

National College of Ireland

BSc (Honours) in Computing - Evening

Software Development

Academic Year 2020/2021

Student Name: Daniel Costel Neagu

Student Number: x17128463

Student Email: x17124863@student.ncirl.ie

Farmers Market Technical Report

Contents

Executive Summary	3
1.0 Introduction.....	4
1.1. Background.....	4
1.2. Aims	5
1.3. Technology	5
1.4. Structure	6
2.0 System.....	7
2.1. Requirements.....	7
2.1.1. Functional Requirements.....	7
2.1.1.1. Use Case Diagram.....	8
2.1.1.2. Requirement 1: View/Browse Products	9
2.1.1.3. Requirement 2: Register User	11
2.1.1.4. Requirement 3: Log in	13
2.1.1.5. Requirement 4: Update or Delete Account	15
2.1.1.6. Requirement 5: Add to Basket	17
2.1.1.7. Requirement 6: Manage Basket.....	19
2.1.1.8. Requirement 7: Upload Products.....	20
2.1.1.9. Requirement 8: View Orders.....	22
2.1.2. Data Requirements	24
2.1.4. User Requirements	27
2.1.5. Environmental Requirements	27
2.1.6. Usability Requirements	28
2.2. Design & Architecture.....	28
2.3. Graphical User Interface (GUI).....	30
Farmers Market E-Commerce Web site GUI Layout	30
Admin Dashboard For Farmers Market E-Commerce App GUI Layout...	40
2.4. Testing.....	46

Site Accessibility Testing	46
Usability Design Testing	48
2.5. Evaluation.....	53
3.0 Conclusions.....	57
4.0 Further Development or Research	58
5.0 References.....	58
6.0 Appendices	59
6.1. Project Proposal.....	59
7.0 Objectives.....	61
8.0 Background	61
9.0 Technical Approach.....	62
10.0 Special Resources Required.....	63
11.0 Project Plan	63
6.0 Gantt Chart.....	64
7.0 Technical Details	65
8.0 Evaluation.....	66
9.0 References	66
11.1. Reflective Journals	67
Project Journal	69
Introduction.....	70
Week 1: 28th September / 04th October	71
Market Research	72
Idea	72
Week 2: 05th October / 11th October	72
Week 3: 12th October / 18th October	72
Week 4: 19th October / 25th October	73
Vision Mission Statement	73
Week 5: 26th October / 1st November	74
Week 6: 2nd November / 8th November	74
Week 7: 9th November / 15th November	75
Week 8: 16th November / 22nd November	75
Week 9: 23rd November / 29th November	75

Week 10: 30th November / 6th December	76
Week 11: 7th December / 13th December	77
Week 12: 14th December / 20th December	77
Week 13: 21st December / 27th December	78
Week 14: 28th December / 3rd January	78
Week 15: 4th January / 10th January	78
Week 16: 11th January / 17th January	79
Week 17: 18th January / 24th January	80
Week 18: 25th January / 31st January	80
Week 19: 1st February / 7th February	81
Week 20: 8th February / 14th February	81
Week 21: 15th February / 21st February	82
Week 22: 22nd February / 28th February	82
Week 23: 1st March / 7th March	82
Week 24: 8th March / 14th March	83
Week 25: 15th March / 21st March	83
Week 26: 22nd March / 28th March	84
Week 27: 29th March / 4th April	84
Week 28: 5th April / 11th April	85
Week 29: 12th April / 18th April	85
Week 30: 19th April / 25th April	86
Week 31: 26th April / 2nd May	86

[Executive Summary](#)

This report exhibits my ideas on how I have managed to design and develop an E-Commerce Web site, the newly built Web site is having its activity-oriented around the Farmers Market model.

The customers can browse, view and buy the available merchandise for sale on the Farmers Market E-Commerce Web site at <https://farmers-market-e-commerce.herokuapp.com/>.

The produces and goods are uploaded by the local farmers and producers. To make sure that the goods on sale are real, they are revised and approved by one of the admins via the Admin Dashboard For Farmers Market E-Commerce Application available at <https://farmers-market-admin-app.herokuapp.com/>.

The shoppers can purchase goods as soon they login into the system, of course after they register first into the system with us. For the farmers and producers to be able to access our system first they are screened to make sure that they are part of the local community, then they will be given a trader account where they can upload their products to be able to advertise and sell them.

This project is trying to move the conventional brick and mortar model of buying and selling of goods and produces, found in a market as the Farmers Market environment, to the online environment. This is possible by moving it to the Cloud by using modern methods and technologies.

One main benefit by doing so, the online environment is given access for more people to this type of market and not only that, but it is restarting a very affected sector of the economy at this current time.

1.0 Introduction

The purpose of this document is to give an overview of the technical details, documentation and to show how I used them to successfully deploy a state-of-the-art E-Commerce Web site. The deployment is on the Heroku Cloud provider platform which is providing a new way of facilitating the trading of local grown and outsourced produces and goods.

1.1. Background

I undertook the development of this project as I was directly affected at the begging of this year by the closure of my local Farmers Market, due to current measures put in place by the Government to stop the spread of the Covi-19 virus Pandemic.

One of this hardship was that by not being able to physically visit and purchase anymore my favourite local sourced products from the Framers Market.

Another factor that contributed to my decision to start developing a new alternative to this disrupted market was the fact that I was recently relocated with my accommodation and work, and going to one of these local Farmers Markets in person was out of my reach as I wasn't within the allowed travelling limits anymore imposed by the authorities thru the year.

As I wanted to access and have all my necessary and favourite artisan foods, cheeses, meats, organic vegetables and fruits I went online, and I searched to find a way to purchase them again. The online search didn't return any relevant information regarding any place from where to get them directly or indirectly from.

Reflecting on this current situation I got the idea on how to fix this problem, by developing my E-Commerce website I will bring back the local community to buy and sell locally again at a larger scale and in a much safer way, from the reach of a click.

1.2. Aims

By developing this project, I am creating a new online platform where it is facilitating the trade of local goods and where the local people can purchase again their favourite fresh and in-season products as they did once in their local Farmers Markets, but this time online by creating an account with us. Buying online is giving them safety by not exposing them to the risk of contracting the virus from other people next to them in a crowded market.

My new E-Commerce Web site is making available to its customers a large variety of fresh and in-season produce, with artisan products too that are worked and grown locally by the local farmers and producers that are the most affected by these strange times that we all are living in.

The Farmers Market E-Commerce Web site is available to be accessed online via a multitude of web browsers and network attached devices like smartphones, PCs and tablets.

By providing to the local farmers my newly built E-Commerce Platform, I am enabling them to advertise and sell their products as an alternative to this disrupted market by the current Global Pandemic.

1.3. Technology

Tools and technologies used to complete my Web site are my personal laptop that is running Windows 10, where a copy of my progress was made to keep it locally on a separate partition on the disk drive and a backup repository that was uploaded in my GitHub account and is available to view at <https://github.com/DanielCostelNeagu/Final-Project>.

Development

Visual Studio Code is a free code editor made by Microsoft and it was used to edit my code for my E-Commerce Web site Application.

Popular coding languages and libraries used for this project are HTML5 to display the Web site, CSS for styling of the Web site, JavaScript and Node.js to enable interaction on the Web site, JAVA for the back end server, JSON and React libraries for rendering dynamic content.

Storage

A document-oriented database program is used in this project from MongoDB Inc. which is a cross-platform for NoSQL Database environment, as it is fully managed, has built-in controls methods for security protection, it is scalable and cost-efficient comparing from their graphs on the website.

Deployment

Heroku is the chosen Cloud Provider used to deploy my Web site Application, as it comes free of charge. Another factor that made me decide on deploying my Web site Application on Heroku is that it can host all three servers, the back-end, the admin application and the Farmers Market Web site.

1.4. Structure

A short description of the structure of my document and with an overview that is addressing in each section is listed below:

Section 1

In this part of the document, I will make an introduction to the Report with the Background on explaining my decision making on undertaking to develop this project, the Aim that I am trying to achieve by developing my E-Commerce Web site Application.

Here in this section, I will show the technology used in this project development and for the last part, the Structure section where I will give a brief description of the structure of my document enumerating each section of it.

Section 2

This section is the System where are enumerated all the functional requirements necessary for the project development, I am making use of Case Diagrams to show all the features of the System.

Section 3

In this section, I am presenting my Conclusion drawn for the project build and documentation.

Section 4

In this section, I am including all my References used throughout my document and I will use the NCI Library's guidance to reference the information in Harvard Style.

Section 5

This section contains the Project Proposal and all Monthly Project Journal's as an Appendix and is supplementary to the main body of this report.

2.0 System

2.1. Requirements

2.1.1. Functional Requirements

In this section, I am listing the functional requirements as use cases in their rank order by giving them a short description of what my application is going when is utilised.

1)View Products:

This is allowing anyone to view and browse the website without being registered, a list of products is displayed to them.

2)Register User:

This use case shows the interaction of the User with the application by creating an account in the database system and where a User name and a secure password are going to be created to be able to identify each person that is going to do any transaction on the application.

3)Login:

This use case enables a User to identify themselves in the system so any transactions can be recorded into their records.

4)Update or Delete Account:

Any user can make changes in his profile account or permanently deleting his details from the database.

5)Add to Basket:

This use case enables the creation of a virtual basket or shopping chart where the customer can add products so he can proceed to purchase them at checkout.

6)Manage Basket:

A user can at any time change the content of products in his basket by adding or removing items and making a payment or cancel the payment.

7)Upload Products:

A Farmer or Producer can upload produces to the listing list so they can sell them.

8)View Orders:

Users can view their transactions done in the application system and a detailed order report and status can be obtained from it.

2.1.1.1. Use Case Diagram

An overview of Farmers Market System functional requirements is provided below in Figure 1 under the form of a Use Case Diagram.

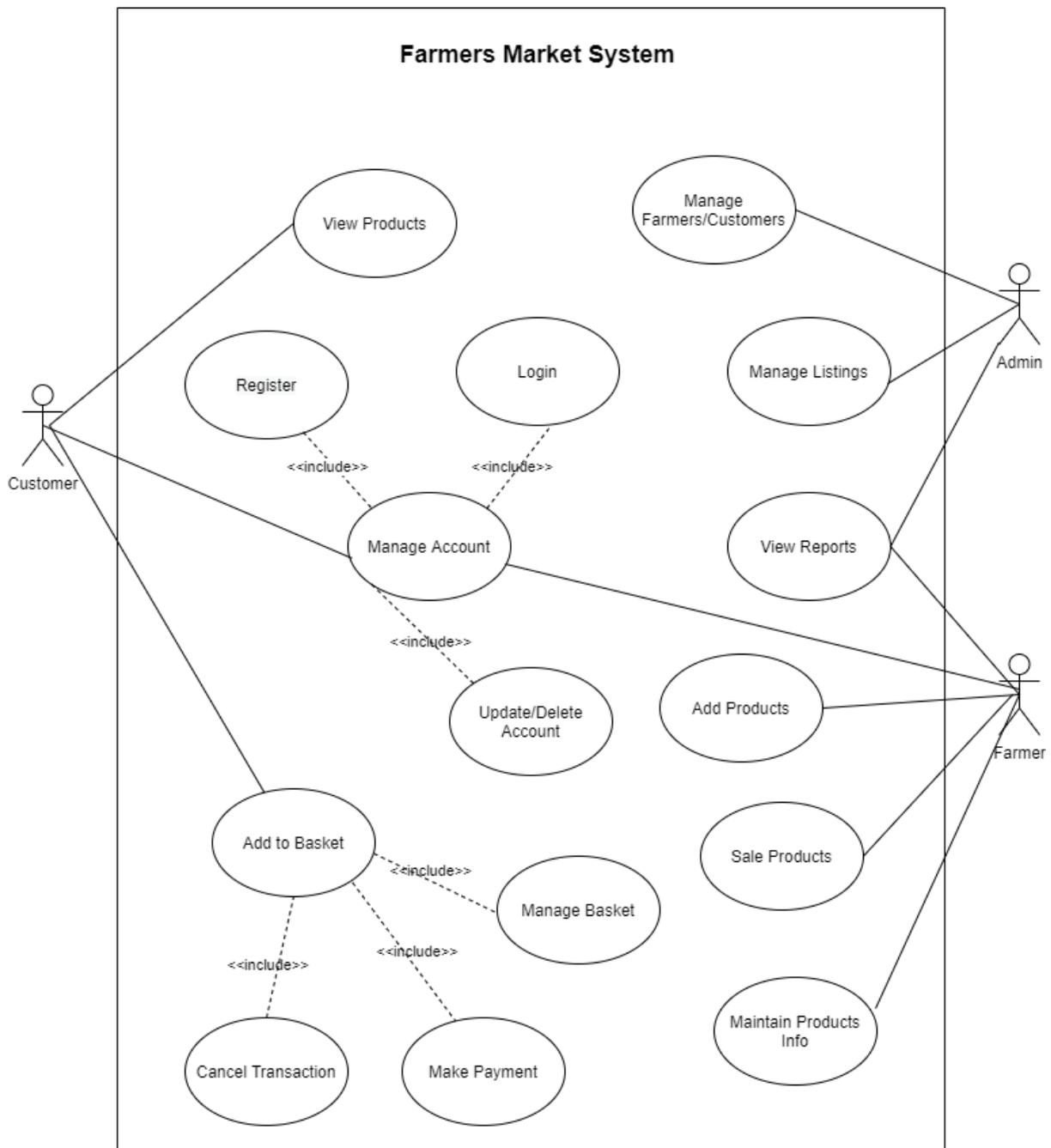


Figure 1

2.1.1.2. Requirement 1: View/Browse Products

Description & Priority

After the user accesses the E-Commerce application online using his favourite web browser on any device connected to the internet like a PC, Laptop, Tablet or Smart Phone he or she can view the Home landing page of the website. Here the user/viewer can freely view and

browse the products catalogue offered at that time, no registration or login into the system is required. This Function is a high priority as it is the main listing of products on the website.

Use Case 1

Scope

The scope of this use case is to advertise and show to the User the products available in stock for sale.

Actors

Customer, Farmer, Producer, Display Screen, Database.

Use Case Diagram

A Use Case Diagram is shown below in Figure 2 of the interaction between the Actors of the system.

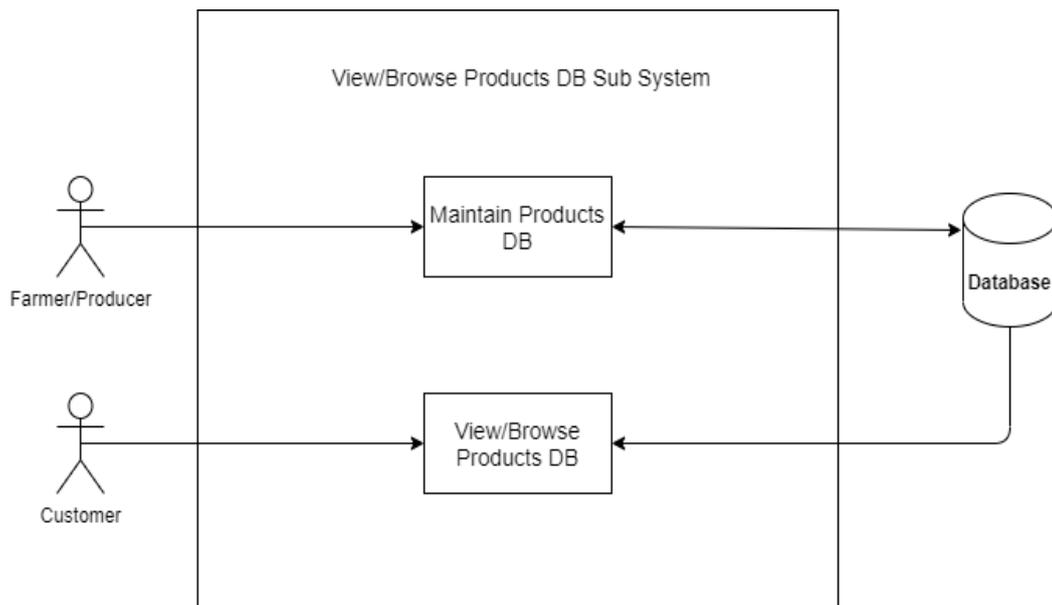


Figure 2

Flow Description

Precondition

The website must be opened on a web browser connected to the internet.

Activation

This use case starts when a user slides/ browses product listing thru the online catalogue.

Main flow

1. The user goes to the home page.
2. The user clicks on the products page link.
3. The system opens a product catalogue stored in DB System.
4. The user views the products.

Termination

The user exits the system before any selection of products is made.

Post condition

The system goes into a wait state until User interaction is detected.

2.1.1.3. Requirement 2: Register User

Description & Priority

This Function is of high priority, it ensures that a new User can get credential to access the system and to make a transaction. The new User is asked to provide his first name, last name, email address, and to create a password. After the successful creation of an account the user data it will be stored in the database.

Use Case 2

Scope

The scope of this use case is to allow User interaction with the system by creating an account with a username and password.

Actors

Customer, Farmer, Producer, Admin, Display Screen, Database.

Use Case Diagram

The Use Case Diagram shown below in Figure 3 describes the Registration of a new User subsystem in the database.

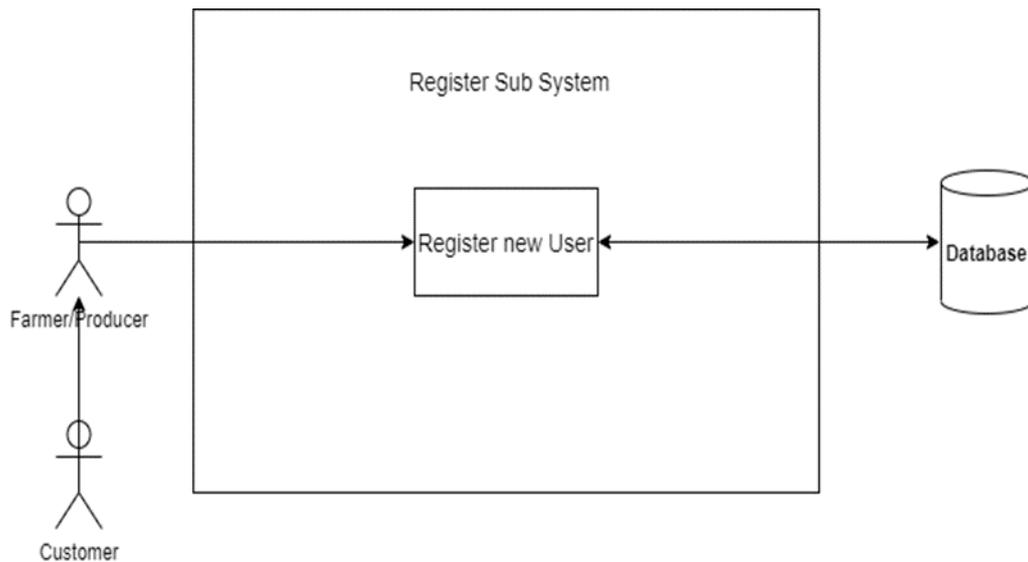


Figure 3

Flow Description

Precondition

The system is in idle mode on the register page.

Activation

This use case starts when the User presses the Register button.

Main flow

1. The system launches the form to be filed in.
2. The User provides all the mandatory details to create an account.
3. The system creates a new account.
4. User can access the system.

Termination

The system redirects the User to the Login page.

Post condition

The system goes into a wait state until User interaction is detected.

2.1.1.4. Requirement 3: Log in

Description & Priority

This Function is a high priority. It ensures the system has a record of who is using the system to make transactions on it, and they must be registered in the system before they could access it.

Use Case 3

Scope

The scope of this use case is to authenticate the User.

Actors

Customer, Farmer, Producer, Admin, Display Screen, Database.

Use Case Diagram

The below diagram from Figure 4 is represented the Login use case of an existing User in the system.

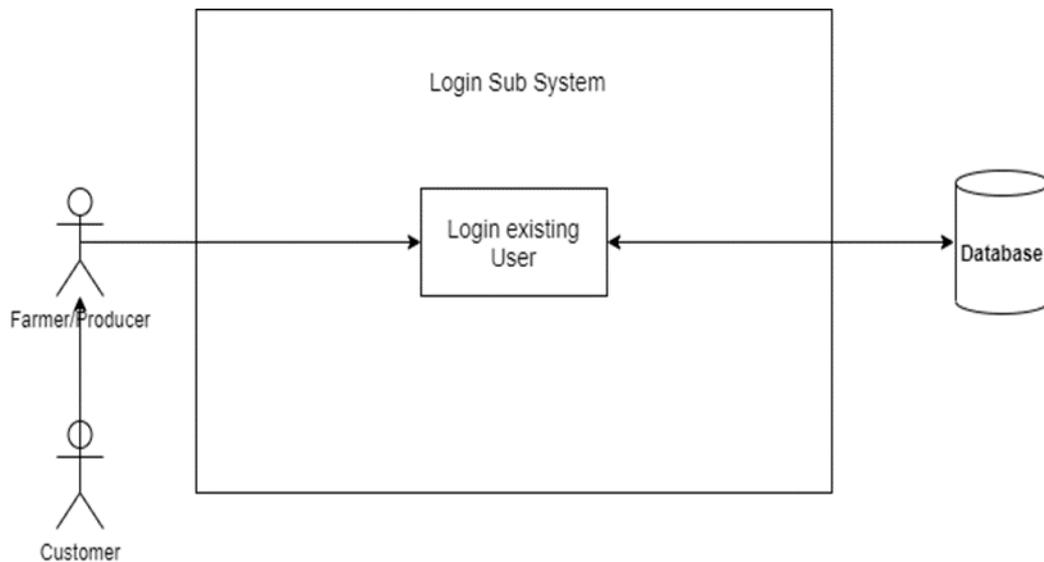


Figure 4

Flow Description

Precondition

The system is idle, and the User is already registered in the system.

Activation

This use case starts when User clicks Log in button.

Main flow

1. The User access the login page.
2. The User is the first time and needs to register (See A1)
3. The User enters username, password and then presses Login button.
4. The User is a Farmer/Producer (See A2)
5. The User enters the wrong credentials.
6. The User doesn't remember account details (See E1).

Alternate flow

A1: Register

1. The system redirects the User to the Register page.
2. The User enters the full name, address, email, phone number, password, and re-types password confirmation.
3. The system redirects the User to the Login page after a successful registration.
4. The use case continues at position 3 of the main flow

A2: Farmer/Producer

1. The system automatically detects what type of User it is.
2. The system redirects to the Farmer/Producer Page.
3. The use case continues in the main flow from position 3.

Exceptional flow

E1: Account Recovery

1. The User clicks the account recovery link.
2. The system prompts User for the email address.

3. The User enters an email address.
4. The system verifies if the email address is registered in the system.
5. The system sends a recovery link into User's email. If not registered User, cannot access the system.
6. The User clicks the link.
7. The system opens a registration page where User can recover the password.
8. The system redirects to the Login page.
9. The use case continues at position 3 of the main flow.

Termination

The User exits the login screen upon valid credentials.

Postcondition

The User is logged in.

2.1.1.5. Requirement 4: Update or Delete Account

Description & Priority

This function has a high priority as it allows Users to make changes to their profile or if needed to delete entirely the account too. By allowing the Users to update their details into the system it is enabling the business to know where to deliver the purchased merchandise on the platform. All saved user information's are kept on the database.

Use Case 4

Scope

The scope of this use case is to maintain an accurate information profile about the users registered on the system.

Actors

Customer, Farmer, Producer, Admin, Display Screen, Database.

Use Case Diagram

In Figure 5 from below it is shown the Update/Delete User in the profile subsystem and the interaction between the user and the system.

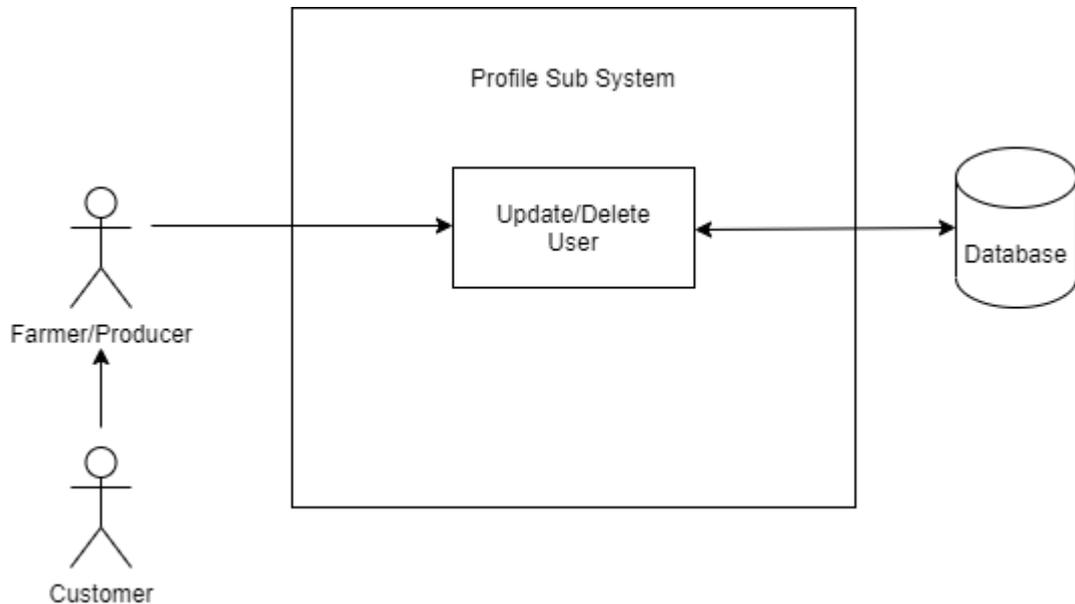


Figure 5

Flow Description

Precondition

The system is in idle mode on the My Profile page.

Activation

This use case starts when the User presses the My Profile link located under the name displayed and is found in any page header.

Main flow

- 1.The system launches the form to be filed in to be updated.
- 2.The User is a Farmer/Producer (See A1)
- 3.The User provides all the information wanted to be updated in the system.
- 4.The system updates and saves profile in the Database system.
- 5.The User can review the updated profile.

Alternate flow

A1: Farmer/Producer

- 1.The system automatically detects what type of User it is.
- 2.The system redirects to the Farmer/Producer Page.
- 3.The use case continues in the main flow from position 3

Termination

The User exits the details form on the My Profile page upon completion.

Post condition

The system goes into a wait state until User interaction is detected.

2.1.1.6. Requirement 5: Add to Basket

Description & Priority

The Add to Basket or Shopping Cart function is of high priority for the system. This function it allows the Users to virtually place the desired items to purchase next on a holder before the checkout is selected by pressing the Place Order button.

Use Case 5

Scope

The scope of this use case is to allow User interaction with the system by creating a virtual shopping basket to track products to be purchased.

Actors

Customer, Display Screen, Database.

Use Case Diagram

The Use Case Diagram shown below in Figure 6 describes the Add to Basket process, the items are placed in a holder on the database and stored until removed or they are paid by the user.

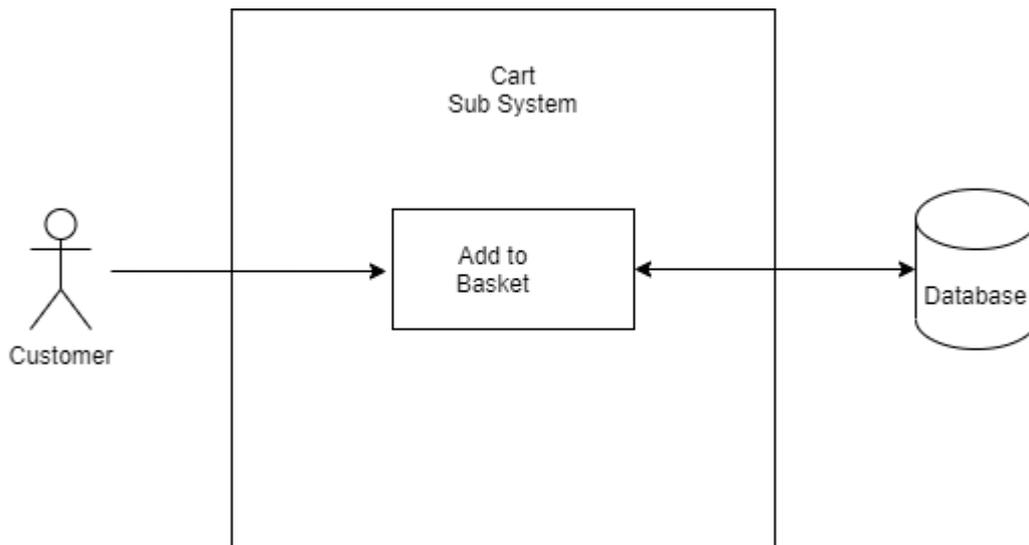


Figure 6

Flow Description

Precondition

The user must select at least one item and place it into the Basket.

Activation

This use case starts when a User presses Add to Cart button on a Product List Page.

Main flow

1. The system identifies the item added to the cart.
2. The system updates the Cart table in the database.
3. The system displays the item/items in the cart.

Exceptional flow

The User exits the system before they have added a product into the basket.

Termination

The system presents the next Place Order button.

Post condition

The system goes into a wait state until User interaction is detected.

2.1.1.7. Requirement 6: Manage Basket

Description & Priority

The Manage Basket of the Shopping Cart function is of high priority for the system. This function it allows the Users to view and edit all merchandise placed in the Basket. The user can alter the content of it if need by removing items, increase or decrease the quantity of any product in the Basket, can chose an existing delivery address or just add a new one in the system and then can choose the payment method before pressing the Place Order button.

Use Case 6

Scope

The scope of this use case is to allow the User to make changes in the Basket.

Actors

Customer, Display Screen, Database.

Use Case Diagram

A Use Case Diagram is shown below in Figure 7 of the interaction between the User and the database of the system.

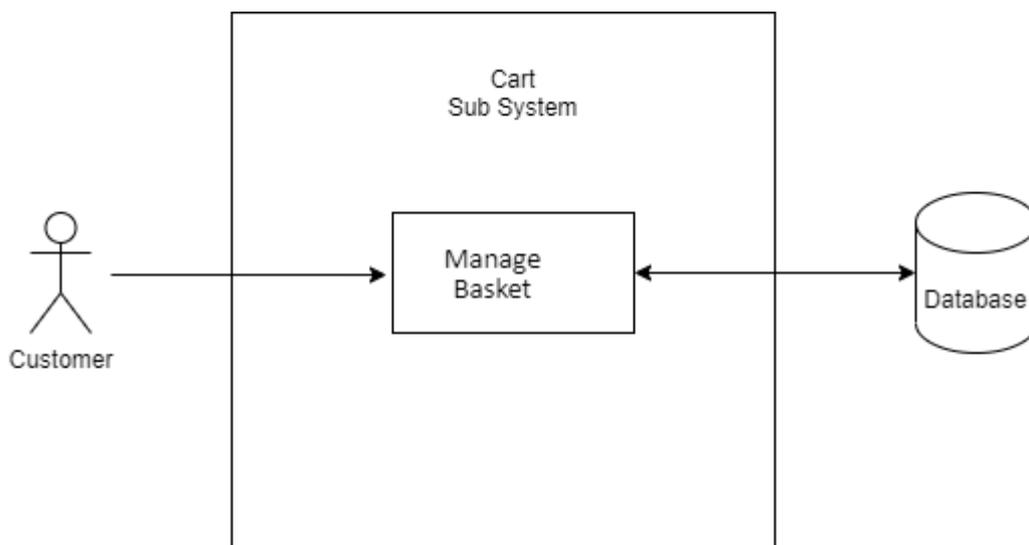


Figure 7

Flow Description

Precondition

The basket must have an item in it to allow to perform edit action.

Activation

Active upon selecting edit information's by a User in the My Cart page.

Main flow

1. The system identifies item into basket.
2. The User edits information (see A1).
3. The system updates Cart table in the database.
4. The system displays new saved basket.

Alternate flow

A1: User Not Logged In

1. The system detects an unregistered User.
2. The User is prompted to Login or to Register.
3. User completes Login or Registration process.
4. The Use case continues at position 2 of the main flow.

Exceptional flow

The User exits the system before they have edited the information on the basket content.

Termination

The system presents the next Confirm Order button to the User.

Post condition

The system goes into a wait state until User interaction is detected.

2.1.1.8. Requirement 7: Upload Products

Description & Priority

The Upload Product function is a very high priority for the system as it is the main way of showing to the Customers what the E-Commerce Web site has for sale. By not implementing this action into the system, the application is not meeting the minimum requirement of functionality. The Farmer or the Producer can upload their produces into the system only thru this functionality.

Use Case 7

Scope

The Upload Products use case scope is to allow stoking and supply of products in the Farmers Market E-Commerce Web site.

Actors

Farmer, Producer, Display Screen, Database.

Use Case Diagram

In the below Use Case Diagram from Figure 8 is shown the interaction in between the Farmers, Producers, and the system.

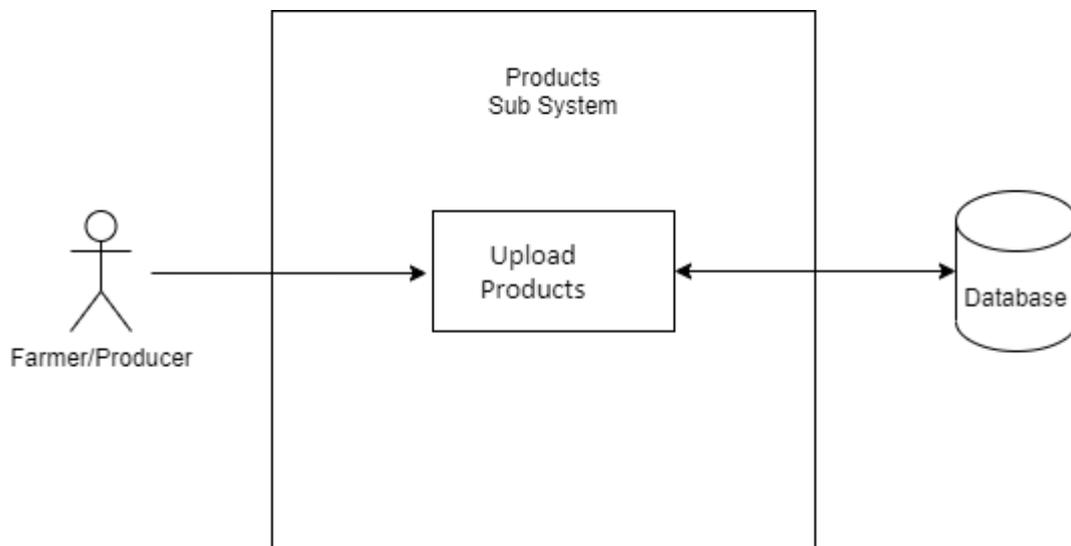


Figure 8

Flow Description

Precondition

The Farmers Market Admin Dashboard must be opened on a web browser on a device connected to the internet.

Activation

This use case starts when a Farmer or Producer navigates on the Products page in the Admin Dashboard.

Main flow

1. The system identifies the User.
2. The User uploads product (see A1).
3. The system updates Product table in the database.
4. The system displays new Product.

Alternate flow

A1: User Not Logged In

1. The system detects an unregistered User.
2. The User is prompted to Login or to Register.
3. User completes Login or Registration process.
4. The Use case continues at position 2 of the main flow.

Exceptional flow

The User exits the system before they have uploaded a product.

Termination

The system presents the Save Changes button next to the User.

Post condition

The system goes into a wait state until User interaction is detected.

[2.1.1.9. Requirement 8: View Orders](#)

Description & Priority

The View Orders page is a high priority functionality for the system as it allows the User being a Customer, a Farmer, a Producer or an Admin to view and take actions thru this page on appropriate order made on the platform.

Use Case 8

Scope

The scope of this use case is to give an option for a User to view and act on them as a response to orders made into the system.

Actors

Customer, Farmer, Producer, Admin, Display Screen, Database.

Use Case Diagram

The Use Case Diagram shown below in Figure 9 describes the View Orders action of a new Order made in the subsystem of the database.

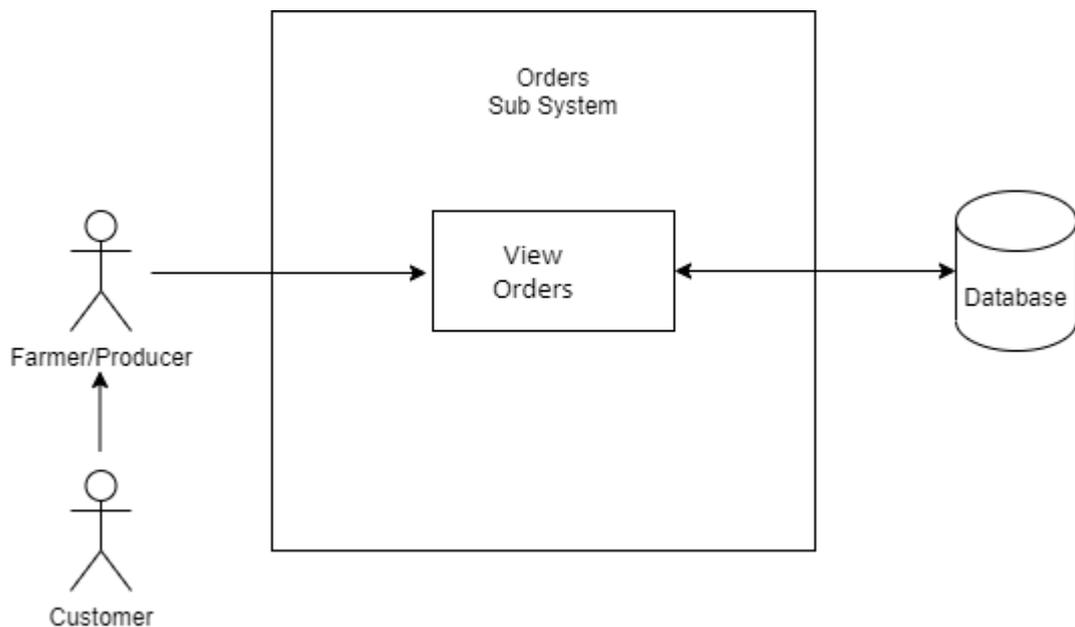


Figure 9

Flow Description

Precondition

The system is idle, and the User is already logged in to the system.

Activation

This use case starts when User clicks on the Orders page link.

Main flow

1. The User access the Orders page.
2. The User views Orders (see A1).
3. The User updates Order information.
4. The system displays new updated Order.

Alternate flow

A1: User Not Logged In

1. The system detects an unregistered User.
2. The User is prompted to Login or to Register.
3. User completes Login or Registration process.
4. The Use case continues at position 2 of the main flow.

Exceptional flow

The User exits the system before they have viewed an order of a product.

Termination

The system presents the Order status next to the User.

Post condition

The system goes into a wait state until User interaction is detected.

2.1.2. Data Requirements

For this project, I am using MongoDB Atlas, which is a document-oriented database service for modern platforms nowadays. By using a fully managed and intelligent automated database like MongoDB it helps in securing a high level of availability, scalability, and security compliance of services. Another two benefits of using Mongo BD is that speeds up the application development and reduces the complexity when it comes to deployment.

The database is hosted on a cluster in AWS, the cluster running version is 4.4.6 and is a General shared tier type called M0 Sandbox. The location of the database is in Ireland (EU-west-1) and it contains three replica set nodes. The maximum size of the database is 512MB and it can have up to 500 connections to it.

On the database, we can find stored seven tables that reflect the Farmers Market E-Commerce Platform needs to support the system. These collections of tables are Categories, Products, Carts, Orders, Users, Users Addresses and Pages. As the MongoDB is a schema-less NoSQL document database I had to use an Object Data Modelling (ODM) library named Mongoose that manages relationships between data, provides schema validation, and is used to translate between objects in code and the representation of those objects in MongoDB (FreeCodeCamp, 2018) with the help of Node.js too. For better visualisation of the relation and insights of how this process works please see the following diagram from below in Figure 10.

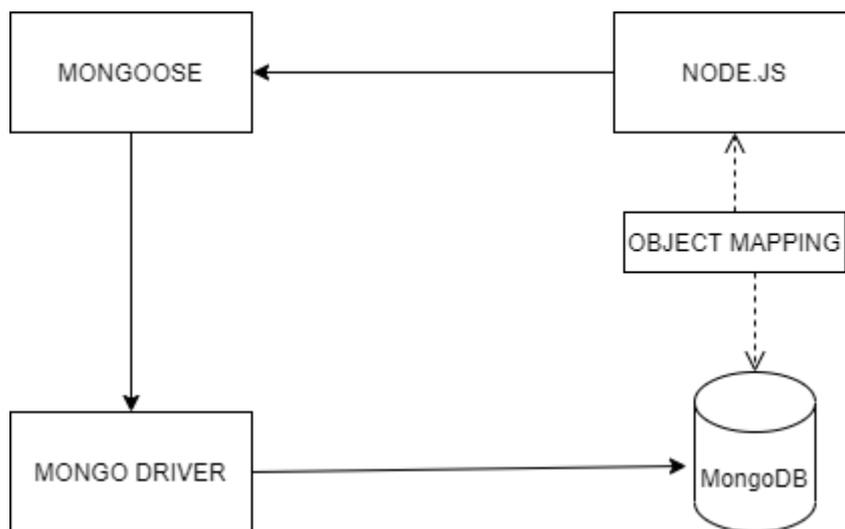


Figure 10

2.1.3. Non-Functional Requirements

In this section, I am enumerating all the non-functional requirements necessary for the Farmers Market to achieve during project implementation and life cycle.

Performance/Response Time Required

The system performance and response time are part of the main aspects of the overall functionality of the Farmers Market E-Commerce Web site. Because the platform has so

many interconnecting pieces deployed on many platforms, it is essential that all must work together seamlessly and give to the end-user the sensation that is interacting over the web with only one device. Network outage that can occur in between the server's cloud lead to a major financial disaster for the Farmers Market business. For this to be prevented I have the database reproduced on three separate nodes in AWS just to make sure that I will minimise any potential risk of data loss when a user is interacting with the system.

Availability Requirement

The Farmers Market E-Commerce Web site, the Admin Dashboards Web site, the database server, and the back-end server must be online 24/7 to provide high services to its users. The application is up and running on a very reliable cloud provider but as it is on a free tyre plan some lag in retrieving some information from the back-end server may occur, but is not a major issue as a page refers does the trick to reload the page.

Accessibility Requirement

The Web sites of the Admin Dashboard and the E-Commerce Web site that is the one exposed to the customers via the web are simple, intuitive, and easy to use. On these interfaces, we can find commune elements used over the E-Commerce industry, whit an emphasis on recognition rather than recall. When moving from a page to another the system is consistent and it presents to the user a nice feeling of continuity.

Security Requirement

Security is the most important part of the project, without good security practice in mind we can barely build an application these days as it will be hacked in a matter of short time and at the minimum can expose all the sensitive data store on our database about Users. Knowing that good security comes in layers and to prevent any wrongdoing that could happen I have all the data stored on the most secure cloud provider where they are employing the best secure environment for my data. The cloud provider promises that their platform is designed to help me build secure, high-performing, resilient because they have the most efficient infrastructure (aws.com, 2021).

Another layer of security employed in my application design is that I will be using jwt tokens to secure the communication in between the interacting services in my application. As they are an industry-standard RFC 7519, they can be encrypted to provide secrecy between parties

(jwt.io, 2021) and I am limiting the expiration time for each session at only one hour on connection.

Maintainability Requirement

As a software developer, I am aware of pitfalls that a not well written and a full of bugs code can fail into. A well-maintained code deployed in an Agile methodology framework is the way to go, and I am using this method to keep on the right track with my project build. Another factor that I consider is the System structure, I am separating the application into different modules that are specific for their functionality, one for Admins, Farmers and Producers and one for Customers. All the code is auto-updated in the system by Heroku on every new release made in the GitHub account where the project is stored.

Reusability Requirement

To reuse the code is a must, use and reuse, this helped me to save on time resources in building my project. When it comes to fixing an error, I only have to change the implementation in one place and that it will reflect in all the places that are currently using it. This method leads to a better design by being an intuitive and more user-friendly application.

2.1.4. User Requirements

In this section I am giving recommendations on user requirements when a User wants to use the E-Commerce platform, the following requirement must be met at a minimum:

PC, Tablet or Smart Phone - Any of this device is needed to be able to access the Web site.

Internet Access - This requirement is needed to allow access to the Web site.

Web Browser - This requirement is needed to allow access to view, buy, update information on to the system platform, any Web Brower will do.

2.1.5. Environmental Requirements

In this section, I am enumerating the environmental requirements that have been essential when I developed the application.

Text Editor - Visual Studio Code, is the text editor used to code the application.

Laptop – Lenovo laptop was used to run Visual Studio Code on a Windows 10 OS.

Web Browser – Google Chrome was used to test the code on and to do research.

Internet Access - This is required to access the world wide web with the web browser.

2.1.6. Usability Requirements

To allow the transaction to take place on the Farmers Market Web site the user needs to be logged in and hold an account into the system. The system prompts the user to log in or to create an account whit us as soon it detects the Place Order button is clicked on a product that is added into the Cart, this happens only if the user is not logged in already.

The view and browsing of the Farmers Market E-Commerce Web site do not require holding an account.

2.2. Design & Architecture

The Farmers Market Platform is built on MERN Full-Stack Framework which is an open-source solution that is providing a strong starting point when it comes to using MongoDB, Express, React.JS and Node.JS (MERN) technologies for any web-based applications (Muhd Mohaiminul Islam, 2021).

The Farmers Market E-Commerce Web site, the Famers Market Admin Dashboard Web site (the client-side/front-end) and the back-end server (server-side) are built and deployed individually from each other servers hosted on Heroku Cloud Provider. In Figure 11 from below is the Architecture Diagram of the system topology.

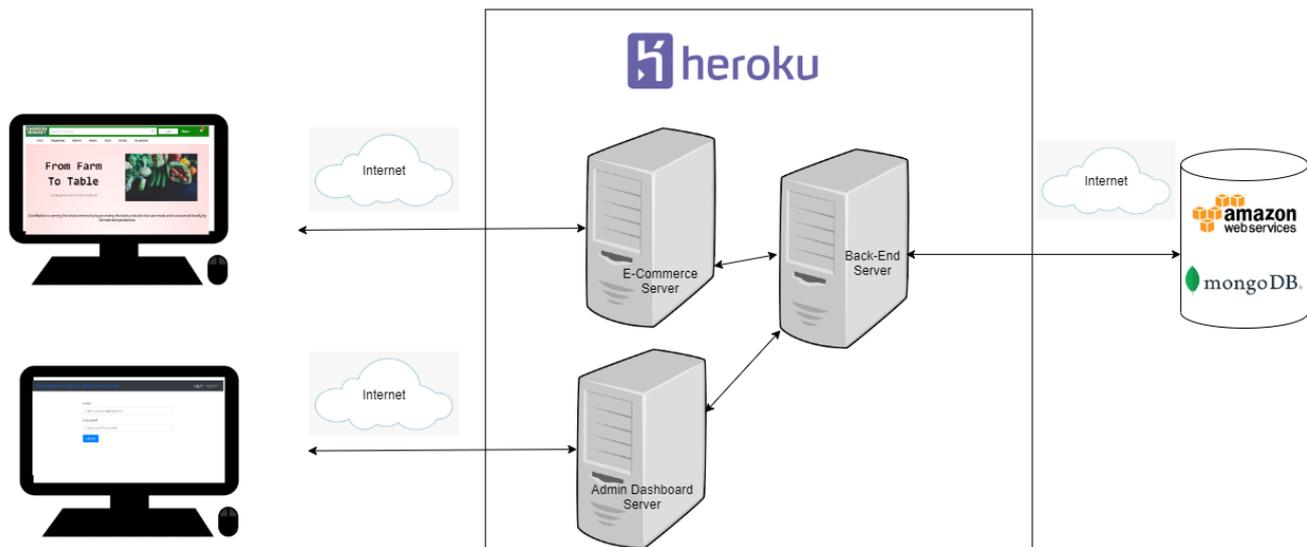


Figure 11

React.JS

For Users to be able to interact with the E-Commerce Platform, the interfaces are built using React.JS library. This method enables the creation of User Interface components of web pages in the application that are residing in the front-end in a browser.

Node.JS

All the interactions on the E-Commerce Platform that are making data changes are directed to the Node.JS that is a JavaScript running environment on a back-end server.

Express.JS

The two front-end servers of the application connect through Express.JS, which is a framework layer to provide back-end logic residing above Node.JS, to the back-end server for serving them with requested data.

MongoDB

The back-end server is connected and communicating with the database system which is hosted on Amazon Web Services Cloud via MongoDB. This data is grabbed by the Node.JS based Express server when is requested by the front-end server and then presented to the end-user in a browser.

In the below illustration from Figure 12 is describing the above MERN stack framework explanation.

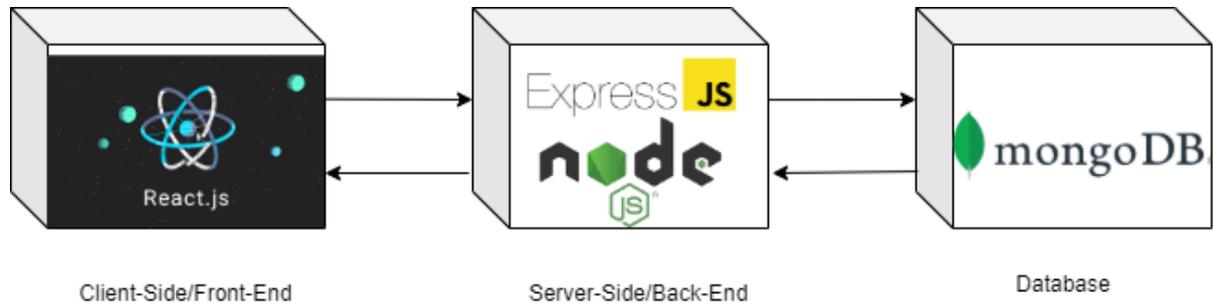


Figure 12

2.3. Graphical User Interface (GUI)

Farmers Market E-Commerce Web site GUI Layout

The Landing Page or the Home Page of the Farmers Market E-Commerce Web site that is used by the customers to view and purchase goods on it is shown in the next few screenshots below after a successful logged in into the system. As we can see the customer name is displayed in the website header portion along with his stored products for purchase in the Cart icon.

From Farm To Table

Locally grown and locally produced



Our Market is serving the local community by providing the best products that are made and outsourced locally by farmers and producers.

Fresh Produce By Categories



Vegetables



Fruits



Bakery

Download The App

Download the Application for Android and IOS devices.



Useful Links

- Coupons
- Click and Collect
- Return Policy

Follow us

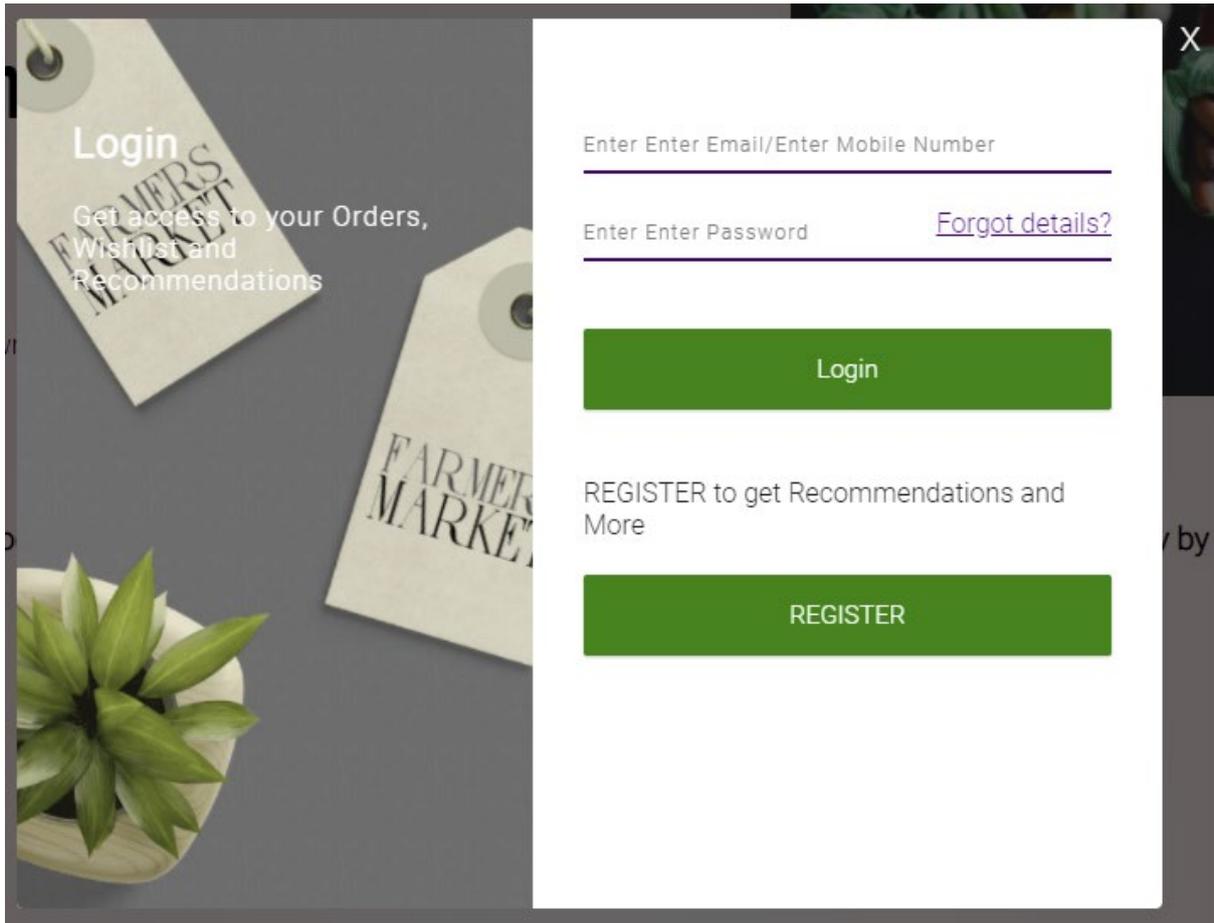
- LinkedIn
- GitHub
- Facebook

For a Customer to be able to buy a product they need first to register on the platform using the registration form that is shown in the following screenshot.

The screenshot displays a registration form overlaid on a background image. The background image features a 'Login' section with the text: 'Login', 'Get access to your Orders, Wishlist and Recommendations', and 'FARMER'S MARKET' on a tag. The registration form itself has the following elements:

- Input field: Enter First Name
- Input field: Enter Last Name
- Input field: Enter Enter Email/Enter Mobile Number
- Input field: Enter Enter Password
- Green button: Register
- Text: REGISTER to get Recommendations and More
- Green button: LOGIN

Or if they are already registered into the system they need to Login by using the form shown in the below screenshot.



In the next screengrab from below is presented one possible way of how to view the Products Page that belong to a category. In this example, I used the Berries subcategory of the Fruits main category by hover the mouse hover over it and clicking it.

FARMERS MARKET Search for products Daniel Smith More 3

Fruit Vegetables Bakery Meats Dairy Drinks Essentials

Citrus Berries Melons Apples, Pears & Plums



Raspberries
€1.99



Blueberries
€5.01



Strawberries
€4.99

Download The App

Download the Application for Android and IOS devices.






Useful Links

Coupons
Click and Collect
Return Policy

Follow us

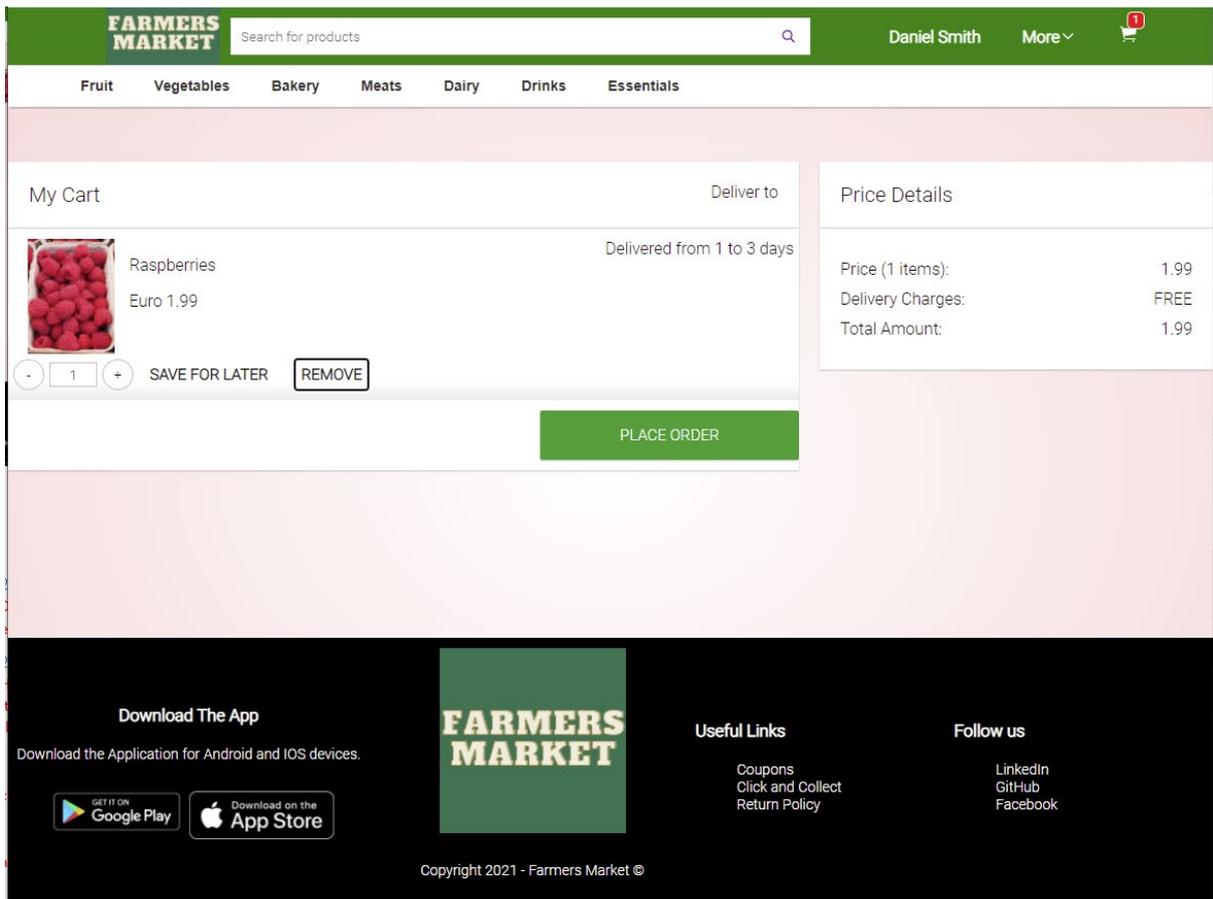
LinkedIn
GitHub
Facebook

Copyright 2021 - Farmers Market ©

Then the selected Product is rendered on a new page, the Raspberries are shown in the next screenshot with the afferent details that can be found on a Product Details Page of most E-Commerce Web sites out there.



After the customer selects the ADD TO CART option from the above example, the product is placed in the Cart shown in the below screenshot of the Cart Page. Here we can see that the quantity can be modified, the product can be removed from the cart, saved for later or purchased by clicking the Place Order button. Here the Price details are shown too and if an additional product will be added to the basket the price breakdown will be reflecting this action.



When the Place Order button is selected the product is placed at the checkout stage and here the final adjustments can be done by the customer in the next steps.

The Checkout Process is done in four steps, in the first step is Logging in to proceed if not already logged in, the second step is to choose a delivery address if any is saved in the system from previous transactions or he can add a new one in necessary as is shown in the below screenshot.

FARMERS MARKET Search for products Daniel Smith More

Fruit Vegetables Bakery Meats Dairy Drinks Essentials

1 LOGIN
Daniel Smith - christian2smit@yahoo.ca

2 DELIVERY ADDRESS

Daniel Smith **HOME** 9876543210
140 Main Road
Dublin - Leinster - D11 0KM

Daniel Costel **WORK**
4 Grace Park Heights
Dublin - -

+ ADD NEW ADDRESS

Enter Full Name Enter 10-digit mobile number

Enter Eircode Enter County

Enter Full Address

Enter City/Town Enter Region/Province

Enter Landmark (Optional) Enter Alternate Phone (Optional)

Address Type
 Home Work

SAVE AND DELIVER HERE

Price Details

Price (1 items):	1.99
Delivery Charges:	FREE
Total Amount:	1.99

At step three of the Checkout process, the Order Summary is presented to the customer and if wanted again the product quantity can be increase, decrease, or removed as necessary. The next screengrab has presented the case when the customer chooses to increase the quantity up to three Raspberries baskets, the price changes too in the Price Details box to give the new total amount to be paid.

1 LOGIN Daniel Smith - christian2smit@yahoo.ca	Price Details Price (3 items): 5.97 Delivery Charges: FREE Total Amount: 5.97
2 DELIVERY ADDRESS Daniel Smith - 140 Main Road - D11 0KM	
3 ORDER SUMMARY	
 Raspberries Delivered from 1 to 3 days Euro 1.99 <input type="button" value="-"/> <input type="text" value="3"/> <input type="button" value="+"/> <input type="button" value="SAVE FOR LATER"/> <input type="button" value="REMOVE"/>	
Order confirmation email will be sent to christian2smit@yahoo.ca <input type="button" value="CONTINUE"/>	
4 PAYMENT OPTIONS	

After the Continue button is pressed in step three by the customer the fourth step is displayed on the same form page with the payment option available, in our case the cash on delivery is present.

4 PAYMENT OPTIONS
<input checked="" type="radio"/> Cash only on delivery <input type="button" value="CONFIRM ORDER"/>

After the Confirm Order button is pressed the next step is to display the Order Details on a new page as is shown in the next screenshot from below.

On the new page, we can see that the product count is reduced to zero in the cart icon from the Page Header as the cart got cleared at the Checkout stage, then an overview of the order just made is presented to the customer showing the delivery address, an invoice with few options is present too along with a status that reflects the progress done so far on order by the fulfilment centre team.

FARMERS MARKET Search for products Daniel Smith More

Fruit Vegetables Bakery Meats Dairy Drinks Essentials

Delivery Address: Daniel Smith, 140 Main Road, Phone number 9876543210

More Actions: Download Invoice

Raspberries €1,99

2021-5-15

Ordered — Packed — Shipped — Delivered

A historical orders view is possible for all orders made on the system thru the Orders Page, this option can be accessed via the Orders button available in the drop-down menu situated on the top of the page Header, as is shown in the below screenshot.

FARMERS MARKET Search for products Daniel Smith More

Fruit Vegetables Bakery Meats Dairy Drinks Essentials

Home > My Account > My Orders >

Milk €0,99 pending

Carrots €0,99 pending

T-shirt €1,99 pending

Water €1,99 pending

T-shirt €1,99 pending

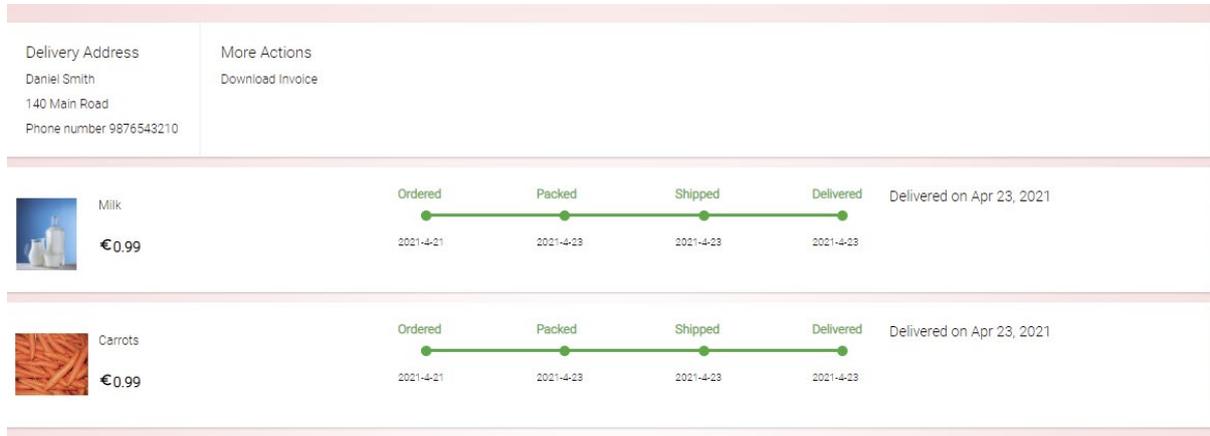
Raspberries €1,99 pending

Raspberries €1,99 pending

<https://farmers-market-e-commerce.herokuapp.com/account/orders> €1,00 pending

My Profile
Farmers Market Plus Zone
Orders
Wishlist
Rewards
Gift Cards
Logout

After the customer selects the desire order to view, and Order Page is opened, and the order is displayed. For this example, the order, in this case, has the status Delivered with the date for each item in the order.



Each status from above is updated via the Admin Dashboard Web site by the fulfilment centre team and then the transaction gets auto-updated in the customer transaction.

Admin Dashboard For Farmers Market E-Commerce App GUI Layout

The second Web site built for the Farmers Market E-Commerce Platform is an administrator-oriented application that is facilitating the necessary support functionality of the business. The Admin Dashboard is built with minimum graphic representations, this is meant to help the application load and react as fast as possible considering the fact that is deployed on a free tyre server where memory and CPU power are at a minimum.

For a user to be able to access the Admin Dashboard he must be logged in to the application, this is possible by pressing the Login button situated in the Header element. In the below screenshot is shown the Login form for the application where is taking as input the user email address and password to successfully Login after the Submit button is activated.

Email

Password

But if the user is not already registered into the system, he can do so by pressing the Register button found at the top of the page in the Header. Here as we can see in the next screenshot from below, the same form is used again like at the Log In stage but this time with two additional line fields where the user is asked to input its first and second name. The registration is completed as the required details are inserted by the user and the Submit button is triggered.

First Name

Last Name

Email

Password

After all the necessary credential is usefully are submitted, the user now can access the main part of the application. The screenshot below is shown what is displayed in the Category Page, here the user can view all the existent produces categories and subcategories at this time. As an option, the user can add a new category by pressing the Add button.

HOME

PAGE

CATEGORY

PRODUCTS

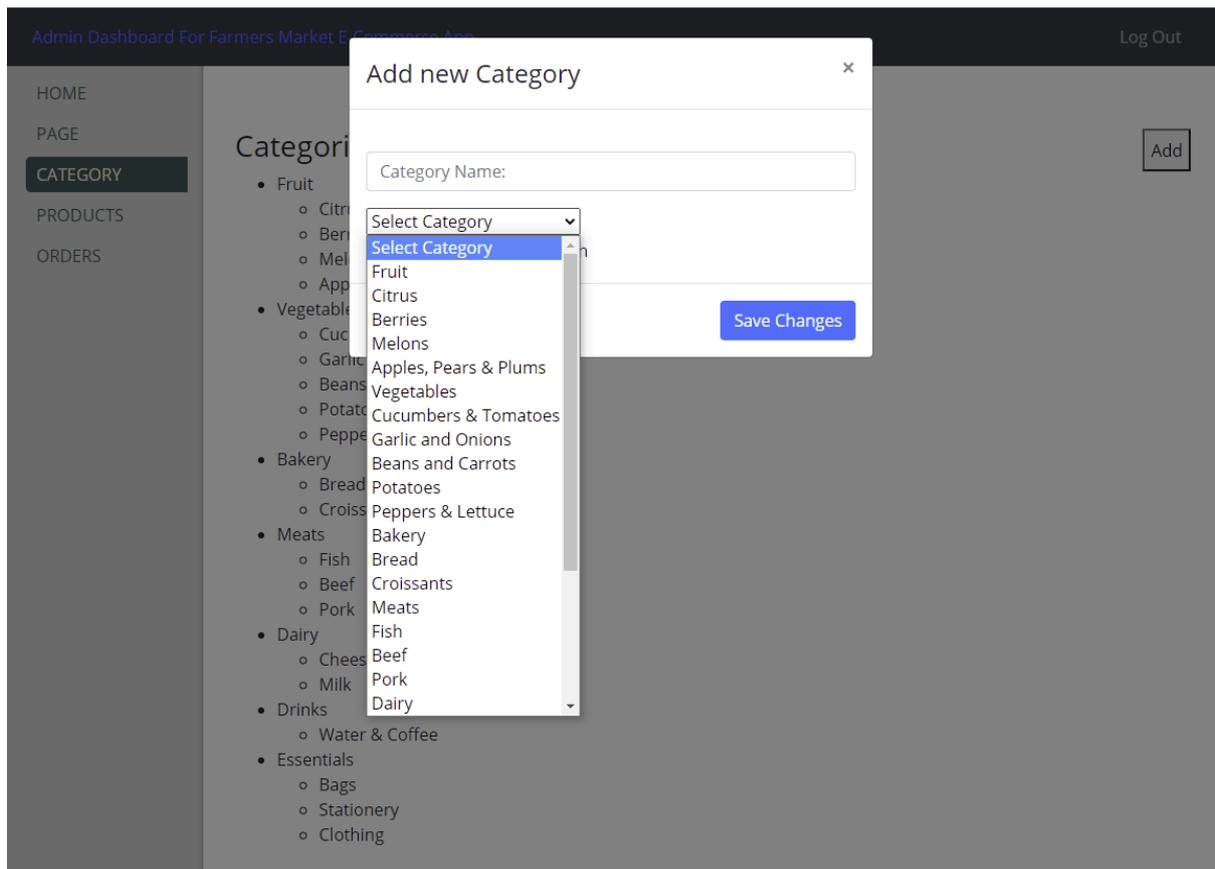
ORDERS

Categories

Add

- Fruit
 - Citrus
 - Berries
 - Melons
 - Apples, Pears & Plums
- Vegetables
 - Cucumbers & Tomatoes
 - Garlic and Onions
 - Beans and Carrots
 - Potatoes
 - Peppers & Lettuce
- Bakery
 - Bread
 - Croissants
- Meats
 - Fish
 - Beef
 - Pork
- Dairy
 - Cheese
 - Milk
- Drinks
 - Water & Coffee
- Essentials
 - Bags
 - Stationery
 - Clothing

After the Add button is pressed by the user for the intent to insert a new category into the system, a form is opened on the page to facilitate this process. Here is it shown in the next screengrab, the user must insert the new subcategory or category's name along with some other necessary details and then press the Save Changes button. The new changes that are done by this action are reflected in the Farmers Market E-Commerce Web site too.



Another essential element of this application is that it allowing the user to view all the product currently available in the system for sale. To view the Products Page the user must navigate and press the Products tab in the left sidebar on the page. On the Products Page are displayed, as we can see from the below screenshot, all the product, along with few actionable buttons that will allow the user to add a new product, view specific product information or delete it.

Admin Dashboard For Farmers Market E-Commerce App Log Out

HOME

PAGE

CATEGORY

PRODUCTS

ORDERS

Products Add

#	Name	Price	Quantity	Category	
1	Green Apples	1.99	180	Apples, Pears & Plums	Info Delete
1	Red Apples	2.99	180	Apples, Pears & Plums	Info Delete
1	Pears	5.01	180	Apples, Pears & Plums	Info Delete
1	Plums	10.01	200	Apples, Pears & Plums	Info Delete
1	Raspberries	1.99	180	Berries	Info Delete
1	Blueberries	5.01	200	Berries	Info Delete
1	Strawberries	4.99	300	Berries	Info Delete
1	Watermelons	2	60	Melons	Info Delete
1	Potatoes	1.99	500	Potatoes	Info Delete
1	Garlic	2.99	300	Garlic and Onions	Info Delete
1	Onions	1.99	300	Garlic and Onions	Info Delete
1	Green Beans	1.99	200	Beans and Carrots	Info Delete

If the Add button is triggered the user has the option to insert into the system a new product, as we can see in the next screenshot from below. The user must input into the product form all the required information and then hit the Save Changes button to take effect.

The new changes that are done by this action are updated into the Farmers Market E-Commerce Web site too and displayed to the customers.

Admin Dashboard For Farmers Market E-Commerce App Log Out

HOME

PAGE

CATEGORY

PRODUCTS

ORDERS

Products Add

#	Name	Price	Quantity	Category	
1	Green Apples	1.99	180	Apples, Pears & Plums	Info Delete
1	Red Apples	2.99	180	Apples, Pears & Plums	Info Delete
1	Pears	5.01	180	Apples, Pears & Plums	Info Delete
1	Plums	10.01	200	Apples, Pears & Plums	Info Delete
1	Raspberries	1.99	180	Berries	Info Delete
1	Blueberries	5.01	200	Berries	Info Delete
1	Strawberries	4.99	300	Berries	Info Delete
1	Watermelons	2	60	Melons	Info Delete
1	Potatoes	1.99	500	Potatoes	Info Delete

×

Name

Quantity

Price

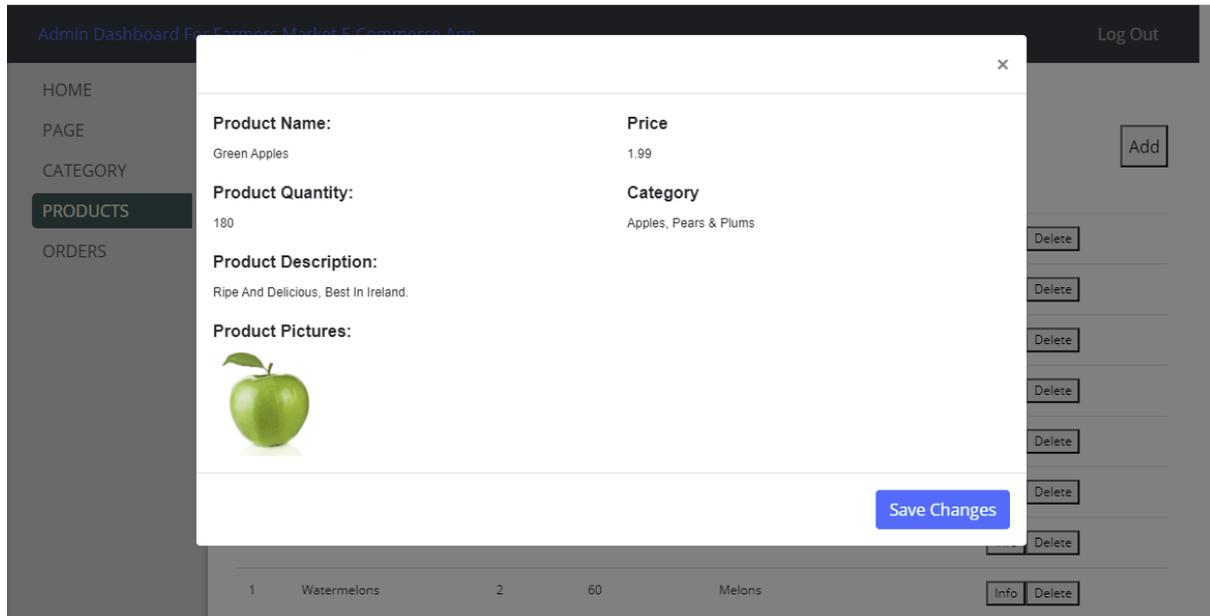
Description

Select Category ▼

Choose File No file chosen

Save Changes

After the creation of the new product, the user can review the newly created item by pressing the Info button that is in line with the product box. The next screenshot form below it is shown the output of this action.



The orders made by the customer on the Farmers Market E-Commerce Web site, and then received at the fulfilment centre, can be actioned in the Orders Page that can be accessed by navigating to the left-hand sidebar and pressing its corresponding tab.

On this page, all the orders are displayed in order of arriving into the system, where the user can see the order details, a unique order reference number that is given to each one and a process status that is shown in a colour-coded line to reflect the progress done so far by the team on the order.

To action an order the user has the option to select the order status by selecting the corresponding stage that the order is at, packed, shipped or delivered and then hit the Confirm button too.

Any new change done by this action are updated into the Farmers Market E-Commerce Web site too and displayed on the customers' Orders Page.

All the described user interaction in the above example are shown in the next screenshot below.

HOME

PAGE

CATEGORY

PRODUCTS

ORDERS

60870aeb813d4d27dcd6e88d

Items Stakes	Total Price 13.98	Payment Type Cod	Payment Status Pending
-----------------	----------------------	---------------------	---------------------------

ordered packed shipped delivered

2021-4-26 2021-4-30

608bec31daefce31ccbeb248

Items Sausages	Total Price 1.99	Payment Type Cod	Payment Status Pending
-------------------	---------------------	---------------------	---------------------------

ordered packed shipped delivered

2021-4-30

2.4. Testing

Site Accessibility Testing

The main goal of building an E-Commerce Web site where a user can trade local goods online is to provide information and to help the user in achieving his task.

While it may be stylish to use subtle, low contrast colours in our designs, we should avoid going too far and creating an experience that's difficult to read. It's a best practice to look over the colours we've used in our site design and see if they meet recommended contrast minimums. One of the tools that we have used for testing on how this project beehive is OATMEAL Colour Contrast which is an Accessibility Testing tool for the Web and is developed and used by eBay (2021), this tool ensures consistent accessibility testing and

makes detailed accessibility testing methods available to everyone, not just accessibility experts.

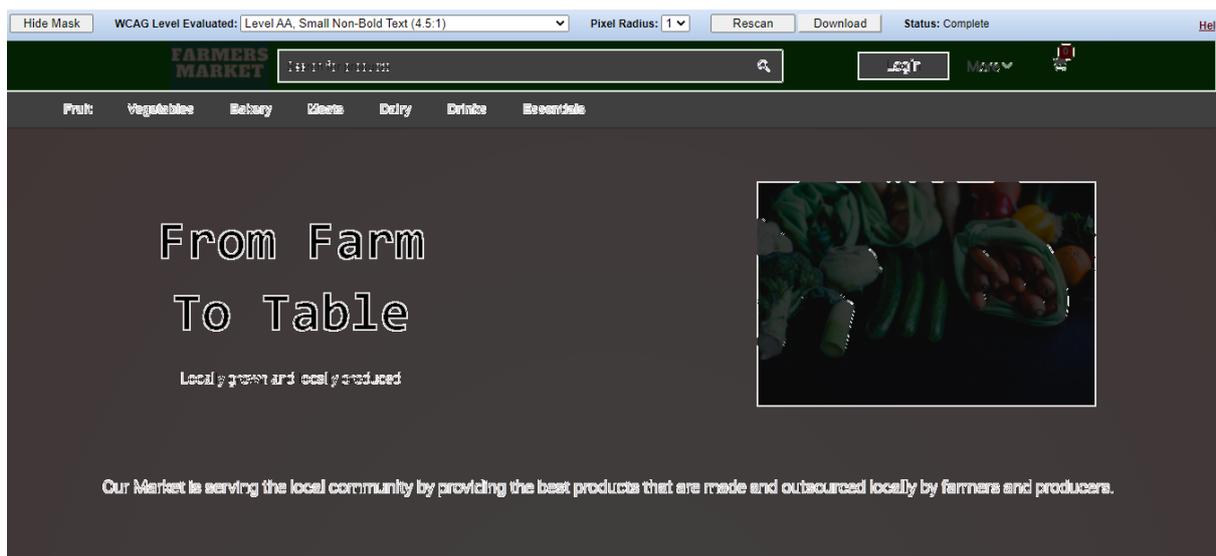
The Colour contrast testing method is employed to help the end-users with moderately to low vision to read text on the Web site without the need for contrast-enhancing assistive technology. This is accomplished by meeting a minimum contrast ratio between foreground text and the background colour behind it. The contrast ratio is usually presented in a few ways.

For example, here are three ways of communicating the same contrast ratio:

- 4.5 to 1
- 4.5:1
- 4.5

The minimum contrast ratio for most text will be 4.5 to 1, and when developing for mobile platforms, strive to meet a 7:1 ratio.

After installing the WCAG tool in the Chrome web browser and using the method described above to examine the Landing Page I can see that the contrast between the foreground and background is clear and readable.



Usability Design Testing

To get a good result of a Usability Design Test sessions I would need a minimum of five participants to participate in this stage, but as we all know of the recent restrictions imposed as a result of the current pandemic, the testing it will be done only by me.

as they are the most valuable resource of feedback for our project and because they have already participated in the requirement gathering phase of the project and we built the website on their requirements, they should be familiar with the website structure and design.

By conducting the testing session morally with five participants I can identify problems in my Web sites design, it may possibly lead to uncovering opportunities to improve them and I may be learning more about our target users behaviour and preferences as 76% of the customers would switch to a competitor if they had just one bad experience with a brand (Playbookux, 2021).

As part of the Usability Design Testing session, I employed five testing techniques to measure my interaction with the website as a participant in the testing session. These five techniques are:

1. Five Second Test,
2. Trunk Test,
3. Heuristic Evaluation,
4. Think Aloud,
5. Click Test.

Testing Introduction

I have set up the environment and I made myself as comfortable as possible and looked over the task to be performed and the goals of the test.

Here I will be measuring the system performance too by how effective, efficient and satisfactory is by measuring the error rate, the time spent to complete a specific task and steps to do so, and see how happy made me as a participant to complete all these proposed tasks.

All the task chosen for the testing session is relevant for each test.

I will consider starting each task from the home page for every test – thus limiting the number of possible wrongs turns I might make on subsequent tasks.

Five Second Testing

I have started the testing session with the Five Second Test which is my first technique and tasked myself as a participant to observe the Home Page of the E-Commerce Website for only five seconds and then I switched to another window, then I asked myself “What is the site for?”, by doing this action it will allow me to identify as a developer if the main page purpose is understood and accurate.

After conducting this test, I found out that the Design of the Main Page of my Website is clear and the content of it was recalled back almost without any effort. I also found out that by not using too much descriptive text on the main page and by using big pictures with call-back actions buttons it helps identify immediately that the site is an E-Commerce Web site where you can buy your local produces. An overall the result of this test is showing me to see whether key visuals or call-to-action buttons have the right impact on the user that it will make him stay or leave the site.

Trunk Testing

For the second test technique employed in my testing session, I used Trunk Test and is used in identifying the possible site navigation problems. I looked over the Orders Page of the Web site and I asked myself as a participant to locate the page I am on, the major sections, it is any sign of traceback as quickly as possible to the home page, the questions to respond for this test are:

- What is the site name?
- What page is this?
- What are the major sections?
- Where am I on this site?

The main scope of the above questions is to find out how good the website design is, by seeing how quick users can determine where on-site are, based on content and visual clues like breadcrumbs and navigation.

After timing each response, I found out that the website design is easy to navigate due to the menu bar in the Header at the top of the page, the name of the site was easy to spot too as it is placed at the very top of the site in the logo, the major section of the site was identified successfully do to the menu bar. In general, I am happy with this result as it shows me that the information’s on the Web site is well structured and presented to the end-users.

Heuristic Evaluation Testing

Heuristic Evaluation is my third method of testing used in my session, which is a technique for judging a product's compliance against recognized industry standards. To perform a heuristic evaluation is difficult for a single individual to do because one person will never be able to find all the usability problems in an interface but using five participants as a normal procedure in this test it is possible to improve the effectiveness of the method significantly (Jakob Nielsen, 1994).

I applied Jakob Nielsen technique of finding and explain the problems encountered by me as a participant to the test when I was trying to Add to Cart a product from the Web site collection. I will use the next ten rules to iterate my metrics as a test facilitator:

1. Visibility of system status.
2. Match between system and the real world.
3. User control and freedom.
4. Consistency and standards.
5. Error prevention.
6. Recognition rather than recall.
7. Flexibility and efficiency of use.
8. Aesthetic and minimalist design.
9. Help users recognize, diagnose, and recover from errors.
10. Help and documentation.

The findings of this test technique are revealing that is fairly easy for the user/participant (me) of the test to add to cart one of the products. After the participant (me) leaves the Home page and navigates on the desired Product Page, the full product details are displayed on a new page and a call-to-action button is presented, to me as a participant, under each Product picture. The Add to Cart button presented here and is easily recognisable as it is an industry-standard.

Think Aloud Testing

One of the most valuable usability testing technique that I can employ to find more insights into my project during usability design is the Think Aloud Testing method.

I believe that one of the main benefits of conducting this test is helping me as a designer to understand how the user approaches the Web site and what considerations the user keeps in mind when using it. If a different path is taken, by the participant to accomplish his task on my website, other than the one planed and expected by me as a designer, then I should rethink and redressed the approach to the task's goal.

I decided to use this technique as my fourth method of testing, and I tasked myself as a participant to vocalize my process of interaction with my website while using it and to give the findings for my request of removing the item from the Cart content that took place at the previous test.

I can conclude as a designer that at the end of this particular test I as a participant to it I am happy with the experience of navigating and completing the task in a reasonable time and error-free.

Click Testing

The last testing technique used to collect any evidence on the performance of my design website and to record the user's reaction to it is the Click Testing method. This technique examines what participants click on the first click, to complete their intended task and which can aid in evaluating the intuitiveness of the site's buttons, menus, links and such, in context with my site's design.

Click Testing allow me to evaluate the effectiveness of the linking structure of my website, including the navigation, to see how and if users/participant to the test get around the website and complete their intended task.

The task was to increase the quantity of a product that is on a Cart page.

For this test, I had another successful completion with a very high score. The task was easy to accomplish as on this page is a button to allow the quantity changes to be made on every product that is in the Cart.

Debrief

Debriefing should be conducted at the end of each test session and then go thru and analyse the actions performed by the participant. By conducting debriefing, it provided us with additional insight on why that participant performed such actions and further insight on why some problems occurred.

To be able to measure the participant's satisfaction with regards to the website design an exit survey must be conducted too. Using the Systems Usability Scale (SUS) survey method will not tell us what specific problems our prototype faces, but it will give us a red or green light to know how badly the usability design needs to improve.

The SUS survey is a 10-item questionnaire with 5 response options as they are displayed below.

Items:

1. I think that I would like to use this system frequently.
2. I found the system unnecessarily complex.
3. I thought the system was easy to use.
4. I think that I would need the support of a technical person to be able to use this system.
5. I found the various functions in this system were well integrated.
6. I thought there was too much inconsistency in this system.
7. I would imagine that most people would learn to use this system very quickly.
8. I found the system very cumbersome to use.
9. I felt very confident using the system.
10. I needed to learn a lot of things before I could get going with this system.

The format used to represent and measures the responses on a scale from 1 to 5 is:

Strongly Disagree				Strongly Agree
1	2	3	4	5
○	○	○	○	○

By calculating the SUS results after each question posed to participants, we obtain a final score, where a score higher than 68% means that I did a good job.

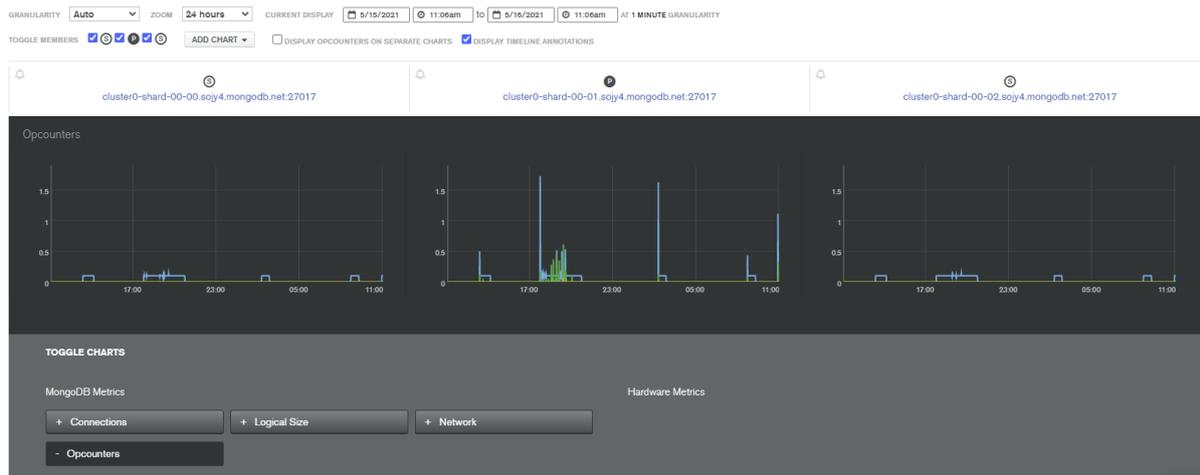
2.5. Evaluation

The project evaluation was conducted by monitoring the online metrics that are built-in tools and they come along with the MongoDB environment of the database deployment.

The database dashboard can output the average rates in time intervals in a graph that are colour-coded by lines that represent:

- The number of commands performed per second over the selected sample period.
- The number of queries performed per second over the selected sample period.
- The number of updates performed per second over the selected sample period.
- The number of deletes performed per second over the selected sample period.
- The number of inserts performed per second over the selected sample period.
- The number of getMores performed per second on any cursor over the selected sample period, but on a primary, this number can be high even if the query count is low as the secondaries "getMore" from the primary often as part of replication.

The application is only up for one week already at the time of this report and a view of what is reported on the tooling dashboard is shown in the below screenshots whit an afferent graph for each node in the cluster.



The next screenshot is shown the number of currently active connections to the database server where a stack is allocated per connection.



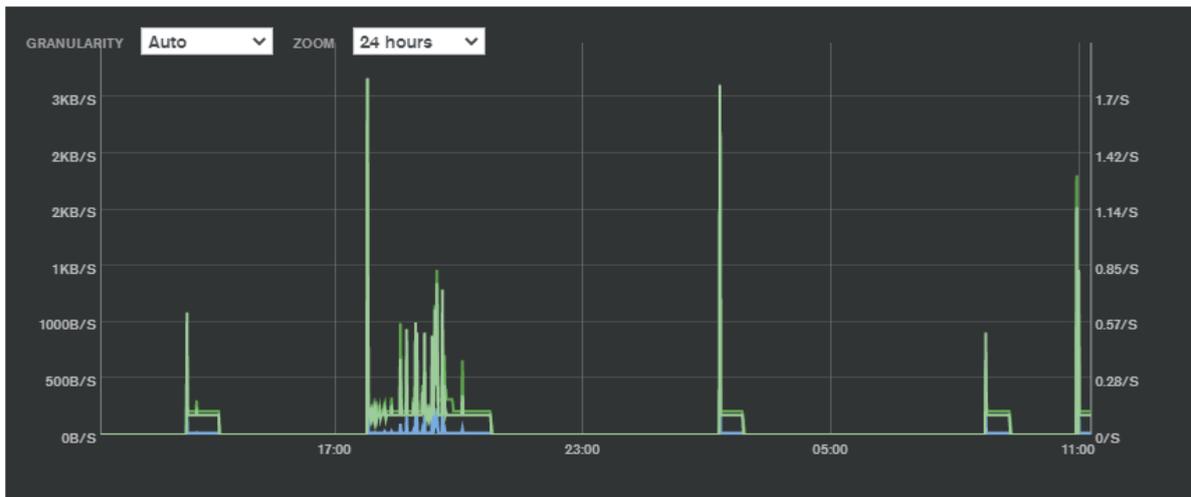
At the network level, the system reports the average rate of physical (after any wire compression) bytes per second sent to this database server over the last 24 hours sample period into the system represented in the below graph by a blue coloured line.

Another coloured line (light green) in the graph from below is representing the average rate of physical (after any wire compression) bytes out per second sent from this database server per second over the 24-hour selected sample period.

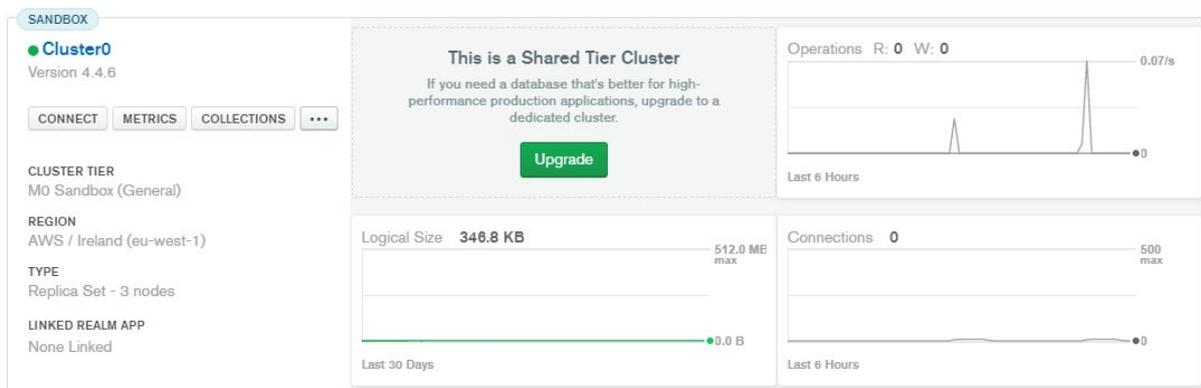
To measure the average number of requests rate sent to this database server per second over the 24-hour selected sample period is represented in the below graph by a dark green coloured line.

An interesting insight obtain from this query is that a connection to the database was made around 3 o'clock in the nighttime, is interesting to see because the only person who has access to this project is only me and I was not up to do work or to access the resources at that time.

Network



An overview of the System Status is shown in the below screenshot capture from the Metrics tab located in the MongoDB dashboard.



System Status: All Good

The system only can be monitored now only from the database dashboard at it gives granulated information as needed, but I believe it can be improved as soon an upgrade is done on the price plan.

At this moment Heroku Cloud provider does not provide free tyres plans for any metrics monitoring tools on any of the three servers deployed on their platform, but as soon an

upgrade in the price plan is done the tools are available as we can see in the below screenshot.



Metrics are not currently available for this app

Metrics are only available for apps running Hobby or Professional dynos.
[Learn about metrics in the Dev Center](#)

[Add credit card to install metrics](#)

Some useful information I managed to obtain from the Heroku App information environment regarding each application deployed on their Cloud Platform used in this project.

In the next three screenshots, I will show the main overview of the information obtain from Heroku with two important features that include the total amount of space available and the framework used to build the application.

App Name

e-commerce-back-end-server

Region	 Europe
Stack	heroku-20
Framework	 Node.js
Slug size	436.1 MiB of 500 MiB

App Name

farmers-market-e-commerce

Region	 Europe
Stack	heroku-20
Framework	 React.js (create-react-app) multi
Slug size	169.5 MiB of 500 MiB

App Name

farmers-market-admin-app

Region	 Europe
Stack	heroku-20
Framework	 React.js (create-react-app) multi
Slug size	79.4 MiB of 500 MiB

3.0 Conclusions

One of the most important opportunities for Farmers Market E-Commerce Platform is that at this moment in time is no other company offering these services online, this is a good moment to launch a new platform online because more and more people are ordering online. The market value of this project can increase overnight.

I can say that the limitation of this Project are the price plans that the services are hosted on, with a bit of investment on few better plans for database and on Heroku servers, the performances will increase, by the speed of the websites reaction and the size of them too.

The application development and deployment were very enjoyable for me as all the knowledge gain over the course of the four years spent in the college managed to give me all the necessary information needed to successfully create an E-Commerce Application Platform that is in trend with today's technologies used on the market. I did not have any

knowledge of coding at all before joining the National of College classes and the ends result is very impressive in my opinion.

4.0 Further Development or Research

This newly build E-Commerce Project has major potential to become the favourite online shopping Web site dealing with the trade of local outsourced goods, which allows multiple producers and farmers to come together to trade, not only in the Irish marketplace but can be used all over the globe too.

For this to happen a major investment is needed to access better plan services provided by the Cloud providers that are hosting the logical part of the business but in getting qualified staff to operate on the fulfilment teams as well.

If time would be more on my side, I will make some changes to the project code to add and enhance some features. One of these features will be the creation of an electronic payment app that will allow customers to do transactions on it for the products purchased online on the platform. Another feature will be to integrate a delivery company to facilitate this service to our customers.

5.0 References

Nick Karnik (2018) *Introduction to Mongoose for MongoDB* [Online] Available at: <https://www.freecodecamp.org/news/introduction-to-mongoose-for-mongodb-d2a7aa593c57/> [Accessed 4th May 2021]

aws.amazon.com (2021) *AWS Cloud Security Infrastructure and services to elevate your security in the cloud* [Online] Available at: <https://aws.amazon.com/security/> [Accessed 4th May 2021]

jwt.io (2021) *Introduction to JSON Web Tokens* [Online] Available at: <https://jwt.io/introduction> [Accessed 4th May 2021]

Muhd Mohaiminul Islam (2021). *MERNjs an open-source full-stack JavaScript solution* [Online] Available at: <https://mern.js.org/> [Accessed 4th May 2021]

eBay 2021 *OATMEAL Accessibility Testing for the Web* [Online] Available at: <https://ebay.gitbook.io/oatmeal/color-contrast/> [Accessed 9th May 2021]

PlaybookUX 2021 *10 Popular Usability Testing Methods* [Online] Available at:
<https://www.playbookux.com/10-popular-usability-testing-methods/> [Accessed 9th May
2021]

Jakob Nielsen (1994) *How to Conduct a Heuristic Evaluation* [Online] Available at:
<https://www.nngroup.com/articles/how-to-conduct-a-heuristic-evaluation/> [Accessed 9th May
2021]

6.0 Appendices

In this section, I am attaching supplementary documentation that is supplementary to the main body of the report.

6.1. Project Proposal



National College of Ireland

Project Proposal

Farmers Market

Date: 3rd of November 2020

BSc (Honours) in Computing

Software Development

Academic Year 2020/2021

Student Name: Daniel Costel Neagu

Student Number: x17128463

Student Email: x17124863@student.ncirl.ie

Contents

1.0	Objectives	61
2.0	Background	61
3.0	Technical Approach	62
4.0	Special Resources Required	63
5.0	Project Plan	63
6.0	Gantt Chart	64
7.0	Technical Details	65
8.0	Evaluation	66
9.0	References	66

7.0 Objectives

The most challenging part of this E-commerce Web Application is to make it work successfully, other than the technical implementation and learning of different interconnected systems and technologies from my side, it will be to move the classical approach view of a Farmers Market as a brick and mortar place of commerce and bring it to an Online E-Commerce Platform where the customers and farmers come together and do business.

This E-commerce Web Application is specially built to help the local Farmers and Producers, that are the most affected by these strange times that we all are living in, by providing them with this new Platform, where they can advertise and sell their products.

It will benefit the local community as well, where the local people can purchase again their favourite fresh products as they did once from the Farmers Markets, but this time, safely within reach of a click on the Application and within the Government guidance of respecting Social Distancing Measures.

8.0 Background

As we are aware of the current Global Pandemic caused by the virus SARS Covid.19 and the effects that it can have on all of us, the Governments of all the Country's in the World, are trying to reduce the number of people getting sick by this virus.

In Ireland sadly, almost every month the authorities are taking more and more actions on restricting peoples movements In and out of their County, City or even limiting the distance on how far the citizens can travel outside their homes, leaving areas of the city where people used to go and shop at their local Farmers Market unreachable to most of them.

As I looked at my current living situation, I soon realized that by recently moving to a different house, it's going to be a bit difficult for me now to access some areas of the city that I was fun of them before. The main cause of this matter is the result of new future restrictions to come in the nearest future from the authorities that are put in place to help to stop spreading the Virus Covid_19.

One of these places was my local Farmers Market with my favourite artisan food, cheeses, meats, organic vegetable and fruits. As I wanted to be able to continue to enjoy having them even when the restrictions will restart again to be stricter, I began looking online to see from where I can purchase them from an E-Commerce Website and having them delivered to my new address straight from the Farmers Market. The result of my search revealed that at the moment it is a major gap in the market with the presence of an E-Commerce Farmers Markets Platform, as no Website was available to trade with any products.

By not being able to find any source online from where I can purchase them, I soon realized that this a great opportunity for me to create an E-Commerce Web Application where I can bring together the local Farmers, Producers and Customers. This trade can be extended to other nationwide local Farmers Markets, serving each area in the state and allowing them to trade on my platform.

9.0 Technical Approach

At first, before I start working on building my application, I must understand how all is going to function in the big picture. One of the very first steps taken in this direction, I will start documenting the requirements specifications for the project. As part of my research in this area of gathering the requirements for my project, I will have to go back on my notes from previous semesters on the Business Analysis module and I have to read over the BABOK guide for a quip refresher.

Another step that is taken, will be by building a Mockup sketch by designing a low fidelity wireframe using the Balsamiq tool for better visualisation of the application that I want to build. In this way, if I noticed anything out of place in the prototype, I can modify the design and find the best solution for the UI.

The actual work of developing the application can start at this stage after gathering the requirements and designing the prototype. All the coding part is going to be done on Visual Studio text editor and all the libraries used for the project are going to install in there.

The deployment will be the next step in my project and for the moment I didn't decide whom to choose as a hosting services provider.

10.0 Special Resources Required

One of my special requirements for my project will be to buy the domain name for Farmers Market.ie and then depending on the packages received on the purchase I will be able to choose the Cloud provider for the application. One of them to consider will be Heroku, as I must take into consideration the database type that they support on their platform.

11.0 Project Plan

The Project Plan in Figure 1 from the page below is created by using Microsoft Excel and after watching the recommended YouTube tutorial hosted by Dr Eugene O'Loughlin, Lecturer in Computing at National College of Ireland (2013). The objectives of the project are represented in the Task column with its dedicated duration time for completion, calculate in days. A Status is appointed to each task according to the activity performed on it at the moment.

	A	B	C	D	E
	<u>Task Name</u>	<u>Start Date</u>	<u>Duration (Days)</u>	<u>End Date</u>	<u>Status</u>
1					
2	Overall Project Duration	29.9.20	242	29.5.21	InProgress
3	Project Video Pitch Idea	14.10.20	2	16.10.20	Completed/Approved
4	Project Video Pitch Editing	16.10.20	2	18.10.20	Completed
5	Reflecting Journals				
6	October Journal	29.9.20	32	31.10.20	Completed
7	November Journal	1.11.20	29	30.11.20	InProgress
8	December Journal	1.12.20	30	31.12.20	Pending
9	January Journal	1.1.21	30	31.1.21	Pending
10	February Journal	1.2.21	27	28.2.21	Pending
11	March Journal	1.3.21	30	31.3.21	Pending
12	April Journal	1.4.21	29	30.4.21	Pending
13	Project Proposal Document				
14	Project Proposal	3.11.20	5	8.11.20	Completed
15	Documentation	3.11.20	3	6.11.20	Completed
16	GanttChart	7.11.20	1	8.11.20	Completed
17	Project Ethics Form	6.11.20	1	7.11.20	Completed
18	Mid Point Presentation				
19	Implementation	9.11.20	43	22.12.20	Pending
20	Documentation	9.11.20	38	17.12.20	Pending
21	Video Presentation	17.12.20	5	22.12.20	Pending
22	Software Development				
23	Implementation	23.12.20	137	9.5.21	Pending
24	Front End Dev.	23.12.20	40	1.2.21	Pending
25	Back End Dev.	1.2.21	59	1.4.21	Pending
26	Connecting Front & Back	1.4.21	29	30.4.21	Pending
27	Deployment	30.4.21	9	9.5.21	Pending
28	Final Presentation				
29	Implementation	1.5.21	8	9.5.21	Pending
30	Documentation	3.5.21	6	9.5.21	Pending
31	Video Presentation	9.5.21	7	16.5.21	Pending
32	Project Showcase	24.5.21	5	29.5.21	Pending

Figure 13 (Project plan)

6.0 Gantt Chart

The Gantt chart is incorporating the specific tasks and they are represented on the horizontal bar and the date on the top bar indicating how long the specific tasks are supposed to last.

For a better visual representation of my project timeline, reflecting the project plan is shown below in Figure 2 or you may click on the attached Excel folder here:



Microsoft Excel
97-2003 Worksheet

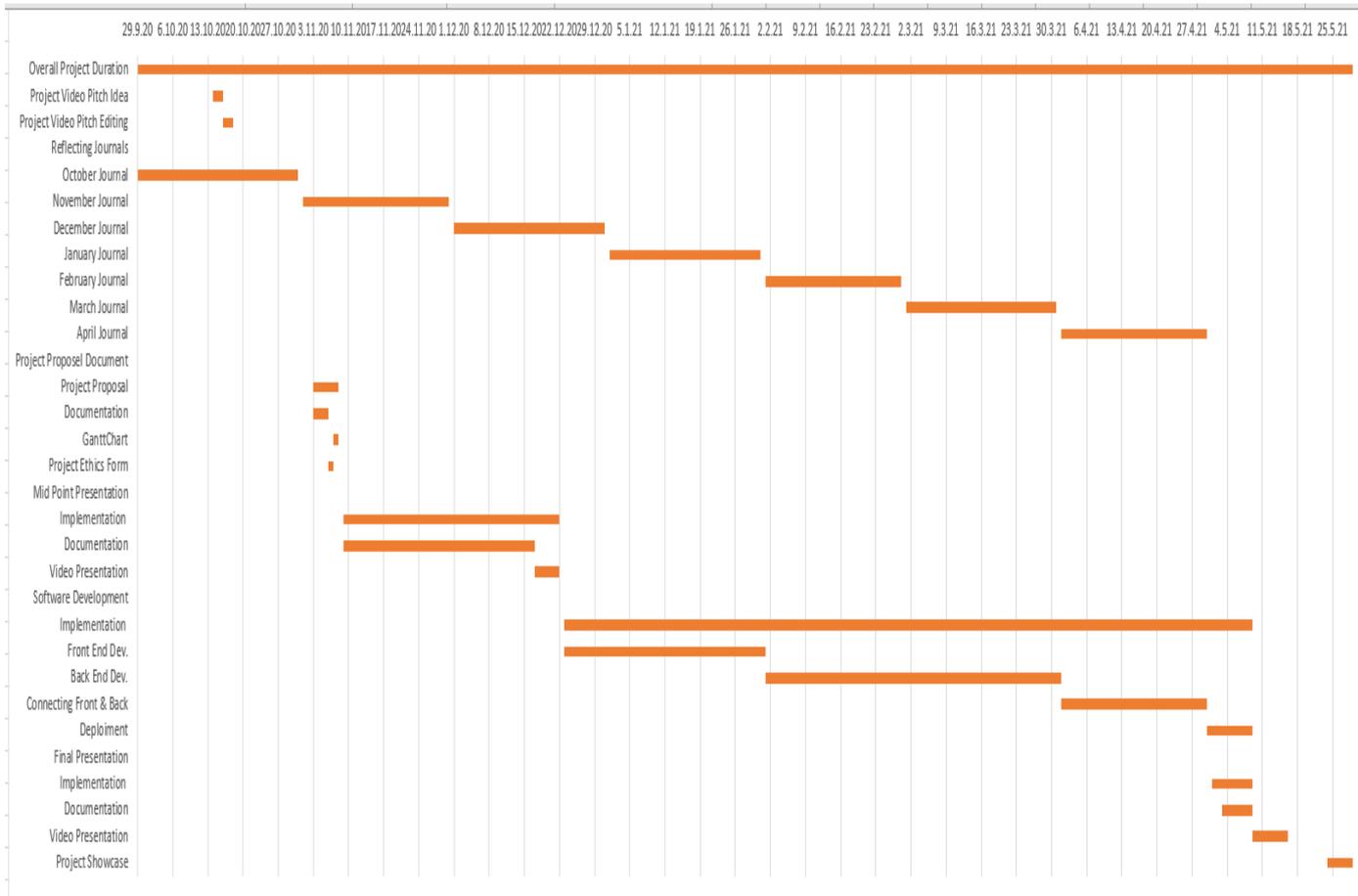


Figure 14 (Gantt chart)

7.0 Technical Details

Choosing the right tools to build a fully functional E-commerce Web Application and to be deployed successfully is going to be quite some challenge. I am more than sure that some changes to the technologies chosen by me, may occur from the moment I am writing this document up to the finish.

To make it simple for me I will split this project into three deliverable categories. I will implement a well-defined structure with dates to keep track of my progression, all these deliverables can be viewed in the Software Development Tasks described in section 5.0 of the Project Plan.

The first part of the project is to build the Front End of the Website, using a common coding language like Hypertext Markup Language (H.T.M.L.) to help me to display

the Website that I will design and to be displayed on any web browser chosen by the viewer. Cascading Style Sheets (C.S.S.) to give a little bit of styling on how the web page is going to be displayed on various devices and screens sizes. I will use a scripting language too, like JavaScript (J.S.) and Node.js to help me to add some interactivity to my website. One of the libraries that I will use in this project is React, it will be used to render dynamic components defined in JSON.

For the second part of this project, I will use Java on the server-side or the back end of the build, is to help me to produce a well define structure. For stock management and any other general-purpose data that needs to be captured and stored, I will use a relational database from Azure(Microsoft Azure, 2020), which is built for the Cloud and I can use JSON to create and interact with it.

And the third part of the project I will be the connection of the front end with the back end of the application and deployment. For these two actions, I have to look for more details later on, because I will be constrained by the Cloud provider and I will have to choose the options available at the time of deployment.

8.0 Evaluation

Considering the environment and the unfamiliar times that we live in, evaluation of the system has to be done by myself as I didn't manage to secure any contacts to build this project. I will use mock-up data to populate the necessary field, at a list in this way I will test my database and I will create some fictive profiles for suppliers and customers side, at least in this way I will make a proof of concept that my application is up and running with no problems.

9.0 References

Microsoft Azure.com (2020) *The database for modern applications* [Online] Available at: <https://azure.microsoft.com/en-us/services/sql-database/> [Accessed 4th December 2020]

Eugene O'Loughlin (2013) *How To... Create a Basic Gantt Chart in Excel 2013* [Online] Available at: <https://www.youtube.com/watch?v=QdsjVN3du78>, [Accessed 3 November 2020]

11.1. Reflective Journals



Reflective Journal

12.0

2020 / 2021

BSHCSDE4

Student Name: Your Name

Student No.X17128463

Project Journal

Contents

Project Journal.....	69
Introduction.....	70
Week 1: 28th September / 04th October.....	71
Market Research	72
Idea	72
Week 2: 05th October / 11th October.....	72
Week 3: 12th October / 18th October.....	72
Week 4: 19th October / 25th October.....	73
Vision Mission Statement	73
Week 5: 26th October / 1st November.....	74
Week 6: 2nd November / 8th November	74
Week 7: 9th November / 15th November	75
Week 8: 16th November / 22nd November.....	75
Week 9: 23rd November / 29th November	75
Week 10: 30th November / 6th December.....	76
Week 11: 7th December / 13th December	77
Week 12: 14th December / 20th December	77
Week 13: 21st December / 27th December	78
Week 14: 28th December / 3rd January.....	78
Week 15: 4th January / 10th January	78
Week 16: 11th January / 17th January	79
Week 17: 18th January / 24th January	80

Week 18: 25th January / 31st January	80
Week 19: 1st February / 7th February	81
Week 20: 8th February / 14th February	81
Week 21: 15th February / 21st February	82
Week 22: 22nd February / 28th February	82
Week 23: 1st March / 7th March.....	82
Week 24: 8th March / 14th March	83
Week 25: 15th March / 21st March	83
Week 26: 22nd March / 28th March	84
Week 27: 29th March/ 4th April	84
Week 28: 5th April / 11th April.....	85
Week 29: 12th April / 18th April.....	85
Week 30: 19th April / 25th April.....	86
Week 31: 26th April / 2nd May	86

Introduction

My name is Daniel Costel Neagu, my student number is X17128463 and I am enrolled in my final year here at the National College of Ireland, pursuing a degree in Bachelor of Science (Honours) in Computing with a specialisation in Software Development.

I have to say this year is going to be very challenging for me as I just moved to a new place and still dealing with setting things around, a new job to be started in October and that it will involve more time and commitment from my side on learning new things as they are required so I can perform my duties. I look forward to this year classes and from the look of it, I will have to be putting an extra

effort in here too, because the new way of delivering the classes online is not the same when you are in college and getting to interact and asking for help when stuck with a problem.

For my Final Project, I have decided on creating an E-Commerce Web Application that will promote the local farmers and producers of goods and produces to be sold and trade on it.

This is my Reflection Journal and I intend to make a new entry in it every week, updating the progress made on my project with all the possible interesting findings, in this way I will keep my focus on finishing it on time.

Week 1: 28th September / 04th October

My final year in college has started and I got more details on how the course will be delivered and some dates to keep in mind for deliverables.

In my first class on Software Project, I was advised to look for an idea so I can start my Project and I got the date to submit our Project Pitch Video too.

This week I looked at my current living situation and I soon realized that as I recently just moved home it is a bit difficult for me now to access some areas of the city that I was fun of them before. The main cause of this matter is the result of new restrictions to come in the nearest future from the authorities that are put in place to help to stop spreading the Virus Covid_19.

One of these places is my local Farmers Market with my favourite artisan food, cheeses, meats, organic vegetable and fruits. As I wanted to be able to continue to have them even when restrictions will restart again, I began looking online to see where I can purchase them online and having them delivered to my new address from the Farmers Market.

Market Research

The only information available online about my favourite Farmers Market was on the Visitdublin.com website, but only with the location on Google Maps and the day of the week and time that is held on. Future research was done online on a few other websites like <https://www.ireland.com/what-is-available/food-and-drink/farmers-markets/>, <https://www.ireland.com/what-is-available/food-and-drink/farmers-markets/destinations/republic-of-ireland/dublin/dublin-city/>.

Idea

I soon realized that this a great opportunity to create an E-Commerce Web Application where I can bring together the local Farmers, Producers and Costumers to trade again even in these strange times that we all are living in, but this time, safely within reach of a click on the App and within the Government guidance of respecting Social Distancing Measures.

Week 2: 05th October / 11th October

At the end of the second week on Saturday, I received more information about the project deliverables and requirements. A piece of major information that I got from the lecturer online was that if the Project that we have chosen to work on is not going to be approved by the program comity after we submit our Pitch Video, we will be giving a list from where we can choose from a new project already approved by them.

As today is Sunday and I have few free moments, I started to work on a short document, where I will present and argue why my project is worth being approved by the program comity. I put everything in writing as this helps me to visualize and assess my idea better, with few bullet points where they will be expanded upon next week.

Week 3: 12th October / 18th October

Sunday the 18th of October is the due day for the Video Pitch where I can present and hope for approval of my Final Project.

Today is Wednesday and I added more information to my text document containing my small presentation, I see that is getting shape and hopefully, by Friday evening I will be ready and I will be able to start recording the video using Microsoft Streams. I choose Microsoft Streams because I don't have any experience with any editing software yet now and by the look of it form an introduction on how to use Stream is easy and not time-consuming.

Today is Friday and the Project Video Pitch was recorded as planned in the morning and I decided to postpone the upload of the video for Sunday morning, in this way I will have the chance to rework it or to improve it if needed, or if new information is given to us at Saturday's class.

The big day for Video Pitch submission has arrived, I submitted the video 9 hours before the deadline and now I only have to wait for the approval.

Week 4: 19th October / 25th October

Vision Mission Statement

While waiting for the confirmation of approval for my E-Commerce project, I went and started to write the Vision Mission Statement for the company. This idea I got from the Strategic Management Module presented to us this semester and the result sounds like this “Our company mission is to reconnect the local community by offering them the best E-Commerce Online Platform to make their trading easier, by getting together the local producers, farmers and customers. We are promising that we will facilitate with our resources all the transactions and we will be available 24/7 at our customers' disposal.”

Week 5: 26th October / 1st November

Earlier this week on Tuesday I received an email from one of the lecturers for the Software Project class, announcing to us that the reviews for our Video Pitch are still going on but at this stage, we have a supervisor assign to us. In conclusion, I can contact my supervisor and request a meeting once my Video Pitch review has been approved.

At the end of this week on Sunday, I have to submit the Reflecting Journal for this month. As I started a new job last week I have to be honest that I got we bit behind with college work and now I feel the pressure to have it all ready for submuniton on time and have a good quality of work too, and I will do my best if I keep a tight time table.

The plan for next week is to finish a few more continuous assignments, as many deadlines for them are approaching fast. I don't want to repeat this year if I fail it because is not an option for me, considering this situation that we are in, is not too pleasant at all, is very hard to describe it.

Week 6: 2nd November / 8th November

My focus for this week is to finish on time and submit on Sunday the latest Project Proposal and the Ethics Declaration From for this Module. But as I have other CA's with deadlines for this Thursday and Sunday for the Strategic Management and Multimedia & Mobile Application Development Modules, work on my Final Project must be done in parallel with two CA's.

Today is Wednesday and I have uploaded the CAs for Strategic Management Module and on time and Mobile Application Development Modules. I have to say that I was a bit worried about not being able to reach the total required number of words in the Analysis, but I managed to pass over 2000 words and I only got 5 per cent similarity on my submission.

Thursday afternoon I uploaded on Turnitin my Project Proposal and Saturday evening after class I submitted the Ethics Declaration Form, it was quite easy as I will use mock-up data for this project.

Next week my focus will be to start working on few others CAs that due dates are approving fast and for the Project part of the tasks I will have to start doing some refreshment reading on Node.js and front end for web designing.

Week 7: 9th November / 15th November

This week I was busy with other work that was necessary to be done for the class CA's and some overtime in work and I didn't manage to do much work for the Final Software project class.

The class on Saturday was very interesting, a presentation on how I can possibly commercialise my final project was made by the guest presenter for our class. I will keep this in mind and hopefully, at the end of the course, I will have something fully working properly to be able to successfully deploy online and get some capital for improvements and expansion.

Next week looks very busy too, but I hope to have some spare time to work on my final year project assignment too.

Week 8: 16th November / 22nd November

This week I managed to create and write an IEEE format paper for one of the CA's, the amount of work that I put on this paper was demanding because this format is different from the one used by me in the past, but as I have seen from the documentation for my Final Project I will have a similar IEEE document to submit too and I have to be honest that this work was good practice for me.

In this Saturday's class, an overview on how I have to do the report writing and referencing was presented by the NCI Library staff, this information was important to me as I got a better understating of this process and helped me a lot in finishing one of my CA's that had included in an IEEE paper.

Next week focus will be to set up a separate Git Hub account from the college one, preferably to set it up using my personal email address as we have been advised in Saturday class for the Software Project.

Week 9: 23rd November / 29th November

For this week my focus is to set up a new Git Hub account using my personal email address and after Saturday's class for my final Software Project, I will start pushing up on Git Hub my local folder from

my laptop to have it as a backup copy just in case something goes wrong and I lose data from my original folder.

Today is Wednesday and work for the Final Project has resumed and I managed to create a new Git Hub account, next step is to see the Saturday class where the lecturer is going to be presenting us the methods on how to commit my work in Git Hub.

Today during the class, I managed to push my local repository to Git Hub, I had to install Git Bash too to be able to commit my work. The upload was successfully done and from now on I have an online version of my work to access it at any time. I was advised to commit every 15 to 20 minutes into my work, in this way I will avoid having conflicts or missing work in between my repositories.

Next week focus will be on submitting the current reflecting journal for Tuesday and have to finish some other CA's too.

Week 10: 30th November / 6th December

Today is Tuesday and I finished to edit my last entry in November's Reflective Journal, and I submitted it after that with no problems. I must mention that yesterday I emailed my project supervisor to clarify some details regarding the mid-point presentation report. Aqeel responded today and very helpful he set up a meeting for this Friday on Microsoft Teams so we can discuss this matter in more details.

The meeting with the project supervisor went very well today, not only he helped me understand better and where to look for information regarding the structure of the mid-point presentation, but he also managed to make me do some changes to my project structure too. Here is worth mentioning that he advises me in considering deploying my application on the NCI cloud platform and to be able to cope with gathering the requirements for my project is to create personas for user stories that will interact with my system.

Another point was made from his side that I have to choose a back end programming language when it comes to code de server-side, one of them is to consider work in Java as it is a very reliable and powerful programming language.

For next week the focus will be to do the Technical Report documentation that is necessary for the Mid-Point presentation. Another task for next week is to select the cloud provider that I will host and deploy my project in.

Week 11: 7th December / 13th December

My focus for this week is to secure access to the National College of Ireland Cloud platform OpenStack as I was advised last week by my project supervisor, and that on Thursday I will start to prepare the Technical Report for the mid-point presentation.

Today is Tuesday and I emailed the IT department here in College with my request of accessing the OpenStack Cloud, their response was prompt and helpful by guiding me to access other Cloud providers as the OpenStack platform with the campus is only available for Masters in Cloud Computing and Data Analytics, however, I could enrol on the public Cloud offered by NCI thru his subscriptions, like AWS, Azure and IBM.

Today is Sunday and I have almost all the requirements, that I was working on this week for the Technical Report, done and ready to be submitted on the Mid-Point presentation.

Next week focus is to start coding the website prototype and carry with other college assignment work on other modules.

Week 12: 14th December / 20th December

The focus for this week is to finish coding the front end of the website that is part of my final year project called Farmers Market as the presentation day of the prototype is coming soon.

For this week I still must work on a few other assignments that are part of the college requirements as they needed to be submitted on Friday and Sunday.

Today is Sunday and I managed to achieve all my proposed tasks so far for this week, with the completion and submission of my assignments and having the prototype coded done and ready to be able to make a short video presentation on it next week.

Next week I have planned to do the video presentation for the project prototype, but first I will have to do the PowerPoint presentation to accompany my video presentation and a small script for it in a word document so that it will be easier for me to present it.

Week 13: 21st December / 27th December

This week on Tuesday is the deadline for submitting the Mid-Point work done so far for this project that includes a video presentation explaining the project, a PowerPoint slide pack and the Technical Report documentation.

Today is Tuesday and I successfully uploaded all my work done for this project up to this point on Moodle submission link to be checked by the college project supervisor assigned to me.

Next week focus will be to finish the reflecting journal for December and then uploading it on Moodle. Some other work that must be done on top of project work for next week is to finish two more projects on different college modules that are due next week.

Week 14: 28th December / 3rd January

This week focus is to have this month reflective journal ready for Sunday and to carry with work on other assignments and TABAs for this semester.

Today is Sunday and I am finishing my December Journal entries on it for this week and after that, I will be submitting it on Moodle.

For next week, my focus is to carry on working on the last two TABAs left for this semester to be able to have them ready on time for submission day and then after that I will look into some tutorials to carry on with the coding part of my project.

Week 15: 4th January / 10th January

This week focus is to review the project stack to be able to understand it better and make sure I employ the right technologies to develop my product. When I started this project I only knew that is going to be an E-Commerce website application, and with the technologies taught in college during the past three years it will come in handy getting my information to help me build a successful application.

My focus is on MERN architecture which is a Full-Stack architecture, as it allows me to easily construct a three tier project (Front-end, Back-end and Database) using JavaScript and JSON all the way through.

For the Front-end tier, my MERN stack project is used React.JS, for the middle tier (Back-end/server-side) I will use Express.JS and Node.JS and for the last tier of the framework, I will use MongoDB as my Database.

This week I managed to collect all the material necessary regarding the technologies that I will use in my project from now on and I managed to review them successfully to understand better how to work with them.

Next week focus is to start creating the database model that will hold data about users and admin with their different rights and attributes. Another task for next week is to meet up with the project supervisor to update him on my project progression.

Week 16: 11th January / 17th January

This week focus is to work on the Database side of the project and get in touch with my faculty designated project supervisor over the team's video call chat meeting.

Today is Thursday and I have met up with my project supervisor to talk about my progression made so far on the project. The meeting went well, and I have received some information regarding my mid term submission on the project. During the week I have started to work on creating an account on MongoDB and deploying it on the Amazon Web Services Cloud Provider side and to be able to perform get and post actions on the database I started using Postman as an API. I am using Mongoose, which is a MongoDB object modelling tool designed to work in an asynchronous environment and it supports both promises and call-backs. Mongoose provides a straightforward, schema-based solution

to my application data model. It includes built-in type casting, validation, query building, business logic hooks and more, out of the box.

For this part of the project, I had to initiate the project to create the JS packages and create the server with the database schema. Other technologies used and installed are express validator and nodemon, I had to import dotenv library to be able to use .env

Next week focus is to start working on creating and adding on the model Database side for the Products Categories, Produces and to create a Shopping Cart to hold Produces.

Week 17: 18th January / 24th January

This week focus is to add more data to my Database by creating Produces model, Category model, and Cart.

Product creation was made on the back-end with the help of multer npm to use on the DB side: npm install --save multer and with npm install --save shortid to save files.

Creating the Category sections in the back-end on DB and I am using Slugify library: npm install --save slugify. The Shopping Cart was created too and it can hold produces added by the user after they are signed in, the admin now can add produces and categories in the DB.

Next week focus is to start my React.JS application for the Admin dashboard and meet up with my project supervisor over the teams.

Week 18: 25th January / 31st January

This week focus is to work on my React.JS application for the Admin dashboard, the front end and connected it with the back end that was created last week.

I created a new React app for the project called Admin Dashboard and with the help of Route as a logic component, I will be able to navigate to the Home, Register and Login pages. Then I installed redux, thunk libraries and to be able to handle API calls as a centralized API I used axios and to allow all URLs in between my apps(like back end and front end for the moment) I used cors library.

The Admin Dashboard connects to the back end and authenticates the user and admins.

The meeting hold on Wednesday with my project supervisor went well and he suggested that I separate access to the admin dashboard by using user categories.

Next week focus will be on adding more functionality to the admin dashboard.

Week 19: 1st February / 7th February

This week focus is to review the admin dashboard and by creating a register and login pages for the admin to be able to access the application so in this way he can add categories and products to the database from the Admin Dashboard.

After using Postman last week to authenticate and register the admin, today I will create the Login and Register Pages for the admin to be able to do all the necessary additions on the database straight from the Admin Dashboard. To allow all URLs in between my apps (like back end and front end for the moment) I used the cors library with the command: `npm i cors --save`. And to handle APIs calls as a centralized API I used axios and issued the command: `npm i axios --global`.

Purposed work on the Dashboard App for this week is completed successfully, the admin is capable to register, login, create and add new Categories and Products.

Next week focus is to create and complete the Website Profile requirements for the mid-February submission date as is part of the Project Showcase Profile.

Week 20: 8th February / 14th February

This week focus is to complete the Website Profile requirements for the mid-February submission date as is part of the Project Showcase Profile in May.

The Website Profile for the Showcase Profile is done and I have added to it my Personal Bio, the project overview descriptions, profile photo and few project photos at the required size, resizing the images at the requested sizes have proved to be more challenging the expected as I didn't have specific software to do so.

Next week focus is to create the company logo and then gather and sort the images needed for the categories and produces that are going to be sold online.

Week 21: 15th February / 21st February

This week focus is to gather and sort more images needed for the categories and produces to sell on the platform from copyright free websites with stock pictures and to create the company logo.

The websites used to gather images to use in my project are downloaded from pexels.com, pixabay.com and unsplash.com. After the downloads I proceed at removing all the additional metadata from them, then I got them renamed appropriately and resize each one individually to my scope of use in the project.

The logo for the Farmers Market was created successfully using Tailor Brands online tool. At the same time, I created some virtual merchandise products with the logo displayed on them.

Next week focus is to create a new React front-end application and start creating the in-line menu from where the categories and products in stock are displayed.

Week 22: 22nd February / 28th February

This week focus is to create a new React front-end application for the Farmers Market website and start creating the in-line dropdown menu bar with sub-headers from where the categories and products in stock are displayed.

The new App is created, and the app is called farmers-market and it was done by running the command: `npx create-react-app farmers-market`.

Then I installed more library's: `npm install --save react-router-dom redux react-redux redux-thunk axios` to make life easier.

Next week focus is to create the Get products API and Render it into the new front-end app.

Week 23: 1st March / 7th March

This week focus is to create the Get products API and then render it into the new front-end application created last week.

The products API is created and now the front-end and the back end are communicated and displaying the products that are available in the database. At the moment the products are rendered in the application by selecting the subcategory belonging to and then put in a holding container that reflects the price range of that subcategory starting from the smallest price to the highest one.

Next week focus is to create the header part for the Farmers Market Application that will be populated with the company logo, a search bar, login and register buttons and so on.

Week 24: 8th March / 14th March

This week focus is to create the Header component for the Farmers Market Application that will be populated with the company logo, a search bar, login and register buttons and the shopping cart.

The new Farmers Market Header element is done and is getting displayed on the page nicely giving the user a much-recognised view of it, a view that can be found on any E-commerce website out there and getting the user to be familiar with the Header components without the need to learn them.

Next week focus is to enable the user login and authentication functionality on the Farmers Market Application.

Week 25: 15th March / 21st March

This week focus is to create User login functionality and authentication functionality on the Farmers Market Application.

User login functionality is now enabled in my newly build E-Commerce website, the user authentication is getting done against the user records on the system and the name of the user is getting displayed on the header of the website to make it feel and look more custom-tailored for the user.

Next week focus is to enable the User logout functionality onto the Farmers Market Application and redesign the logo and adjust the padding in between header components.

Week 26: 22nd March / 28th March

This week focus is to create User logout functionality and then fix some small issues with the application accessibility's design.

Now the user can log out from the system successfully after the logout functionality was implemented in the project code.

A new logo was redesigned and uploaded as the old one was a bit blurry. The new logo was designed using the free online version of the Canva logo maker tool.

Under accessibility design issues, the padding in between the header's elements have been fixed this week too, now the Farmers Market E-Commerce website is displaying all its incorporated elements

Next week focus is to create the product details page where the website's users can view the chosen selected product with more details.

Week 27: 29th March/ 4th April

This week focus is to create the Product Details page where the website's customers can view the chosen selected product with more details.

The Product Details page is created, and the customer can view the chosen product on a separate page with all the expected features to be found on a similar E-Commerce website page.

On the Product Details page the customer can view the description and information's about the current product displayed, the number of rated stars, how many ratings and how many reviews the product received from other customers. In this way, the Customer can get a better understating of what is buying and get a reassuring peace of mind that is getting a good quality product.

Here the product is presented on the left-hand side of the page with a few small thumbnail images and a bigger size image that is the main product picture, on this page we can find any applicable discounts made and any available offers with this product offer.

Still on this page, the customer can find two more options buttons that are going to allow him to add the product to his shopping cart or to buy now the product.

Next week focus is to create the shopping Cart Page that is going to allow the customers to place and view their selected products to buy before checking out.

Week 28: 5th April / 11th April

This week focus is to create the Shopping Cart Page that is going to allow the customers to place and view their selected products to be bought before checking out.

The Shopping Cart Page is composed of two main parts, one part is dealing with displaying the cart content and the second part is displaying the breakdown of the total bill.

The Shopping Cart Page was created and now the customers can view all added shopping products added to the cart with its different time needed to be delivered to the customer address. A quantity incrementation option, a save for later, remove and place order options buttons are available to the customer's discretion to be used too as needed.

On this page, the customer can view as well the total price details that there are going to be charged on, with a price breakdown showing the price for each item and the delivery charges.

Next week focus is to enable the cart quantity and the price details to be updated on the current page by adding or removing the number of a product's quantity and then the price breakdown to reflect these new changes.

Week 29: 12th April / 18th April

This week focus carries on finishing the Cart Page that I have started last week, now I only have to implement and enable the quantity of the product to be changed and the price to show the new updates made by this action in the Cart Page.

The new implementation is enabled as planned and now the Cart Page is fully functional. After the development of this page was completed this week, the customer can successfully increase or decrease the quantity of produces from their cart or remove entirely produces from it too.

The price details now reflect all the changes that the customer does on the cart and by pressing the Place Order button the customer can proceed to the next page that is the Checkout Page, this page will be the next step taken towards completing his purchase on the Farmers Market website.

Next week focus is to create the Checkout Page, on this new page that I will develop the customer can update his details with the address where the purchases can be delivered and the payment method too.

Week 30: 19th April / 25th April

This week focus is to create the Checkout Page that is going to allow the customer to edit user details for the current purchases to be made.

The Checkout Page is now fully developed and is enabling the customer to finalize user purchase in few steps.

On this page, the customer can add a new address, edit or select an already existing address where the products can be delivered too. Still on this page, the customer can review the order summary and take actions by modifying it, the price breakdown can be reviewed again and he can select a payment option but for the moment this option will be cash on delivery.

A confirm order button is present after all the necessary steps are completed to allow the purchases to be done.

Next week focus is to create an Order Details Page following the Checkout Page on completion of the transaction, then the Orders Page which is a holder from where all the orders made on the platform by the user can be viewed from. On the Admin application side, the Orders Page will be enabled where an admin can actively act on an order made from a transaction coming from the Farmers Market website.

Week 31: 26th April / 2nd May

This week focus is to create the Farmers Market Order Details, Orders Pages and the APIs that it will allow the user to review all the purchase made on the platform by the current account holder. This

week I will create the Admin Orders Page too on the Admin application to perform actions on the incoming and outgoing transactions made on the platform.

Order Details Page is now up and running, here the user can view again the recent transaction made and is displayed just after the Checkout Page. The same page can be displayed when a specific order is selected from the Orders Page and the user can view the progress made by the seller as the most important part of the information on this page among other specific information. The state of progress of the order is represented in an infobox that states if the order was Packed, Shipped or Delivered by the Farmers Market team.

Orders Pages is completed too and here the user can view all time orders made on the account by them. On this page, the user can select a specific order and review it on the Order Details Page.

Admin Orders Page is up and running too and is giving the admin to view and take actions on all orders coming into the system from all the user. Here the admin can change the order status from Packed, Shipped or Delivered accordingly. As soon as an order gets updated the user can view it on his Order Details Page.

For the next two weeks focus will be to improve and fix my code, create the Footer as a component, create the Home Page and deploy the code on the cloud.