

Yummyshome Food Ordering & Management System

W.L.S.P. Jayawardena

2020



Yummyshome Food Ordering & Management System

**A dissertation submitted for the Degree of Master of
Information Technology**

**W.L.S.P Jayawardena
University of Colombo School of Computing
2020**



Abstract

“Yummyshome” food ordering and management system is an online food ordering system. This system is a mobile friendly web application. The aim of this project is to provide more organized platform on the Internet for “Yummyshome” to do their business. Even though business owners use various internet based applications like social media to promote business it is not sufficient to get the best out of the web without having a company owned web site. Yummshome website is an attempt to take the business to a better place on the internet. This system displays food item categories, item images, their details and prices in an orderly manner. The customers can easily and quickly view items and order the quantity they need. The admin can do several management tasks such as add item categories, add items details, edit item categories edit item details, delete item categories, delete item details and view orders. It is possible to obtain several reports needed for management tasks of the business.

The system is a mobile friendly web application that has been developed using ASP.Net C# with Microsoft SQL Server database. It is a unique solution compared to the solutions built using content management systems.

Declaration

The thesis is my original work and has not been submitted previously for a degree at this or any other university/institute.

To the best of my knowledge it does not contain any material published or written by another person, except as acknowledged in the text.

Student Name: W.L.S.P. Jayawardena

Registration Number: 2017/MIT/034

Index Number: 17550341

Signature:

Date: 21/6/2020

This is to certify that this thesis is based on the work of

Ms. W.L.S.P.Jayawardena under my supervision. The thesis has been prepared according to the format stipulated and is of acceptable standard.

Certified by:

Supervisor Name: Prof. Kapila Dias

Signature:

Date:

Acknowledgements

I would like to express my appreciation to all those who provided me the possibility to complete this report. A special gratitude goes to my supervisor Prof. Kapila Dias who always provided suggestions, guidelines and encouragement at any time I was in need of help.

I would also like to acknowledge with much appreciation Rajeeva Kothalawala the owner of “Yummyshome” for providing all necessary details, images and support to make the project more realistic.

I should express my gratitude towards my family for giving me time to complete the project. My thanks go to my friends who gave me suggestions and ideas in developing the project.

I would like to give credit to question and answer site “Stack Overflow” that helped me to learn the software and develop my solution without searching for much help from real world individuals.

Table of Contents

Content	Page number
1. Introduction	1
1.1. Identification of the problem	1
1.2. Motivation	1
1.3. Goals & objectives	2
1.4. Scope	2
1.5. Boundaries	2
2. Background	3
2.1. Similar Systems	3
2.2. Web sites available on the internet	3
2.3. Useful facts found from similar	8
2.4. Technologies & design strategies	9
3. Methodology	10
3.1. Alternate Solutions	11
3.2. The chosen solution	12
3.3. How the present system works	12
3.4. Problems in the present system	13
3.5. Proposed system	13
3.6. Design	14
3.6.1. Use case diagram	14
3.6.2. E.R. Diagram	15
3.6.3. Sequence Diagram	16
3.6.4. Logo design	17
3.6.5. Interface Design	17
4. Evaluation	23
4.1. Functionalities	23
4.2. Problems Encountered	25
4.3. Changes made to the design	25
4.4. Testing	26
5. Conclusion	38
5.1. Deficiencies in the system	38
5.2. Future Work	38
References	39
Appendices	40
A) System Documentation	40
B) User Documentation	53
C) Additional Test Results	54

List of Figures

Figure Number	Title of the figure	Page Number
1.1	Sponge web site home page	4
1.2	Sponge web site category page	5
1.3	Sponge web site item details	5
1.4	Pizza Hut Sri Lanka Home Page	6
1.5	Pizza Hut Sri Lanka Home page-Delivery data	6
1.6	The Cakery Home page	7
1.7	Cakery Item page	8
3.1	Use Case diagram	14
3.2	E.R Diagram	15
3.3	Sequence Diagram	16
3.4	Logo Design	17
3.4.1	Admin Login form	17
3.4.2	User Login form	17
3.4.3	Customer Registration form	18
3.4.4	Home page design	18
3.4.5	Cake category page design	19
3.4.6	Regular cake items page design	20
3.4.7	Dessert items page design	20
3.4.8	Custom birthday cake page design	21
4.11	Home Page test result for bigger screen	26
4.12	Home page test result for small screen	26
4.2	Cake category page test result	27
4.3.1	Customer registration form test result	27
4.3.2	Customer table details test result	28
4.3.3	Admin login test result	28
4.3.4	Admin login redirection page test result	29
4.4.1	User login form test result	29
4.4.2	Special Offer page for customers test result	29
4.5.1	Add edit delete categories test result	30
4.5.2	Display categories test result	30
4.6.1	Add edit delete items test result	31
4.6.2	Item table details test result	31
4.6.3	Add/Edit/Delete offers test result	32
4.7.1	Item selection for order test result	32
4.7.2	Total price calculation test result	33
4.7.3	Add item to order test result	33
4.7.4	Shop More test result	33
4.7.5	Add second item test result	34
4.7.6	Checkout test result	34
4.7.7	Order added test result	35
4.7.8	Orderline table test result	35
4.7.9	Customer details of registered customers	36
4.8	Order report	37
4.9	Income Report	37

List of Tables

Table Number	Title of the table	Page Number
3.1	Test Plan	21

List of Abbreviations

Abbreviation	Meaning	Page Number
SMS	Short Message Service	1
GPS	Global Positioning System	8
HTTP	Hyper Text Transfer Protocol	10
HTML	Hypertext Mark Up Language	10
CLI	Common Language Infrastructure	10
PHP		11
IMAP	Internet Message Access Protocol	11
POP3	Post Office Protocol	11
LDAP	Light Weight Directory Access Protocol	11
MVC	Model View Controller	11

1. Introduction

1.1. Identification of the problem

The present system communicates with the customers mainly using telephone calls, face to face visits or social media. With the development of internet customers search for food ordering places and items from the internet rather than from other methods. Customers prefer to get detailed and trustworthy information about items to decide on the right product to buy. According to the present trend it is not easy for businesses to get orders and also to improve the business without web presence.

Online food ordering systems provide facilities for the customers to go through item lists available, compare items and decide about the right item to place orders. Customers can also go through reviews written by the others to make decisions about items. An online ordering system would help to enhance customer relationships. It helps the customer to place orders 24/7.

Sometimes customers find it difficult to make cash payments when placing orders. A website can offer alternative payment methods like on delivery or by bank deposit.

A web site helps to deliver promotional material to the customers easily. Sharing contents on social media applications greatly help sales promotion. When browsing items from a web site customers can view related products offers and deals. A web site can make the customers to shop 24/7 and the shop to be available to take orders 24/7. Web presence of a shop can reduce the expenses of advertising on TV and other printed media

A web application can easily provide information on orders and cash flow. The system can store customer data and order details in a database and use it to improve the business and its services to the customers. Email advertisements and promotional details can be sent via email and SMS. The owners can easily administrate the ordering process and monitor the income and demand for various food items.

1.2. Motivation

Whenever I liked to have a meal from a new place or a place I visit infrequently I have a habit of searching the shop from the web before calling them or ordering from them. I use the calling facility on the site or get the numbers from their pages or their web sites. I also compare the features of these web sites and their actual services. Some shops have built their web sites very well and maintain their reputation on the web as well as their original shops. Some shops are having very good street outlets but their web presence is very poor. Mainly large scale shops are on the web. But there are very good small scale places like Yummyshome that make tasty quality food for reasonable prices. Ms. Rajeeva the owner of Yummyshome makes tasty & high quality food. Even though she uses social media for her business it is felt that a web site could be used to reach the customers better. These businesses can increase the customer base & earn more if they enter into the web. It also help the customers to see the items sold by these small places in detail and enjoy a unique meal that differs from general items they order very often from the common places.

1.3. Goals/Objectives

- i). Improve customer satisfaction by providing faster and easy access to the item details they need
- ii). Provide 24/7 ordering facility
- iii). Reduce the calls receive from customers to inquire about food details
- iv). Expand the customer base
- v). Manage the food details and maintain them up to date
- vi). Obtain summary reports to evaluate the business performance and to improve the business
- vii). Obtain customer feedback to improve the business

1.4. Scope

1.4.1. Functionalities available for customers

Functionalities available for customers are given below

- viii). Customer account facility - Customers will be allowed to create their own accounts, login to account, edit the account and delete if required
- ix). Standard items -The web site will offer standard food items that are prepared according to a set recipe. Customers can View standard items available in Yummyshome ,Select items and order the standard items. Remove items from the order if required and place the order.
- x). Customizable items - A new item category would be available for the customers to customize the items according to their wish. They can choose customization features like cake icing type, natural or sugar flowers, flavours e.t.c. After customization they can view the price for the customized item.
- xi). Rate & Reviews - Customers can rate the items and write reviews for Yummyshome
- xii). Upload pictures - Allow customers to send pictures for the fancy birthday cakes they need to get done
- xiii). Order cancellation-Cancel the order if required with the allowed period of time
- xiv). Share the site - Share the web site and items in social media

1.4.2. User Role Management

Admin – The admin has the privilege to manage category data and item data. Admin can view reports and print reports also. There is only one admin for the system.

Customer – Customers can view item category details, item details and place orders. Customers can register in the system by providing personal details. The registered customers will have access to offers.

1.4.3. Functionalities for Yummyshome Administrator

Functionalities for Yummyshome Owner/Administrator are as follows

- i). Login to admin account
- ii). Add new categories of items
- iii). Edit/Delete available food categories
- iv). Add new items
- v). Edit /delete available item data

- vi). Edit/delete customizable features from items
- vii). View monthly income report from online orders
- viii). View sales comparison chart from each category of food during a period
- ix). View the best sellers by number of orders during a given period of time

1.5. Boundaries :

Online payment through a payment gateway would not be done. Instead an alternative method like deposit to owner's bank account and pay on delivery options would be made available. Email account integration to receive direct email messages would not be included.

2. Background

Review of similar systems and technologies

Living in a digital world has changed the lifestyles and preferences of people. Customers who used to walk into a restaurant to buy or order food items are no longer prepared to waste their time and energy for ordering food. They need to compare items, their prices ingredients e.t.c before placing an order. Customer reviews and comments are considered as information to make buying decisions.

According to the present trends the businesses find it difficult to survive without web presence. The businesses that are not on the web tend to be left out in the competitive market. It increases the customer relationships with a business. There are many businesses similar to "Yummyshome Food Ordering and Management system" that have adopted the new technology and established their web presence well and boosted sales. Some of the similar systems have been considered below

2.1. Similar Systems

2.1.1. Past MIT projects

MIT students of last batches had done several projects that are similar to Yummyshome system.

1. U.S.Weerasekera(Online Restaurant)
2. S.A.H.K Subasinghe (Online Restaurant Management & Food Ordering system)
3. W.A.R.N Silva(Fine Dine)

The above projects also had some similar features needed by Yummyshome system. Those systems were helpful to get a general idea about a food ordering and management system even though certain features and the approaches were different. Restaurant management systems deal with food ordering and other functions line table reserving also. Yummyhome system is only for food ordering and management only. Selecting food items from item categories and item lists were similar functionalities used in all systems. Ordering items and checkout processes were similar to the functionalities of Yummyshome system

2.2. Web sites available on the Internet

2.2.1. Sponge

Sponge Home Page has many categories as Cakes, Savouries, Catering, Online Menu, About and Contact. It provides birthday cakes, custom designs and edible prints. Sponge transforms the imagination of the customer to reality by preparing the design the way the customer want it. Customers can request for a quote that should be printed on the cake. It has a sample gallery with many sample cakes. It has a separate online menu for the cakes. Customers can place their orders for the standard cakes from the online menu. For each cake it provides number of serves per kg and price. The contact page has company contact details and send a message facility also. Sponge has many features similar to the features of Ymmyshome system [1].

Strengths

- Easy access to cake categories page from home page (figure 1.1, figure 1.2)
- It has Order Online button for the customers who order online
- Online products have been categorized according to the type of the cakes
- It has a search button on the online Items page to search items easily
- It provides more details on a product like weight, number of serves and price according to weight value. These details are provided at the click of a product. Therefore it is easy for the customers to order the correct amount for the number of serves needed. (figure 1.3)
- It has links to go to cake pages on click of 1,2,3.... First, Previous, Next and Last
- Well detailed contact page is available with a map to find the location

Weaknesses

- × Register and sign in links are not prominent. It is not that much visible because of the small font used on the link
- × Company Name has not been typed in a stylish manner in bigger font to make it prominent
- × More space has been occupied by the background image

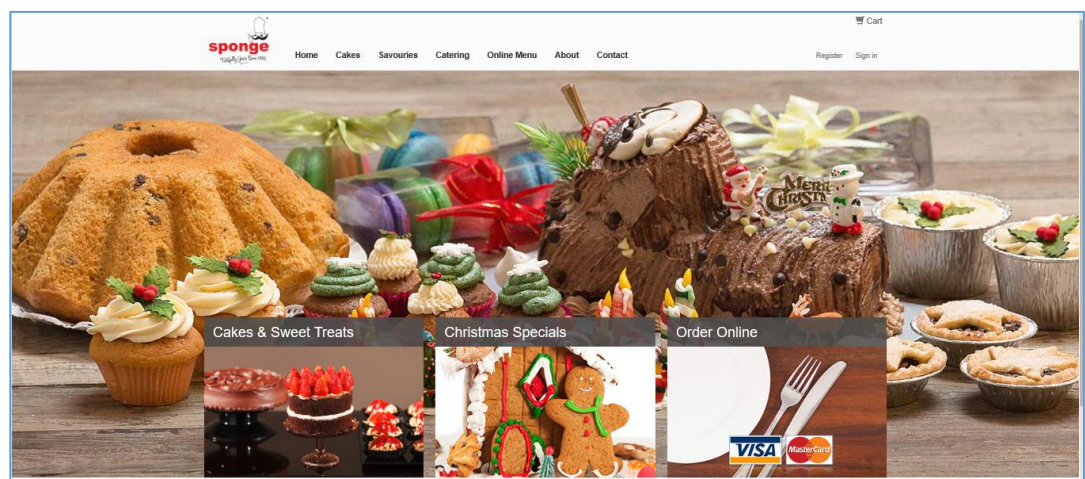


Figure 1.1 – Sponge web site home page

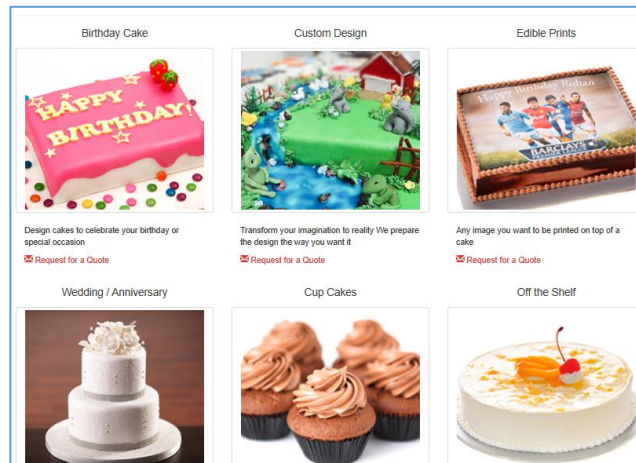


Figure 1.2 – Sponge web site category page

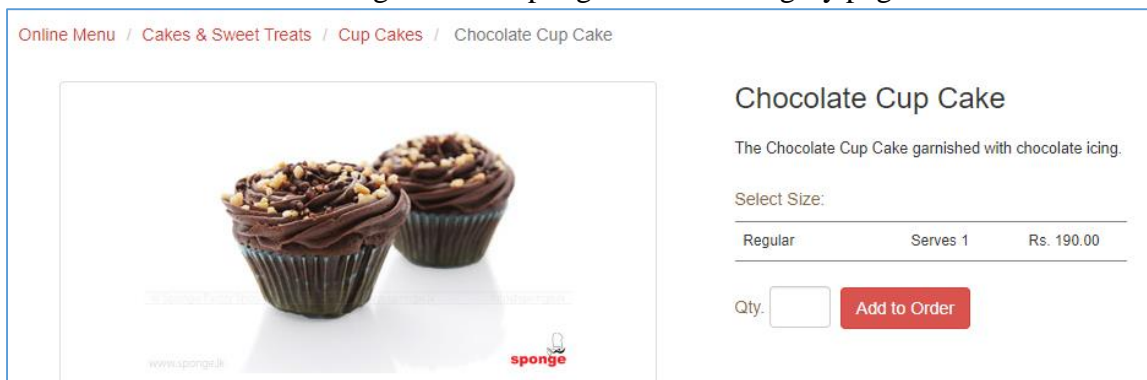


Figure 1.3 – Sponge web site item details

2.2.2. Pizza Hut Sri Lanka

This web site focus more on delivering items island wide. Therefore it has included delivery data entering option on home page itself. It first asks for delivery locations and pick up locations before making any type of item selections for the orders. It provides most popular deals on home page attracting the customers for new deals. It has item categories, sign in & privacy policies also. This web site is more professional and advanced than the requirements of Yummyshome system. But it provides ideas and information for a food ordering system. [2]

Strengths

- The ordering options Delivery or Takeaway has been provided on the home page itself. As the main interest of the people is ordering it is very convenient for the users (figure 1.4, figure 1.5)
- Deals can be accessed very quickly from the large navigation button provided to access the deals on the home page
- Most popular deals have been displayed in large fonts, It also provides a link to other deals
- While browsing the products it displays shopping cart on the right for the users to add the items to the cart easily

- Signup/Register page provides several ways to sign up as guest, facebook, google or account This is a more flexible approach for signing up
- Customization of items can be done easily by selecting the options from the given list. It has a free typing text box to write any special instructions needed

Weaknesses

- × A search button has not been provided to search a specific product known by users quickly. It needs to go through the provided categories to go to a known product also.
- × It has not provided the details of the outlets available
- × It does not provide much support for the users who want to go through the products and then visit the outlets to dine in.
- × Carriers button is available. But it does not navigate to a page to view the carriers

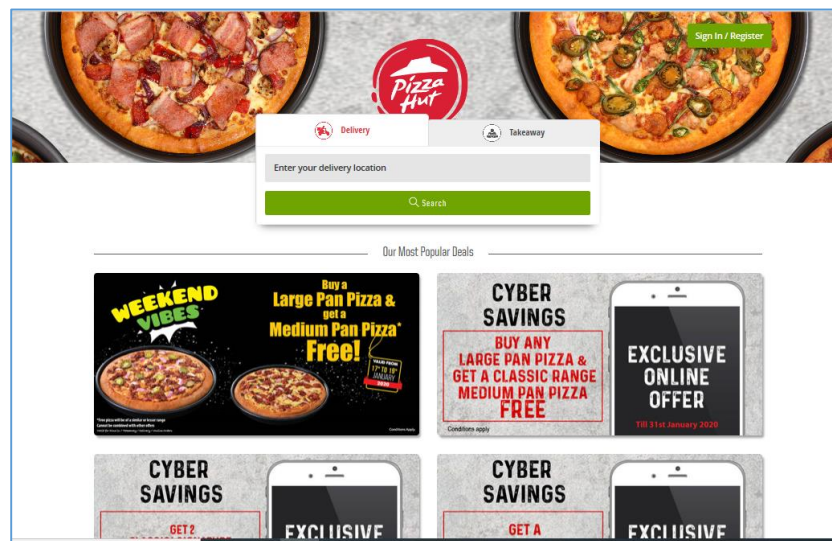


Figure 1.4 Pizza Hut Sri Lanka Home Page

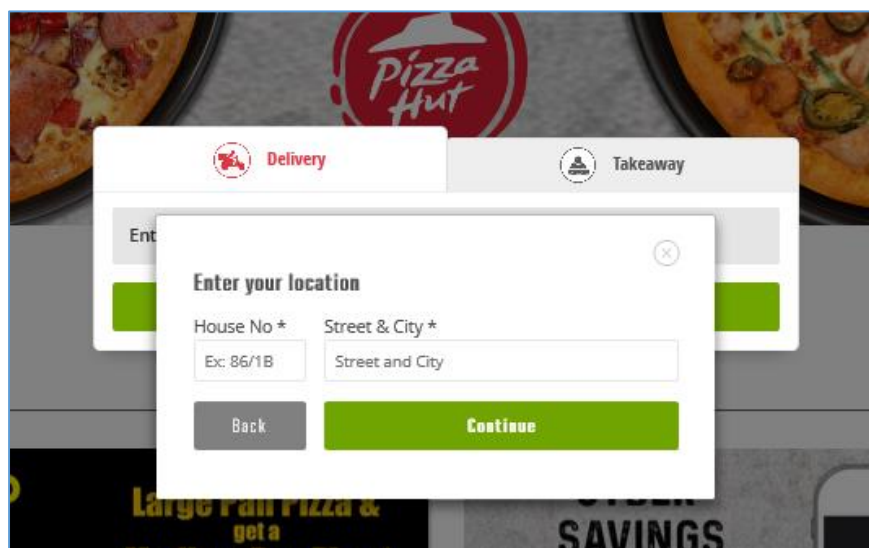


Figure 1.5 – Pizza Hut Sri Lanka Home page-Delivery data

2.2.3. The Cakery

The Cakery has Home, products, about us, sit down menu, catering, offers, sign up and blog. Products are categorized as layer cakes, logs, loaf cakes, flat cakes, designed cakes e.t.c. with images. About us page has contact details, about the business, the map and messaging facility. Special offers page has details about present offers [3].

Strengths

- It's a simple design
- Available options are clearly given on the home page (Figure 1.6)

Weaknesses

- × It has used more space on the home page for contact details and company name
- × Scrolling is needed below the company details to see products and services of the company
- × It is annoying to scroll for the product categories while half of the page is permanently covered by the company details section (Figure 1.7)
- × Colour combination used for the site has made it less attractive
- × The font used in the “Sit Down Food Menu” page is not clear. The font makes it difficult to read the contents. There is no navigation button in the menu

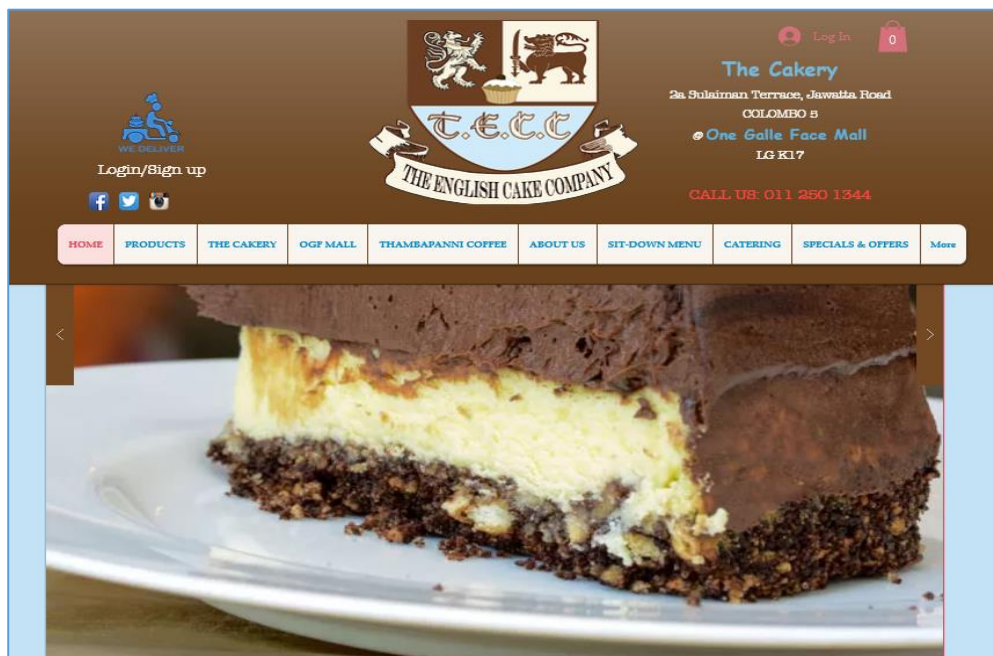


Figure 1.6 – The Cakery Home page

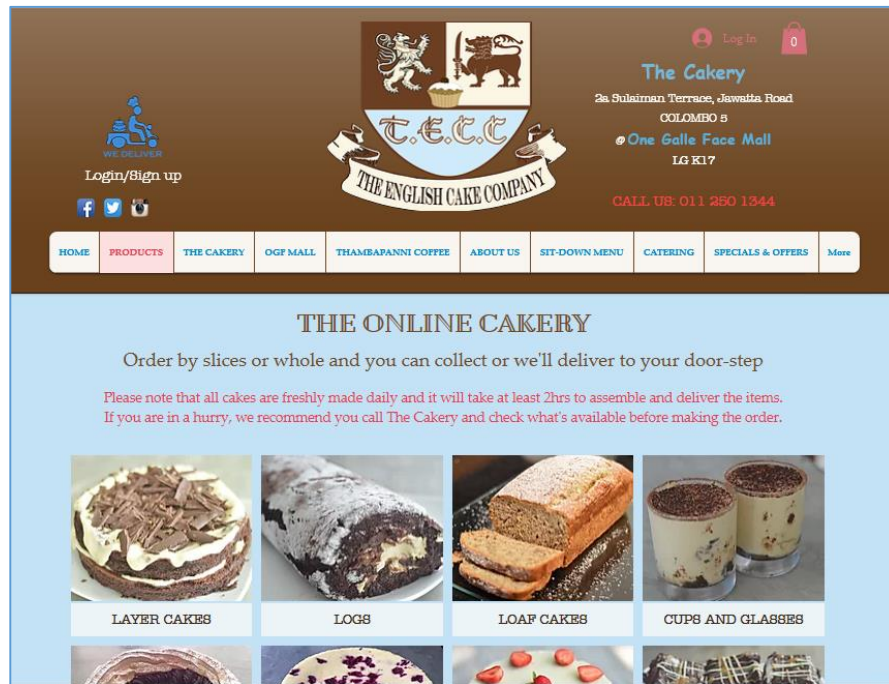


Figure 1.7 Cakery Item page

2.3. Useful Facts found from the similar systems

- According to above systems well categorized item pages provide faster and easier access to items. Therefore Yummyshome system can use simple category and item pages with basic details.
- Simple sign up pages are appealing to get more customers to register. Therefore signup pages of Yummyshome system can be made simple and easy to fill avoiding unnecessary details
- Navigation from page to page should be straight forward. Buttons and links of Yummyshome system should be made easily recognizable with a consistent colour and size.
- Home page of “Yummyshome” should be made attractive and company name should be made prominent by using larger and clear font.
- Contact pages should provide information for the customers to build their trust on the business. Therefore address, email address and telephone numbers would be provided in Yummyshome contact page.
- Variety of payment methods help the customers to choose between their preferences to make payments. “Yummyshome” can consider about providing several online payment methods in future.
- Social media sharing options can be made available on “Yummyshome” web site.

2.4. Technologies & Design Strategies

2.4.1. Technologies Considered

The following technologies available for developing the system were considered [4,5]

2.4.1.1. Responsive Web Site

Responsive web sites respond according to the type of the device like mobile phones and tablets. Devices can have different screen sizes, orientations, platform of the device and different features like touch sensitive screen. The layout of the responsive web site changes according to the capabilities of the device it is opened on. A mobile site can be launched faster than a developing other applications. It is also cheaper than designing a mobile app. But it takes a longer time to navigate using a mobile site compared to an app. Therefore the user experience on a responsive web site may not be good.

2.4.1.2. Mobile App

Mobile apps are applications designed to run on devices like smart phones and tablets. Mobile apps generally contain limited functions. There are three types of mobile apps called native, web based and hybrid. Native apps run on specific hardware. Web based ones use internet connectivity to provide most functions or all the functions.

Mobile apps have become popular as there are large number of smart phone users. Many sites provide mobile apps to reach the vast number of mobile users. Mobile apps have the ability to integrate with tools and features of a device to provide good user experience. It can use features like the camera, GPS and location services. It is easy to send location based material to the users. Loading time of a mobile time faster because they do not have unnecessary site elements.

Mobile apps takes more time to develop and it can be more expensive. Developers have to consider the restrictions on storage, memory and speed when developing mobile apps.

2.4.1.3. Mobile Friendly Web Site

Mobile friendly websites work on different devices in the same way. Mobile friendly sites do not use the features like list boxes, combo boxes and drop down lists much as it is difficult to operate those on mobile devices. Mobile friendly web sites generally have static contents. It provides simple navigation methods. Images are made smaller. It does not rely on the operating system of the device for its function. The user does not have to adjust settings manually to view the web site it is automatically adjusted for the device. Many people now search for information and shopping sites using their mobile devices. Specially items like food people search on the way. Therefore mobile friendly web site is a good way to reach large number of customers

2.4.2. Comparison of the Technologies

A mobile app can be run on specific mobile phone operating systems for which it is designed by the developer. A mobile app is suitable when there is a requirement to cater to large number of mobile phone users. But designing a mobile app needs special skills and time to develop. Mobile app can only be accessed from a device like a smart phone or

tablet. But a responsive website can be accessed from any device connected to the internet. Mobile apps should be downloaded and installed. But responsive sites no installation required. It is not an easy task to maintain and bug fixing in a mobile app. But in a web site it is an easy task. Anyway we cannot forget the fact that many users use mobile phones for searching for food outlets. Therefore it is important to cater to the mobile phone users also through a web site. A mobile friendly web site can reach the users of other devices as well as mobile phone users.

2.4.3. Chosen Technology

Mobile friendly web site

Justification

This is considered as suitable because Yummyshome is not going to interact with a very large number of mobile phone users and also developing customer loyalty is not a major concern of Yummyshome system. Mobile friendly web site is cost effective and the time taken to develop would be less. A mobile friendly web site can reach the users of other devices as well as mobile phone users. Site maintenance will be easier for a business like Yummyshome.

3. Methodology

3.1. Alternate Solutions

3.1.1. Alternate Solution 1 – ASP.Net

ASP.net is a product developed by Microsoft. ASP.Net can be considered as a server side web development framework. It is designed to work with HTTP internet protocol. It was released in the year 2002 after Active Server Pages of Microsoft. Compared to other applications, set up and configuration requirements are very less in ASP.Net.

It is a language independent software. ASP.Net code can be written in several languages like C#, Visual Basic, Java script and J#.

It has provided faster way to combine HTML with server side code. It is of light weight. It has set of libraries including web library that provides a lot of facilities for web applications. .Net applications can be executed on CLI(Common Language Infrastructure). It uses code behind mode to separate code from the design. This provides security for the code. ASP.Net uses caching to improve the performance of an application.

ASP.Net code are compiled and then executed. It makes the programs fast. It is easy to find and correct errors during the compilation process. .NET has been considered more robust in security since everything is built-into the framework and also it's backed by reputed Microsoft Company. This provides a sense of trust for the developers. [6,8]

3.1.2. Alternate Solution 2 – Laravel

Laravel is a PHP based open source web-framework used to develop web applications easily. It can be considered as a server side PHP web framework. It has many built in features that makes it easy to use. It has built in libraries to help to develop the applications. Functionalities can be easily added. It also has an automatic testing facility. Laravel's query builder helps to create database queries using simple chain method. Compared to other content management systems Laravel provides better control over the application you are doing. Laravel applications are more scalable. It provides coding facility to have better control of the application. The developers who have good knowledge in PHP find Laravel very easy to manage.

It can be a new platform for new developers. New developers may find problems for executing Laravel codes and classes. Community support is still less compared to other platforms. [7]

3.1.3. Alternate Solution 3 – PHP

PHP is a server-side scripting language that can be used for web development. It can be used to manage session tracking, managing dynamic content and web development. It is integrated with many popular databases like MySQL, SQL Server, Oracle, Informix and PostgreSQL. PHP code is interpreted by PHP interpreter that is implemented as a module in the web server. PHP is popular because of its simplicity efficiency, flexibility, security and familiarity. PHP has free open source libraries in the source distribution. It supports common protocols such as IMAP,POP3 and LDAP. It uses a syntax similar to C language. Developers familiar with C language find it easier. PHP runs on many platforms like Windows Linux and Mac. Community support for PHP is high. However error handling method in PHP is poor and also it is not good for larger projects. [8]

3.1.4. Alternate Solution 4 – CodeIgniter

Code Igniter is a PHP MVC framework. It is object oriented. It helps to develop applications faster than doing it from scratch. It has a rich library for common tasks. Libraries can be accessed through a simple interface. It has a simple and elegant toolkit to develop applications faster. It allows third party plugins to implement complex functions. It has encryption and good security features. Developers who are already familiar with PHP can master CodeIgniter easily. CodeIgniter is lightweight and is also simple to configure for specific requirements. However it does not have a strong collection of features like other modern PHP frameworks. Code maintainability of CodeIgniter can be difficult. [7,9]

3.2. The Chosen Solution

- ❖ Web Application with Microsoft ASP.Net with C#

Justification

- .NET is a general purpose platform used for development
- Tight integration with visual studio
- Stable code
- More focus on functionalities
- Great community get support
- More help and study material available on the internet for the developers
- .NET apps are easier to debug
- It is considered more robust in terms of security
- It's backed by Microsoft therefore it has a sense of trust.

3.3. How the present system works

3.3.1. Information and popularity of Yummyshome

Yummyshome has become popular among many customers by the verbal publicity given by the customers who have tasted yummy food. Friends of friends tend to order food after tasting food prepared by Yummyshome. Facebook page also helps to get orders from known and unknown customers who are on facebook.

3.3.2. Present ordering process

3.3.2.1. By a personal visit

Some customers pay a personal visit to Ms Rajeeva's place, go through her albums to pic designs of cakes, inquire prices discuss and finalize the designs, ingredients and prices.

3.3.2.2. Ordering over the phone

Customers call and get the information about items, ingredients, changes they need in design or ingredients, discuss prices and then place orders. The orders are written in a diary on the specific day of the delivery of items.

3.3.2.3. Order by Facebook messages

Some customers go through Facebook pictures uploaded. Send messages on their inquiries. Ms. Rajeeva go through the messages when she is free and reply to the queries. If anyone places an order she notes down it down in the diary.

3.3.3. Preparing summaries on item details and orders

Monthly income details, fast going items, orders made for a particular day are prepared by manually checking the details on the diary

3.4. Problems in the present system

- i). When customers visit the owners place to get information it disturbs her day today activities
- ii). Some customers take long time to go through images & discuss about ingredients e.t.c
- iii). When she is away from home she may not be having the order book to check the present orders and to take new orders
- iv). There is no special format on each page of the diary to record orders in a consistent manner
- v). It is difficult to search for information on ingredients and number of servers from each food item to answer to customer calls
- vi). The owner has to manually go through the diary to get summaries on monthly orders
- vii). She has to check the diary to see how many times a particular item has been ordered during a particular time period
- viii). When orders are deleted it has to be cut making the page untidy. Sometimes the space on a page is not sufficient to record all details needed.
- ix). Sometimes there are chances of going wrong in a picture selected by a customer because picture number in the album is manually written on the order page or the picture given by customer is clipped to the relevant page

3.5. Proposed System

3.5.1. A web based food ordering and management system

3.5.2. Justification of the proposed system

- Food items prepared by Yummyshome can be easily displayed on the web site with required details
- Customers can leisurely go through items displayed on the web site and choose the items they want easily
- Telephone calls to inquire about food items will be reduced
- Customers don't have to pay personal visits and spend on telephone calls to inquire about food items and their prices
- It is easy for the owner to go through order details
- The problem of unavailability of required data to answer customer queries is reduced
- Summary reports can be generated quickly
- Security of the system is improved as admin password is needed to enter details, edit details and delete details

3.6. Design

3.6.1. Use Case Diagram

Use case diagram is shown in figure 3.1

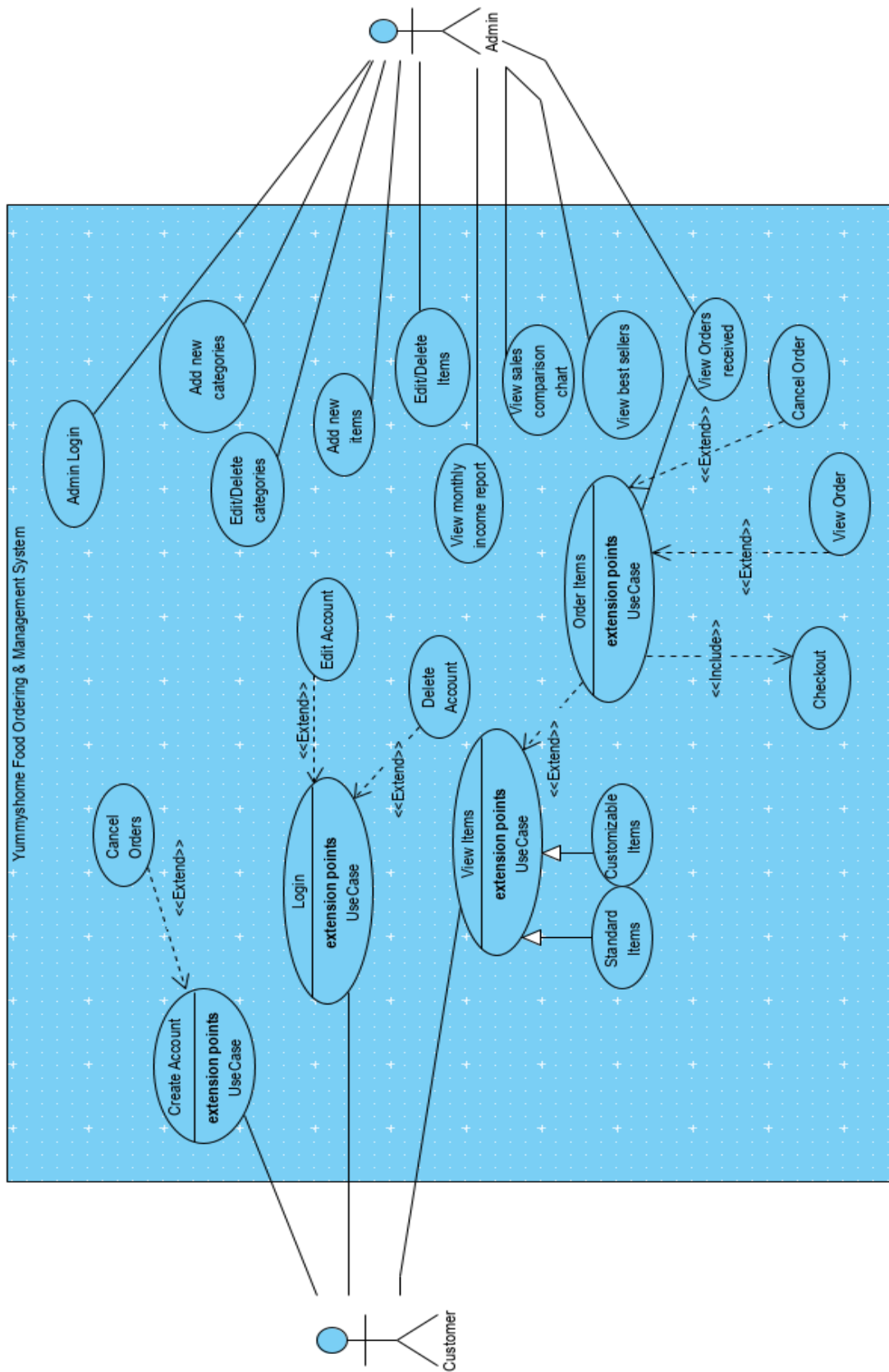


Figure 3.1 – Use Case Diagram

3.6.2. ER diagram

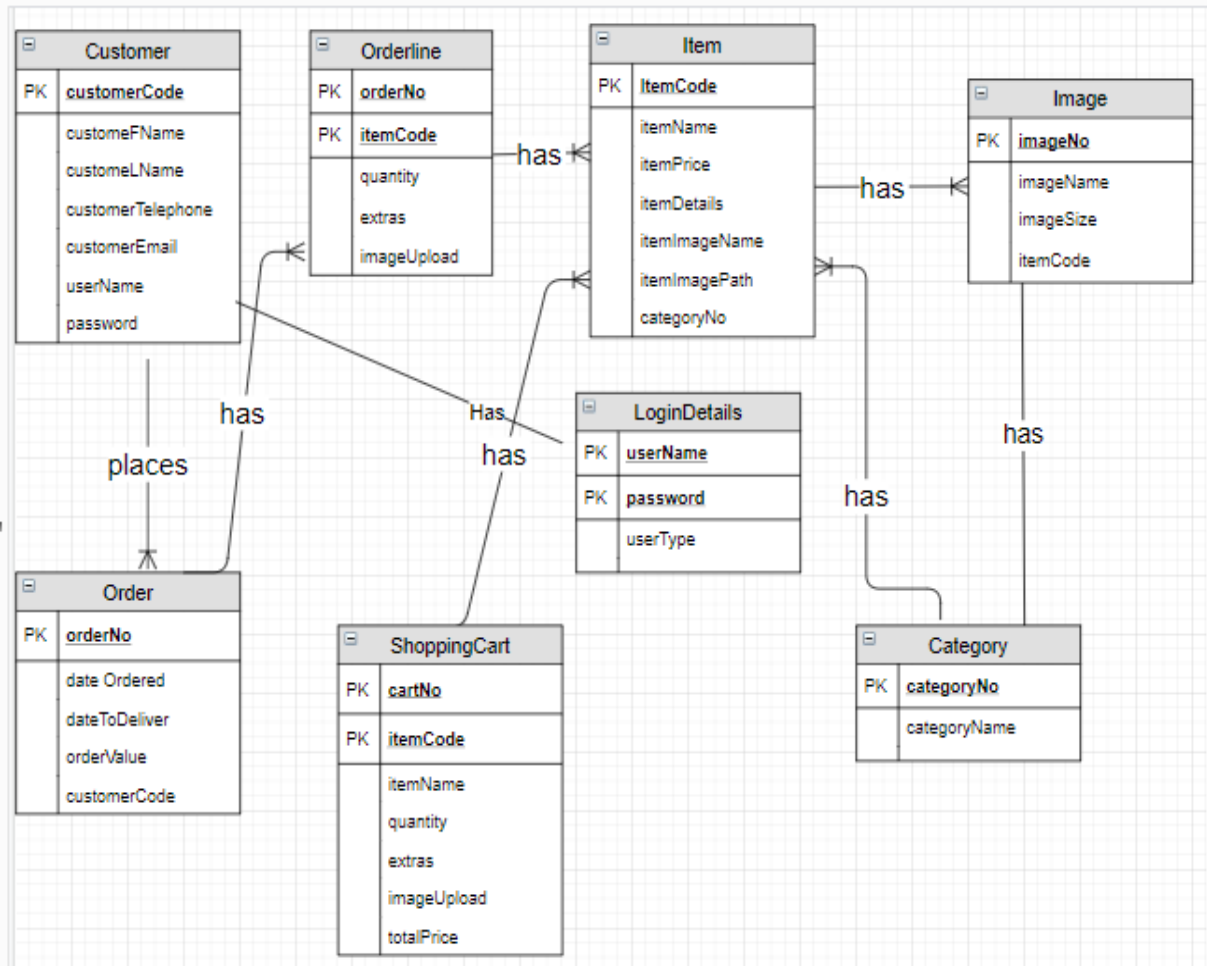


Figure 3.2 – E.R. Diagram

3.6.3. Sequence Diagram

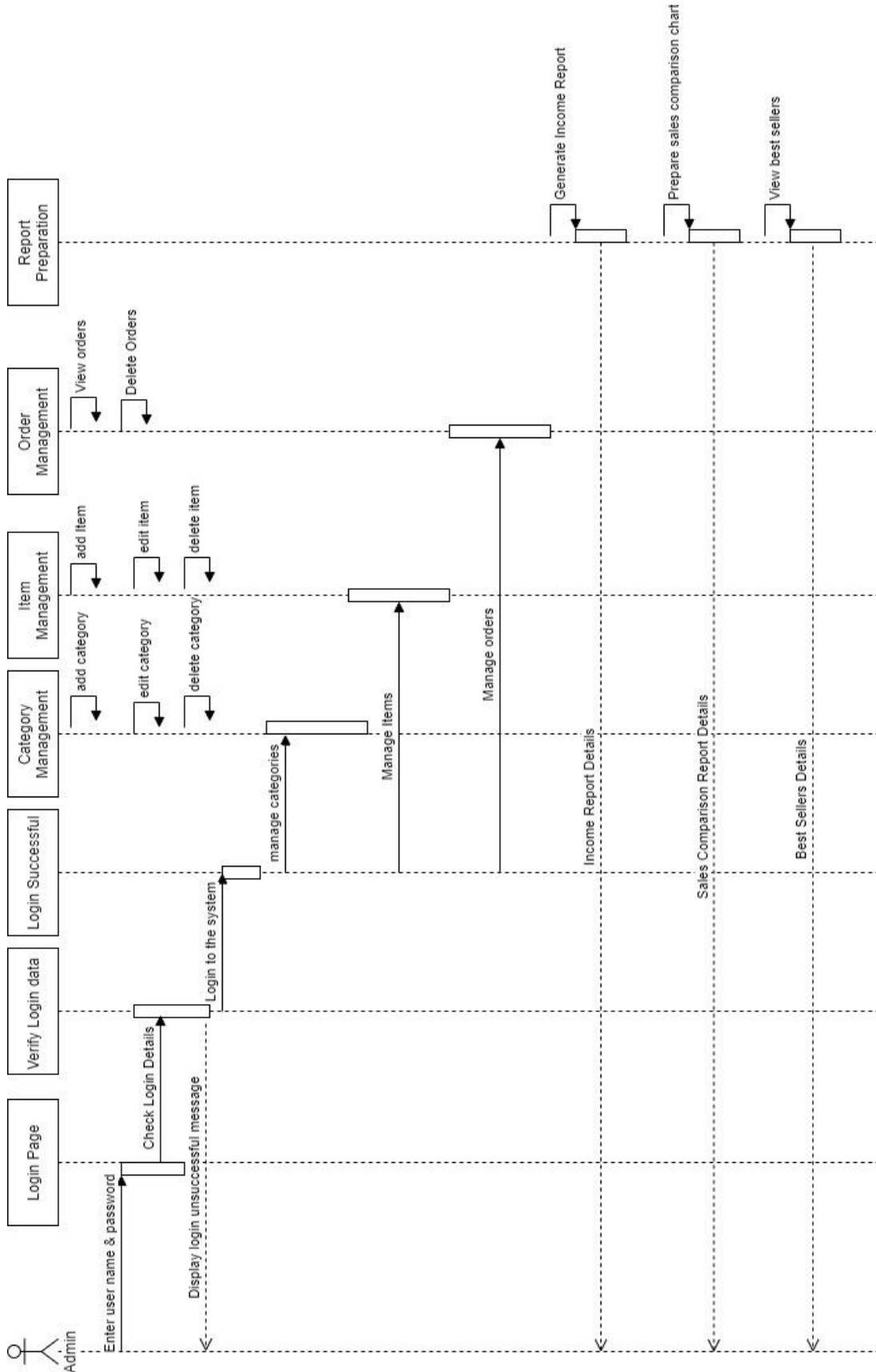


Figure 3.3 – Sequence Diagram

3.6.4. Logo Design



Yummyshome does not have a logo for the business at the moment. Therefore a new logo would be designed with a suitable image and the business name

Figure 3.4 – logo Design

3.6.5. Interface Design

3.6.5.1. Admin login form

The Admin Login form is enclosed in a blue border. At the top is a purple header box with the text 'Admin Login'. Below it are two input fields: 'User Name' containing the text 'admin' and 'Password' which is empty. At the bottom are two pink buttons: 'Cancel' and 'Sign in'.

Changes to the data in the database would be allowed only through the admin login. Therefore an admin login is needed. It provides security for the data

Figure 3.4.1 – Admin Login form

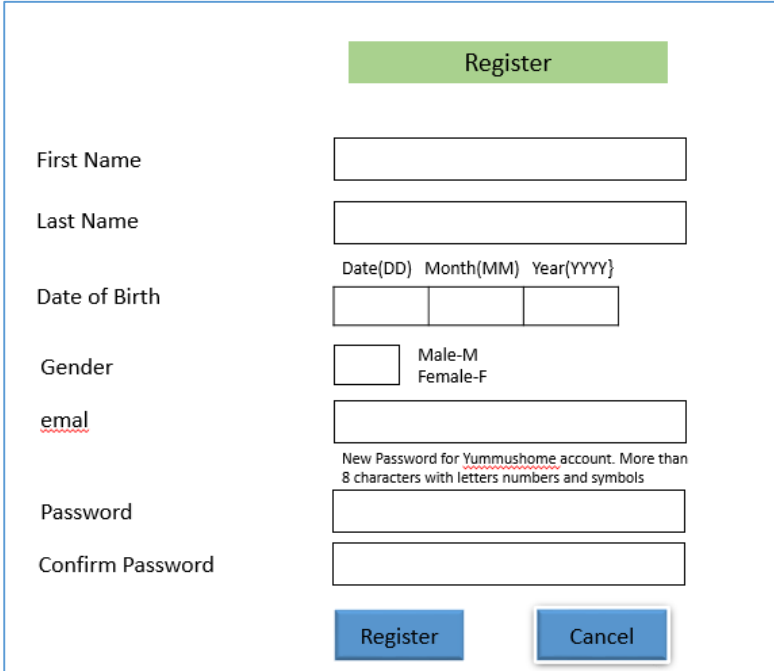
3.6.5.2. User login form

The User Login Form is enclosed in a blue border. At the top is a green header box with the text 'User Login Form'. Below it are two input fields: 'email' which is empty and 'Password' which is empty. At the bottom are two blue buttons: 'Cancel' and 'Sign in'.

Registered customers get special offers and discounts. Therefore a customer login has been designed. They can log in with their email address and the password created for Yummyshome

Figure 3.4.2 – User Login Form

3.6.5.3. Customer Registration form



The registration form includes a green 'Register' button at the top. Below it are input fields for First Name, Last Name, and Date of Birth (split into DD, MM, and YYYY). The Gender section has radio buttons for Male (M) and Female (F). There are fields for email, Password, and Confirm Password. A note specifies: 'New Password for Yummushome account. More than 8 characters with letters numbers and symbols'. At the bottom are blue 'Register' and 'Cancel' buttons.

The details shown in the design are entered through this interface to the database

Figure 3.4.3 – Customer Registration Form

3.6.5.4. Home Page

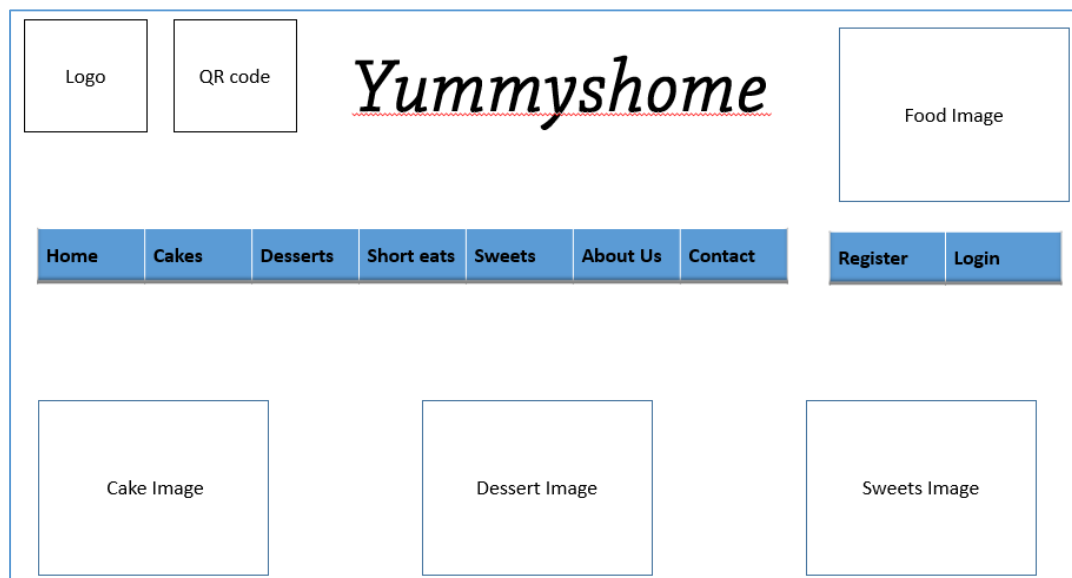


Figure 3.4.4 – Home Page Design

The home page contains company name in prominent visible font and font size. All important buttons to navigate to the required pages are displayed

3.6.5.5. Cake Category Page

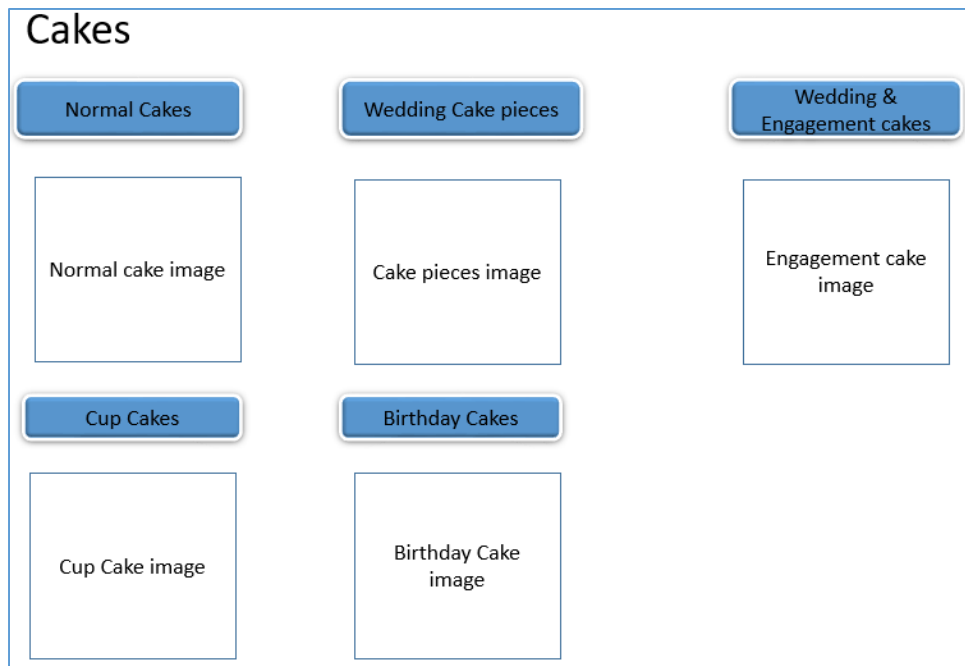


Figure 3.4.5 – Cake category page design

There are several categories of cakes therefore all categories are displayed on cake categories page with a suitable picture. Navigation buttons are used to allow navigation to each category

The above format would be used for all item category pages

3.6.5.6. Regular Cakes Page

Regular Cakes

Cake Image 1	Cake name Description and price	Cake Image 2	Cake name Description and price
Cake image 3	Cake name Description and price	Cake image 4	Cake name Description and price
Cake image 5	Cake name Description and price	Cake image 6	Cake name Description and price
Repeat the above for all cakes			

Figure 3.4.6 – Regular cake items page design

All normal cakes are displayed on Normal cakes page. Customers can get all details of each cake from this page. If the customer wishes to order he can add the items to the order

3.6.5.7. Desserts Page

Desserts			
Dessert Image 1	Dessert name Description and price	Dessert Image 2	Dessert name Description and price
Dessert image 3	Dessert name Description and price	Dessert image 4	Dessert name Description and price
Dessert image 5	Dessert name Description and price	Dessert image 6	Dessert name Description and price
Repeat the above for all desserts			

Figure 3.4.7 – Dessert items page design

Desserts page display all desserts and their descriptions. If the customer wants to add products to the order he can add from this page

The above format would be used for all item pages

3.6.5.8. Custom Birthday cakes page

Custom Birthday Cakes

Provide following details we would contact you with price & other details

First Name

Last Name

email

Contact Number

Type of the cake

Number of servings

Date required

image

Description

Figure 3.4.8 – Custom Birthday cake page design

3.6.5.9. Test Plan

Test No.	Test Component	Inputs	Actions	Expected Output
1	Home Page & Category pages			Home page displayed. Category page displayed when a category button is clicked
2	Customer Registration form	Customer FName, CustomerLname, CustomerTeleph one, CustomerEmail, UserName, Password	Submit	Customer details are added to the customer table
			Cancel	Clears the text boxes in the registration form
3	Login form	User Name	Sign Up	

		Password		
			Cancel	
4	Add/Edit/Delete Categories	CategoryNo CategoryName	Add	Enters data to the category table
			Edit	Edit & update category table
			Delete	Delete the selected record from the table
5	Add/Edit/Delete Items	ItemCode, ItemName, ItemPrice, ItemDetails, CategoryNo, ItemImageName, ItemImagePath	Add	Enters Item details in the item table
			Edit	Updates a record in the item table
			Delete	Remove a record from the item table
6	Add/Edit/Delete Offers	ItemCode, ItemName, ItemPrice, ItemDetails, CategoryNo, ItemImageName, ItemImagePath Discount Rate	Add	Enters Item details in the item table
			Edit	Updates a record in the item table
			Delete	Remove a record from the item table
7	DisplayOrderRegularCakes	Select Item	Click on Select Item button	Item gets selected from the grid view
		Quantity	Enter quantity	User can type quantity in the text box
			Show total price	Total price for the given quantity is displayed
			Add item to order	The selected item is added to the shopping cart table
			View Cart	Displays item details in the shopping cart

			Shop More	Navigates to category page
8	Checkout	Delivery Date Name Surname Telephone Email Payment method	Checkout	Displays order details and email sent confirmation
9	DisplayOrderEngagement cakes	Same as Test component 5		
10	DisplayOrderBirthdayCakes	Same as Test component 5		
11	DisplayOrderWeddingCakes	Same as Test component 5		
12	DisplayOrderDesserts	Same as Test component 5		
13	DisplayOrderSweets	Same as Test component 5		
14	DisplayOrderShorteats	Same as Test component 5		
15	View Orders		Click on by date	Displays orders for a particular date
			Click on by Week	Displays orders for a particular week
			By Month	Displays orders for a particular month
16	View Reports			

Table 3.1 – Test plan

4. Evaluation

4.1. Functionalities

4.1.1. Functionalities for web site maintenance & administration

The site has a login system. When the admin logs on as admin with admin username and password he is directed to the “site maintenance and management” page. In that page following features are available for the admin. They are

Login

Add/ Edit/ Delete/ View categories

Add/ Edit/ Delete/ View items

View Orders

View Reports

4.1.2. The functionalities available for customers are

- Register – When the customer clicks on Register Button it navigates to register page. When the customer has finished entering details he can click the Register button on the registration page. Then the details get saved in the customer table.
- Login – The customers can login to the site by entering user name and password. The user name and password are checked with the information in the “LoginDetails” table.
- View Item Categories

Cakes page has sub categories called Regular cakes, Birthday Cakes, Wedding Cakes & Engagement Cakes and Cup Cakes. Other Item pages do not have sub categories.

- View Item details and images
When the user clicks on a button that has the name of a particular item category user is navigated to the required item page. The item page displays available items in a grid view
- Select an item to order
Each row of the grid view has “select to order” button to select the required item. The selected item details are displayed.
- Enter required quantity of the item and view price for the given quantity-
The user has to enter the required quantity of item in Kg. Then the price for the given quantity can be viewed by clicking on the “View Total Price” button.
- Add item to the order
After that the user can click on “Add Item to the shopping cart”. Then the item details are entered in the shopping cart & cart number is remembered in a session variable.
- Shop More
If the customer wants to shop more he can click on “Shop More” button to go to category pages
- Remove an item from the order
- View Items in the order
- Cancel order
- Confirm order and checkout

- Checkout page displays the items in the shopping cart and checkout details input form. Then the customer has to enter personal details and other order details. Final details of the order is displayed. After entering the required details the user can checkout.
- Share the site on Facebook and Whatsapp in mobile phone using the share buttons on the home page

4.2. Problems encountered

The main problem faced was the software runs slow and it took more time to develop because of this reason.

There are many ways to develop certain features. Therefore it took time to refer and choose the best way to develop the features in my project.

It is time consuming to include many validation checks to avoid errors and make the system robust.

Long lines of coding takes a longer time to develop even though a unique solution can be created. Similar functionalities and attractive web sites can be created using new systems like content management systems in less time.

4.3.Changes made to the design

Instead of creating two login pages one login page was created. User type has been included in the “LoginDetails” table to recognize user.

Initial design of placing item images and details on pages was changed to match with the grid view output of ASP.Net as it is easier to implement it that way.

Images table designed to record image names and locations was removed and images were named in a way to recognize its type from the name itself. For examples birthday cake 1 is named BCake1 and Sweets item 2 is named SWT2.

Order confirmation email added

4.4. Testing

Test No 1 - Home page & Category Pages

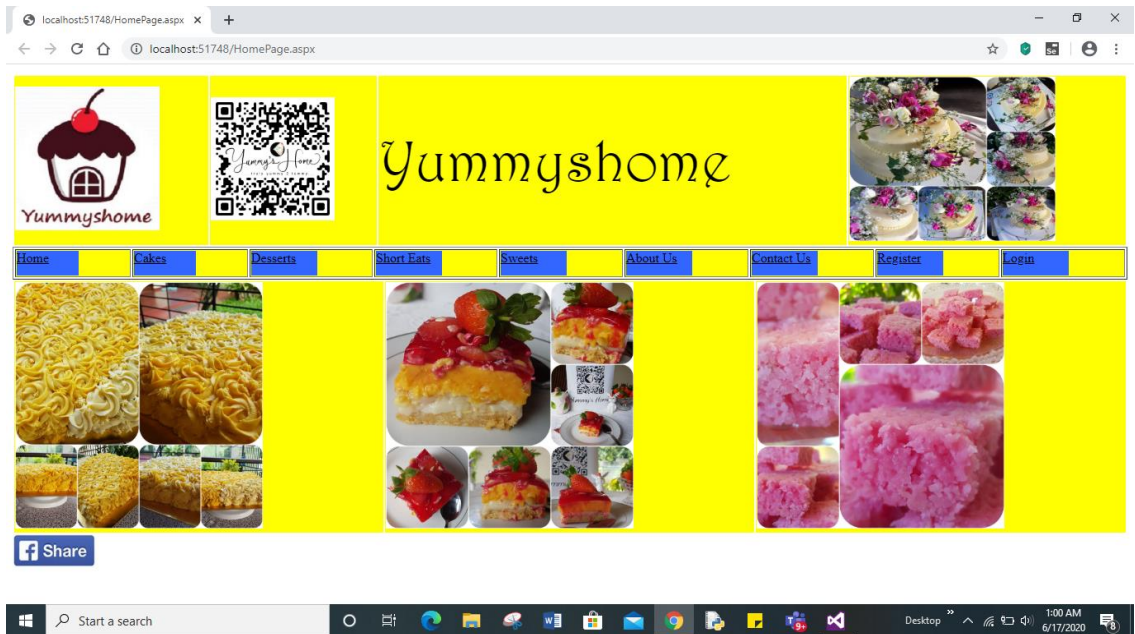


Figure 4.1.1 – Home page test result for bigger screen

Home Page Whatsapp share button for mobile users.
Displayed only when screen is small



Figure 4.1.2 – Home page test result for small screen

Comment – Home page displayed accurately

Cake category page test

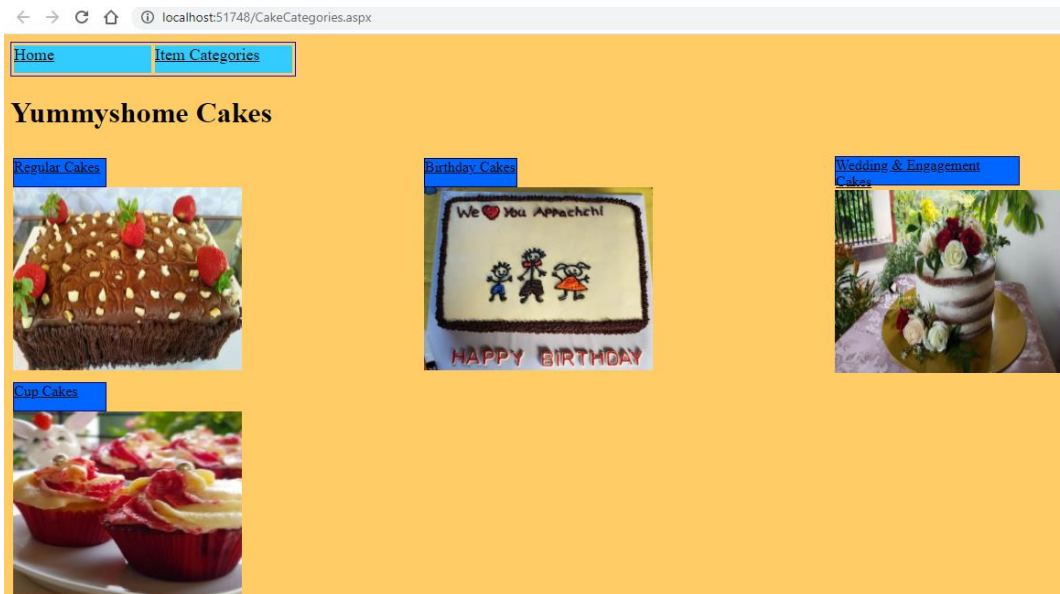


Figure 4.2 – Cake category page test result

Comment – Cake Categories page has been displayed correctly

Test No 2- Customer Registration form

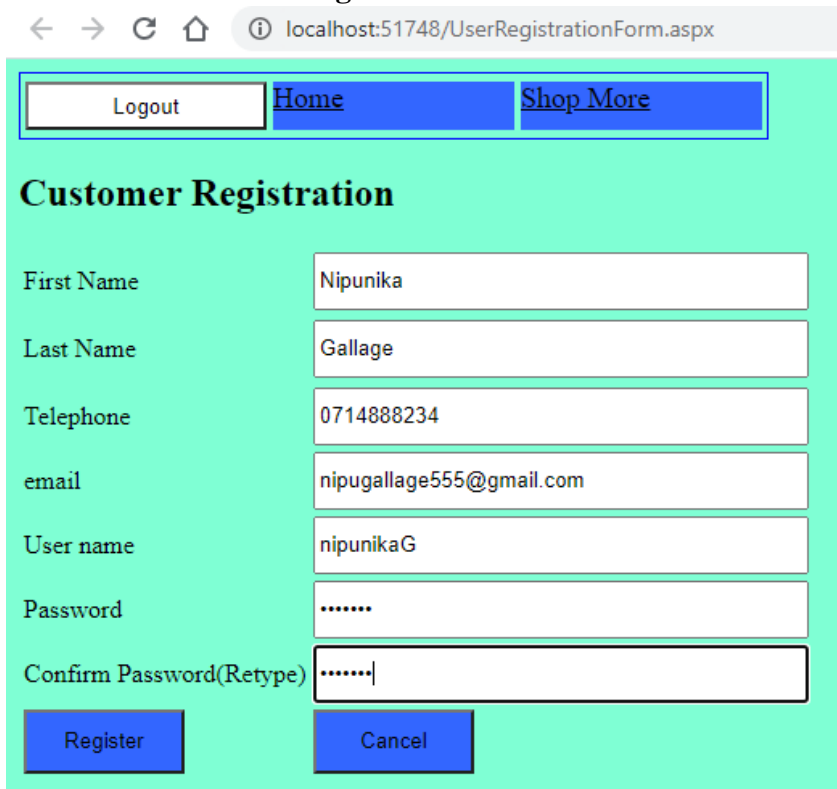


Figure 4.3.1-Customer registration form test result

Record entered in the Customer table

47	Jayani	Kothalawala	1234567890	sarah.kothalaw...	NULL
48	ffff	cccccccc	1234567	sarah.kothalaw...	NULL
49	ssssss	jjjjjjjjj	2222555567	sarah.kothalaw...	NULL
50	Mithila	Gunasekera	077345123	mithila@gmail...	Mithila
51	abcccc	ttyyu	1234567890	sarah.kothalaw...	NULL
52	aaaa	aaaaa	1234567890	aaaa	aaaa
53	Sarah	Perera	0714786453	sarah.kothalaw...	sksarah
54	Sarah	Perera	0714000822	sarah.kothalaw...	NULL
1053	Nipunika	Gallage	0714888234	nipugallage555...	nipunikaG

Figure 4.3.2 – Customer table details test result

Comment – Customer details are entered in the customer table

Test number 3 - Login Forms

Admin Login

The figure displays two side-by-side screenshots of a web application's login form titled "Login to Yummyshome".

The left screenshot shows the form with the following state:

- User Name: admin
- Password: [masked]
- Buttons: Cancel and Login

The right screenshot shows the form with the following state:

- User Name: admin
- Password: [empty]
- Buttons: Cancel and Login
- Error message: Check your user name and password (displayed in red text below the Password field)

Figure 4.3.3 Admin login test result

Comment – Login approved and Admin panel is displayed if correct user name and password are entered. Error message is displayed if the user name and password are incorrect.

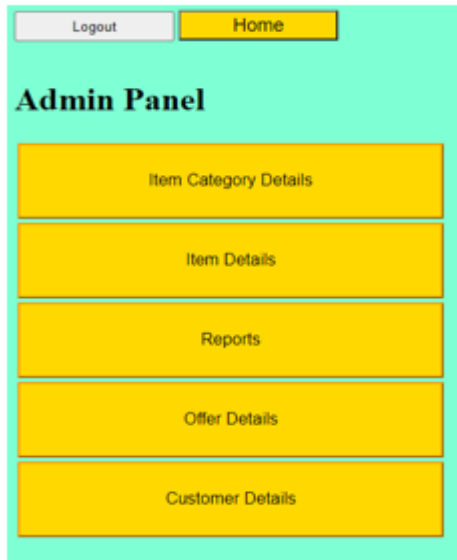


Figure 4.3.4 – Admin login redirection page test result
 Comment – If login is successful admin user is navigated to Admin Panel

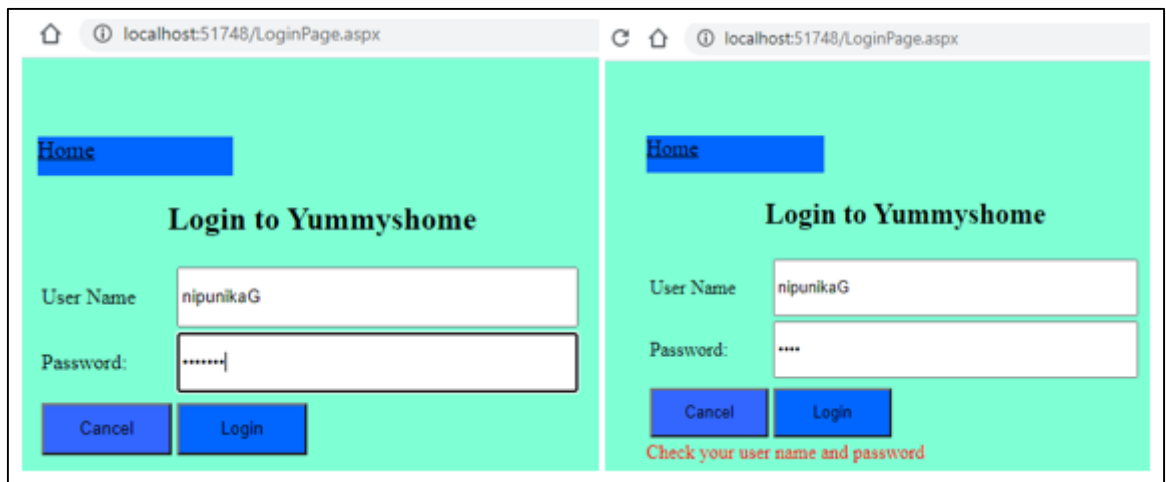


Figure 4.4.1 – User login form test result

Comment – User name and Password entered to Login table as expected. Offers page displayed

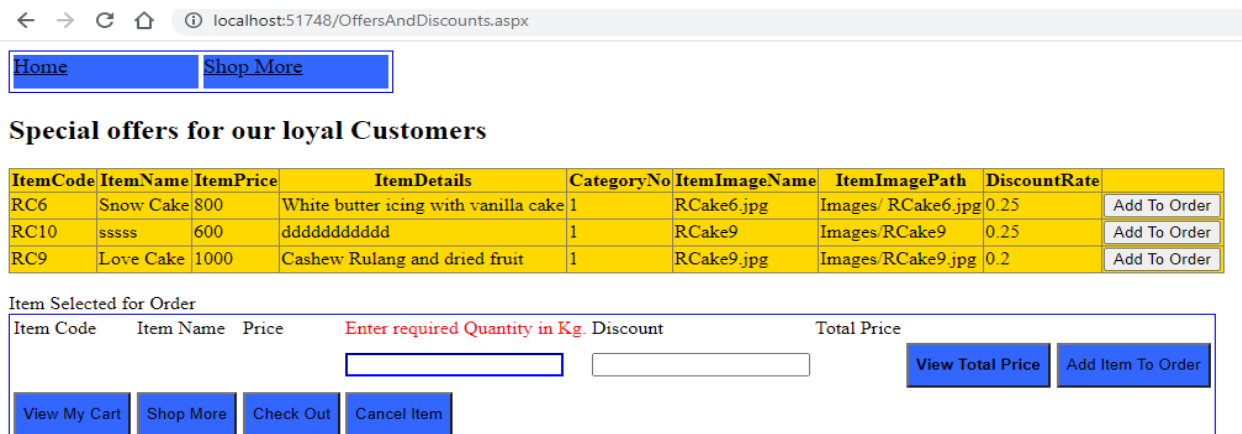


Figure 4.4.2 - Special Offer page for customers test result

Test No. 4 - Add/Edit/Delete Categories

localhost:51748/AddEditDeleteCategories.aspx

Logout **Admin Panel**

Add/Edit/Delete Categories

	<u>Category Number</u>	<u>Category Name</u>
Edit Delete	1	Regular Cakes
Edit Delete	2	Engagement Cakes & Wedding Cakes
Edit Delete	3	Birthday Cakes
Edit Delete	4	Cup Cakes
Edit Delete	5	Desserts
Update Cancel	6	Sweets <input type="text"/>

Enter New Category Details

Category Number Category Name

Figure 4.5.1 – Add edit delete categories test result

Comment – Items category added to the ItemCategory table

Categories in Category table

CategoryNo	CategoryName
1	Regular Cakes
2	Engagement C...
3	Birthday Cakes
4	Cup Cakes
5	Desserts
6	Sweets
NULL	NULL

Figure 4.5.2 – Categories in table test result

Comment – Item category data is entered in the Category table

Test No. 5 - Add/Edit/Delete Items

Logout [Admin Panel](#)

Add Edit Delete Items






	ItemCode	ItemName	ItemPrice	ItemDetails	CategoryNo	ItemImageName	Image
Edit Delete	BC1	Round Natural Rose	2000	Chocolate cake with butter icing. Natural flowers	3	Bcake1.jpg	
Edit Delete	BC12	Tennis Court	2500	Butter Icing with parchment	3	Bcake12.jpg	
Edit Delete	BC13	Angel Cake	2500	Butter Icing with artificial ornament decor	3	Bcake13.jpg	
Edit Delete	BC6	Road Fancy Cake	1500	Butter cake & butter Icing	3	Bcake6.jpg	
Edit Delete	DST2	Strawberry Delight	2000	Strawberry Trifle with jelly	5	Dst2.jpg	

Figure 4.6.1 – Add edit delete items test result

dbo.Item [Data] AddDisplayEditDeleteCategory.aspx NewItemDataInputWith GridView.aspx NewItemDataInputForm.aspx

Max Rows: 1000

	ItemCode	ItemName	ItemPrice	ItemDetails	CategoryNo	ItemImageNa...	ItemImagePath
	BC1	Round Natural ...	2000	Chocolate cake...	3	Bcake1.jpg	Images/Bcake1...
▶	BC6	Road Fancy Cake	1500	Butter cake & b...	3	Bcake6.jpg	Images/Bcake6...
	EC1	2 Layer Natural ...	7500	Butter Cake wit...	2	Ecake1	Images/Ecake1
	RC2	Butter Lover	1000	Butter Cake wit...	1	Rcake2.jpg	Images/Rcake2...
	RC5	Chocolate Cake	500	Chocolate Icing	1	Rcake5.jpg	Images/Rcake5...
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Figure 4.6.2 – Item table details test result

Comment - Item added to the table correctly

Test No. 6 - Add/Edit/Delete offers

Logout [Admin Panel](#)

Add/Edit/Delete Offers

	ItemCode	ItemName	ItemPrice	ItemDetails	CategoryNo	ItemImageName	ItemImagePath	DiscountRate
Edit Delete	RC6	Snow Cake	800	White butter icing with vanilla cake	1	RCake6.jpg	Images/RCake6.jpg	0.25
Edit Delete	RC9	Love Cake	1000	Cashew Rulang and dried fruit	1	RCake9.jpg	Images/RCake9.jpg	0.2

Enter New Item Details

Item Code Item Name Item Price Item Details

[Add Item](#)

Figure 4.6.3 – Add/Edit/Delete offers test result



Test No. 7 - Display & Order Sweets

Item selection for an order

← → ↻ 🏠 ⓘ localhost:51748/DisplayOrderSweets.aspx

[Home](#) [Item Categories](#)

Sweets

Item Code	Item Name	Details	Price per Kg. LKR	Image	
SWT1	Cononut Toffee	Strawberry Flavour	1500		Select To Order
SWT4	Milk Toffee	Condensed milk	2000		Select To Order

Item Selected for Order

Item Code	Item Name	Price	Enter required Quantity in Kg.	Total Price
			<input type="text"/>	View Total Price Add Item To Order

[View My Cart](#) [Shop More](#) [Check Out](#) [Cancel Item](#)

Figure 4.7.1 – Item selection for order test result

Comment – When Add To Order button of the item is clicked the item is successfully selected for the order

Enter required quantity and click View Total Price button.

Item Code	Item Name	Price	Enter required Quantity in Kg.	Total Price
SWT1	Cononut Toffee	1500	<input type="text" value="2"/>	3000

Buttons: View My Cart, Shop More, Check Out, Cancel Item, View Total Price, Add Item To Order

Figure 4.7.2 – Total price calculation test result

Comment – Customer enters the quantity required. Total price is calculated and displayed.

Customer clicks “Add Item To Order” button. Then item is added to the cart and it displays “Item added to the shopping cart”

Item Code	Item Name	Price	Enter required Quantity in Kg.	Total Price
SWT1	Cononut Toffee	1500	<input type="text" value="2"/>	3000

Buttons: View My Cart, Shop More, Check Out, Cancel Item, View Total Price, Add Item To Order

Item added to the shopping cart

Figure 4.7.3 – Add item to order test result

Comment – Item added to the order and message “Item added to the shopping cart” is displayed

Shopmore button click to add another item to the order

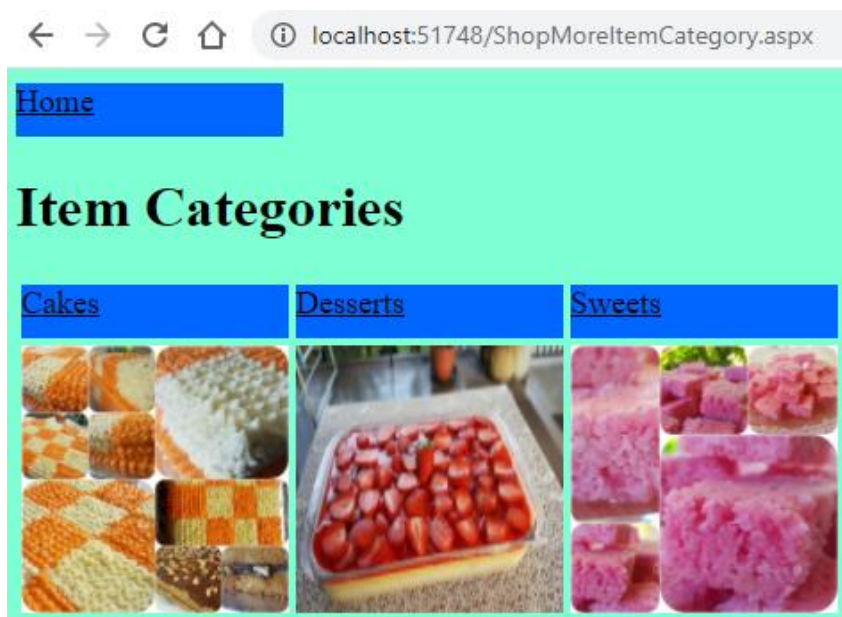


Figure 4.7.4 – Shop More test result

Comment – It navigates to Item Categories page when Shop More button is clicked

Adding 2nd item to the order

localhost:51748/DisplayOrderRegularCakes.aspx

Home Item Categories

Regular Cakes

Item Code	Item Name	Details	Price per Kg. LKR	Image	
RC2	Butter Lover	Butter Cake with butter icing	1000		Select To Order
RC5	Chocolate Cake	Chocolate Icing	600		Select To Order

Item Selected for Order

Item Code	Item Name	Price	Enter required Quantity in Kg.	Total Price
RC2	Butter Lover	1000	<input type="text" value="3"/>	

View My Cart Shop More Check Out Cancel Item View Total Price Add Item To Order

Item added to the shopping cart

Figure 4.7.5 – Add second item test result
When Checkout button is clicked Customer is navigated to the checkout page

localhost:51748/Checkout.aspx

Home Shop More Checkout

Checkout

Items in your cart

ItemCode	ItemName	Quantity	TotalPrice
RC2	Butter Lover	3	3000
SWT1	Cononut Toffee	2	3000

Enter Delivery Data

Delivery Date MM/DD/YYYY

Name

Surname

Telephone

email

Payment Method

Order Confirmation

```
Date :11/19/2020 7:49:34 PM
Order Number: 1062
Customer Name : Sarah
System.Web.UI.WebControls.TextBox
Item Number :RC2
Item Name : Butter Lover
Quantity : 3
Price :Rs.3000
Item Number :SWT1
Item Name : Cononut Toffee
Quantity : 2
Price :Rs.3000

Order Total Rs.: 6000Payment Option :
Bank Deposit
Bank : Peoples Bank Kelaniya Rajeewa
Kothalawala
AccountNumber :196-2-001-7-0045762
```

Order confirmation email sent successfully

Figure 4.7.6 – Checkout test result

Comment – Items in the shopping cart is displayed and customer data is requested

Order Added to the orders table

OrderNo	DateOrdered	DateToDeliver	OrderValue	CustomerCode
38	8/4/2020	8/11/2020	6000	26
39	8/4/2020	8/11/2020	4500	27
40	8/4/2020	8/12/2020	5400	28
41	8/4/2020	8/13/2020	2000	29
42	8/8/2020	8/13/2020	8000	30
43	8/11/2020	8/17/2020	9600	31
44	10/8/2020	10/12/2020	8000	34
45	10/10/2020	10/12/2020	4500	35
46	10/10/2020	10/13/2020	6000	36
47	10/10/2020	10/13/2020	10000	37
48	10/10/2020	10/12/2020	12500	38
49	10/10/2020	10/15/2020	2400	40
50	10/10/2020	10/16/2020	3000	41
51	10/10/2020	10/13/2020	6000	42
52	10/10/2020	11/18/2020	6000	43
53	10/10/2020	10/14/2020	3000	44
54	10/10/2020	10/15/2020	4000	45
55	10/10/2020	10/14/2020	4000	46
56	10/10/2020	10/18/2020	3000	47
57	10/10/2020	10/20/2020	4000	48
58	10/10/2020	10/16/2020	2400	49
59	10/11/2020	10/20/2020	10500	51
60	10/11/2020	10/20/2020	9000	54
1062	11/19/2020	11/25/2020	6000	1056

Figure 4.7.7 – Order added test result

Comment – Order added correctly to the Orders table

Items added to the OrderLine table

OrderNo	ItemCode	Quantity
43	DST2	3
43	SWT4	2
44	DST2	3
44	RC2	2
45	DST3	3
46	DST2	3
47	BC12	4
48	BC13	5
49	RC5	4
50	RC2	3
51	DST3	4
52	DST2	3
53	RC2	3
54	RC2	4
55	RC2	4
56	RC5	5
57	SWT4	2
58	RC5	4
59	BC6	3
59	DST3	4
60	DST2	3
60	SWT1	2
1062	RC2	3
1062	SWT1	2

Figure 4.7.8 Orderline table data test result

Comment- Items are entered in the orderline table

View Customer Details

CustomerCode	CustomerFName	CustomerLName	CustomerTelephone	CustomerEmail
1	Anushka	Perera	0774513423	anu555@gmail.com
3	Mary	Smith	0712678567	marysmith@yahoo.com
4	Thilini	Jayasinghe	071123456	thilini@gmail.com
5	Isuru	Senevirathna	07724456789	isuru@yahoo.com
6	Isuru	Senevirathna	0772345678	isuruS@gmail.com
7	Hiruni	Tharuka	0775231567	hirunit@gmail.com
8	Diana	Herft	0714938654	diherft55@gmail.com
9	Ashini	Ranasinghe	076567876	ashranasinghe@yahoo.com
10	Rishan	Mudalige	0777456342	rrmudalige@gmail.com
11	Niluka	Pathirana	0112675436	nnp123@gmail.com
12	Kamani	Sirisena	0725678786	silverline22@yahoo.com
13	Renuka	Weerathunga	0714235643	rweerathunga@gmail.com
14	Zikra	Rizwan	0777123786	RizwanShakira@gmail.com
15	Sherin	Fernando	0714876345	sherif111@gmail.com
16	Rosan	Ranaweera	0778423456	roshanranaweera@gmail.com
17	Reya	Shanmugam	0777346781	rrrprincess@gmail.com
18	Sumith	Medagama	011285673	sfm345@yahoo.com
19	Senal	Silva	0712367546	senal.silva@gmail.com
20	Rashmi	Denipitiya	0778674521	ras123denipitiya@gmail.com
21	Menaka	Salgadu	0714888433	mmmsss123@gmail.com
22	Paul	Karunaratne	0112453657	paulk555@yahoo.com

Figure 4.7.9 Customer Details of the registered customers

Test No. 8 DisplayOrderBirthdayCakes

This is similar to Test No.6 - Included in Appendix D

Test No. 7 DisplayOrderEngagement cakes

This is similar to Test No.6 – Included in Appendix D

Test No. 9 DisplayOrderShorteats

This is similar to Test No. 6 - Included in Appendix D

Test No. 10 DisplayOrderDesserts

This is similar to Test No. 6- Included in Appendix D

Test No. 11 View Orders Report

← → ↻ 🏠 ⓘ localhost:51748/CakeOrdersReportViewer.aspx

Back Cake Orders

1 of 1 Find | Next

Order No	Date To Deliver	Item Code	Item Name	Quantity	Item Details
25	6/2/2020 12:00:00 AM	RC2	Butter Lover	4	Butter Cake with butter icing
32	6/23/2020 12:00:00 AM	RC2	Butter Lover	2	Butter Cake with butter icing
35	6/25/2020 12:00:00 AM	RC2	Butter Lover	3	Butter Cake with butter icing
36	8/10/2020 12:00:00 AM	RC5	Chocolate Cake	2	Chocolate Icing
44	10/12/2020 12:00:00 AM	RC2	Butter Lover	2	Butter Cake with butter icing
49	10/15/2020 12:00:00 AM	RC5	Chocolate Cake	4	Chocolate Icing
50	10/16/2020 12:00:00 AM	RC2	Butter Lover	3	Butter Cake with butter icing
53	10/14/2020 12:00:00 AM	RC2	Butter Lover	3	Butter Cake with butter icing
54	10/15/2020 12:00:00 AM	RC2	Butter Lover	4	Butter Cake with butter icing

Figure 4.8 Order Report

Comment – Order details are displayed in the report

← → ↻ 🏠 ⓘ localhost:51748/IncomeReportViewer.aspx

Income Report

1 of 1 Find | Next

47	10/10/2020 12:00:00 AM	10000
48	10/10/2020 12:00:00 AM	12500
49	10/10/2020 12:00:00 AM	2400
50	10/10/2020 12:00:00 AM	3000
51	10/10/2020 12:00:00 AM	6000
52	10/10/2020 12:00:00 AM	6000
53	10/10/2020 12:00:00 AM	3000
54	10/10/2020 12:00:00 AM	4000
55	10/10/2020 12:00:00 AM	4000
56	10/10/2020 12:00:00 AM	3000
57	10/10/2020 12:00:00 AM	4000
58	10/10/2020 12:00:00 AM	2400
59	10/11/2020 12:00:00 AM	10500
60	10/11/2020 12:00:00 AM	9000
1062	11/19/2020 12:00:00 AM	6000
	Income	203450

Figure 4.9 Income Report

5. Conclusion

5.1. Deficiencies in the system

Available validation checks are not sufficient to avoid considerable number of input errors. As it takes more time to apply validation for many fields therefore it is not possible within the time frame that was available to develop the system.

Selecting multiple items from a single page at once and storing in the cart is not possible.

Placing custom orders with own images and customized features has not been implemented even though fields have been allocated in the shopping cart table and OrderLines table for this purpose due to lack of time. It can be developed as future work.

Rating system to rate each item is not available.

5.2. Future Work

Allow customers to change flavours, colours, icing type of regular cakes and allow them to get prices of those items immediately- The new trend in the business is to provide what the customers exactly need through customization. Therefore it is important to provide more customizing features in the system.

Allow customers to upload their own pictures for fancy cakes like birthday cakes and wedding cakes.

Displaying multiple pictures for each item can attract customers.

Providing related items when customers order items.

Sending promotion emails to registered customers- Customer information can be used in more useful manner to increase sales and to make the shop more popular among customers.

The facility for the customers to view items by price range and flavours can be included . Easier and faster searching facilities are important to provide better service to the customers by minimizing search time.

Discounts and offers can be provided. When customers order items a discount can be calculated and displayed for registered customers.

Adding Google reviews and ratings to the products can be included to help the customers to make buying decisions easily.

Online payment facility through credit card, debit card and third party payment systems can be included.

References

- [1]. The Sponge, Accessed on 18th Nov. 2020[Online]:Available
<https://www.sponge.lk/>
- [2]. Pizza Hut Sri Lanka, Accessed on 18th Nov. 2020[Online]:Available
<https://www.pizzahut.lk/>
- [3]. The Cakery, Accessed on 18th Nov 2020[Online]:Available
<https://www.tecclk.com/specials-c9ac>
- [4]. Tungsten, Responsive vs. Mobile-Friendly Websites: What's the Difference?, Oto Hlincik, February 18, 2015, Accessed on 18th Dec. 2019[Online]:Available
<https://www.atomic74.com/click/responsive-vs-mobile-friendly-websites-whats-the-difference>
- [5]. PixelCrayons, Best Web Development Frameworks Comparison, Roger James, May 22, 2019, Accessed on 10th Dec. 2019[Online]:Available
<https://www.pixelcrayons.com/blog/best-web-development-frameworks-comparison/>
- [6]. EDUCBA, ASP.NET vs PHP, Accessed on 18th Nov. 2019[Online]:Available
<https://www.educba.com/asp-net-vs-php/>
- [7]. Guru99, Laravel vs CodeIgniter: Which is Better?, Accessed on 15th Dec. 2019[Online]:Available
<https://www.guru99.com/laravel-vs-codeigniter.html>
- [8]. OnGraph Technologies, ASP.NET vs PHP: Which One To Choose and Why For App Development?, Alka Singh, November 26, 2019, Accessed on 27th Dec. 2020[Online]:Available
<https://www.ongraph.com/asp-net-vs-php-which-one-to-choose/>
- [9]. JavaTPoint, What is CodeIgniter, Accessed on 28th Nov. 2019[Online]:Available
<https://www.javatpoint.com/what-is-codeigniter>

Appendices

A). System Documentation

Add/Display/Edit/Delete Categories code

AddDisplayEditDeleteCategory.aspx

```
<%@ Page Language="C#" AutoEventWireup="true"
CodeBehind="AddEditDeleteCategories.aspx.cs"
Inherits="YummyshomeFinal.AddEditDeleteCategoriesNew" %>

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
<title></title>
</head>
<body>
<form id="AddEditDeleteCategories" runat="server">
<div style="background-color:aquamarine">
<asp:Button ID="BtnLogout" runat="server" Height="30px" OnClick="Button1_Click"
Text="Logout" Width="100px" />
<asp:LinkButton ID="BtnAdminPanel" PostBackUrl="~/AdminPanel.aspx" runat="server"
Font-Size="Large" BackColor="#33CCFF" ForeColor="Black" Height="30px" Width="150px">Admin
Panel</asp:LinkButton>
<h1> Add/Edit/Delete Categories</h1>
<asp:GridView ID="GridView1" style="background-color:#ffd800" runat="server"
AutoGenerateColumns="False" AutoGenerateDeleteButton="True" AutoGenerateEditButton="True"
DataKeyNames="CategoryNo" DataSourceID="ItemCategoryDataSource" AllowPaging="True"
AllowSorting="True" Height="285px" Width="523px">
<Columns>
<asp:BoundField DataField="CategoryNo" HeaderText="Category Number"
ReadOnly="True" SortExpression="CategoryNo" />
<asp:BoundField DataField="CategoryName" HeaderText="Category Name"
SortExpression="CategoryName" />
</Columns>
</asp:GridView>
<asp:SqlDataSource ID="ItemCategoryDataSource" runat="server"
ConnectionString="<%%$ ConnectionStrings:YummyshomeFinalConnectionString %>"
SelectCommand="SELECT * FROM [Category]" UpdateCommand="UPDATE [Category] SET
[CategoryNo]=@CategoryNo, [CategoryName]=@CategoryName WHERE [CategoryNo]=@CategoryNo"
DeleteCommand="DELETE FROM [Category] WHERE
[CategoryNo]=@CategoryNo"></asp:SqlDataSource>
</div>
<div style="background-color:aqua">
<table>
<tr>
<td>
<asp:Label ID="LblAddNewCategory" runat="server" Text="Enter New Category
Details"></asp:Label>
<asp:Label ID="LblCategoryAddedMesgbox" runat="server" Text=" "
ForeColor="Red"></asp:Label>
</td>
</tr>
<tr>
<td>
</td>
</tr>
</table>
</div>
</body>
</html>
```



```

                <asp:Label ID="LblCategoryNo" runat="server" Text="Category
Number"></asp:Label>
            </td>
            <td>
                <asp:Label ID="LblCategoryName" runat="server" Text="Category
Name"></asp:Label>
            </td>
        </tr>

        <tr>
            <td>
                <asp:Button ID="btnAddICategory" runat="server" Text="Add Category"
OnClick="btnAddCategory_Click" ForeColor="#000099" Height="30px" BackColor="#3399FF" />
            </td>
            <td>
                <asp:TextBox ID="TxtCategoryNo" runat="server" Width="75px"
AutoComplete="off"></asp:TextBox>
            </td>
            <td>
                <asp:TextBox ID="TxtCategoryName" runat="server" Width="200px"
AutoComplete="off"></asp:TextBox>
            </td>
        </tr>
    </tr>
    <td>
        <asp:Label ID="LblMessageBox" runat="server" Text=" " ForeColor="Red"
></asp:Label>
    </td>
</tr>
</table>
</div>
</form>
</body>
</html>

```

Code behind page for AddDisplayEditDeleteCategory

AddDisplayEditDeleteCategory.aspx.cs

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Data;
using System.Data.SqlClient;

namespace YummyshomeFinal
{
    public partial class AddEditDeleteCategoriesNew : System.Web.UI.Page
    {
        SqlConnection conn = new SqlConnection(@"Data
Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\Users\sarah\Documents\YummyshomeFinal\Y
ummyshomeFinal.mdf;Integrated Security=True;Connect Timeout=30");

        protected void Page_Load(object sender, EventArgs e)
        {
        }
    }
}

```

```

protected void btnAddCategory_Click(object sender, EventArgs e)
{
    LblMessageBox.Text = "";
    //Check whether the user has entered the category number and name

    if (String.IsNullOrEmpty(TxtCategoryNo.Text))
    {
        LblMessageBox.Text = ("Please enter Category Details");
    }
    if (String.IsNullOrEmpty(TxtCategoryName.Text))
    {
        LblMessageBox.Text = ("Please enter Category Details");
    }
    else
    {
        conn.Open();
        //Check whether the category already exists

        string sqlquery = "select Count(*) from [dbo].[Category] where
[CategoryNo]='" + TxtCategoryNo.Text + "'";
        SqlCommand sqlcom = new SqlCommand(sqlquery, conn);
        int RecordCount = (int)sqlcom.ExecuteScalar();
        if (RecordCount>0)
        {
            LblMessageBox.Text = "Category already Exists";
        }
        else
        {
            SqlCommand cmd = conn.CreateCommand();
            cmd.CommandType = CommandType.Text;
            cmd.CommandText = "insert into Category values('" + TxtCategoryNo.Text +
"', '" + TxtCategoryName.Text + "')";
            cmd.ExecuteNonQuery();
            TxtCategoryNo.Text = "";
            TxtCategoryName.Text = "";
            LblMessageBox.Text = "New category added to the category table";

            //Display updated category list
            string sqlquery2 = "select * from [dbo].[Category]";
            SqlCommand sqlcom2 = new SqlCommand(sqlquery2, conn);
            SqlDataAdapter sda = new SqlDataAdapter(sqlcom2);
            DataTable dt = new DataTable();
            sda.Fill(dt);
            GridView1.DataBind();
            conn.Close();
        }
    }
}

protected void Button1_Click(object sender, EventArgs e)
{
    Session.Abandon();
    Request.Cookies.Clear();
    Response.Redirect("HomePage.aspx");
}
}
}

```

Code for Display & Order Regular Cakes

DisplayOrderRegularCakes.aspx

```
<%@ Page Language="C#" AutoEventWireup="true"
CodeBehind="DisplayOrderRegularCakes.aspx.cs"
Inherits="YummyshomeFinal.DisplayRegularCakes" %>

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title></title>
</head>
<body style="background-color:aquamarine">
    <form id="form1" runat="server">
        <div style="background-color:aquamarine">
            <table style="border: thin solid #0000FF" position:"absolute" left:"500px"
top:"500px" >
                <tr>
                    <td>
                        <asp:LinkButton ID="HomeButton" PostBackUrl="~/HomePage.aspx"
runat="server" Font-Size="Large" BackColor="#3366FF" ForeColor="Black" Height="30px"
Width="150px">Home</asp:LinkButton>
                    </td>
                    <td>
                        <asp:LinkButton ID="LinkButton1"
PostBackUrl="~/ShopMoreItemCategory.aspx" runat="server" Font-Size="Large"
BackColor="#3366FF" ForeColor="Black" Height="30px" Width="150px"> Item
Categories</asp:LinkButton>
                    </td>
                </tr>
            </table>
            <h1>Regular Cakes</h1>
            <asp:GridView ID="GridView1" runat="server" AutoGenerateColumns="False"
OnSelectedIndexChanged="GridView1_SelectedIndexChanged" BackColor="#FFd800"
BorderColor="#FF3300" AllowPaging="True">
                <columns>
                    <asp:BoundField HeaderText="Item Code" DataField="ItemCode" />
                    <asp:BoundField HeaderText="Item Name" DataField="ItemName" />
                    <asp:BoundField HeaderText="Details" DataField="ItemDetails" />
                    <asp:BoundField HeaderText="Price per Kg. LKR" DataField="ItemPrice" />
                    <asp:ImageField HeaderText="Image" DataImageUrlField="ItemImagePath"
ControlStyle-Height="150" ControlStyle-Width="150">
                        <ControlStyle Height="150px" Width="150px"></ControlStyle>
                    </asp:ImageField>
                    <asp:ButtonField ButtonType="Button" CommandName="Select" Text="Select To
Order" />
                </columns>
            </asp:GridView>
            </div>

            <div style="margin-left:5px";"margin-top:700px" ; >
                <br />
                <asp:Label ID="Label1" runat="server" Text="Item Selected for Order"></asp:Label>

                <table style="border: thin solid #0000FF">
                    <tr>
                        <td>
                            <asp:Label ID="LblItemCode" runat="server" Text="Item
Code"></asp:Label>
                        </td>
                    </tr>
                </table>
            </div>
        </div>
    </form>
</body>
</html>
```

```

                <td>
                    <asp:Label ID="LblItemName" runat="server" Text="Item
Name"></asp:Label>
                </td>

                <td>
                    <asp:Label ID="LblPrice" runat="server" Text="Price"></asp:Label>
                </td>
                <td>
                    <asp:Label ID="LblQuantity" runat="server" Text=" Enter required
Quantity in Kg."></asp:Label>
                </td>
                <td>
                    <asp:Label ID="LblTotalPrice" runat="server" Text="Total
Price"></asp:Label>
                </td>
            </tr>
            <tr style="border: thin dashed #0000FF">
                <td>

                    <asp:Label ID="LblItemCodeValue" runat="server" Text=" "></asp:Label>
                </td>
                <td>

                    <asp:Label ID="LblItemNameValue" runat="server" Text=" "></asp:Label>
                </td>
                <td>

                    <asp:Label ID="LblItemPriceValue" runat="server" Text="
"></asp:Label>
                </td>
                <td>
                    <asp:TextBox ID="TxtQuantity" autocomplete="off" runat="server"
BorderColor="Blue"></asp:TextBox>
                </td>
                <td>
                    <asp:Label ID="LblTotalValue" runat="server" Text=" "></asp:Label>
                </td>
                <td>
                    <asp:Button ID="BtnViewPrice" runat="server" Text="View Total Price"
OnClick="BtnViewPrice_Click" Font-Bold="True" BackColor="#3366FF" Height="40px" />
                </td>
                <td>
                    <asp:Button ID="BtnConfirmItem" runat="server" Text="Add Item To
Order" OnClick="BtnConfirmItem_Click" BackColor="#3366FF" Height="40px" />
                </td>
            </tr>
            <tr>
                <td>
                    <asp:Button ID="BtnViewCart" runat="server" Text="View My Cart"
OnClick="BtnViewCart_Click" BackColor="#3366FF" Height="40px" />
                </td>
                <td>
                    <asp:Button ID="BtnShopMore" runat="server" Text="Shop More"
BackColor="#3366FF" Height="40px" OnClick="BtnShopMore_Click" />
                </td>
                <td>
                    <asp:Button ID="BtnCheckOut" runat="server" Text="Check Out"
BackColor="#3366FF" Height="40px" OnClick="BtnCheckOut_Click" />

```

```

        </td>

        <td>
            <asp:Button ID="BtnCancelItem" runat="server" Text="Cancel Item"
BackColor="#3366FF" Height="40px" OnClick="BtnCancelItem_Click" />
        </td>
        <td>
            <asp:Label ID="LblMessageBox" runat="server" Text=" " ForeColor="Red"
Font-Size="Large"></asp:Label>
            <asp:Label ID="LblNewCartNo" runat="server" Text=" "></asp:Label>
        </td>
    </tr>

</table>

</div>
</form>
</body>
</html>

```

Code behind page for Display Order Regular Cakes

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Data;
using System.Data.SqlClient;
using System.IO;

namespace YummyshomeFinal
{
    public partial class DisplayRegularCakes : System.Web.UI.Page
    {
        protected void Page_Load(object sender, EventArgs e)
        {
            // Display Regular cakes in grid view
            SqlConnection conn = new SqlConnection(@"Data
Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\Users\sarah\Documents\YummyshomeFinal\Y
ummyshomeFinal.mdf;Integrated Security=True;Connect Timeout=30");
            conn.Open();
            string sqlquery = "select
ItemCode,ItemName,ItemDetails,ItemPrice,ItemImagePath from [dbo].[Item] where
CategoryNo=1";
            SqlCommand sqlcom = new SqlCommand(sqlquery, conn);
            SqlDataAdapter sda = new SqlDataAdapter(sqlcom);
            DataTable dt = new DataTable();
            sda.Fill(dt);
            GridView1.DataSource = dt;
            GridView1.DataBind();
            conn.Close();
        }

        protected void GridView1_SelectedIndexChanged(object sender, EventArgs e)
        {
            //Display selected item data in text boxes
            SqlConnection conn = new SqlConnection(@"Data
Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\Users\sarah\Documents\YummyshomeFinal\Y
ummyshomeFinal.mdf;Integrated Security=True;Connect Timeout=30");
            conn.Open();
            GridViewRow RecordLine = GridView1.SelectedRow;

```

```

        LblItemCodeValue.Text = RecordLine.Cells[0].Text;
        LblItemNameValue.Text = RecordLine.Cells[1].Text;
        LblItemPriceValue.Text = RecordLine.Cells[3].Text;
        conn.Close();
    }

    protected void BtnViewPrice_Click(object sender, EventArgs e)
    {
        //Calculate and display Total price for a given quantity
        if (String.IsNullOrEmpty(TxtQuantity.Text))
        {
            LblMessageBox.Text = ("Please enter required quantity");
        }
        else
        {
            LblTotalValue.Text = (Convert.ToDouble(LblItemPriceValue.Text) *
            Convert.ToDouble(TxtQuantity.Text)).ToString();

            Session["CakeItemTotalValue"] = (Convert.ToDouble(LblItemPriceValue.Text)
            * Convert.ToDouble(TxtQuantity.Text)).ToString();
        }
    }

    protected void BtnConfirmItem_Click(object sender, EventArgs e)
    {
        SqlConnection conn = new SqlConnection(@"Data
        Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\Users\sarah\Documents\YummyshomeFinal\Y
        ummyshomeFinal.mdf;Integrated Security=True;Connect Timeout=30");
        conn.Open();
        if (Session["CartNo"] == null)
        {
            //Get the new cart number. Maximum value of the cart number field + 1
            SqlCommand cmd2 = new SqlCommand("SELECT MAX(CartNo)+1 as CartNo FROM
            ShoppingCart", conn);
            string NewCartNo = cmd2.ExecuteScalar().ToString();

            if (String.IsNullOrEmpty(TxtQuantity.Text))
            {
                LblMessageBox.Text = ("Please enter required quantity");
                LblMessageBox.Attributes.Add("style", "text-decoration:blink");
            }
            else
            {
                //Calculate price for the entered quantity
                double ItemTotalValue =
                Convert.ToDouble(Convert.ToDouble(LblItemPriceValue.Text) *
                Convert.ToDouble(TxtQuantity.Text));
                //Enter selected item data to the shopping cart table
                SqlCommand cmd = conn.CreateCommand();
                cmd.CommandType = CommandType.Text;
                cmd.CommandText = "insert into ShoppingCart
                (CartNo,ItemCode,ItemName,Quantity,TotalPrice) values('" + NewCartNo + "','" +
                LblItemCodeValue.Text + "','" + LblItemNameValue.Text + "','" + TxtQuantity.Text + "','"
                + ItemTotalValue + "')";
                cmd.ExecuteNonQuery();
                LblMessageBox.Text = ("Item added to the shopping cart");
                //LblNewCartNo.Text = NewCartNo;
                Session["CartNo"] = NewCartNo;
            }
        }
        else
        {

```

```

        int CartNo = Convert.ToInt32(Session["CartNo"]);
        double ItemTotalValue = (Convert.ToDouble(LblItemPriceValue.Text) *
Convert.ToDouble(TxtQuantity.Text));
        //Enter selected item data to the shopping cart table
        SqlCommand cmd = conn.CreateCommand();
        cmd.CommandType = CommandType.Text;
        cmd.CommandText = "insert into ShoppingCart
(CartNo,ItemCode,ItemName,Quantity,TotalPrice) values('" + CartNo + "','" +
LblItemCodeValue.Text + "','" + LblItemNameValue.Text + "','" + TxtQuantity.Text + "','"
+ ItemTotalValue + "')";
        cmd.ExecuteNonQuery();
        LblMessageBox.Text = ("Item added to the shopping cart");
    }

    conn.Close();
}

protected void BtnViewCart_Click(object sender, EventArgs e)
{
    // Navigate to MyCart Page to view items in the cart
    Response.Redirect("MyCart.aspx");
}

protected void BtnCheckOut_Click(object sender, EventArgs e)
{
    //Ridirect to Checkout page
    Response.Redirect("Checkout.aspx");
}

protected void BtnShopMore_Click(object sender, EventArgs e)
{
    //Ridirect to shop more Item category page
    Response.Redirect("ShopMoreItemCategory.aspx");
}

protected void BtnCancelItem_Click(object sender, EventArgs e)
{
    LblItemCodeValue.Text = "";
    LblItemNameValue.Text = "";
    LblItemPriceValue.Text = "";
    TxtQuantity.Text = "";
    LblTotalValue.Text = "";
    LblTotalValue.Text = "";
    LblMessageBox.Text = ("Item removed from the shopping cart");
}
}
}

```

Code for Checkout page

Checkout.aspx

```
<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="Checkout.aspx.cs"
Inherits="YummyshomeFinal.Checkout" %>

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
  <title></title>
</head>
<body style="background-color:aquamarine">
  <form id="form1" runat="server">
    <table>
      <tr>
        <td>
          <asp:LinkButton ID="HomeButton" PostBackUrl="~/HomePage.aspx" runat="server"
Font-Size="Large" BackColor="#3366FF" ForeColor="Black" Height="30px" Width="150px"
BorderColor="#3366FF">Home</asp:LinkButton>
        </td>
        <td>
          <asp:LinkButton ID="LnkBtnShopMore" runat="server" BackColor="#3366FF" Font-
Size="Large" ForeColor="Black" Height="30px" PostBackUrl="~/ShopMoreItemCategory.aspx"
Width="150px">Shop More</asp:LinkButton>
        </td>
        <td>
          <asp:Button ID="BtnViewOrderCheckout" runat="server" Text="Checkout" Font-
Size="Large" OnClick="BtnViewOrderCheckout_Click" BackColor="#0066FF" Height="30px"
Width="150px" />
        </td>
      </tr>
    </table>
    <div style="width:50%; float:left;">
      <h1>Checkout</h1>
      <h2>Items in your cart</h2>

      <asp:Label ID="LblCustomerCart" runat="server" Text=" "></asp:Label>
      <asp:GridView ID="GrdShoppingCart" runat="server"></asp:GridView>

      <h2>Enter Delivery Data</h2>
      <table>
        <tr>
          <td><asp:Label ID="LblDeliveryDate" runat="server" Text="Delivery Date
MM/DD/YYYY"></asp:Label></td>
          <td><asp:TextBox ID="TxtDeliveryDate" autocomplete="off" runat="server"
Height="25px" Width="150px"></asp:TextBox></td>
        </tr>

        <tr>
          <td><asp:Label ID="LblFName" runat="server" Text="Name"></asp:Label></td>
          <td><asp:TextBox ID="TxtFName" runat="server" autocomplete="off"
Height="25px" Width="250px"></asp:TextBox></td>
        </tr>

        <tr>
          <td><asp:Label ID="LblLname" runat="server"
Text="Surname"></asp:Label></td>
          <td><asp:TextBox ID="TxtLname" runat="server" autocomplete="off"
Height="25px" Width="250px"></asp:TextBox></td>
        </tr>

        <tr>
          <td><asp:Label ID="LblTelephone" runat="server"
Text="Telephone"></asp:Label></td>
```



```

        <td><asp:TextBox ID="TxtTelephone" runat="server" autocomplete="off"
Height="25px" Width="150px"></asp:TextBox></td>
    </tr>
    <tr>
        <td><asp:Label ID="Lblemail" runat="server"
Text="email"></asp:Label></td>
        <td><asp:TextBox ID="TxtEmail" runat="server" autocomplete="off"
Height="25px" Width="250px"></asp:TextBox></td>
    </tr>
    <tr>
        <td>
            <asp:Label ID="LblSelectPaymentMethod" runat="server" Text="Payment
Method"></asp:Label>
        </td>
        <td>
            <asp:ListBox ID="LstPaymentOption" runat="server">
                <asp:ListItem>Bank Deposit</asp:ListItem>
                <asp:ListItem>Pay at Order Collection</asp:ListItem>
            </asp:ListBox>
        </td>
    </tr>
</table>
</div>
<div style="width:50%; float:right;">
    <h2>Order Confirmation</h2>
    <table>
        <tr>
            <td>
                <asp:TextBox ID="TxtOrderDetails" runat="server" Height="400px"
TextMode="MultiLine" Width="300px"></asp:TextBox>
            </td>
        </tr>
        <tr>
            <td>
                <asp:Label ID="LblMailDeliveryConfirmation" runat="server" Text="
"></asp:Label>
            </td>
        </tr>
    </table>
</div>
</form>
</body>
</html>

```

Code behind page for Check out page

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Data;
using System.Data.SqlClient;
using System.IO;
using System.Net.Mail;

namespace YummyshomeFinal
{
    public partial class Checkout : System.Web.UI.Page
    {
        protected void Page_Load(object sender, EventArgs e)
        {
            //Get cart number from previous page

            var Value = Convert.ToInt32(Session["CartNo"]);

            // Display Cart Items in grid view

            SqlConnection conn1 = new SqlConnection(@"Data
Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\Users\sarah\Documents\YummyshomeFinal\Y
ummyshomeFinal.mdf;Integrated Security=True;Connect Timeout=30");

            conn1.Open();
            string sqlquery = "select ItemCode,ItemName,Quantity,TotalPrice from
[dbo].[ShoppingCart] where CartNo='"+Value+"'";
            SqlCommand sqlcom = new SqlCommand(sqlquery, conn1);
            SqlDataAdapter sda = new SqlDataAdapter(sqlcom);
            DataTable dt = new DataTable();
            sda.Fill(dt);
            GrdShoppingCart.DataSource = dt;
            GrdShoppingCart.DataBind();
            conn1.Close();
        }

        protected void BtnViewOrderCheckout_Click(object sender, EventArgs e)
        {
            SqlConnection conn2 = new SqlConnection(@"Data
Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\Users\sarah\Documents\YummyshomeFinal\Y
ummyshomeFinal.mdf;Integrated Security=True;Connect Timeout=30");
            conn2.Open();

            //Enter customer details in Customer table

            SqlCommand cmd5 = conn2.CreateCommand();
            cmd5.CommandType = CommandType.Text;
            cmd5.CommandText = "insert into
Customer(CustomerFName, CustomerLName, CustomerTelephone, CustomerEmail) values('"
+TxtFName.Text+ "', '" +TxtLName.Text+ "', '" +TxtTelephone.Text+ "', '" +TxtEmail.Text+ "'
)";

            cmd5.ExecuteNonQuery();

            //Get last customer number plus 1 to write the new record
            SqlCommand cmd4 = new SqlCommand("SELECT MAX(CustomerCode) as CustomerCode
FROM Customer", conn2);
            int ThisCustomerNo = Convert.ToInt32(cmd4.ExecuteScalar().ToString());
        }
    }
}
```

```

//Prepare order details for orders table
//get the system date
DateTime DateToday = DateTime.Now;
DateTime MyDate = Convert.ToDateTime(TxtDeliveryDate.Text);

// Calculate order total from gridview column 2

double OrderTotal = 0;

for (int i = 0; i <= GrdShoppingCart.Rows.Count - 1; i++)
{
    OrderTotal = OrderTotal +
Convert.ToDouble((GrdShoppingCart.Rows[i].Cells[3]).Text);
}

//enter order data in the order table
SqlCommand cmd1 = conn2.CreateCommand();
cmd1.CommandType = CommandType.Text;
cmd1.CommandText = "insert into
Orders(DateOrdered,DateToDeliver,OrderValue,CustomerCode) values('" +DateToday+ "',''+
+MyDate+ "',''+OrderTotal+ "',''+ThisCustomerNo+ '');"
cmd1.ExecuteNonQuery();

// Get order number of the order
SqlCommand cmd2 = new SqlCommand("SELECT MAX(OrderNo) as OrderNo FROM
Orders", conn2);
int ThisOrderNo = Convert.ToInt32(cmd2.ExecuteScalar().ToString());

// Enter all items displayed in shopping cart gridview in orderline table

for (int i =0; i <= GrdShoppingCart.Rows.Count-1; i++)
{
    SqlCommand cmd3 = conn2.CreateCommand();
    cmd3.CommandType = CommandType.Text;
    cmd3.CommandText = "insert into Orderline(OrderNo,ItemCode,Quantity)
values('" + ThisOrderNo + "',''+ GrdShoppingCart.Rows[i].Cells[0].Text + "',''+
GrdShoppingCart.Rows[i].Cells[2].Text + '');"
    cmd3.ExecuteNonQuery();
}

// display order confirmation details
var CartIntValue = Convert.ToInt32(Session["CartNo"]);

//Display order details in text box
string TextLine = "";
foreach (GridViewRow row in GrdShoppingCart.Rows)
{
    TextLine=TextLine+("Item Number :"+row.Cells[0].Text+"\n"
+"Item Name : " + row.Cells[1].Text+"\n"
+"Quantity : " + row.Cells[2].Text)+"\n"
+"Price :Rs."+row.Cells[3].Text+"\n";
}

TextLine="Date :"+DateToday+"\n"+"Order Number: " + ThisOrderNo.ToString()+
"\n"
+"Customer Name : "+TxtFName.Text+" "+TxtLname+"\n"
+TextLine+"\n"+"Order Total Rs.: "+OrderTotal.ToString();

if (LstPaymentOption.Text == "Bank Deposit")
{
    TextLine = TextLine + "Payment Option : " + LstPaymentOption.Text+"\n"
+ "Bank : "+Peoples Bank Kelaniya Rajeewa Kothalawala+"\n"

```

```

        + "AccountNumber :196-2-001-7-0045762 "+"\\n";
    }

    TxtOrderDetails.Text = TextLine;

    //Send order confirmation email
    MailMessage Message = new
MailMessage("sarah.priyanganie@gmail.com",TxtEmail.Text, "Order
Confirmation",TxtOrderDetails.Text);
    Message.IsBodyHtml = true;
    SmtplibClient Client = new SmtplibClient("smtp.gmail.com", 587);
    Client.EnableSsl = true;
    Client.Credentials = new
System.Net.NetworkCredential("sarah.priyanganie@gmail.com", "psTM8181");
    Client.Send(Message);
    LblMailDeliveryConfirmation.Text = "Order confirmation email sent
successfully";
    conn2.Close();
    Session["CartNo"] = null;
    }
}
}
}

```

Code for View Order details Report

```

<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="ViewOrders.aspx.cs"
Inherits="YummyshomeFinal.ViewOrders" %>

<%@ Register assembly="Microsoft.Reporting.WebForms, Version=12.0.0.0,
Culture=neutral, PublicKeyToken=89845dcd8080cc91"
namespace="Microsoft.Reporting.WebForms" tagprefix="rsweb" %>

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title></title>
</head>
<body>
    <form id="form1" runat="server">
        <div>

        </div>
        <asp:ScriptManager ID="ScriptManager1" runat="server">
        </asp:ScriptManager>
        <rsweb:ReportViewer ID="ReportViewer1" runat="server" Font-Names="Verdana" Font-
Size="8pt" WaitMessageFont-Names="Verdana" WaitMessageFont-Size="14pt">
            <LocalReport ReportPath="Report2.rdlc">
                <DataSources>
                    <rsweb:ReportDataSource DataSourceId="ObjectDataSource1"
Name="DataSet1" />
                </DataSources>
            </LocalReport>
        </rsweb:ReportViewer>
        <asp:ObjectDataSource ID="ObjectDataSource1" runat="server"
SelectMethod="GetData"
TypeName="YummyshomeFinal.ViewOrdersDatasetTableAdapters.OrdersTableAdapter"></asp:Object
DataSource>
    </form>
</body>
</html>

```

B).User Documentation

User Documentation for Admin Users

Admin Login for site maintenance and management

1. Click on login button from the home page
2. Enter admin user name and password
3. If correct user name and password are entered admin is navigated to the “Site maintenance & Management” page

Item Category Maintenance

1. Click on “Item Category Maintenance” from “Site maintenance & Management” Page.
2. Click on “Add Category” button to add new category data , “Display Categories” to view existing Categories. “Edit Category” and “Delete Category” Buttons can be used to edit and delete categories respectively

Item Data Management

1. Click on “Item Data Maintenance from “Site maintenance & Management” Page.
2. Enter new item details in the form and click Add Item to add the details to the database table
3. Click on edit Item to Edit and Delete Item to delete an item

Reports

Click on Reports button to view Reports

User Documentation for Customer Users

Customer Registration

1. Click on “Register” button of Yummyshome home page
2. Fill the information requested in the expected form
3. Click on “Register” button to save the information and register.
4. Click on “Cancel” to cancel the information entered

View Items

Click on any category button “Cakes” “Desserts” “Short eats” or “Sweets” to view the items in each category

Cakes category has a sub categories page called Regular cakes, Birthday Cakes, Wedding & engagement Cakes, and Cup cakes.

Order Items

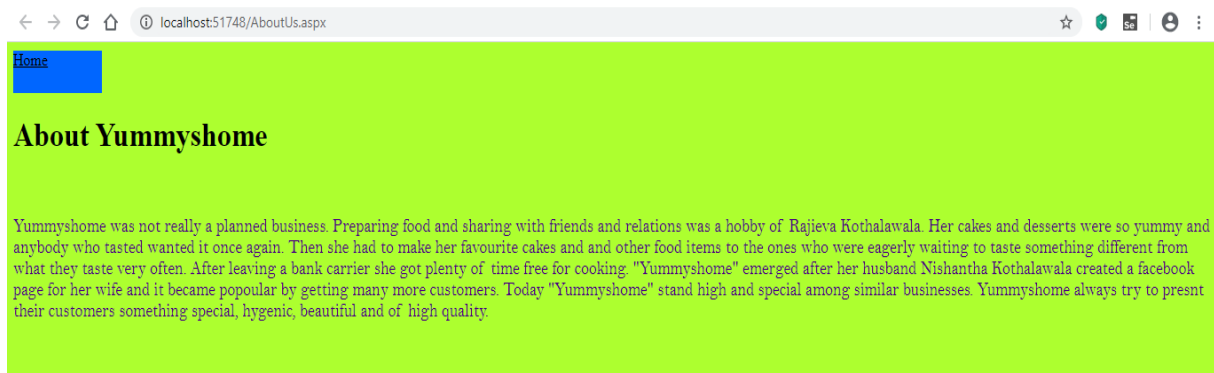
1. Open a wanted category page
2. Choose an item and click on “Select To Order” button of any item to select the item for ordering.
3. Enter required quantity in Kg.

4. Click on “View Total Price” to view the price of the item for the given quantity if required.
5. Click on “Add Item To Order” to add the item to the shopping cart
6. If the customer want to shop more items click on “Shop More” to navigate to the item category page otherwise click on “Checkout” to finish shopping
7. If “Shop More” is clicked the category page is displayed. Click on any other category/Item and add to order
8. Finally when “Checkout” is clicked user can navigate to check out page. Then enter personal details needed for placing the order

C).Additional Test Results

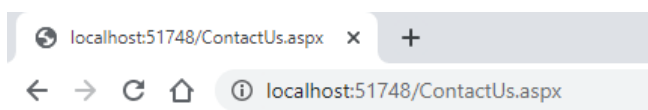
About Us Button click on home page

Shows About Yummyshome



Result of Contact Us button click on home page

Shows Yummyshome contact details



Address : 20, Waragoda road, Kelaniya

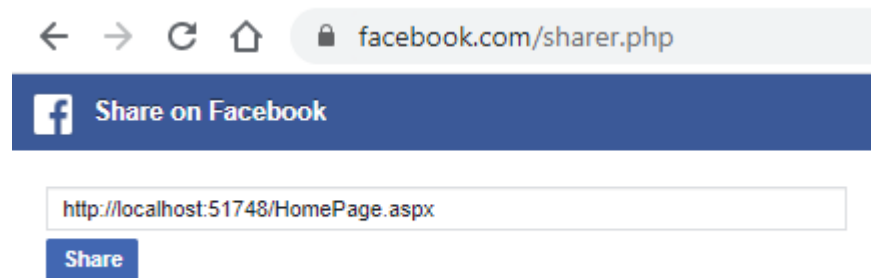
Land Phone : 0112465391

Mobile : 0714111822 : 0778464381

email : rajikothalawala@gmail.com

Result of FB Share button click on homepage

Navigates to Share on Facebook



Birthday Cake item page

The screenshot shows a web browser address bar with the URL `localhost:51748/DisplayOrderBirthdayCakes.aspx`. The page has a light green header with two navigation buttons: "Home" and "Item Categories". Below the header is the section title "Birthday Cakes". The main content is a table with the following columns: Item Code, Item Name, Details, Price per Kg. LKR, Image, and an "Add To Order" button.

Item Code	Item Name	Details	Price per Kg. LKR	Image	
BC1	Round Natural Rose	Chocolate cake with butter icing. Natural flowers	2000		Add To Order
BC12	Tennis Court	Butter Icing with parchment	2500		Add To Order
BC13	Angel Cake	Butter Icing with artificial ornament decor	2500		Add To Order

Desserts Item page

localhost:51748/DisplayOrderDesserts.aspx

[Home](#) [Item Categories](#)

Desserts

Item Code	Item Name	Details	Price per Kg. LKR	Image	
DST2	Strawberry Delight	Strawberry Trifle with jelly	2000		Add To Order
DST3	Caramel Pudding	Traditional Caramel	1500		Add To Order

Item Selected for Order

Item Code	Item Name	Price	Enter required Quantity in Kg.	Total Price		
			<input type="text"/>		View Total Price	Add Item To Order
View My Cart	Shop More	Check Out	Cancel Item			

Error Message “Please enter required quantity” is displayed if View Total Price button is clicked, without entering required amount in kg.

Item Selected for Order

Item Code	Item Name	Price	Enter required Quantity in Kg.	Total Price		
DST2	Strawberry Delight	2000	<input type="text"/>		View Total Price	Add Item To Order
View My Cart	Shop More	Check Out	Cancel Item	Please enter required quantity		