



|  |
| --- |
| **Project1 - Buyer/Seller matchmaking website**  **Technical Solution Design** |



Version: V1.0

Date: 22/05/2017

Sponsor: RMIT

Author: Jiajun He, Jason Le, Saelon Kim

Document Control

**Distribution**

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Issued** | **Recipient** | **Position** |
| 1 | 21/05/2017 | Homy Ashrafzadeh | Supervisor |
|  |  |  |  |

**Amendment History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Section** | **Page** | **Version** | **Comment** |
| <Enter Doc. Section No.> | <Enter Page No.> | <Enter Version No.> | <Enter Comments to explain the reason for the document text or other changes,    e.g., Updated text after walkthrough with the stakeholders, or    e.g., Updated section after technical consultation> |
|  |  |  |  |

.

**Staff or Entities Consulted**

|  |  |
| --- | --- |
| **Name** | **Position / Organization** |
| Jiajun He | Development team |
| Jason Le | Project Leader |
| Saelon Kim | Development team |

**Related Documents**

|  |  |  |
| --- | --- | --- |
| **Name** | **Author** | **Description** |
| <Enter Document Name> | <Enter Author> | <Enter Document Description> |
|  |  |  |

***Preface***

The purpose of this document is to outline the Technical Solution Design for the Budding Sharemarket Investor application project.This document describes the technical environment, overall project architecture, system architecture, application functionalities and features, database architecture, implementation instructions, non-functional specifications, summary of test results, known issues and risks.

[**1. Introduction**](#_xvbmwprndg7x) **6**

[**2. Technical Environment**](#_158op5pdgz9) **7**

[**3. Overall Architecture**](#_2kwthinhb037) **9**

[**4. System Architecture**](#_25jwuhbqfk1i) **10**

[4.1 Functionalities/features](#_nm2cgovgyg17) 10

[4.1.1 Register](#_163108fj05e7) 10

[4.1.2 Login/Logout](#_1x9dm0qs5q05) 11

[4.1.3 Buy Shares](#_cy71w5al9cch) 12

[4.1.4 Sell Shares](#_3h04q6fvai6u) 14

[4.1.5 Application provides live updates of share pricing and fluctuations](#_p1p48l869zkq) 15

[4.1.6 Application provides player’s current balance](#_uzrs13j8mzcm) 15

[4.1.7 Application maintains an updated leader board](#_t2gdruqk0jff) 16

[4.1.8 Application offers appropriate admin functionality](#_jdwp58c0a02o) 16

[**5. Database Architecture**](#_g25rt76gdc92) **17**

[**6. Implementation Instructions**](#_3azz36iqykn) **18**

**7**[**. Summary of test results**](#_sqr022plndrp) **19**

**8**[**. Known Issues & Risks**](#_bxifl0fmaniz) **19**

**9**[**. Appendix**](#_299l28uhgi6d) **19**

# 

# 

# 

# 

# 

# 

# 1. Introduction

Summary of the technical solution that was completed including:

· Brief project description

· Brief description of technical environment

· Estimated level of complexity

Project1 - The buyer/seller matchmaking website is a B-To-B based platform mainly focus on the food industry, it allows local farmers / food suppliers and buyers to find their potential client/business partner much easier and efficient. The platform will allow users to: create an account based on their preference type, log into the website and look for their interested client/supplier and get each other’s contact details to contact with. Data will be retrieved via a web service.

The matchmaking website will be hosted online on an AWS environment and built with a combination of PHP, mySQL, HTML and CSS. Users will be able to navigate to the site, login or register, and create/edit a post, or search for items of their liking. There will also be admin functionality built into the site. GitHub will be used for a version control tool, and some testing tools such as Lean testing.

The application will initially display a lower level of complexity and as the project development progresses, it will transition into a moderate level of complexity while remaining objective to the project requirements.

# 

# 2. Technical Environment

|  |  |
| --- | --- |
| **Component** | **Description** |
| **PHP** | PHP is used for all server-side scripting to ensure secure access and modification of data. It also has the ability to do database queries and has a more secure login. |
| **HTML5** | Used to create the web pages of the project |
| **CSS3** | Used to style the web pages |
| **AJAX** | Used for API functions and layout design |
| **Laravel** | The main framework used to build the PHP project including structuring the location of pages, styling, scripts and databases. |
| **GitHub** | Used for hosting our source control. |
| **Sublime Text, Notepad++** | Tools of choice, used individually by group members to perform coding tasks. |
| **Composer** | A tool for dependency management in PHP. |
| **AWS** | AWS will be used to host the project. It will allow tracking of different versions of our project and where the client can interact with the product: ec2-52-62-25-8.ap-southeast-2.compute.amazonaws.com/test2/project1/public/ |
| **Lean Testing** | Used to plan, track and record tests and bugs. |
| **Trello** | Online tool used to manage the project sprints - current in progress, coming up next week, completed, and ongoing. |
| **XAMPP** | Used locally to run the project individually before pushing to GitHub |
| **Google Drive, Docs, Sheets** | Used to storage and collaboration of all documentation related to the project. |

# 3. Overall Architecture

## **Class Diagram**

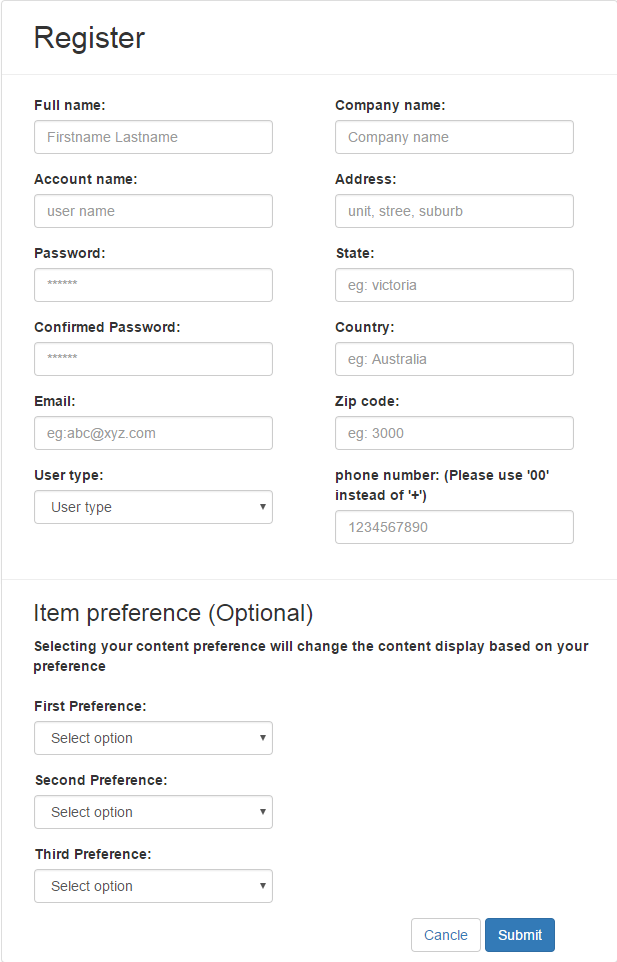
# class diagram.jpg

# 4. System Architecture

Navigate to: [ec2-52-62-25-8.ap-southeast-2.compute.amazonaws.com/test2/project1/public](http://ec2-52-62-25-8.ap-southeast-2.compute.amazonaws.com/test2/project1/public)

1. Account - Register
2. Login / Logout
3. View seller’s posts which is sorted by user interest
4. View seller’s posts
5. View buyer’s posts
6. Navigation bar
7. View buyer/seller’s user profile and post history
8. View product description of the post / Contact seller or buyer
9. Search items (search bar)
10. Create post
11. Account details (My account) / Delete account
12. Edit profile
13. Change password
14. My post history / Delete post / Edit post
15. View admin functionality

### **1. Register**



1. Navigate to the website:

<http://ec2-52-62-25-8.ap-southeast-2.compute.amazonaws.com/test2/project1/public>

1. Click on the sign up button
2. Fill out the Name field (Firstname Lastname)
3. Fill out the Account Name field (User account name)
4. Fill out the Password field (Password must be at least 6 characters, must contains at least one uppercase, one lowercase and one number)
5. Confirm the password chosen by typing it again into the Confirmed Password field
6. Fill out the Email field
7. Select the User type
8. Fill out the Company Name field
9. Fill out the Address field
10. Fill out the State field
11. Fill out the Country field
12. Fill out the Zip code field
13. Fill out the Phone number field
14. Choose the Item preference field (It is optional)

a. Choose the First Preference field (optional)

b. Choose the Second Preference field (optional)

c. Choose the Third Preference field (optional)

1. Finish register the account by pressing the Submit button

a. If successful, you will be redirected to the Login page

b. If validation fails, you will be notified on the form requirements such as:

- The username field is required

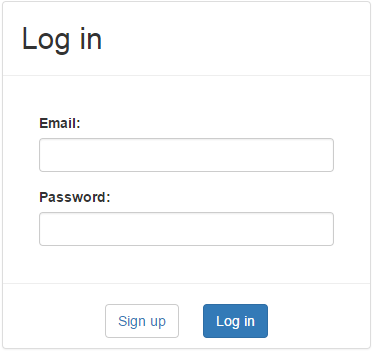
- The confirm password must be at least 6 characters

- The zip number must be a number

1. If you wish to cancel registration click the Cancel button

a. If successful, you will be redirected to the Login page

### **2. Login / Logout**



< Login >

1. Navigate to the Login page: <http://ec2-52-62-25-8.ap-southeast-2.compute.amazonaws.com/test2/project1/public>
2. Fill out the Email Address field
3. Fill out the Password field
4. Click on Login and you will be redirected to the Dashboard(index) page

\*\* User must be logged in to navigate whole website except Register page and Login page.

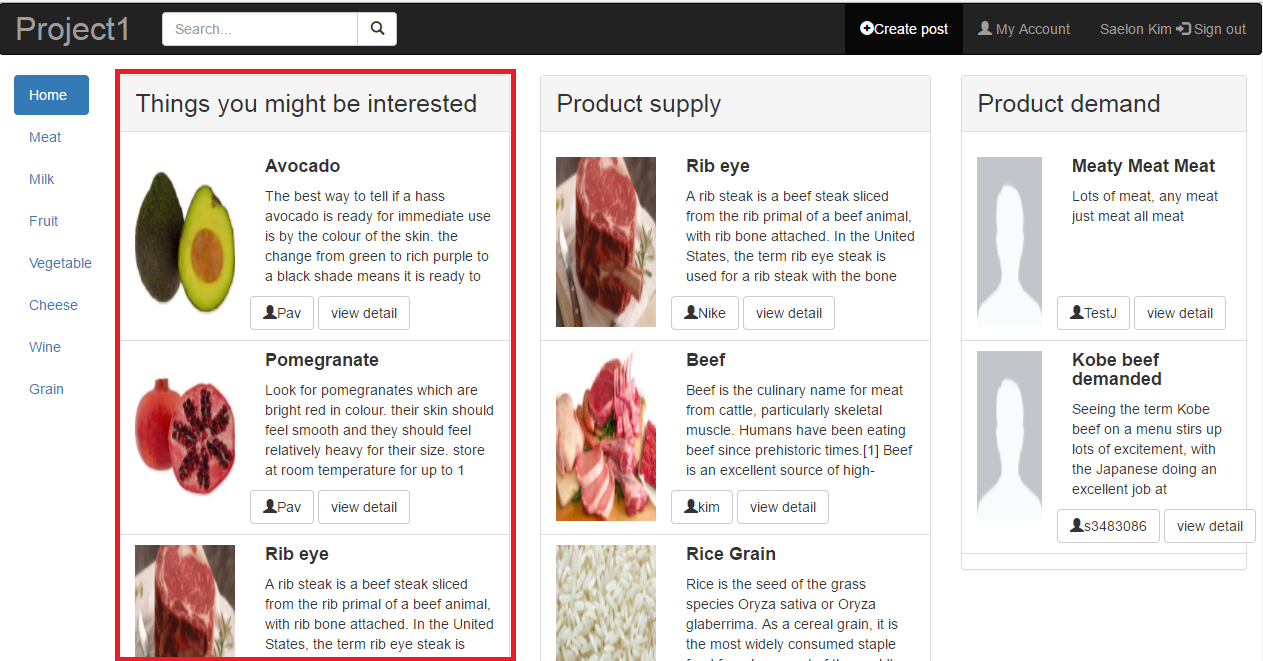


< Logout>

1. While logged in, throughout the website you have the option to logout by clicking/pressing the Logout button located on top right of the page

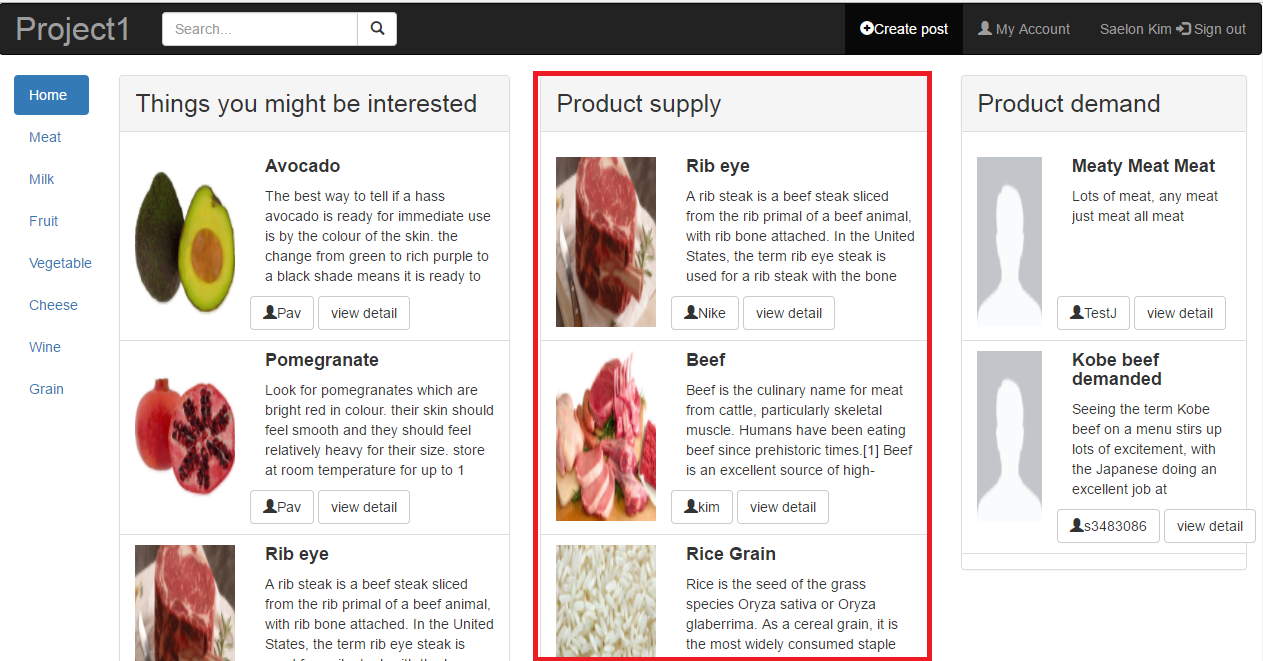
a. If successful, you will be redirected to the login page and be noticed “You have successfully logged out” message.

### **3. View seller’s posts which is sorted by user interest**



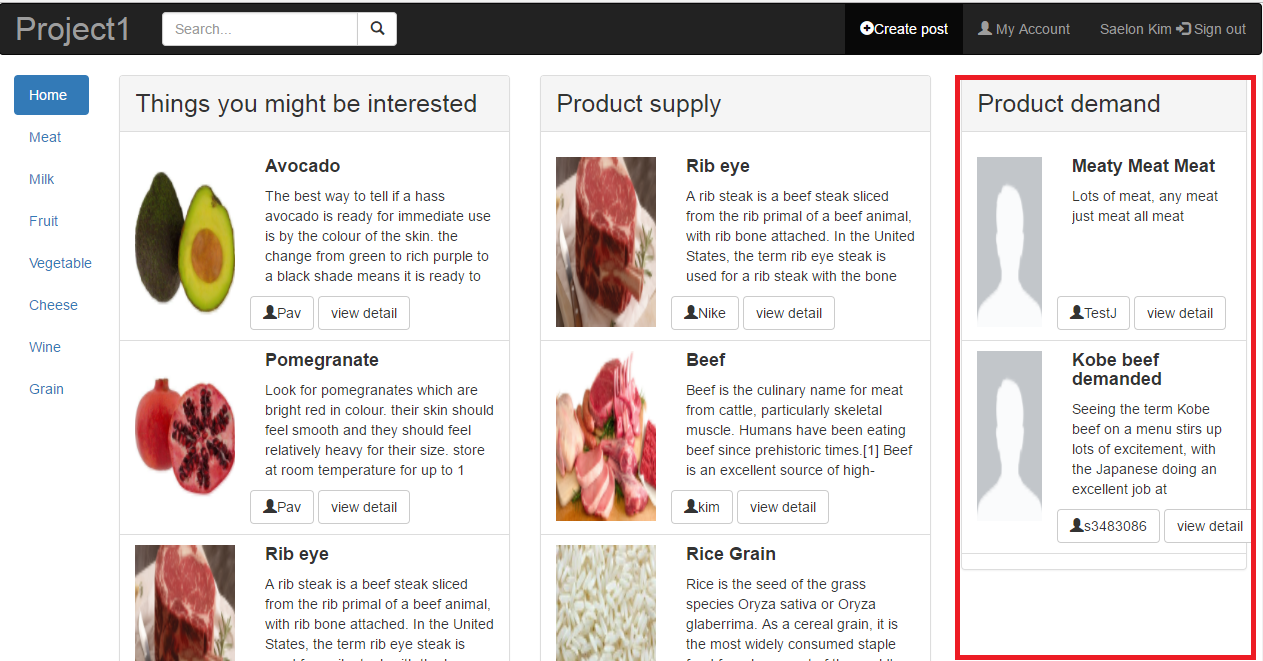
1. Navigate to Index page: <http://ec2-52-62-25-8.ap-southeast-2.compute.amazonaws.com/test2/project1/public/index>
2. View Things you might be interested section in the left second of the dashboard which includes the image of post, the post title, the post description, the user id of post and the view detail button.

### **4. View seller’s posts**



1. Navigate to Index page: <http://ec2-52-62-25-8.ap-southeast-2.compute.amazonaws.com/test2/project1/public/index>
2. View Product supply section in the right second of the dashboard which includes the image of post, the post title, the post description, the user id of post and the view detail button.

### **5. View buyer’s posts**



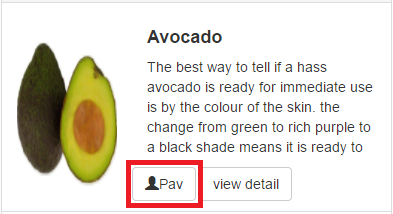
1. Navigate to Index page: <http://ec2-52-62-25-8.ap-southeast-2.compute.amazonaws.com/test2/project1/public/index>
2. View Product demand section in the right first of the dashboard which includes the image of buyer, the buyer post title, the buyer post description, the user id of buyer post and the view detail button.

### **6. Navigation bar**

|  |  |
| --- | --- |
|  |  |
| < Navigation bar > | < Click Meat - The page is redirected to display all meat posts > |

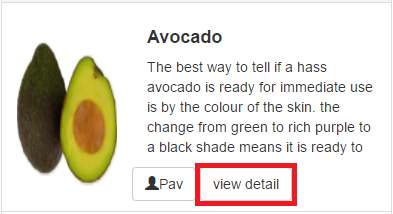
1. Navigate to Index page: <http://ec2-52-62-25-8.ap-southeast-2.compute.amazonaws.com/test2/project1/public/index>
2. Click the Meat category and you will be redirected to the display all meat posts page
3. Click the Milk category and you will be redirected to the display all milk posts page
4. Click the Fruit category and you will be redirected to the display all fruit posts page
5. Click the vegetable category and you will be redirected to the display all vegetable posts page
6. Click the Cheese category and you will be redirected to the display all cheese posts page
7. Click the Wine category and you will be redirected to the display all wine posts page
8. Click the Grain category and you will be redirected to the display all grain posts page
9. Click the Home button and you will be redirected to the Index page.

### **7. View buyer/seller’s user profile and post history**



1. Navigate to the website
2. Click on the user id button of buyer or seller which is in the bottom left of the post. Once you click the user id button, you will be redirected to User profile page.
3. View User profile dashboard which includes the image of user, the full name of user, the user id, the phone number, the email address and the post histories of user.

### **8. View product description of the post / Contact seller or buyer**



< View detail – type (1) >



< View detail – type (2) >

1. Navigate to the website
2. Click on the View detail button of buyer or seller post. Once you click the button, you will be redirected to Product description page

|  |
| --- |
|  |
| < Product description page > |
|  |
| < Security protection window pops up when user clicks Contact Seller (or Buyer) button > |

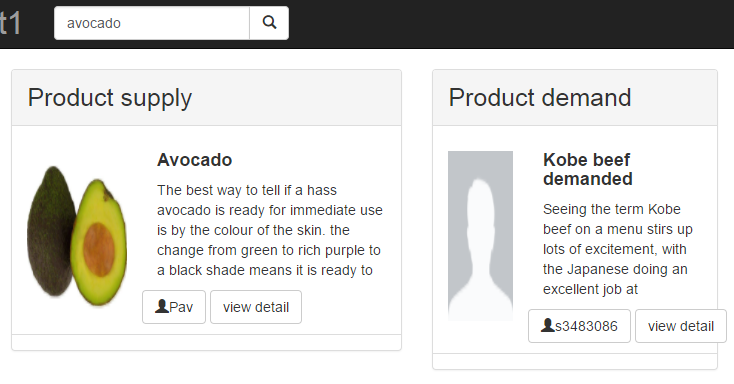
1. View Product description dashboard which includes the post title, the buyer/seller user name, the encrypted buyer/seller contact number, the encrypted buyer/seller email address, the post created date and time, the updated date and time, the type if item, and image of item and the post description
2. Click the Contact Seller (or Buyer) button to unseal the hidden contact number and email address. Once you click the Contact Seller (or Buyer) button, you will get the Security protection window.
3. Click the box on the left side of “I am not a bot” message
4. Click the Submit button in the right corner of the Security protection window.

### **9. Search items (search bar)**

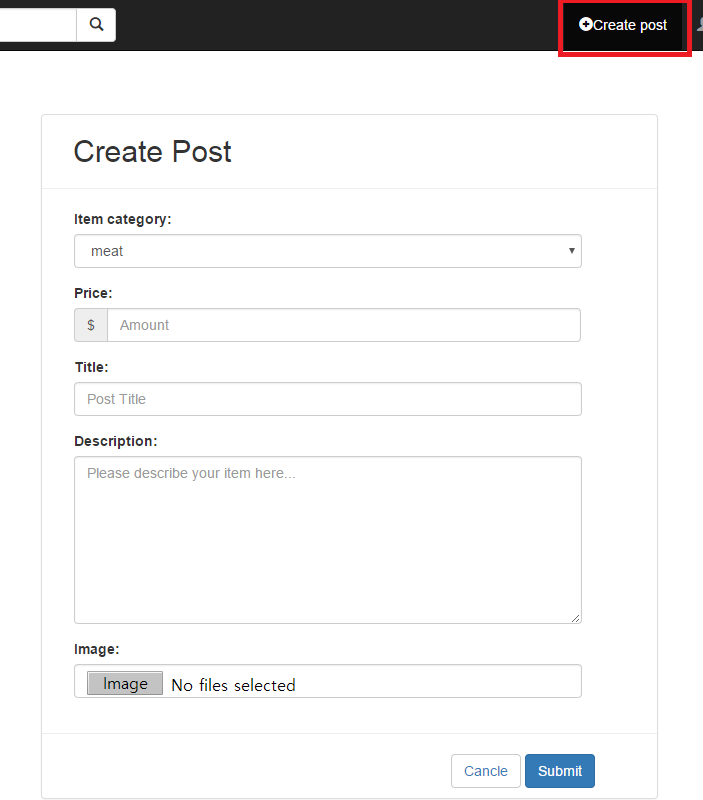


1. Navigate to the website
2. Click the search box in the left second top of the website and type specific keyword that you want to search
3. Press enter key on your keyboard or click on the magnifying glass button and you will be redirected to the search result page which is matched by keyword that you typed in

\*\* Search function doesn’t search buyer’s posts or information. Search function ONLY search seller’s posts or information.



### **10. Create post**



1. Navigate to the website
2. Click the Create post button in the third right top of the website and you will be redirected to the Create post page
3. Select the Item category
4. Fill out the Price field (Number only. Example: 20.00)
5. Fill out the Title field (Example: Fresh beef for sale!)
6. Fill out the Description field (Describe your item to sell or buy)
7. Click the Image button and upload your product image in your computer folder
8. Finish creating post by pressing the Submit button

a. If successful, you will be redirected to the Index page

b. If validation fails, you will be notified on the form requirements such as:

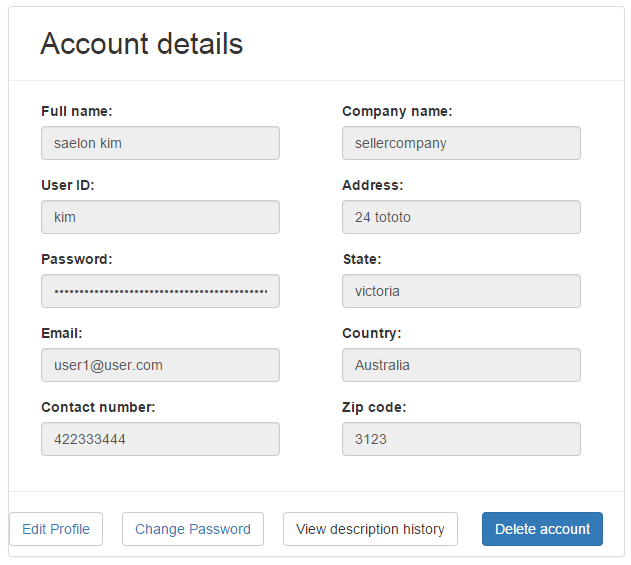
- The price field is required

- The image field is required

### **11. Account details (My account) / Delete account**



1. Navigate to the website
2. Click the My account button in the right second top of the website and you will be redirected to the Account details page
3. You can see your account detail in this page

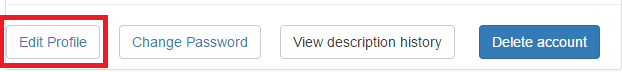


1. Click the Delete account button in the bottom end of the dashboard and you will be redirected to the Login page.

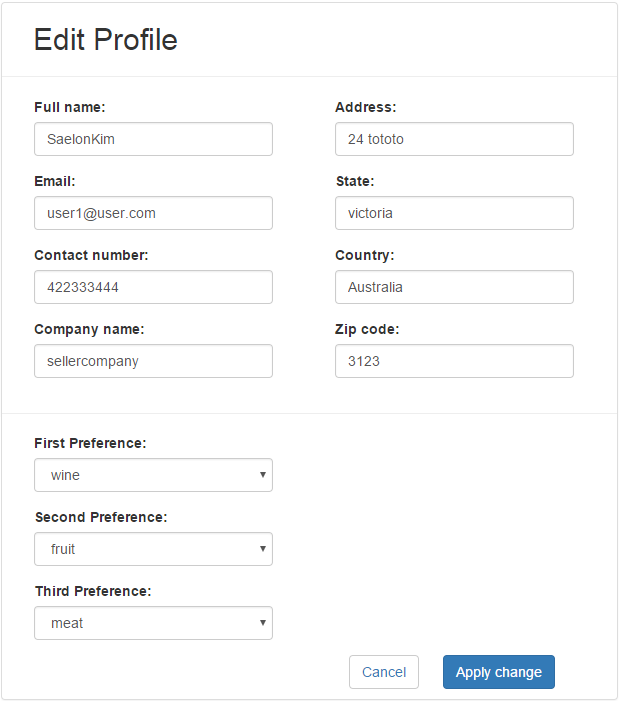
\*\* Once you delete your account, you will not be able to retrieve it. Also all your posts will be deleted.

### **12. Edit profile**

1. Navigate to the My account page: <http://ec2-52-62-25-8.ap-southeast-2.compute.amazonaws.com/test2/project1/public/my_account>



1. Click the Edit profile button in the bottom first of the dashboard and you will be redirected to the Edit profile page



1. Edit the Full name field
2. Edit the Email field
3. Edit the Contact number field
4. Edit the Company name field
5. Edit the Address field
6. Edit the State field
7. Edit the Country field
8. Edit the Zip code field
9. Select the First preference (optional)
10. Select the Second preference (optional)
11. Select the Third preference (optional)
12. Finish editing user profile by pressing the Apply change button

a. If successful, you will be redirected to the My account page and be notified by “your profile has been updated” message

b. If validation fails, you will be notified on the form requirements such as:

- The fname (full name) may only contain letters

- The email field is required

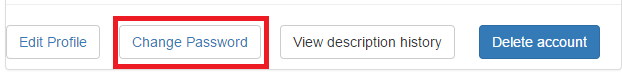
- The phonenum (phone number) must be between 10 and 13 digits

1. If you wish to cancel editing your profile click the Cancel button

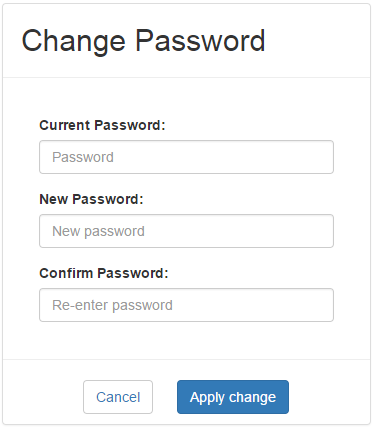
a. If successful, you will be redirected to the My account page

### **13. Change password**

1. Navigate to the My account page: <http://ec2-52-62-25-8.ap-southeast-2.compute.amazonaws.com/test2/project1/public/my_account>



1. Click the Change Password button in the bottom second of the dashboard and you will be redirected to the Change Password page



1. Fill out the Current password field
2. Fill out the New password field
3. Fill out the Confirm password field (Confirm password must match with New password)
4. Finish changing your account password by pressing the Apply change button

a. If successful, you will be redirected to the My account page and be notified by “Your password has successfully changed!” message

b. If validation fails, you will be notified on the form requirements such as:

- Current password and new password should not be the same, please try again

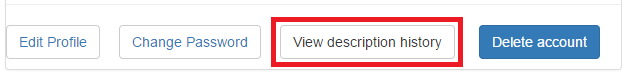
- The current password field is required

1. If you wish to cancel editing your account password click the Cancel button

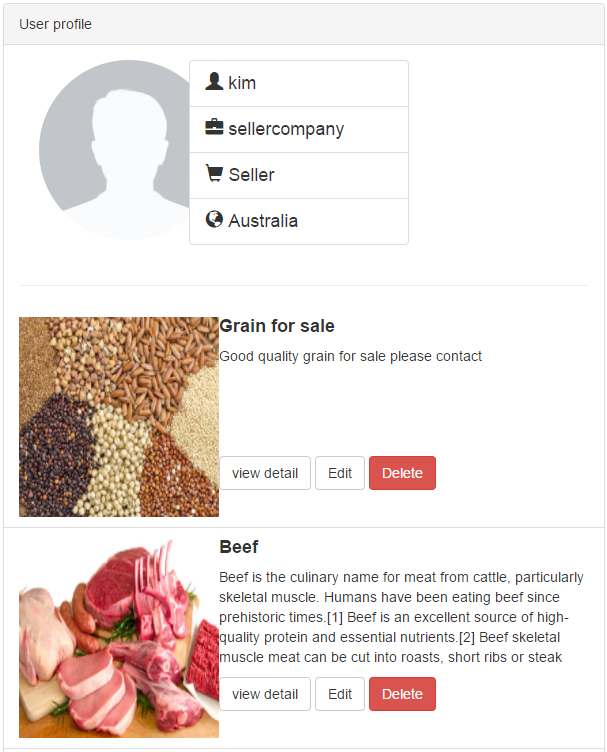
a. If successful, you will be redirected to the My account page

### **14. My post history / Delete post / Edit post**

1. Navigate to the My account page: <http://ec2-52-62-25-8.ap-southeast-2.compute.amazonaws.com/test2/project1/public/my_account>



1. Click the View description history button in the bottom third of the dashboard and you will be redirected to the User profile page



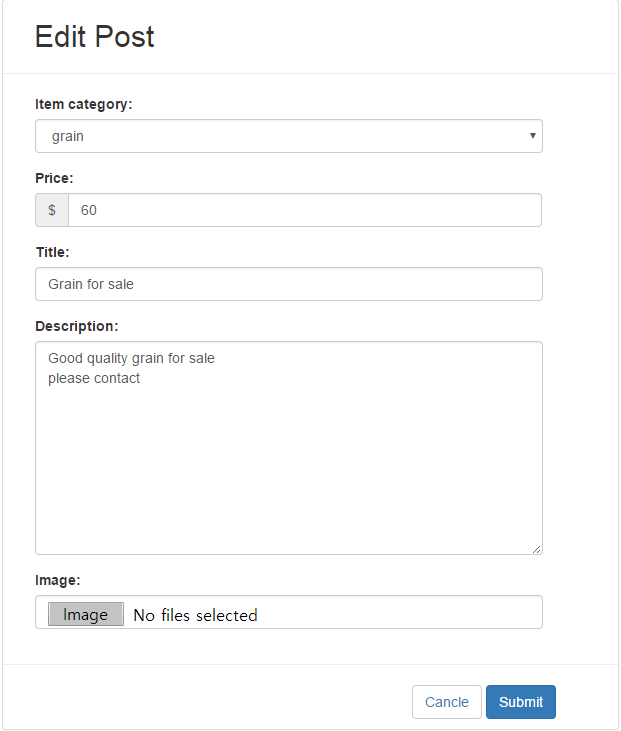
< Delete Post >

1. Click the Delete button in your post box. You will be notified “Your post has been deleted” message and be stayed in User profile page.

\*\* Once you delete your post, you will not be able to retrieve it. All your posts will be deleted from the database.

< Edit Post >

1. Click the Edit button in your post box and you will be redirected to Edit post page



1. Select the Item category field
2. Edit the Price field
3. Edit the Title field
4. Edit the Description field
5. Select the Image from your computer folder by clicking Image button
6. Finish changing your post by pressing the Submit button

a. If successful, you will be redirected to the Index page

b. If validation fails, you will be notified on the form requirements such as:

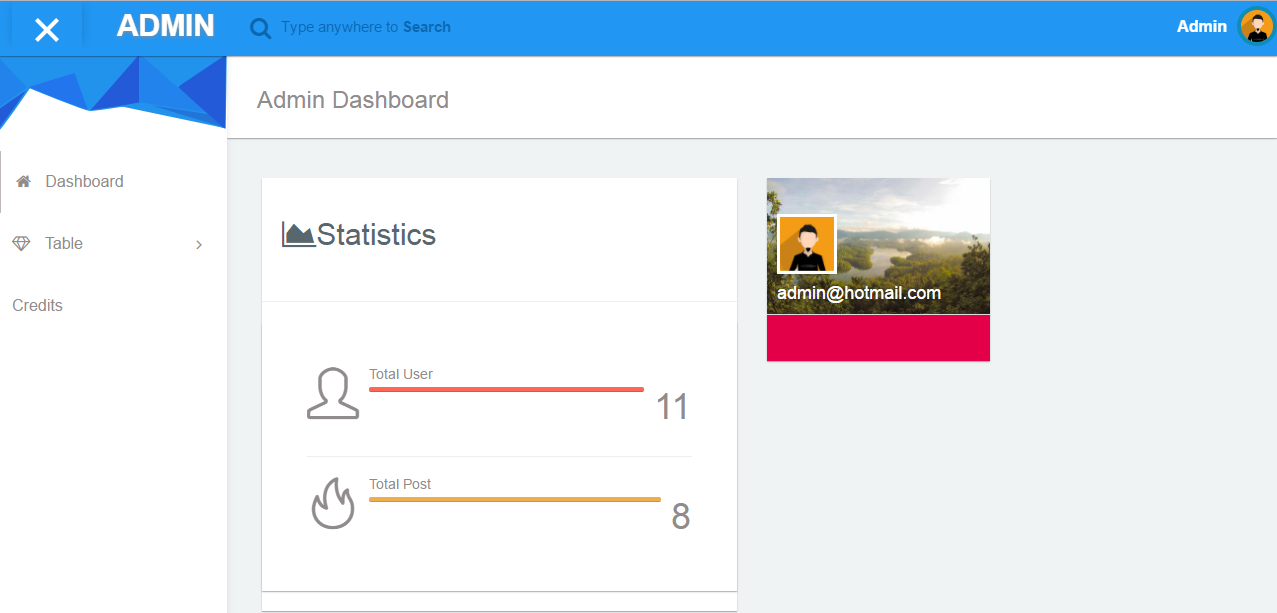
* The username field is required

1. If you wish to cancel editing your account password click the Cancel button

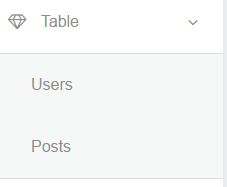
a. If successful, you will be redirected to the Index page

### **15. View admin functionality**

1. Navigate to the admin login page: <http://ec2-52-62-25-8.ap-southeast-2.compute.amazonaws.com/test2/project1/public/admin/login>
2. Log in as Admin



< Admin dashboard >



1. Click the Table in the left navigation bar
2. Select the Users from the drop down menu
3. Delete User Account from the Data tables
4. Select the Posts from the drop down menu
5. Delete User Posts from the Data tables

# 

# 

# 5. Database Architecture

phpMyAdmin has been used as the database solution for a number of reasons. The simplicity and portability of phpMyAdmin saves time during development as it allows the database to be embedded in the project files allowing easy access for all developers.The traffic of the final product is expected to be classified as low which also makes phpMyAdmin a suitable solution. Laravel, the PHP framework used to develop the application has built-in support for phpMyAdmin including the ability to have multiple databases being used concurrently. Due to the nature of the project it is unlikely that the application will be expanded to the point where a more powerful database solution would be needed.

The project features one separate databases:

|  |  |
| --- | --- |
| **Main database** | Contains user account information, post information, user preference, admin information |

**Main database**

The main database is where all user data is stored. This includes registration information, balance, transactions history and currently owned shares.



# 6. Implementation Instructions

The application only requires basic hardware since it is not requiring many CPU or RAM resources and either a Windows or MAC environment. It can be deployed on a cloud environment such as AWS free of charge.

Migrating data from a local environment to a cloud environment consists of a simple task of uploading a zipped folder, which contains the entire project, to a cloud environment such as AWS, followed by deployment which consists of selecting the most current version of the app and choosing the deploy option. The AWS systems will enable the most recent version of the website when the user navigates to the homepage.

To ensure there was no data loss, we kept the source code in a cloud environment called GitHub and also, each member of the group kept a local copy of the entire project.

Within GitHub, we have a number of commits since the beginning of the project and at any given time we can reverse changes if proven undesirable.

# 7. Summary of test results

We also used the Lean Testing tool to conduct tests and identify bugs. We have split tests into components that are related to each test. Here is a summary of the test results:

|  |  |
| --- | --- |
| Home Page | Logout |
|  | Category listing |
|  | Display Posts |
|  | Display Posts with interests priority |
| Login Page | Input User data |
|  | User login with false information |
|  | login with blank fields |
| Registration Page | Create User Account |
|  | Register with incorrect password field |
|  | Register with blank fields |
|  | Register with duplicate email |
| Post related | Create Post |
|  | Edit Post |
|  | Delete Post |
| User | Edit Account |
|  | Delete Account |
|  | View User Accounts |
|  | View Contact details |
| Search | Keyword Search |
|  | Multiple keyword search |
|  | Search through Description |
|  | Search through Title |
| URL | Admin Redirect |
|  | User Redirect |
| Admin | User Delete |
|  | Post Delete |

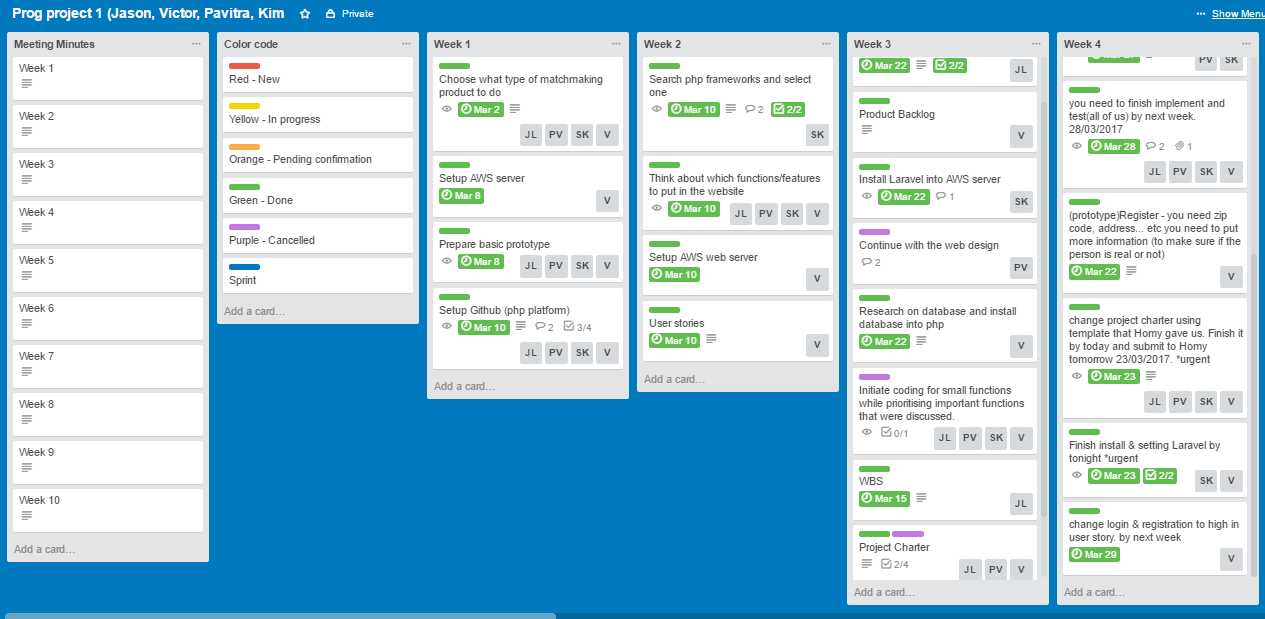
# 

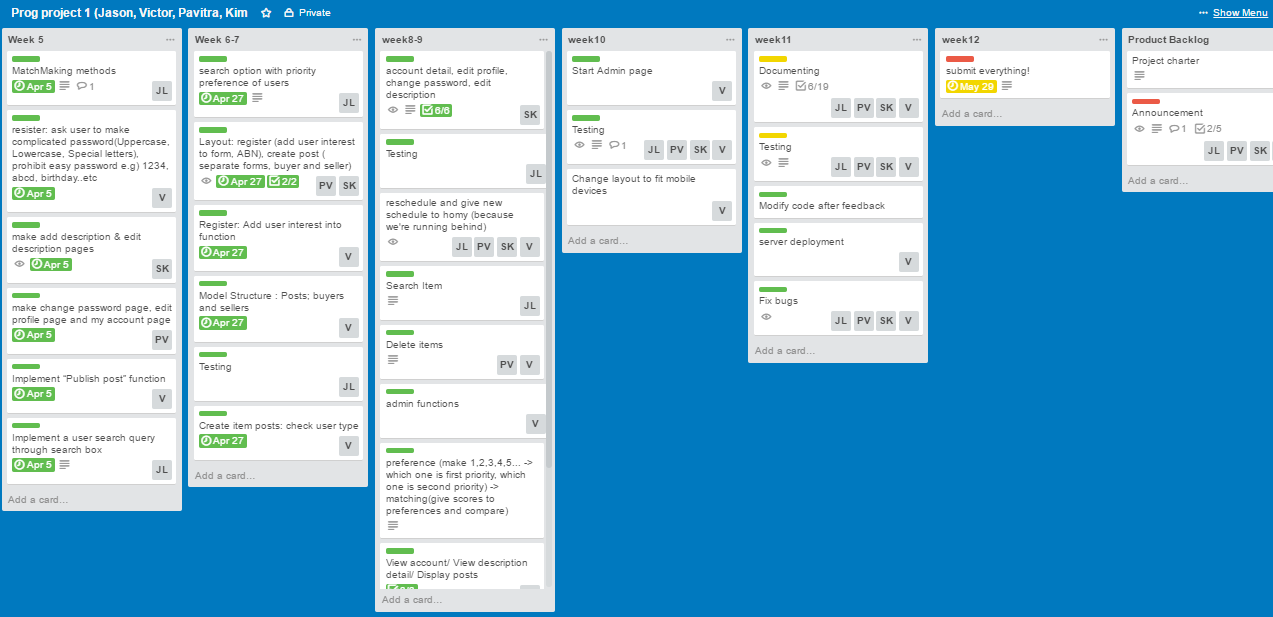
# 8. Known Issues & Risks

The Risk Register form is used to record risks related to the project.

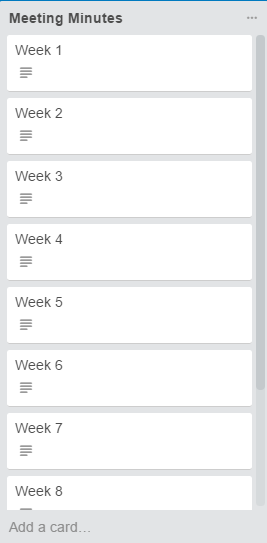
# 9. Appendix

Trello

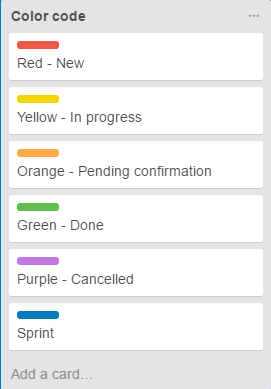




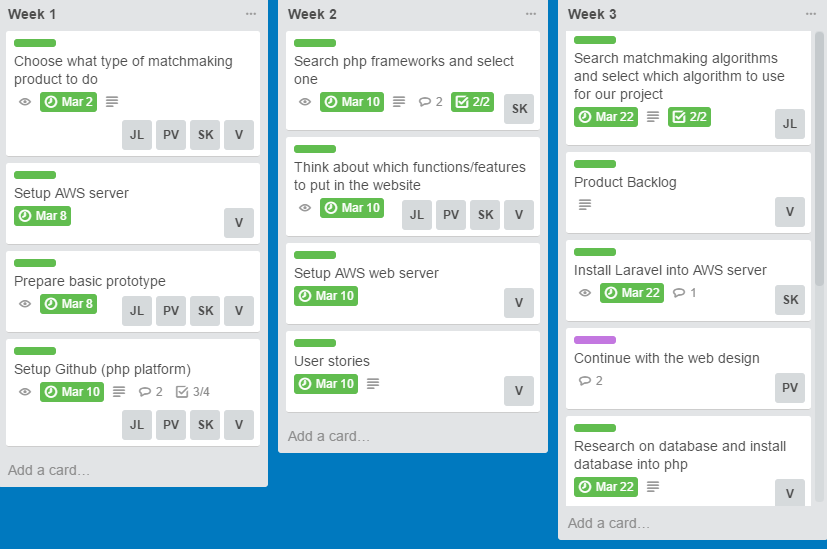
We used Trello as a tool for managing the project. We have structured Trello in Sprints according to the Scrum methodology, and grouped the sprints into columns named in a way to make it easier to track the progress of each sprint by every week.



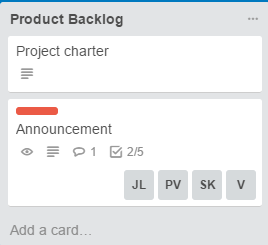
We made “Meeting Minutes” column which contains every week meeting minutes document file google drive url link.



We made a “Color code” column and checked the weekly tasks progress by different color labels.



We made "Week (1-12)" columns which contains weekly tasks and we grouped sprints by week. We have put the person who is aimed for their tasks and due date for each task. We checked all progresses by different color codes.



We made “Product Backlog” column and we created a list of product features by product requirements. This column has been changed by every week group meeting or every week meeting with supervisor.

10. Code Standard

* Functions, variables and classes are named using camelcase.
* Commenting written above important functions describing their purpose.
* Code kept simple and easy to understand.
* Methods and functions kept simple with small amount of parameters.
* Code measured and tested extensively to ensure a high standard.
* Naming is unambiguous.
* Words are spelled correctly and grammar is adhered to in comments.
* Disused/commented code is removed.