

Design and Development of Food Ordering and Management System for One Six Eight Restaurant

Ma Xiao Ting¹, Nayef Abdulwahab Mohammed Alduais^{1*}

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

DOI: <https://doi.org/10.30880/aitcs.2024.05.01.028>

Received 24 June 2023; Accepted 18 May 2024; Available online 30 August 2024

Abstract: Web-Based Ordering and Management System for One Six Eight Restaurant is developed due to several issues such as platform used can cause confusion on customers who placed order at the same time, human error on summarising large amounts of orders, and manual reporting can cause waste in time and materials. Thus, the aim of the proposed system is to transform the manual food ordering process into a systematic process where waterfall model methodology with structured approach is used in this study. Context Diagram (CD), Data Flow Diagram (DFD), and Entity Relationship Diagram (ERD) is conducted. Moreover, the proposed system is successfully developed using Hypertext Preprocessor (PHP) with the database My Structured Query Language (MySQL). Functional Testing and User Acceptance Test is done to check the system. The proposed system has achieved its aim where the system is able to improve the ordering process of One Six Eight Restaurant.

Keywords: Web-based, waterfall model, structured approach

1. Introduction

One Six Eight Restaurant is a restaurant located near Universiti Tun Hussein Onn Malaysia, Parit Raja, Johor. It provides services for UTHM students to place their orders. As known, they only provide dinner services for UTHM students. The overall process involves two main users which are admin and customers. The admin will be the owner of the One Six Eight Restaurant whereas the customers will be UTHM students. As known by the interview, all steps in the ordering and managing process are in manual form, this brings some problems towards the restaurant.

First, the platform used to place orders can cause confusion on students who place orders at the same time. Messaging applications such as WhatsApp and WeChat are used by One Six Eight Restaurant to collect their orders placed by UTHM students. Unfortunately, when students place their orders at the same time, messages sent will overlap each other. This causes problems on students which might miss out on the previous message and orders will be messed up. Human error on summarising large amounts of orders is often done. As the owner of One Six Eight Restaurant mentioned, the method

*Corresponding author: nayef@uthm.edu.my

| This is an open access article under the CC BY-NC-SA 4.0 license.

used to summarise their large amount of orders is using Microsoft Excel and human calculation. This will cause them to miscount on their orders if they are not fully focused. They need to count again if they get disturbed. Manual reporting can cause waste in time and materials. As known from One Six Eight Restaurant, the time spent on human calculation and summarising can be used to prepare orders. Unfortunately, the system is still in manual. This causes them need to put some time into calculating a large amount of orders as in the time used can be done other meaningful things.

Therefore, the objective for the proposed system are to design web-based food ordering and management system for One Six Eight Restaurant to transform manual food ordering process into systematic food ordering process, to develop an efficient platform for One Six Eight Restaurant to systematic food ordering process and solve reporting issues, and to test and evaluate the proposed system in the perspective of functionality and non-functionality testing.

At the end of the project, the proposed system can overcome problems faced by the owner of One Six Eight Restaurant and UTHM Students. First, the problem of overlapping orders can be settled. Students can place their orders without worrying that their orders will be missed, this works on the restaurant owner too. Next, Owner of One Six Eight Restaurant can be more confident in their summarising report as the system will help them. Human error can be decreased. Lastly, time and materials can be saved. Owner of the restaurant can save their time on manually calculating large amounts of orders and prepare their orders in a more efficient way.

2. Related Work

2.1 Online Food Ordering System

COVID-19 has significantly increased demand for online ordering and this causes online food ordering to become the new way of eating out [1]. Thus, an online food ordering system lets the business provide and manage their orders placed online for delivery or take away. The system allows customers to go through a digital menu on a website or application to place their order, and pay for it online. Besides this, the system also provides an administrative interface that allows the restaurant to take and process customer orders. Advantages can be gained by both users and restaurants as users can save their money on transportation whereas restaurants can receive large amounts of orders without rushing to tidy up those messy orders that are recorded using papers.

2.2 Restaurant Management System

A management system is the method which an organisation uses to coordinate the various aspects of its operations in order to accomplish its goals. In the hospitality industry, the management system helps to manage the various operations of a restaurant. A restaurant management system (RMS) is a type of point-of-sale (POS) software designed specifically for restaurants, bars, food trucks, and other food service businesses [2]. It can be used to track inventory, manage customers and orders, payments, and generate reports. The system can be applied in both small and large restaurants. Restaurant management systems bring benefits to restaurants as it helps the restaurant to be more efficient and organised. It also helps restaurants to improve customer services.

2.3 Web-based System

A web-based system is an application that may be accessed via HTTP. The phrase "web-based" refers to applications that operate in a web browser. Web-based applications used to be extremely limited in terms of capability. However, technological, security, and internet speed advancements have considerably expanded the potential scope of web-based systems. We now have web-based business accounting systems, CRM systems, a web-based Microsoft Office, and other tools. Web-based applications have a number of advantages over native, client-based software.

2.4 Comparison of Existing System with the Proposed System

Table 1 shows the comparison table between the existing system and proposed system. As shown in Table 1, three existing systems provide functions of user login and signup module for customers to sign up an account on further purchase. Next, the user reset password function is also provided for users who forgot their password. An email verification will be sent to users' email for them to verify and reset their password. Furthermore, users are able to perform CRUD functions on the food ordering process, users are able to add, edit, update and delete items they want to purchase. Unfortunately, three existing systems are not confirmed on the admin site as to whether the system can perform CRUD functions on user management and food management. Same as the reporting management module, it is not able to identify whether the current system has or not. Moreover, three existing systems do provide delivery service but they did not provide free delivery services. For a clear and clean design interface, both Pizza Hut and Restoran Mahhub have a clear and understanding design interface for users to access the system, but Nando's Malaysia has a complicated design interface where users need to take some time to understand how to purchase food in the system. Lastly, the search food function for the user food menu is not provided in 3 existing systems.

Table 1 Comparison table between the existing system and proposed system

Feature / System	Pizza Hut	Restoran Mahhub	Nando's Malaysia	Proposed System
User signup module	Yes	Yes	Yes	Yes
User login module	Yes	Yes	Yes	Yes
User reset password function	Yes	Yes	Yes	Yes
User able to perform CRUD functions on food ordering	Yes	Yes	Yes	Yes
Admin able to perform CRUD functions on user management	No	No	No	Yes
Admin able to perform CRUD functions on food management	No	No	No	Yes
Admin reporting management module	No	No	No	Yes
Free delivery services	No	No	No	Yes
Clear and clean design interface	Yes	Yes	No	Yes
Search function for user food menu	No	No	No	Yes

3. Methodology/Framework

In project development, project management implements a wide range of tools, techniques, and methodologies [3]. In this project, the waterfall model is used. The waterfall model is a software development life cycle model developed by Royce in the 1970s [4]. The waterfall model is one method in the System Development Life Cycle (SDLC) that works on each phase of the waterfall that must be completed before moving on to the next phase [5]. Since the waterfall model requires development to be done in sequential order, each step must be completed before moving on to the next, making changes requires a formal process. These steps are repeated indefinitely until the software is finished [6]. There are a total 5 phases in Waterfall Methodology which is requirement analysis, design, implementation, testing and maintenance.

The model is chosen for this project as the overall has been fixed and well-defined. The overall project will have specific and unchanging requirements. Thus, the waterfall model is decided to be used in this project. There are a total 5 phases in the waterfall model which are analysis phase, design phase, implementation phase, testing phase and maintenance phase. Detailed explanation of each phase will be discussed on the upcoming sub-topic. Figure 1 shows the diagram of the Waterfall Model.

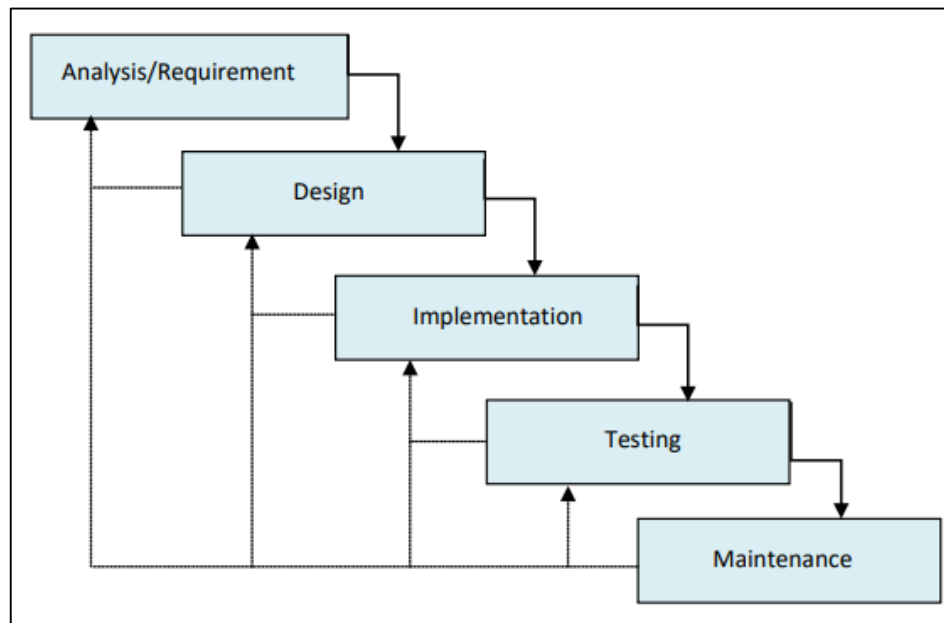


Figure 1: Waterfall Model (Lawal and Ogbu, 2021)

3.1 Analysis Phase

The analysis phase, also known as software requirement specifications (SRS), is a comprehensive description of the behaviour of the software that will be developed [5]. The purpose of this phase is to analyse the overall project outline for future development. In the requirement analysis phase, the overall project outline is analysed and decided to complete the project smoothly. An interview is done with the restaurant owner of One Six Eight to understand the current situation. After the interview session, the overall project outline is done as in deciding the title of the project, problem statements which are faced by the restaurant owner, objectives to fulfil the problem statement, and the project scope. Besides this, expected outcomes and project significance need to be stated too. Planning of the overall project is done too. A Gantt Chart of the overall project is done to specify clearly on the process to develop the project. The Gantt Chart will be shown in Appendix A.

Then, research of the project is done to understand more on the system project. Related articles and journals are searched to collect information for further development of the proposed system. A review of existing systems is conducted to understand the current system features, and comparison is done among the existing systems to get a conclusion on the advantages of the features to be applied in the proposed system. Functional and non-functional requirements are defined for future development too. Moreover, the hardware and software requirements for the project were determined. Lastly, methodology of the project is decided to have a structure plan on the development process.

3.2 Design Phase

This phase consists of the planning and problem-solving methods for software solutions. It means that the developers will define the solution's plan, which will comprise algorithm design, software architecture design, logical diagram scheme, data structure description, and so on [5]. The purpose of this phase is about designing software, which includes additional efforts to build software. In the design

phase, system design will be done to fulfil the requirements requested by the owner restaurant. First, different diagrams such as Data Flow Diagram (DFD), context diagram (CD), and Entity Relationship Diagram (ERD) are drawn for future development. Then, a system architecture diagram is drawn to analyse the system. Besides this, database design and interface design is done for upcoming system development.

A context diagram defines how external entities interact with an internal software system. Context diagrams are high-level diagrams that do not go deeply into the specifics of the system. Instead, it draws out a complete system in a basic, clear, and to be understand. Figure 2 shows the context diagram of the proposed system.

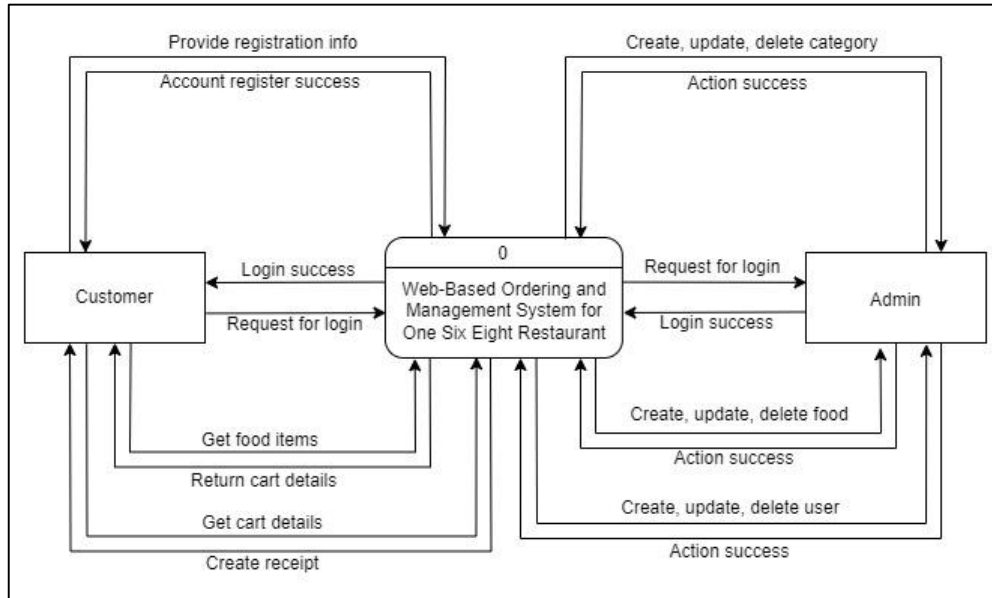


Figure 2: Context diagram of the proposed system

The context diagram is concerned with how external entities interact with the system. It is the simple type of data flow diagram. Data flow diagram is different from a context diagram as it provides information about a system that a context diagram does not have. They provide more details on how the system works. Figure 3 shows the data flow diagram level 0 of the proposed system.

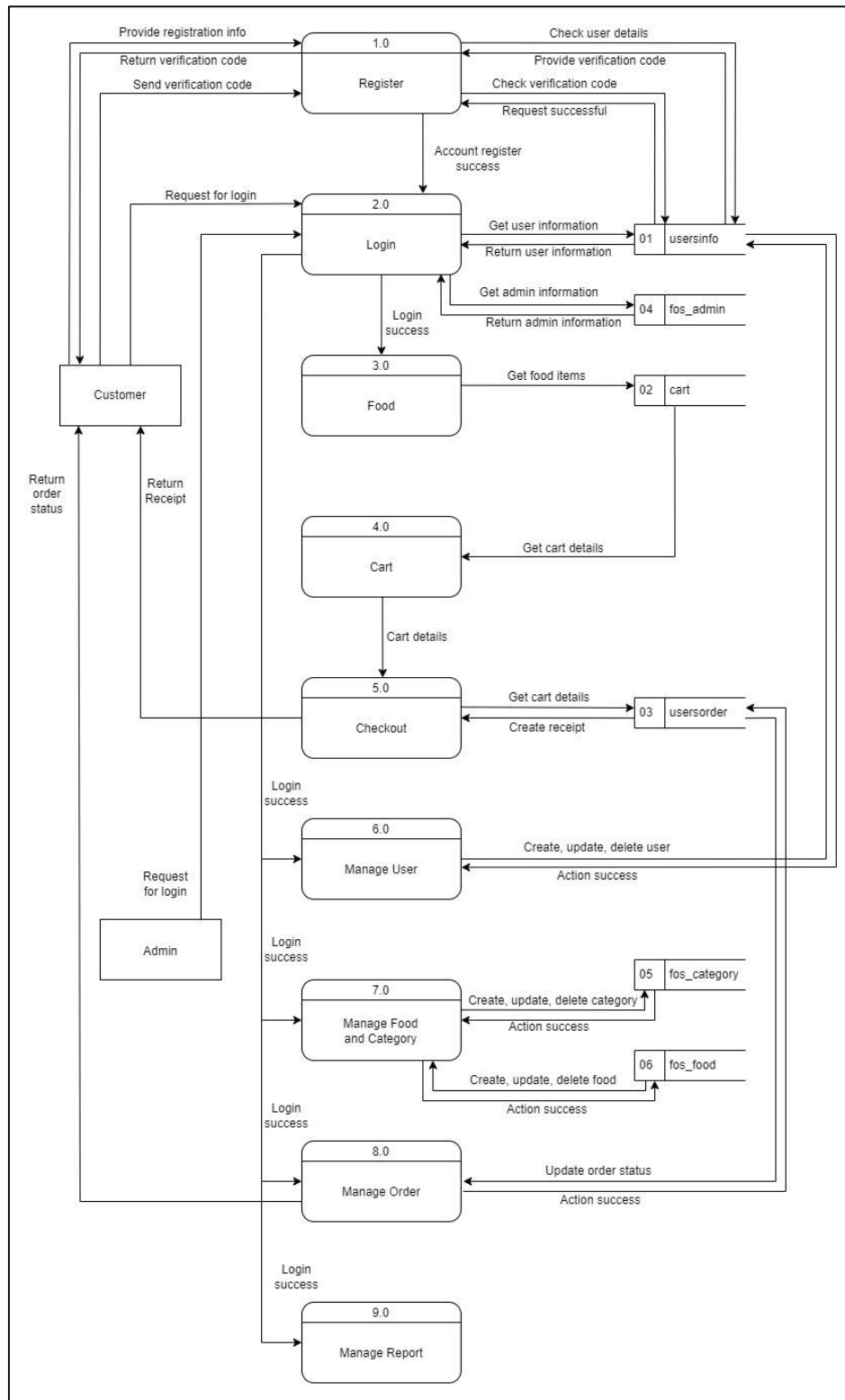


Figure 3: Data flow diagram level 0 of the proposed system

The system architecture is a model that logically specifies the system's points of view, structure, and behaviour. In other terms, system architecture is a visual representation and explanation of how a system functions and communicates with other system components. In Figure 4, the system architecture of the proposed system is shown.

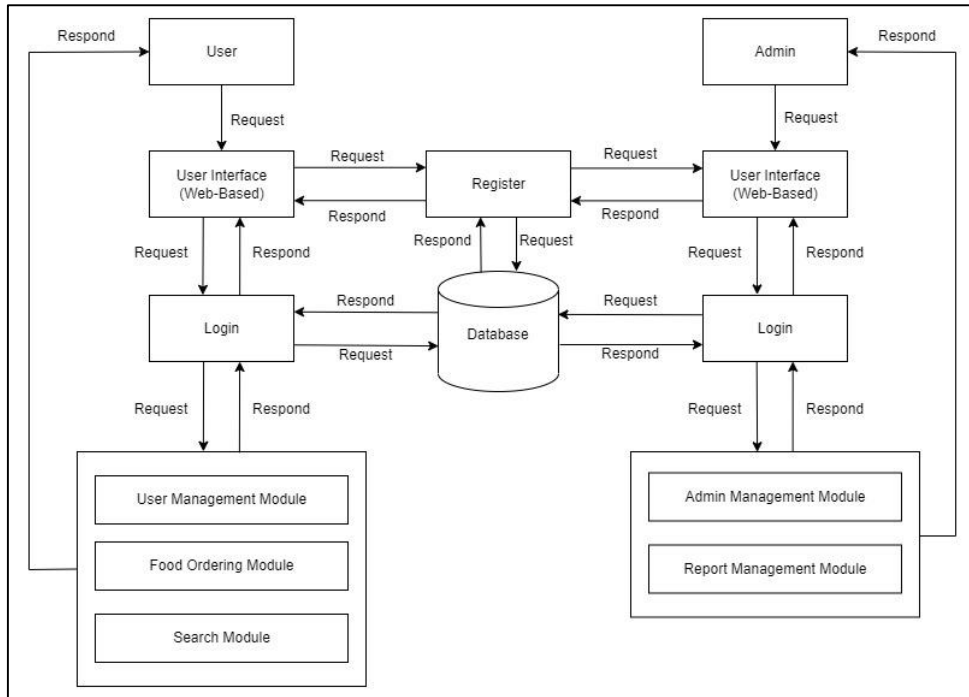


Figure 4: System architecture diagram of the proposed system

An Entity Relationship Diagram (ERD), also known as an entity relationship model, is an illustration of the relationships that occur between people, objects, places, concepts, or events within an information technology (IT) system [7]. Entity relationship diagrams act as a visual starting point for the design of databases and can also be used to evaluate information system requirements across an organisation. Therefore, Figure 5 shows the entity relationship diagram of the proposed system.

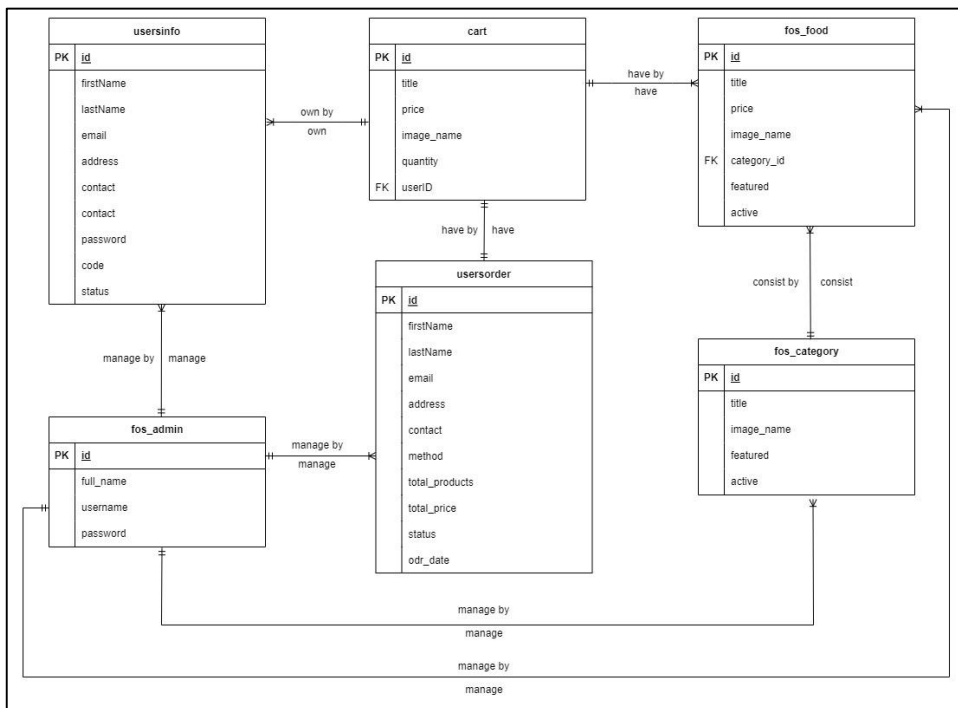
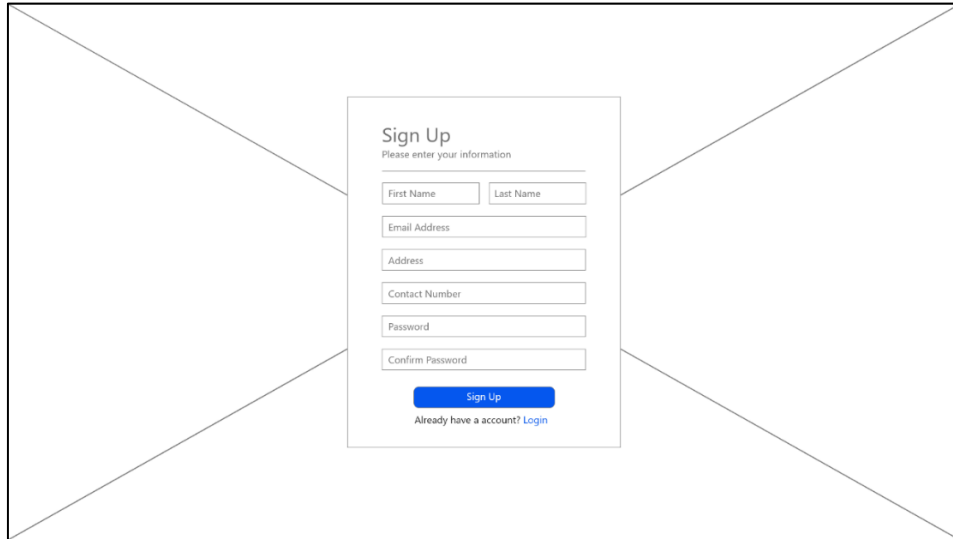


Figure 5: Entity relationship diagram of the proposed system

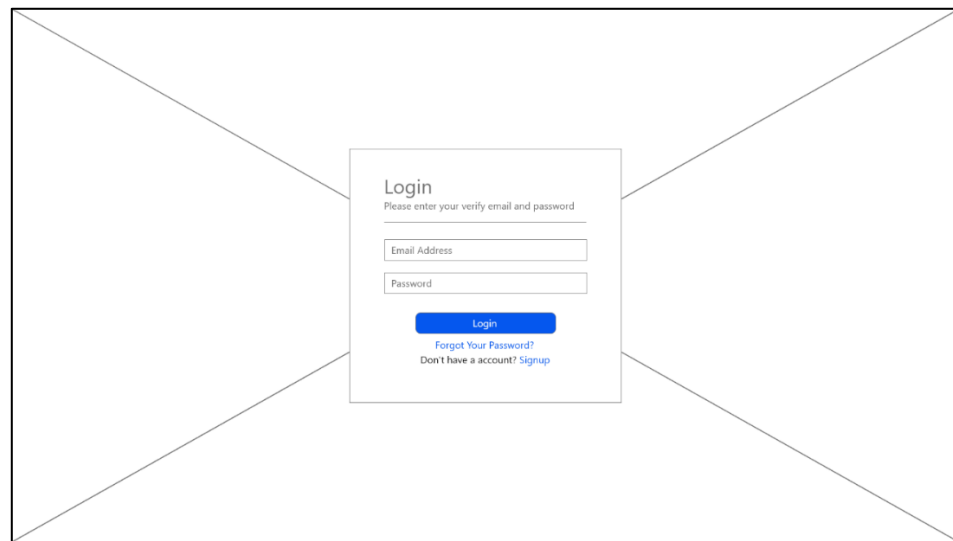
User interface design refers to the creation of a user interface for software, websites, or applications. It is about programming the appearance of objects in order to facilitate usage and improve the user

experience. Hence, the user interface design for customer and admin of the proposed system will be shown below. Figure 6 shows the sign up interface for user, Figure 7 shows the login interface for users, Figure 8 shows the forgot password interface for users, Figure 9 shows the add to cart interface for users, Figure 10 shows the checkout interface for users, Figure 11 shows the login interface for admin, Figure 12 shows the add user interface for admin.



The image shows a wireframe of a 'Sign Up' form. The form is centered within a larger rectangular frame that has diagonal lines from the corners to the center, suggesting a page layout. The form itself has a title 'Sign Up' and a subtitle 'Please enter your information'. Below the subtitle are several input fields: 'First Name' and 'Last Name' (two separate boxes), 'Email Address', 'Address', 'Contact Number', 'Password', and 'Confirm Password'. At the bottom of the form is a blue button labeled 'Sign Up' and a link that says 'Already have an account? Login'.

Figure 6: Sign up interface for users



The image shows a wireframe of a 'Login' form. Similar to Figure 6, it is centered in a larger frame with diagonal lines. The form has a title 'Login' and a subtitle 'Please enter your: verify email and password'. It contains two input fields: 'Email Address' and 'Password'. At the bottom, there is a blue button labeled 'Login' and two links: 'Forgot Your Password?' and 'Don't have an account? Signup'.

Figure 7: Login interface for users

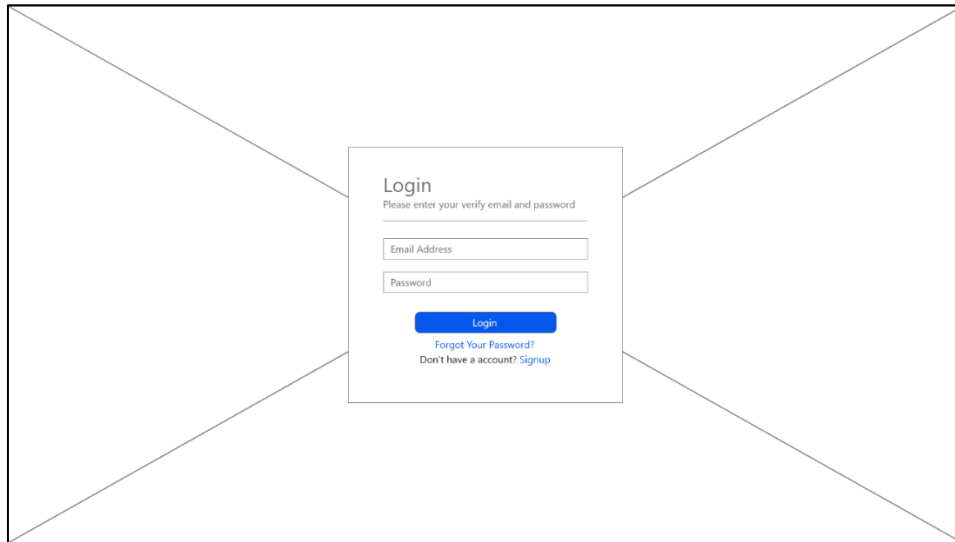


Figure 8: Forgot password interface for users

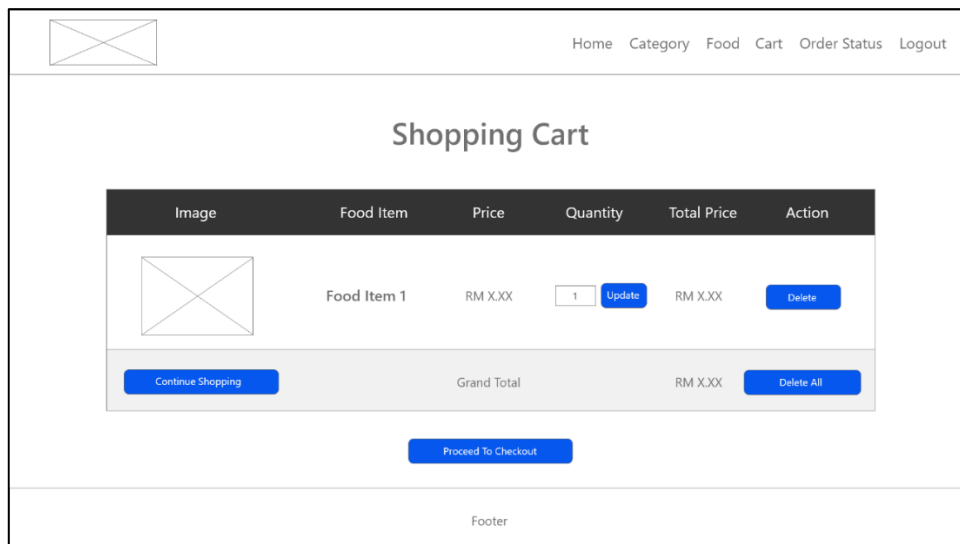


Figure 9: Add to cart interface for users

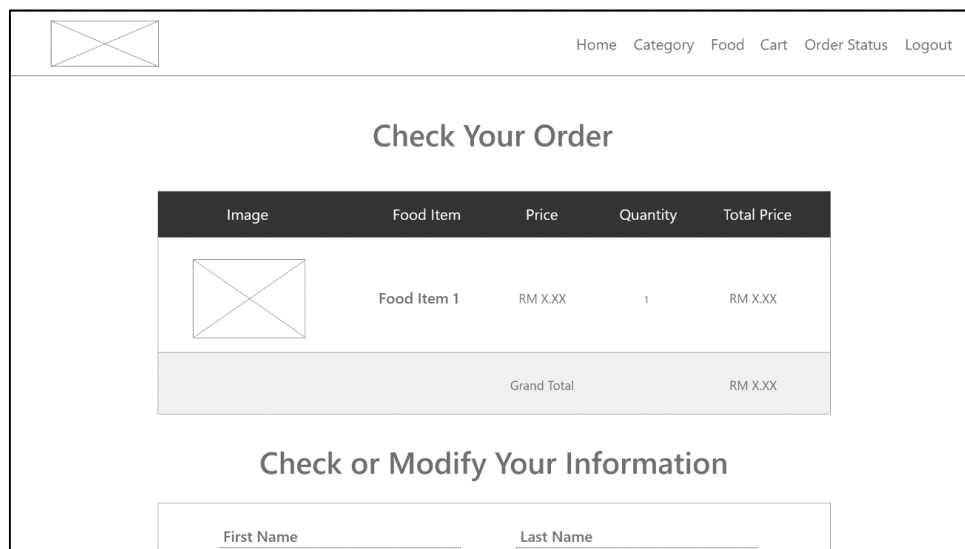


Figure 10: Checkout interface for users

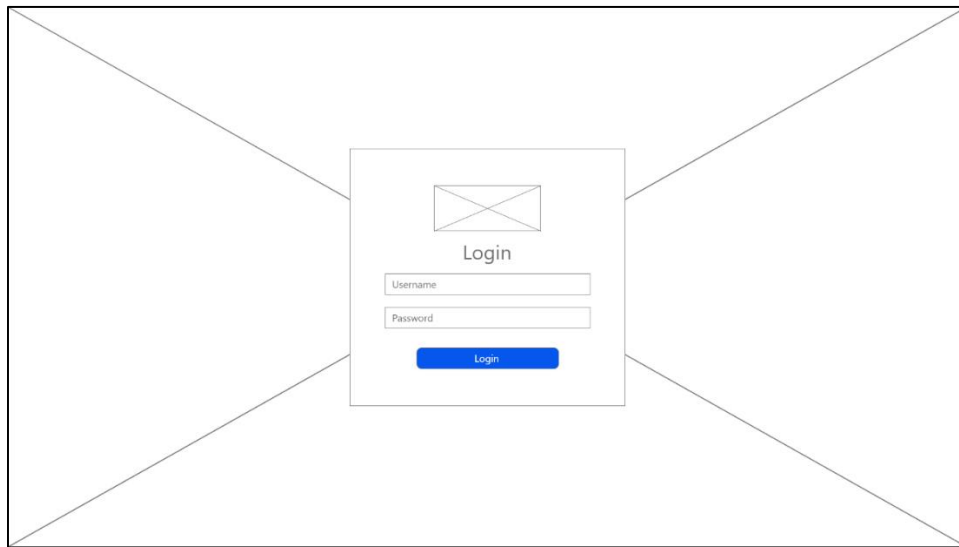


Figure 11: Login interface for admin

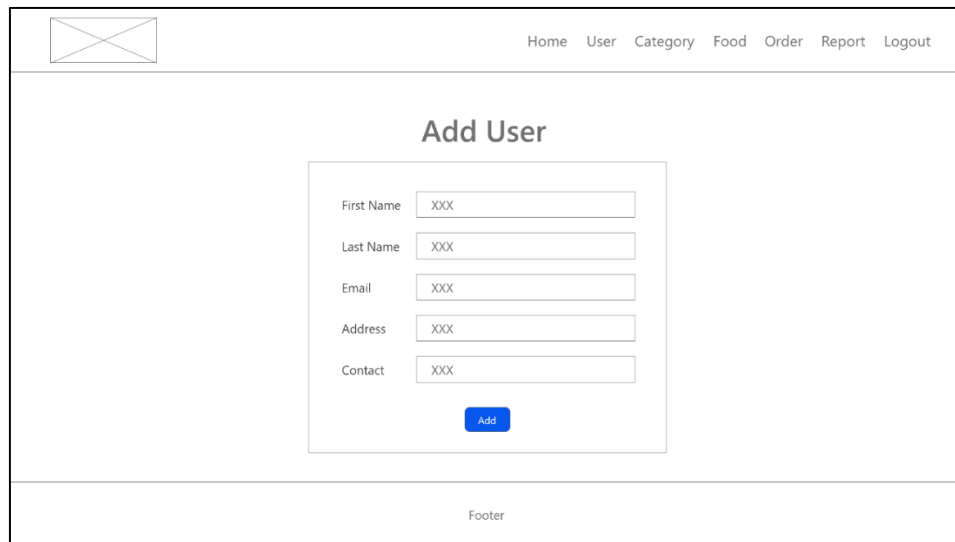


Figure 12: Add user interface for admin

3.3 Implementation Phase

Implementation phase is the phase at which the original code is written and compiled into an operating system, from which the database and text file are formed [5]. The design document which is done in the design phase will be executed into a real life system which can function and be used in real life situations. The purpose of this phase is to use the hardware and software requirements planned in the analysis phase to apply in the implementation phase for the development process. This phase will continue until the requirement of the Food Ordering and Management system of One Six Eight Restaurant is archived. First, the user interface design which was designed before in the design phase will be executed by using the software Visual Studio Code to code the proposed system. The programming language used to develop the proposed system will be Hypertext Markup Language (HTML), Cascading Style Sheets (CSS) and Hypertext Preprocessor (PHP). The database will be created in phpMyAdmin using MySQL database-related language and connected to the proposed system.

Figure 13 shows the sign up page for user and Figure 14 shows the code segment of sign up page for users. Figure 15 shows the login page for users whereas Figure 16 shows the code segment of login page for users. Figure 17 shows the forgot password page for users and Figure 18 shows the code segment of forgot password page for users. Figure 19 shows the add to cart page for users whereas Figure 20 shows the code segment of add to cart page for users. Figure 21 shows the checkout page for users and Figure 22 shows the code segment of checkout page for users. Figure 23 shows the login page for admin whereas Figure 24 shows the code segment of login page for admin. Figure 25 shows the add user page for admin whereas Figure 26 shows the code segment of add user page for admin.

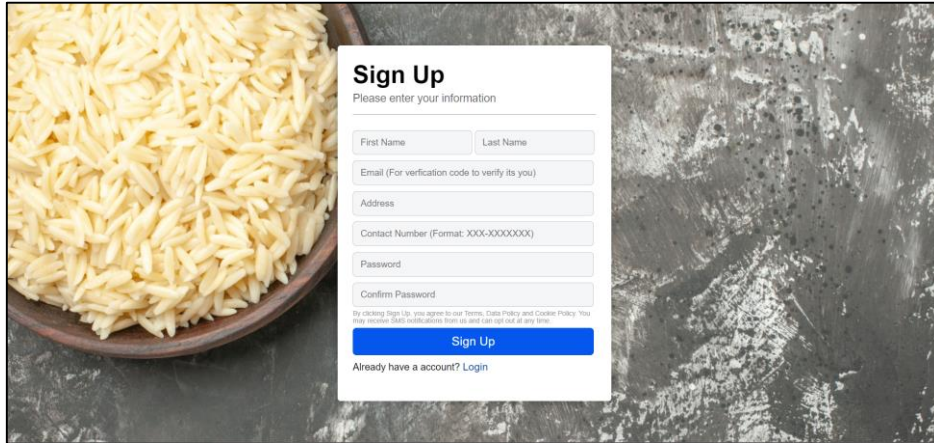


Figure 13: Sign Up Page for Users

```

// When Sign Up Button Clicked
if (isset($_POST['firstName'])) {
    $firstName = mysqli_real_escape_string($conn, $_POST['firstName']);
    $lastName = mysqli_real_escape_string($conn, $_POST['lastName']);
    $email = mysqli_real_escape_string($conn, $_POST['email']);
    $address = mysqli_real_escape_string($conn, $_POST['address']);
    $contact = mysqli_real_escape_string($conn, $_POST['contact']);

    // check password length
    // if password is less than 8 character, then error message will show
    // else, if password meets the requirement, then check the password match with confirm password or not
    // if not match, then error message will show
    // else, password is matched.
    if (strlen(trim($_POST['password'])) < 8) {
        $errors['password'] = 'Use 8 or more characters with a mix of letters, numbers & symbols';
    } else {
        // if password not matched so
        if ($_POST['password'] != $_POST['confirmPassword']) {
            $errors['password'] = 'Password not matched';
        } else {
            $password = md5($_POST['password']);
        }
    }
}
    
```

Figure 14: Code Segment of Sign Up Page for Users

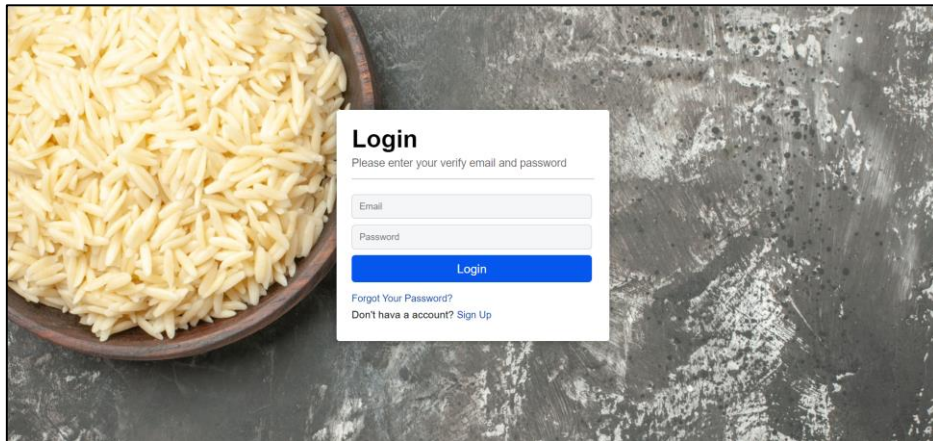


Figure 15: Login page for users

```
// if login Button clicked so
if (isset($_POST['login'])) {
    // $email = mysqli_real_escape_string($conn, $_POST['email']);
    $email = $_POST['email'];
    $raw_password = md5($_POST['password']);
    // $raw_password = mysqli_real_escape_string($conn, $raw_password);

    $sql = "SELECT * FROM usersinfo WHERE email = '$email'";
    $res = mysqli_query($conn, $sql);
```

Figure 16: Code segment of login page for users

```
if (mysqli_num_rows($res) > 0) {
    $sql2 = "SELECT * FROM usersinfo WHERE email = '$email' AND password = '$raw_password'";
    $res2 = mysqli_query($conn, $sql2);
    if (mysqli_num_rows($res2) > 0) {
        // $row=mysqli_fetch_assoc($result);
        $row = mysqli_fetch_assoc($res2);
        $status = $row['status'];
        $userID = $row['id'];

        if ($status === 'Verified') {
            $name = $row['firstName'] . " " . $row['lastName'];
            $_SESSION['userID'] = $userID;
            $_SESSION['email'] = $row['email'];
            // $_SESSION['password'] = $row['password'];
            header('location: index.php');
        } else {
            $info = "It's look like you still haven't verify your email $email";
            $_SESSION['message'] = $info;
            header('location: otp.php');
        }
    } else {
        $errors['email'] = 'Password did not matched';
    }
} else {
    $errors['email'] = 'Invalid Email Address';
}
```

Figure 16: Code segment of login page for users (cont.)

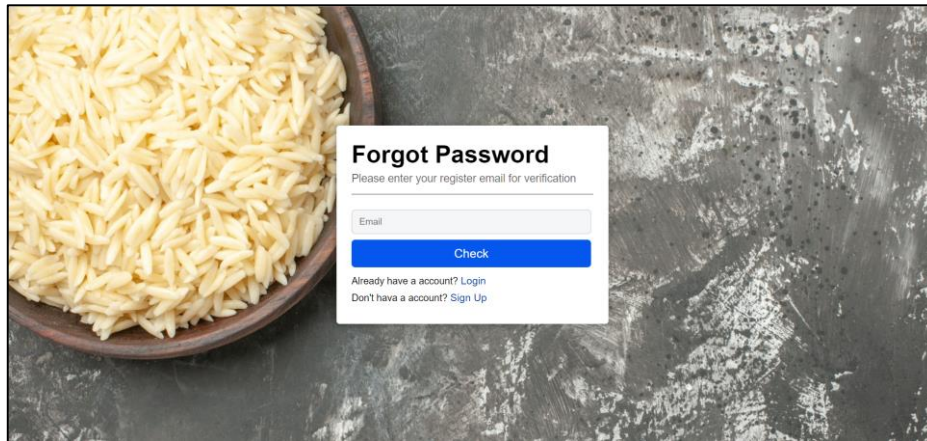


Figure 17: Forgot password page for users

```
// if forgot button will clicked
if (isset($_POST['forgot_password'])) {
    $email = $_POST['email'];
    $_SESSION['email'] = $email;

    $emailCheckQuery = "SELECT * FROM usersInfo WHERE email = '$email'";
    $emailCheckResult = mysqli_query($conn, $emailCheckQuery);
```

Figure 18: Code segment of forgot password page for users

```
// if query run
if ($emailCheckResult) {
    // if email matched
    if (mysqli_num_rows($emailCheckResult) > 0) {
        $code = rand(999999, 111111);
        $updateQuery = "UPDATE usersInfo SET code = $code WHERE email = '$email'";
        $updateResult = mysqli_query($conn, $updateQuery);
        if ($updateResult) {
            $subject = 'Email Verification Code';
            $message = "The verification code is $code";
            $sender = 'From: celinema5122@gmail.com';

            if (mail($email, $subject, $message, $sender)) {
                $message = "We've sent a verification code to your Email <br> $email";

                $_SESSION['message'] = $message;
                header('location: verifyEmail.php');
            } else {
                $errors['otp_errors'] = 'Failed while sending code!';
            }
        } else {
            $errors['db_errors'] = "Failed while inserting data into database!";
        }
    } else {
        $errors['invalidEmail'] = "Invalid Email Address";
    }
} else {
    $errors['db_error'] = "Failed while checking email from database!";
}
```

Figure 18: Code segment of forgot password page for users (cont.)

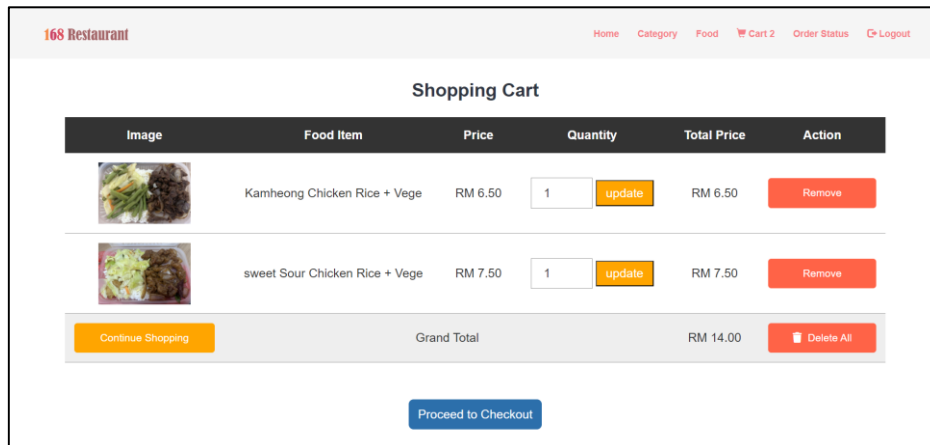


Figure 19: Add to cart page for users

```

<?php
if(isset($_POST['update_cart']))
{
    $update_value = $_POST['cart_quantity'];
    $update_id = $_POST['cart_id'];
    $update_quantity_query = mysqli_query($conn, "UPDATE cart SET quantity = '$update_value' WHERE id = '$update_id'");

    if($update_quantity_query)
    {
        $message[] = 'cart quantity updated successfully!';
        header('location:addToCart.php');
    }
};

if(isset($_GET['remove']))
{
    $remove_id = $_GET['remove'];
    mysqli_query($conn, "DELETE FROM cart WHERE id = '$remove_id'");
    header('location:addToCart.php');
};

if(isset($_GET['delete_all']))
{
    mysqli_query($conn, "DELETE FROM cart WHERE userID = '".$_SESSION['userID']."'");
    header('location:addToCart.php');
};
?>
    
```

Figure 20: Code segment of add to cart page for users

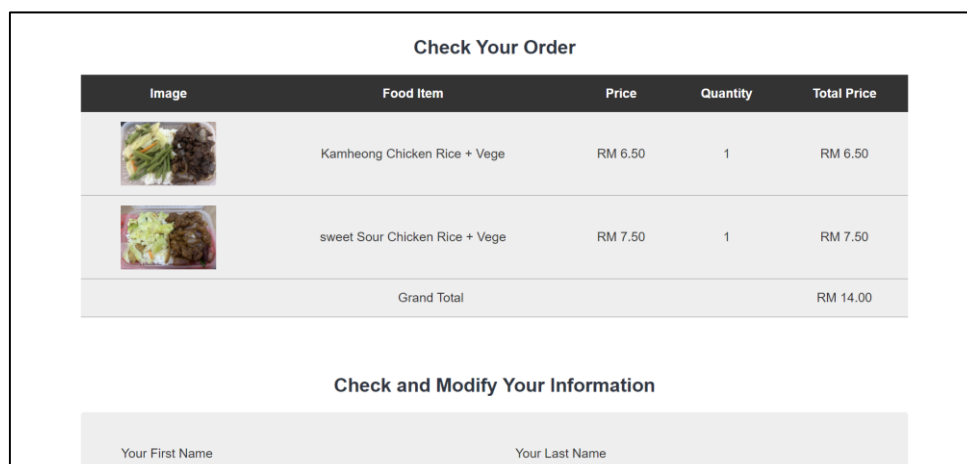


Figure 21: Checkout page for users

```

if(isset($_POST['order_btn']))
{
    $firstName = $_row['firstName'];
    $lastName = $_row['lastName'];
    $email = $_row['email'];
    $address = $_POST['address'];
    $contact = $_POST['contact'];
    $method = $_POST['method'];

    $odr_date = date("Y-m-d h:i:sa"); //order date

    $status = "Ordered"; //ordered, on delivery, delivered, cancelled

    $cart_query = mysqli_query($conn, "SELECT * FROM cart");
    $price_total = 0;

    if(mysqli_num_rows($cart_query) > 0)
    {
        while($product_item = mysqli_fetch_assoc($cart_query))
        {
            $product_name[] = $product_item['title'] . ' (' . $product_item['quantity'] . ')';
            $product_price = number_format($product_item['price'] * $product_item['quantity'], 2);
            $price_total += $product_price;
        };
    };

    $total_product = implode(', ', $product_name);
    $detail_query = mysqli_query($conn, "INSERT INTO usersOrder (firstName, lastName, email, address, contact, method, total_products, total_price, odr_date, status)
VALUES ('$firstName', '$lastName', '$email', '$address', '$contact', '$method', '$total_product', '$price_total', '$odr_date', '$status')") or die('query failed');
}
    
```

Figure 22: Code segment of checkout page for users

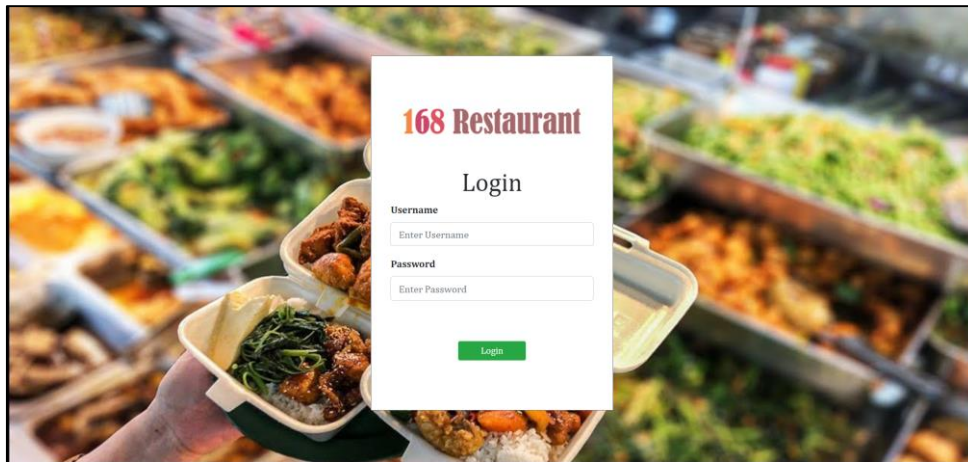


Figure 23: Login page for admin

```

<?php
//check whether the submit button is clicked or not
if(isset($_POST['submit']))
{
    //process for login page
    //get the data from login FORM
    $username = mysqli_real_escape_string($conn, $_POST['username']);
    $raw_password = md5($_POST['password']);
    $raw_password = mysqli_real_escape_string($conn, $raw_password);

    //sql to check whether the user with username and password exist or not
    $sql = "SELECT * FROM fos_admin WHERE username='$username' AND password='$raw_password'";

    //execute the query
    $res = mysqli_query($conn, $sql);

    //count the rows to check whether the user exist or not
    $count = mysqli_num_rows($res);

    if($count == 1)
    {
        //user available and login success
        $_SESSION['login'] = "<div class='success'>Login Success!</div>";
        $_SESSION['user'] = $username; //to check whether the user is logged in or not and logout will unset it

        //redirect to admin homepage
        header('location:'.SITE_URL.'admin/');
    }
    else
    {
        //user not available and login fail
        $_SESSION['login'] = "<div class='error text-center'>Login Fail! Please Try Again!</div>";
        //redirect to admin homepage
        header('location:'.SITE_URL.'admin/login-admin.php');
    }
}
}
    
```

Figure 24: Code segment of login page for admin

Home User Category Food Order Logout

Add User

First Name:

Last Name:

Email:

Address:

Contact Number:

© 2023 One Six Eight Restaurant (JM0480806-T) | All Rights Reserved.

Figure 25: Add user page for admin

```
<?php
//process the value from the FORM and save in database
//check whether the submit button is clicked or not

if(isset($_POST['submit']))
{
    //Button Clicked
    //echo "Button Clicked";

    //get data from FORM
    $firstName = $_POST['firstName'];
    $lastName = $_POST['lastName'];
    $email = $_POST['email'];
    $address = $_POST['address'];
    $contact = $_POST['contact'];

    //create sql query to save the data into database
    //left is column name from db
    //right is value from FORM

    $sql = "INSERT INTO usersInfo SET
        firstName = '$firstName',
        lastName = '$lastName',
        email = '$email',
        address = '$address',
        contact = '$contact';
";

    // Execute query and save data to database
    $res = mysqli_query($conn, $sql) or die(mysqli_error());
}
```

Figure 26: Code segment of add user page for admin

```

// Execute query and save data to database
$res = mysqli_query($conn, $sql) or die(mysqli_error());

//check whether the data is insert or not and display appropriate message
if($res == TRUE)
{
    //data insert
    // echo "Data insert successfully";

    //create a session variable to display message
    $_SESSION['add'] = "<div class='success'>User Added Successfully</div>";
    //Redirect Page to manage user
    header("location:".SITE_URL. 'admin/manage-user.php');
}
else
{
    //data insert
    //echo "Failed to insert data. Please try again!";

    //create a session variable to display message
    $_SESSION['add'] = "<div class='error'>Failed to Add User</div>";
    //Redirect Page to add user
    header("location:".SITE_URL. 'admin/manage-user.php');
}
}
?>

```

Figure 26: Code segment of add user page for admin

3.4 Testing Phase

Testing phase may be referred to as verification and validation, which is the process of determining whether a software solution fits the original requirements and specifications to reach the intended purpose [5]. The purpose of this phase is to check whether the functionality and usability works after the development process. This is to avoid problems encountered by the users when the system is released. Therefore, in this phase, the proposed system needs to overcome several tests to ensure the functionality and usability works fine without any errors or mistakes. First, testing on functionality is done to check each function which has been developed. Any types of error which are detected during the testing process must be amended and correct to ensure the application is fully functional again. Then, the user acceptance test is done to ensure users can apply the functions provided in the system smoothly.

3.5 Maintenance Phase

This is the final stage of the waterfall model. Maintenance involves fixing errors that were not discovered in the previous step [5]. Modification of the system will be done in this phase to refine the output. Errors which reflect during the testing phase will be corrected in this phase. Improvement of the performance and quality will also be done in this final stage as the proposed system should be able to function well once the system is well developed. The purpose of this phase is to ensure zero errors are found during the final checking and the system will be well performed. Therefore, in this phase, checking, correcting, enhancing will be done. First, errors which are found during the previous steps or testing phase should be corrected immediately. Then, checking on the system is done. This process will repeat until there are zero errors found. Then, enhancement of the system will be done if there is any comment from the user acceptance test. Checking on errors is done again to avoid there being any errors during the enhancement. Lastly, final checking on the overall system will be done to ensure the final product of the system is well done with zero errors.

4. Results and Discussion

The testing part of this project will involve two types of testing which is functional testing and user acceptance testing. These tests must be performed to ensure that the proposed system satisfies all of the requirements specified during the analysis phase. The functional testing is type of testing that consist of comparing the system's output to the requirements and specifications to confirm that it fulfils all the criteria. The results of functional testing of the proposed system will be shown in Table 2.

Table 2: Functional Testing of the proposed system

Functions	Test Case	Expected Result	Actual Results
Register	Empty or incomplete data input	An error message will display if the input field is empty	Pass
	Complete data input with invalid user information	An error message will display if the input field is wrong	Pass
	Complete data input with correct information	Successfully register and redirect to login page	Pass

Table 3: Functional Testing of the proposed system (cont.)

Functions	Test Case	Expected Result	Actual Results
Login	Empty or incomplete data input	An error message will display if the input field is empty	Pass
	Complete data input with invalid email and password	An error message will display if the input field is wrong	Pass
	Complete data input with email and password	Successfully login and redirect to the home page	Pass
Forgot Password	Empty or incomplete email	An error message will display if the input field is empty	Pass
	Complete data input with invalid email	An error message will display if the input field is wrong	Pass
	Complete data input with email	Successfully reset password	Pass

5. Conclusion

In conclusion, the proposed system has been successfully developed and achieved most of the objective, project scope, expected results, and project significance. Although the proposed system has mostly achieved the objective and project significance, it does not achieve all as in there are a few limitations which cause the system not to be perfect due to the limited time and lack of knowledge on the developing process. First, there is no online payment method for users to pay their order online. They are only able to pay their order using cash and some users might have problems with cash since most businesses are using online payment rather than cash. Second, the report management should be added in the system for admin to summarise the sales report for each day, week, month and year. Moreover, the user interface for both user and admin should be improved as it looks too simple without being attractive.

Besides limitation, some recommendation is suggested for future development. First, the online payment method should be available to the proposed system for customers to pay online directly without paying cash when order is arrived. Since most of the payment methods are using online payment rather than paying using cash. Besides this, the user management module can add a profile management function for users to edit their personal information such as address and contact number. Although the proposed system can edit the personal information during confirming the order placed by the customer before confirming payment. But the profile management function will help users to save their time on changing their information every time when they place their order. Besides this, the sales report can be added into the system for checking the sales. Then, invoices for each order can be printed by the system should be added too. This helps the restaurant to place the invoice on each lunchbox for them to verify the order details of the customer. Moreover, the user interface for both user and admin can be improved. The responsive of the website should be done to fit any size of the device. Lastly, a mobile application of food ordering for One Six Eight Restaurant can be developed in the future for food ordering.

Acknowledgment

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

References

- [1] Chai, L. T., & Yat, D. N. C. (2019). Online food delivery services: Making food delivery the new normal. *Journal of Marketing advances and Practices*, 1(1), 62-77.
- [2] Piyatissa, W. B. A. C. (2021). *Web Based Restaurant Management System* (Doctoral dissertation).
- [3] Islam, S., & Evans, N. (2020). Key success factors of PRINCE2 project management method in software development project: KSF of PRINCE2 in SDLC. *International Journal of Engineering Materials and Manufacture*, 5(3), 76-84.
- [4] Aroral, H. K. (2021). Waterfall Process Operations in the Fast-paced World: Project Management Exploratory Analysis. *International Journal of Applied Business and Management Studies*, 6(1), 91-99.
- [5] Heriyanti, F., & Ishak, A. (2020, May). Design of logistics information system in the finished product warehouse with the waterfall method: review literature. In *IOP Conference Series: Materials Science and Engineering* (Vol. 801, No. 1, p. 012100). IOP Publishing.
- [6] Lawal, A., & Ogbu, R. C. (2021). A Comparative Analysis of Agile and Waterfall Software Development Methodologies. *Bakolori Journal of General Studies*, 11(2), 1-2.

[7] Biscobing, J. (2019). Entity Relationship Diagram (ERD). Data Management. <https://www.techtarget.com/searchdatamanagement/definition/entity-relationship-diagram-ERD>

Appendix A

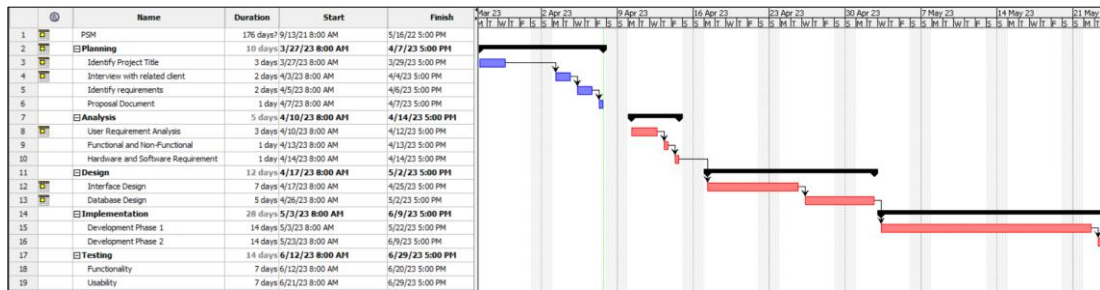


Figure 27: Gantt Chart