SMART AID FOR SHOPPER

NURHIDAYAH BINTI MOHD LAZIM

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

SMART AID FOR SHOPPER

NURHIDAYAH BINTI MOHD LAZIM

This report is submitted in partial fulfillment of the requirements for the Bachelor of [Computer Science (Software Development)] with Honours.

FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2021

DECLARATION

I hereby declare that this project report entitled

[SMART AID FOR SHOPPER]

is written by me and is my own effort and that no part has been plagiarized without citations.

Hidayah

STUDENT : ______________________________ Date : 05/09/2021
(NURHIDAYAH BINTI MOHD LAZIM)

I hereby declare that I have read this project report and found this project report is sufficient in term of the scope and quality for the award of Bachelor of [Computer Science (Software Development)] with Honours.

UmmiRaba'ah

SUPERVISOR : ______________________________ Date : 05/09/2021
(TS. DR. UMMI RABAAH BINTI HASHIM)

DEDICATION

In the name of Allah, the Almighty and the most Merciful

Firstly, I would like to express my heartfelt gratitude to my beloved parents, who have always been by my side and have been my backbone from the time I was a child till now, for the endless support and encouragement that both of you have given to me. Last but not least, there is nothing I can do to compensate all of your efforts and sacrifices. But, I hope that the graduation graduate scroll in my hands may make you proud and elevate your dignity as a parents who have tried your best to give the finest towards me.

ACKNOWLEDGEMENTS

My deepest gratitude goes to my supervisor, Dr. Ummi Rabaah for her guidance, advice and constructive comments on project development process, requirement specification and so forth. She also provides a lot of feedback to me during the report writing which increase the overall quality of this project. I would also like to take this opportunity to say thank you to all my friends who has always been giving me support in anything I’m focusing on and for helping me out in the system design for helping me coding and solve the bugs happened in my system. Their helps are truly appreciated. Lastly and definitely not easily forgotten my sincere gratitude and appreciation go to my beloved parents who have been giving me support and motivation throughout my project. I could not have completed these things effectively without the help of everyone. As everyone is aware, being a degree-seeking student is not an easy process. However, with the presence of a loved one, it could give me the strength and enthusiasm to continue this study successfully.

ABSTRACT

Smart Aid for Shopper System (SAFSS) is an online system designed to help shoppers identify products location more quickly and correctly in a supermarket. Furthermore, it makes it easy for shoppers to evaluate the expense of a need more sensibly. In addition, compared to the manual method, SAFSS advances the critical products management system in a more ordered, efficient, and systematic manner. This system has two users: administrator and shopper. Consumers can avoid wasting a significant amount of time at the store while also contributing to the congestion of many other shoppers. In addition, the system is responsive which it can fit in any devices.

ABSTRAK

Smart Aid for Shopper System (SAFSS) adalah sistem dalam talian yang direka untuk membantu pembeli mengenal pasti lokasi produk dengan lebih cepat dan betul di pasar raya. Selain itu, memudahkan pembeli menilai perbelanjaan keperluan dengan lebih bijak. Di samping itu, dibandingkan dengan kaedah manual, SAFSS memajukan sistem pengurusan produk kritikal dengan cara yang lebih teratur, cekap, dan sistematik. Sistem ini mempunyai dua pengguna: pentadbir dan pembeli. Pengguna dapat mengelakkan membuang banyak masa di kedai sambil menyumbang kepada kesesakan banyak pembeli lain. Di samping itu, sistem ini responsif yang dapat dimuat di mana-mana peranti.

TABLE OF CONTENTS

PAGE

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

FYP	**-**	**Final Year Project**
SAFSS	**-**	**Smart Aid for Shopper System**
SCM	**-**	**Software Configuration Management**
UML	**-**	**Unified Modeling Language**

Chapter 1: INTRODUCTION

1.1 Introduction

Shopping for our daily needs is an essential part of life. Shopping service are one of important things in a store that offering advice or assistance with shopping. Therefore, shopping services becomes one of the crucial part for shopper. Unfortunately, it is an aspect that many shoppers struggle with, and ultimately end up doing poorly in finding the exact location for a certain product item. The purpose of developing the Smart Aid for Shopper system is to ease shopper in their shopping. Which is, this system providing the exact location of the product item to a shopper. It is a straightforward and user-friendly system that users can access via any devices.

1.2 Problem Statements

Problem statements that has been identified are:

- Shopper have problem in finding to detect certain product's location.
- Shoppers have a problem with inadvertently overspending in the supermarket.

1.3 Objectives

The objectives of this project system are:

- To assist shopper, save time and energy by providing shopper to know the exact location of the product item.
- To assist shopper in budgeting their goods from overspend by providing calculation that will calculate total spending.
- To test developed web-based application.

1.4 Scope

There are totally six modules are being built for this system:

1) Admin Management

    - Create, update, add, delete and display the details of product.
    - Register for new selected admin.

2) Products Management

    - Update the products details.

3) Authentication

    - Shopper registration to log into the system.
    - Shopper log into the system.
    - Admin log in to the system.

4) Search product location

    - Search product location to ease shoppers

5) Shop List

    - To add product to a temporary shop list for easy budgeting.
    - Adding product can be made by product search.

6) Graph Analysis

    - To assist admin, by viewing the monthly product report in a selected category.

There are two target users:

1) Admin

- Able to control overall of the system. Also can create, add, update, and delete products data provided in the system.

2) Shopper

- Able to search product, check product price and calculate total products that have been added into the list.

1.5 Project Significance

This project application is developed to gives a lot benefits to the shopper due to it ease the shopper in shopping things. Where it provides for shoppers to know the exact location of the product, saving time and helps shopper from overspend.

1.6 Expected Output

Smart Aid for Shopper System (SAFSS) will be developed that shopper could locate the exact location of the product item. Also, it allows shopper to add goods in shop list to calculate total budgeting. Admin can add, update and delete product items.

1.7 Conclusions

In conclusion, this chapter explained briefly about the project system, the scope of this application including project objective, problem statement and the expected output.

CHAPTER 2: LITERATURE REVIEW AND PROJECT METHODOLOGY

2.1 Introduction

Literature reviews were done based on the surveyed of the previous scenario. The review covered on concept and theory which it will be used in generating Smart Aid for Shopper System(SAFSS). Literature review is showing the research that related to the project topic. The meaning of literature review is a critical look of existing research that significant to the works that are carried out. It is one of the crucial part because it shows the project's target.

This chapter contains all the research that has been done on the previous scenario. All weaknesses on the current scenario were identified in order for this project to overcome and the strength of the existing system are identified and studied so that it can be implemented in the system.

The methodology consists of several phases as guidelines that are to be achieved. The project requirements such as software and hardware, for the project development are determined. The project milestone from the start until the delivery phase are also will be brief and list in this chapter. The milestone and Gant chart is important as a guideline to ensure the project can be finishing according to the schedule and plan.

2.2 Fact and Findings

Basically, fact and finding is a discovery of fact or accurate information. Generally, in terms of fact and findings, there are no further studies or articles related that can be an example to follow when built this system. In this section, it conveys about the fact scenario previously, to find out the strength and weaknesses of the system. There are some drawbacks that can be found by using the questionnaire method with several individuals.

2.2.1 Domain

The Smart Aid for the Shopper System will be used at any supermarket available. The current scenario easily leads to the problem. The SAFSS is one of crucial things in shopping lifestyle due to it assist shopper to find an exact way to the product item.

As we can see, Smart Aid for Shopper is a simple and convenient system that ease the shoppers. The aim of this system to be develop for shopper to help in finding their goods location exactly. Also, allows shopper to budget their goods from overspend in the supermarket. Besides, to help shopper in comparing the price of a certain product with another product. Other than that, the system assist admin to manage their product with efficiently and properly. The development of the system is to expect improvement on the shopping and the product services.

2.2.2 Existing System

In fact, there are no such system like this before has been presented so far. The majority of them are more concerned with the sale of merchandise. However, there are no features available to assist shoppers in finding where the product is placed more accurately and can save shoppers time.

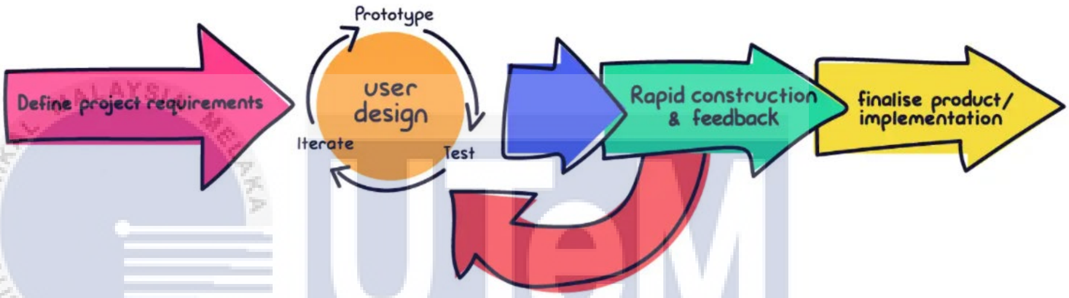
Looking at the results from questionnaire and previous scenario when shopping, shoppers manually search for product items by following the signboards provided. However, they were unable to locate certain product item accurately. So, they need to ask nearby employees to facilitate and expedite their movement. Therefore, using this system can overcome such problems.

Developing this system will avoid shoppers from wasting time in the supermarket for hours. For example, finding a goods in a supermarket takes time compared to a consumer who already knows where it is. Also, with detailed info about the product such as the price of the product can help the shopper to compare prices between products. Thus, the shopper can determine the items that need to be purchased more quickly. By using this system, it can help shoppers more effectively while reducing congestion caused by a large number of shoppers in supermarkets. Lastly, the system allows the administrator to handle all product items in a more organized and seamless manner by categorizing the product item in a selected category.

2.2.3 Technique

There are several approaches can be uses in this project. The first approach is observation. The observation is the primary techniques where we observe the scenario before developing the system. In addition, questionnaire is one of techniques that we used for information gathering during system analysis phase of a system development.

2.3 Project Methodology


Define project requirements
Prototype
user design
Iterate
Test
Rapid construction & feedback
finalise product/ implementation

Generally, Rapid Application Development methodology are used in this project. A rapid application development approach has a plethora of benefits for both software developer and clients. There is an increase in productivity when speed and agility are prioritized, which improves project results and allows for delivery in a matter of days or weeks. Other than that, RAD is the ability to change requirements at any point in the development cycle. It consists of four phase which are define project requirements, where defining a loose set of project requirements, equivalent to what would be accomplished during project scoping in traditional development cycles. This planning stage is brief-emphasizing a higher priority on prototype iterations-but critical to the ultimate success of a project.

Second, is prototype. After a project has been scoped, start to develop the first models and prototypes. The objective is to create a functioning design that can be shown to the client as soon as possible. To guarantee that the client's demands are satisfied, developers collaborate with them until a final product is ready. This phase is frequently repeated as the project progresses. During the early stages of prototyping, it is typical for developers to cut corners in order to deliver a functional product that is acceptable to the product owner. Through prototyping, the development team can easily evaluate the feasibility of complex components. As a result, software is more resilient, less prone to errors, and more well-structured for future design enhancements.

Third, is rapid construction where application coding, system testing, and unit integration occurs, converting prototype and beta systems into a working model. This phase may be repeated as needed to accommodate additional components and changes. To progress the programmed fast, teams typically utilize low-code or rapid application development technologies.

Lastly is finalize product or implementation where developers handle the technical debt accumulated during early prototyping, optimizing implementation to enhance stability and maintainability as they finalize the product for launch. During the implementation phase, development teams deploy components to a live production environment, where full-scale testing and training may take place. Before confidently turning over a finished product to a shopper, teams produce detailed documentation and execute other essential maintenance activities.

2.4 Project Requirements

2.4.1 Software Requirement

- Sublime Text 3 Editor
- MySQL phpMyAdmin
- XAMPP
- Star UML
- Microsoft Word

2.4.2 Hardware Requirement

- Asus TUF Gaming
- TM Unifi router

2.4.3 Other Requirements

There are no other requirements a long in developing the project system.

2.5 Project Schedule and Milestones

A Gantt chart is a project management tool that helps with the planning and scheduling of projects of all kinds, but they are especially effective for simplifying complex tasks. The Gantt chart for this project is shown in the table below.

2.5.1 Gantt Chart

Table 2.1: Gantt Chart

Task / Week	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1. Planning • Proposal preparation • Define problem statement, objectives and scope • Discuss current system. • Collect information from current system	■	■														

2. Design - Design an interface - Design database - Design application - Create design specification			▨	▨	▨	▨										
3. Implementation - Integrate coding - Combine all modules - Run coding - Perform an initial testing							▨	▨	▨	▨	▨	▨	▨			
4. Testing - Perform system testing - Document issues found - Fix bugs and error found								▨		▨		▨	▨	▨	▨	▨

2.6 Conclusion

In conclusion, Rapid Application Development has effectively achieved the goal of lowering project costs while maintaining quality. Next, Rapid Application Development has shown its ability to accelerate the development process by combining its approach, people, management, and high-tech computer assisted tools in an appropriate manner. The following chapter will go into further detail regarding the examination of the developed Smart Aid for Shopper System.

CHAPTER 3: ANALYSIS

3.1 Introduction

The process of identifying user expectations for new or updated software is known as requirement analysis. In the domain of software engineering, it is also known as requirements gathering or requirements capture. Requirements analysis refers to the activities involved in determining the demands or criteria that must be met for a new or changed system. The requirements included the functional and non-functional requirements which are essential for the development of the system.

3.2 Problem Analysis

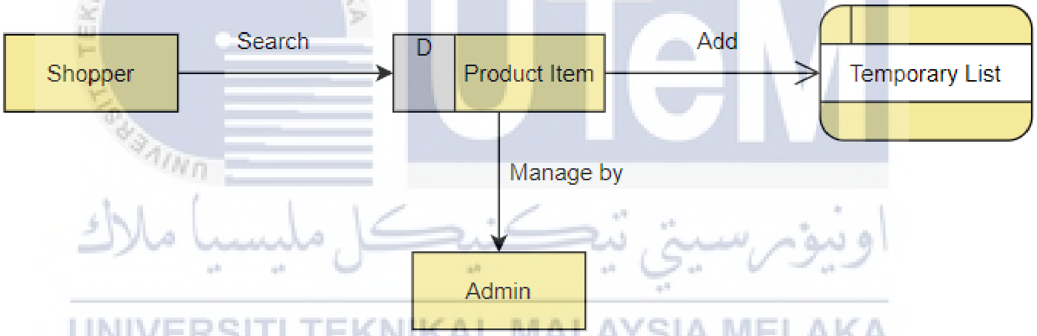


Figure 3.1 Data Flow Diagram for SAFSS

Generally, Shopper want to find product item’s location or details about the product. At the same time, they can add product item into the temporary list if they wish. The system will calculate the total spending for all selected product item that have been added by Shopper itself. Product item are being manage by Admin.

3.2.1 Analysis of Current Scenario

The aim of analyzing the current scenario is to discover the shortcomings and issues with the current scenario. By listing out and enhance the system by recognizing the deficiencies of the current scenario.

Currently, based on observation and questionnaire with several individuals. Mostly, shopper will take more time in supermarket due to they need to find the exact location of their selected goods and waste more time by comparing the price of one product with another product. As a shopper, they manually searched for goods according to the signage provided. However, if they were unable to locate the goods accurately and had to ask nearby employees to facilitate and expedite the relocation.

3.2.2 Analysis of Proposed System

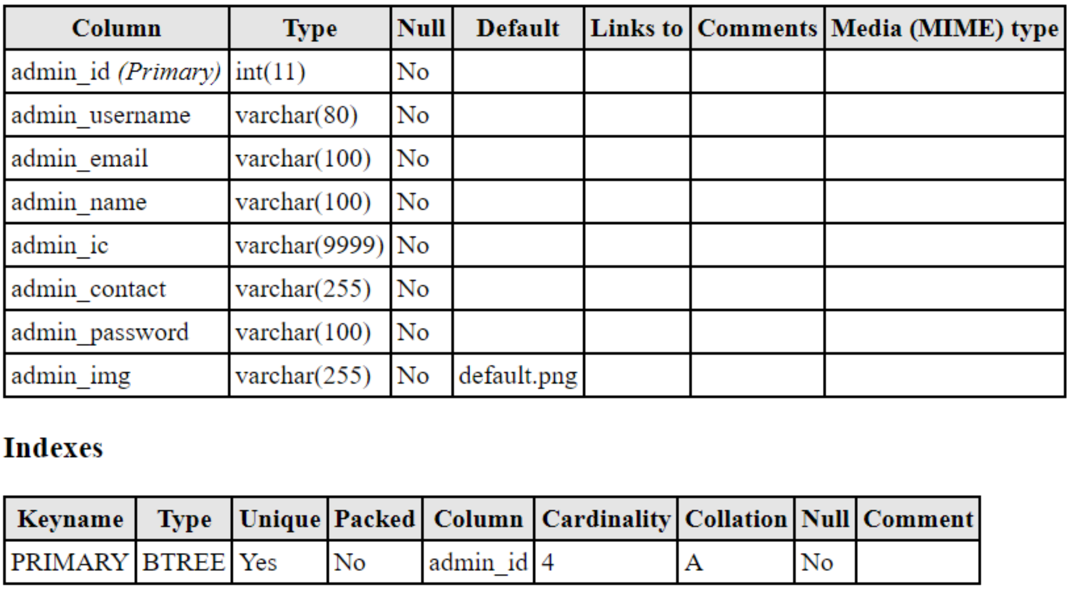
Smart Aid for Shopper System (SAFSS) is created especially for shopper. By using SAFSS, it helps to improve the shopping services by display all the product list and category. The interesting about this system is, it shows the details of the product's place and the product's price. Also, it allows shopper to estimate the budget to be spent in a supermarket to avoid overspending. Aside from that, administrators may regulate products in a more ordered and systematic manner by employing this approach.

3.3 Requirement analysis

3.3.1 Data Requirement (Data Dictionary)

Data dictionary used to record all data and table information in database. Data requirement is a requirement that specifies a compulsory sort of information. Figure 3.2 to Figure 3.7 shows data dictionary of this system.

3.3.1.1 Admin



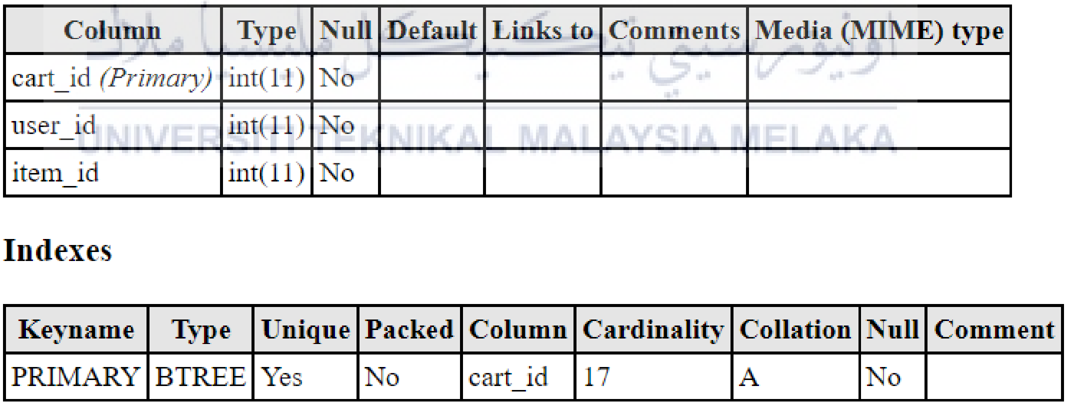
Column	Type	Null	Default	Links to	Comments	Media (MIME) type
admin_id *(Primary)*	int(11)	No				
admin_username	varchar(80)	No				
admin_email	varchar(100)	No				
admin_name	varchar(100)	No				
admin_ic	varchar(9999)	No				
admin_contact	varchar(255)	No				
admin_password	varchar(100)	No				
admin_img	varchar(255)	No	default.png			

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	admin_id	4	A	No	

Figure 3.2 Data Dictionary of Admin

3.3.1.2 Cart



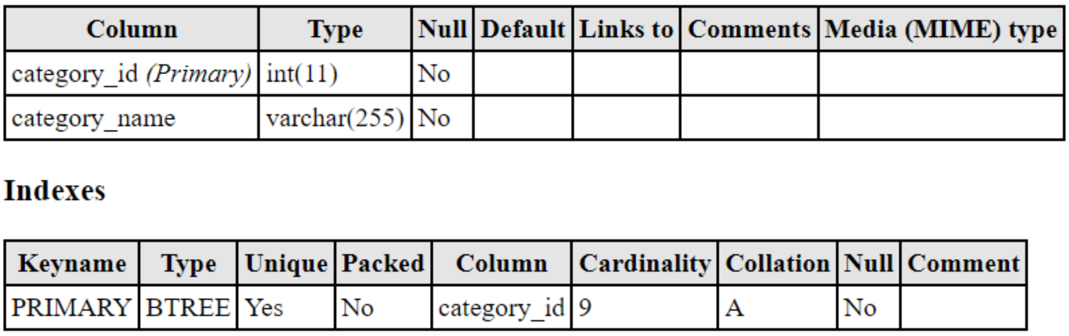
Column	Type	Null	Default	Links to	Comments	Media (MIME) type
cart_id *(Primary)*	int(11)	No				
user_id	int(11)	No				
item_id	int(11)	No				

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	cart_id	17	A	No	

Figure 3.3 Data Dictionary of Cart

3.3.1.3 Category



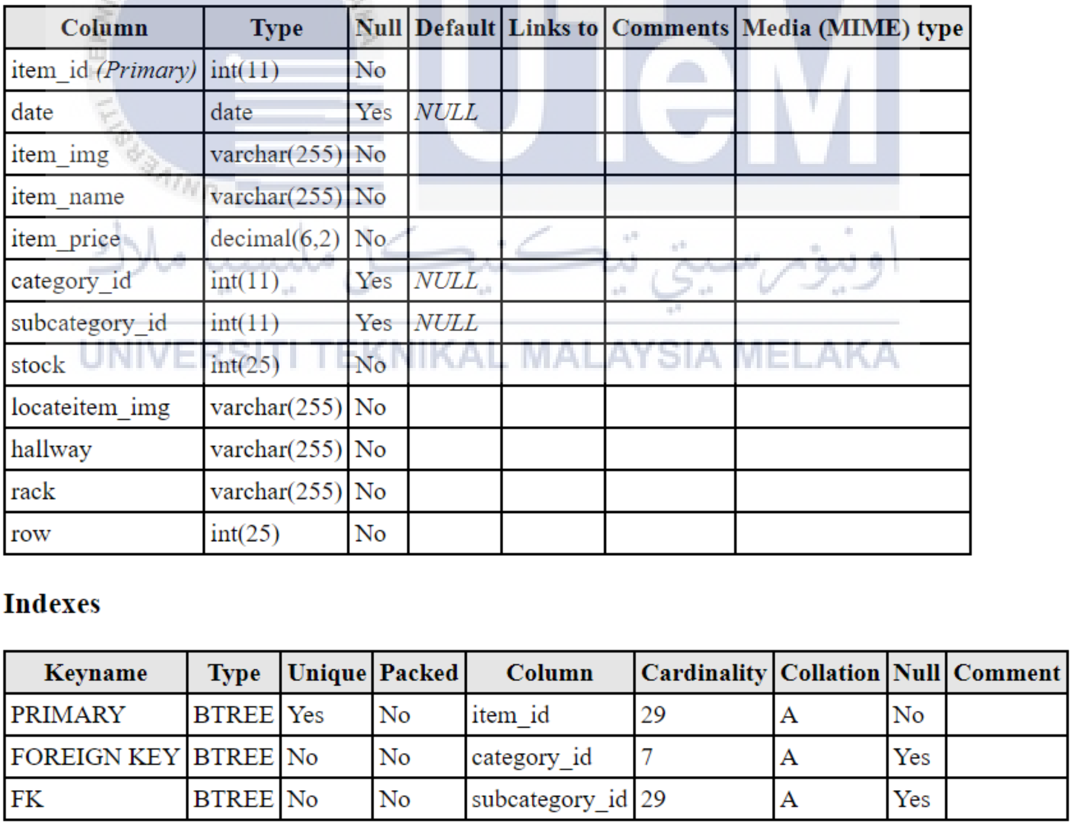
Column	Type	Null	Default	Links to	Comments	Media (MIME) type
category_id *(Primary)*	int(11)	No				
category_name	varchar(255)	No				

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	category_id	9	A	No	

Figure 3.4 Data Dictionary of Category

3.3.1.4 Item



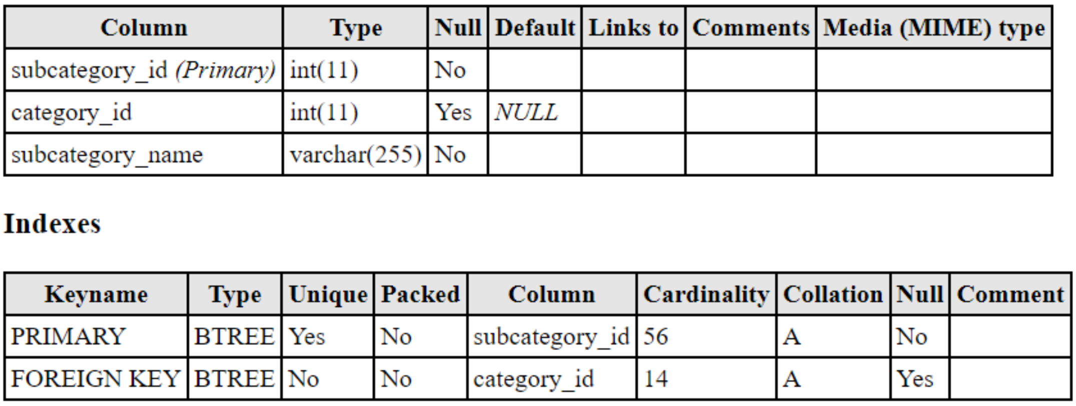
Column	Type	Null	Default	Links to	Comments	Media (MIME) type
item_id *(Primary)*	int(11)	No				
date	date	Yes	*NULL*			
item_img	varchar(255)	No				
item_name	varchar(255)	No				
item_price	decimal(6,2)	No				
category_id	int(11)	Yes	*NULL*			
subcategory_id	int(11)	Yes	*NULL*			
stock	int(25)	No				
locateitem_img	varchar(255)	No				
hallway	varchar(255)	No				
rack	varchar(255)	No				
row	int(25)	No				

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	item_id	29	A	No	
FOREIGN KEY	BTREE	No	No	category_id	7	A	Yes	
FK	BTREE	No	No	subcategory_id	29	A	Yes	

Figure 3.5 Data Dictionary of Item

3.3.1.5 Subcategory



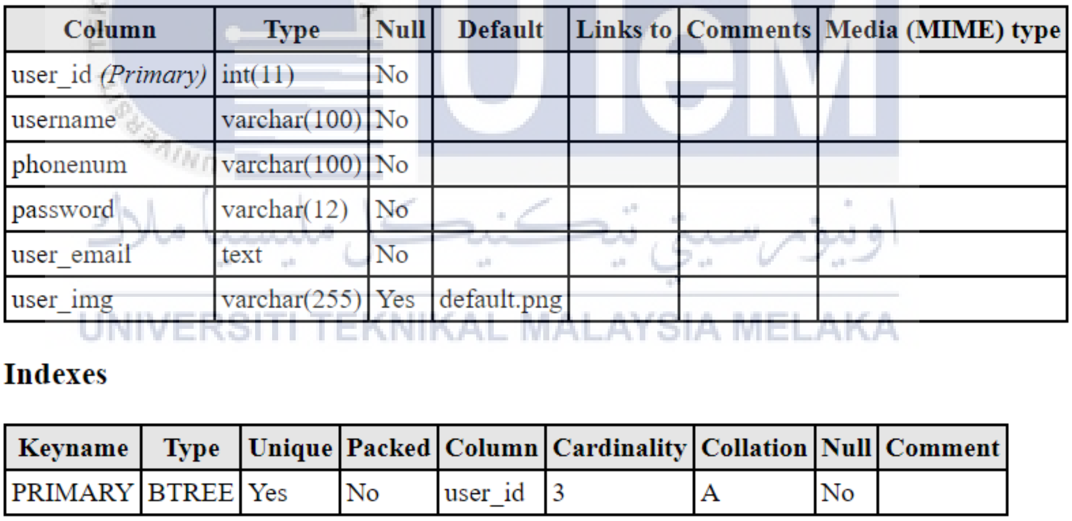
Column	Type	Null	Default	Links to	Comments	Media (MIME) type
subcategory_id *(Primary)*	int(11)	No				
category_id	int(11)	Yes	*NULL*			
subcategory_name	varchar(255)	No				

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	subcategory_id	56	A	No	
FOREIGN KEY	BTREE	No	No	category_id	14	A	Yes	

Figure 3.6 Data Dictionary of Subcategory

3.3.1.6 User



Column	Type	Null	Default	Links to	Comments	Media (MIME) type
user_id *(Primary)*	int(11)	No				
username	varchar(100)	No				
phonenum	varchar(100)	No				
password	varchar(12)	No				
user_email	text	No				
user_img	varchar(255)	Yes	default.png			

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	user_id	3	A	No	

Figure 3.7 Data Dictionary of User

3.3.2 Functional Requirement

The functional requirement describes the service that the programme must provide. It specifies how the system should respond when the user interacts with it. The Smart Aid for Shopper System functional criteria are as follows:

Table 3.1 Functional Requirement

Functional Requirement	Requirement	FR No.	Description
Authentication	Register	FR_01	The system should allow users to register into the system.
		FR_02	The system should validate format using the format given
		FR_03	The system should remind users not to leave any empty field
		FR_04	The system should notify user if they have successfully or unsuccessfully registered
	Login	FR_05	The system should allow user to access their own account
		FR_06	The system should verify the email entered according to the format
		FR_07	The system should check for correct password
		FR_08	The system should notify user if they have login into the system
Authorization	Shopper	FR_09	Shopper should be able to view their profile
		FR_10	Shopper should be able to update their profile
	Admin	FR_11	Admin should be able to view their profile
		FR_12	Admin should be able to update their profile
		FR_13	Admin should be able to view the product management
Manage	Shopper	FR_14	Shopper should be able to view the products
		FR_15	Shopper should be able to add product into the temporary list
		FR_16	Shopper should be able to search products item
	Admin	FR_17	Admin should be able to add others admin and products
		FR_18	Admin should be able to update others admin and products

		FR_19	Admin should be able to analyze monthly product report

3.3.3 Non-functional Requirement

Non-functional requirements specify the quality characteristics of a software system. Scalability, usability, security, portability, and other characteristics were among those sought. The following are the system's non-functional requirements:

Table 3.2 Non-functional Requirement

Aspect	Requirements
Availability	• The system will be able to operate normally whenever the users want to use.
Scalability	• The system should be able to expand the system if necessary
Portability	• The system should be able to run on various operating system, which are MacOS, Windows and Android
Data Integrity	• Data shall 100% consistent at all site and at all time.
Usability	• The system should be friendly for user to navigate and use. • The error message will pop out on the screen to alert the user after making

3.3.4 Others Requirement

For software requirements, XAMPP is needed as a cross-platform web server for Apache server, MySQL, PHP. MySQL is needed as the database management that will store all the data. For PHP programming language, Sublime Text 3 which is the text editor is used.

3.4 Conclusion

In conclusion, at this chapter we discussed about the problem analysis with current scenario and the proposed system. Functional requirement, and non-functional requirement also is explained in this chapter. Nest chapter, will discussed design phase.

CHAPTER 4: DESIGN

4.1 Introduction

This chapter will concentrate on the system's design. The system design will be created depending on the user’s requirement. It is one of vital part because the system design will serve as the guide for the system developer during the building phase, which will include coding and testing operations. Design is concerned with developing the system's human interface. It also entails organizing and modifying the database structure.

4.2 High-Level Design

High level software design provide us the view of system at abstract level. It shows the major components of the system which provides the information about the different components of software due to every system is made of different components we called them sub-system. Also, this diagram focus on what are the different components involve in the system.

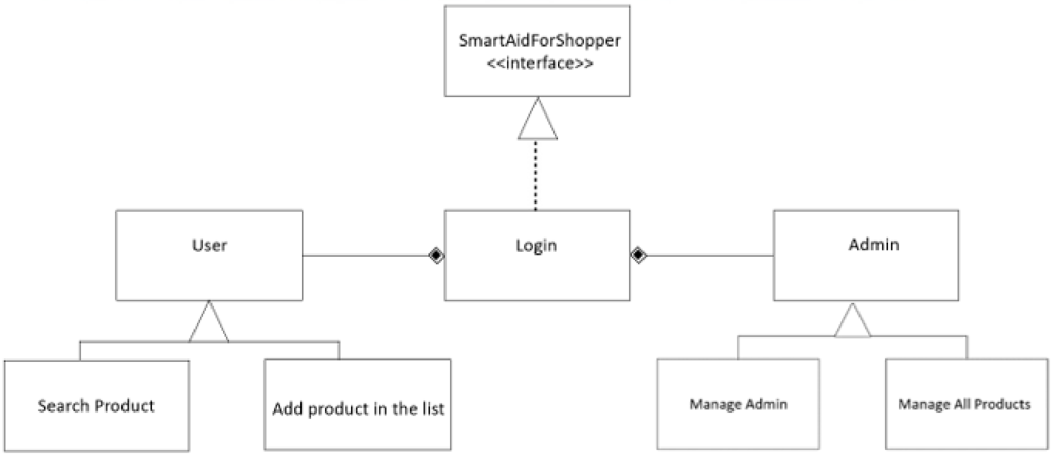


Figure 4.1 High-Level Design for SAFSS

4.2.1 System Architecture

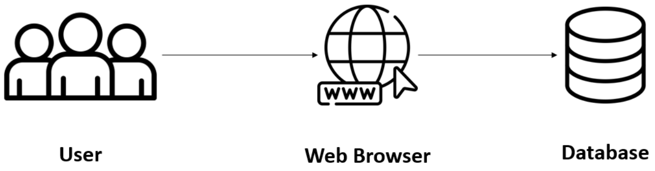
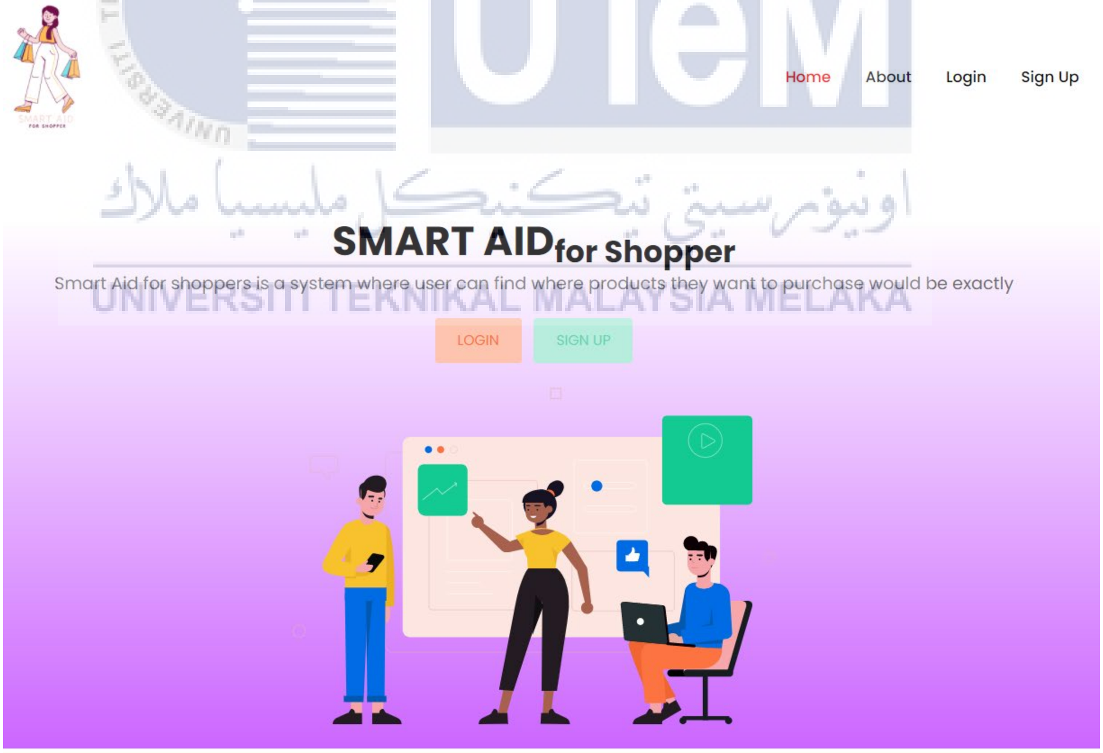


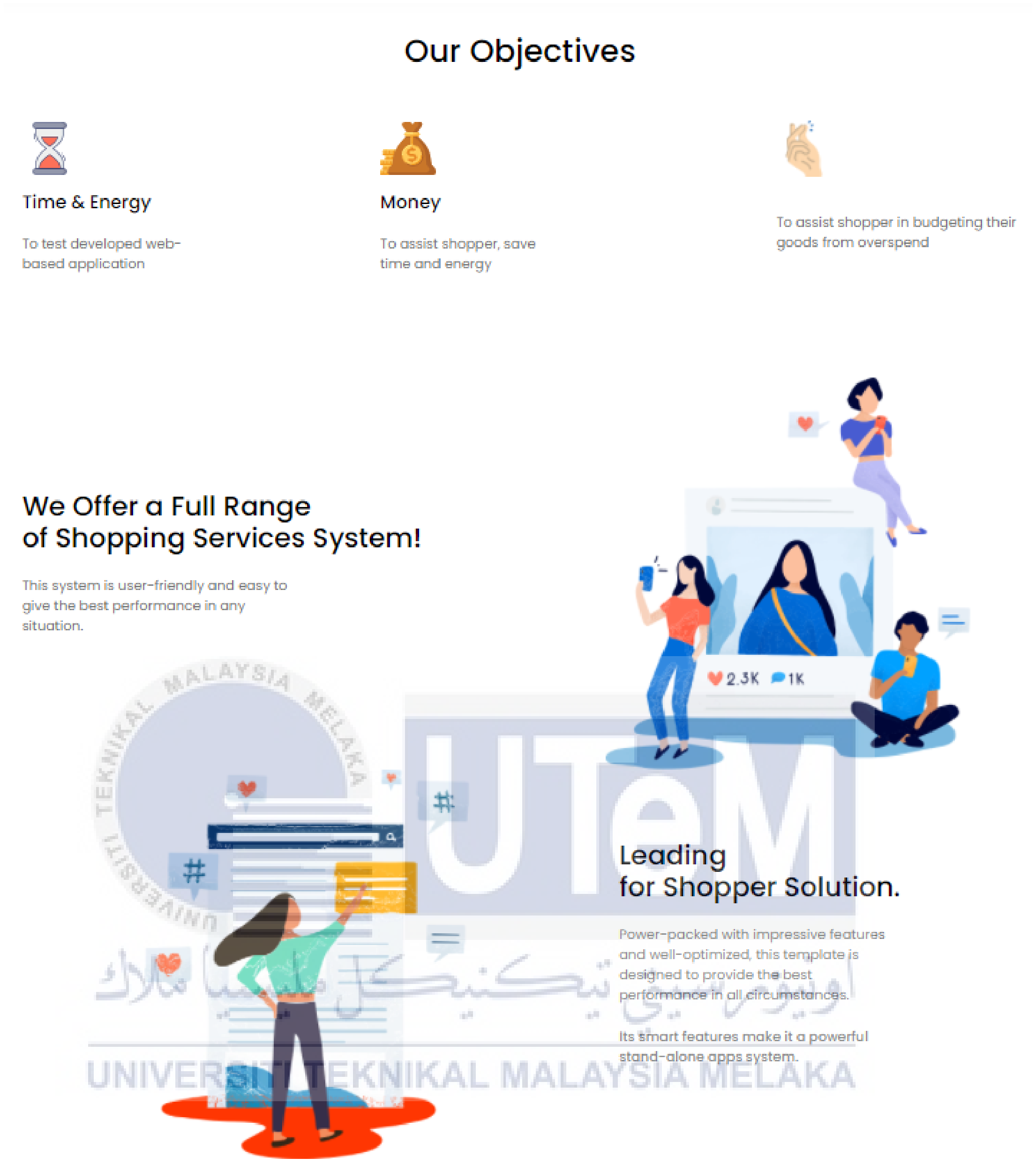
Figure 4.2: System Architecture of SAFSS

A web browser will be used to access this system. The activities that occur when a user logs into the system are saved in the database and can be used for the following session if required.

4.2.2 User Interface Design

a) Navigation Design


Home
About
Login
Sign Up
SMART AID for Shopper
Smart Aid for shoppers is a system where user can find where products they want to purchase would be exactly
LOGIN
SIGN UP



Our Objectives

Time & Energy

To test developed web-based application

Money

To assist shopper, save time and energy

To assist shopper in budgeting their goods from overspend

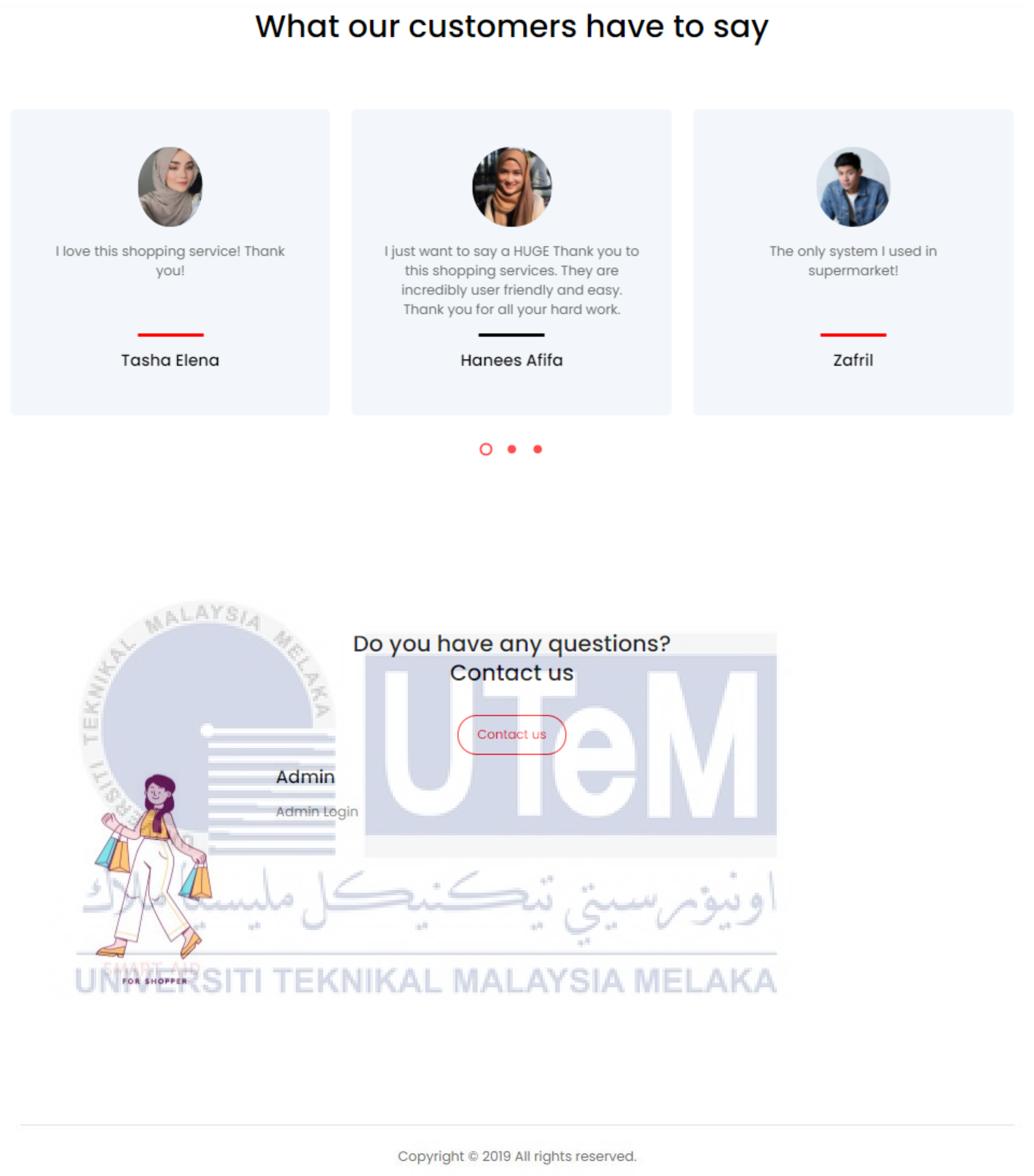
We Offer a Full Range of Shopping Services System!

This system is user-friendly and easy to give the best performance in any situation.

Leading for Shopper Solution.

Power-packed with impressive features and well-optimized, this template is designed to provide the best performance in all circumstances.

Its smart features make it a powerful stand-alone apps system.



What our customers have to say

I love this shopping service! Thank you!

Tasha Elena

I just want to say a HUGE Thank you to this shopping services. They are incredibly user friendly and easy. Thank you for all your hard work.

Hanees Afifa

The only system I used in supermarket!

Zafril

Do you have any questions?
Contact us

Contact us

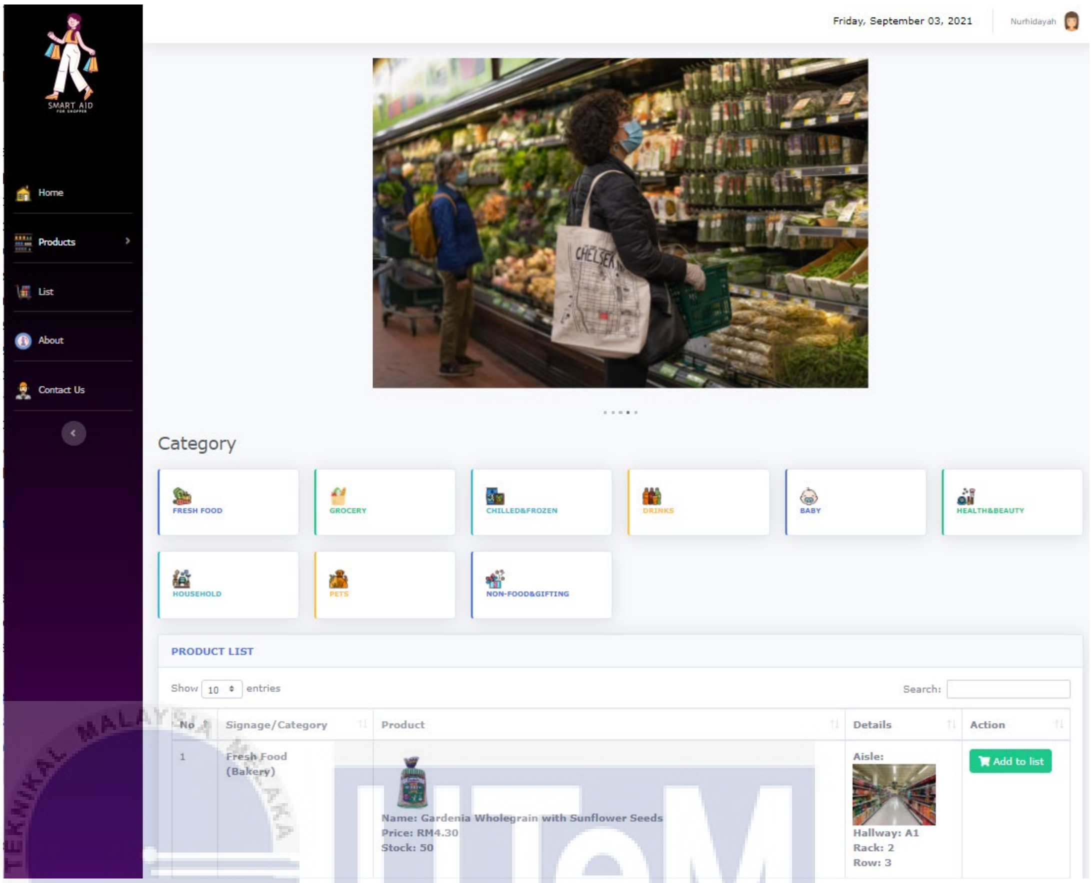
Admin

Admin Login

Copyright © 2019 All rights reserved.

Figure 4.3 Landing Page of SAFSS

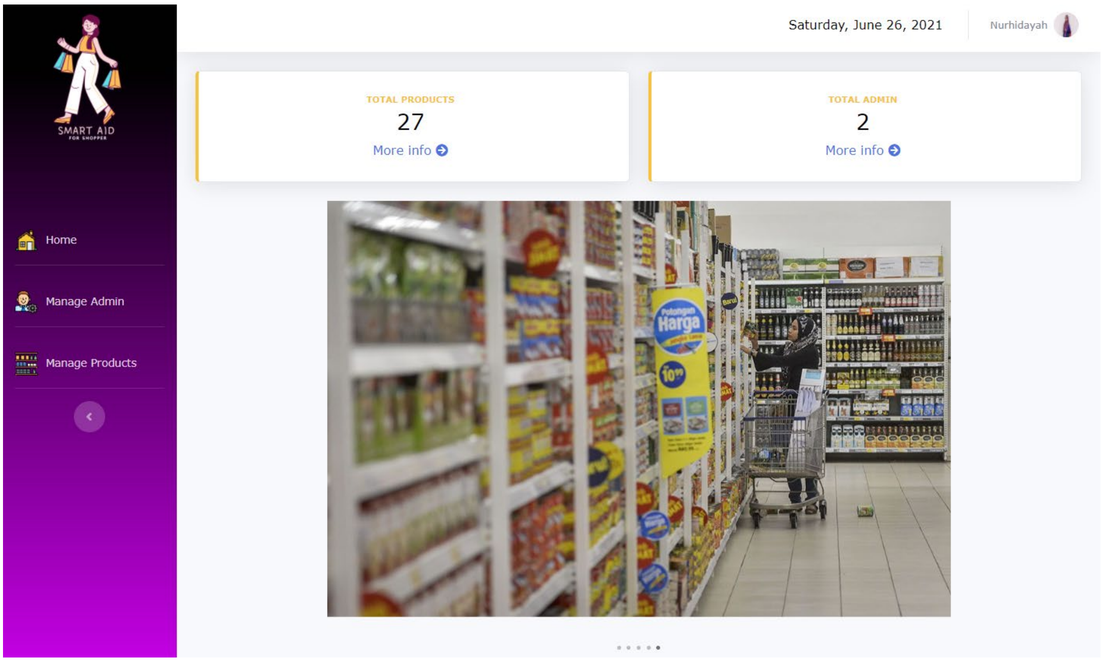
Landing Page is the main site of Smart Aid for Shopper System, where it displays an objective and intro about the system. Here, user can login or sign up to access into the system.



No	Signage/Category	Product	Details	Action
1	Fresh Food (Bakery)	Name: Gardenia Wholegrain with Sunflower Seeds Price: RM4.30 Stock: 50	Aisle: Hallway: A1 Rack: 2 Row: 3	Add to list

Figure 4.4 Main Page of Shopper

The main page of shopper displays all the category and all of the product list. Also, the search bar is used for user to search any goods they want to find. As well as get information about the location of the product item.



Saturday, June 26, 2021

Nurhidayah

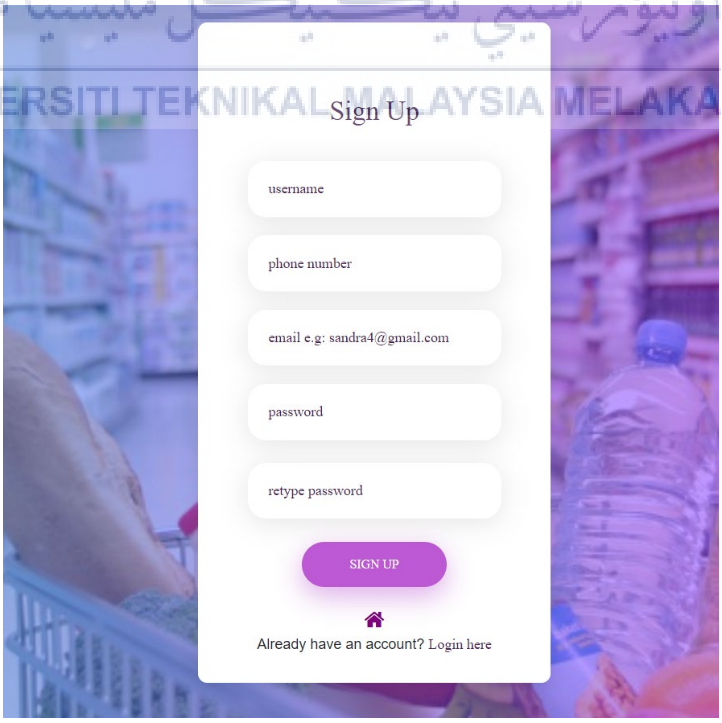
TOTAL PRODUCTS
27
More info

TOTAL ADMIN
2
More info

Figure 4.5 Main Page of Admin

This is the main site for admin after successfully login into the system. Here, it displays the total products and the total admin.

b) Input Design



Sign Up

username

phone number

email e.g: sandra4@gmail.com

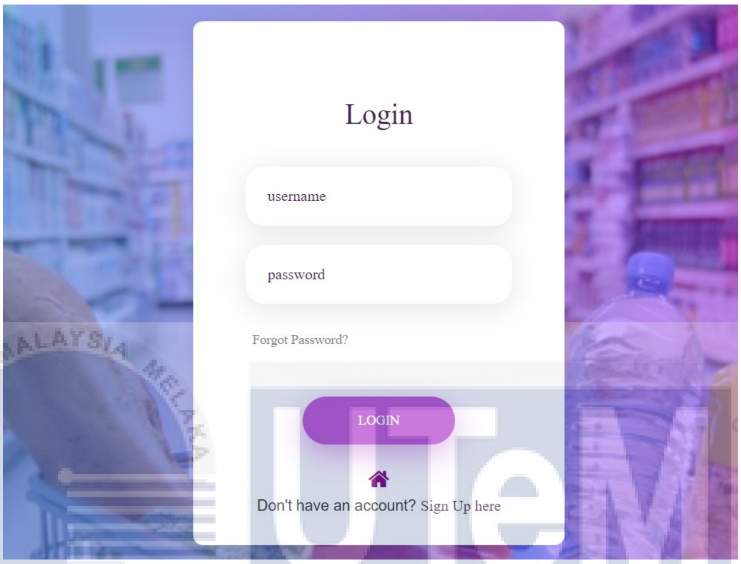
password

retype password

Already have an account? Login here

Figure 4.6 Sign Up for Shopper

To log into the system, shopper need to sign up first. In sign up site, shopper need to input username, phone number, email, password and retype password. All of the input are required. Shopper cannot insert the same username with the others user. If not, there will an output of exception message. Same with the phone number and email, shopper need to follow the system format and insert valid email. While for password, it need to be same with the retype password.



Login

username

password

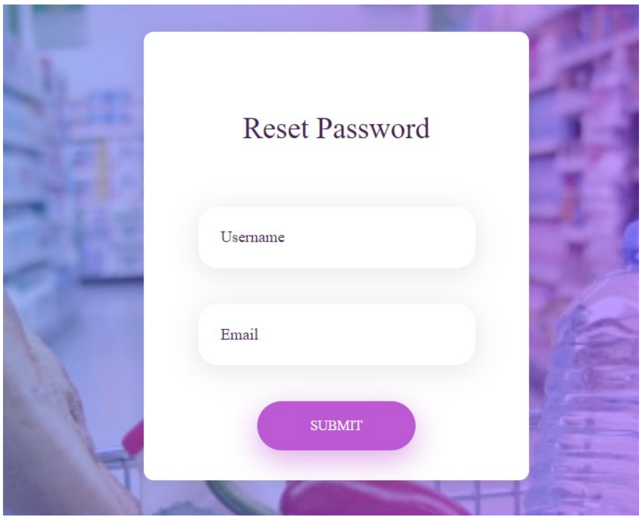
Forgot Password?

LOGIN

Don't have an account? Sign Up here

Figure 4.7 Login for Shopper

After shopper successfully signed up into the system, shopper need to insert username and password. Then click Login button.



Reset Password

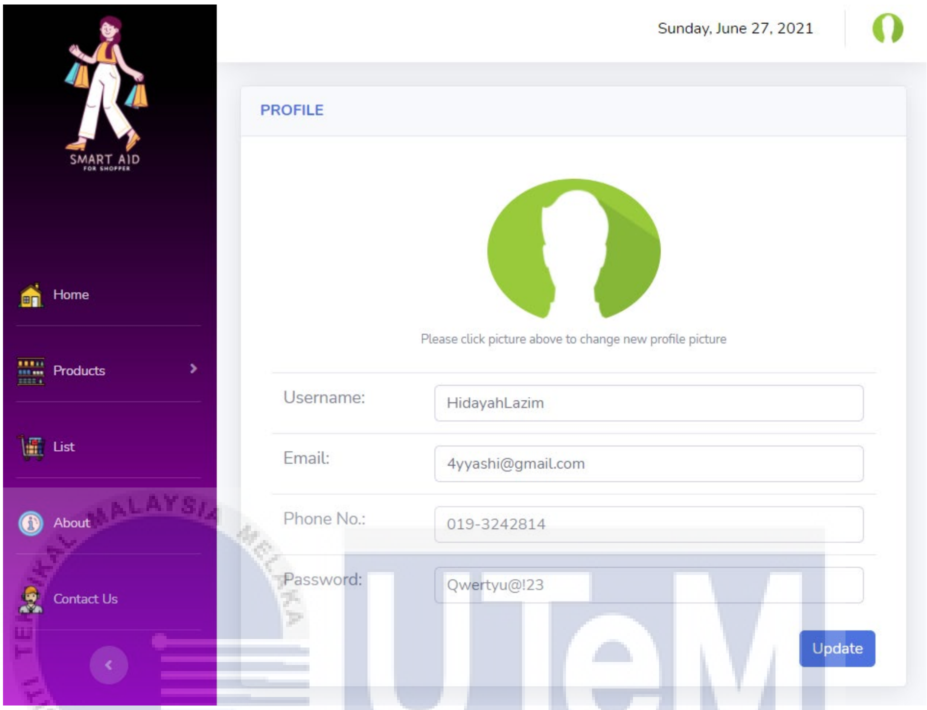
Username

Email

SUBMIT

Figure 4.8 Forgot Password for User

In case, shopper or admin forgot their password, they need to click the 'Forgot Password?' link that provided. Then need to put their username and email. After that, the system will send a link for shopper or admin to reset a new password.



Username: HidayahLazim

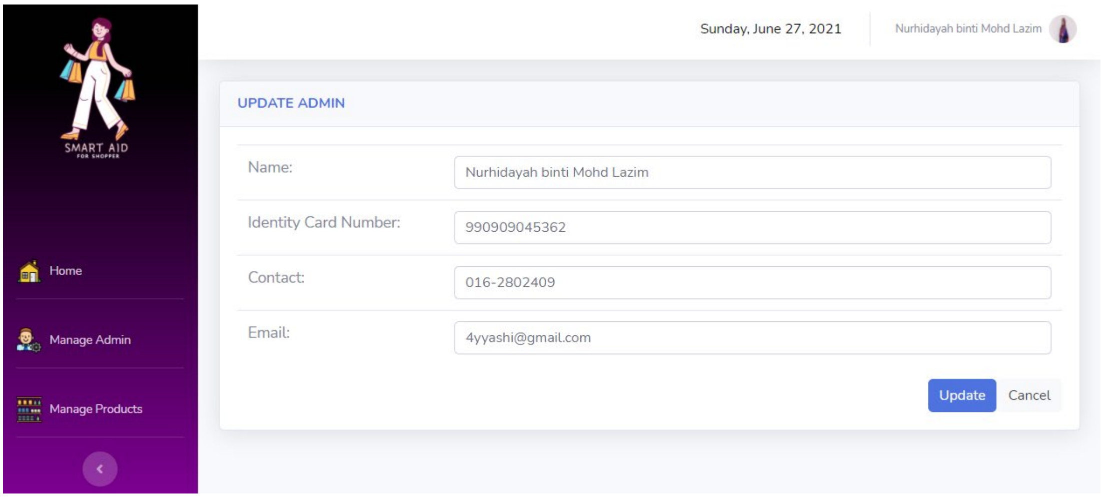
Email: 4yyashi@gmail.com

Phone No.: 019-3242814

Password: Qwertyu@!23

Figure 4.9 Shopper Profile

For a first time shopper, the profile picture will be in a default picture. Here, shopper can update their data such as shopper image, username, email, phone number or password.



Sunday, June 27, 2021 | Nurhidayah binti Mohd Lazim

UPDATE ADMIN

Name: Nurhidayah binti Mohd Lazim

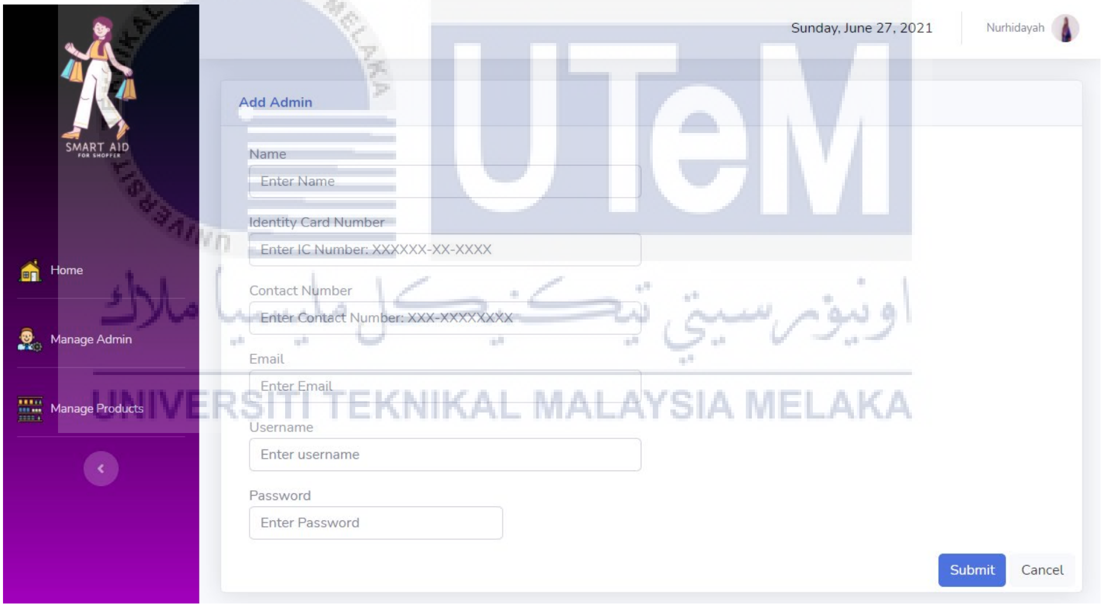
Identity Card Number: 990909045362

Contact: 016-2802409

Email: 4yyashi@gmail.com

Figure 4.10 Update Admin

Admin have a right to update any admin data if there is mistaken after input the data.



Sunday, June 27, 2021 | Nurhidayah

Add Admin

Name: Enter Name

Identity Card Number: Enter IC Number: XXXXXX-XX-XXXX

Contact Number: Enter Contact Number: XXX-XXXXXXXX

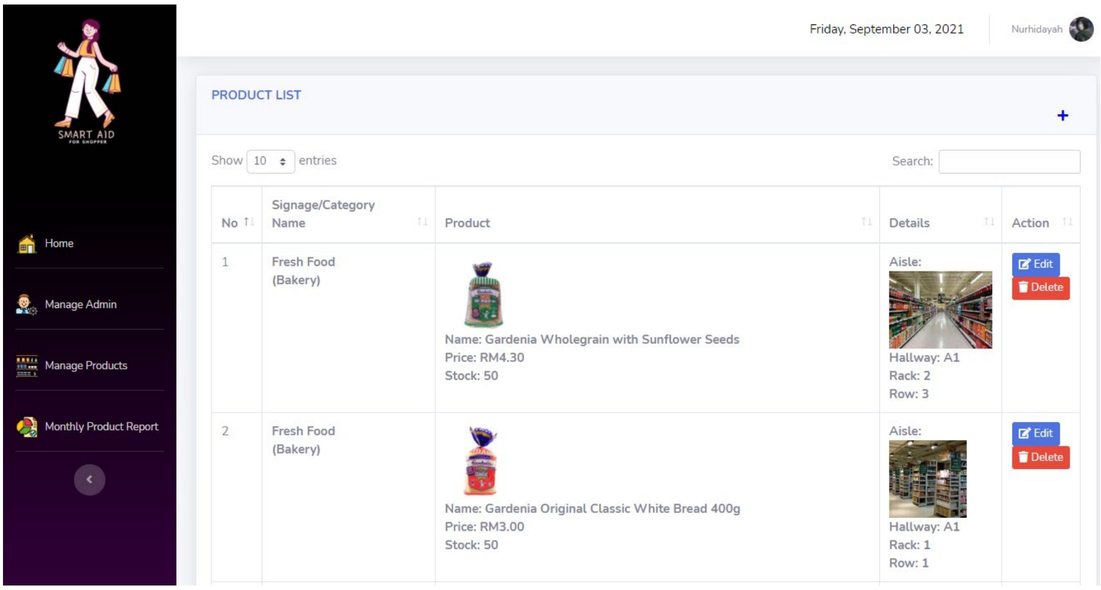
Email: Enter Email

Username: Enter username

Password: Enter Password

Figure 4.11 Add Admin

To add other admin into the system, admin need to input name, identity card number, username, contact number, email, and password. All of the input are required. Admin must follow the system format. If not, there will an output of exception message.



Friday, September 03, 2021

Nurhidayah

PRODUCT LIST

+

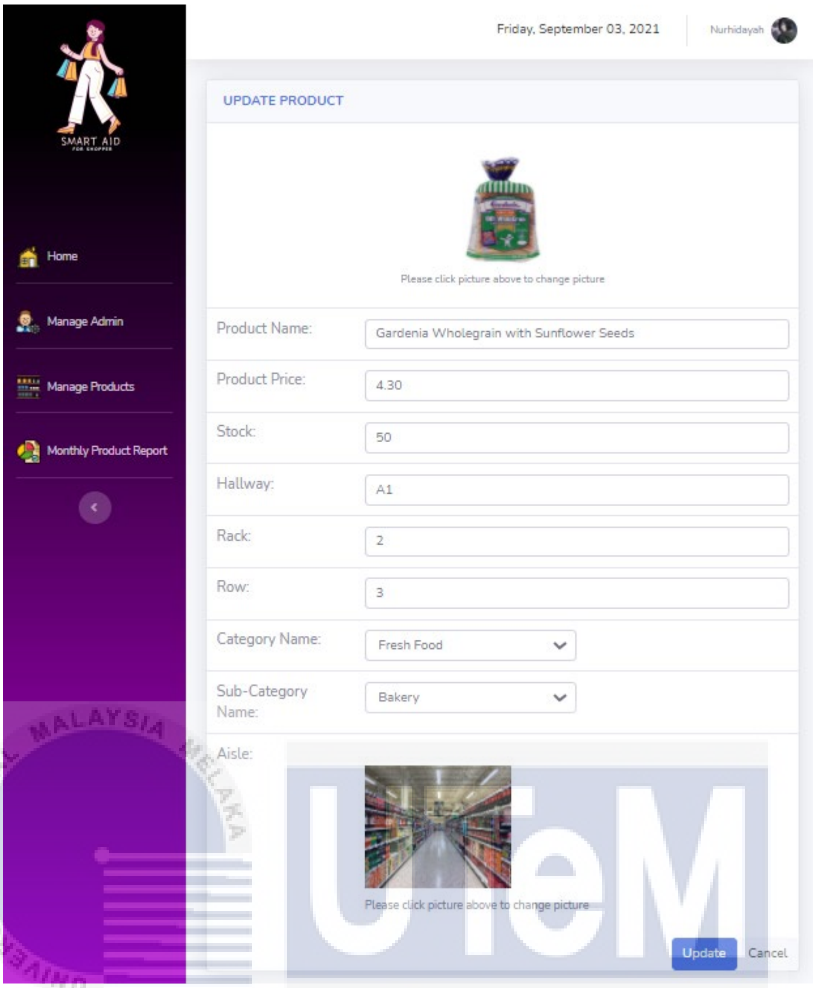
Show 10 entries

Search:

No	Signage/Category Name	Product	Details	Action
1	Fresh Food (Bakery)	Name: Gardenia Wholegrain with Sunflower Seeds Price: RM4.30 Stock: 50	Aisle: Hallway: A1 Rack: 2 Row: 3	Edit Delete
2	Fresh Food (Bakery)	Name: Gardenia Original Classic White Bread 400g Price: RM3.00 Stock: 50	Aisle: Hallway: A1 Rack: 1 Row: 1	Edit Delete

Figure 4.12 Manage Product Item

Admin can manage the product by add new product, update or delete the product item if necessary.


SMART AID
FOR SHOPPER
Home
Manage Admin
Manage Products
Monthly Product Report
Friday, September 03, 2021
Nurhidayah
UPDATE PRODUCT
Please click picture above to change picture
Please click picture above to change picture
Update
Cancel

Product Name: Gardenia Wholegrain with Sunflower Seeds

Product Price: 4.30

Stock: 50

Hallway: A1

Rack: 2

Row: 3

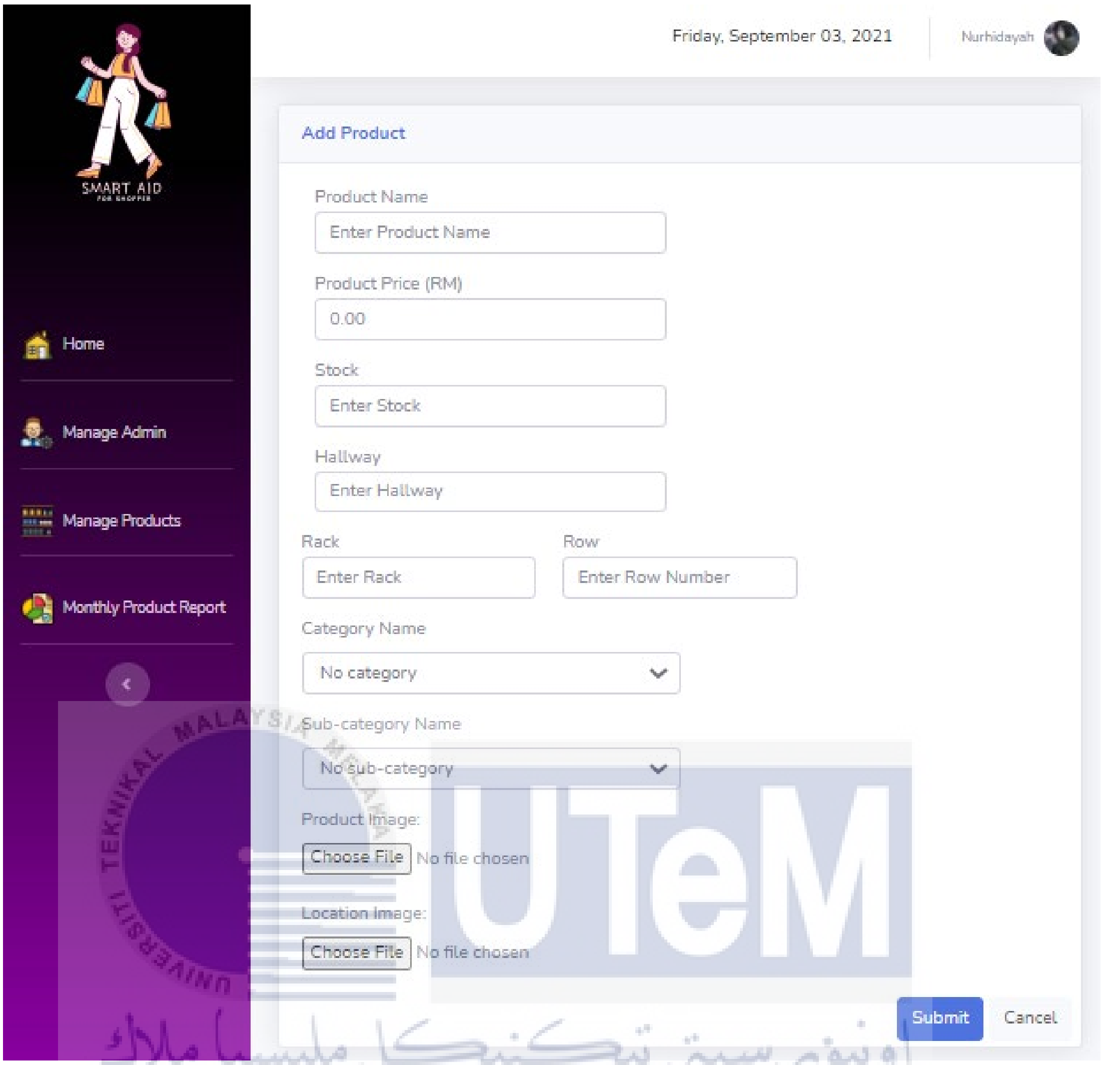
Category Name: Fresh Food

Sub-Category Name: Bakery

Aisle:

Figure 4.13 Edit Product Item

Admin have a right to update or edit the product if there is mistaken in the product data.



Add Product

Product Name
Enter Product Name

Product Price (RM)
0.00

Stock
Enter Stock

Hallway
Enter Hallway

Rack
Enter Rack

Row
Enter Row Number

Category Name
No category

Sub-category Name
No sub-category

Product Image:
No file chosen

Location Image:
No file chosen

Figure 4.14 Add Product Item

Admin need to insert the product name, product price, the stock, hallway, rack, row, category name, sub-category name, product image and the aisle image.

c) Output Design

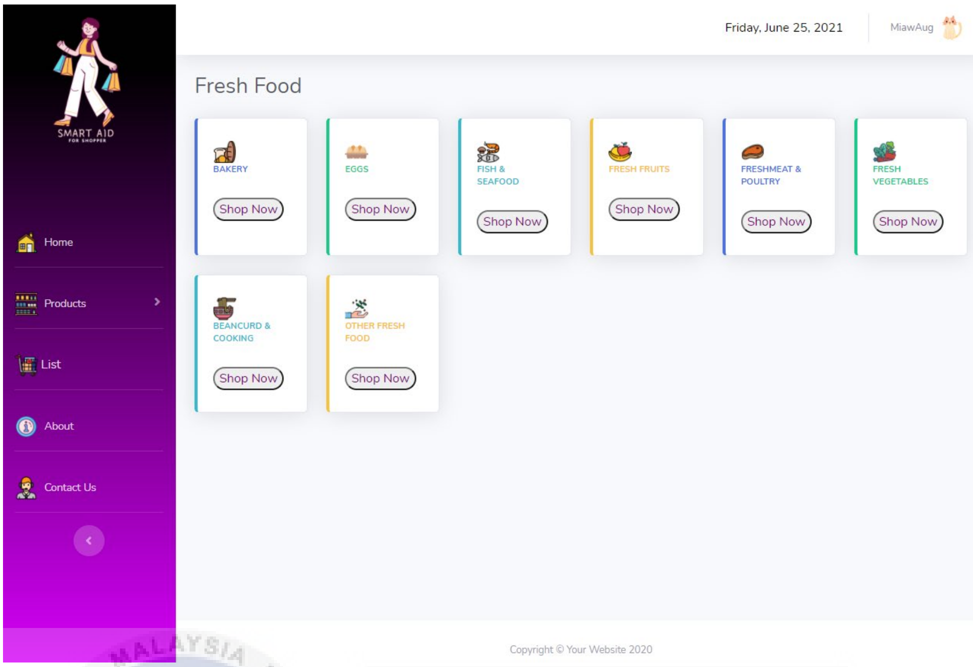
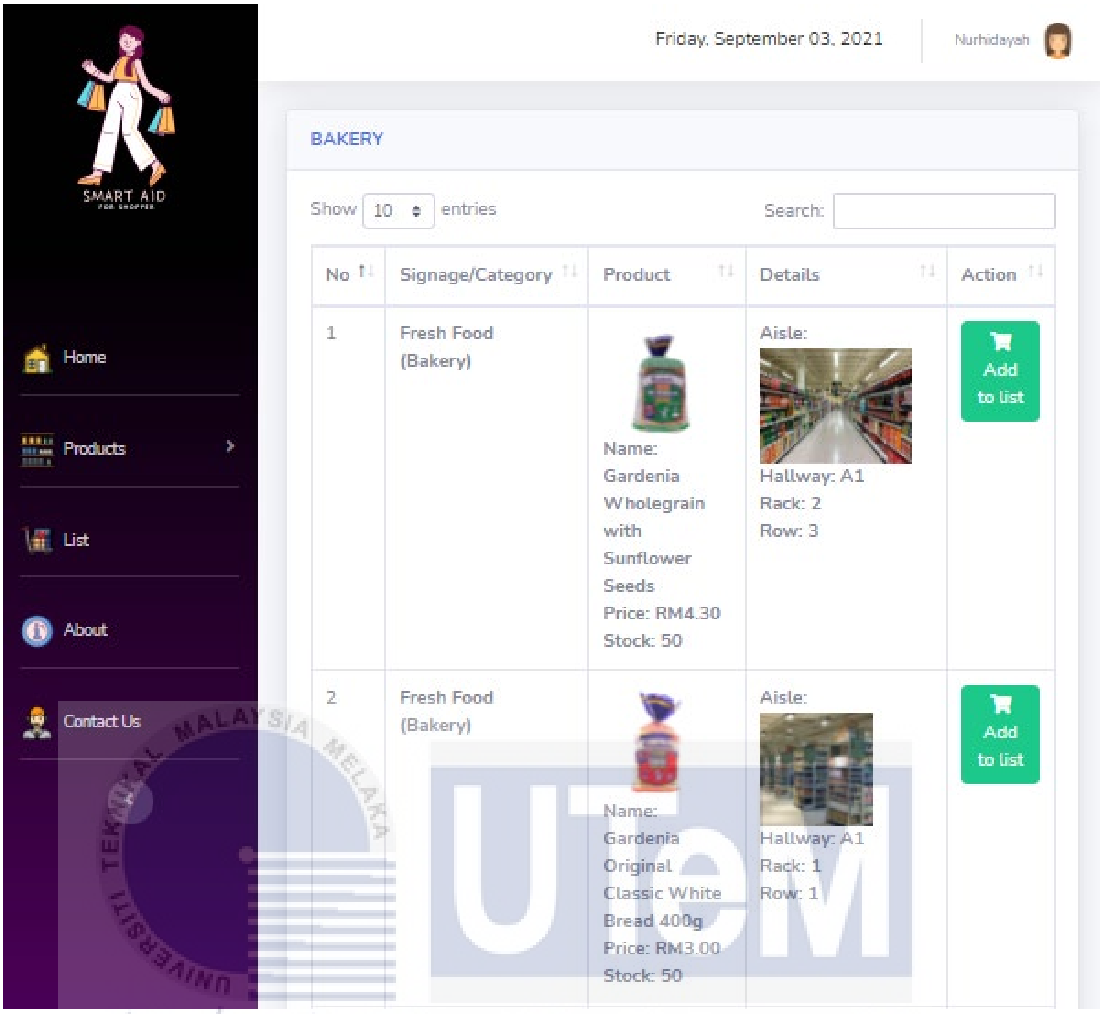


Figure 4.15 Products of Fresh Food

The subcategory list of fresh food.



Friday, September 03, 2021 | Nurhidayah

BAKERY

Show 10 entries Search:

No	Signage/Category	Product	Details	Action
1	Fresh Food (Bakery)	Name: Gardenia Wholegrain with Sunflower Seeds Price: RM4.30 Stock: 50	Aisle: Hallway: A1 Rack: 2 Row: 3	Add to list
2	Fresh Food (Bakery)	Name: Gardenia Original Classic White Bread 400g Price: RM3.00 Stock: 50	Aisle: Hallway: A1 Rack: 1 Row: 1	Add to list

Figure 4.16 Display of Product Item

For example, shopper select 'Shop Now' for Bakery. Here, the system display goods based on what shopper select.

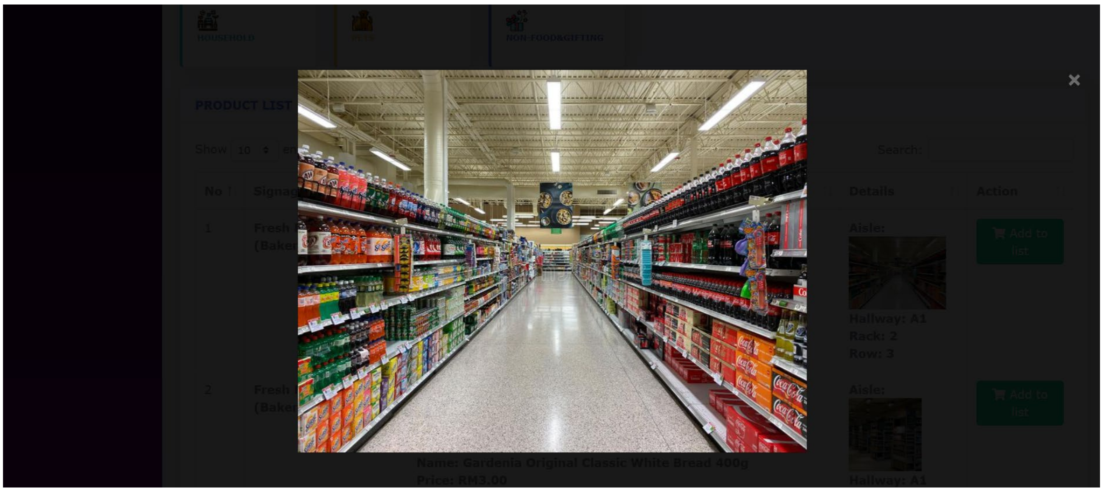
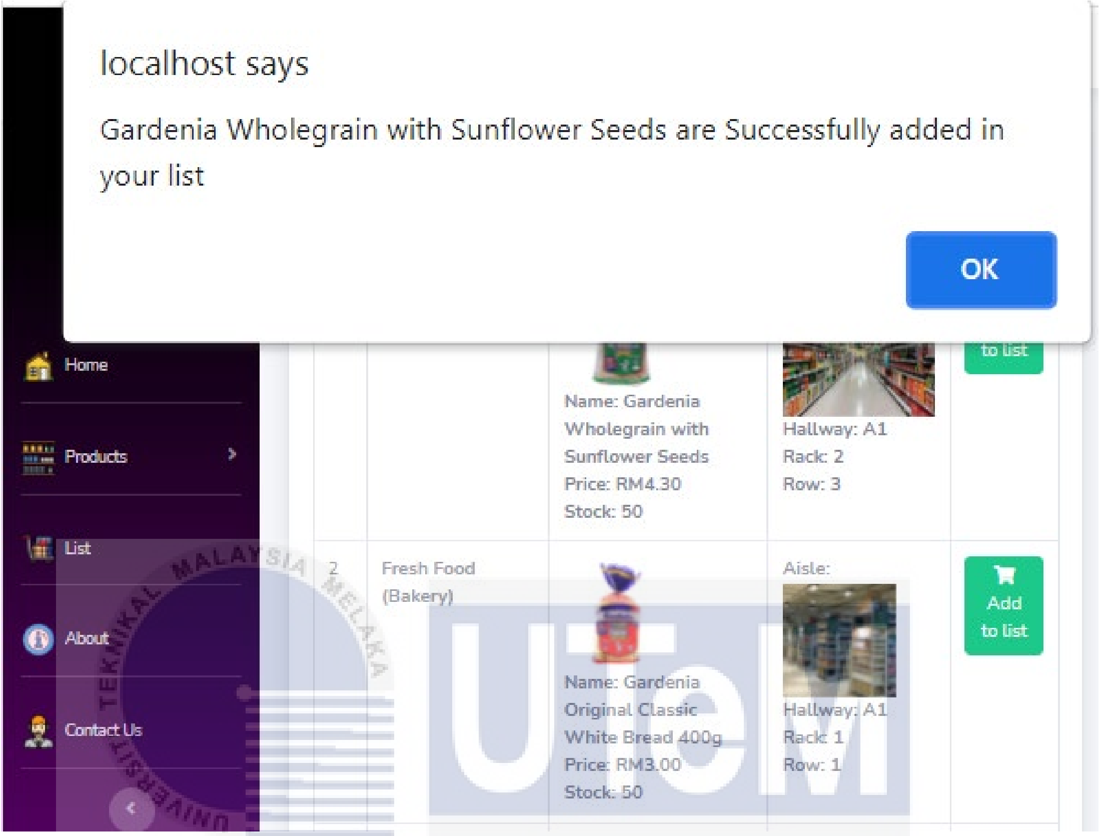


Figure 4.17 Big Picture of Aisle Image

When shopper or admin click the aisle image, the picture will become big due to it help user to have a clear view of product's placed.



localhost says

Gardenia Wholegrain with Sunflower Seeds are Successfully added in your list

Name: Gardenia Wholegrain with Sunflower Seeds
Price: RM4.30
Stock: 50

Hallway: A1
Rack: 2
Row: 3

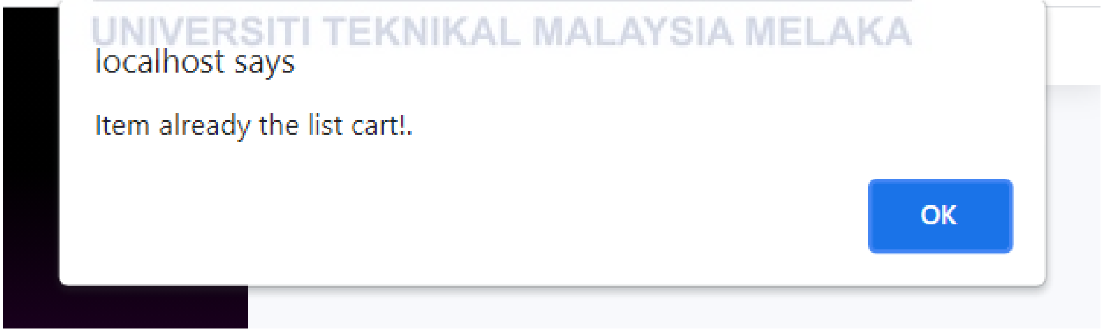
2 Fresh Food (Bakery)

Aisle:

Name: Gardenia Original Classic White Bread 400g
Price: RM3.00
Stock: 50

Hallway: A1
Rack: 1
Row: 1

Figure 4.18 Add Product Item to the List

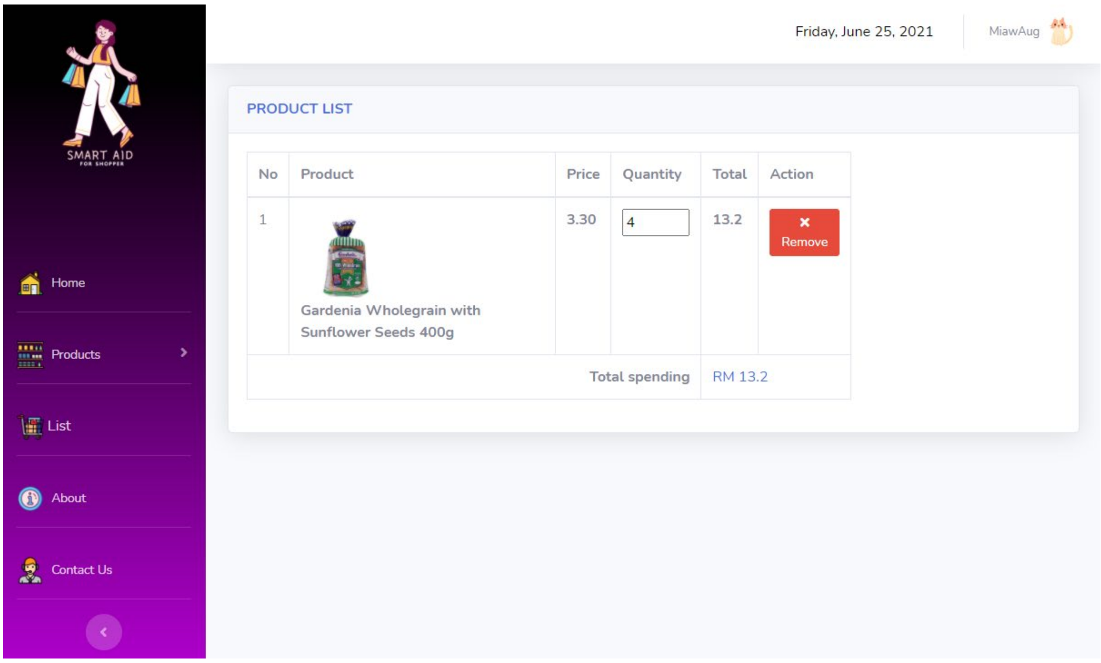


localhost says

Item already the list cart!.

Figure 4.19 Item already added in the list

If shopper click the 'Add to list' button, it will add the product into the list. Then, if shopper click for the second time at the same product, it will output an exception message says 'Items Are already in the list cart'.



Friday, June 25, 2021

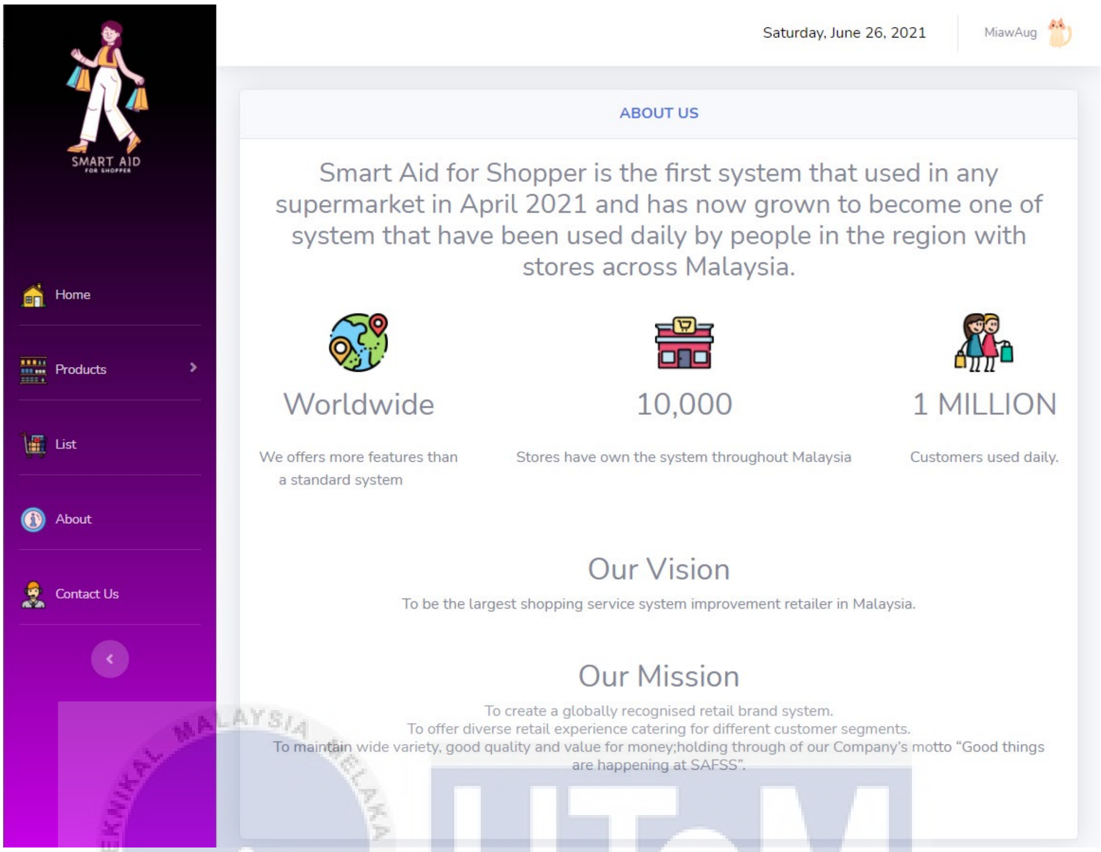
MiawAug

PRODUCT LIST

No	Product	Price	Quantity	Total	Action
1	Gardenia Wholegrain with Sunflower Seeds 400g	3.30	4	13.2	✖ Remove
			Total spending	RM 13.2	

Figure 4.20 List Menu

In the list site, shopper can manage their spending on any products by removing or add more product. If the total spending is overspending from their budget they can remove the product or decrease the unit product number.



ABOUT US

Smart Aid for Shopper is the first system that used in any supermarket in April 2021 and has now grown to become one of system that have been used daily by people in the region with stores across Malaysia.

Worldwide

We offers more features than a standard system

10,000

Stores have own the system throughout Malaysia

1 MILLION

Customers used daily.

Our Vision

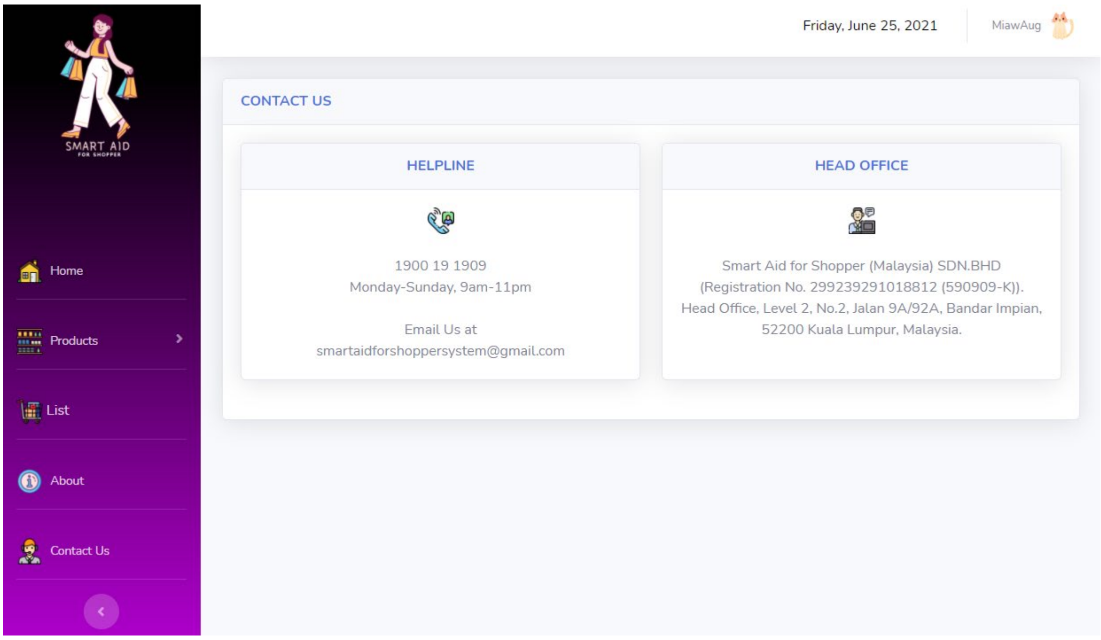
To be the largest shopping service system improvement retailer in Malaysia.

Our Mission

To create a globally recognised retail brand system.
To offer diverse retail experience catering for different customer segments.
To maintain wide variety, good quality and value for money;holding through of our Company's motto "Good things are happening at SAFSS".

Figure 4.21 About us

At this about site, generally it tells shopper the details of Smart Aid for Shopper. Such as the vision and mission.



CONTACT US

HELPLINE

1900 19 1909
Monday-Sunday, 9am-11pm

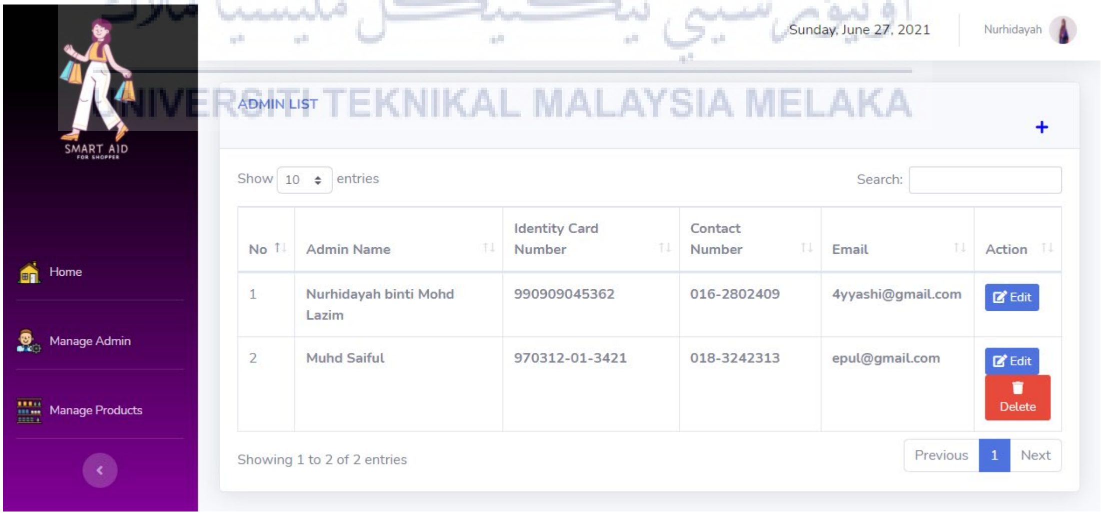
Email Us at
smartaidforshoppersystem@gmail.com

HEAD OFFICE

Smart Aid for Shopper (Malaysia) SDN.BHD
(Registration No. 299239291018812 (590909-K)).
Head Office, Level 2, No.2, Jalan 9A/92A, Bandar Impian,
52200 Kuala Lumpur, Malaysia.

Figure 4.22 Contact us

At contact us site, the system will display the helpline number, email and address. In case, if any shopper wants to share any problem or feedback.



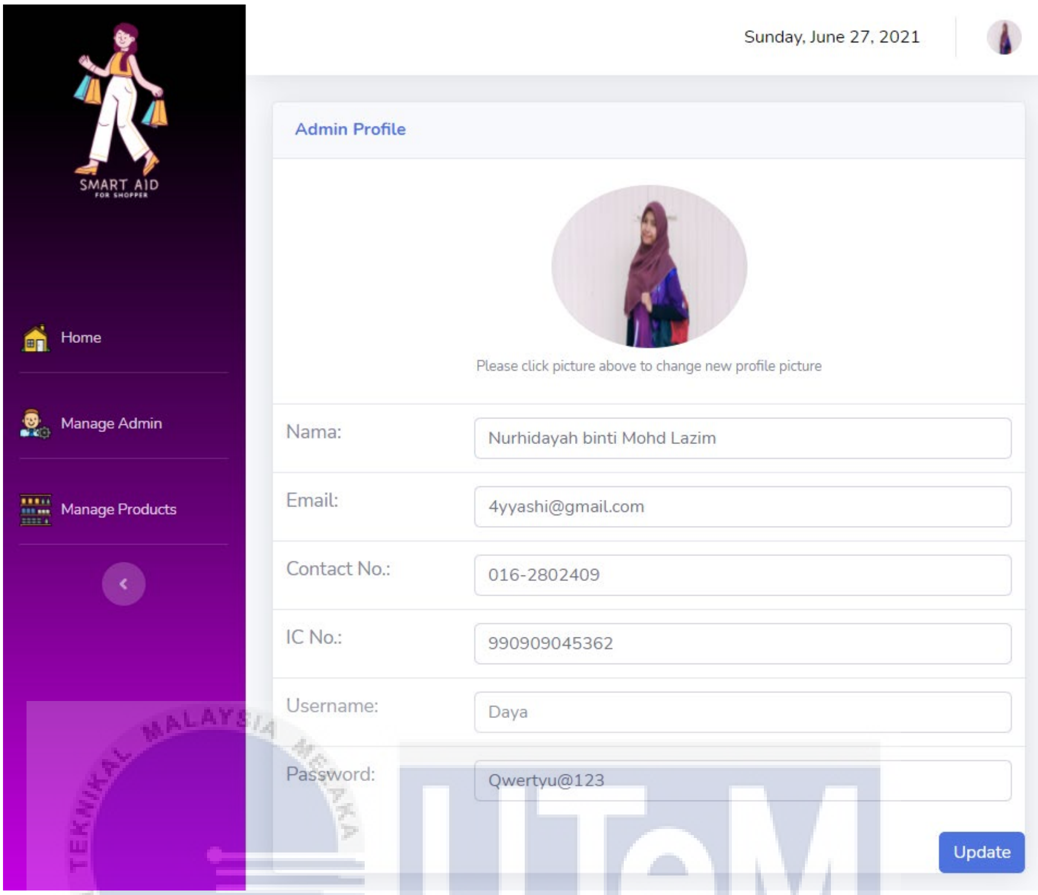
ADMIN LIST

No	Admin Name	Identity Card Number	Contact Number	Email	Action
1	Nurhidayah binti Mohd Lazim	990909045362	016-2802409	4yyashi@gmail.com	Edit
2	Muhd Saiful	970312-01-3421	018-3242313	epul@gmail.com	Edit Delete

Showing 1 to 2 of 2 entries

Figure 4.23 Manage Admin

At this site, the system shows who are responsibility in managing the products. As an admin, they can edit or delete data.



Admin Profile

Please click picture above to change new profile picture

Nama: Nurhidayah binti Mohd Lazim

Email: 4yyashi@gmail.com

Contact No.: 016-2802409

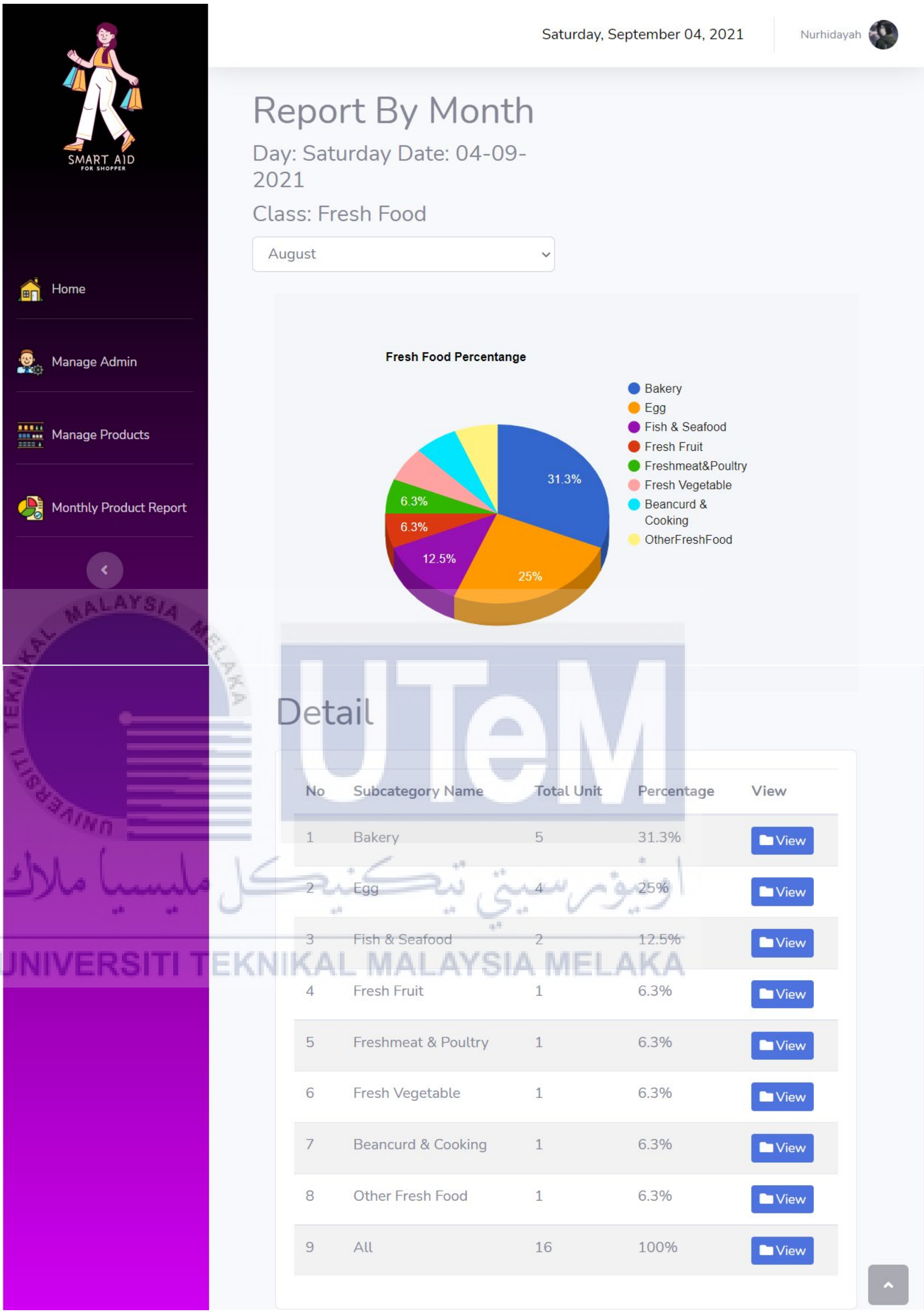
IC No.: 990909045362

Username: Daya

Password: Qwertyu@123

Figure 4.24 Admin Profile

The system will display the admin profile based on who are login into the system. Besides, admin can update or edit their data.



Saturday, September 04, 2021 Nurhidayah

Report By Month

Day: Saturday Date: 04-09-2021

Class: Fresh Food

August

Detail

No	Subcategory Name	Total Unit	Percentage	View
1	Bakery	5	31.3%	View
2	Egg	4	25%	View
3	Fish & Seafood	2	12.5%	View
4	Fresh Fruit	1	6.3%	View
5	Freshmeat & Poultry	1	6.3%	View
6	Fresh Vegetable	1	6.3%	View
7	Beancurd & Cooking	1	6.3%	View
8	Other Fresh Food	1	6.3%	View
9	All	16	100%	View

Figure 4.25 Monthly Product Report

At this site, Admin can analyze monthly product report where they can know total unit product, view product list and the graph percentage.

4.2.3 Database Design

4.2.3.1 Conceptual and Logical Database Design

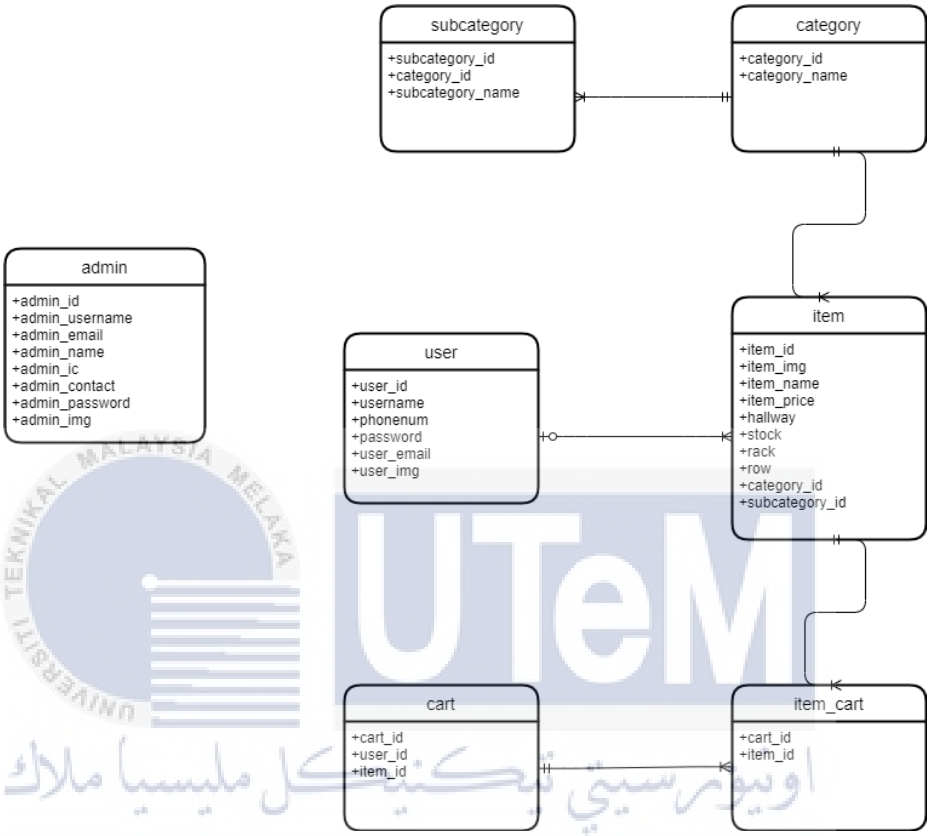


Figure 4.26 Conceptual Design for SAFSS

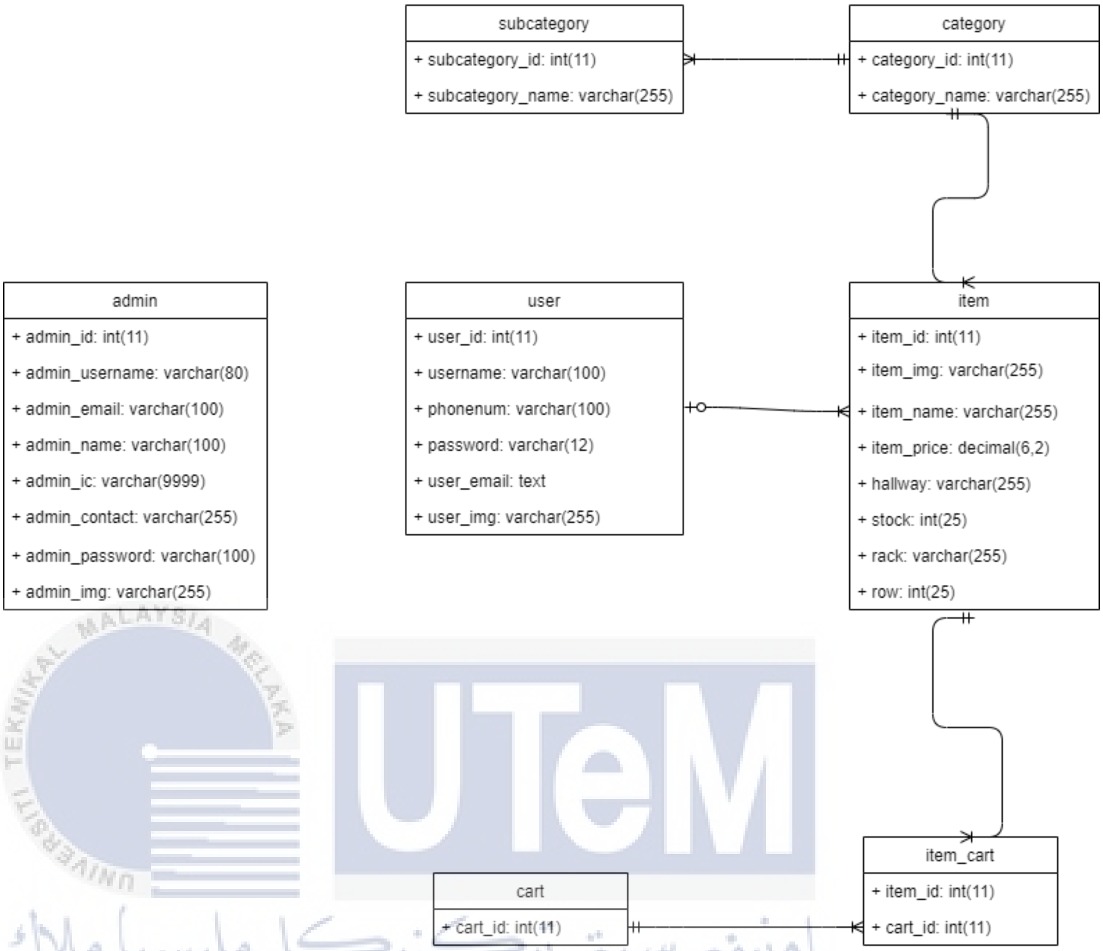


Figure 4.27 Logical Design for SAFSS

4.3 Detailed Design

In the preceding two designs, detailed design deals with the execution of what is regarded as a system and its sub-systems. It delves deeper into modules and their implementations. It describes the logical structure of each module as well as their interactions with other modules.

4.3.1 Software Design

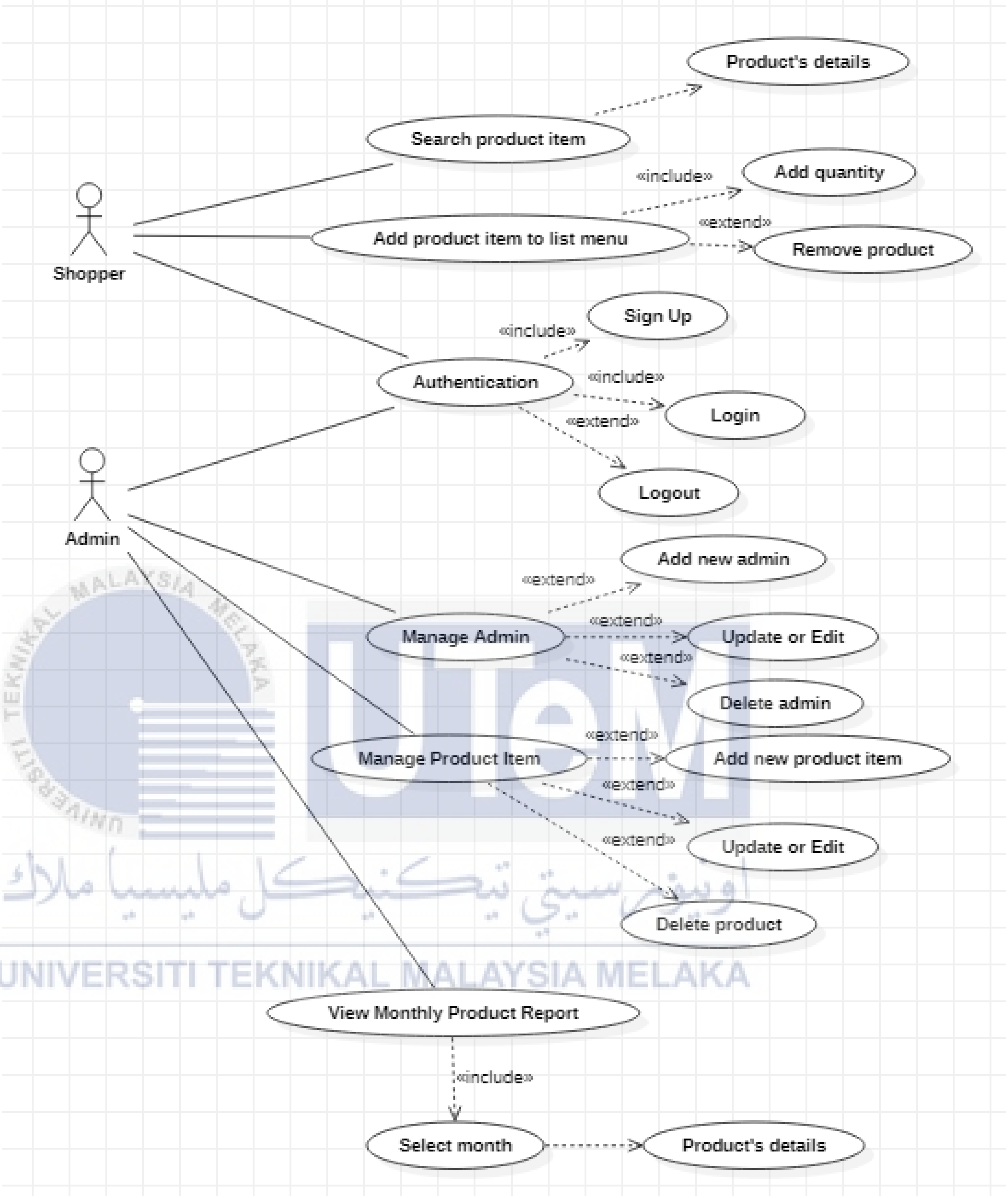
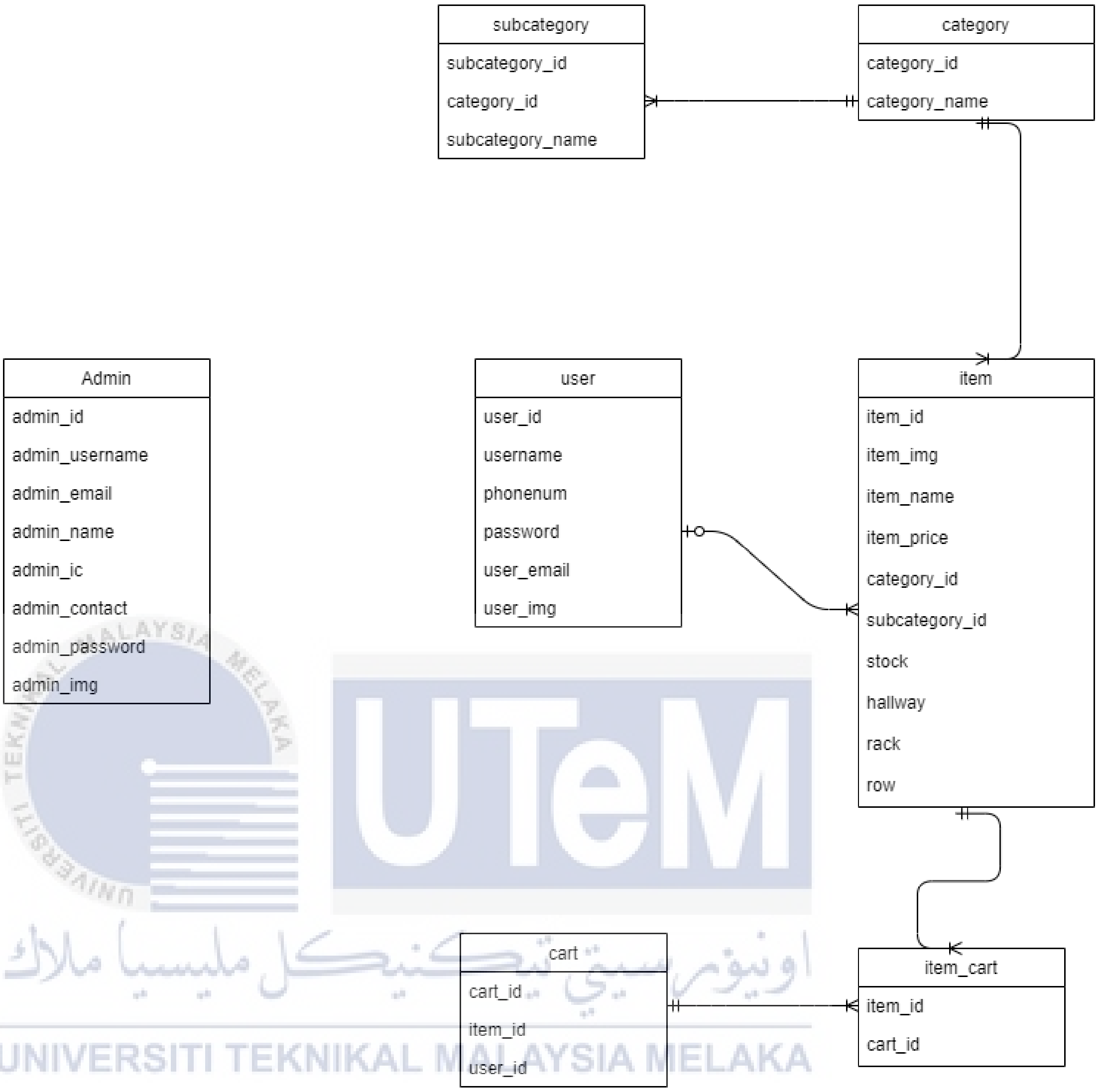


Figure 4.28 UML of SAFSS

4.3.2 Physical Database Design



4.4 Conclusion

To summarize, this chapter covered four major topics: system architecture, database design, module integration and interface design, and system design. This project must be clear, fulfil the demands of the user, and achieve the objectives that have been set.

CHAPTER 5: IMPLEMENTATION

5.1 Introduction

This chapter will go through the system development environment, system configuration management, and security features. The system development environment describes the tools utilized and the procedures involved in creating this project. The method of ensuring consistency of a project's performance, functional, and physical properties is described in system configuration management.

5.2 Software Development Environment setup

Diagram of environment architecture for Smart Aid for Shopper System (SAFSS) will be illustrates in figure 5.1 below.

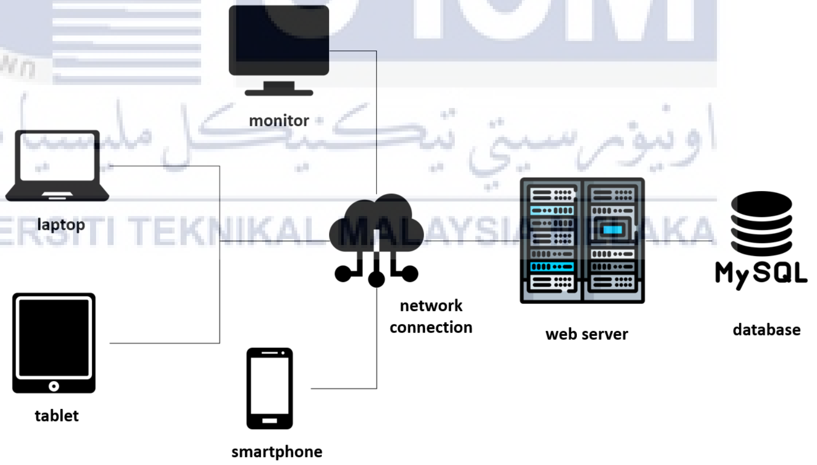


Figure 5.1: Diagram of Environment Architecture

This Smart Aid for Shopper System (SAFSS) is a fully web-based system which uses the PHP programming language with basic functionality of displaying the data and others. Web-based information system is usually to describe applications that run in a web browser. It also be used to describe applications that have a small component solution loaded on the client gadget such as smartphone, tablet, monitor, laptop. This diagram involves client side, network connection, server and database. SAFSS is connected with server which is Apache through web browser such as Google

Chrome. All data that are received from the system will be stored in MySQL database which is connected with the server. This system makes use of the MySQL database, which is an open-source relational database. The most frequent application for MySQL is as an online database.

5.3 Software Configuration Management

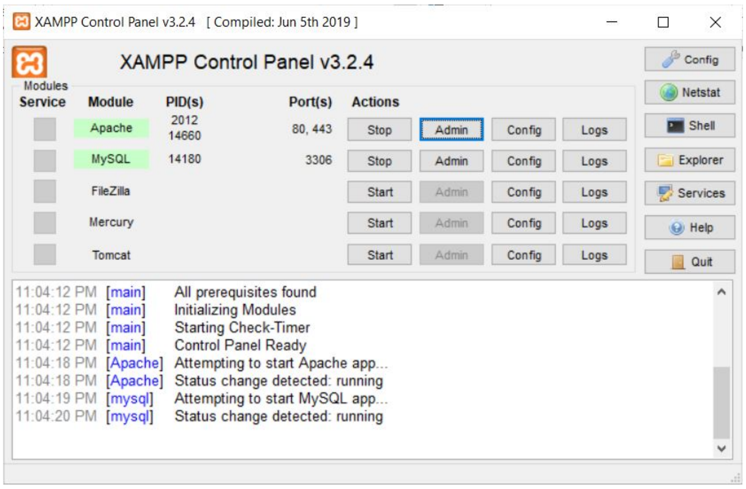
In this section, System configuration management is a procedure for methodically managing, organizing, and controlling changes in documents, codes, and other entities across the Software Development Life Cycle. The major objective is to enhance production while minimizing errors. SCM is a component of the multidisciplinary field of configuration management, and it can precisely ascertain who made which modification

5.3.1 Configuration environment setup

Smart Aid for Shopper System(SAFSS) is developed by php programming using source code editor software which is Sublime Text 3 as shown in Figure 5.2. Sublime Text 3 is a free software editor that can be downloaded from web browser.

Figure 5.2: Sublime Text 3

For server and database configuration, it uses XAMPP Control Panel v3.2.4 which it is a free open source web server that consists of Apache, PHP, MariaDB and other components. SAFSS uses Apache server on the web and MySQL phpMyAdmin database to store and manipulate all data in this system. Figure 5.3 below shows the control panel window of XAMPP.



XAMPP Control Panel v3.2.4 [Compiled: Jun 5th 2019]

XAMPP Control Panel v3.2.4

Service	Module	PID(s)	Port(s)	Actions			
	Apache	2012 14660	80, 443	Stop	Admin	Config	Logs
	MySQL	14180	3306	Stop	Admin	Config	Logs
	FileZilla			Start	Admin	Config	Logs
	Mercury			Start	Admin	Config	Logs
	Tomcat			Start	Admin	Config	Logs

```
11:04:12 PM  [main]     All prerequisites found
11:04:12 PM  [main]     Initializing Modules
11:04:12 PM  [main]     Starting Check-Timer
11:04:12 PM  [main]     Control Panel Ready
11:04:18 PM  [Apache]   Attempting to start Apache app...
11:04:18 PM  [Apache]   Status change detected: running
11:04:19 PM  [mysql]    Attempting to start MySQL app...
11:04:20 PM  [mysql]    Status change detected: running
```

Figure 5.3: XAMPP Control Panel

5.3.2 Version Control Procedure

Version Control Procedure is the process by which different drafts and versions of a document are managed. The goals are to keep track of the system develop and provide full back up in case there is any modification are needed to be undo. In this project system, version control is done manually where file folders are saved time to time. Table 5.1 below shows the version control procedure for SAFSS.

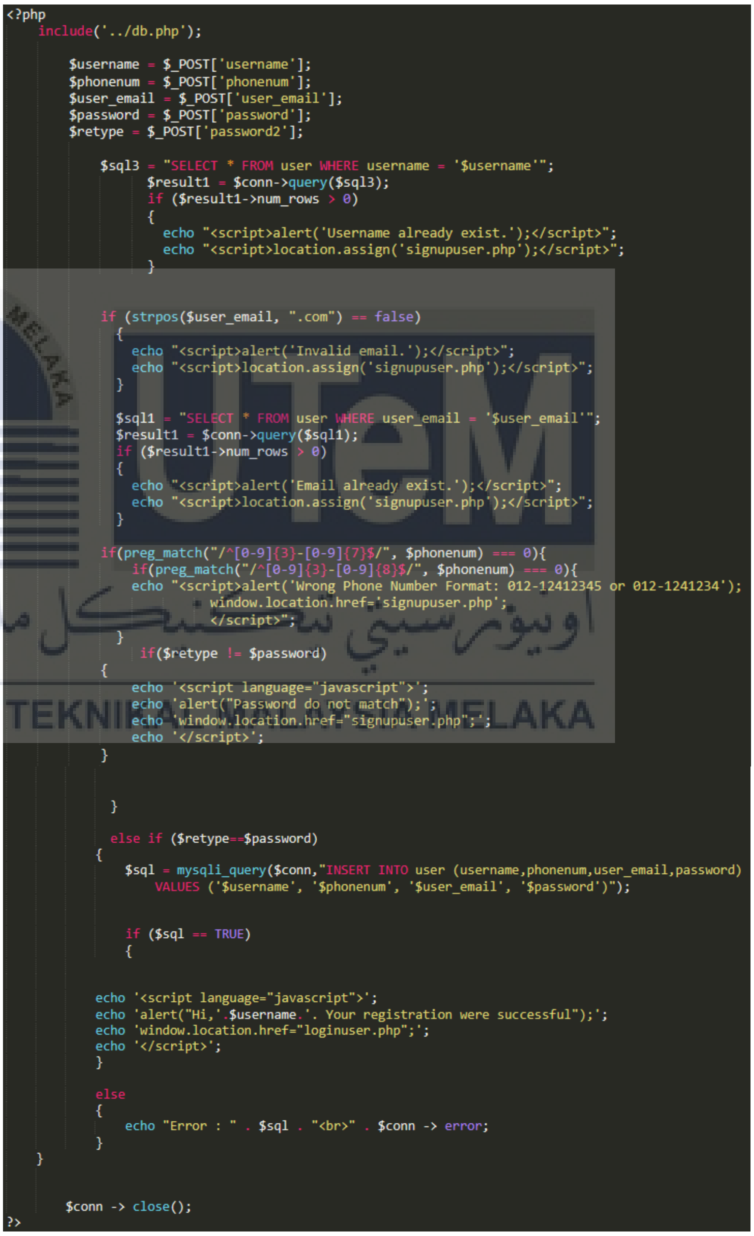
Table 5.1: Version Control Procedure

Folder Name	Date Modify	Version Number
smartaid1	21/04/2021	1
smartaid2	30/05/2021	2
smartaid3	30/05/2021	3
smartaid4	07/06/2021	4
smartaid5	08/06/2021	5
smartaid	19/08/2021	6

5.4 Implementation Status

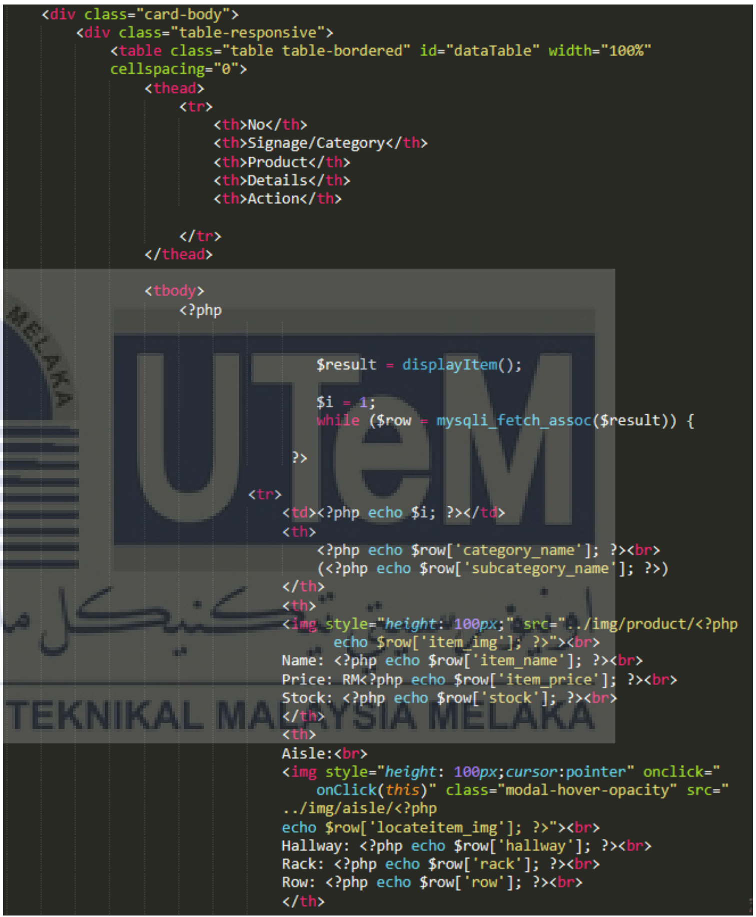
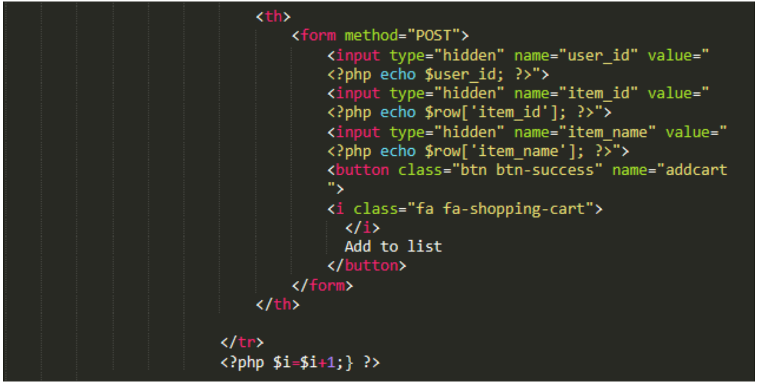
The progress of the development status for each module in the system are described as shown in Table 5.2 below.

Table 5.1 Authentication

Module Name: Authentication	
Description	This module will allow specialized users to log into the system using their own credentials.
Duration to complete	4 days
Date completed	27/06/2021
File size	3 KB
Implementation code	

```
<?php
    include('../db.php');

        $username = $_POST['username'];
        $phonenum = $_POST['phonenum'];
        $user_email = $_POST['user_email'];
        $password = $_POST['password'];
        $retype = $_POST['password2'];

            $sql3 = "SELECT * FROM user WHERE username = '$username'";
                  $result1 = $conn->query($sql3);
                  if ($result1->num_rows > 0)
                  {
                    echo "<script>alert('Username already exist.');</script>";
                    echo "<script>location.assign('signupuser.php');</script>";
                  }


            if (strpos($user_email, ".com") == false)
              {
                echo "<script>alert('Invalid email.');</script>";
                echo "<script>location.assign('signupuser.php');</script>";
              }

              $sql1 = "SELECT * FROM user WHERE user_email = '$user_email'";
              $result1 = $conn->query($sql1);
              if ($result1->num_rows > 0)
              {
                echo "<script>alert('Email already exist.');</script>";
                echo "<script>location.assign('signupuser.php');</script>";
              }
            if(preg_match("/^[0-9]{3}-[0-9]{7}$/", $phonenum) === 0){
                if(preg_match("/^[0-9]{3}-[0-9]{8}$/", $phonenum) === 0){
                echo "<script>alert('Wrong Phone Number Format: 012-12412345 or 012-1241234');
                          window.location.href='signupuser.php';
                          </script>";
              }
                 if($retype != $password)
            {
                echo '<script language="javascript">';
                echo 'alert("Password do not match");';
                echo 'window.location.href="signupuser.php";';
                echo '</script>';
            }


             }

             else if ($retype==$password)
            {
               $sql = mysqli_query($conn,"INSERT INTO user (username,phonenum,user_email,password)
                   VALUES ('$username', '$phonenum', '$user_email', '$password')");


               if ($sql == TRUE)
               {


           echo '<script language="javascript">';
           echo 'alert("Hi,'.$username.'. Your registration were successful");';
           echo 'window.location.href="loginuser.php";';
           echo '</script>';
           }

           else
           {
               echo "Error : " . $sql . "<br>" . $conn -> error;
           }
    }


        $conn -> close();
?>
```

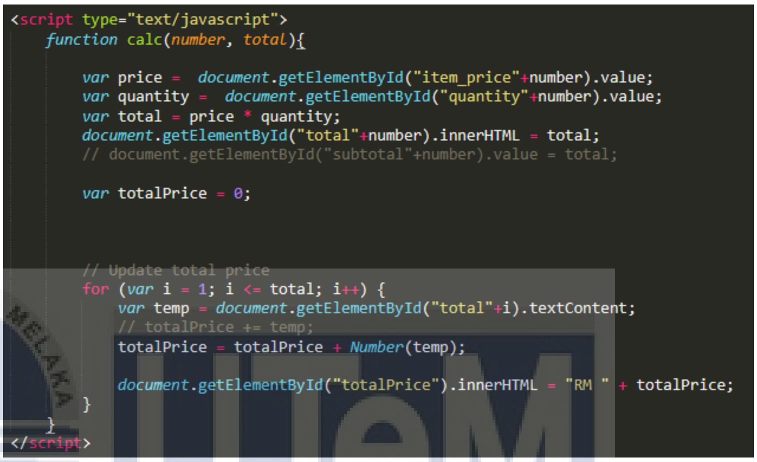
Table 5.2 Search Product Location

Module Name: Search product location	
Description	This module will allow shopper to search any product details and product's location.
Duration to complete	4 days
Date completed	31/08/2021
File size	3 KB
Implementation code	 

```
<div class="card-body">
    <div class="table-responsive">
        <table class="table table-bordered" id="dataTable" width="100%"
        cellspacing="0">
            <thead>
                <tr>
                    <th>No</th>
                    <th>Signage/Category</th>
                    <th>Product</th>
                    <th>Details</th>
                    <th>Action</th>

                </tr>
            </thead>

            <tbody>
                <?php


                                $result = displayItem();

                                $i = 1;
                                while ($row = mysqli_fetch_assoc($result)) {

                             ?>

                        <tr>
                            <td><?php echo $i; ?></td>
                            <th>
                                <?php echo $row['category_name']; ?><br>
                                (<?php echo $row['subcategory_name']; ?>)
                            </th>
                            <th>
                            <img style="height: 100px;" src="../img/product/<?php
                                echo $row['item_img']; ?>"><br>
                            Name: <?php echo $row['item_name']; ?><br>
                            Price: RM<?php echo $row['item_price']; ?><br>
                            Stock: <?php echo $row['stock']; ?><br>
                            </th>
                            <th>
                            Aisle:<br>
                            <img style="height: 100px;cursor:pointer" onclick="
                                onClick(this)" class="modal-hover-opacity" src="
                            ../img/aisle/<?php
                            echo $row['locateitem_img']; ?>"><br>
                            Hallway: <?php echo $row['hallway']; ?><br>
                            Rack: <?php echo $row['rack']; ?><br>
                            Row: <?php echo $row['row']; ?><br>
                            </th>
```

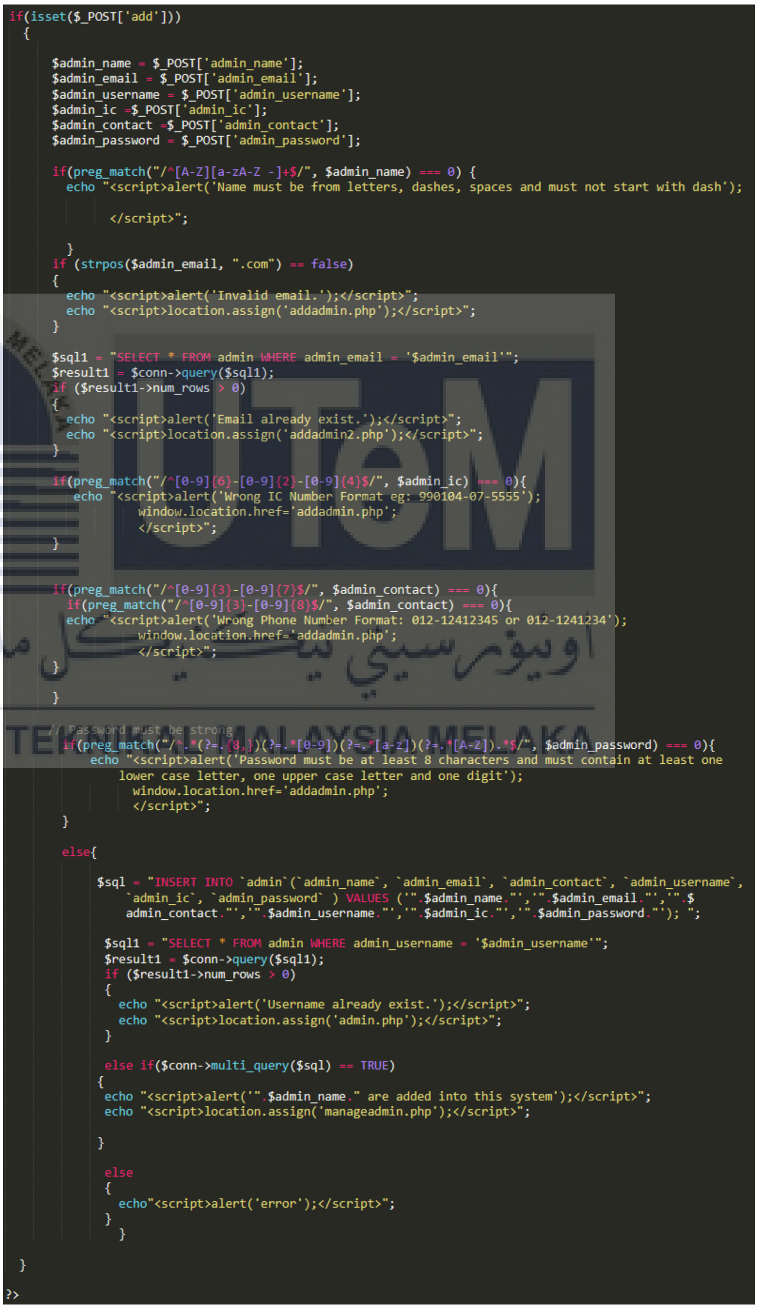
```
                        <th>
                            <form method="POST">
                                <input type="hidden" name="user_id" value="
                                <?php echo $user_id; ?>">
                                <input type="hidden" name="item_id" value="
                                <?php echo $row['item_id']; ?>">
                                <input type="hidden" name="item_name" value="
                                <?php echo $row['item_name']; ?>">
                                <button class="btn btn-success" name="addcart
                                ">
                                <i class="fa fa-shopping-cart">
                                  </i>
                                  Add to list
                                </button>
                            </form>
                        </th>

                    </tr>
                    <?php $i=$i+1;} ?>
```

Table 5.3 Shop List

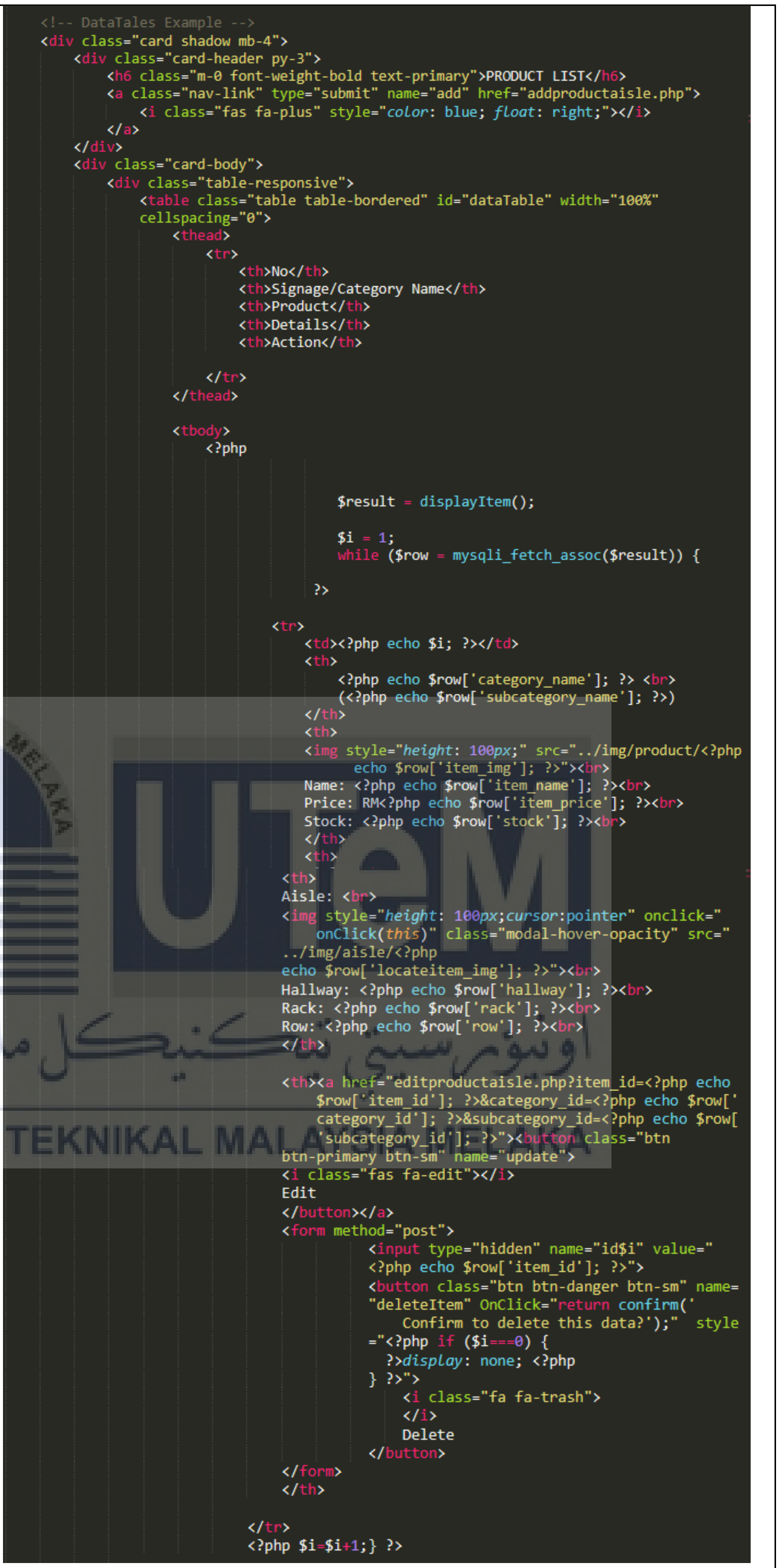
Module Name: Shop List	
Description	This module will allow shopper to add their product item into a list and the system will tells their total budgeting.
Duration to complete	4 days
Date completed	17/06/2021
File size	19 KB
Implementation code	

```
<script type="text/javascript">
    function calc(number, total){

        var price =  document.getElementById("item_price"+number).value;
        var quantity =  document.getElementById("quantity"+number).value;
        var total = price * quantity;
        document.getElementById("total"+number).innerHTML = total;
        // document.getElementById("subtotal"+number).value = total;

        var totalPrice = 0;



        // Update total price
        for (var i = 1; i <= total; i++) {
            var temp = document.getElementById("total"+i).textContent;
            // totalPrice += temp;
            totalPrice = totalPrice + Number(temp);

            document.getElementById("totalPrice").innerHTML = "RM " + totalPrice;
        }
    }
</script>
```

Table 5.4 Admin Management

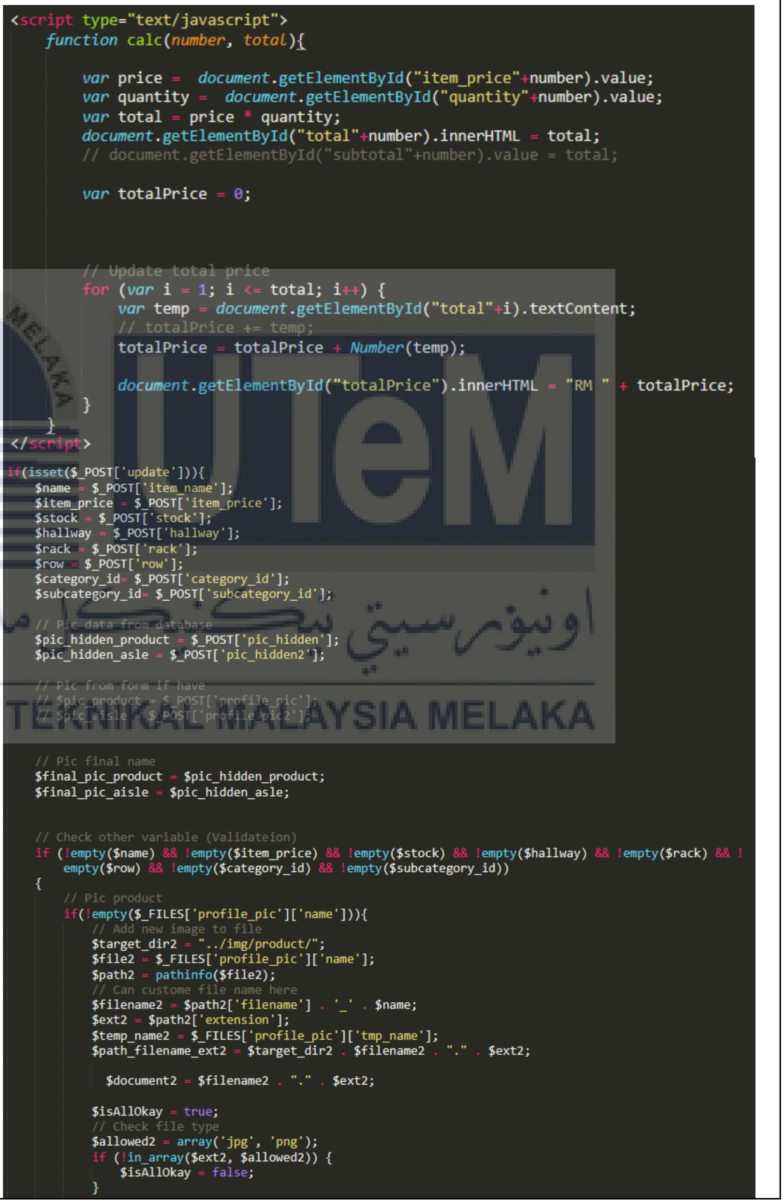
Module Name: Admin Management	
Description	This module will allow admin to manage admin.
Duration to complete	1 week
Date completed	24/08/2021
File size	18 KB
Implementation code	• Add admin  • Manage product

```php
if(isset($_POST['add']))
  {

      $admin_name = $_POST['admin_name'];
      $admin_email = $_POST['admin_email'];
      $admin_username = $_POST['admin_username'];
      $admin_ic =$_POST['admin_ic'];
      $admin_contact =$_POST['admin_contact'];
      $admin_password = $_POST['admin_password'];

      if(preg_match("/^[A-Z][a-zA-Z -]+$/", $admin_name) === 0) {
        echo "<script>alert('Name must be from letters, dashes, spaces and must not start with dash');

              </script>";

        }
      if (strpos($admin_email, ".com") == false)
      {
        echo "<script>alert('Invalid email.');</script>";
        echo "<script>location.assign('addadmin.php');</script>";
      }

      $sql1 = "SELECT * FROM admin WHERE admin_email = '$admin_email'";
      $result1 = $conn->query($sql1);
      if ($result1->num_rows > 0)
      {
        echo "<script>alert('Email already exist.');</script>";
        echo "<script>location.assign('addadmin2.php');</script>";
      }

      if(preg_match("/^[0-9]{6}-[0-9]{2}-[0-9]{4}$/", $admin_ic) === 0){
         echo "<script>alert('Wrong IC Number Format eg: 990104-07-5555');
                  window.location.href='addadmin.php';
                  </script>";

      }

      if(preg_match("/^[0-9]{3}-[0-9]{7}$/", $admin_contact) === 0){
        if(preg_match("/^[0-9]{3}-[0-9]{8}$/", $admin_contact) === 0){
        echo "<script>alert('Wrong Phone Number Format: 012-12412345 or 012-1241234');
                  window.location.href='addadmin.php';
                  </script>";
      }

      }

     // Password must be strong
       if(preg_match("/^.*(?=.{8,})(?=.*[0-9])(?=.*[a-z])(?=.*[A-Z]).*$/", $admin_password) === 0){
          echo "<script>alert('Password must be at least 8 characters and must contain at least one
              lower case letter, one upper case letter and one digit');
                window.location.href='addadmin.php';
                </script>";
       }

       else{

            $sql = "INSERT INTO `admin`(`admin_name`, `admin_email`, `admin_contact`, `admin_username`,
              `admin_ic`, `admin_password` ) VALUES ('".$admin_name."','".$admin_email."','".$
              admin_contact."','".$admin_username."','".$admin_ic."','".$admin_password."'); ";

             $sql1 = "SELECT * FROM admin WHERE admin_username = '$admin_username'";
             $result1 = $conn->query($sql1);
             if ($result1->num_rows > 0)
             {
               echo "<script>alert('Username already exist.');</script>";
               echo "<script>location.assign('admin.php');</script>";
             }

             else if($conn->multi_query($sql) == TRUE)
            {
             echo "<script>alert('".$admin_name." are added into this system');</script>";
             echo "<script>location.assign('manageadmin.php');</script>";

            }

             else
             {
               echo"<script>alert('error');</script>";
             }
               }

  }

?>
```

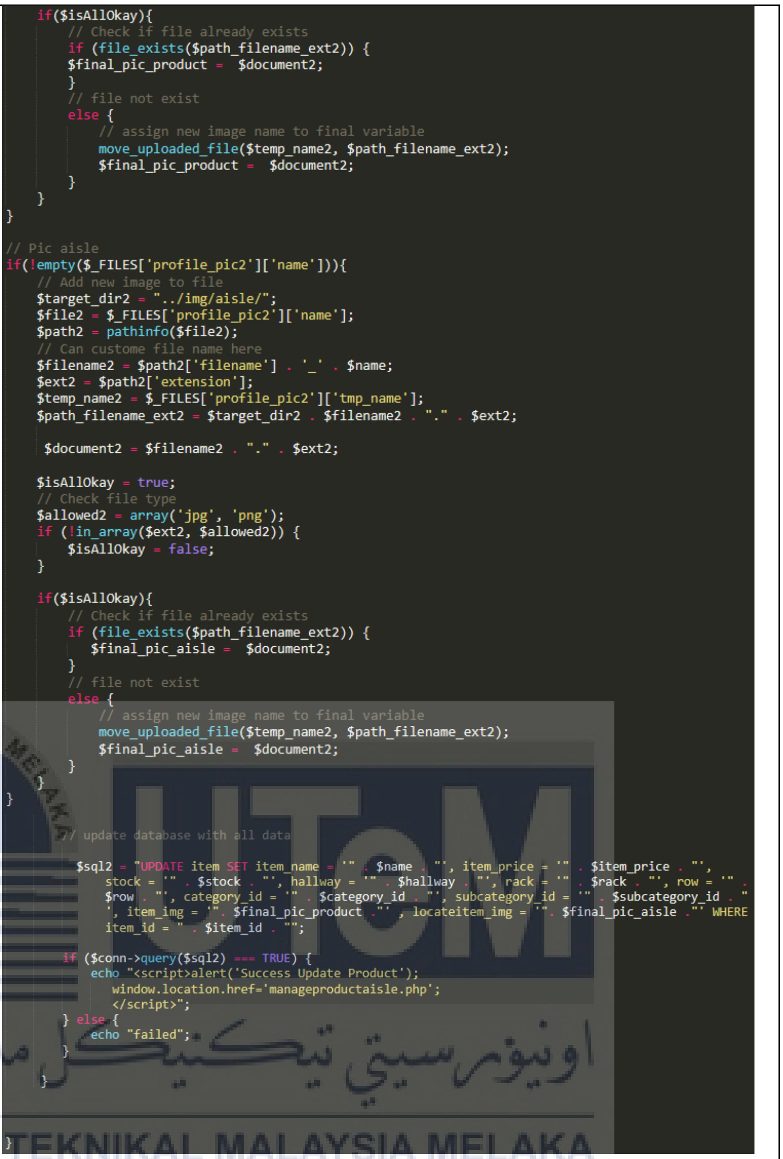


```
<!-- DataTales Example -->
<div class="card shadow mb-4">
    <div class="card-header py-3">
        <h6 class="m-0 font-weight-bold text-primary">PRODUCT LIST</h6>
        <a class="nav-link" type="submit" name="add" href="addproductaisle.php">
            <i class="fas fa-plus" style="color: blue; float: right;"></i>
        </a>
    </div>
    <div class="card-body">
        <div class="table-responsive">
            <table class="table table-bordered" id="dataTable" width="100%"
            cellspacing="0">
                <thead>
                    <tr>
                        <th>No</th>
                        <th>Signage/Category Name</th>
                        <th>Product</th>
                        <th>Details</th>
                        <th>Action</th>

                    </tr>
                </thead>

                <tbody>
                    <?php


                                    $result = displayItem();

                                    $i = 1;
                                    while ($row = mysqli_fetch_assoc($result)) {

                                 ?>

                            <tr>
                                <td><?php echo $i; ?></td>
                                <th>
                                    <?php echo $row['category_name']; ?> <br>
                                    (<?php echo $row['subcategory_name']; ?>)
                                </th>
                                <th>
                                <img style="height: 100px;" src="../img/product/<?php
                                      echo $row['item_img']; ?>"><br>
                                Name: <?php echo $row['item_name']; ?><br>
                                Price: RM<?php echo $row['item_price']; ?><br>
                                Stock: <?php echo $row['stock']; ?><br>
                                </th>
                                <th>
                             <th>
                             Aisle: <br>
                             <img style="height: 100px;cursor:pointer" onclick="
                                 onClick(this)" class="modal-hover-opacity" src="
                             ../img/aisle/<?php
                             echo $row['locateitem_img']; ?>"><br>
                             Hallway: <?php echo $row['hallway']; ?><br>
                             Rack: <?php echo $row['rack']; ?><br>
                             Row: <?php echo $row['row']; ?><br>
                             </th>

                             <th><a href="editproductaisle.php?item_id=<?php echo
                                 $row['item_id']; ?>&category_id=<?php echo $row['
                                 category_id']; ?>&subcategory_id=<?php echo $row[
                                 'subcategory_id']; ?>"><button class="btn
                             btn-primary btn-sm" name="update">
                             <i class="fas fa-edit"></i>
                             Edit
                             </button></a>
                             <form method="post">
                                       <input type="hidden" name="id$i" value="
                                       <?php echo $row['item_id']; ?>">
                                       <button class="btn btn-danger btn-sm" name=
                                       "deleteItem" OnClick="return confirm('
                                           Confirm to delete this data?');"  style
                                       ="<?php if ($i===0) {
                                         ?>display: none; <?php
                                       } ?>">
                                           <i class="fa fa-trash">
                                           </i>
                                           Delete
                                       </button>
                             </form>
                             </th>

                         </tr>
                         <?php $i=$i+1;} ?>
```

Table 5.5 Product Management

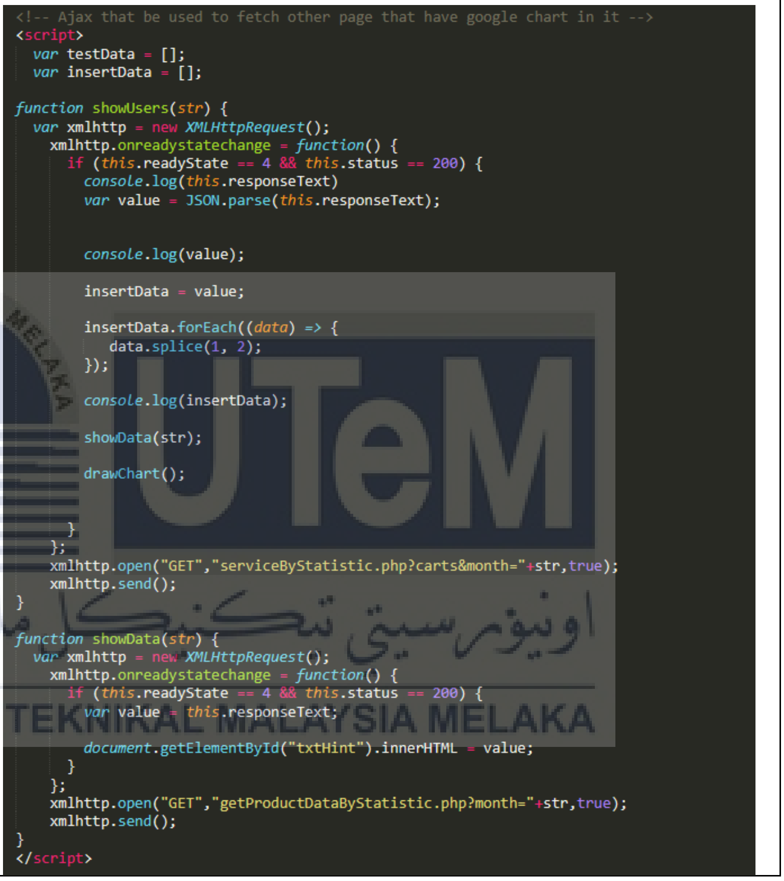
Module Name: Product Management	
Description	This module will allow admin to update any product item and their location.
Duration to complete	5 days
Date completed	24/08/2021
File size	37 KB
Implementation code	

```
<script type="text/javascript">
    function calc(number, total){

        var price =  document.getElementById("item_price"+number).value;
        var quantity =  document.getElementById("quantity"+number).value;
        var total = price * quantity;
        document.getElementById("total"+number).innerHTML = total;
        // document.getElementById("subtotal"+number).value = total;

        var totalPrice = 0;


        // Update total price
        for (var i = 1; i <= total; i++) {
            var temp = document.getElementById("total"+i).textContent;
            // totalPrice += temp;
            totalPrice = totalPrice + Number(temp);

            document.getElementById("totalPrice").innerHTML = "RM " + totalPrice;
        }
    }
</script>
if(isset($_POST['update'])){
    $name = $_POST['item_name'];
    $item_price = $_POST['item_price'];
    $stock = $_POST['stock'];
    $hallway = $_POST['hallway'];
    $rack = $_POST['rack'];
    $row = $_POST['row'];
    $category_id= $_POST['category_id'];
    $subcategory_id= $_POST['subcategory_id'];

    // Pic data from database
    $pic_hidden_product = $_POST['pic_hidden'];
    $pic_hidden_asle = $_POST['pic_hidden2'];

    // Pic from form if have
    // $pic_product = $_POST['profile_pic'];
    // $pic_aisle = $_POST['profile_pic2'];

    // Pic final name
    $final_pic_product = $pic_hidden_product;
    $final_pic_aisle = $pic_hidden_asle;


    // Check other variable (Validateion)
    if (!empty($name) && !empty($item_price) && !empty($stock) && !empty($hallway) && !empty($rack) && !
        empty($row) && !empty($category_id) && !empty($subcategory_id))
    {
        // Pic product
        if(!empty($_FILES['profile_pic']['name'])){
            // Add new image to file
            $target_dir2 = "../img/product/";
            $file2 = $_FILES['profile_pic']['name'];
            $path2 = pathinfo($file2);
            // Can custome file name here
            $filename2 = $path2['filename'] . '_' . $name;
            $ext2 = $path2['extension'];
            $temp_name2 = $_FILES['profile_pic']['tmp_name'];
            $path_filename_ext2 = $target_dir2 . $filename2 . "." . $ext2;

              $document2 = $filename2 . "." . $ext2;

            $isAllOkay = true;
            // Check file type
            $allowed2 = array('jpg', 'png');
            if (!in_array($ext2, $allowed2)) {
                $isAllOkay = false;
            }
```

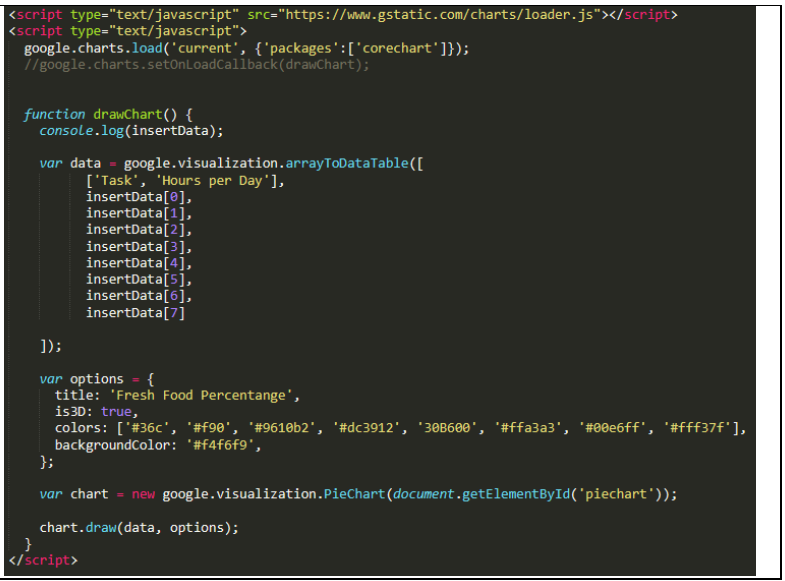


```php
    if($isAllOkay){
        // Check if file already exists
        if (file_exists($path_filename_ext2)) {
        $final_pic_product =  $document2;
        }
        // file not exist
        else {
            // assign new image name to final variable
            move_uploaded_file($temp_name2, $path_filename_ext2);
            $final_pic_product =  $document2;
        }
    }
}

// Pic aisle
if(!empty($_FILES['profile_pic2']['name'])){
    // Add new image to file
    $target_dir2 = "../img/aisle/";
    $file2 = $_FILES['profile_pic2']['name'];
    $path2 = pathinfo($file2);
    // Can custome file name here
    $filename2 = $path2['filename'] . '_' . $name;
    $ext2 = $path2['extension'];
    $temp_name2 = $_FILES['profile_pic2']['tmp_name'];
    $path_filename_ext2 = $target_dir2 . $filename2 . "." . $ext2;

     $document2 = $filename2 . "." . $ext2;

    $isAllOkay = true;
    // Check file type
    $allowed2 = array('jpg', 'png');
    if (!in_array($ext2, $allowed2)) {
        $isAllOkay = false;
    }

    if($isAllOkay){
        // Check if file already exists
        if (file_exists($path_filename_ext2)) {
           $final_pic_aisle =  $document2;
        }
        // file not exist
        else {
            // assign new image name to final variable
            move_uploaded_file($temp_name2, $path_filename_ext2);
            $final_pic_aisle =  $document2;
        }
    }
}

       // update database with all data

         $sql2 = "UPDATE item SET item_name = '" . $name . "', item_price = '" . $item_price . "',
            stock = '" . $stock . "', hallway = '" . $hallway . "', rack = '" . $rack . "', row = '" .
            $row . "', category_id = '" . $category_id . "', subcategory_id = '" . $subcategory_id . "
            ', item_img = '". $final_pic_product ."' , locateitem_img = '". $final_pic_aisle ."' WHERE
            item_id = " . $item_id . "";

       if ($conn->query($sql2) === TRUE) {
          echo "<script>alert('Success Update Product');
             window.location.href='manageproductaisle.php';
             </script>";
       } else {
          echo "failed";
       }

    }

}
```

Table 5.6 Graph Analysis

Module Name: Graph Analysis	
Description	This module will allow admin to analyze monthly product report
Duration to complete	4 days
Date completed	29/08/2021
File size	17 KB
Implementation code	

```
<!-- Ajax that be used to fetch other page that have google chart in it -->
<script>
  var testData = [];
  var insertData = [];

function showUsers(str) {
  var xmlhttp = new XMLHttpRequest();
    xmlhttp.onreadystatechange = function() {
      if (this.readyState == 4 && this.status == 200) {
        console.log(this.responseText)
        var value = JSON.parse(this.responseText);


        console.log(value);

        insertData = value;

        insertData.forEach((data) => {
           data.splice(1, 2);
        });

        console.log(insertData);

        showData(str);

        drawChart();


      }
    };
    xmlhttp.open("GET","serviceByStatistic.php?carts&month="+str,true);
    xmlhttp.send();
}

function showData(str) {
  var xmlhttp = new XMLHttpRequest();
    xmlhttp.onreadystatechange = function() {
      if (this.readyState == 4 && this.status == 200) {
        var value = this.responseText;

        document.getElementById("txtHint").innerHTML = value;
      }
    };
    xmlhttp.open("GET","getProductDataByStatistic.php?month="+str,true);
    xmlhttp.send();
}
</script>
```



```
<script type="text/javascript" src="https://www.gstatic.com/charts/loader.js"></script>
<script type="text/javascript">
  google.charts.load('current', {'packages':['corechart']});
  //google.charts.setOnLoadCallback(drawChart);


  function drawChart() {
    console.log(insertData);

    var data = google.visualization.arrayToDataTable([
          ['Task', 'Hours per Day'],
          insertData[0],
          insertData[1],
          insertData[2],
          insertData[3],
          insertData[4],
          insertData[5],
          insertData[6],
          insertData[7]

    ]);

    var options = {
      title: 'Fresh Food Percentange',
      is3D: true,
      colors: ['#36c', '#f90', '#9610b2', '#dc3912', '30B600', '#ffa3a3', '#00e6ff', '#fff37f'],
      backgroundColor: '#f4f6f9',
    };

    var chart = new google.visualization.PieChart(document.getElementById('piechart'));

    chart.draw(data, options);
  }
</script>
```

5.5 Conclusion

Finally, this chapter is required since the project integrates all of the designs from Chapter 4. This includes the architecture of the project, the relationship, and the configuration settings. After the implementation is complete, the following step will be to check. The testing phase is the stage at which all of the goals specified in the first chapter must be satisfied by the project's technique.

CHAPTER 6: TESTING

6.1 Introduction

In this chapter, testing is carried out to determine whether the system fulfils the prerequisite condition of the system state in the objective. A set of tests will be performed to ensure that there are no defects and that any errors discovered are reported and re-evaluated until the system meets its requirements. Through test environment, timetable, approach, and others will be defined in this section to indicate whether or not this system has been thoroughly tested.

6.2 Test Plan

A test plan is a document that describes the scope and operations of software testing which consists of test organization, test environment and test schedule. A test plan specifies the procedure to be followed in order to ensure that a product or system satisfies its design criteria and other requirements.

6.2.1 Test Organization

Test organization explain about the personnel involved in testing SAFSS. Each of test is tested by different personnel as shown below in Table 6.1.

Table 6.1: Test Organization

Testing Activity	Testing Member
Unit Testing	Ts. Dr. Ummi Rabaah Binti Hashim
Integration Testing	Ts. Dr. Ummi Rabaah Binti Hashim
System Testing	Ts. Dr. Ummi Rabaah Binti Hashim
User Acceptance Testing	Nor Hanis Afifah Binti Harun

6.2.2 Test Environment

Test environment is a setup of hardware and software in order to execute test cases.

Table 6.2: Test Environment

Software/Hardware Tools	Specifications
Operating System	Windows 10 Home Single Language
Memory Capacity	4GB RAM or higher
Processor Type and Speed	Intel® Core ™ I5-4200U CPU @ 1.60GHz 2.30GHz
Web Server	XAMPP
Web Browser	Google Chrome
Database	MySQL

6.2.3 Test Schedule

Testing activities and it cycles must be documented. Test schedule documents all system testing and evaluation of the system. Basically, there are four type of tests documented in this test schedule; Unit Testing, Integration Testing, System Testing and User Acceptance Testing.

Table 6.3: Test Schedule

Testing Type	Description	Start Date	End Date
Unit Testing	Ensure that the system is correctly coded and performs functionalities of prerequisites.	1 August 2021	6 August 2021
Integration Testing	Testing the subsystem interface.	8 August 2021	13 August 2021
System Testing	Ensure the system smoothly function	15 August 2021	20 August 2021

	and satisfy all of its prerequisites.		
User Acceptance Testing	Ensure the system is ready to be used by the end user.	22 August 2021	27 August 2021

6.3 Test Strategy

Test strategy is one of the most essential documents in software testing. It defines the project's testing strategy and describes how the testing process should be carried out in order to accomplish certain test goals and a shared aim in delivery quality. There are no bottom-up or top-down approach that use in this project due to whole project is not doing in an organization, besides doing it alone. The testing method utilized in this project is black-box testing. Black-box testing is a type of behavior testing that focuses on the inputs and outputs without knowing the internal code implementation. The generic steps to carry out are chooses valid inputs which are positive test scenario to check whether System Under Test(SUT) processes them correctly. Also, some invalid inputs which are negative test scenario are chosen to verify that the SUT is able to detect it. Then, determines the expected outputs for all those inputs where compares the actual outputs with the expected outputs. Any issues discovered will be addressed and re-tested.

6.3.1 Classes of Test

There are four classes of tests which consists of unit testing, integration testing, system testing and user acceptance testing. Each of classes of test will be explained in detail below.

6.3.1.1 Unit Testing

Unit testing validates individual units of source code to see whether it is working properly or vice versa. Any flaws or bugs discovered during this testing will be corrected before each module is combined into a complete system.

6.3.1.2 Integration Testing

Integration testing is a test of integrated modules, where it merged in order to integrate and function together. Integration testing is done between error handling processes that are linked for one error to another. Users of this system, for example, must successfully login in order to access all system features such as seeing the dashboard and searching product item's location details. The system that is developed must be linked with one page to another page.

6.3.1.3 System Testing

System testing involves testing the entire system from the beginning, which is the login page till the end. Each module is tested in all areas, including functional and non-functional criteria, to ensure that the system is ready to deploy and works properly.

6.3.1.4 User Acceptance Testing

Acceptance testing is the process through which users test the system to ensure that it meets all criteria and is acceptable to the users. End users in the intended environment carry it out. Acceptance testing is carried out utilizing a subset of the studies that were utilized during system testing.

6.4 Test Design

Test data and test description will be discussed in detail in this section.

6.4.1 Test Description

Identified test cases and expected results are designed and documented as shown in Table 6.4 below.

Table 6.4: Test Description

Test Case ID	TC001		
Description	To evaluate shopper registration functionality		
Module	Authentication		
Prepared By	Nurhidayah Mohd Lazim	Date Prepared	09/08/2021

Tested by: Date Tested:					
ID	Test Scenario	Test Data	Expected Result	Actual Result	Status
TC001_01	Fill in all fields in the form with correct format	Username: Nurhidayah Phone Number: 016-2802409 Email: 4yyashi@gmail.com Password: Dayah@123 Retype Password: Dayah@123	An alert message "Hi, Nurhidayah. Your registration were successful" will be display		Pass
TC001_02	Fill in all fields in the form except one field	Username: Syabil Phone Number: 019-9009909 Email: (null) Password: Syabil10@123 Retype Password: Syabil10@123	An alert message "Enter valid email" will be display		Pass
TC001_03	Fill in all fields in the form with the same	Username: Nurhidayah Phone Number: 018-2302501 Email: nurhid4yahmohdlazim@gmail.com	A Pop up message "Username already		Pass

	username with others shopper	Password: Hidayah@111 Retype Password: Hidayah@111	exist." will be display		
TC001_04	Fill in all fields in the form with the same email with others shopper	Username: Ikrimah Phone Number: 014-4833202 Email: 4yyashi@gmail.com Password: Ikrimah@123 Retype Password: Ikrimah@123	A Pop up message "Email already exist." will be display		Pass
TC001_05	Fill in all fields in the form with wrong phone number format	Username: Ikrimah Phone Number: 0144833202 Email: 4yyashi@gmail.com Password: Ikrimah@123 Retype Password: Ikrimah@123	A Pop up message "Wrong Phone Number Format: 012-12412345 or 012-1241234." will be display		Pass
TC001_06	Fill in all fields in the form but retype password not match with password.	Username: Ikrimah Phone Number: 0144833202 Email: 4yyashi@gmail.com Password: Ikrimah@123 Retype Password: Ikrimah23	A Pop up message "Password do not match." will be display		Pass

Test Case ID	TC002		
Description	To evaluate shopper login functionality		
Module	Authentication		
Prepared By	Nurhidayah Mohd Lazim	**Date Prepared**	09/08/2021

Tested by:

Date Tested:

ID	**Test Scenario**	**Test Data**	**Expected Result**	**Actual Result**	**Status**
TC002_01	Enter a valid username and password	Username: Ikrimah Password: Ikrimah@123	Successfully login into the system	Welcome to Smart Aid!	Pass
TC002_02	Enter a valid username and an invalid password	Username: Ikrimah Password: abc1234	A Pop up message " Invalid Username or password" will be display		Pass
TC002_03	Enter an invalid username and valid password	Username: Akrimah Password: Ikrimah@123	A Pop up message " Invalid Username or password" will be display		Pass

TC002_04	Enter an invalid username and invalid password	Username: Akrimah Password: Akrimah@123	A Pop up message " Invalid Username or password" will be display		Pass

Test Case ID	TC004		
Description	To provide shopper find the exact product's location		
Module	Search Product Location		
Prepared By	Nurhidayah Mohd Lazim	**Date Prepared**	09/08/2021

Tested by:

Date Tested:

ID	**Test Scenario**	**Test Data**	**Expected Result**	**Actual Result**	**Status**
TC004_01	Enter any product item name in search bar	Search: Egg	Successfully show all various type of Egg's brand and the aisle image, hallway, rack and row number. Also, successfully		Pass

			see aisle image in a big picture after clicked it.		
TC004_02	Enter any product item name in search bar	Search: Gardenia	Successfully show all various type of Gardenia bread's brand		Pass
TC004_03	Enter any category name	Search: Baby	Successfully show all baby's brand		Pass

Test Case ID	TC005		
Description	To allow shopper add product item into shop list		
Module	Shop List		
Prepared By	Nurhidayah Mohd Lazim	**Date Prepared**	09/08/2021

Tested by: **Date Tested:**					
ID	**Test Scenario**	**Test Data**	**Expected Result**	**Actual Result**	**Status**
TC005_01	Click button "Add To List"	-	Successfully added selected product item by shopper		Pass

			into a shop list.		
TC005_02	Click List menu	-	Successfully list all product item that have been selected by shopper		Pass
TC005_03	Get total spending by input a quantity in the box provided	Product item: Gardenia Wholegrain with Sunflower Seeds Quantity: 4	Successfully calculate total spending for shopper		Pass
TC005_04	Click "Remove" button on any selected product item	-	An alert message "Confirm to remove this item?" will be display and click "OK" then a message "Successfully Removed" will be display.		Pass

Test Case ID	TC006		
Description	To create, update, and delete admin partner		
Module	Admin Management		
Prepared By	Nurhidayah Mohd Lazim	**Date Prepared**	09/08/2021

Tested by:

Date Tested:

ID	**Test Scenario**	**Test Data**	**Expected Result**	**Actual Result**	**Status**
TC006_01	Fill in all fields in the form	Name: Nur Arissa Identity Card Number: 980101-04-5875 Contact Number: 012-6463656 Email: arrisa@gmail.com Username: Arrisa Password: Arissa@123	An alert message "Nur Arissa are added into the system" will be display		Pass
TC006_02	Fill in all fields in the form except one field	Name: (null) Identity Card Number: 980101-04-5875 Contact Number: 012-6463656 Email: arrisa@gmail.com Username: Arrisa	A Pop up message " Please fill out this field" will be display		Pass

		Password: Arissa@123			
TC006_03	Fill in all fields in the form with invalid name format	Name: dddafe45 Identity Card Number: 980101-04-5875 Contact Number: 012-6463656 Email: arrisa@gmail.com Username: Arrisa Password: Arissa@123	A Pop up message "Name must be from letters, dashes, spaces and must not start with dash" will be display		Pass
TC006_04	Fill in all fields in the form with invalid email format	Name: Nur Arissa Identity Card Number: 980101-04-5875 Contact Number: 012-6463656 Email: arrisa@gmail Username: Arrisa Password: Arissa@123	A Pop up message "Invalid email" will be display		Pass
TC006_05	Fill in all fields in the form with existed email	Name: Nur Arissa Identity Card Number: 980101-04-5875 Contact Number: 012-6463656	A Pop up message "Email already exist." will be display		Pass

		Email: ssyabil71@gmail.com Username: Arrisa Password: Arissa@123			
TC006_06	Fill in all fields in the form with invalid identity card format	Name: Nur Arissa Identity Card Number: 980101045875 Contact Number: 012-6463656 Email: arrisa@gmail.com Username: Arrisa Password: Arissa@123	A Pop up message "Wrong IC Number Format eg: 990104-07-5555" will be display		Pass
TC006_07	Fill in all fields in the form with invalid phone number format	Name: Nur Arissa Identity Card Number: 980101-04-5875 Contact Number: 01264636568 Email: arrisa@gmail.com Username: Arrisa Password: Arissa@123	A Pop up message "Wrong Phone Number Format: 012-12412345 or 012-1241234" will be display		Pass
TC006_08	Update name	Name: Muhd Saifulllah	An alert message "Success		Pass

		Identity Card Number: 970312-01-3421 Contact Number: 012-6463656 Email:	update Admin" will be display		
TC006_09	Click the delete button that is parallel to the account name which admin want to remove	-	An alert message "Confirm to delete this data?" will be display and click "OK" then a message "Successfully Delete" will be display.		Pass

Test Case ID	TC007		
Description	To evaluate product registration functionality		
Module	Product Management		
Prepared By	Nurhidayah Mohd Lazim	**Date Prepared**	09/08/2021

Tested by: **Date Tested:**					
ID	**Test Scenario**	**Test Data**	**Expected Result**	**Actual Result**	**Status**

TC007_01	Fill in all fields in the form	Product Name: Gardenia Original Classic Jumbo White Bread 600g Product Price: 4.00 Stock: 50 Hallway: A1 Rack: 2 Row: 1 Category: Fresh Food Subcategory: Bakery Product Image: gardeniasaver.png Aisle Image: aisle1.jpg	An alert message "Gardenia Original Classic Jumbo White Bread 600g successfully added into the system" will be display		Pass
TC007_02	Fill in all fields in the form except one field	Product Name: (null) Product Price: 14.99 Stock: 30 Hallway: P1 Rack: 3 Row: 1 Category: Pets Subcategory: Cat Care Product Image: whiskasdryfood.png Aisle Image: aisle4.jpg	A Pop up message " Please fill out this field" will be display		Pass

TC007_03	Fill in all fields in the form but with the same product name	Product Name: Gardenia Original Classic Jumbo White Bread 600g Product Price: 4.00 Stock: 50 Hallway: A1 Rack: 2 Row: 1 Category: Fresh Food Subcategory: Bakery Product Image: gardeniasaver.png Aisle Image: aisle1.jpg	A Pop up message "Product item already exist" will be display		Pass

Test Case ID	TC008		
Description	To analyze monthly product report		
Module	Graph Analysis		
Prepared By	Nurhidayah Mohd Lazim	**Date Prepared**	09/08/2021

Tested by: **Date Tested:**					
ID	**Test Scenario**	**Test Data**	**Expected Result**	**Actual Result**	**Status**

TC008_01	Click Fresh Food ‘Report’ button and select month	Month: August	Percentage of the Fresh Food Category and the details will be display		Pass

6.4.2 Test Data

Test data contain field of modules and test data which is valid data and invalid data when testing the full system. The summary of test data for testing cycle 1 and cycle 2 are as shown below.

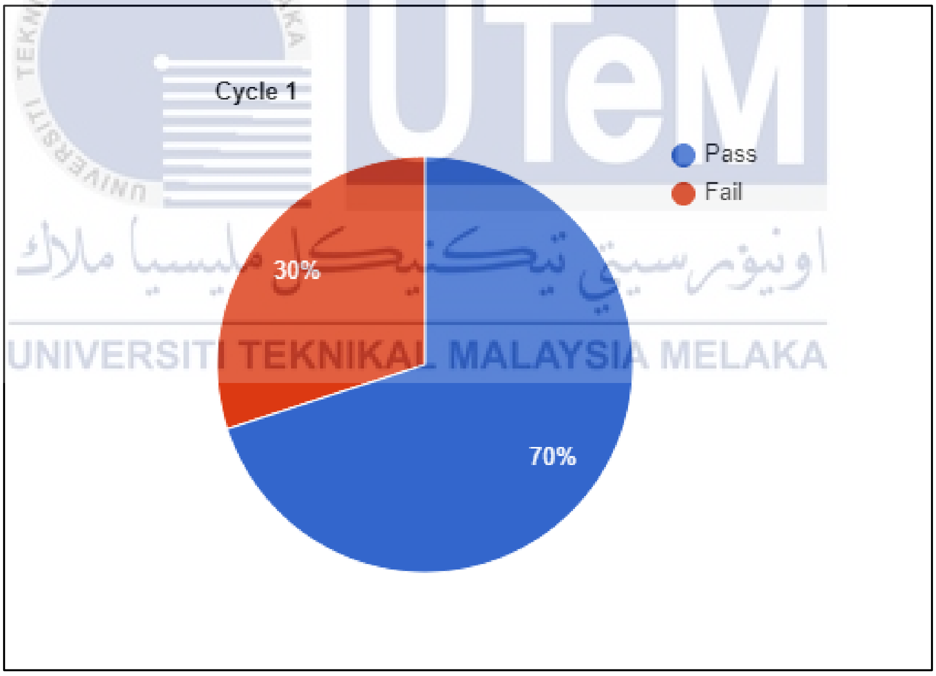


Figure 6.1: Testing Cycle 1

The test results of testing cycle 1 are shown in Figure 6.1 above. It demonstrates that 70 percent of the tests pass the first cycle, whereas 30 percent fail.

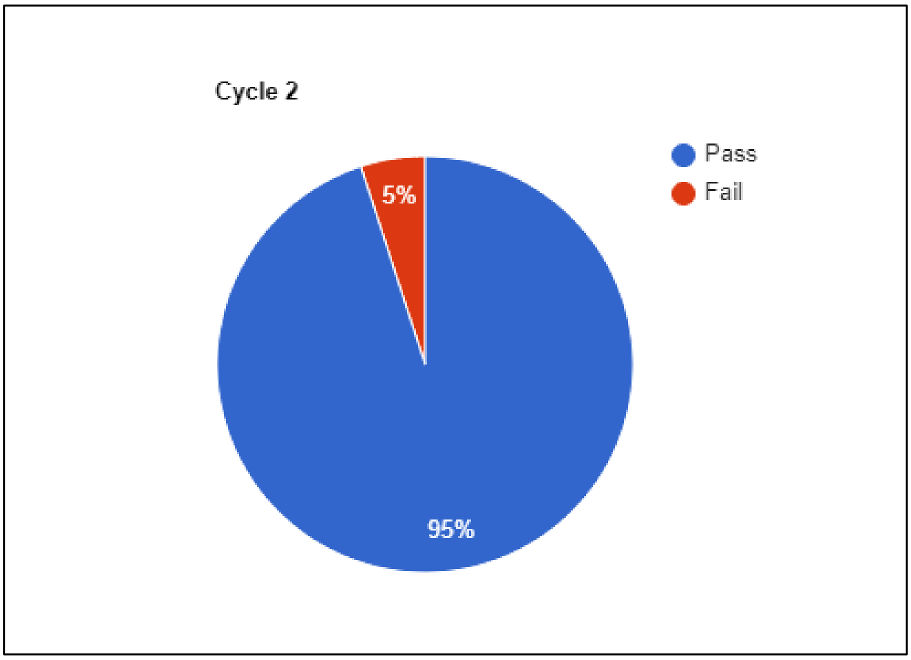


Figure 6.2: Testing Cycle 2

As shown in Figure 6.2 above is the result of second cycle of testing. In comparison to the first cycle of testing, the proportion of pass results has increased from 70 percent to 95 percent, while the percentage of fail results has decreased from 30 percent to 5 percent. Problems discovered during the first round of testing are rectified, and the system is enhanced.

6.5 Test Result and Analysis

The outcomes of each test case are reported in this section. Each test result analysis described includes the following elements: test case ID, description, and test results. The pass status of a test shows that the test result was successful, whereas the fail status indicates that the test failed. If a test fails, the cause of the failure is recorded.

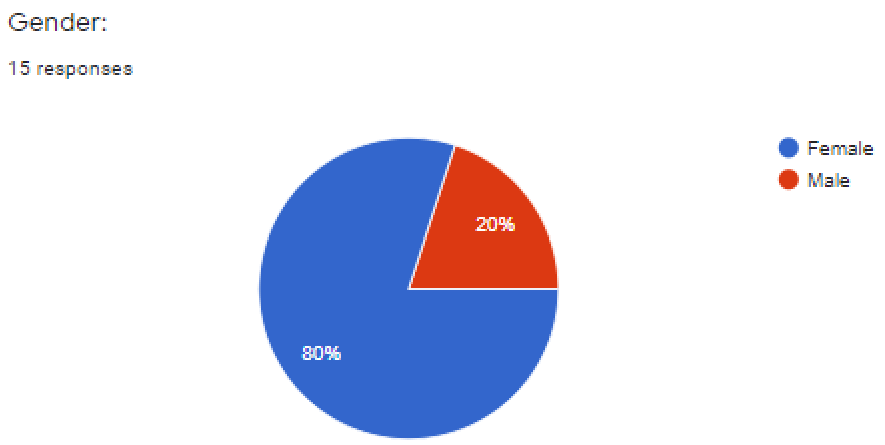


Figure 6.1 Gender responses

Figure 6.1 represents 80% of respondents are female while 20% of respondents are male that involved in user acceptance questionnaire.

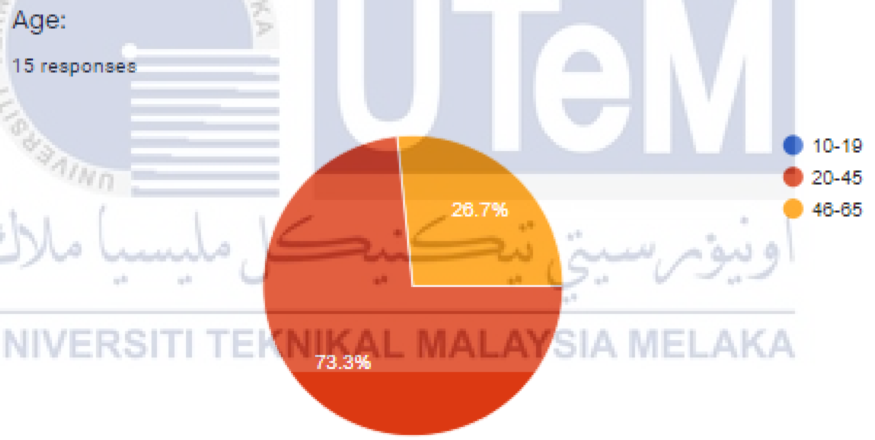


Figure 6.2 Age responses

Figure 6.2 represents 73.3% of 20 to 45 years old of respondents. 26.7% of respondents are range 46 to 65 years old.

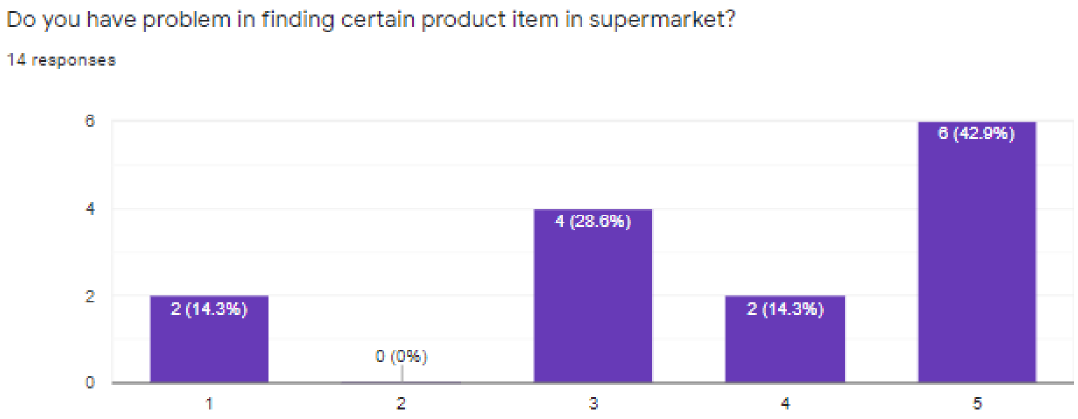


Figure 6.3 Percentage of having problem

Figure 6.3 shows 42.9% agree on having problem in finding certain product item in supermarket.

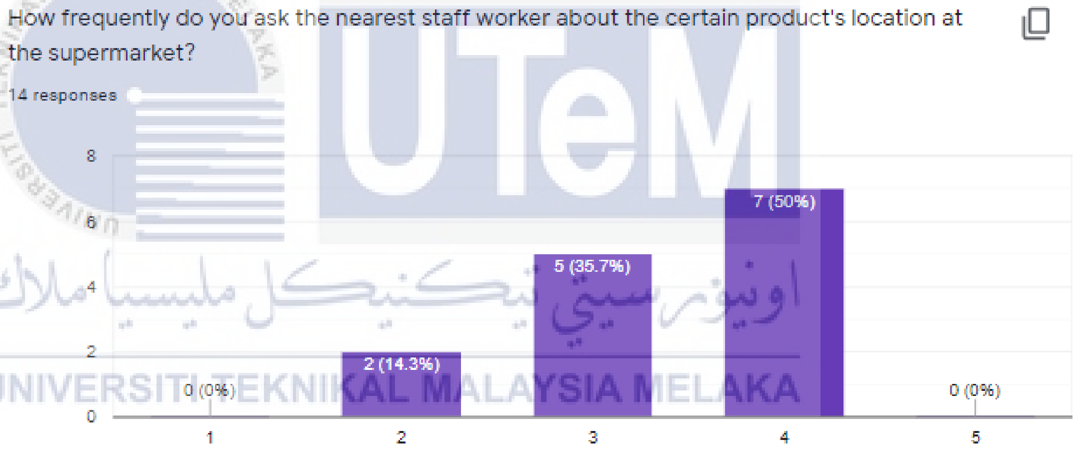


Figure 6.4 Percentage of frequently ask

Figure 6.4 shows 50% agree on frequently ask the nearest staff worker about the certain product’s location at the supermarket.

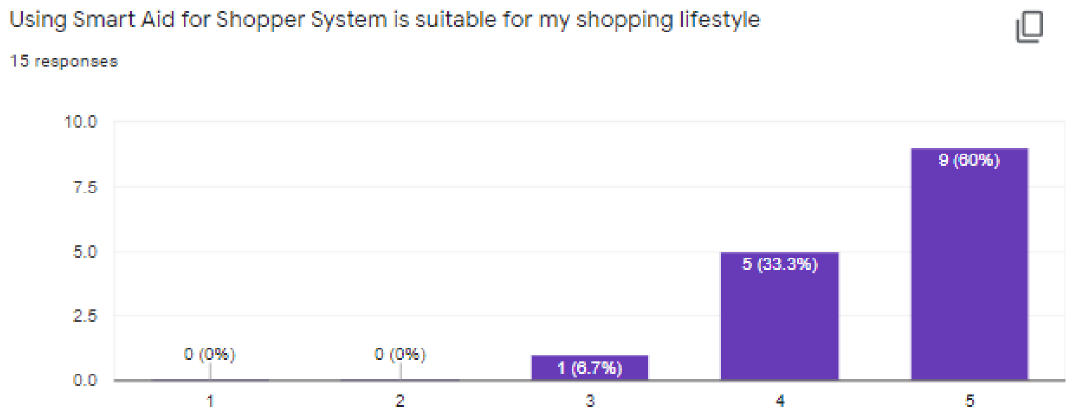


Figure 6.5 Percentage of agree on using SAFSS

Figure 6.5 shows 60% agree by using Smart Aid for Shopper System is suitable for shopping lifestyle.

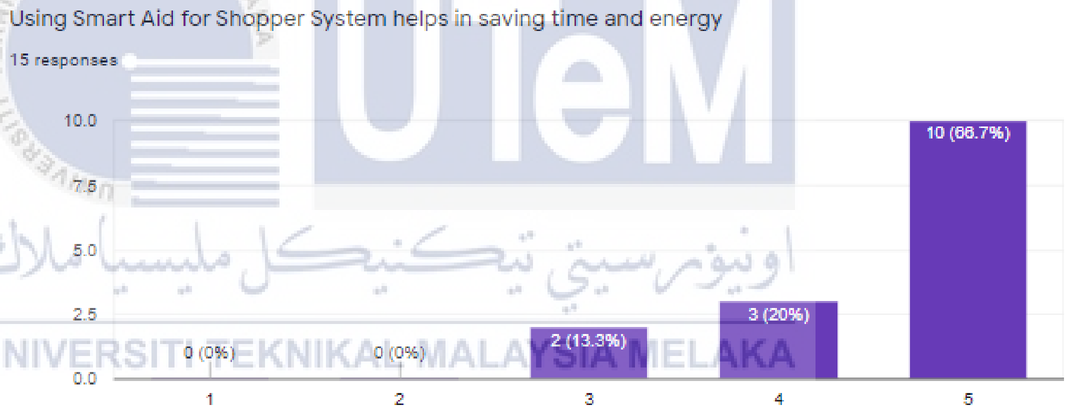


Figure 6.6 Percentage of SAFSS in time

Figure 6.6 shows 66.7% agree on by using Smart Aid for Shopper System will help shopper saving time and energy.

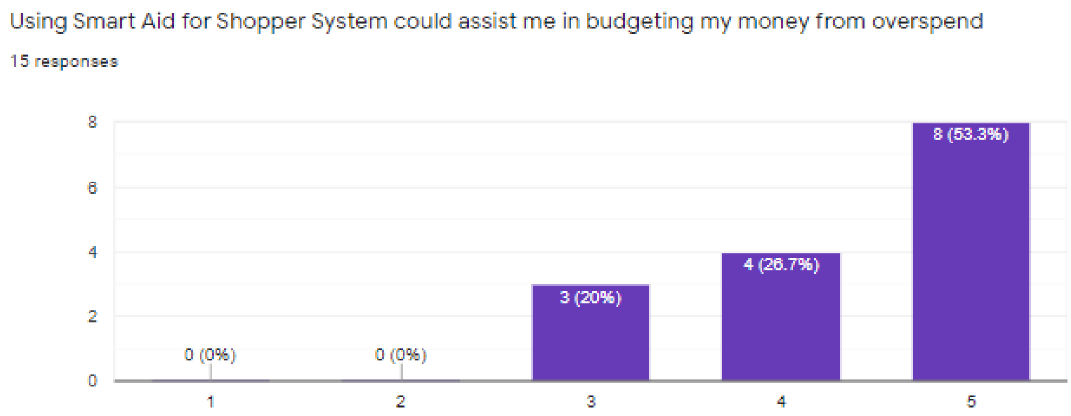


Figure 6.7 Percentage of assist from overspend

Figure 6.7 shows 53.3% agree on by using Smart Aid for Shopper System could assist in budgeting money from overspend at supermarket.

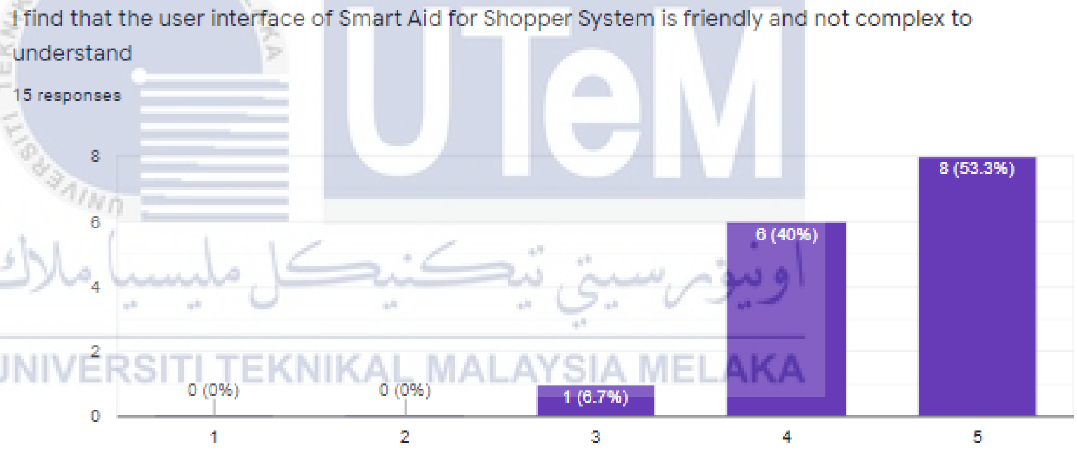


Figure 6.8 Percentage of friendly user interface

Figure 6.8 shows 53.3% agree that user interface of Smart Aid for Shopper System is friendly and not complex to understand.

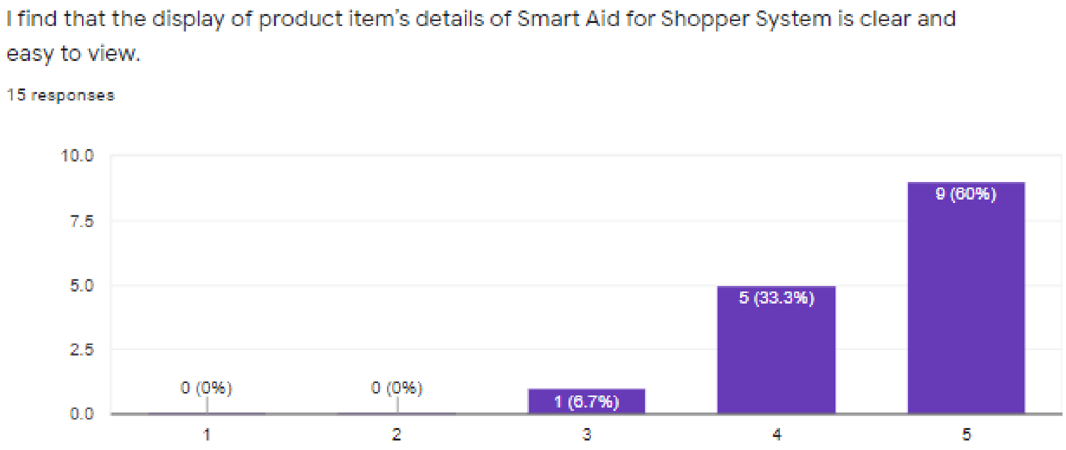


Figure 6.9 Percentage of display product item

Figure 6.9 shows 60% of respondents find that the display of product item's details of SAFSS is clear and easy to view.

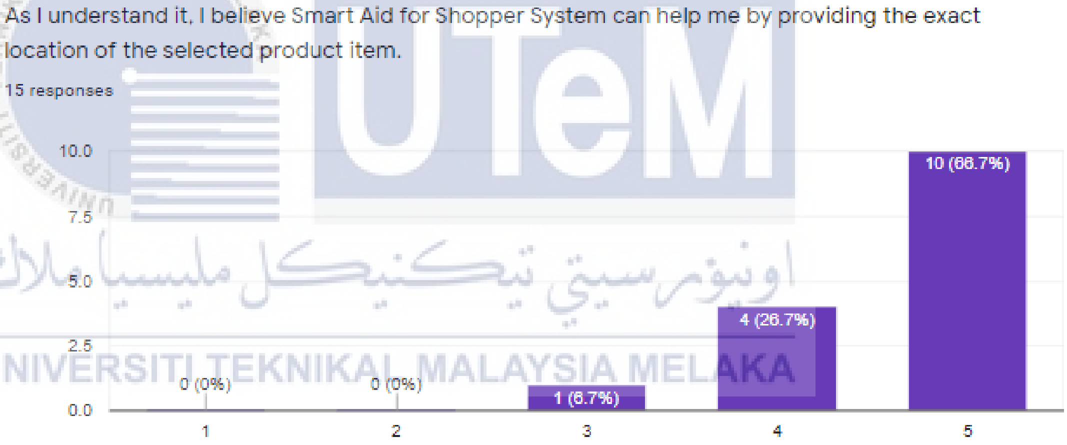


Figure 6.10 Percentage of respondents believe

Figure 6.10 shows 66.7% of respondents believe Smart Aid for Shopper System could help in providing the exact location of the selected product item.

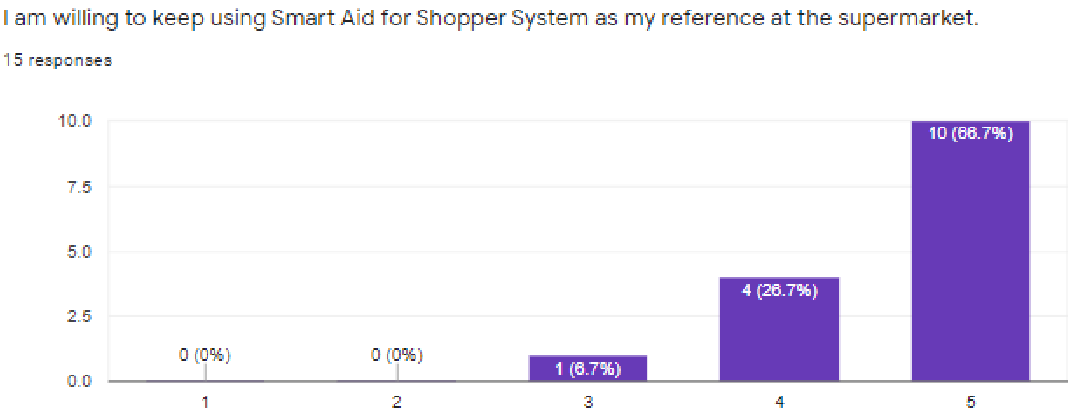


Figure 6.11 Percentage of using SAFSS

Figure 6.11 shows 66.7% of respondents willing to keep Smart Aid for Shopper System as reference at the supermarket.

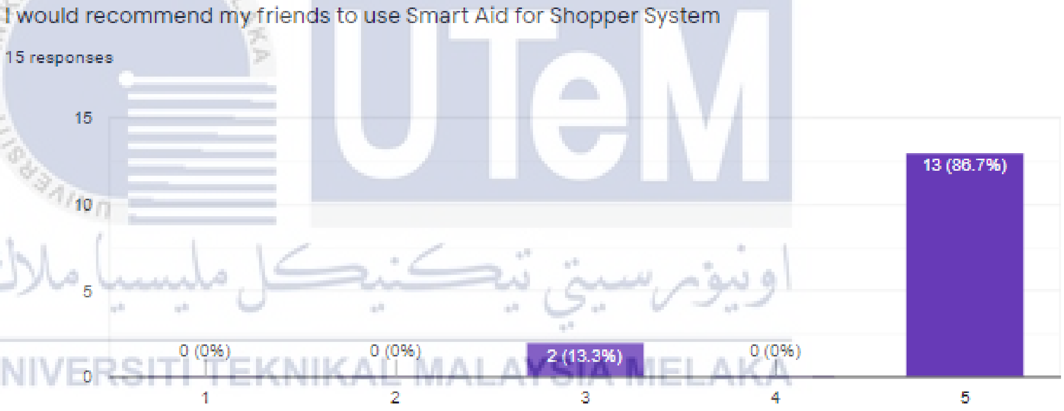


Figure 6.12 Percentage of respondents recommended

Figure 6.12 shows 86.7% of respondents would recommend Smart Aid for Shopper System to their acquaintances.

6.6 Conclusion

This chapter describes the project's testing strategy and identifies test cases. Testing activities and results have been documented in order for the system to improve and achieve the required objectives and functionality. The next chapter will provide observations on the system's strengths and flaws, as well as recommendations for future improvements.

CHAPTER 7: CONCLUSION

7.1 Introduction

This chapter will summarize the entire system that has been built in terms of its strengths and flaws, future improvement recommendations, and project contribution.

7.2 Observation on Weaknesses and Strengths

Weaknesses and strengths of the project system are described as below.

7.2.1 Weaknesses

The project system is lack of security since it not encrypted in the database where leaking of information is still possible. There is no feedback where shopper can leave any comment. Lastly, there is no product's report for every day and year that admin can analysis.

7.2.2 Strengths

Smart Aid for Shopper System is easily to access and a friendly system. This project system is available to users at all times. While on their way to supermarket with their vehicle, shopper can explore more the system, get a strategy how they spent well on the supermarket by adding the selected product item into the shop list. In addition, admin can manage

7.3 Propositions for Improvement

This system will be better if it incorporated navigation, such as a GPS system, which is one of the required suggestions because it greatly aids in properly directing shopper to the product's location.

Other than that, a barcode scanner that can track the product item's price in an easier and time-saving manner for shoppers is a highly useful tool.

Furthermore, transaction activity is beneficial since it provides the most impact, as the shopper could scans the products and the system continues to count the number of goods, and they only have to pay and complete the session.

7.4 Project Contribution

This project system is quite useful for usage in any supermarket or store for individual shopper. It gives benefits for shopper in terms of shopping services which it helps in saving time and shopper wiser in estimating the amount of goods that will be spent. Also, it convenient for the admin to manage the system in a more organized way.

7.5 Conclusion

In conclusion, the project system has been developed and meets the objective conclusively due to all the phases from planning phase to testing phase are completely carried out throughout the whole project. However, there is still a lot of potential for improvement in terms of usefulness in the future. With the recommended improvements and project flaws outlined in this chapter, it is reasonable to expect that Smart Aid for Shopper System can be enhanced and contribute as a much better in side of shopping services.

BIBLIOGRAPHY

CODEBOTS. (2021). Retrieved from What is Rapid Application Development (RAD)?: https://codebots.com/app-development/what-is-rapid-application-development-rad

Guru99. (2021). Retrieved from What is a Functional Requirement in Software Engineering?: https://www.guru99.com/functional-requirement-specification-example.html

Lexico. (2021). *Oxford Lexico*. Retrieved from Definition: https://www.lexico.com/definition/shopping_service

Sisney, L. (n.d.). *Organizational Physics*. Retrieved from The Difference Between Organizational Structure and Organizational Chart: https://organizationalphysics.com/2015/02/09/organizational-design-the-difference-between-organizational-structure-and-an-org-chart/

Super Cowboy. (2004). Retrieved from Super Cowboy Online Store: https://www.supercowboy.com.my/melaka-branch

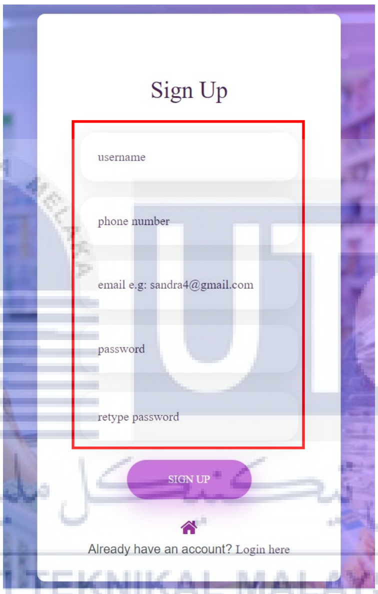
TESCO. (2021). Retrieved from Shop Online: https://www.tesco.com.my/

Visual Paradigm. (n.d.). Retrieved from Conceptual, Logical and Physical Data Model Design: https://www.visual-paradigm.com/support/documents/vpuserguide/3563/3564/85378_conceptual,l.html

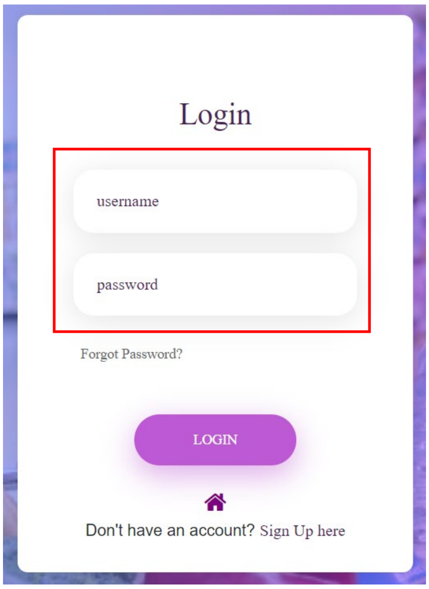
APPENDIX A

User Manual

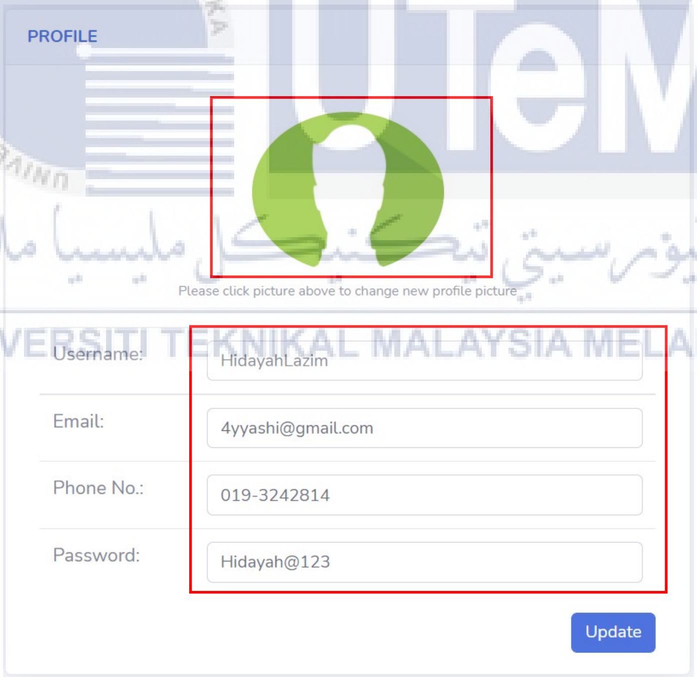
1. For a first time user, you need to sign up first to log into the system.


Sign Up
username
phone number
email e.g: sandra4@gmail.com
password
retype password
SIGN UP
Already have an account? Login here

2. After successfully sign up, you need to log in using your username and password.



3. It’s necessary if you want to update your profile picture and edit other details.



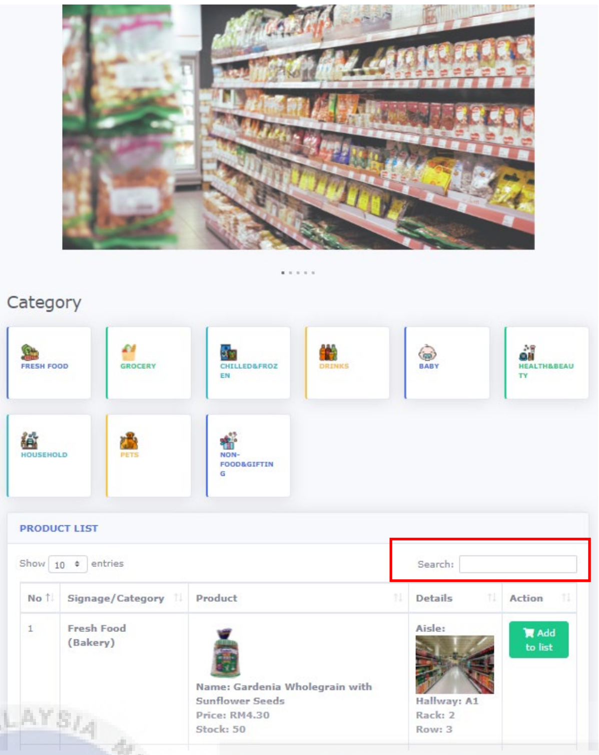
Username: HidayahLazim

Email: 4yyashi@gmail.com

Phone No.: 019-3242814

Password: Hidayah@123

4. There are two choices where you can find your product item. First at the home site, at the bottom of category you will find a search bar, there you can type any product you want.



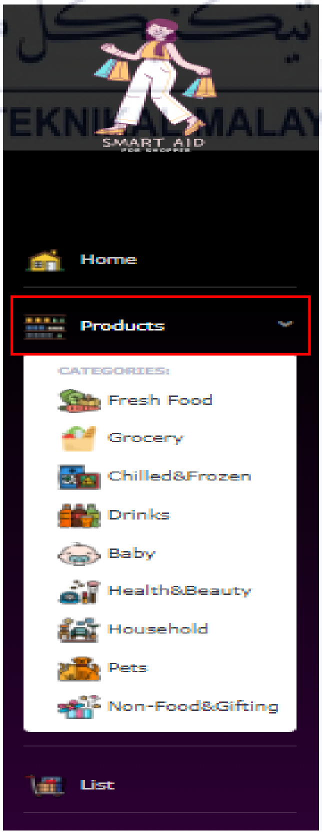
PRODUCT LIST

Show 10 entries

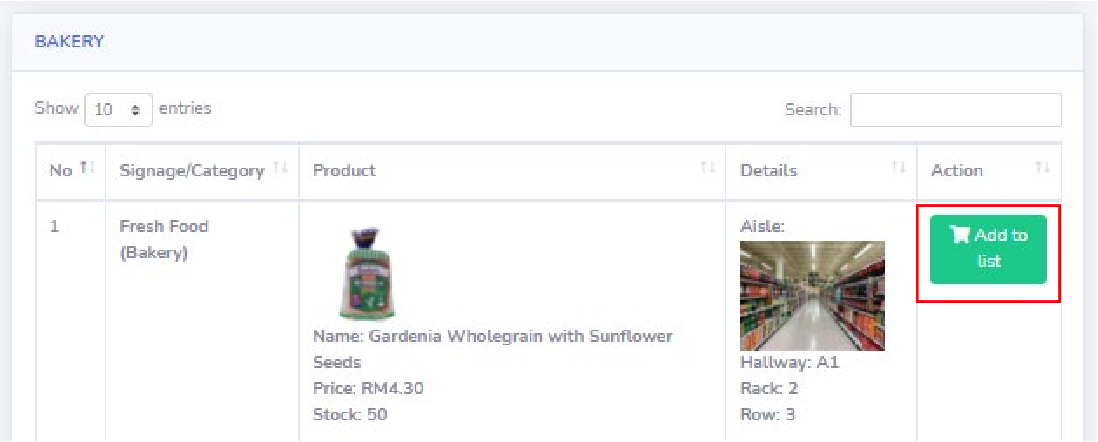
Search:

No	Signage/Category	Product	Details	Action
1	Fresh Food (Bakery)	Name: Gardenia Wholegrain with Sunflower Seeds Price: RM4.30 Stock: 50	Aisle: Hallway: A1 Rack: 2 Row: 3	Add to list

Second choice, is at the navigation bar called as 'Product', where it consists of various type of category. Select any category you want, then click 'Shop Now'. The system will display product item in their category.



5. To calculate your total spending, click ‘Add to List’ button at the product item. Then the system will successfully add your chosen into the list menu.

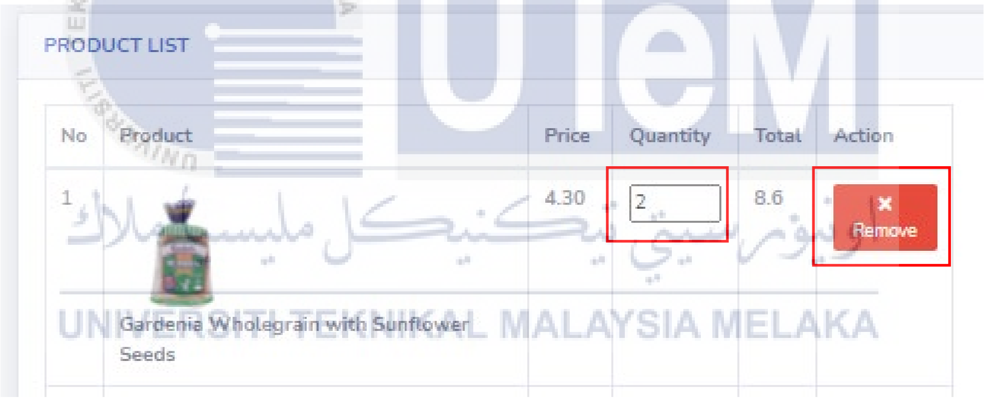


BAKERY

Show 10 entries Search:

No	Signage/Category	Product	Details	Action
1	Fresh Food (Bakery)	Name: Gardenia Wholegrain with Sunflower Seeds Price: RM4.30 Stock: 50	Aisle: Hallway: A1 Rack: 2 Row: 3	Add to list

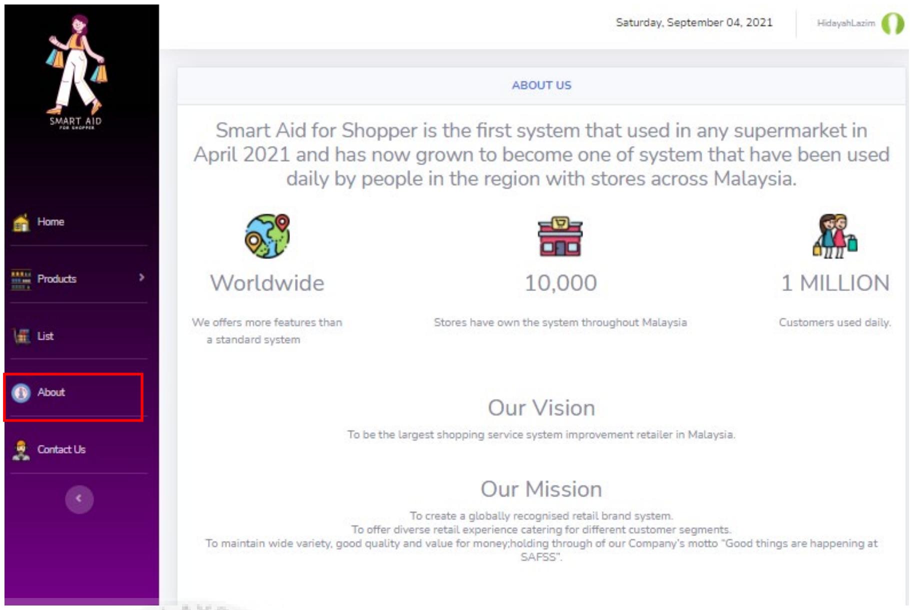
6. At the list menu, you can add or decrease the quantity. Also you are allowing to remove the item you don’t want.



PRODUCT LIST

No	Product	Price	Quantity	Total	Action
1	Gardenia Wholegrain with Sunflower Seeds	4.30	2	8.6	✖ Remove

7. To know about our system, click ‘About’ to acknowledge more.



Saturday, September 04, 2021

ABOUT US

Smart Aid for Shopper is the first system that used in any supermarket in April 2021 and has now grown to become one of system that have been used daily by people in the region with stores across Malaysia.

Worldwide

We offers more features than a standard system

10,000

Stores have own the system throughout Malaysia

1 MILLION

Customers used daily.

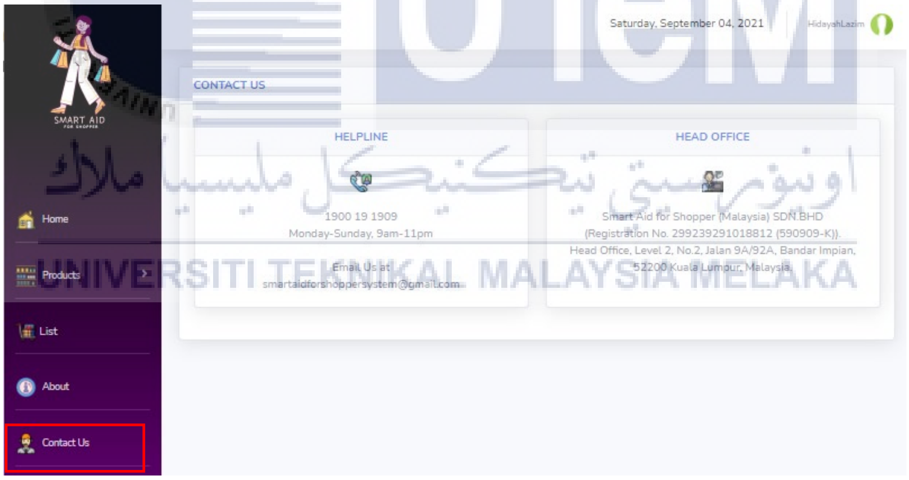
Our Vision

To be the largest shopping service system improvement retailer in Malaysia.

Our Mission

To create a globally recognised retail brand system.
To offer diverse retail experience catering for different customer segments.
To maintain wide variety, good quality and value for money;holding through of our Company's motto "Good things are happening at SAFSS".

8. To contact us, click ‘Contact Us’ at navigation bar.



Saturday, September 04, 2021

CONTACT US

HELPLINE

1900 19 1909
Monday-Sunday, 9am-11pm

Email Us at
smartaidforshoppersystem@gmail.com

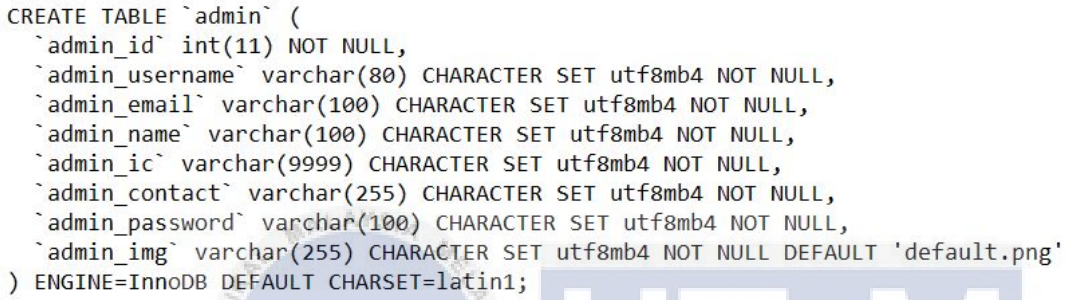
HEAD OFFICE

Smart Aid for Shopper (Malaysia) SDN.BHD
(Registration No. 299239291018812 (590909-K)).
Head Office, Level 2, No.2, Jalan 9A/92A, Bandar Impian,
52200 Kuala Lumpur, Malaysia.

APPENDIX B

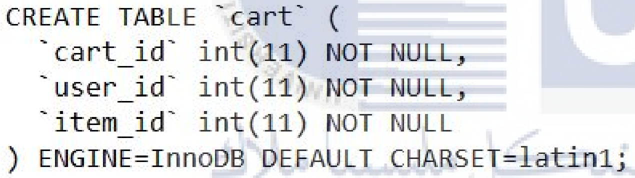
PHYSICAL DATABASE DESIGN

Table: Admin



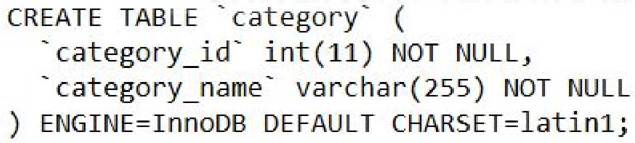
```
CREATE TABLE `admin` (
  `admin_id` int(11) NOT NULL,
  `admin_username` varchar(80) CHARACTER SET utf8mb4 NOT NULL,
  `admin_email` varchar(100) CHARACTER SET utf8mb4 NOT NULL,
  `admin_name` varchar(100) CHARACTER SET utf8mb4 NOT NULL,
  `admin_ic` varchar(9999) CHARACTER SET utf8mb4 NOT NULL,
  `admin_contact` varchar(255) CHARACTER SET utf8mb4 NOT NULL,
  `admin_password` varchar(100) CHARACTER SET utf8mb4 NOT NULL,
  `admin_img` varchar(255) CHARACTER SET utf8mb4 NOT NULL DEFAULT 'default.png'
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

Table: Cart



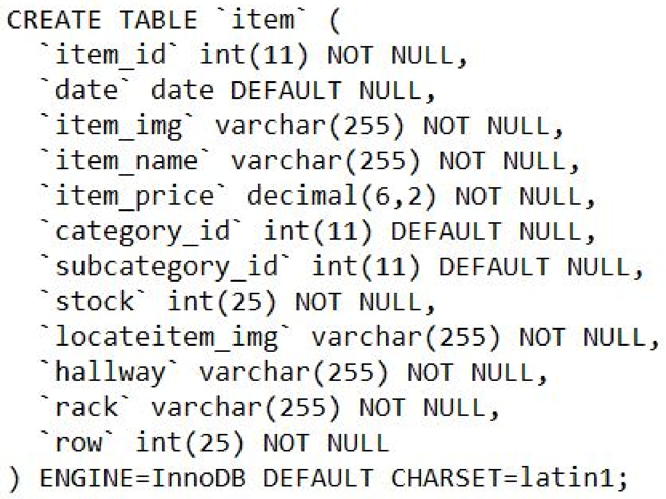
```
CREATE TABLE `cart` (
  `cart_id` int(11) NOT NULL,
  `user_id` int(11) NOT NULL,
  `item_id` int(11) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

Table: Category



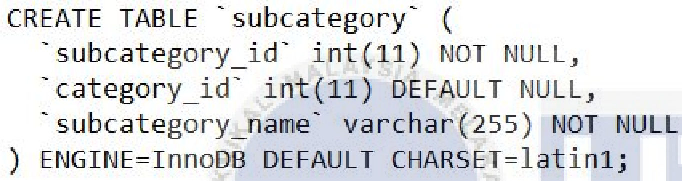
```
CREATE TABLE `category` (
  `category_id` int(11) NOT NULL,
  `category_name` varchar(255) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

Table: Item



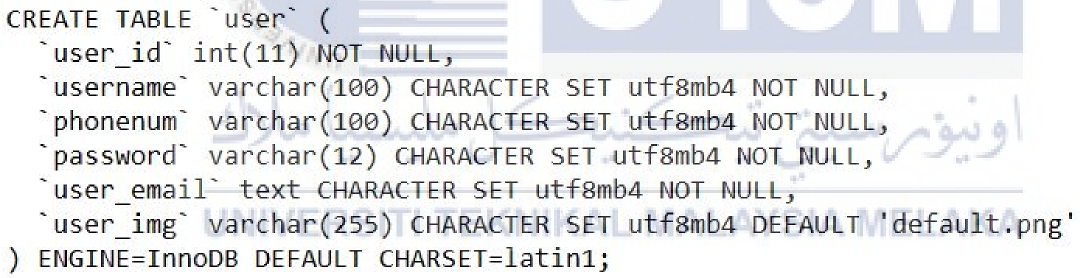
```
CREATE TABLE `item` (
  `item_id` int(11) NOT NULL,
  `date` date DEFAULT NULL,
  `item_img` varchar(255) NOT NULL,
  `item_name` varchar(255) NOT NULL,
  `item_price` decimal(6,2) NOT NULL,
  `category_id` int(11) DEFAULT NULL,
  `subcategory_id` int(11) DEFAULT NULL,
  `stock` int(25) NOT NULL,
  `locateitem_img` varchar(255) NOT NULL,
  `hallway` varchar(255) NOT NULL,
  `rack` varchar(255) NOT NULL,
  `row` int(25) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

Table: Subcategory



```
CREATE TABLE `subcategory` (
  `subcategory_id` int(11) NOT NULL,
  `category_id` int(11) DEFAULT NULL,
  `subcategory_name` varchar(255) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

Table: User



```
CREATE TABLE `user` (
  `user_id` int(11) NOT NULL,
  `username` varchar(100) CHARACTER SET utf8mb4 NOT NULL,
  `phonenum` varchar(100) CHARACTER SET utf8mb4 NOT NULL,
  `password` varchar(12) CHARACTER SET utf8mb4 NOT NULL,
  `user_email` text CHARACTER SET utf8mb4 NOT NULL,
  `user_img` varchar(255) CHARACTER SET utf8mb4 DEFAULT 'default.png'
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```