COFFEE HUB

A report Which highlights the requirements as well as how the app was built and tested.

Akhil Montrose - 81788
The University of Trinidad and Tobago – akhil.montrose788@we.utt.edu.tt

Table of Contents

Declaration

I AKHIL MONTROSE student ID: 81788 Hereby Certify that I have not received any assistance in the creation of this document. This is my own original work.

_____ Date: _____

Signature

List of Acronyms

Acronym	Meaning
UI	User Interface
API	Application Programming Interface
UML	Unified Modeling Language
HTML	HyperText Markup Language
CSS	Cascading Style Sheets
SQL	Structured Query Language
PHP	Personal Home Page (previously), PHP: Hypertext Preprocessor (currently)
DBMS	Database Manage System

Introduction

Problem:

During this time of economic stability and a global pandemic, business have been forced to close down due to health concerns. Due to that, business have not been able to run as smoothly and effectively as they usually would. This is the case for all non-essential businesses such as restaurants, bars, cinemas to name a few. Today we will be discussing one particular coffee establishment called Coffee Hub..

Like all business affected by the pandemic, Coffee Hub was not able to provide meals for customers so this project's aim is to create a solution for this problem.

Solution:

To solve this problem, the goal is to design an online web app that will allow customers to browse the business menu online and be able to make orders and have it delivered to them at their location. Customers will order a meal, an employee of Coffee Hub will view the order, have it ready and have a driver deliver the coffee at their location.

Thesis

The topic of this project is create a web application. The idea of this paper is to create a web application for a business which allows them to function the say way they would prior to the pandemic occurring.

The main reasons that support this topic are:

- Create an online system that will allow the business to keep record of all their business transactions.
- Allow customers to be able to make orders for their coffee without physically be present at the establishment (due to the global pandemic).
- Allow the business to function at an otherwise full capacity without having gatherings at the establishment.

One main opposing viewpoint of this idea:

- There may be individuals who may not be able to access the website due to computer literacy, lack of internet access or lack of a device to access the website in general.

Main Points of this Chapter

In this chapter we discussed the problem that the company, Coffee Hub faced as well as the proposed solution to that problem. In the following chapter, we will be discussing the background and the literature review of the project.

Background & Literature Review

Development

This project will be based on the development of a web application. A web application is computer program that uses a web browser to perform a particular function. It is also called a web app. Web apps are present on many websites. A simple example is a contact form on a website. It is a client-server program. The client side is what the user interacts with while the server side is what does the processing the of the information that the user sent. An example of this is when you try to log in to an account. The client will interact with the forms and after they entered their information, it will be sent to the server for it to be processed. This project will be incorporating this same technology.

Previous System

Prior to the pandemic, the company would utilize a file based system. They would record or transactions that would take place with customers. While the company managed by using this system, there are drawbacks.

- Important documents would be misplaced.
- Because it is file based, paper clutter occurs.
- Files are destroyed.
- There is a limit when sharing data.
- Security problems occur when files are misplaced/destroyed.
- Data redundancy. There are multiple copies of the same files.

Existing Systems

In Trinidad and Tobago, we have two popular coffee shows, which are Rituals and Starbucks. They both have an online web application that people can access but one only displays details about the coffee that they have on sale at one of their establishments as well as you cannot make an account or order on the menu (Starbucks).

Rituals on the other hand has a web app which allows users to either create an account or just make an order as a guest. They can also browse the menu as well as have their coffee delivered

to the or ready for pickup. Below we will be doing a comparison of Rituals and what we intend to build for the web application of Coffee Hub.

	Rituals	**Starbucks**	**Coffee Hub**
Web Application	Yes	Yes	Yes
Membership	Yes but not necessary	No	Yes
Browse online menu	Yes	Yes	Yes
Order items online	Yes	Specific locations.	Yes
Customers can pick-up	Yes	Need to make order at a specific location	Yes
Android/IOS app from their respective Appstore	Yes	No	No
Offers Delivery	Yes	Only Drive-thru	Yes

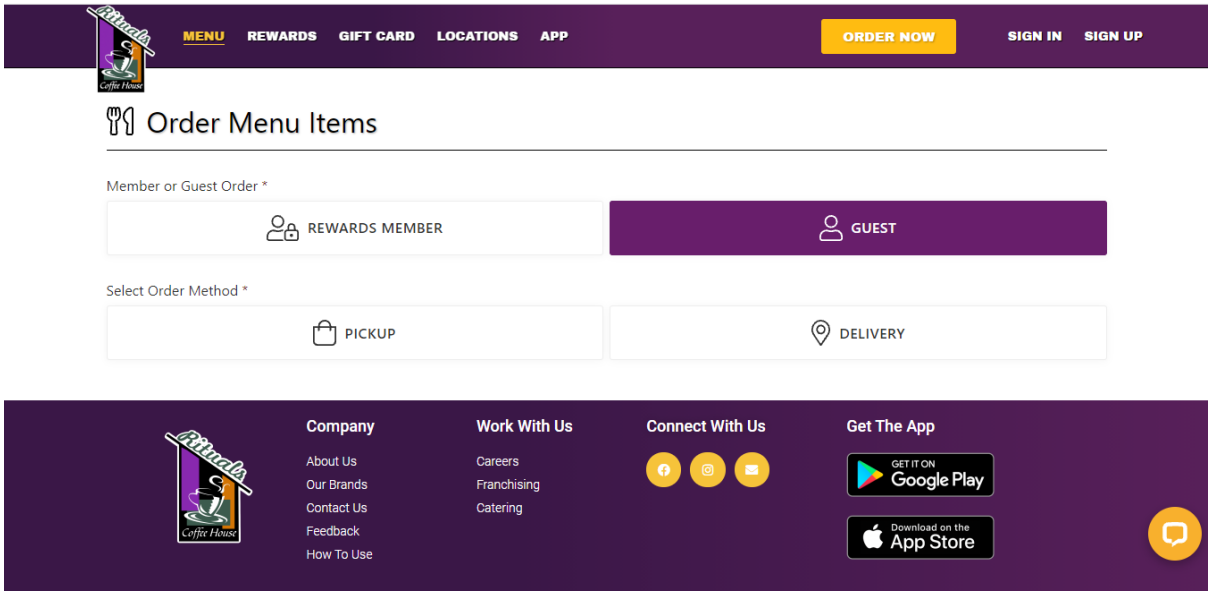


Image showing UI for Rituals Coffee House [Website](#).

The image above shows the order page for rituals coffee. They have an option for members as well as if they want to make an order for pickup/delivery.

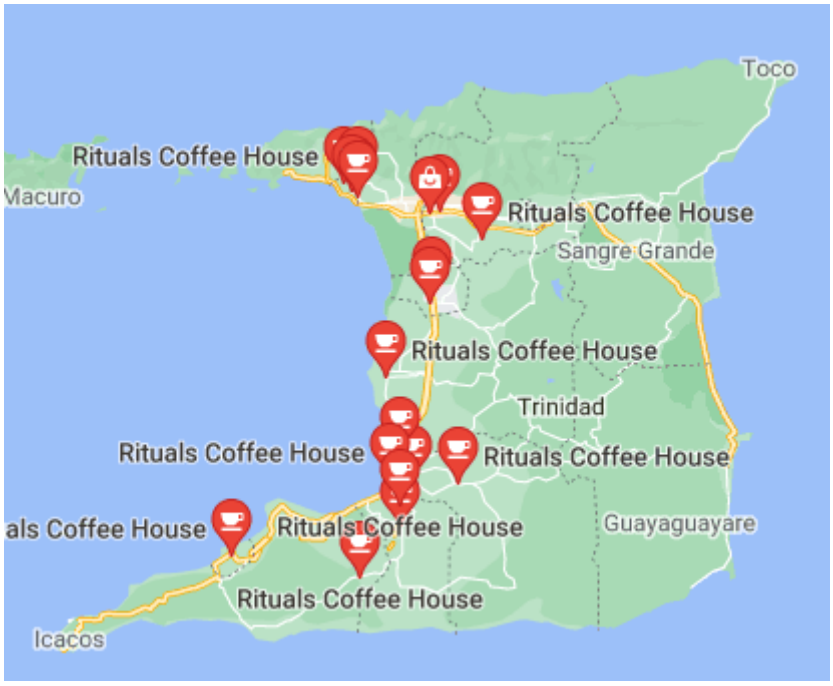


Image 2 showing the locations of all the rituals coffee establishments in Trinidad

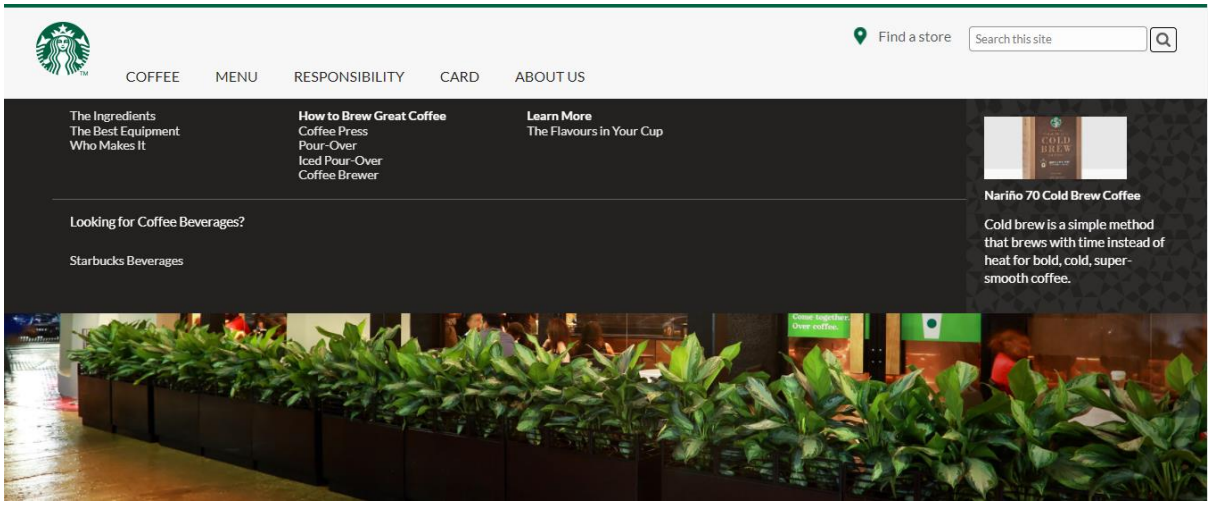


Image showing UI for Starbucks [Website](#).

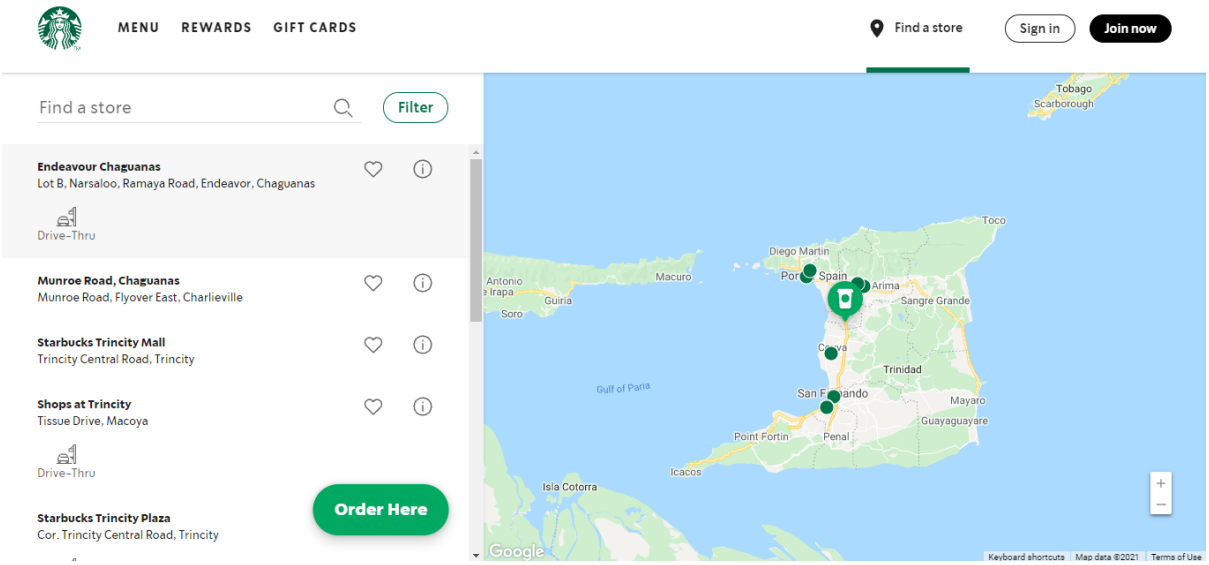


Image showing the [map] of all the Starbucks Location in Trinidad

(This page is outside of the StarbucksTT page)

As you can see, there are only two major coffee establishments in the country where both of them customers can make an order but Starbucks only offer pickups while rituals coffee do both pickup and delivery. This system that we are building for coffee hub will be an online web based app that will give the edge to the two competing businesses.

Why This Choice was Taken

Building a web app will have its benefits which include:

- It will be cost effective overall.
- Easily accessible for the employees.
- Employees will be able to keep track of their daily task.
- It offers a greater reach for customers.
- 24/7 accessibility.
- Improved productivity.
- It improves customer engagement.
- It also strengthens the Brand Image.

Main Points of this Chapter

In this chapter we discussed the problems that the establishment faced, outlining the previous system that they utilized and its shortcomings. We also outlined the other establishments that

sells coffee in Trinidad & Tobago and compared them. Finally, we outlined the benefits of building a web application for coffee hub.

In the next chapter, we will be discussing the methods that were taken to build the web application.

Methodology

Approach to solving the problem

In this chapter, we will be outlining:

- What was conducted at the business to determine the best possible solution for the problem.
- How we are going to transition from using a paper based system to a database.
- The API/UI that will be used (How it will be divided between a view/actual customer/employee)
- How input is taken into the system via the UI.

Solution

A survey was done at Coffee Hub and it was almost unanimous that they would love to have a better way of storing their information as well as being able to access it the data whenever needed. When we asked the customers what their most common daily tasks are, they included:

- Taking orders.
- Keeping a record of all the coffee they have available for a particular day.
- Store information about purchases made.

Requirements Analysis

Similarly we did a survey to customers of the business and they stated that because of the pandemic, they were restricted to go as they please into the establishment. We asked what it is that can be done so that they can still have their coffee and they said:

- Create a system that they can access online.
- Must be able to view all of the coffee that available for them to choose from.
- Make an order for that they want.
- Have it delivered to them.

Because of the feedback we have received, we found out what are the main entities that they company deal with and they are:

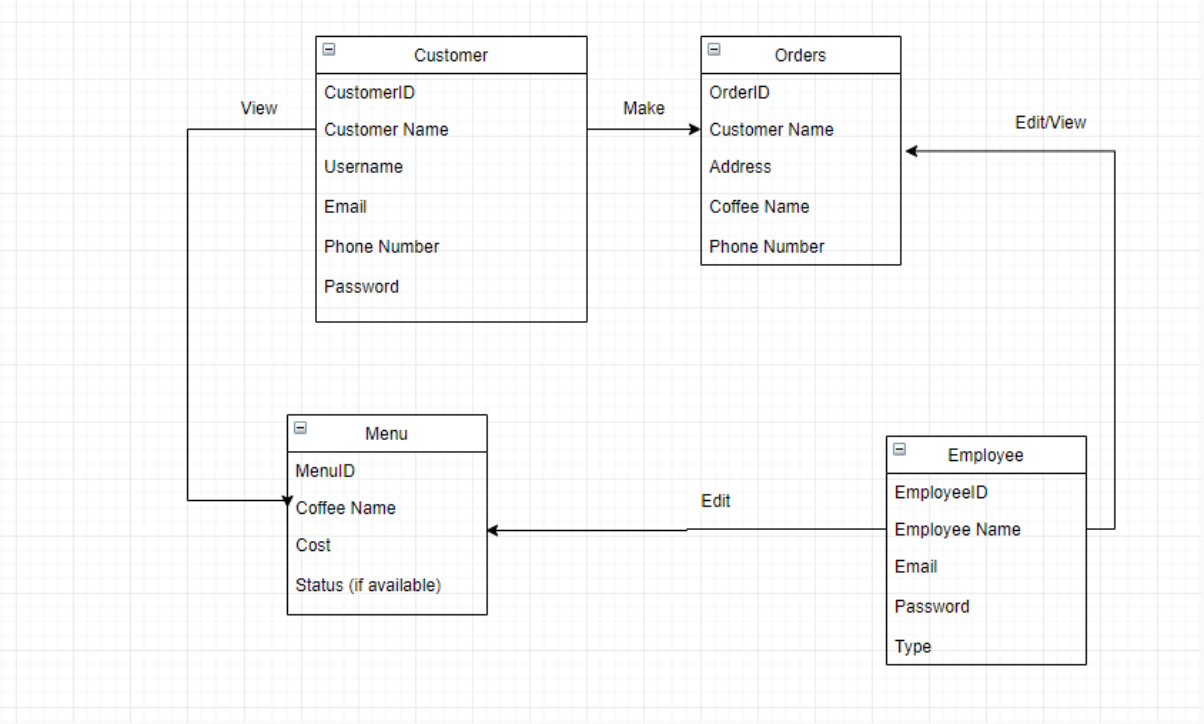
- Customers

- Employees
- Orders
- Menu

In the next section we will be discussing that these four points mean and how a system will be built around it.

Database

We have discovered that they four main items that the customer/employee interact with are the Customer, employee, order, and menu. Both customers and employees wants to be able to view things on the menu as well as make orders. Below is a UML class diagram which illustrates the relationship between these 4 entities.



For this project we will be utilizing a Centralized Database. A Centralized Database Management System (Network DBMS) is a collection of information at a specific location accessible from numerous points. The advantages of using this database architecture includes:

- Ability to access all the information all at one location.
- It is easier to physically secure
- Higher levels of security can be obtained.

- There are fewer breakdowns present within the system.
- It can help businesses to stay close to a focused vision.
- Easier to make backups.

The architecture that we are going to use to store the information is SQL. All the information will be stored on a server (backups included.).

API

The goal of building the API is to investigate what it is the users would like to perform. At coffee hub, the main focus centres around coffee (Viewing/creating orders or new things on the menu/Editing/Deleting etc). These are the common actions that both customers and employees perform on a daily basis.

The Programming language that is going to be used by the customer/employee to view/interact with the system is HTML/CSS.

Hypertext Markup Language (HTML) is a computer language that makes up most web pages and online applications. A hypertext is a text that is used to reference other pieces of text, while a markup language is a series of markings that tells web servers the style and structure of a document.

HTML is not considered a programming language as it cannot create dynamic functionality. Instead, with HTML, web users can create and structure sections, paragraphs, and links using elements, tags, and attributes.

There are a number of things that HTML can do which include:

- Web Development
- Internet Navigation
- Web Documentation

For this project we will be focusing on Web Development.

CSS stands for cascading style sheets. In short, CSS is a design language that makes a website look more appealing than just plain or uninspiring pieces of text. Whereas HTML largely determines textual content, CSS determines visual structure, layout, and aesthetics. HTML is

a markup language, and CSS is a style sheet language. Think "look and feel" when you think CSS.

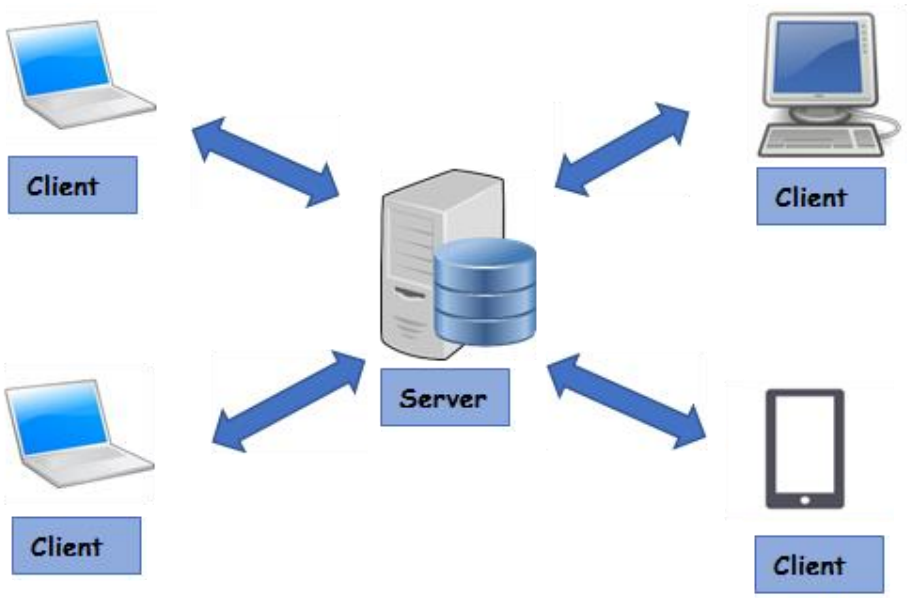
The aim of creating a web app using HTML/CSS is to simulate what the customer/employee would actually do in the real world. From the user's perspective the transition should be seamless.

Input of data via forms

For the project, the web app will be utilizing form to input data. Information that will be taken into through the forms and the database include:

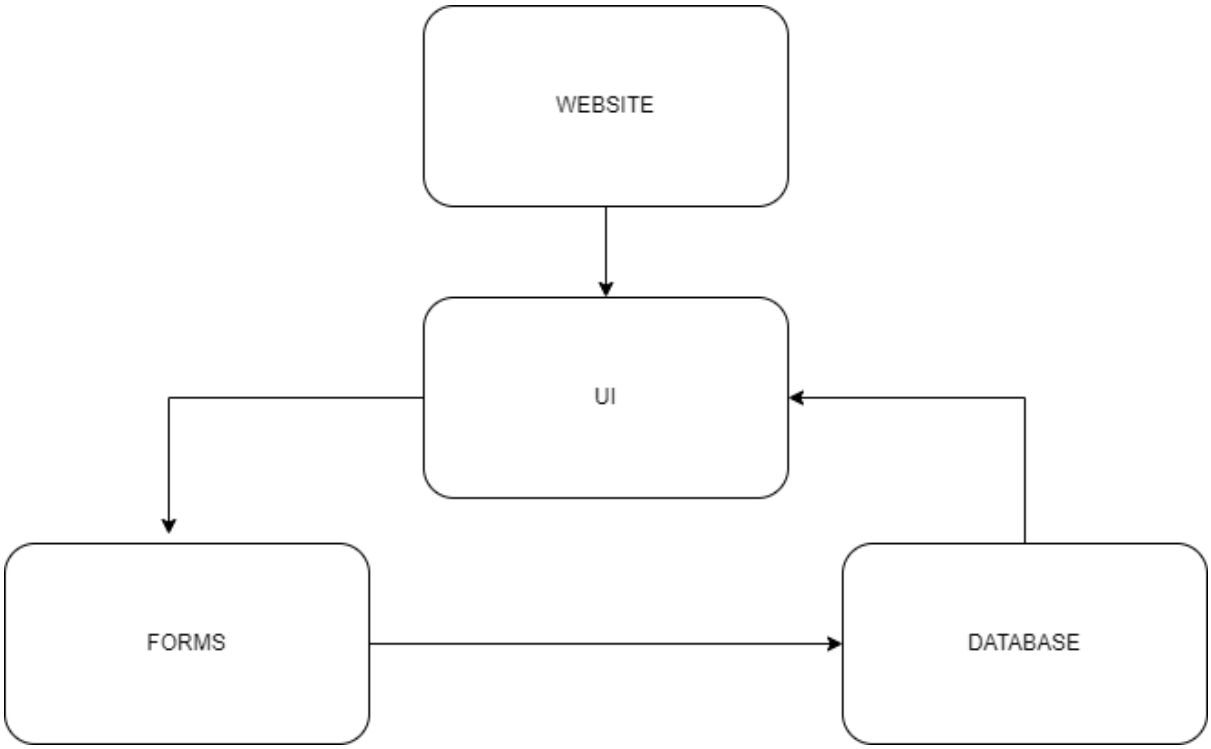
- Registration for new customer/employees.
- Login information for customer/employee.
- Creating forms for items on the menu.
- Editing information on menu and employee forms
- Creating forms for orders.

The way that the forms will be able to capture the data and sending it to the database is through the use of the programming language PHP. The backend will be utilizing PHP to ensure that information is not only inputted but retrieved and edited.



Information is inputted/requested from the client and then that request is sent to be server either to be stored in a database or to be retrieved and displayed back to the client side.

Sub-Components

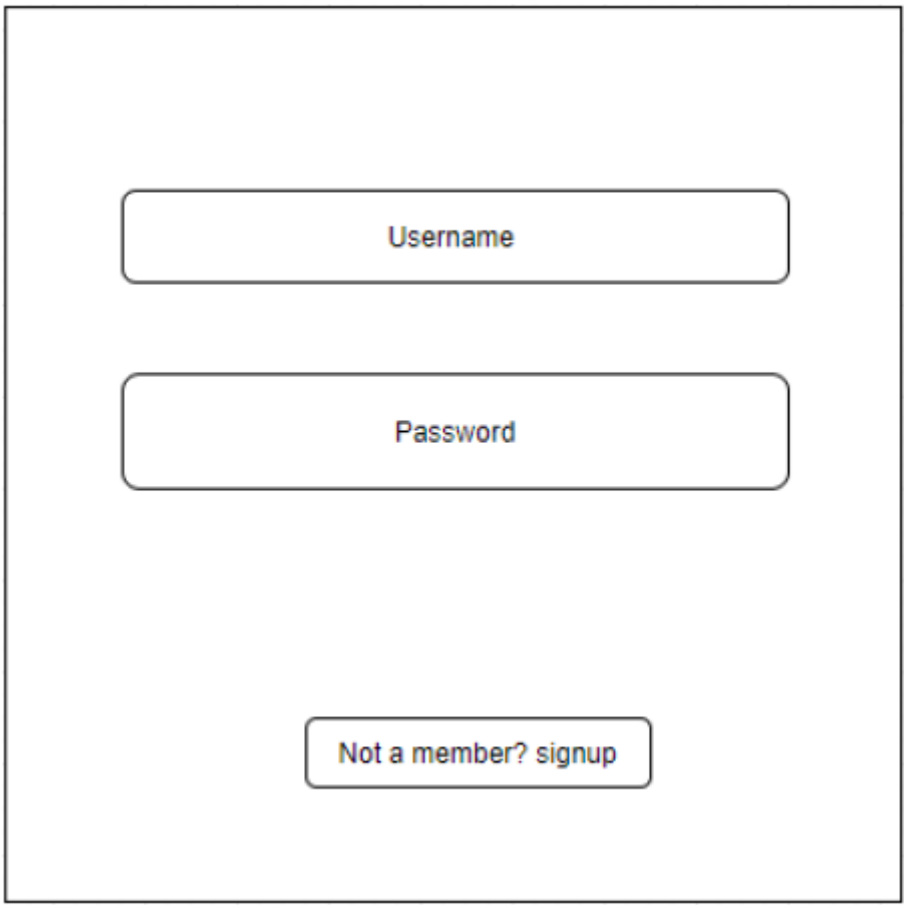


Forms

The forms in this website will be used to input data for:

- User information when signing up (customer/employee).
- When logging in.
- When the admin is creating new objects to be displayed on the website.
- Validation will be done using html validation as well as JavaScript.
- PHP will be used to process the data entered in the forms to the database.

Example of how a form would look like:



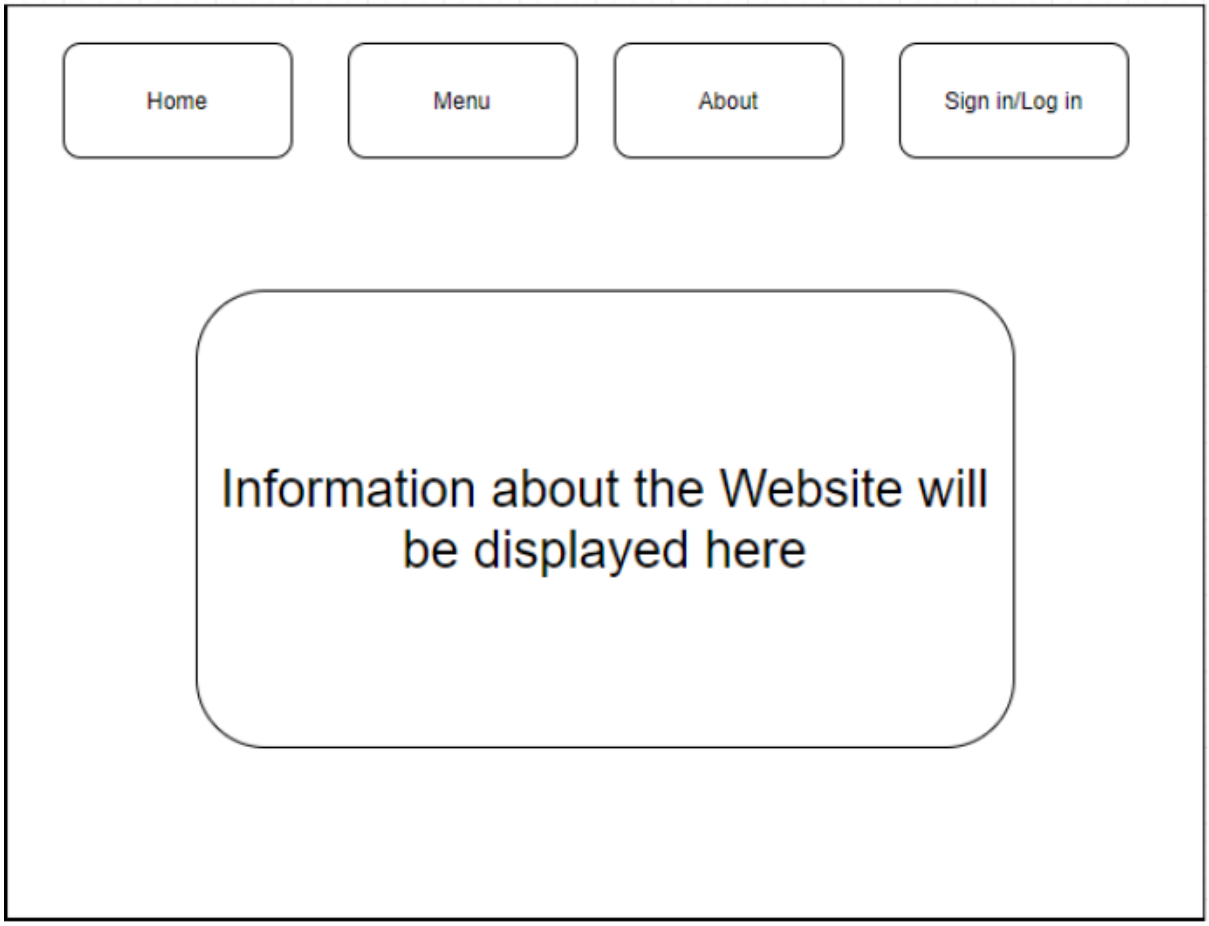
User Interface (UI)

The user interface will be what they users of the website will be viewing/navigating through.

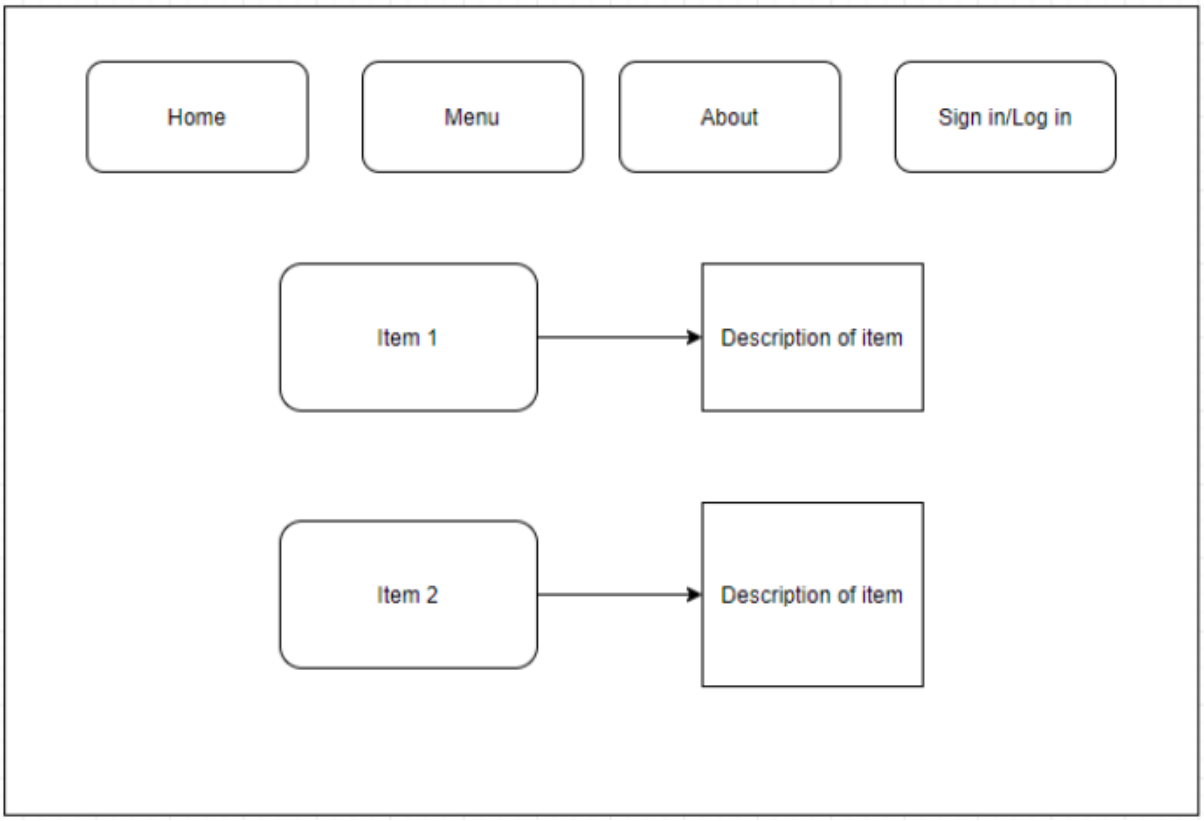
The UI will function as a link for the other sub-components.

- The UI will link the pages so that will lead to the forms e.g.: sign-up and login.
- Database information will be fetched and displayed using the UI.

Example of how the Home UI will look:



Menu UI:


Home
Menu
About
Sign in/Log in
Item 1
Description of item
Item 2
Description of item

Database

The database that will be utilized for this application is SQL.

- This database will be storing information on:
    - User information (customer/employee)
    - Items served on the menu.
    - Orders placed by customers.

Software Method

The software method that this project will be utilizing is the Agile Model.

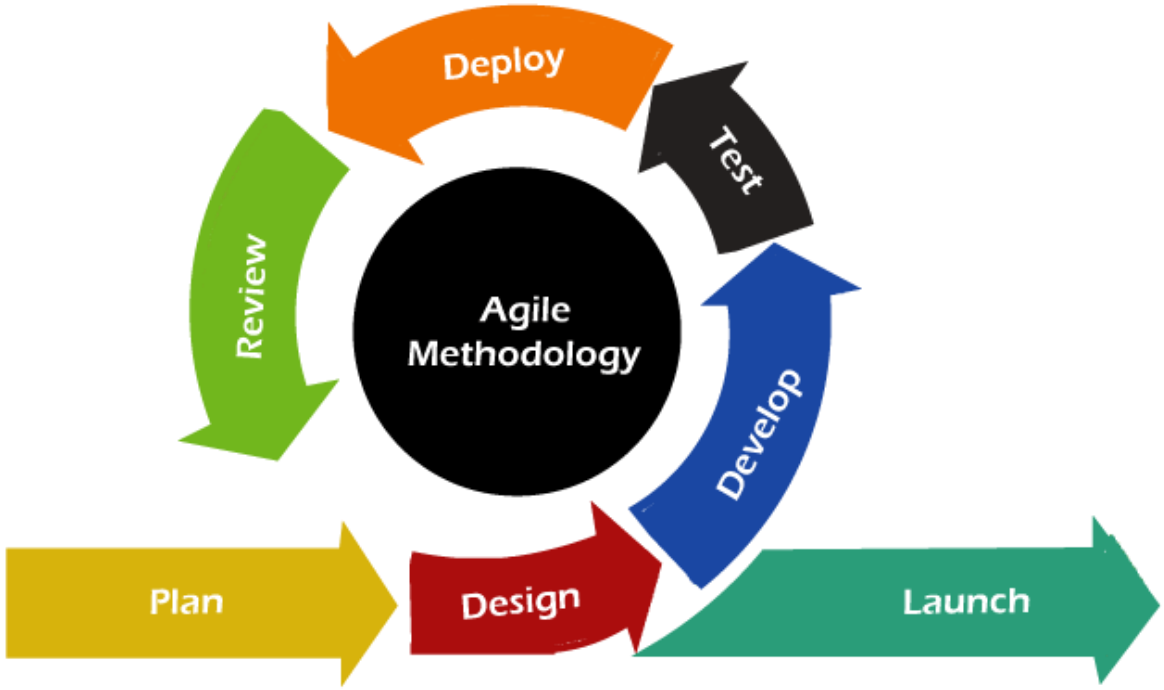


Image showing the Agile Model and the required steps

The Agile model in software development refers to a software development approach based on a combination of iterative and incremental processes with focus on adaptability and customer satisfaction. This model has seven iterative phases:

1. Planning
2. Design
3. Develop
4. Test
5. Deploy
6. Review
7. Launch (after each iteration).

Benefits of using this method include:

- Better management control over the entire project.
- All of the iterations of this model is very transparent.
- Predictable results in terms of time, cost, and deliveries.

Main Points of this Chapter

In this chapter we discussed the approach we used to solve the problem as well as the project major components (database, API, forms etc) and how it will be built as well as how we acquired information from customer/employees of the company. Finally we discussed the software method used and the benefits of using that method (Agile methodology).

In the next chapter we will be discussing the status report of this project.

Implementation, Testing and Analysis

Implementation and Component Testing

In this section, we will be discussing in detail how we are going about the implementation and testing of the major components of the system which include:

- Forms
- UI (user interface)
- Website
- Database

Forms

The forms in this website was designed and implemented using a source code editor call Visual Studio Code (VSCode). This editor using multiple programming languages which include:

1. JavaScript
2. HTML
3. Java
4. CSS
5. TypeScript

For the building of this project, we will be using HTML, CSS, PHP, and JavaScript (where needed).

HTML and CSS are used to display and design the website to all the user(s) to display and navigate with ease. These two languages will encompass the front end of the project. It will also be used as a way to filter that information can be inputted into the forms.

PHP will be used to link the forms from the user side (screen) to the database. This will work as an authentication to check that the information that is being sent from the user end is in agreement with what is in the database. PHP will function as the backend of the application.

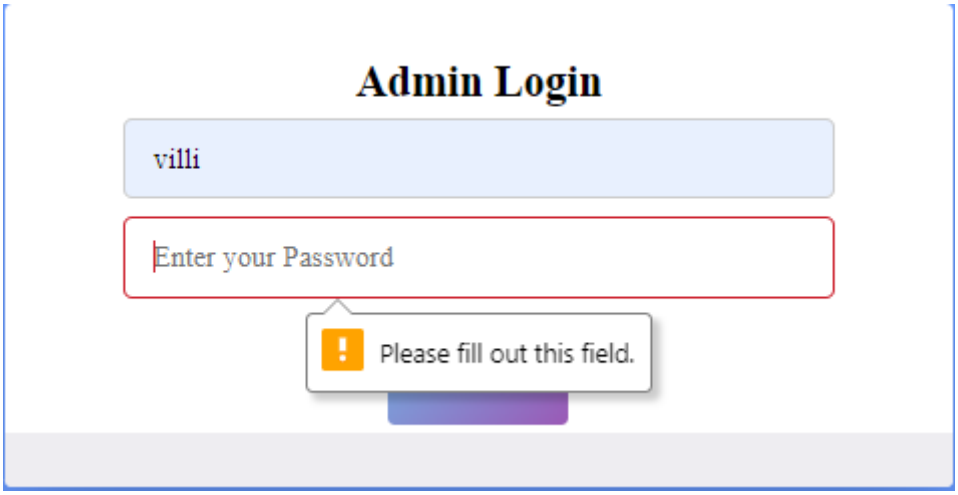
The forms were created the way that they are to enable visibility foe the user using it (Admin, Customer, Employee and Guest).

For the forms, we will be running a few tests to see if it is functioning as it should. Below is a table that shows:

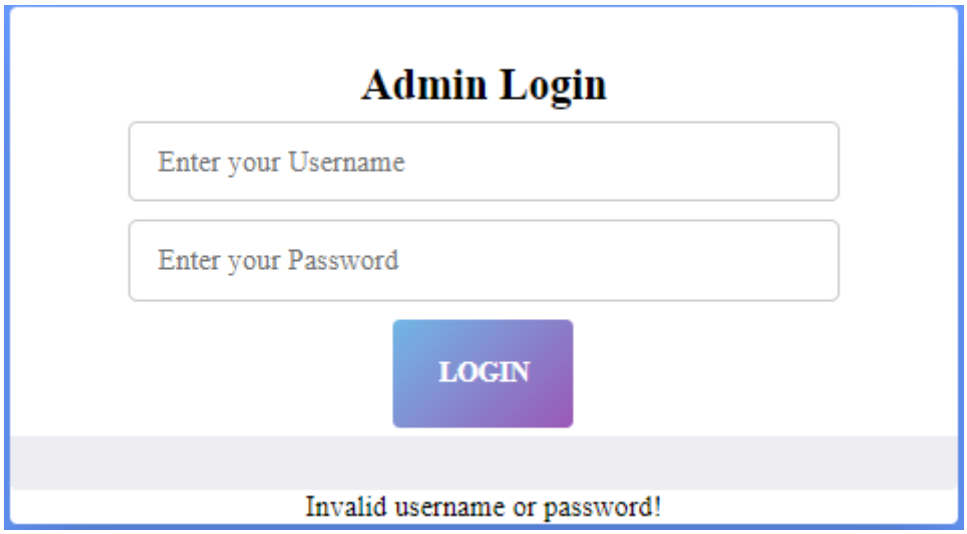
- Description of the tests being performed.
- Element being tested
- Expected results
- Actual results
- What the actual results mean

There are five main forms for this website: 3 Login Forms and two registration forms. In these tests because the forms are built with similar functionalities, we will group them into two tests which will be for registration and login.

Login Forms (Customer, Admin and Employee)



Error message when you do not input all data.



Error Message when you input wrong credentials

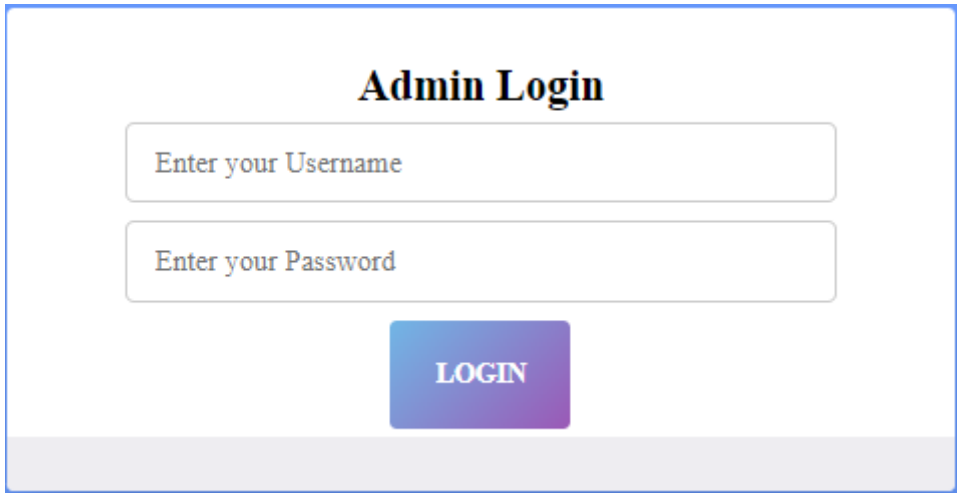


Image showing design of form

*** Please note that the admin login was used alone because the other login forms function the same. Please check code in appendix to see the functionality.*

Test and Description	Element tested	Expected Results	Actual Results	Explanation of actual results
Blackbox Testing: Testing the application with no prior knowledge of how the system works.	Inputs into the forms (username and passwords for the three forms).	All input are validated and inputted if correct credentials in inputted. Error message will show if wrong input is inputted.	No error was observed. Information was validated and inputted.	The actual results were the same as the expected results. The element is functioning as it should.
Unit Testing: This test will be checking the individual parts of the form to see that if performs as it should.	Inputs (username and passwords).	All information will be validated and inputted into database. If An error had occurred, the system would give an error message.	All individual parts are functioning as it should. All the information is being inputted into the system. Error messages are working as it should.	The actual results were the same as the expected results. The element is functioning as it should
Load Time Testing: This type of test will determine how long it will take for input to be validated and inputted into the database from the form	The entire process of loading up forms and inputting the data into all the fields on the form.	The time taken should not take more than a second to show form and less than a second to input data into the fields.	The actual actions did not take more than a second.	The actual results were the same as the expected results. The element is functioning as it should.

Whitebox testing: this testing will check the internal working of the forms to see if it is functioning as it should.	This will be checking all input field (usernames and passwords on all login forms) to see if something other than the required input in inputted in the form will be transferred to database.	Error message will show if incorrect input is submitted but if correct input is submitted no error message will occur	Error message only occurred if incorrect credentials is submitted.	Error messages are working as it should.

Registration Forms (Employee and Customer)

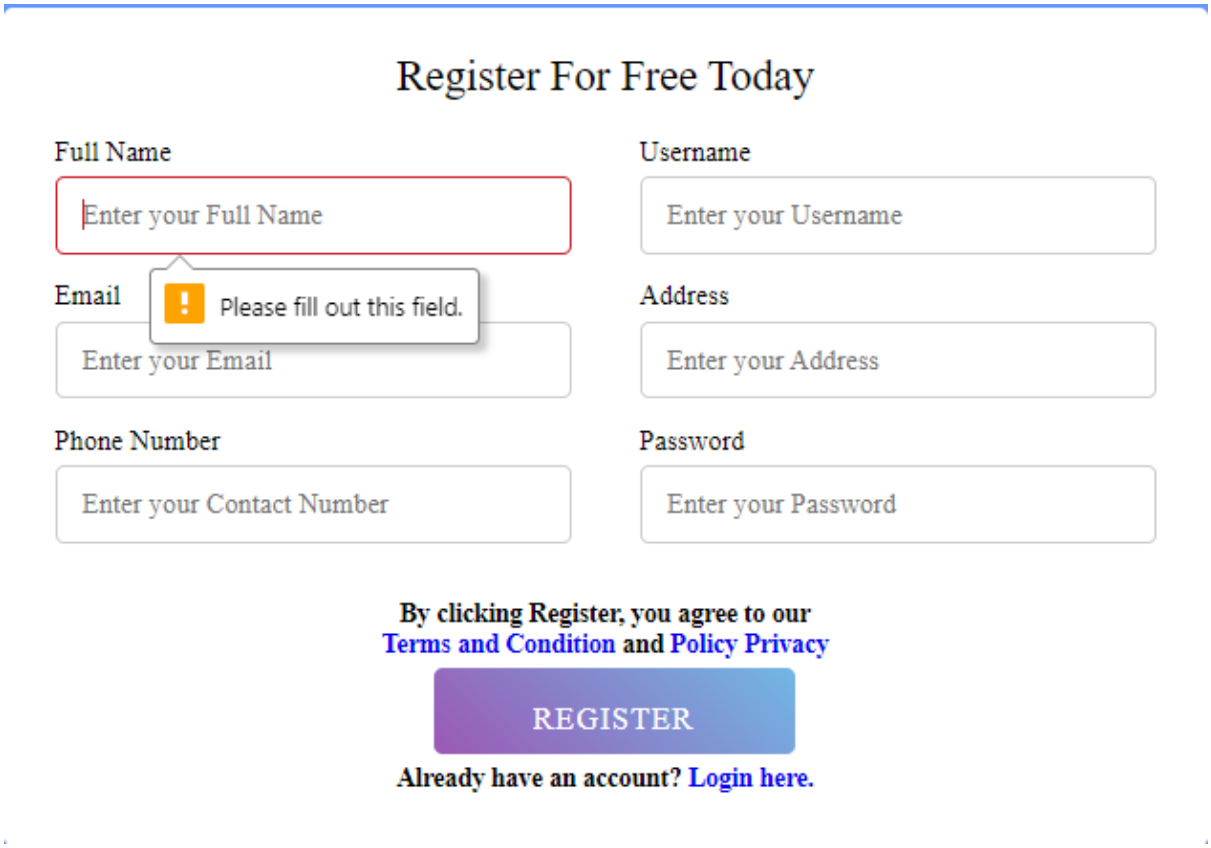


Image showing pop-up in input field when a field is missing

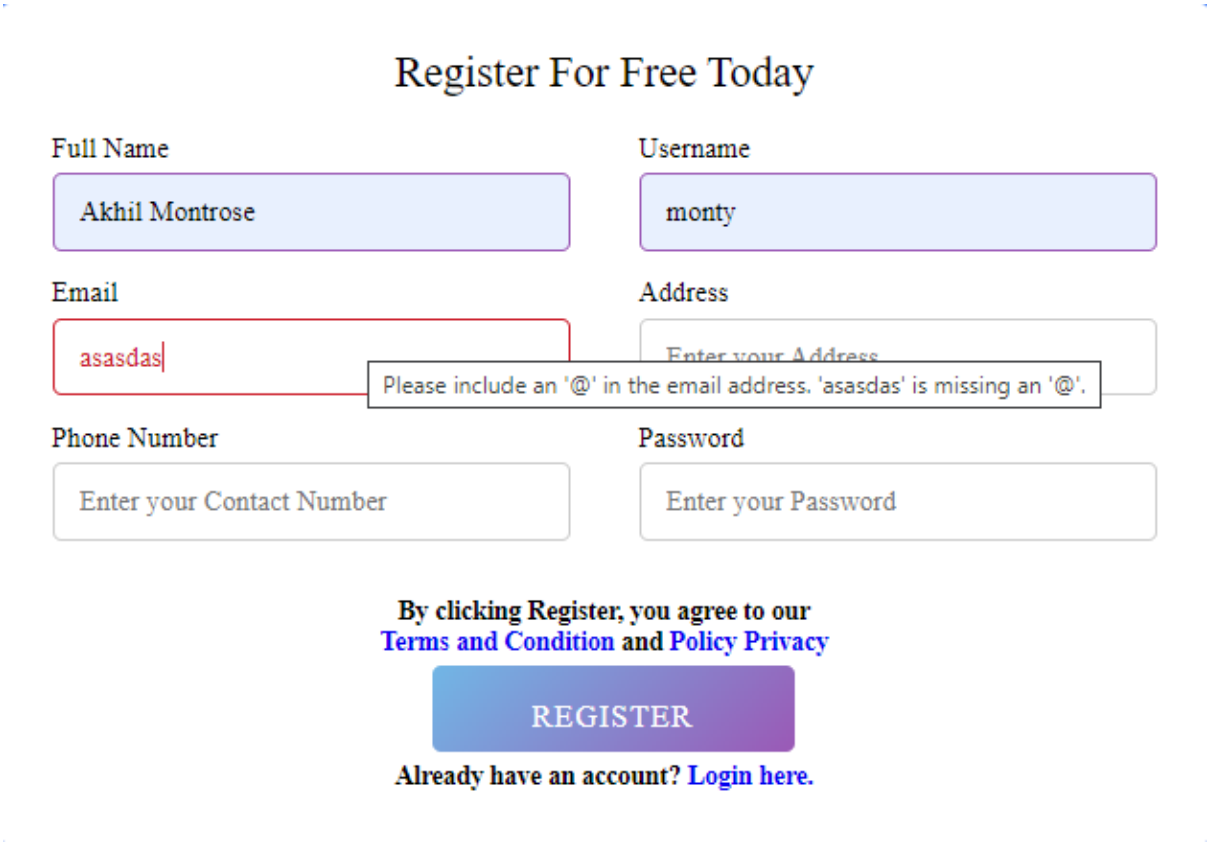


Image showing error message when you do not put in the correct data in an input.

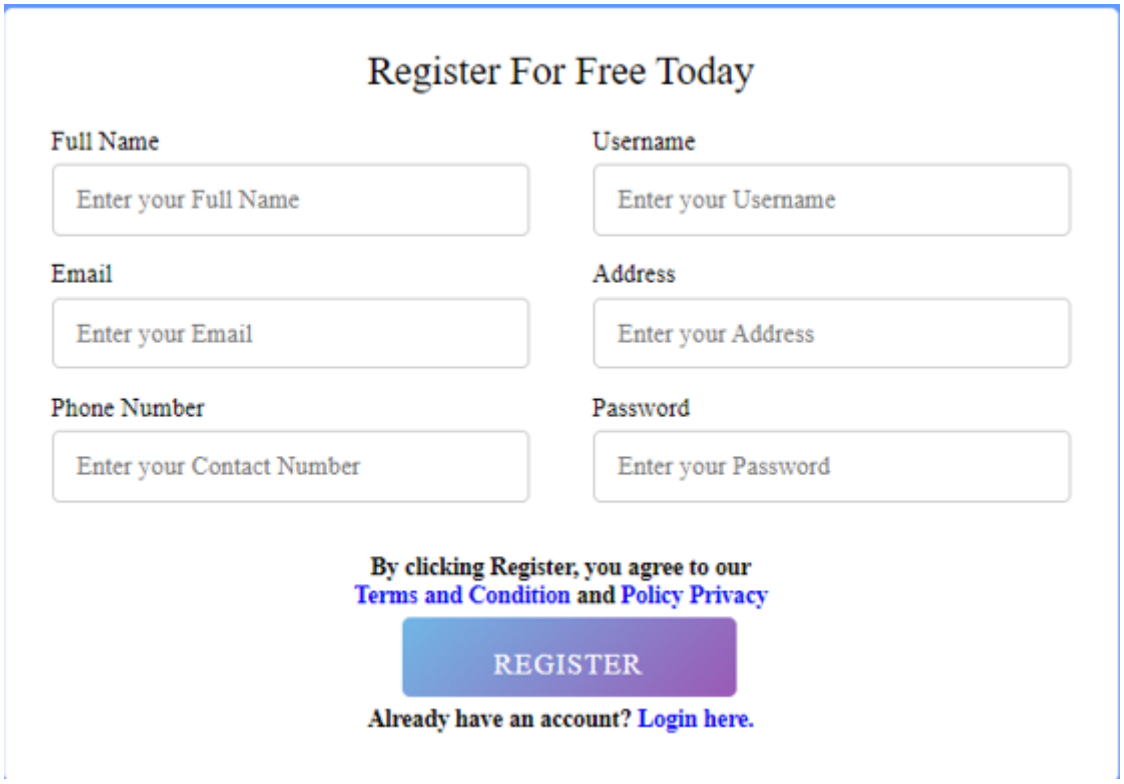


Image showing design of registration form.

***** The tests were performed on only one registration form and one login form. The same functionality applies to the all the other forms. Please check code in appendix.***

Test and Description	Element tested	Expected Results	Actual Results	Explanation of actual results
Blackbox Testing: Testing the application with no prior knowledge of how the system works.	Inputs into the forms (all credentials for registration of customers and employees)	All input are validated and inputted if correct credentials in inputted. Error message will show if wrong input is inputted.	No error was observed. Information was validated and inputted.	The actual results were the same as the expected results. The element is functioning as it should.
Unit Testing: This test will be checking the individual parts of the form to see that if performs as it should.	Inputs (all credentials for registration of customers and employees)	All information will be validated and inputted into database. If An error had occurred, the system would give an error message.	All individual parts are functioning as it should. All the information is being inputted into the system. Error messages are working as it should.	The actual results were the same as the expected results. The element is functioning as it should
Load Time Testing: This type of test will determine how long it will take for input to be validated and inputted into the database from the form	The entire process of loading up forms and inputting the data into all the fields on the form.	The time taken should not take more than a second to show form and less than a second to input data into the fields.	The actual actions did not take more than a second.	The actual results were the same as the expected results. The element is functioning as it should.

Whitebox testing: this testing will check the internal working of the forms to see if it is functioning as it should.	This will be checking all input field (all credentials for registration of customers and employees)to see if something other than the required input in inputted in the form will be transferred to database.	Error message will show if incorrect input is submitted but if correct input is submitted no error message will occur	Error message only occurred if incorrect credentials is submitted.	Error messages are working as it should.

User Interface (Website)

Like the forms section, we will be using HTML, CSS, PHP, and JavaScript (where needed).

HTML and CSS are used to display and design the website to all the user(s) to display and navigate with ease. These two languages will encompass the front end of the project. It will also be used as a way display information on the screen.

For the UI, we will be running a few tests to see if it is functioning as it should. Below is a table that shows:

- Description of the tests being performed.
- Element being tested
- Expected results
- Actual results
- What the actual results mean

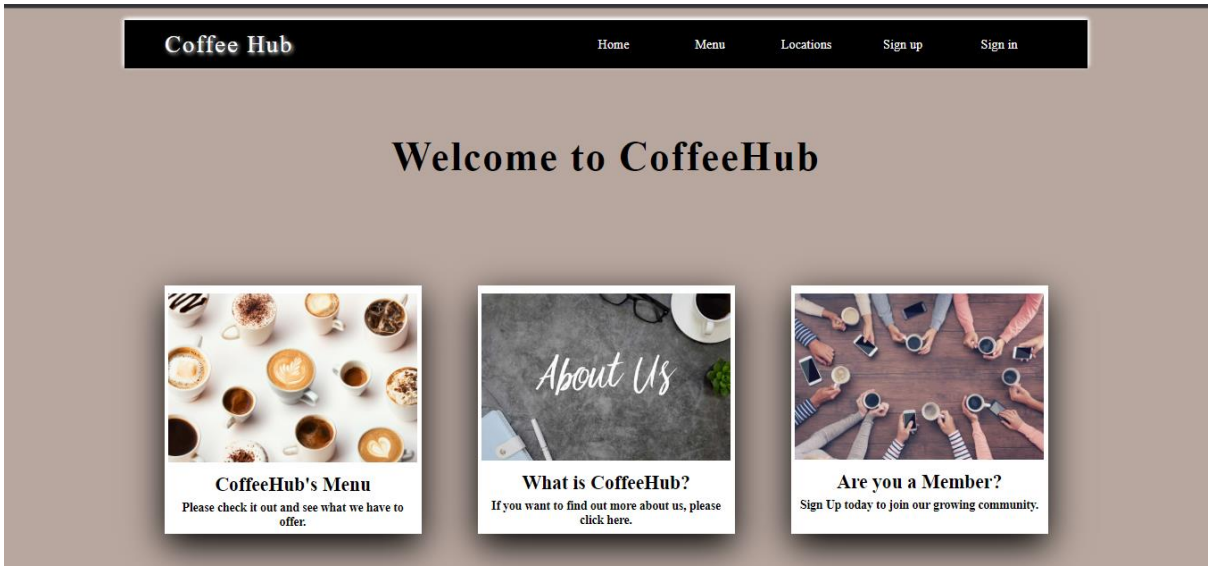


Image showing the design of the website

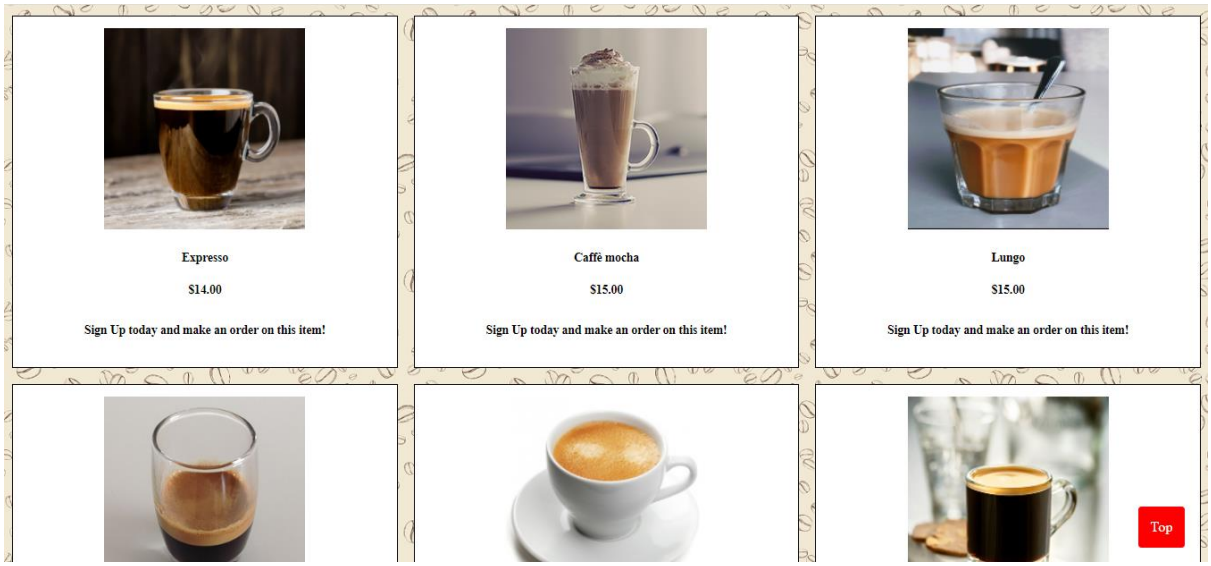


Image showing design of menu page on website

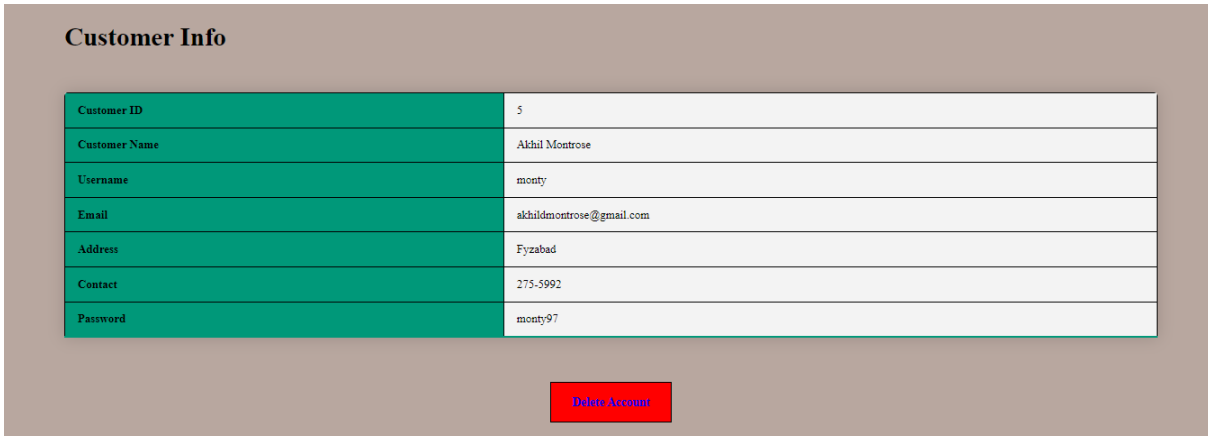


Image showing customer information for user on website

Test and Description	**Element tested**	**Expected Results**	**Actual Results**	**Explanation of actual results**
UI Testing: This will be testing the visibility/readability of the entire system.	All the pages on the website	No visual issues	No visual issues	The actual results were the same as the expected results.

Load Time testing: This will be testing the load time of all the pages that display information.	Pages and their load times	Load time to display information is very fast	Pages with information is loading up as expected.	The actual results were the same as the expected results.
Black box testing: Testing the application with no prior knowledge of how the system works.	We will be testing the general interface of all the pages.	No visual issues.	No visual issues.	The actual results were the same as the expected results.
Whitebox testing: this testing will check the internal working of the forms to see if it is functioning as it should.	Testing if the UI is displaying as it should.	No visual issues	No visual issues present	The actual results were the same as the expected results.

Database

The database of this web app will be using the SQL language and it is being hosted on a Server called XAMPP.

SQL is a Language that is used to managed relational databases. XAMPP is a web server that is used to run this web application. It is also used to create and manage SQL databases.

For the Database, we will be running a database test to see if it is functioning as it should. Below is a table that shows:

- Description of the tests being performed.
- Element being tested
- Expected results
- Actual results
- What the actual results mean

Image showing Xampp Dashboard

Test and Description	**Element tested**	**Expected Results**	**Actual Results**	**Explanation of actual results**
Database Testing	Testing the database of the system (schema, tables etc) using queries (inserts, select, update, delete, drop, etc). This will determine if a user can perform all the functions a common DBMS can do.	All tables and everything associated with database is functioning as it should.	The tables, schema, etc are all functioning as it should.	The expected results are the same as the actual results

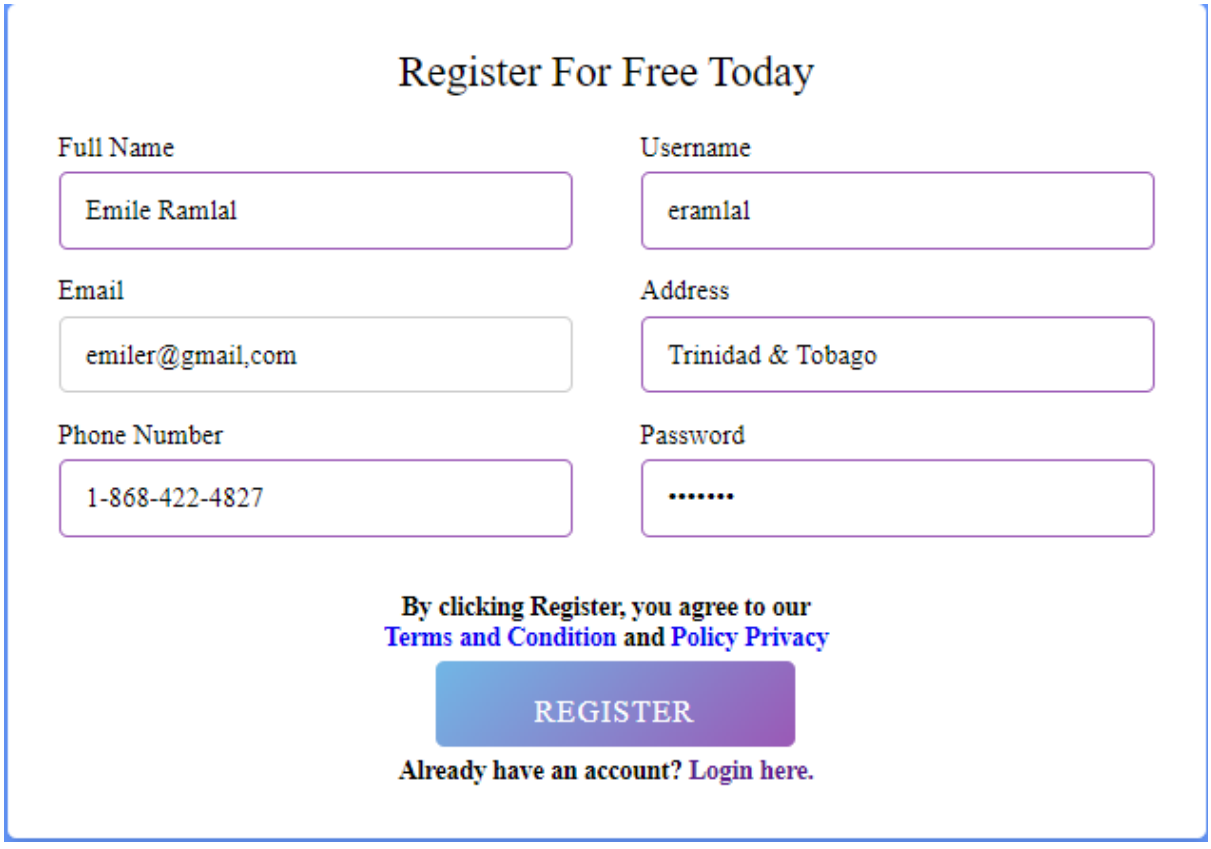
System Testing

In this section, multiple components will be tested simultaneously. The components that will be tested together are:

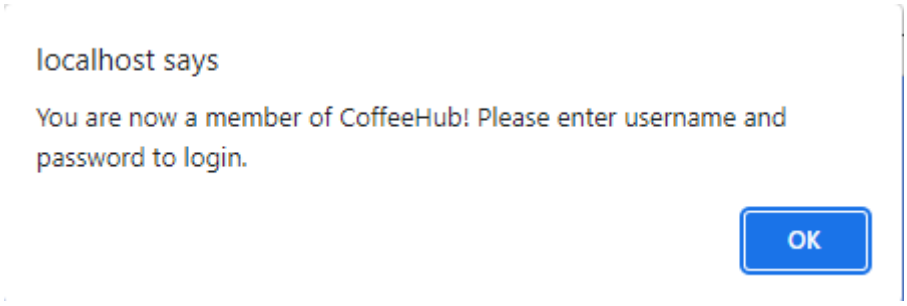
- Forms & Database
- User Interface (Website) & Database
- Buttons & Database
- Testing the entire system

Forms & Database

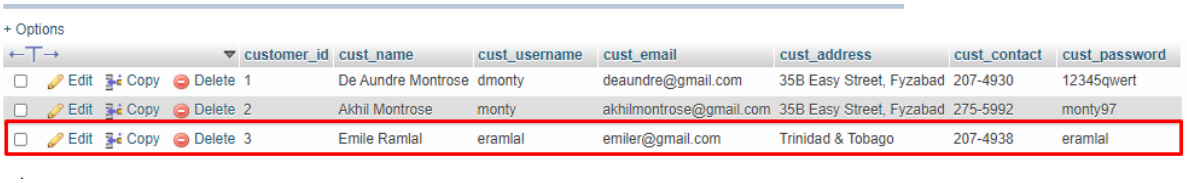
Like the previous test with the forms above, we will be using tests on two type of forms, login, and registration. With that, we will be checking to see if the forms are linking to the database and if the database responds to the website.



Registration form showing input fields for new customer



Message showing that input was accepted into database and user is registered in the system



+ Options

	customer_id	cust_name	cust_username	cust_email	cust_address	cust_contact	cust_password
Edit Copy Delete	1	De Aundre Montrose	dmonty	deaundre@gmail.com	35B Easy Street, Fyzabad	207-4930	12345qwert
Edit Copy Delete	2	Akhil Montrose	monty	akhilmontrose@gmail.com	35B Easy Street, Fyzabad	275-5992	monty97
Edit Copy Delete	3	Emile Ramlal	eramlal	emiler@gmail.com	Trinidad & Tobago	207-4938	eramlal

Image showing record from form in database

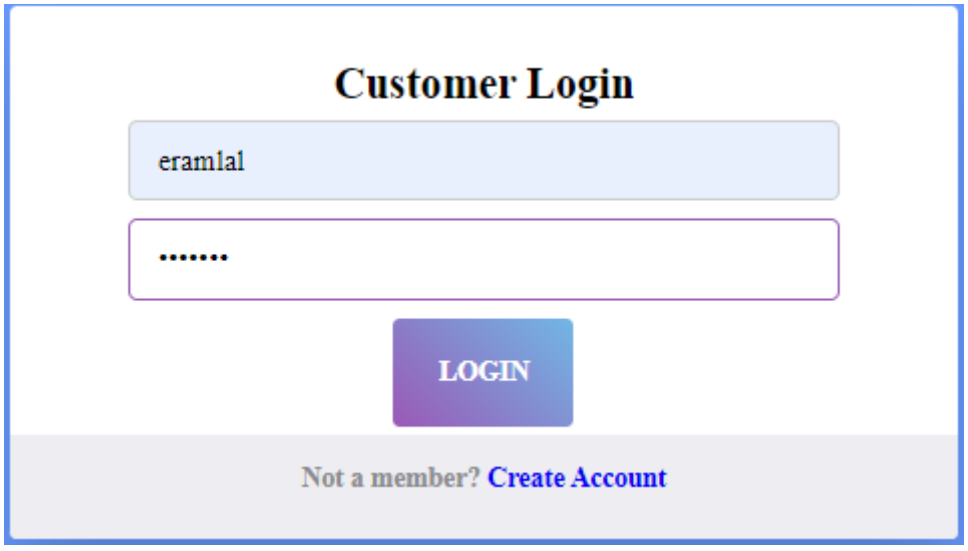


Image showing credentials for customer login.

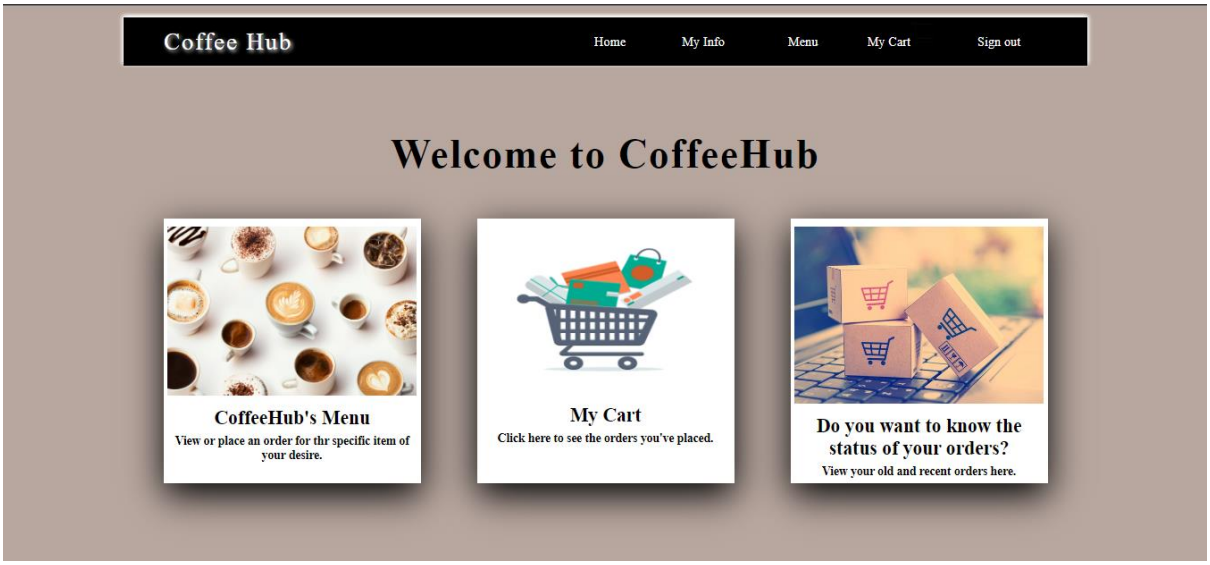


Image showing customer page after logging in.

***** The tests were performed on only one registration form and one login form. The same functionality applies to the all the other forms. Please check code in appendix.***

Registration forms:

Test and Description	Element tested	Expected Results	Actual Results	Explanation of actual results
Whitebox testing (Registration): this testing will check the internal working of the forms to see if it is functioning as it should.	All the input fields of the forms to see if it will be saved to the database.	All inputted information is inputted into database as expected	All information was inputted into the database (Check images above)	The actual results is the same as the expected results.

Login forms:

Test and Description	**Element tested**	**Expected Results**	**Actual Results**	**Explanation of actual results**
Whitebox testing (Login): this testing will check the internal working of the forms to see if it is functioning as it should.	All the input fields of the forms to see if the credentials will be checked with that is in the database and will send a response back to the user via the website.	All inputted information is checked with the info in the database and user was able to login into their respective page according to the type of user	All inputted information has been check with the info in the database and user was able to login into their respective page according to the type of user	The actual results is the same as the expected results.

Website and Database

For this test, we will be checking how information is displayed from the database and onto the website and if everything is functioning as it should.

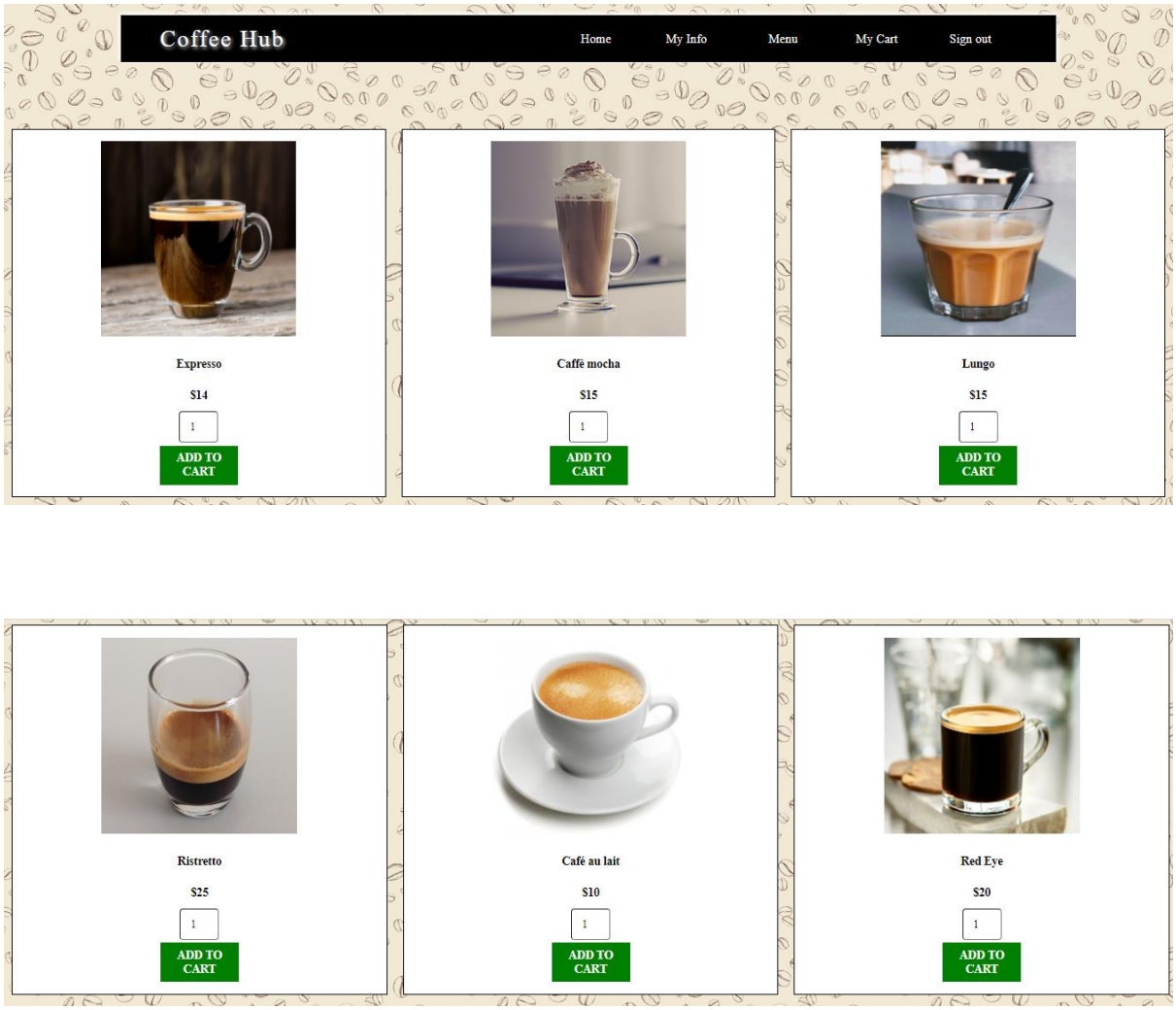


Image showing menu items being generated from database.

Customer Info

Customer ID	5
Customer Name	Akhil Montrose
Username	monty
Email	akhildmontrose@gmail.com
Address	Fyzabad
Contact	275-5992
Password	monty97

Delete Account

Image displaying information from the database.

My Orders

ID	Order Date	Item Name	Item Cost	Quantity	Item Total	Status
2	2022-05-20 17:17:56	Café au lait	$10.00	1	$10.00	Delivered
4	2022-05-20 17:27:14	Café au lait	$10.00	1	$10.00	Delivered
4	2022-05-20 17:27:15	Chocolate Muffin	$11.00	1	$11.00	Delivered
5	2022-05-20 19:03:26	Blueberry Muffin	$12.00	1	$12.00	En Route
5	2022-05-20 19:03:26	Expresso	$14.00	1	$14.00	En Route
6	2022-05-29 23:05:09	Blueberry Muffin	$12.00	1	$12.00	Delivered
6	2022-05-29 23:05:09	Banana Muffin	$9.00	1	$9.00	Delivered
6	2022-05-29 23:05:09	Expresso	$14.00	1	$14.00	Delivered

Image showing past orders made by customers on the website (information extracted from database).

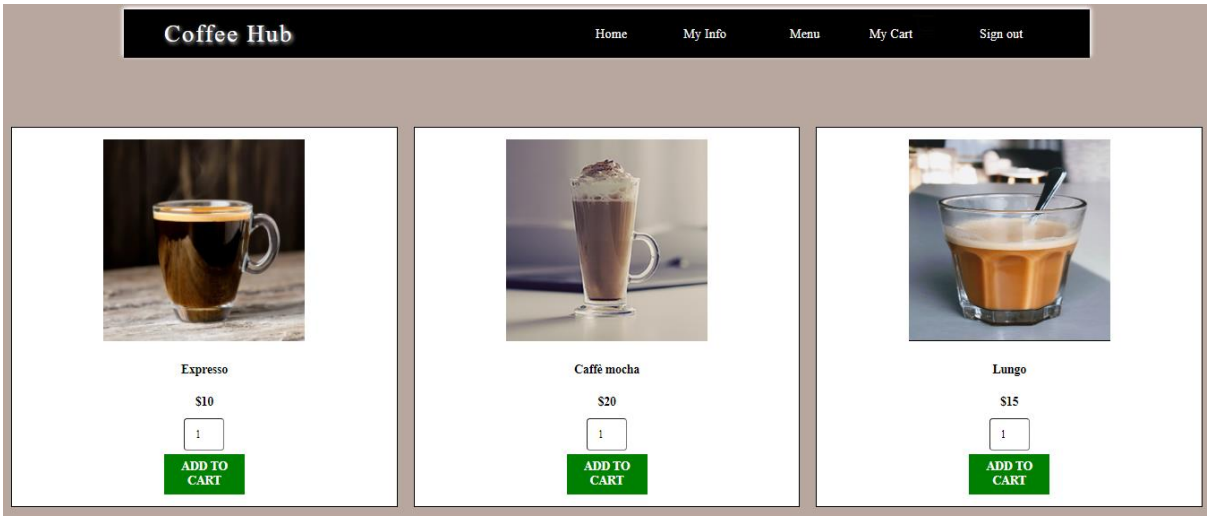


Image showing menu for customers and add to cart button.

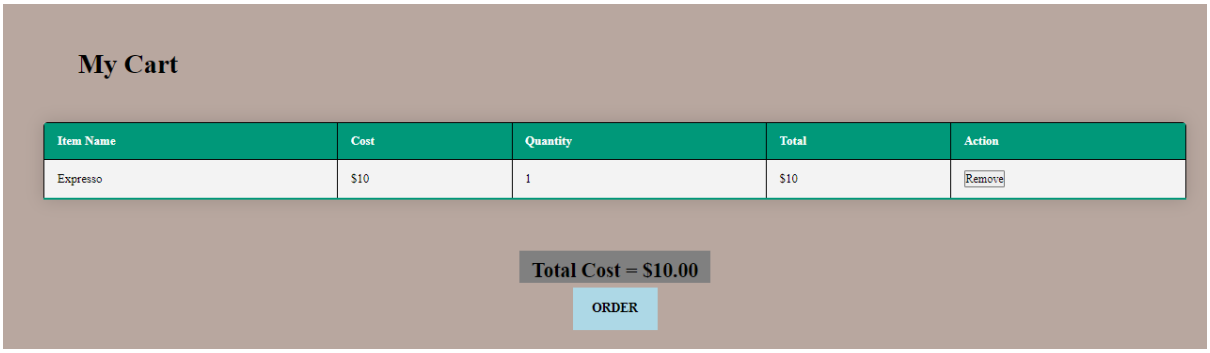
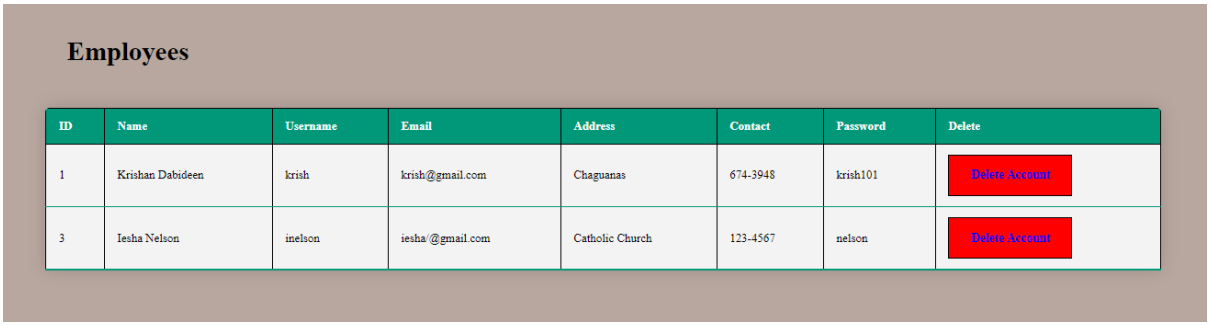


Image showing cart page on customer

5	2022-05-20 19:03:26	Blueberry Muffin	$12.00	1	$12.00	Akhil Montrose	Fyzabad	275-5992	En Route	Edit
5	2022-05-20 19:03:26	Expresso	$14.00	1	$14.00	Akhil Montrose	Fyzabad	275-5992	En Route	Edit
6	2022-05-29 23:05:09	Blueberry Muffin	$12.00	1	$12.00	Akhil Montrose	Fyzabad	275-5992	Delivered	Edit
6	2022-05-29 23:05:09	Banana Muffin	$9.00	1	$9.00	Akhil Montrose	Fyzabad	275-5992	Delivered	Edit
6	2022-05-29 23:05:09	Expresso	$14.00	1	$14.00	Akhil Montrose	Fyzabad	275-5992	Delivered	Edit
7	2022-06-05 21:32:05	Expresso	$10.00	1	$10.00	Akhil Montrose	Fyzabad	275-5992		Edit

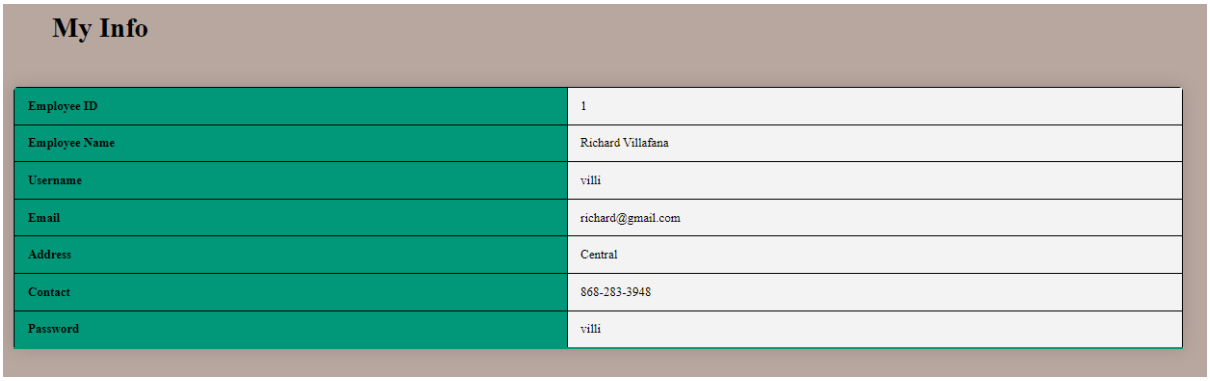
Image showing all customer and order information for admin to view/update (Please note that both admin and employee roles share thus same privilege)



Employees

ID	Name	Username	Email	Address	Contact	Password	Delete
1	Krishan Dabideen	krish	krish@gmail.com	Chaguanas	674-3948	krish101	Delete Account
3	Iesha Nelson	inelson	iesha'@gmail.com	Catholic Church	123-4567	nelson	Delete Account

Image showing all employees registered by the admin on the system



My Info

Employee ID	1
Employee Name	Richard Villafana
Username	villi
Email	richard@gmail.com
Address	Central
Contact	868-283-3948
Password	villi

Image displaying admin information

Employee ID	1
Employee Name	Krishan Dabideen
Username	krish
Email	krish@gmail.com
Address	Chaguanas
Contact	674-3948
Password	krish101

Image showing employee information

Customer:

Test and Description	Element tested	Expected Results	Actual Results	Explanation of actual results
Test to see if everything that is required to be shown on the customer side is shown	All the pages on the customer side	All results (Orders, menu, customer info etc) is shown	All results are shown (See images above).	The actual results are the same as the expected results
Tests to see that the customer can log in and make an order on an item as well as see orders.	All pages that correspond with ordering from menu and displaying in cart and order page	All pages are displaying the correct information (all buttons, tables menu, shopping cart, orders)	All pages are responding and displaying as it should (see images above)	The actual results are the same as the expected results.
Whitebox testing	All website and database components on the customer side	All pages are working as it should. If incorrect actions are taken at any point, error messages will pop up.	All pages are working as intended.	The actual results are the same as the expected results.
Load Time testing: This will be testing the load time of all the pages that display information.	Pages and their load times	Load time to display information is very fast	Pages with information is loading up as expected.	The actual results were the same as the expected results.

Employee:

Test and Description	Element tested	Expected Results	Actual Results	Explanation of actual results
Test to see if everything that is required to be shown on the employee side is shown	All the pages on the employee side	All results (Orders, customer info etc) are shown	All results are shown (See images above).	The actual results are the same as the expected results
Tests to see that the employee can log in and edit order status as well as view employee info	All pages that correspond with orders, and employee info.	All pages are displaying the correct information (all buttons, tables, orders)	All pages are responding and displaying as it should (see images above)	The actual results are the same as the expected results.
Whitebox testing	All website and database components on the employee side	All pages are working as it should. If incorrect actions are taken at any point, error messages will pop up.	All pages are working as intended.	The actual results are the same as the expected results.
Load Time testing: This will be testing the load time of all the pages that display information.	Pages and their load times	Load time to display information is very fast	Pages with information is loading up as expected.	The actual results were the same as the expected results.

Admin:

Test and Description	**Element tested**	**Expected Results**	**Actual Results**	**Explanation of actual results**
Test to see if everything that is required to be shown on the admin side is shown	All the pages on the admin side	All results (Orders, admin info, menu, employee tables, forms, etc) are shown	All results are shown (See images above).	The actual results are the same as the expected results.
Tests to see that the admin can log in and edit order status, add menu, view menu as well as view/add/delete employees	All pages that correspond with what is supposed to be shown	All pages are displaying the correct information (all buttons, tables, orders, menu, etc)	All pages are responding and displaying as it should (see images above)	The actual results are the same as the expected results.
Whitebox testing	All website and database components on the employee side	All pages are working as it should. If incorrect actions are taken at any point, error messages will pop up.	All pages are working as intended.	The actual results are the same as the expected results.
Load Time testing: This will be testing the load time of all the pages that display information.	Pages and their load times	Load time to display information is very fast	Pages with information is loading up as expected.	The actual results were the same as the expected results.

Guest:

Test and Description	**Element tested**	**Expected Results**	**Actual Results**	**Explanation of actual results**
Whitebox testing	All website and database components on the employee side	All pages are working as it should. If incorrect actions are taken at any point, error messages will pop up.	All pages are working as intended.	The actual results are the same as the expected results.
Load Time testing: This will be testing the load time of all the pages that display information.	Pages and their load times	Load time to display information is very fast	Pages with information is loading up as expected.	The actual results were the same as the expected results.
This test will check to see if the menu, and login/registration forms are working as intended	All pages that correspond with a guest	All pages are working/displaying as it should	All pages are working/displaying as it should.	The actual results are the same as the expected results.

Total System Testing

In this section, we will be testing all of the components together (Forms, website, and database).

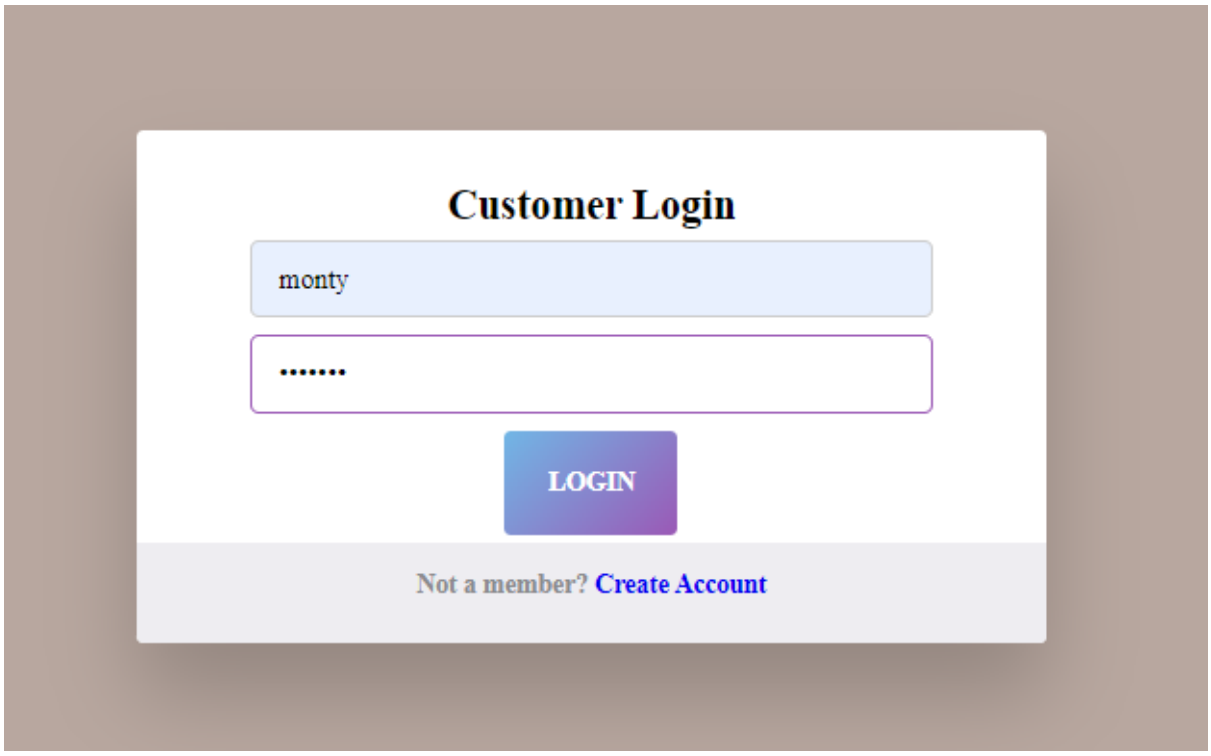


Image showing login for customer

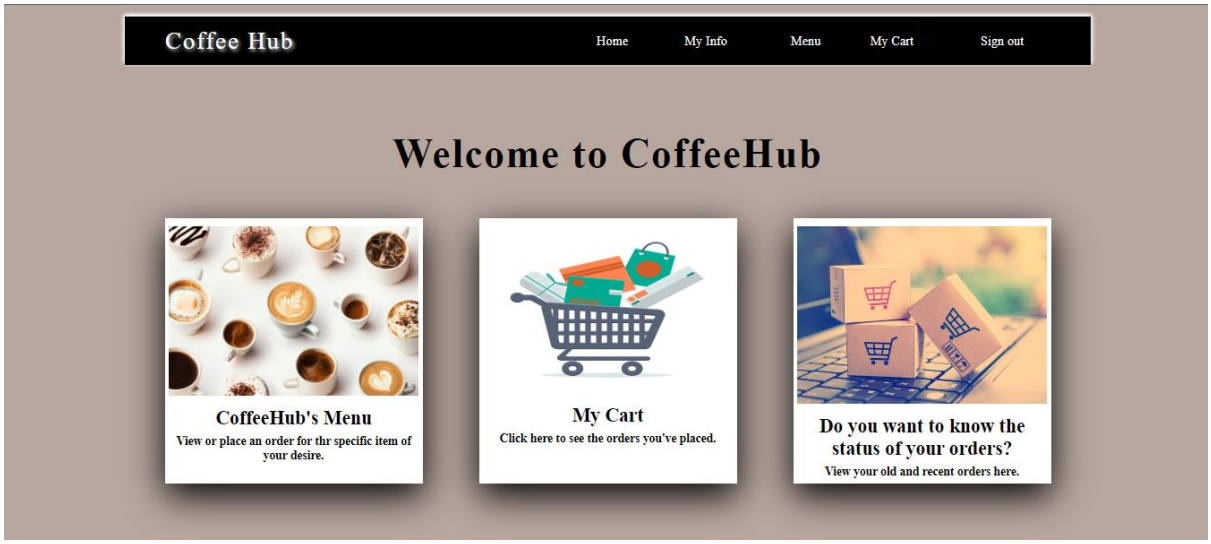


Image showing customer home page.

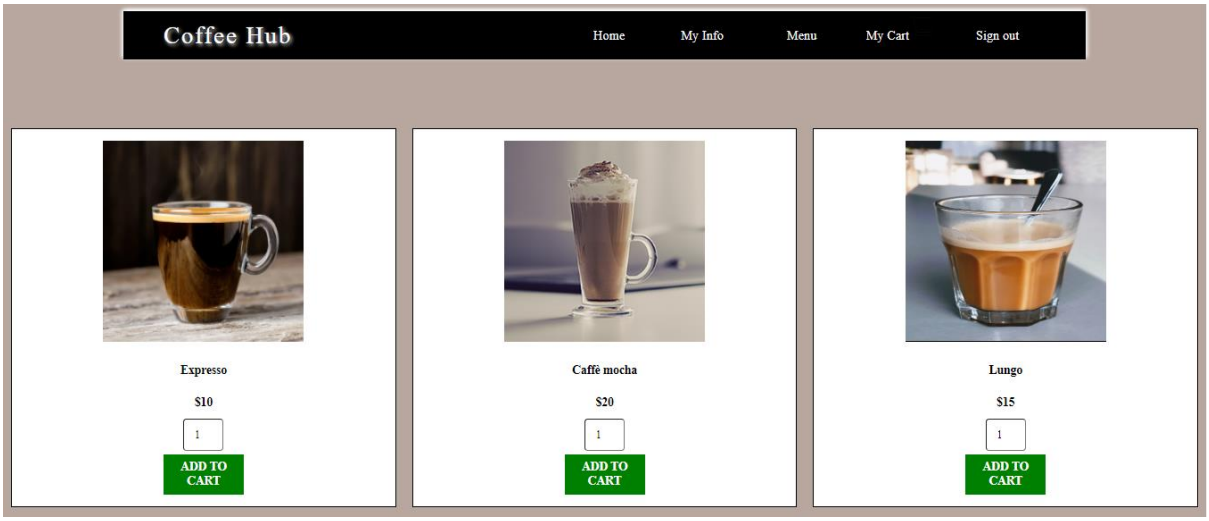
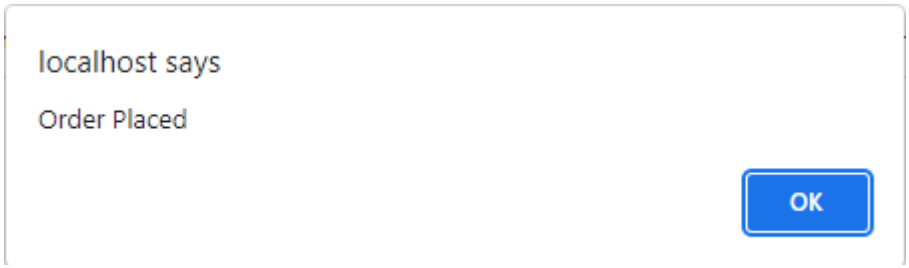


Image showing customer menu page with add to cart button

Image showing two items added to cart and total cost in customer section.



Order placed on items.

My Orders

ID	Order Date	Item Name	Item Cost	Quantity	Item Total	Status
2	2022-05-20 17:17:56	Café au lait	$10.00	1	$10.00	Delivered
4	2022-05-20 17:27:14	Café au lait	$10.00	1	$10.00	Delivered
4	2022-05-20 17:27:15	Chocolate Muffin	$11.00	1	$11.00	Delivered
5	2022-05-20 19:03:26	Blueberry Muffin	$12.00	1	$12.00	En Route
5	2022-05-20 19:03:26	Expresso	$14.00	1	$14.00	En Route
6	2022-05-29 23:05:09	Blueberry Muffin	$12.00	1	$12.00	Delivered
6	2022-05-29 23:05:09	Banana Muffin	$9.00	1	$9.00	Delivered
6	2022-05-29 23:05:09	Expresso	$14.00	1	$14.00	Delivered
7	2022-06-05 21:32:05	Expresso	$10.00	1	$10.00	

Image showing items previously ordered and orders recently ordered. (Status needs to be updated on employee side)

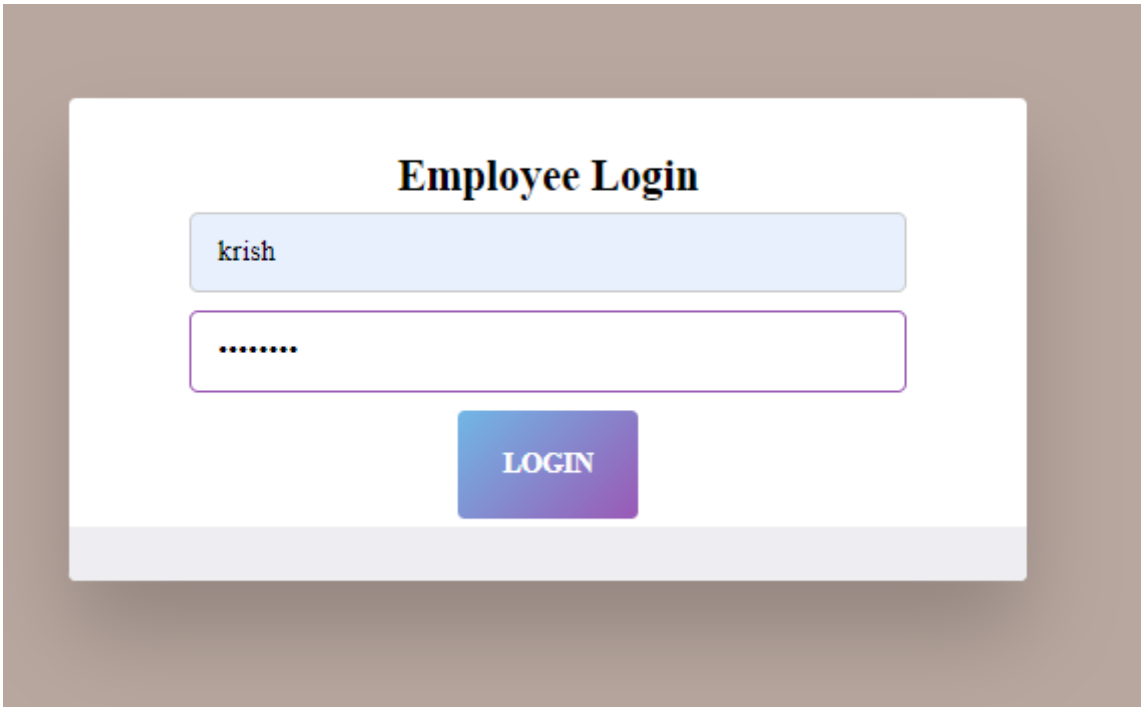
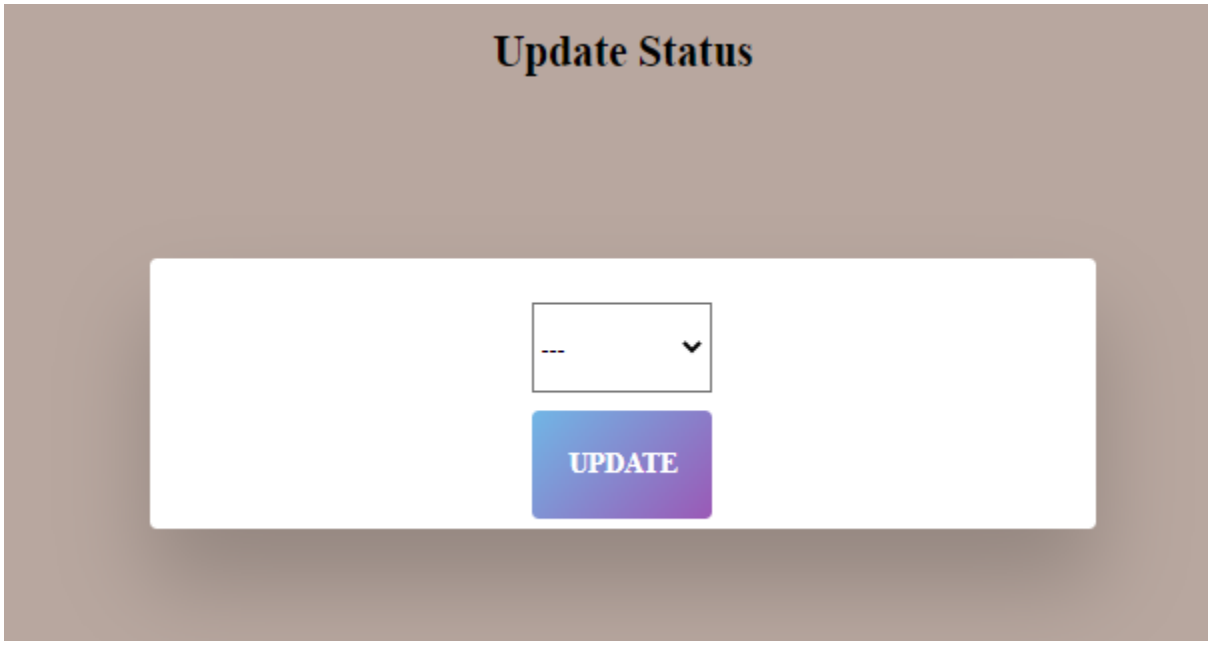


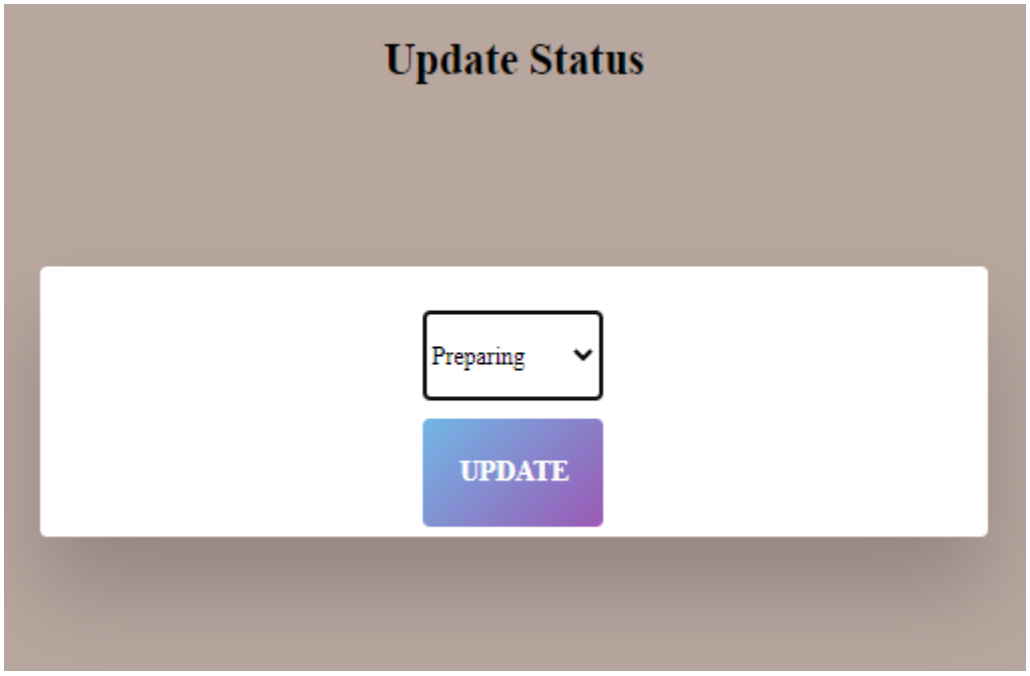
Image showing employee login.

5	2022-05-20 19:03:26	Blueberry Muffin	$12.00	1	$12.00	Akhil Montrose	Fyzabad	275-5992	En Route	Edit
5	2022-05-20 19:03:26	Expresso	$14.00	1	$14.00	Akhil Montrose	Fyzabad	275-5992	En Route	Edit
6	2022-05-29 23:05:09	Blueberry Muffin	$12.00	1	$12.00	Akhil Montrose	Fyzabad	275-5992	Delivered	Edit
6	2022-05-29 23:05:09	Banana Muffin	$9.00	1	$9.00	Akhil Montrose	Fyzabad	275-5992	Delivered	Edit
6	2022-05-29 23:05:09	Expresso	$14.00	1	$14.00	Akhil Montrose	Fyzabad	275-5992	Delivered	Edit
7	2022-06-05 21:32:05	Expresso	$10.00	1	$10.00	Akhil Montrose	Fyzabad	275-5992		Edit

Image showing order section on employee/admin side.



Updating status for orders



5	2022-05-20 19:03:26	Blueberry Muffin	$12.00	1	$12.00	Akhil Montrose	Fyzabad	275-5992	En Route	Edit
5	2022-05-20 19:03:26	Expresso	$14.00	1	$14.00	Akhil Montrose	Fyzabad	275-5992	En Route	Edit
6	2022-05-29 23:05:09	Blueberry Muffin	$12.00	1	$12.00	Akhil Montrose	Fyzabad	275-5992	Delivered	Edit
6	2022-05-29 23:05:09	Banana Muffin	$9.00	1	$9.00	Akhil Montrose	Fyzabad	275-5992	Delivered	Edit
6	2022-05-29 23:05:09	Expresso	$14.00	1	$14.00	Akhil Montrose	Fyzabad	275-5992	Delivered	Edit
7	2022-06-05 21:32:05	Expresso	$10.00	1	$10.00	Akhil Montrose	Fyzabad	275-5992	Preparing	Edit

Image showing the updated status on previously ordered item.

My Orders

ID	Order Date	Item Name	Item Cost	Quantity	Item Total	Status
2	2022-05-20 17:17:56	Café au lait	$10.00	1	$10.00	Delivered
4	2022-05-20 17:27:14	Café au lait	$10.00	1	$10.00	Delivered
4	2022-05-20 17:27:15	Chocolate Muffin	$11.00	1	$11.00	Delivered
5	2022-05-20 19:03:26	Blueberry Muffin	$12.00	1	$12.00	En Route
5	2022-05-20 19:03:26	Expresso	$14.00	1	$14.00	En Route
6	2022-05-29 23:05:09	Blueberry Muffin	$12.00	1	$12.00	Delivered
6	2022-05-29 23:05:09	Banana Muffin	$9.00	1	$9.00	Delivered
6	2022-05-29 23:05:09	Expresso	$14.00	1	$14.00	Delivered
7	2022-06-05 21:32:05	Expresso	$10.00	1	$10.00	Preparing

Image showing the corresponding changes from the employee side also being changed on the customer side.

Customer:

Test and Description	**Element tested**	**Expected Results**	**Actual Results**	**Explanation of actual results**
Test to see if the user can login into the system and access the customer home page	Username and password	Customer is able to login into the system	Customer was able to login into the system (see images above for more details.)	The actual results are the same as the expected results.
Tests to see if the user can add an item to cart and make an order.	Selecting an item from menu and placing an order	Customer is able to add an item(s) to cart and make an order.	Customer was able to add item(s) to cart and make an order (see images above for more details)	The actual results are the same as the expected results.

Tests to see if the customer is able to view the order that was placed as well as the status of the order.	Order table showing items previously ordered as well as the order status.	The table is being presented to the user (customer) as it should.	The table was being presented to the customer as it should. (see images above for more details)	The actual results are the same as the expected results.

Admin/Employee:

Test and Description	**Element tested**	**Expected Results**	**Actual Results**	**Explanation of actual results**
Test to see if the user can login into the system and access the employee/admin home page	Username and password	Employee/admin is able to login into the system	Employee/admin was able to login into the system (see images above for more details.)	The actual results are the same as the expected results.
Test to see if the employee/admin can view/edit the order status of items	Tables displaying information about customer and orders.	Employee/admin is able to update/view order status	Employee/admin was able to update/view the order status of orders. (see images above for more details.	The actual results are the same as the expected results.

User testing

In this section, we will allow three users of varying technical experience to test the website. The user were requested to log into the system, to view the menu, add menu item to cart and place an order. I also requested them to view their order status.

Test 1: Technical User

This user has a lot of experience navigating websites on multiple devices.

Test and Description	**Element tested**	**Expected Results**	**Actual Results**	**Explanation of actual results**
Testing to see how the user navigates through the website	The entire customer side of the website	The user is able to navigate through the website fairly quickly.	The user was able to easily navigate through the website. The user was able sign up and login and make an order. In less than 3 minutes.	The actual results were the same as the expected results.
Visibility: Testing to see how visually friendly the website is.	The entire customer side of the website	The user had no issues with visibility	The user had no visual issues	The actual results were similar to the expected results.

Test 2: Non-Technical User

This user has little experience with navigating websites.

Test and Description	**Element tested**	**Expected Results**	**Actual Results**	**Explanation of actual results**
Testing to see how the user navigates through the website	The entire customer side of the website	The user is able to navigate through the website but not as fast as an experienced user.	The user was able to navigate through the website with some assistance. The user was able sign up and login and make an order. Everything was completed at around 5 minutes.	The actual results were the same as the expected results.
Visibility: Testing to see how visually friendly the website is.	The entire customer side of the website	The user had no issues.	The user was not able to see the mouse at times because of the background.	The actual results were different from the expected results.

Test 3: Moderately Technical User

This user has some experience navigating websites on multiple devices.

Test and Description	**Element tested**	**Expected Results**	**Actual Results**	**Explanation of actual results**
Testing to see how the user navigates through the website	The entire customer side of the website	The user is able to navigate through the website fairly quickly.	The user was able to navigate through the website. The user was able sign up and login and make an order. In less than 4 minutes.	The actual results were the same as the expected results.
Visibility: Testing to see how visually friendly the website is.	The entire customer side of the website	The user had no issues with visibility	The user was able to view the website except the mouse because of the background	The background image of the website was fine but it made we mouse hard to track.

Main Points of this Chapter

In this chapter, we discussed the design, Implementation and Testing of individual components such as website, database, forms, etc. We also covered System Testing which consists of testing multiple components together. Finally we covered User Testing which entails tests carried out on users of varying Technical expertise.

In the next chapter, we will be covering Report Summary and & Conclusion.

Report Summary & Conclusion

In this chapter, we will summarize what was done in the previous chapters in particular:

- The major components of the project that were tested individually.
- The major components that were tested together.
- A summary of the testing done by users.

Design, Implementation and Testing of the Individual Components

Forms

The purpose of the forms in this web application is to accept input from different type of users. The forms were used to accept login/sign up information for customers, employees, and the admin.

The forms were designed in a way that would allow the user using the system to know if they are making an error in a particular field or leaving an entire field blank when it is required to input data. For more information about the from please see pages 23-32 of this document.

The way that the forms were implemented, the languages that were used were HTML, PHP as well as JavaScript (for popups). Please refer to pages 23-32 for more information.

Testing were done on the forms which include:

- Blackbox testing
- Whitebox testing
- Unit testing
- Load time testing.

After tests, it was observed that no problems occurred. Everything from error checking, input checking as well as the overall load time and design of the forms were working as it should. For more information please go to pages 26 and 27 as well as page 31.

User Interface

The purpose of this component is to display the information for the user(s) to view and interact with. The UI had a major part to play in all of the components on the website.

The languages that were used to build the UI were HTML but mostly CSS (Cascading Style Sheets.) For ore information on this, please refer to pages 33-35.

Four tests were done on this component to see if everything functioned as it should according to the requirements. These include:

- Blackbox Testing
- Whitebox Testing
- Load time testing
- Unit testing

After all the tests were carried out, no errors had occurred. Everything that needed to be displayed for the user(s) was displayed. For more information on this, please check pages 34-35.

Database

The purpose of the database is to store information that was inputted into the system. The database was used in multiple ways in this web application. For example:

- Displaying menu.
- Registration of different type of users.
- Adding items to cart.
- Creating orders.
- Displaying past orders as well as user information.

The platform that was used to create the database was XAMPP and the DBMS used was MySQL. The other language that was used to transfer the data from the forms, buttons, etc to the data base and vice-versa was PHP. Please refer to page 37 of the document for more information.

Some things that can be done to improve the database especially the order table is to have a column called total which displays the total for all the items a person bought.

Summary of the Testing of the Entire System

The summary of this section is to show how all of the major components combine together to function as a whole. So in other words, the forms, website, and database were all linked together. You can refer to this in pages 52-58.

Multiple tests were done on two type of users: Customer and admin/Employee. In these tests, we investigated how they both logged in and perform different actions (Customer needed to login, add item to cart, make order, as well as view order status. Employee needs to sign in, view orders as well as change order status on items ordered). These tests incorporated all of the components (forms, database, website(UI)).

After testing, both the customer and admin/employee was able to perform all of their actions without any issues. For more information on this please refer to pages 56-58.

Summary of the User Testing

In this section of the report, we will be summarizing the user tests that were performed on three individuals of varying technical expertise. The test were based on different functionalities of the system (Creating an account, logging in, viewing menu, adding menu to cart, ordering item, and viewing order status.

Two main tests were done in order to see if the user was able to use the system and get appropriate feedback. Th first test was to see if the user can navigate through the entire website. This would include everything from signing up to making an order. The other test was to check the User interface/Visibility of the entire website.

After the tests were completed, the results varied from user. For example, the technical user had no issues whatsoever with navigating through the website as well as the interface. The moderately technical user did fairly well with the tests. The Non-technical user was able to navigate through the website but did it in shorter time. Also both the moderately technical and non-technical user had an issue with seeing the mouse due to the background image while the technical user did not. One can assume that is because they technical user has more experience while the others did not. You can refer to pages 59-61 for more information on the results.

As mentioned above, both users who were non-technical and moderately technical had an issue with seeing the mouse at times. Changes were made so that the mouse would be more visible to them.

Final Thoughts

In this chapter, we discussed all of the design, implementation of the web app that was built as well as the tests that were done in a summarized format.

Coming to the end of this document, after looking at the requirements that were made in phase 1 of this project, the functional requirements were met. Also additional requirements were added into the system. Initially, for example the database prior to building the project would theoretically have four tables but after building, six tables were required for the entire website to function as it should. Also A system was to be built that did not need a shopping cart but when building started, it was suitable to implement a shopping cart into the system to manage orders.

While the major functionalities that were laid out in phase 1 of the project were met, there is one main thing that can be improved in the future. Instead of using a shopping cart based on sessions, a table can be created in order to store the cart items. Within the current system, if a user adds an item in the cart and logs out, all the sessions created in the cart is destroyed. The current system does not impact the functional requirements but in the future, if the system needs to be expanded, a table to store cart information would be recommended.

Some factors that had an effect on the time in which it took to complete this project include:

- Time.
- Other personal engagements.
- Covid infection.

We can conclude that we can make an online system where a user can browse coffee on a menu and place an order on it. It is also possible to create this app using HTML, CSS, PHP, SQL, and JavaScript.

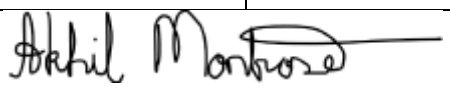
Appendix

Project Proposal

THE UNIVERSITY OF TRINIDAD AND TOBAGO	Project Proposal Form for B.A.Sc. in Computer Engineering		
Section A – Personnel Information			
Student Name	Akhil Montrose	Supervisor(s) Name	Ken Sooknanan
Student ID	81788	Supervisor Email:	Ken.sooknanan@utt.edu.tt
Student Email Address	akhilmontrose62@gmail.com	Supervisor Phone #	1(868)-793-8671
Student Phone #	275-5992		
Section B – Project Information			
Project Title: Coffee Hub			
Project Type (Development or Research): Development			
Project Overview (describe the project in 1-2 lines, and list all its deliverables): The aim of this project is to create a web-based application for an online coffee delivery service named CoffeeHub. 1. Employees using the web app will be able to create an account/login. 2. Customer will be able to create an account/login into the system. 3. Their information (username, address, password etc) will be stored on a database through the use of forms and can also be retrieved as well.			

4. This also is the case for the items on the menu. Items on the menu will be stored in the database and can be retrieved at any point.
5. Employees will be able to Create/Update/Delete/View the menu while
6. The customer will be able to View as well as Order from the menu.

Section C – Administrative Information

Course Code	PROJ3019	Course Title	Final Project Design -ICT
Semester (I or II)	I	Project Time Frame (est.)	September 2021 – May 2022
Student Signature & Date:			
Supervisor Signature & Date:			
Project Coordinator Signature & Date:			

Code

I have opted out of copying and pasting the code in the document because of the vast number of pages of code it took to build this project. It would be troublesome scrolling through this document thus affecting the experience of the reader(s).

To view the code used to build this project please go to my GitHub page at: https://github.com/monty397/FinalP.

Bibliography

What Is a Web Application? How It Works, Benefits and Examples

https://www.indeed.com/career-advice/career-development/what-is-web-application

Nov 05, 2021 (Susan C.) What Is HTML? Hypertext Markup Language Basics Explained

https://www.hostinger.com/tutorials/what-is-html

What Is CSS and Why Should You Use It?

https://blog.devmountain.com/what-is-css-and-why-use-it/

What is PHP?

https://www.php.net/manual/en/intro-whatis.php