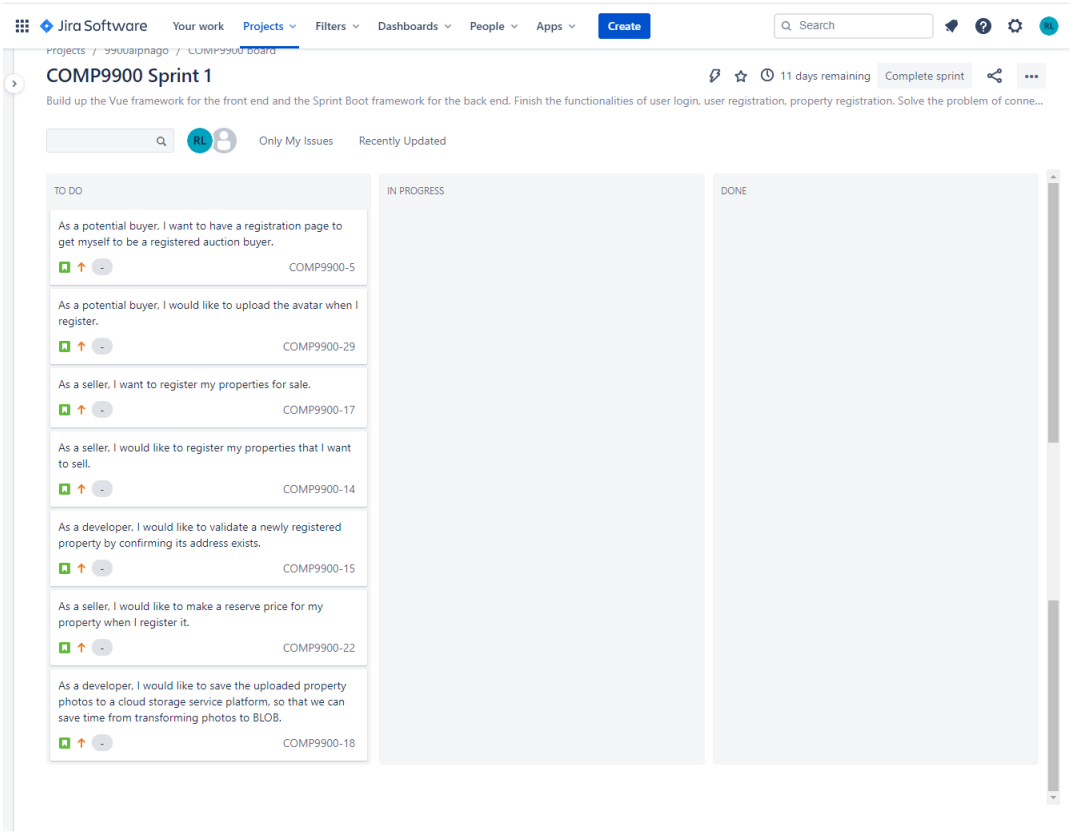


Figure 3.1.2 Sprint 1 Commencement

3.2 Coming Sprints

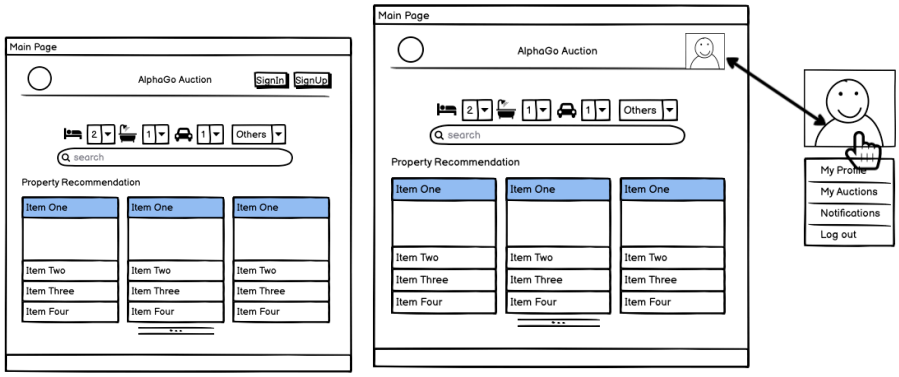
Sprint	Dates	Objectives	Jira Backlog ID
2	Start: Friday, Oct 16, Week 5 Due: Thursday, Oct 22, Week 6	User profile page and property administration page are done. An user should be able to modify his/her personal information, and view properties he/she has purchased or bidden on buy/bid page. Also, sold properties or on sale properties should be also viewed on the selling/sold page. Furthermore, a user can modify the information of his/her owned properties. If a user forgets his/her password, he/she should be able to reset it right now.	COMP9900-9 COMP9900-19 COMP9900-20 COMP9900-21 COMP9900-33
3	Start: Friday, Oct 23, Week 6 Due: Thursday, Oct 29, Week 7	Searching and filtering functionalities should have been done. An RAB is able to make a new bid. The bid is validated before submission. Also, the Google API should be introduced to the property bidding page. Besides, a countdown gadget is scheduled to be included. Retrospective A is on Thursday in Week 7.	COMP9900-6 COMP9900-7 COMP9900-12 COMP9900-23 COMP9900-24 COMP9900-25 COMP9900-32
4	Start: Friday, Oct 30, Week 7 Due: Thursday, Nov 5, Week 8	The remaining parts of the property bidding page, which are a carousel of property photos, and bidding history, should have been finished. Moreover, the notification board functionality should be added. Progress Demo B is scheduled on Thursday in Week 8.	COMP9900-8 COMP9900-10 COMP9900-13 COMP9900-16 COMP9900-26 COMP9900-27 COMP9900-28
5	Start: Friday, Nov 6, Week 8 Due: Thursday, Nov 12, Week 9	The recommendation system is accomplished, and recommended properties will be displayed on the index page. Moreover, the highest bidding price and the bidding history will be made to be real-time updated. With all basic functionalities done, test cases will be designed to check the functionalities. Retrospective B is on Thursday in Week 9. Software and report are due on Monday in Week 10.	COMP9900-11 COMP9900-30 COMP9900-31

For stories COMP9900-30 and COMP9900-31, they are optional because realizing the real-time updating could raise some unknown bugs. And this is the reason why we leave them to the last sprint.

4 Interface and Flow Diagram

4.1 Main Page

Main page contains four parts: Header, Footer, Search bar and Recommendations. After the user logs in, the *Sign In* and *Sign Up* buttons in the header becomes an avatar. When click the avatar, a drop-down menu will appear, containing four parts: My Profile, My Auctions, Notifications and Log out.

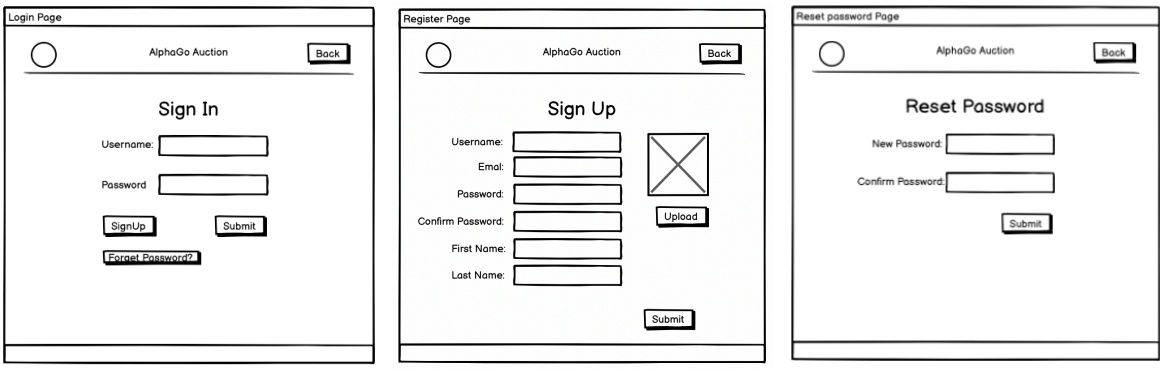


(a) Home (b) After Login

Figure 4.1 Main Page

4.2 Login and Register Page

On the Login page, registration and resetting password functions are provided.
When the user clicks the *forget password* button, this will trigger a validation email to identify the user, then push to the reset password page.

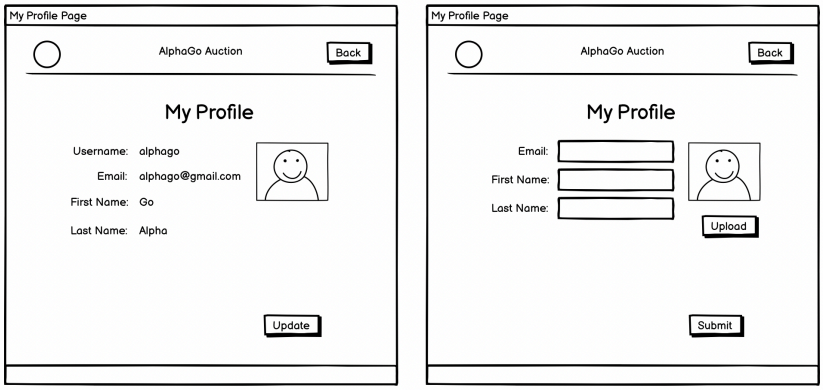


(a) Login (b) Register (c) Reset Password

Figure 4.2.2 User Authentication Page

4.3 Profile Page

A user can find this page by clicking his/her avatar after logging in. This page lists the information and provides the updating function to change his/her profile except the username.



4.3 Profile Page

4.4 Search Page

On the Search Page, a user can search specific properties based on their ideal conditions. Clicking a property's link will be directed to the corresponding bidding page.

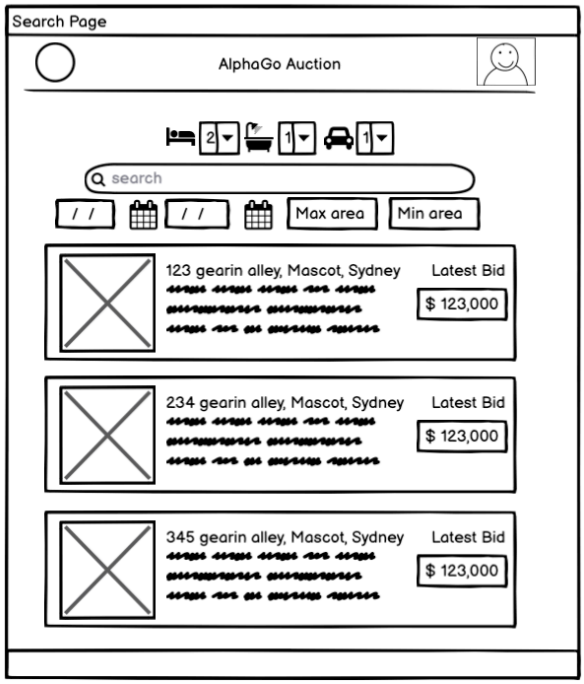


Figure 4.4 Search Page

4.5 Property Page and Property Registration Page

On the Property Page, If you have not registered as a bidder, you need to click the *Register* button and enter your payment information(Figure 4.5.2-a) before you can bid.

If you have already placed a bid on this certain property, you will see your last bid below the *Input* button. All logged-in users can place a new bid by inputting a valid bid price in the input bar on this page. After a user clicks the submit button, there will pop-up a confirmation window as Figure(4.5.2) shows.

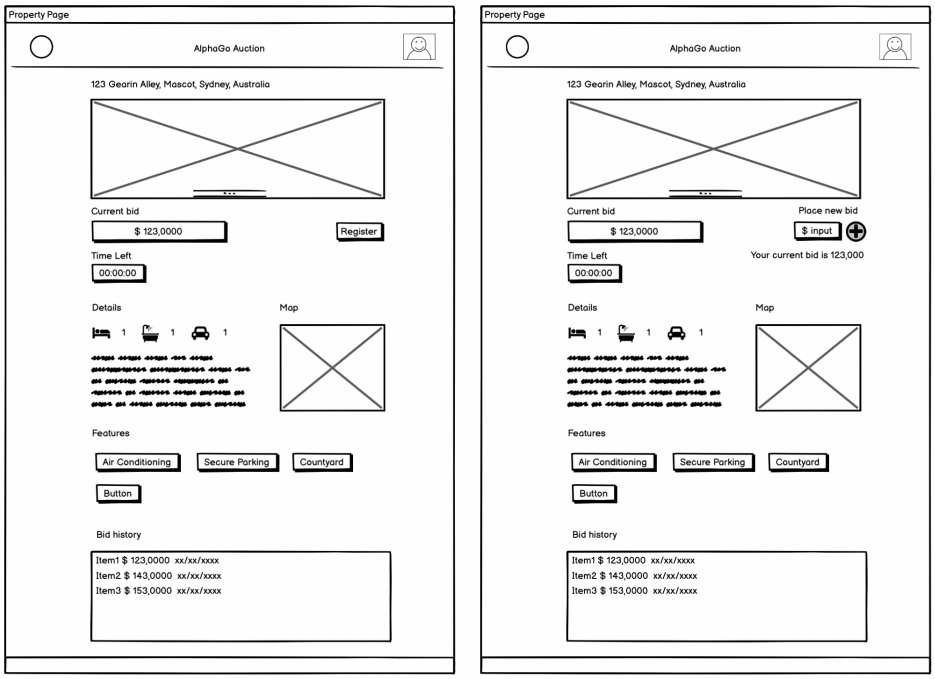
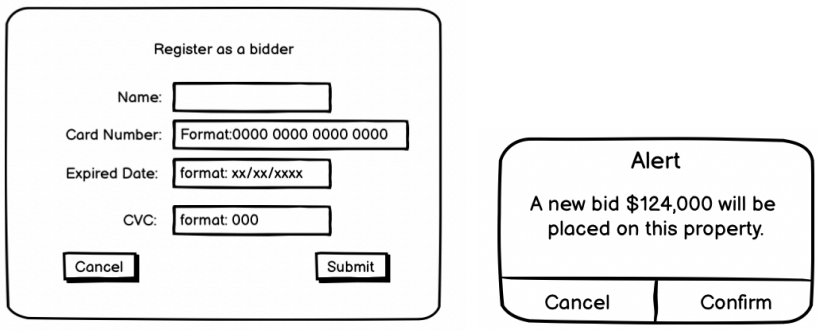


Figure 4.5.1 Property Page



(a) Register as a bidder (b) Alert

Figure 4.5.2 Register as a bidder and Alert

Property Registration will be shown after clicking the register button on the auction management page. If you are a seller, you start the auction and register your property on this page.

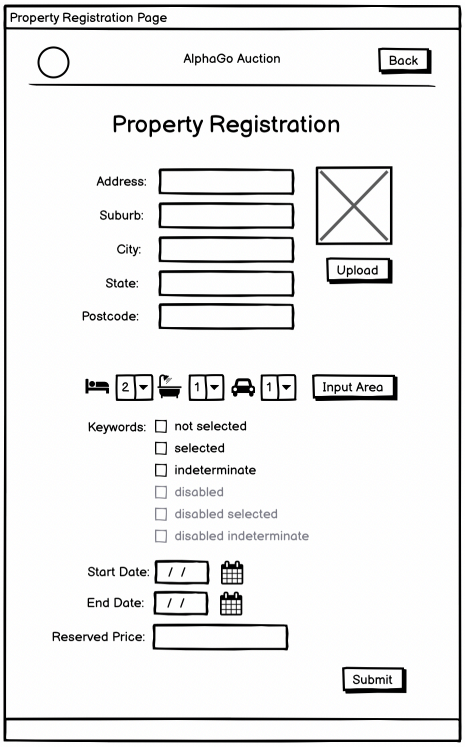


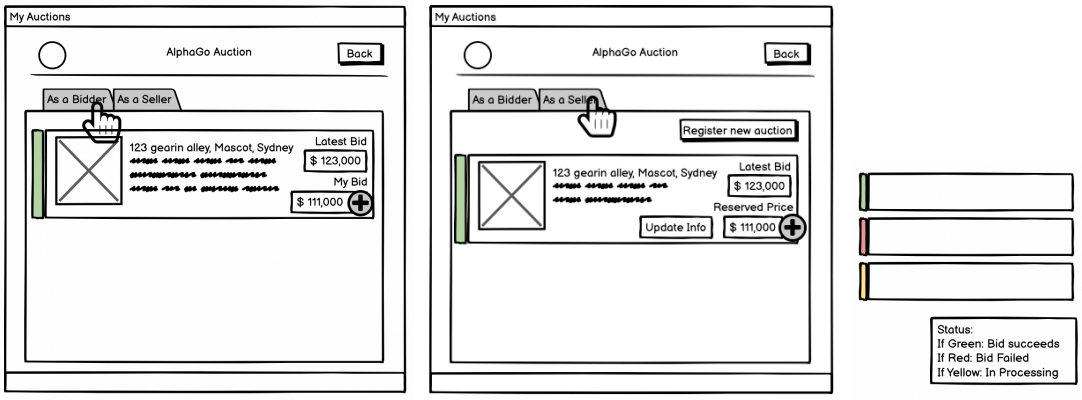
Figure 4.5.3 Property Page

4.6 Auction Management Page

The bidder management page lists the past and current bids. A user can place a new bid on his/her former auction.

On the seller management page, sellers can see the past and current bids as well. Besides, they can manage their deployed auctions, change the information or create a new auction.

In both seller and bidder page, we decided to use the colour to indicate the status of the auctions as Figure(c) shows.



(a) As a bidder (b) As a seller (c) Status

Figure 4.6 Auction Page

4.7 Notification Page

When an auction is over, the notice information will be sent to the related users.

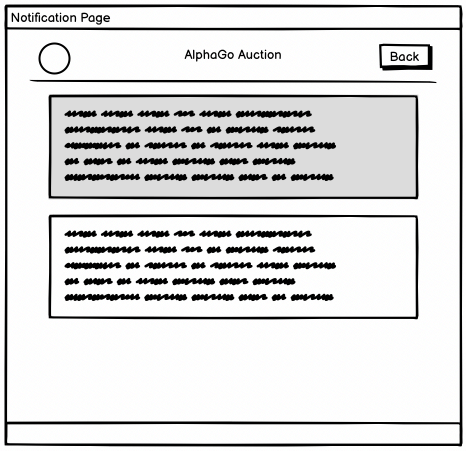


Figure 4.7 Search Page

4.8 Page Relations

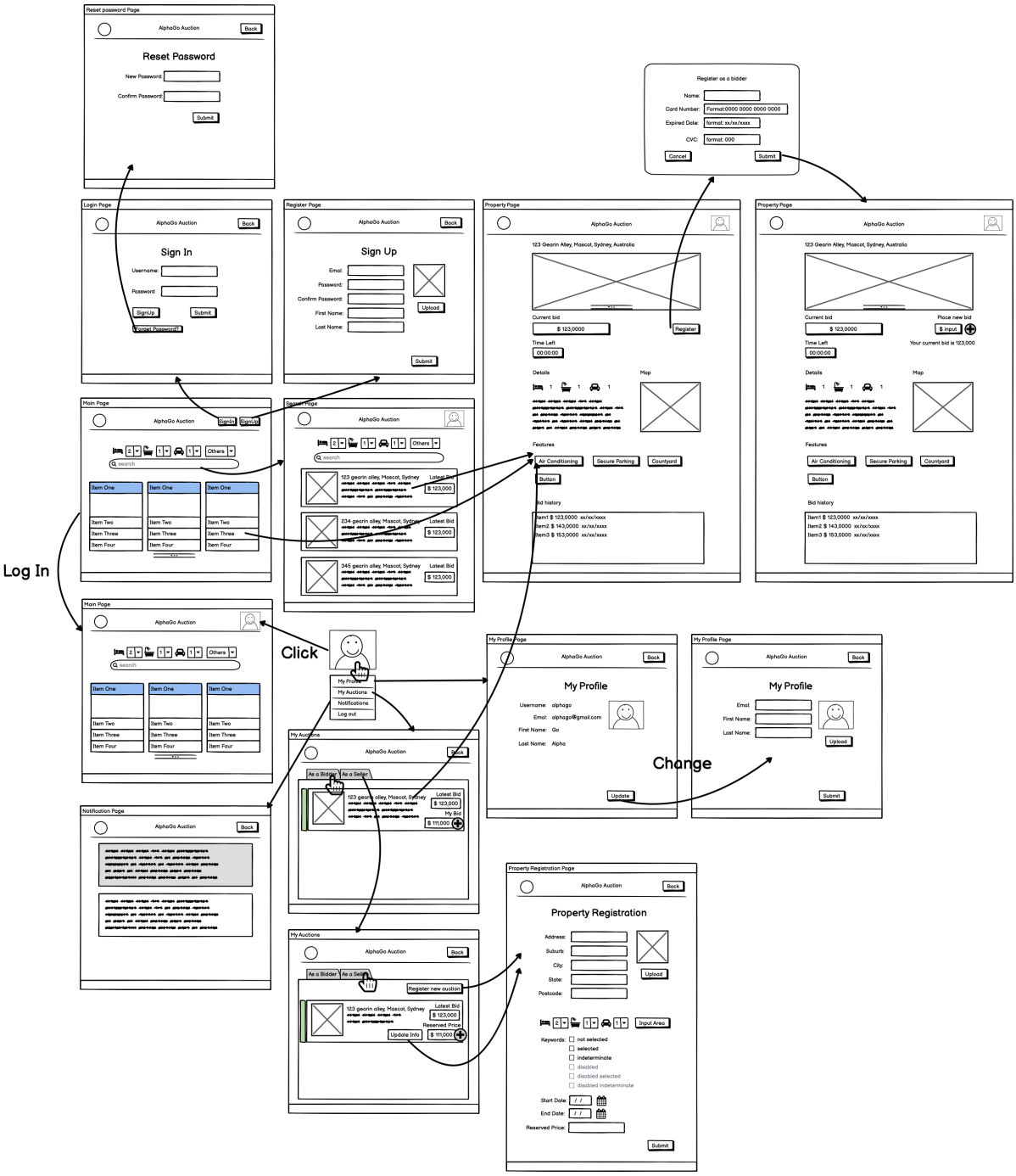


Figure 4.8 Overall Flow Diagram

5 System Architecture

5.1 Separate Front-end and Back-end system

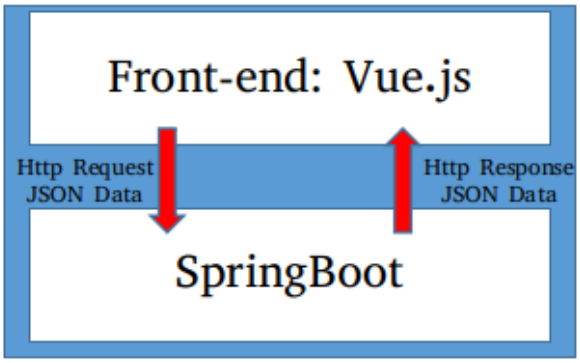


Figure 5.1 Subsystems

We choose to split the whole system into two subsystems: **Front End System** which is a graphical user interface for users to manipulate the auctions more easily and the **Back End System**, which is not accessible to the user but plays a significant role in supporting the auction platform. The two subsystems are not dependent much on each other. In our plan, we develop these two systems separately.

5.2 Front-end

We are using **Vue.js** framework for our front-end system. Comparing to traditional *HTML* and other static template, Vue framework can bring in dynamic and fancy effects. In contrast to other front-end frameworks such as Angular and React, Vue has advantages of flatter learning curve and the higher coding speed.

5.3 Back-end

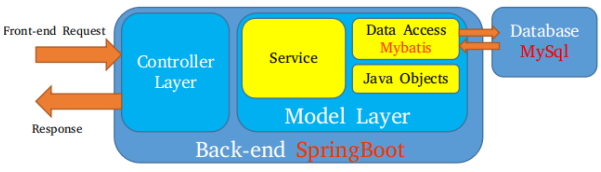


Figure 5.3 Back-end

5.3.1 Basic Framework

The language we used is **Java** and the basic framework chosen to realize the overall architecture is **Spring Boot** which embeds Tomcat server and gathers common java-web development frameworks. Thus, we can build up and deploy the whole system easily.

5.3.2 Back-end Architecture

For decoupling, we adopt a MVC three-layer architecture:

- **Model Layer**: Data persistence and process requests.
  - Service Layer: Deal with specific business.
  - Data Access Object: Communicate with database.
  - Plain Java Objects
- **View Layer**: Represent the data, in our architecture is replace by the front-end system.
- **Controller Layer**: Receive HTTP request based on the url from the Front-end and return results from the Model layer.

5.3.3 Data Access

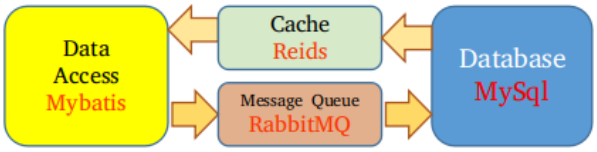


Figure 5.3.3 Data Access

In model layer, we need to fetch and update data in database. To simplify programs and improve development efficiency, we are using an ORM framework Mybatis. Besides, we want to reduce the burden of reading and writing on database. Hence, we introduce **Redis** as the cache of reading data, and a message queue **(RabbitMQ)** to improve asynchronousity and concurrency.

5.3.4 Authentication and Authorization

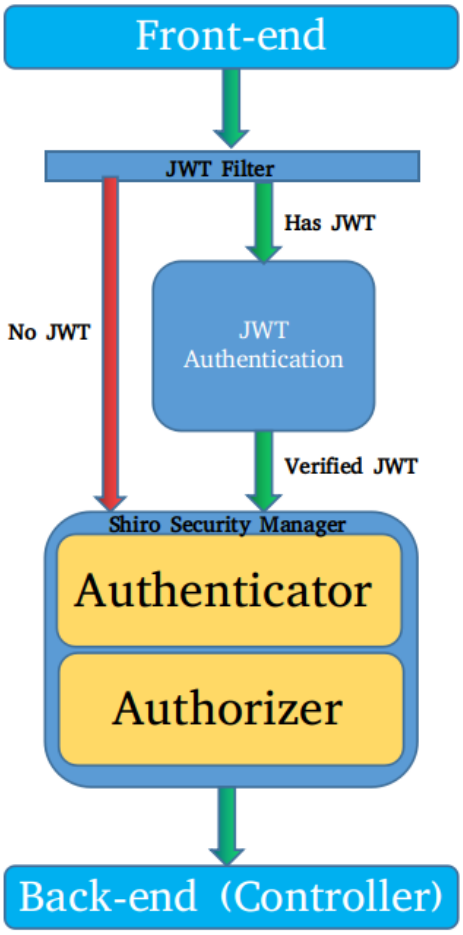


Figure 5.3.4 Authentication and Authorization

In our system, there are three roles: Anonymous, Sellers and Registered Auction Bidders (RAB). Thus, we need an efficient and simple authentication and authorization framework. Therefore, we are going to use a security framework called **Shiro** along with **J**ason **W**eb **T**oken to achieve this need.

5.4 Conclusion

Our system architecture has:

- A clear description showing the presentation and data layers in the system, and what each layer contains
- external actors and how they interact with system
- clear description of the technologies/languages planned for use

And technologies in Use:

- Front-end
  - Vue.js
- Back-end
  - Develop Language: Java
  - Framework: Spring Boot
  - Database in USE: MySQL
  - Persistence Layer: MyBatis
  - Cache and NoSQL: Redis
  - Message Queue: RabbitMQ
  - Authentication and Authorization: Shiro + JWT
  - Cloud: AWS

6 References

- Openn.com.au. 2020. Openn Negotiation: Real Estate Online Auction App | Openn Negotiation. [online] Available at: <https://www.openn.com.au/>
- Davidson, J., Liebald, B., Liu, J., Nandy, P., Van Vleet, T., Gargi, U., ... & Sampath, D. (2010, September). The YouTube video recommendation system. In Proceedings of the fourth ACM conference on Recommender systems (pp. 293-296).
- Zhang, P., & Von Dran, G. M. (2000). Satisfiers and dissatisfiers: A two-factor model for website design and evaluation. Journal of the American society for information science, 51(14), 1253-1268.