

A Development of Aquaponic E-Commerce System

Ng Zhin Yen¹, Ruhaya Ab. Aziz^{1*}

¹Faculty of Computer Science and Information Technology,
Universiti Tun Hussein Onn Malaysia, Parit Raja, Batu Pahat, 86400, MALAYSIA

*Corresponding Author Designation

DOI: <https://doi.org/10.30880/aitcs.2023.04.01.061>

Received 02 August 2022; Accepted 11 June 2023; Available online 30 June 2023

Abstract: Aquaponic is an urban farming method that promotes sustainable agriculture that is around for a long time but the knowledge of it is still at a relatively low level and the availability of aquaponic accessories is very limited. Aquaponic e-commerce system is developed to improve the availability of aquaponic accessories and increase knowledge about aquaponic for the user of the system who are customers and staffs. The functional module of this system includes login and register, manage product, make payment, manage order, manage stock and access forum. This application is developed as a web-based system with MongoDB, React.js, Node.js and Express.js through prototyping software process model. With the development of aquaponic e-commerce system, customer can purchase aquaponic accessories and gain knowledge about aquaponic practice through the forum platform. The system developed fulfilled the requirements determined and user acceptance testing is conducted. This system will hopefully encourage involvement in aquaponic practice.

Keywords: E-commerce, Aquaponic, Web-based

1. Introduction

As the world population grows, food security will become an increasingly important issue to be resolved. 'End hunger, achieve better food security and nutrition and promote sustainable agriculture' is the second of the 17 goals of the United Nations for sustainable development [1]. Humans will need more food to supply the growing demand for food. However, urbanization will also result in the loss of agricultural land which could potentially stress future food systems [2]. Urban agriculture brings benefits such as food security, reducing urban waste, improving urban biodiversity, air quality and high yields [3]. Aquaponics is one of the many methods of urban agriculture. The term 'aquaponics' means the cultivation of fish and plants in a sustainable agricultural system. The rich nutrients of aquaculture water are fed to hydroponic plants [4]. Although aquaponics brings many benefits, it is still not a well-known practice. According to a study conducted, only 33% of respondents have ever heard the term "aquaponics" [5]. This indicates that knowledge of aquaponics is still relatively low. Therefore, aquaponics should be further promoted to encourage more people to get involved in promoting

sustainable agriculture. Accessories for aquaponics should be freely available for the public to enter them easily.

MEA Aquaponics PLT is a company located in the Klang Valley that sells aquaponic accessories at physical farm and online platform via Lazada. In this project, the company was selected as a case study to conduct a needs analysis. Clients can visit the physical farm to see the products of the labor, get assistance in starting out aquaponic and purchase aquaponic equipment for their use. However, due to covid-19 pandemic, their farm is forced to close for operation until further notice. MEA Aquaponics PLT also used to sell through their website. However, clients will be redirected to Lazada instead of purchasing directly through the website. There is no option to add products to cart or instant checkout from their website.

The problems faced with the current system are client are required to physically travel to the farm. It is only suitable for clients within the area or people who happens to visit that area. On their online store, customer must go through the hassle of multiple redirections before being able to buy. In this fast-moving era, this can cause the business to lose potential customers. Besides that, the clients do not have a place to ask for help to select their equipment. Aquaponic system consist of multiple components which can be overwhelming for people starting out. This can deter them from trying out aquaponics at all. Furthermore, the current system still uses manual logging to keep track of the stocks. This can cause human error and also waste of resources.

Hence, an e-commerce system is proposed to solve this issue for both clients and admin. With this ecommerce platform, clients are able to purchase aquaponic equipment from the official site directly. Besides that, clients and admin can share and discuss about aquaponic topics in the forum. This can foster a sense of community among the aquaponics farmers. The stock count can be tracked automatically and this can further reduce the workload of admin.

This article is organized into five sections. The first part is an introduction describing the context of the project. The second section describes the related work. In the third section, the methodology is explained. The implementation and testing of this system are described in the fourth section. In the last section, a conclusion with some instructions for future employment is given.

2. Related Work

MEA Aquaponics PLT is a company which sells aquaponics accessories based in Petaling Jaya, Selangor, Malaysia. Currently, their business process mainly relies on customers visiting the farm physically. Customers can see the products and understand how aquaponics system works. Then, they can browse for the accessories or products they want. They can ask for help from staff to recommend any system or product or clarify any questions. Staff will provide recommendations based on their needs. After selecting everything the customer needs, they will pay at the checkout counter. After paying, staff will log the product stock count manually. Figure 1 shows the existing model for the MEA Aquaponics PLT.

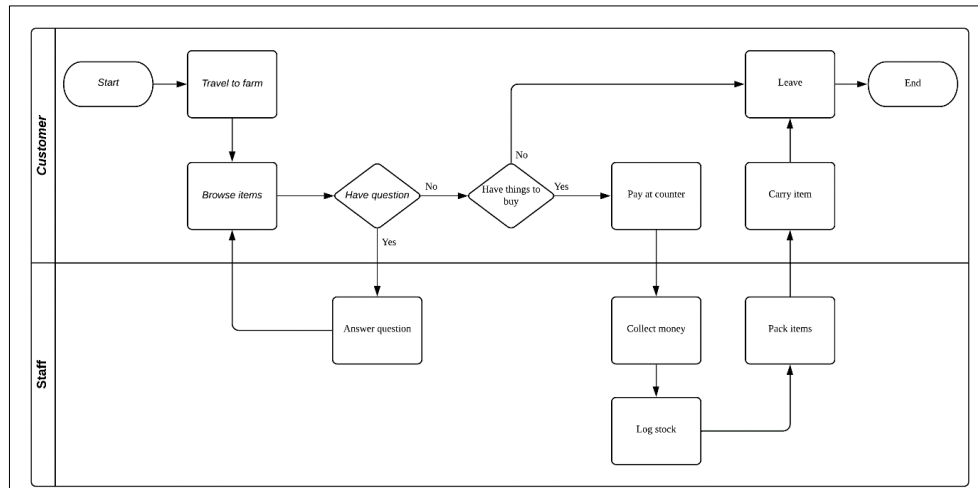


Figure 1: As-is model

The three related existing system are analyzed and compared to the proposed system. The systems are compared in terms of the module in the Aquaponic E-Commerce System. Table 1 shows the results of comparison.

Table 1: System’s Comparison

Module/ System	Lazada	CityFarm	Farm2Fork Vitaponics	Aquaponic E- Commerce System
Type	Web and mobile	Web	Web and mobile	Web
1. Register and login	Available with Google, Facebook and mobile number	Available with Google, Facebook and email address	Not available. Requires Whatsapp or Facebook account	Available with email address
2. Manage product	Available	Available	Available	Available
3. Make payment	Available with online transfer, credit card, Lazada e-wallet and Touch ‘n Go e-wallet	Available with online transfer and credit card	Not available	Available with online transfer and credit card
4. Manage order	Available with order details and delivery status tracking	Available with order details	Not available	Available with order details and delivery status tracking
5. Generate report	Available	Available	Not available	Available
6. Access Forum	Not available	Limited availability with post and comment	Not available	Available with post, comment and chat reply

From the summary of comparison, the advantages and disadvantages of each system is taken into consideration to develop the proposed system. The proposed system will allow customers to browse and search items without an account. To purchase, customers are required to login or register an account. Customers are also able to make payment, view their purchase info and delivery status. There will be a forum which allows discussion and information sharing for customers. For the administrator, they can add, update and delete product information. They can also manage and view stock balance for the store.

3. Methodology/Framework

The software process model chosen for this project is system prototyping of rapid application development (RAD). System prototyping model is chosen to quickly develop a prototype so that stakeholders can test the prototype quickly. Based on feedback from stakeholders, the system can be improved according to the stakeholders' requirement. Two iterations of prototype will be developed. System prototyping comprises of different phases of software development life cycle including planning, analysis, design, implementation and prototype. Table 2 shows the software development activities and their tasks.

Table 2: Software Development Activities

Phase	Task	Output
Planning	<ul style="list-style-type: none"> - Identify business problem and solutions. - Identify title, objective and scope. - Estimation of time of completion and milestone planning. 	<ul style="list-style-type: none"> - Proposal - Gantt chart
Analysis	<ul style="list-style-type: none"> - Analyze existing system - Requirement gathering - Requirement elicitation - Requirement negotiation - Requirement documentation - Requirement validation - Design UML 	<ul style="list-style-type: none"> - Use case diagram - Activity diagram - Sequence diagram - Class diagram - Requirements definition/ statement (functional and nonfunctional) - To-be model
Design	<ul style="list-style-type: none"> - Design user interface to visualize the overall system. - Design database of system 	<ul style="list-style-type: none"> - System architecture - Database design (schema and data dictionary) - User interface design
Implementation	<ul style="list-style-type: none"> - Code the program and conduct unit test. 	<ul style="list-style-type: none"> - System code - Test plan - Test cases
Prototype	<ul style="list-style-type: none"> - Identify and fix error and bugs in the developed prototype. 	<ul style="list-style-type: none"> - Prototype

Requirement analysis is an ordered way of identifying a suitable set of requirements to satisfy the stakeholders' needs [6]. It serves as a guideline for the requirements needed to be accomplished in the expected system. System requirements consists of functional and non-functional requirements, user requirements and system requirements. Table 3 summarizes the functional modules provided in the system.

Table 3: Functional Module of System

No.	Use case	Function	User
1.	Register and login	Allow existing customer and administrator to login to the system or new customer to register a new account	Customer and administrator
2.	Manage product	Allow administrator to add, update and delete details of product on display	Administrator
3.	Make payment	Allow customer to update and delete products in the cart and make payment	Customer
4.	Manage order	Allow customer to view their purchase history and track delivery status	Customer

		Allow administrator to view purchase history and update the delivery status	Administrator
5.	Generate report	Allow administrator to view low stock item and sales	Administrator
6.	Access forum	Allow customer and administrator to add, update and delete post and comment	Customer and administrator

According to [7], functional requirements is defined as the function the developed system must execute. Besides that, it also means the behavioral requirements that defines the input and output of the system. Table 4 shows the functional requirements of the proposed system.

Table 4: Functional Requirement of System

No	Module	Description
1.	Register and login	<ul style="list-style-type: none"> • The system shall allow user to register account with email and password. • The system shall request user to enter information during registration. • The system shall allow user to login into the system using registered email and password. • The system shall only allow user log in with valid username and password. • The system shall alert the user for any invalid input. • The system shall redirect user to their respective main menu upon successful login.
2.	Manage product	<ul style="list-style-type: none"> • The system shall allow customer to view products and product details even without an account. • The system shall allow administrator to create new products. • The system shall allow administrator to input product details and pictures when creating new products. • The system shall allow administrator to update details of existing products. • The system shall allow administrator to delete existing products.
3.	Make payment	<ul style="list-style-type: none"> • The system shall allow only customer with account access to this module. • The system shall allow customer to add products into cart. • The system shall allow customer to change the quantity of cart items. • The system shall allow customer to delete cart items. • The system shall allow customer to make payment for the cart items.
4.	Manage order	<ul style="list-style-type: none"> • The system shall allow only user with account access to this module. • The system shall allow customer to view their purchase history. • The system shall allow customer to view their delivery status. • The system shall allow administrator to view purchase history of users. • The system shall allow administrator to update delivery status of orders. • The system shall allow administrator to view the sales of the company.
5.	Generate report	<ul style="list-style-type: none"> • The system shall allow administrator to view low stock items and sales.
6.	Access forum	<ul style="list-style-type: none"> • The system shall allow customer to view posts and comments. • The system shall allow customer with account to create new post and comment. • The system shall allow customer with account to update and delete their own existing post and comment. • The system shall allow administrator to create new post and comment. • The system shall allow administrator to update and delete all comments.

Non-functional requirements are defined as the criteria about the way the system operates [16]. Table 5 shows the non-functional requirements of the developed system.

Table 5: Non-functional Requirements of System

No	Requirements	Description
1.	Performance	The loading time for the website is no more than 15 seconds.
2.	Operational	The system shall be operating 24/7.
3.	Security	The system shall authenticate and authorize user.
4.	Compatibility	The system should be able to work on any web browser.

Use cases are used to describe relationship between actors and system [17]. Use case diagram shows the overall relationship and activity of the system. Figure 2 shows the use case diagram of the aquaponic

e-commerce system. There are 2 actors who are the customer and administrator. Customer and administrator are both involved in login and register, view order details and access forum use case. Only customer is involved in make payment use case. Administrator is involved two additional use cases which are manage products and generate report use case.

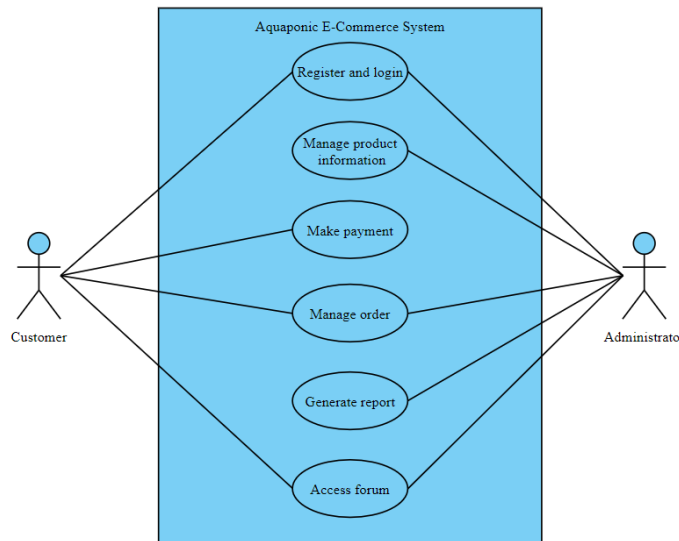


Figure 2: Use Case Diagram

Use case specification is used to describe and explain the details of use case diagrams. Table 6 shows the use case specification for register and login use case.

Table 6: Register and login use case specification

History Log	1.0.0	1. Create initial use case	
Version	1.0.0		
Use Case ID	UC-1		
Use Case Name	Register and Login		
Created By	Ng Zhin Yen	Updated By	Ng Zhin Yen
Date Created	26 November 2021	Last Revision Date	26 November 2021
Actors	Customer, Administrator		
Description	This use case allows user to login and access the system. It also allows new customer to register an account.		
Preconditions	Users need enter user id and password.		
Post conditions	The users will be redirected to the homepage.		
Normal Flow	1.0 User login 1. Users insert valid user id and password. 2. System redirects users to the homepage. 3. If the customers is new user, they need to register account first, see A.1.		
Alternative Flow	A.1 Customer registration 1. New customers insert user id, name, email address, password, and address. 2. System displays ‘Registration Successful’ pop up message. 3. Customers redirect to login page. 4. Return to step 1 in normal flow.		
Exceptions	E.1 Fail to login 1. System displays pop up to inform users that email or password is invalid. E.2 Fail to register 1. System display message to inform uses that registration failed.		
	ID	Requirement	Priority

Related requirement	SRS_REQ_101	System displays login page view.	High
	SRS_REQ_102	Users login to the system with valid user id and password.	High
	SRS_REQ_103	System displays error message if login is invalid.	High
	SRS_REQ_104	Customers register by entering personal information.	High
	SRS_REQ_105	System displays error message if the registration is not successful.	High
	SRS_REQ_106	System store user information in database once registration successful.	High

Table 7 shows the use case specification for manage product.

Table 7: Manage product use case specification

History Log	1.0.0	1. Create initial use case	
Version	1.0.0		
Use Case ID	UC-2		
Use Case Name	Manage Product		
Created By	Ng Zhin Yen	Updated By	Ng Zhin Yen
Date Created	26 November 2021	Last Revision Date	26 November 2021
Actors	Administrator		
Description	This use case allows administrators create, update and delete products.		
Preconditions	Account is logged in as the administrator.		
Post conditions	Administrator redirects to product page.		
Normal Flow	1.0 Create new product 1. Users fill in product name, price, description, category, quantity. 2. System add product to database. 3. To update existing product, see A.1. 4. To delete existing product, see A.2.		
Alternative Flow	A.1 Update existing product 1. Users select existing product to edit. 2. Users fill in updated product details. 3. System update product details in database.		
	A.2 Delete existing product 1. Users select existing product to delete. 2. System displays pop up to request confirmation for deletion. 3. Users confirms deletion. 4. System deletes product from database.		
Exceptions	E.1 Fail to add product 1. System displays pop up message "Product cannot be added successfully".		
Related requirement	ID	Requirement	Priority
	SRS_REQ_201	Administrator views list of products.	High
	SRS_REQ_202	Administrator creates new product with product details.	High
	SRS_REQ_203	Administrator updates product details of existing product.	Moderate
	SRS_REQ_204	Administrator deletes existing product.	High
	SRS_REQ_205	System displays error message if process is unsuccessful.	High
SRS_REQ_206	System stores updated information in database.	High	

Table 8 shows the use case specification for make payment.

Table 8: Make payment use case specification

History Log	1.0.0	1. Create initial use case	
Version	1.0.0		
Use Case ID	UC-3		
Use Case Name	Make payment		
Created By	Ng Zhin Yen	Updated By	Ng Zhin Yen
Date Created	26 November 2021	Last Revision Date	26 November 2021
Actors	Customer		
Description	This use case allows customers to browse and view products and product details. Customer can manage their cart and make payment.		
Preconditions	Account is logged in as customer.		
Post conditions	Order is created and user is redirected to order page.		
Normal Flow	1.0 Browse products 1. Users browse products. 2. Users click on product to view product details. 3. Users add product to cart. 4. If users wish to edit the cart, go to A.1 and A.2. 5. Users checkout and make payment.		
Alternative Flow	A.1 Edit quantity of item in cart 1. Users click on increment or decrement button to change quantity of item in cart. 2. Return to step 4 of normal flow.		
	A.2 Delete item from cart 1. Users click on remove item to delete item from cart. 2. Return to step 4 of normal flow.		
Exceptions	E.1 Fail to add product into cart 1. System displays message “Product is out of stock”.		
	E.2 Payment fail 1. System displays message “Transaction failed”.		
Related requirement	ID	Requirement	Priority
	SRS_REQ_301	Customers view products on display page.	High
	SRS_REQ_302	Customers view product details upon clicking on product.	High
	SRS_REQ_303	Customers add product into cart.	High
	SRS_REQ_304	System displays error message if product is out of stock.	Moderate
	SRS_REQ_305	Customers manage products in cart.	High
	SRS_REQ_306	Customers make payment.	High
	SRS_REQ_307	System displays error message if transaction is unsuccessful.	High
SRS_REQ_308	System store order details in database upon successful transaction.	High	

Table 9 shows the use case specification for manage order.

Table 9: Manage order use case specification

History Log	1.0.0	1. Create initial use case	
Version	1.0.0		
Use Case ID	UC-4		
Use Case Name	Manage order		

Created By	Ng Zhin Yen	Updated By	Ng Zhin Yen
Date Created	26 November 2021	Last Revision Date	26 November 2021
Actors	Customer, Administrator		
Description	Users can view and manage orders.		
Preconditions	Users is logged in to the system.		
Post conditions	The users will be redirected to the order page.		
Normal Flow	1.0 View order page <ol style="list-style-type: none"> 1. Users selects order. 2. System displays order details. 3. If customer wants to cancel order, see A.1. 4. If administrator wants to edit order status, see A.2. 		
Alternative Flow	A.1 Cancel order <ol style="list-style-type: none"> 1. Customers cancel order. 2. System request confirmation from customer. 3. System redirects customer to order page. 4. Return to step 1 of normal flow. 		
	A.2 Edit order status <ol style="list-style-type: none"> 1. Administrators change order status. 2. System redirects to order page. 3. Return to step 1 of normal flow. 		
Exceptions	None		
Related requirement	ID	Requirement	Priority
	SRS_REQ_401	Users view order history and details.	High
	SRS_REQ_402	Customers cancel order.	Low
	SRS_REQ_403	Administrator updates order status.	Low

Table 10 shows the generate report use case specification.

Table 10: Generate report use case specification

History Log	1.0.0	1. Create initial use case	
Version	1.0.0		
Use Case ID	UC-5		
Use Case Name	Generate report		
Created By	Ng Zhin Yen	Updated By	Ng Zhin Yen
Date Created	26 November 2021	Last Revision Date	26 November 2021
Actors	Administrator		
Description	This use case allows administrator to view report		
Preconditions	Account is logged in as administrator		
Post conditions	User redirected to stock page.		
Normal Flow	1.0 Go to dashboard <ol style="list-style-type: none"> 1. Administrator views earnings. 2. Administrator views order status by day. 3. Administrator views low stock items 		
Related requirement	ID	Requirement	Priority
	SRS_REQ_501	Administrator views list of low stock count items.	Moderate
	SRS_REQ_502	Administrator views charts of earnings and order status.	Moderate

Table 11 shows the use case specification for access forum use case.

Table 11: Access forum use case specification

History Log	1.0.0	1. Create initial use case	
Version	1.0.0		
Use Case ID	UC-6		
Use Case Name	Access forum		
Created By	Ng Zhin Yen	Updated By	Ng Zhin Yen
Date Created	26 November 2021	Last Revision Date	26 November 2021
Actors	Customer, Administrator		
Description	Users can view posts and comments. Registered users can add, update and delete posts and comments.		
Preconditions	None		
Post conditions	The users will be redirected to forum page.		
Normal Flow	1.0 View forum 1. System displays forum with posts. 2. Users click on post to view comment. 3. If user wants to create post or comment, see A.1. 4. If user wants to edit post or comment, see A.2. 5. If user wants to delete post or comment, see A.2.		
Alternative Flow	A.1 Create new post or comment. 1. Users enter content for post or comment. 2. Users save changes. 3. Return to step 1 of normal flow.		
	A.2 Edit existing post or comment. 1. Users select post or comment made by themselves. 2. Users enter new content for post or comment. 3. Users save changes. 4. Return to step 1 of normal flow.		
	A.3 Delete existing post or comment. 1. Users select post or comment made by themselves. 2. Users click delete button. 3. Users confirm deletion 4. Return to step 1 of normal flow.		
Exceptions	None		
Related requirement	ID	Requirement	Priority
	SRS_REQ_601	Users view all post and comment in the forum	High
	SRS_REQ_602	Users create new post or comment.	High
	SRS_REQ_603	Customers edit existing post or comment by them.	Moderate
	SRS_REQ_604	Customers delete existing post or comment by them.	Moderate
	SRS_REQ_605	Administrator edits existing post or comment by all users.	High
	SRS_REQ_606	Administrator deletes existing post or comment by all users.	High
SRS_REQ_607	System stores the post and comment in database.	High	

In this system, there are a total of 10 classes. Customer and administrator are both users. Users have access to post and comment. Customer can create order which contains order details. Customer can also access the cart. Both cart and order details have product which have one category. Figure 3 shows the class diagram of aquaponic e-commerce system.

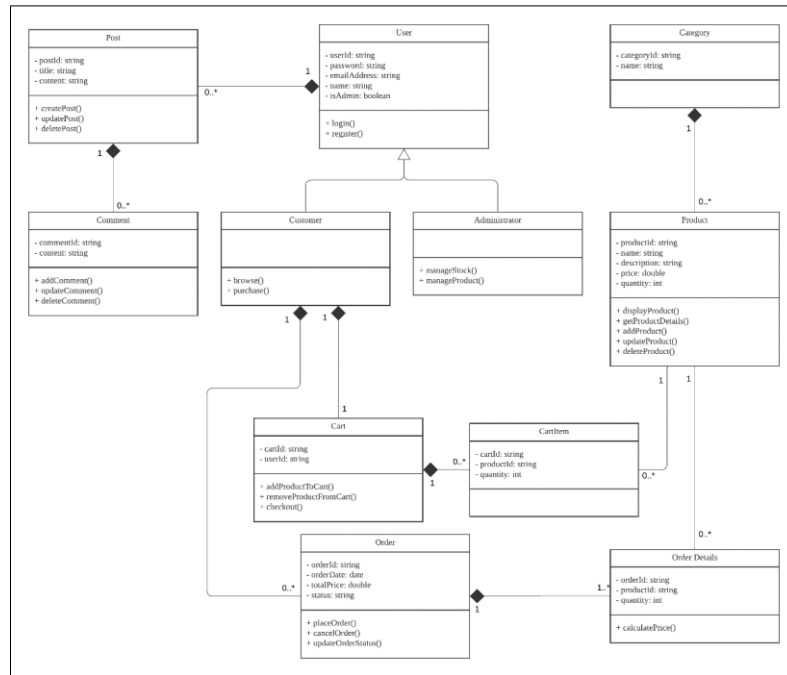


Figure 3: Class Diagram

To-be model illustrates the flow of the expected system. The to-be model of aquaponic e-commerce system is shown in Figure 4.

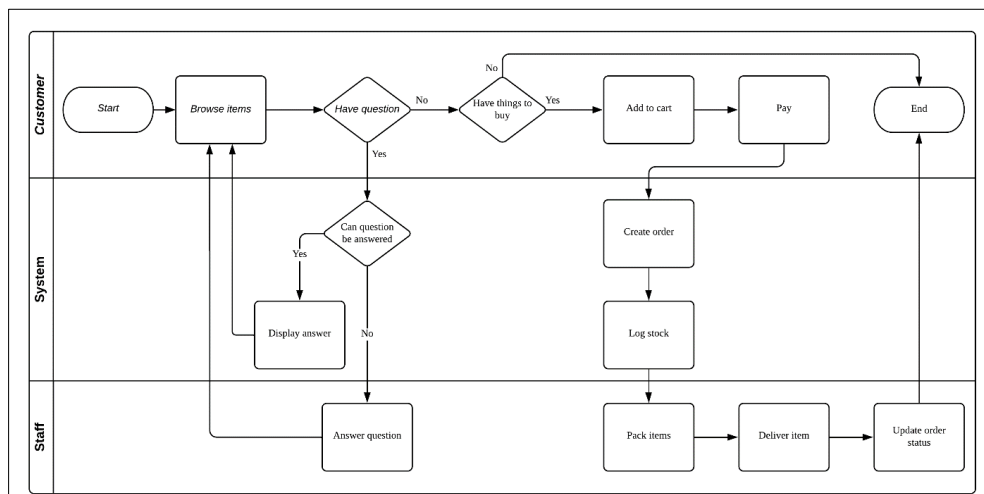


Figure 4: To-be Model

Relational schema refers to a group of objects which are related to each other in a database [21]. The relational schema for database tables is listed as follows.

- i. User (userId, password, email, name, isAdmin)
- ii. Category (categoryId, categoryName)
- iii. Product (productId, productName, productDescription, productPrice, productQuantity, categoryId)
- iv. Order (orderId, orderDate, orderTotal, orderStatus, userId)
- v. OrderDetails (orderId, productId, quantity)
- vi. Cart (cartId, userId)
- vii. CartItems (cartId, productId, quantity)
- viii. Post (postId, post_title, postContent, userId)
- ix. Comment (commentId, commentContent, userId)

Figure 5(a) and Figure 5(b) shows the registration and login page respectively. User will be required to enter all fields to register or log in to the system.

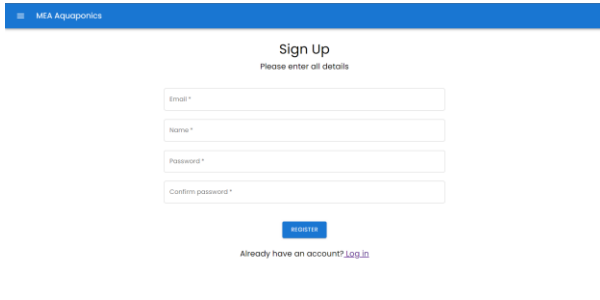


Figure 5(a): Registration page

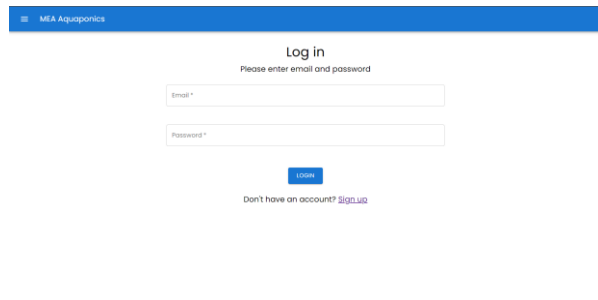


Figure 5(b): Login page

Figure 6 shows the manage product page. Administrator can add, update and delete products.

ID	Product name	Category	Price	Quantity
61c42ebca204d4e11469c091	Cucumber seeds	Seeds	9.90	14
61c431a9ed84596df9f9be917	Tomato seeds	Seeds	9.90	46
61c4b7cc88bebf9b3969456c	Goat Milton Longevity Fertilizer	Fertilizers	44.44	43
61c887e35798da8e0ed4bf1a	Super Max Growth Fertilizer	Fertilizers	32.33	33
61c1d44816a3dc7c46624705	Boku	Kits	150.36	7
61c1d5b16a3dc7c46624707	BioBlizz	Fertilizers	45.99	16
61c1d84b3b00be404f2ac85	Starter kit	Kits	69.69	15
62a1d27763482a08e333199d	Pump WP100-H	Fertilizers	25.99	15
62a5c08b50a27af266226e7d	Hydrofarm AAPW400	Water pump	99.99	56
62a5c2a0efbd38d233399467	Hydrofarm Active Aqua Pump 4	Water pump	599.00	20
62a61ba3f071a18ba95f55ef	Hydrofarm Claw Ball	Glow media	2.70	500

Figure 6: System interface for manage product page

Figure 7 shows the make payment page. Customer can view and remove products in cart or pay.

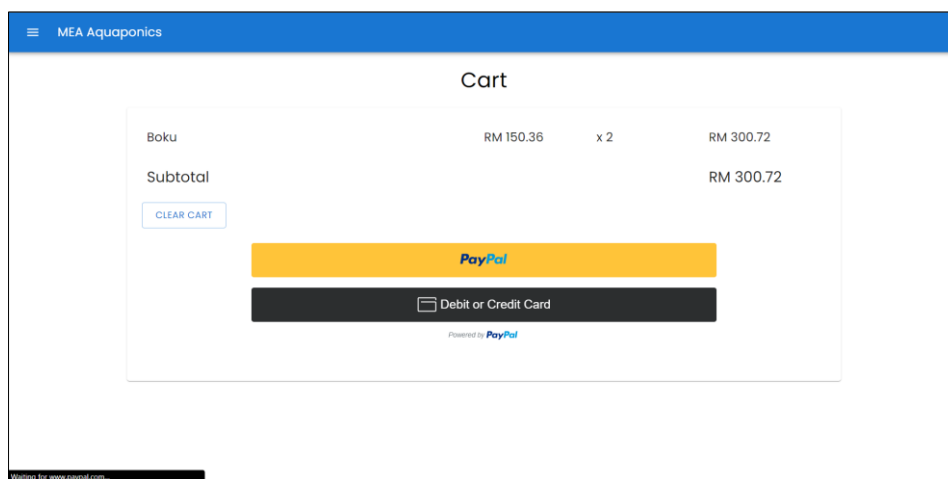


Figure 7: System interface for make payment page

Figure 8 shows the manage order page. Admin can view order details and change order status.

Order ID	Order date	Customer	Order status	Order total	Details
62a593af50a27af266226bad	12/06/2022, 15:20:15	Demo Account	Completed	19.80	
62a30ab22684fb38e6f42f4	10/06/2022, 17:31:14	customer	Processing	44.44	
62a308ea2684fb38e6f42a6	04/06/2022, 17:07:54	customer	Completed	159.18	
62a305702684fb38e6f4275	26/05/2022, 16:48:48	Demo Customer	Completed	9.90	
62a3048c2684fb38e6f4252	21/05/2022, 16:44:28	Demo Customer	Completed	28.70	
62a3032e75c2b225437ae84	14/05/2022, 16:39:10	Demo Customer	Completed	28.70	
62a301dbdfbd28a64fc41f	10/05/2022, 16:33:31	Demo Customer	Cancelled	9.90	
62a30106031cd8e514bcf52d	01/05/2022, 16:29:58	Demo Customer	Cancelled	19.80	
62a2fcf9031cd8e514bcf4b1	01/05/2022, 16:12:35	Demo Customer	Completed	146.49	
62a2ea555241a75f1be46a9	01/05/2022, 14:53:09	Demo Customer	Completed	133.93	
62a2a65431c43bfc23ab4b72	30/04/2022, 10:15:16	Demo Customer	Completed	137.97	

Figure 8: System interface for manage order page

Figure 9 displays the system interface for generate report page. Administrator can view daily sales, order status and low quantity items.

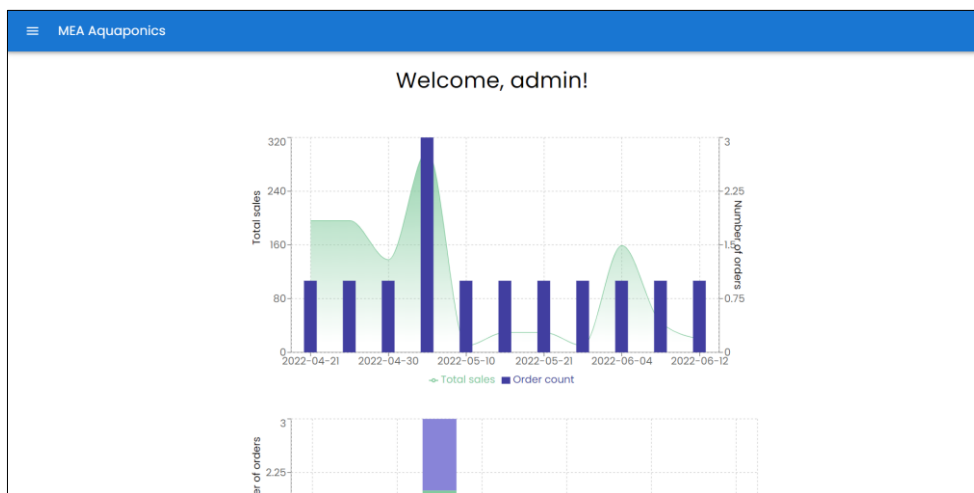


Figure 9: System interface for generate report page

Figure 10 shows the access forum interface. Users can view post and comment, create post, edit and delete post. They can also add, edit and delete comments.

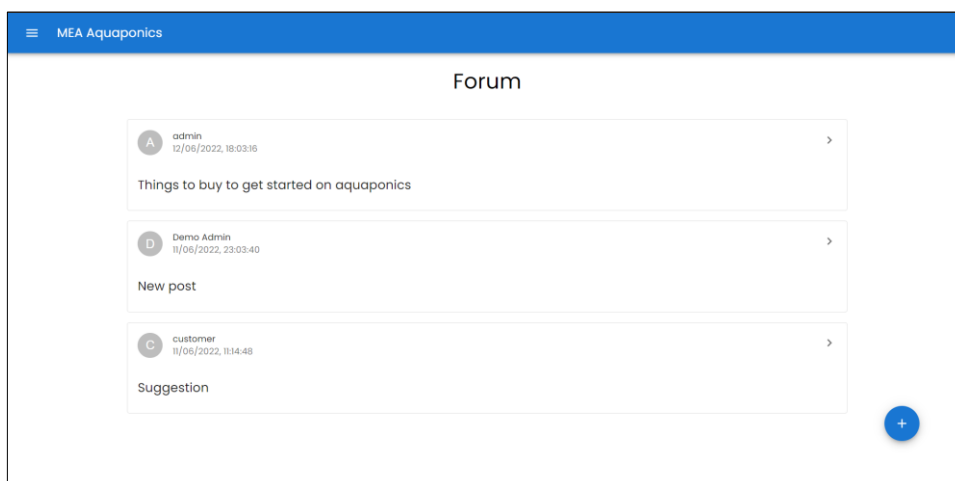


Figure 10: System interface for access forum page

4. Results and Discussion

Testing is done to ensure that the developed system successfully achieve the objective. The testing conducted are functional testing and user acceptance testing.

4.1 Functional testing

Functional testing is conducted to test the functional requirements of the system. The test is conducted on all 6 modules developed in this system. This is to ensure that the system is functioning as it should be. Table 6 shows the test case result of register and login module.

Table 6: Test case result of register and login module

Test Case ID	Description	Expected Output	Actual Output
TEST_100_001	User fills in all field during registration	Registration successful	As expected output
TEST_100_002	User register without filling in all field	Registration unsuccessful	As expected output
TEST_100_003	User did not enter more than 8 characters for password during registration	Registration unsuccessful	As expected output
TEST_100_004	Password inputs do not match during registration	Registration unsuccessful	As expected output
TEST_100_005	User enters valid email and password.	Login successful	As expected output
TEST_100_006	User enters invalid email or password.	Login unsuccessful	As expected output

Table 7 shows the test case result for manage product module. The admin can view, add, update and delete products.

Table 7: Test case result of manage product module

Test Case ID	Description	Expected Output	Actual Output
TEST_200_001	Administrator visits product page.	Products list displayed	As expected output
TEST_200_002	Administrator adds product by filling in all field	New product added to list	As expected output
TEST_200_003	Administrator adds product but did not fill in all field	Error message shown	As expected output
TEST_200_004	Administrator updates product details by changing value in table	Product information updated	As expected output
TEST_200_005	Administrator deletes product by clicking of icon	Product removed from list	As expected output

Table 8 shows the test case result for make payment module. The customer can view products, add and delete cart items and make payment.

Table 8: Test case result of make payment module

Test Case ID	Description	Expected Output	Actual Output
TEST_300_001	Customers visits product page	Products information displayed	As expected output
TEST_300_002	Customer adds product to cart by specifying valid number	Products added to cart	As expected output

Test Case ID	Description	Expected Output	Actual Output
TEST_300_003	Customer adds product to cart by specifying invalid number	Error message shown	As expected output
TEST_300_004	Customer deletes product from cart	Products removed from cart	As expected output
TEST_300_005	Customer clicks on payment option	Customer redirected to payment platform	As expected output
TEST_300_006	Customer completes transaction	Order created and products removed from cart	As expected output
TEST_300_007	Customer did not complete transaction	Error message shown	As expected output

Table 9 shows the test case result for manage order module. The admin can view order details and update order status. The customer can view order details, status and history.

Table 9: Test case result of manage order module

Test Case ID	Description	Expected Output	Actual Output
TEST_400_001	Customers visits order page	Order information displayed	As expected output
TEST_400_002	Administrator visits order page.	Order list displayed.	As expected output
TEST_400_003	Administrator updates order status	Order status updated	As expected output

Table 10 shows the test case result for generate report module. The admin can view reports generate such as daily sales, order status and low stock item list.

Table 10: Test case result of generate report module

Test Case ID	Description	Expected Output	Actual Output
TEST_500_001	User visits dashboard	List of low stock items, charts of earnings and order status shown	As expected output

Table 11 shows the test case result access forum module. User can view and add post or comments. Post or comment owners can update or delete posts or comments. Admin can view, add, update and delete all posts and comments.

Table 11: Test case result of access forum module

Test Case ID	Description	Expected Output	Actual Output
TEST_600_001	User visits forum page	List of post displayed	As expected output
TEST_600_002	User adds post or comment by filling in all fields	New post or comment added and visible to everyone	As expected output
TEST_600_003	User adds post or comment but did not fill in all fields	Error message shown	As expected output
TEST_600_004	User updates post or comment by filling in all fields	Post or comment updated with new content and visible to everyone	As expected output
TEST_600_005	User updates post or comment but did not fill in all fields	Error message shown	As expected output
TEST_600_006	User deletes post or comments	Post deleted.	As expected output

4.2 User acceptance testing

User acceptance testing is conducted to test the acceptance of the system by user. The respondent involved are 2 admin and 3 customers. Figure 11 shows the result of user acceptance testing on the system interface. Majority respondents think that the system interface is good.

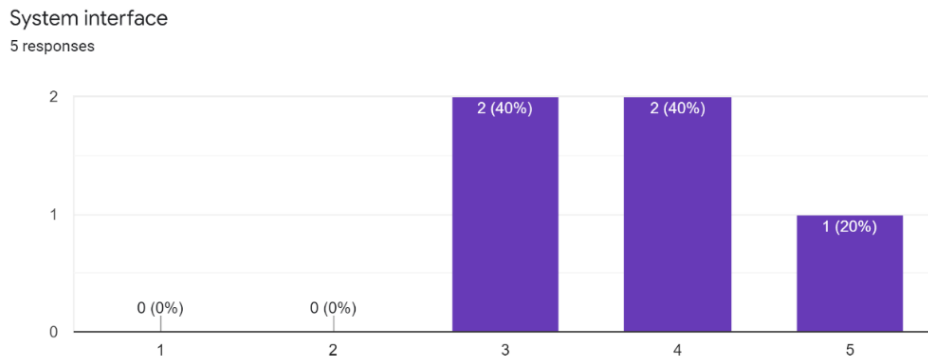


Figure 11: Result of user acceptance testing on the system interface

Figure 12 shows the result of user acceptance testing on the system functionality. Majority respondents rated that the system functionality is good.

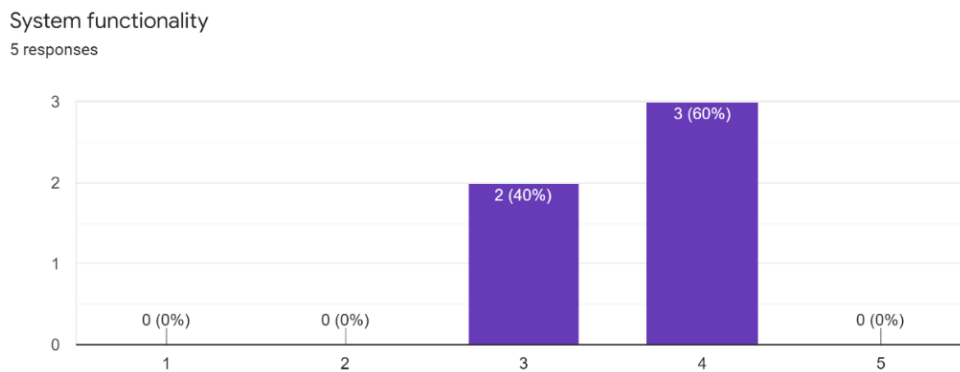


Figure 12: Result of user acceptance testing on the system functionality

Figure 13 shows the result of user acceptance testing on the ease of use. Majority respondents think that the system is easy to use.

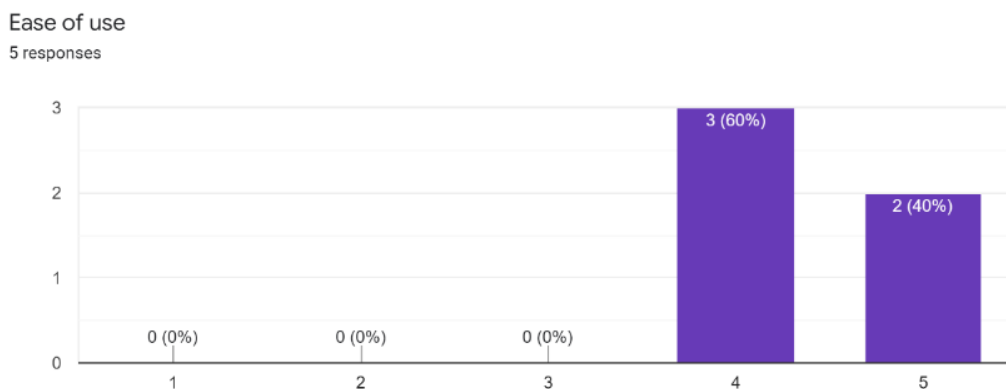


Figure 13: Result of user acceptance testing on the ease of use

5. Conclusion

In conclusion, the aquaponic e-commerce system developed successfully accomplish the requirements. With this system, the business reach can be improved. Besides that, accessibility and availability of aquaponic accessories can increase and perception and knowledge about aquaponic is able to increase. This will encourage more people to get involved in aquaponic practice and increase the sustainability of food to the world. The staff also has less workload by using this system in their operation.

With this system, customers can purchase aquaponic equipment from anywhere. The products will be delivered to the customer after the order is placed. Besides that, customer can communicate and interact with people of the similar interest in aquaponics. Customers can also learn more about aquaponics through the forum. The staff can also use the system to automatically track the quantity of stocks left. Reports about sales and order can also be used by staff to make better decision.

Although all the functionalities of the system are developed, there are limitations to this system. The first limitation is that the system only accept payment from PayPal, credit or debit card. More common payment options in Malaysia like online banking and e-wallet are not accepted by this system. The second limitation is the number of images for the products page. There is only 1 image for each products which makes it hard for customer to visualize the product.

Based on the limitations of the system, several suggestions were made to improve the system. The first suggestion is to integrate more payment options. This will make it more convenient for more customers. The second suggestion is to add more images to product page. This suggestion can make it more attractive to customers to purchase. These suggestions will help make the system better.

Acknowledgment

The authors would like to thank the Faculty of Computer Science and Information Technology, Universiti Tun Hussein Onn Malaysia for its support.

References

- [1] J. Maurice, UN set to change the world with new development goals, *The Lancet*, 386(9999), 1121-1124, 2015.
- [2] C. B. d'Amour, F. Reitsma, G. Baiocchi, S. Barthel, B. Güneralp, K. H. Erb, H. Haberl, F. Creutzig, K. C. Seto, Future urban land expansion and implications for global croplands, *Proceedings of the National Academy of Sciences*, 114(34), 8939-8944, 2017.
- [3] F. Orsini, R. Kahane, R. Nono-Womdim, G. Gianquinto, Urban agriculture in the developing world: a review, *Agronomy for sustainable development*, 33(4), 695-720, 2013.
- [4] R. V. Tyson, D. D. Treadwell, E. H. Simonne, Opportunities and Challenges to Sustainability in Aquaponic Systems, *HortTechnology*, 21(1), 6-13, 2011.
- [5] G. Short, C. Yue, N. Anderson, C. Russell, N. Phelps, Consumer perceptions of aquaponic systems, *HortTechnology*, 27(3), 358-366, 2017.
- [6] J. O. Grady, *System requirements analysis*, Elsevier, 2010.
- [7] M. Glinz, On non-functional requirements, In 15th IEEE international requirements engineering conference (RE 2007) (pp. 21-26), October 2007.

- [8] S. Alsaleh, H. Haron, The most important functional and non-functional requirements of knowledge sharing system at public academic institutions: a case study, Lecture notes on software engineering, 4(2), 157, 2016.
- [9] D. Savic, A.E. da Silva, S. Vlajic, S. Lazarevic, V. Stanojevic, I. Antovic, M. Milic, Use case specification at different levels of abstraction, In 2012 Eighth International Conference on the Quality of Information and Communications Technology (pp. 187-192), September 2012.
- [10] R. Stephens, R. Plew, Database design, Sams Publishing, 2000.