

# Table of Contents

Table of figures .....	3
Table of Tables .....	5
Acknowledgement .....	6
Summary .....	6
Chapter 1: Introduction .....	7
1.1    Introduction of the system .....	7
1.2    Background of the system .....	7
1.2.1    Features .....	7
1.2.2    Scope and aim .....	7
1.2.3    Objectives .....	7
Chapter 2: Analysis .....	8
2.1    Introduction .....	8
2.2    Analysis Methodologies .....	8
2.3    Information Gathering .....	8
Interview .....	9
Questionnaire .....	9
2.4    Feasibility Study .....	10
Technical Feasibility .....	10
Economic Feasibility .....	10
Legal Feasibility .....	10
Operational Feasibility .....	11
Scheduling Feasibility .....	11
2.5    System Requirement Specification .....	11
2.5.1    Functional Requirement .....	11
2.5.2    Non-Functional Requirement .....	13
2.5.3    MoSCoW Prioritization .....	13
2.5.4    Requirement Specification .....	15
2.6    Use Case Diagram .....	16
2.7    System Architecture .....	19
2.7.1    National Language Architecture [NLA] .....	19
2.7.2    Initial Class Diagram .....	20

2.7.3 System Architecture.....	21
Chapter 3: Design .....	22
Introduction.....	22
3.1 Structural Modelling.....	22
3.1.1 Class Diagram .....	22
3.1.2 Data Flow Diagram [DFD].....	25
3.2 Behavioral Modelling.....	27
3.2.1 Activity Diagram.....	27
3.2.2 Sequence Diagram .....	31
3.3 Database Modelling.....	34
3.3.1 ER Diagram.....	34
3.3.2 Data Dictionary .....	35
3.4 UI Modelling.....	36
3.4.1 Prototyping .....	36
Chapter 4: Implementation .....	50
Introduction.....	50
Tools used for project .....	50
Development Environment .....	50
Conclusion.....	51
Chapter 5: Testing.....	52
Testing .....	52
Conclusion.....	69
Appendix.....	70
References.....	117

## Table of figures

Figure 1: Use Case Diagram .....	16
Figure 2: Initial Class Diagram .....	20
Figure 3: System Architecture .....	21
Figure 4: Class Diagram .....	24
Figure 5: Admin DFD .....	26
Figure 6: Registration Form .....	37
Figure 7: Login Form .....	38
Figure 8: User's Profile .....	39
Figure 9: User's Homepage .....	40
Figure 10: About Us .....	41
Figure 11: Contact Page .....	42
Figure 12: Cart Details .....	43
Figure 13: Admin Login Page .....	44
Figure 14: Admin Profile .....	45
Figure 15: Admin Password Change .....	45
Figure 16: Admin Dashboard .....	46
Figure 17: Manage Category .....	47
Figure 18: Manage Products .....	48
Figure 19: Manage Orders .....	48
Figure 20: Manage Testimonials .....	49
Figure 21: Manage Messages .....	49
Figure 22: Manage Access .....	50
Figure 23: Validation .....	54
Figure 24: Valid Details .....	55
Figure 25: Account Created Successfully .....	56
Figure 26: Successful Login .....	57
Figure 27: Invalid Login .....	58
Figure 28: Admin Login Validation .....	59
Figure 29: Valid Info of Admin .....	60
Figure 30: Admin Login Successful .....	61
Figure 31: Add to Cart .....	61
Figure 32: Add to Cart Successful .....	62
Figure 33: Checkout .....	62
Figure 34: Checkout Order Successful .....	63
Figure 35: Message to Admin .....	63
Figure 36: Message from Users .....	64
Figure 37: Before Updating Products .....	64
Figure 38: Adding Product by Admin .....	65
Figure 39: Product Added On Product Details .....	66
Figure 40: Updated Products .....	66
Figure 41: Home Page .....	70
Figure 42: Products Page .....	71
Figure 43: Index.php .....	72

Figure 44: Home.php .....	75
Figure 45: Header.php.....	76
Figure 46: Footer.php .....	76
Figure 47: About Page.....	78
Figure 48: About.php .....	80
Figure 49: Contact Page .....	81
Figure 50: Contact.php.....	83
Figure 51: Login Form .....	84
Figure 52: SignUp Form .....	84
Figure 53: Login and Signup.php.....	87
Figure 54: Add to Cart Page.....	88
Figure 55: Cart.php .....	90
Figure 56: Profile Page.....	91
Figure 57: Profile.php .....	93
Figure 58: Category Page.....	93
Figure 59: Add Category.php .....	95
Figure 60: New Category Form Page .....	96
Figure 61: Add New Category.php .....	96
Figure 62: Edit Category .....	97
Figure 63: Edit Category.php .....	98
Figure 64: Product Page .....	98
Figure 65: Product.php.....	100
Figure 66: New Product Form .....	101
Figure 67: New Product.php .....	102
Figure 68: Edit Product Form .....	103
Figure 69: Edit Product.php .....	104
Figure 70: Orders Page.....	104
Figure 71: Orders.php .....	106
Figure 72: Testimonials Page.....	107
Figure 73: Testimonials.php .....	109
Figure 74: Add Testimonial Form .....	109
Figure 75: Add Testimonial.php .....	110
Figure 76: Messages Page.....	111
Figure 77: Messages.php .....	113
Figure 78: Admin Profile Edit Form .....	113
Figure 79: Edit Profile.php .....	115
Figure 80: Admin Password Edit Form .....	115
Figure 81: Edit Password.php .....	116

## Table of Tables

Table 1: Functional Requirement.....	12
Table 2: Non-Functional Requirement.....	13
Table 3: MoSCoW Prioritization.....	14
Table 4: Use Case .....	18
Table 5: Class Diagram Notation.....	23
Table 6: Notation Diagram of Data Flow Diagram .....	25
Table 7: Notation of activity diagram.....	29
Table 8: Notation of Sequence Diagram .....	32
Table 9: ER Diagram .....	34
Table 10: Table Names .....	35
Table 11: Table Names .....	35
Table 12: Data Dictionary .....	36

## Acknowledgement

I would like to express my humble gratitude to my teachers for guiding me all the way from understanding the basics of web development to develop a complete project. I am very much grateful for their guidance, effort and mentorship that they put to make me understand the stuffs required to develop a functional website. I would also like to acknowledge all my friends who have assisted and supported me during this project.

And last but not least, I would like to express my sincere regards to other people who have helped me directly or indirectly to clear my doubts, confusion and helping me to complete the project in limited period of time.

## Summary

Our country is an agricultural country. Most of the people in our country are engaged in agriculture as their daily livelihood and occupation. People produce a large amount of vegetables, fruits, etc. They sell these items at their local market. Most of these agricultural products go waste since they are produced in large quantity and the number of consumption at local level is small compared to the number of the production. If these highly produced agricultural items are to be sold where the consumption of these products are high then these products will not go waste and people engaged in these sectors will benefit much more.

Therefore, Online Vegetable and Fruits is a solution to overcome the problem people engaged in agricultural fields. This project will help people to sell their products online which will reduce the quantity of agricultural products going on waste. Since the Online Vegetable and Fruits helps people to sell and buy fresh products online, not only the seller will benefit from it buyers will also benefit as it helps people to buy as per their wish. Online Vegetable and Fruits will be user friendly and will be easy to use for those people who have a basic knowledge of online activity. Since the world is advancing to online transactions, Online Vegetable and Fruits will help people to purchase fresh, cost efficient and online delivered products.

# **Chapter 1: Introduction**

## **1.1 Introduction of the system**

Online Vegetable and Fruits is a functional website which is made for selling and buying of the fresh vegetables and fruits online. It will help sellers by providing a platform to sell their products online and buyers to buy fresh vegetables and fruits at the tip of their fingers. The website provides a full detail on the products, and provides an online delivery. It is also working on the return policy in case the unexpected product is mistakenly delivered. It is aiming to expand its store to facilitate more customers.

## **1.2 Background of the system**

Online Vegetable and Fruits is a web application which contains useful information on fresh, healthy vegetables and fruits. It is like a platform to serve users with details on the vegetables and fruits that are available at market on a reasonable price. Traditionally, the selling and buying of the vegetables and fruits were done in person. It consumes most of the time since we used to visit different stores to get the product we need. And as for sellers, they need to rent stores to contain the vegetables and fruits which was a little costly.

But with this online platform, customers can access their preferred choices. Sellers can maintain profit by selling a large amount of products.

### **1.2.1 Features**

Online Vegetable and Fruits is a project which consists of valuable features including CRUD operations. One of the important feature is to buy the products online. Other features include online delivery, administrative features, etc.

### **1.2.2 Scope and aim**

The main purpose of the organization is to provide a platform for the people who can't find a proper platform to sell their products due to which the products go waste and to provide people fresh vegetables and fruits by simply browsing Internet.

- Enhance the overall production, perfection and furnish recommended services.
- To firm the individual details, statistics and subscription of the customer.

### **1.2.3 Objectives**

To refine the drawbacks the farmers are facing with the features of this project are the main objectives of creating this system. Some of those objectives are:

- To provide accurate, efficient and updated details.
- To reduce the manual effort.
- To provide secure, effective transactions.

## Chapter 2: Analysis

### 2.1 Introduction

Analysis means a systematic examination and evaluation of data and facts to uncover and understand cause-effect relationships, thus providing basis for problem solving and decision-making. (Analysis, 2020) We should do a proper analysis before starting any project. Proper analysis of the project will help to make the project much effective, and also facilitates us in different ways.

### 2.2 Analysis Methodologies

Analysis is one the most important part of developing a system which will help us to point out the features and functionality that an application needs to have. It helps us to point out all of the examination that are needed in an application to make the application much more applicable.

For analysis of this project, object-oriented analysis [OOA] methodology has been chosen. It analyzes and designs an application based on the system's logical components that interact with one another. It also combines all characteristics, states, and behaviors together into a single analysis process, instead of splitting them into different stages, as many other analysis methodologies would do.

Following interaction diagrams have been used in my project;

- Class Diagram  
Class Diagram in the Unified Modeling Language (UML) is a type of static structures diagram that shows the collection of classes, attributes, interfaces, their association and relationships among objects.
- Activity Diagram  
Activity Diagram is a behavioral diagram in Unified Modeling Language (UML) which describes the dynamic aspects of the system. It is an advanced version of flow chart that models the flow of one activity to another activity.
- Sequence Diagram  
Sequence Diagram are interaction diagrams that detail how operations are carried out. They are time focus and show the order of the interaction visually by using the vertical axis of the diagram to represent time what messages are sent and when. (Sequence Diagram, 2020)

Detailed insight of above mentioned interaction diagrams will be given in the design phase.

### 2.3 Information Gathering

The simplest way to define information gathering would be the process of collecting information about something you are interested in. Information are gathered in order to understand the requirements of the system. The more the information gathered about the system, the more the probability to obtain relevant results. I have used the following two information gathering techniques which involves direct communication with the clients; interview and Questionnaire.

## Interview

An information converging interview is an administered discussion with a particular motive that utilizes a question-and-answer scheme. In the interview you are eager to conceive the opinions of the interviewee and his/her sentiments about the contemporary state of the system, personal and organizational goals, and informal procedures for interacting with information technologies. While interviewing, seek the opinions of the person you are interviewing. Opinions may be more important and more revealing than facts. (Interviewing in Information Gathering, 2020) Some of the questions that I prefer to ask to the interviewee are;

- What are the general and specific features that you want in the system?
- How can we make the system much more user friendly?
- Are you having any issue while using the system?

## Questionnaire

Questionnaire in an information gathering technique helps system analysts to study attitudes, beliefs, behavior, and characteristics of several people in the system/organization who may be affected by the current and proposed systems. Responses gained through questionnaire using closed questions can be quantified and can be used to survey a large sample of system users to sense problems or raise important issues before interviews are scheduled. It helps to distinguish the behavior and preferences of people in a specific field. (System Analysis, 2020) Some of the questionnaire that I prefer to ask are;

- What is your budget estimation for the project?
- Do you need further maintenance services?
- Have you registered domain for the company?
- Does your company have branding inclusions like logo/banners/established images?
- Do you have technicians to run the website?
- How often do you update the contents in website?
- Do you have any hosting company preference?
- Do you want us to train your staff to operate the website?
- What is your deadline for the project?
- Who are going to use the website?
- Would you like us to handle your other web aspects like digital marketing, search engine optimization (SEO) or you already have other dedicated member?
- Who are your targeted audience?
- Who will provide the contents for the website?
- Do you want to include any 3<sup>rd</sup> party application in your website?
- What are the features you want in your website?

## 2.4 Feasibility Study

A feasibility study is an analysis that takes all of a project's relevant factors into account- including economic, technical, legal, and scheduling considerations- to ascertain the likelihood of completing the project successfully. Feasibility studies are used to discern the pros and cons of undertaking a project before they invest a lot of time and money into it. It also provides a company's management with crucial information which could prevent the company from entering blindly into risky businesses. It is always good to have a contingency plan that you test to make sure it's a viable alternative in case the first plan fails while doing a feasibility study. (feasibility study, 2020)

For the project I have done the following types of feasibility study;

### Technical Feasibility

Technical feasibility study focuses on the technical resources available to the organization. The project is a total website which is made on Laravel (a php web framework). To study technical feasibility of the project, we need to confirm if the existing computer system are compatible with the technologies that we will use. The project will work perfectly across all the devices, browsers since it will be optimized properly. The requirements needed for project to run successfully are freely available and manageable. Therefore, the project is technically feasible.

### Economic Feasibility

Economic feasibility typically involves a cost/benefits analysis of the project which helps organizations to determine the viability, cost, and benefits associated with a project before the financial resources are allocated. Since the project is made up of freely available technologies, basic operating computer system, no cost will be charged. Some cost will be charged during the hosting of the website, for the development team. The project will be small, efficient so there will be training for short period of time which minimizes the cost of training. And while the records of clients and other credentials will be off the paper it will also minimize the cost. Therefore, the project is economically feasible.

### Legal Feasibility

Legal feasibility determines whether any aspect of the proposed project conflicts with legal requirements like data protection acts or social media laws. (Feasibility Study, 2020) For the project, I have used open source tools that have free license which will not permit the violation of the legal conflicts. And as for the website, we have a privacy policy which will clear the data related to the users. Thus, it will help the website to be fully secure which depicts no legal crimes. Therefore, the project is legally feasible.

## Operational Feasibility

Operational feasibility involves undertaking a study to analyze and determine whether and how well the organization's needs can be met by completing the project. It also examines how a project plan satisfies the requirements identified in the requirements analysis phase of system development. (Feasibility Study, 2020) A proper orientation and basic trainings will be provided to the users so the introduction of new system will not affect anyone. In this way, the project will be operationally feasible.

## Scheduling Feasibility

Scheduling feasibility is the most important for project success because a project will fail if not completed on time. An organization estimates how much time the project will take to complete. My project will have enforced several time estimates on which several chunks of works related to the project will be done which will balance the work load and help the project to be completed on time. In this way, my project will be on time.

## 2.5 System Requirement Specification

A System Requirements Specification (SRS) is a documents or set of documentation that describes the features and behavior of a system or software application. It includes a variety of elements that attempts to define the intended functionality required by the customer to satisfy their different users. (System Requirements Specifications, 2020)

### 2.5.1 Functional Requirement

The functional requirement specification documents the operations and activities that a system must be able to perform. It is designed to be read by a general audience. Readers should be able to understand the system, but no particular technical knowledge should be required to understand the document. (Functional Requirements, 2020)

List of functional requirements are shown below;

ID	Title	Description	Dependency	Rational
FR_101	Registration	Add new user	N/A	To access secure system
FR_102	Login	Registered user can login to the system	FR_101	To provide access to a valid user
FR_103	Verify password	To check the password entered with the database	FR_101, FR_102	To authenticate the user
FR_104	Display Dashboard	To provide a functional front page	FR_101, FR_102	To navigate entire system easily
FR_105	Edit Profile	Change personal information	FR_101, FR_102	To update user's information in case of

				emergency
FR_106	Add Category	Add new category of fruits and vegetables	FR_102	Add new type of category of the product
FR_107	Update Category	Add new details on the existing category details	FR_102, FR_106	For any additional information is needed to be add
FR_108	Delete Category	Remove the category	FR_102, FR_106	To remove the unnecessary category
FR_109	Add Product	Add new product	FR_102, FR_106	To add new product under different category
FR_110	Update Product	Add new details on the existing product details	FR_102, FR_109	To add extra information about the product or remove the unwanted details
FR_111	Delete Product	Remove the product	FR_102, FR_109	To remove the product that is unavailable
FR_112	Add to cart	Add product to the cart	FR_102, FR_109	To buy the product
FR_113	Delete from cart	Delete the added product	FR_102, FR_112	To not to buy the product incase of change of mind
FR_114	View Order Details	Lists out all the order	FR_102, FR_112	Details the list of all the order placed by the user
FR_115	Approve Order	Approve the order for delivery	FR_102, FR_112	To approve the order placed by the user.
FR_116	Delete Order	Deletes the order	FR_102, FR_114	To delete the order incase of lack of the product
FR_117	Add Message	Sends message to the admin	FR_102	To send any information that a user possesses to the admin
FR_118	View Message	Lists out all the message	FR_102, FR_117	To view the message the users have send to the admin
FR_119	Delete Message	Deletes the message	FR_102, FR_117	To delete the message the admin has already seen
FR_120	Manage Access	Maintains the level of access	FR_101, FR_102	To provide certain level of authority
FR_121	Logout	Logout at desired time	FR_102	To secure users' data to be misused

Table 1: Functional Requirement

### 2.5.2 Non-Functional Requirement

Non functional requirements define system attributes such as security, reliability, performance, maintainability, scalability, and usability. They serve as constraints or restrictions on the design of the system across the different backlogs.

List of non-functional requirements are shown below;

ID	Title	Description	Dependency
NFR_101	Scalability	The system should be able to manage the users' information.	N/A
NFR_102	Verification	The system should verify the password and the point of access to verify who the user claim to be.	NFR_104
NFR_103	Reliable	The system should be strongly reliable so that the users' information is strongly preserved and protected.	NFR_109
NFR_104	Security	The system should be secure since it contains all the valuable information about the users.	NFR_102
NFR_105	Efficiency	The system should be efficient so that the users can access it without any errors.	NFR_105
NFR_106	Acceptance	The system should accept all the users new or old registered to the system.	NFR_102
NFR_107	Intuitiveness	The system should be easily understandable so that the users find it more way friendly to use.	NFR_106
NFR_108	Usability	It defines that the services provided by the system should be easily useable with no problems.	NFR_103, NFR_105
NFR_109	Availability	The system should be accessible form anywhere so that the user can do what ever they can with the help of the site.	NFR_103, NFR_106
NFR_110	Quality	The system should be qualified for performing all the functional and non-functional requirement of the system.	NFR_105, NFR_107, NFR_108

*Table 2: Non-Functional Requirement*

### 2.5.3 MoSCoW Prioritization

MoSCoW Prioritization, also known as the MoSCoW method or MoSCoW analysis, is a popular prioritization technique for managing requirements. The method is commonly used to help key stakeholders understand the significance of initiatives in a specific release.

The acronym, MoSCoW stands for 4 different categories of initiatives: must-haves, should-haves, could-haves, and will not have. (MoSCoW Prioritization, 2020) The MoSCoW prioritization for my project is as follows:

ID	Functional and Non-Functional Requirements	MoSCoW
MP_01	Registration	Must have
MP_02	Login	Must have
MP_03	Verify Password	Must have
MP_04	Display Dashboard	Should have
MP_05	Edit Profile	Could have
MP_06	Add Category	Must have
MP_07	Update Category	Must have
MP_08	Delete Category	Should have
MP_09	Add Product	Must have
MP_10	Update Product	Must have
MP_11	Delete Product	Should have
MP_12	Add to cart	Must have
MP_13	Delete from cart	Must have
MP_14	View order details	Should have
MP_15	Approve order	Must have
MP_16	Delete order	Should have
MP_17	Add Message	Must have
MP_18	View Message	Must have
MP_19	Delete Message	Should have
MP_20	Manage access	Must have
MP_21	Logout	Must have
MP_22	Scalability	Could have
MP_23	Verification	Must have
MP_24	Reliable	Could have
MP_25	Security	Must have
MP_26	Efficiency	Must have
MP_27	Acceptance	Should have
MP_28	Intuitiveness	Must have
MP_29	Usability	Should have
MP_30	Availability	Should have
MP_31	Quality	Must have

Table 3: MoSCoW Prioritization

#### **2.5.4 Requirement Specification**

A requirement specification is a document that captures complete description about how the system is expected to perform. The requirement specification for the project is as follow:

##### **Software Requirement**

Programming Language: PHP

Framework Used: Laravel

User Interface Design: HTML, BOOTSTRAP, jQuery

Operating System: Windows 7 and above

Server: Apache

Database: MySQL

Web Server Solution Package: Xampp

##### **Hardware Requirement**

RAM: 2GB and above

Processor: Intel Processor Dual Core and above

## 2.6 Use Case Diagram

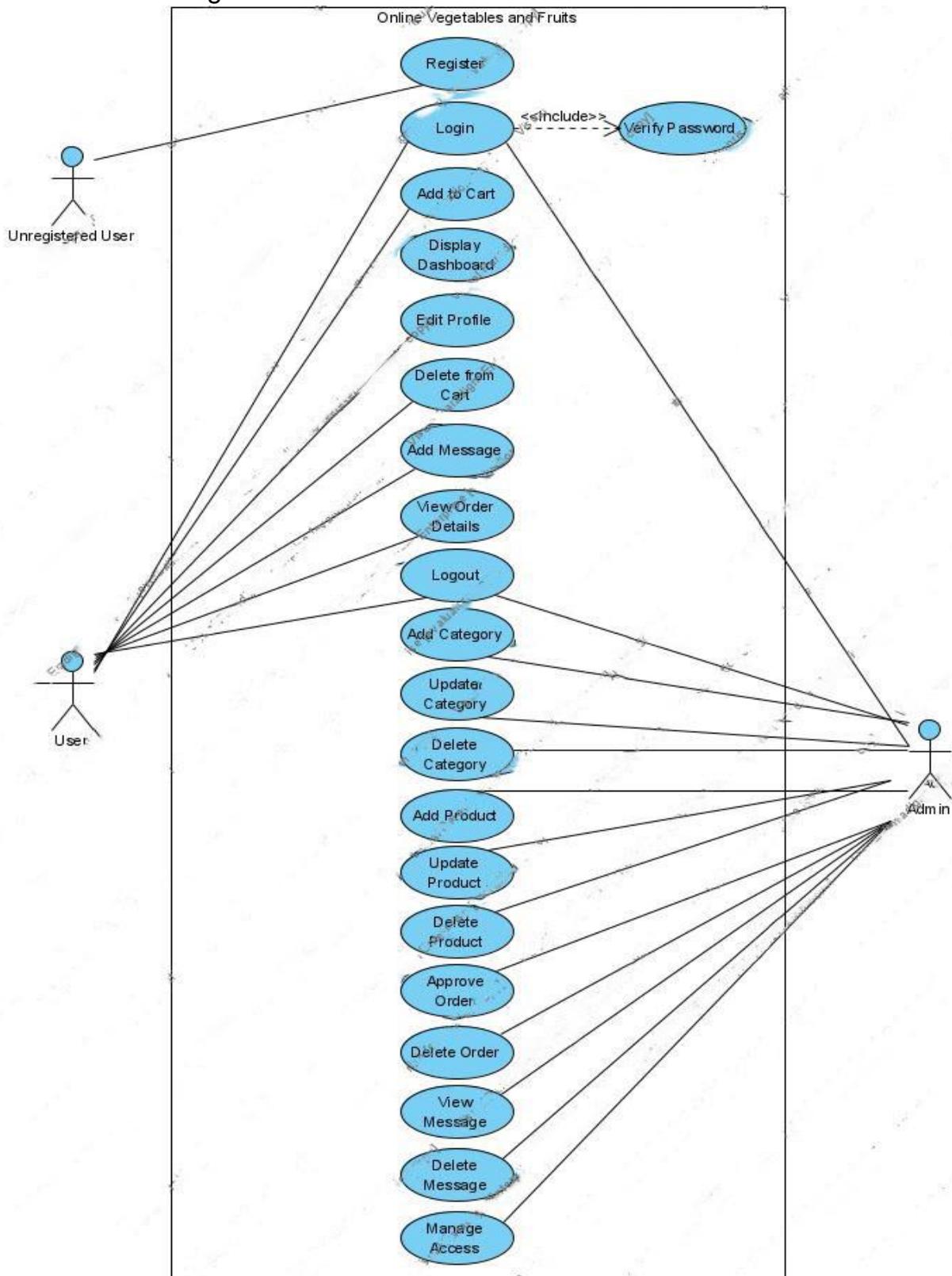


Figure 1: Use Case Diagram

A use case diagram is the primary form of system/software requirements for a new software program underdeveloped. Use cases specify the expected behavior (what), and not the exact method of making it happen (how). A key concept of use case modeling is that it helps us design a system from the end user's perspective. It is an effective technique for communicating system behavior in the user's terms by specifying all externally visible system behavior. (Use Case Diagram, 2020) The use case diagram shown above describes the relationship between the users and the system including the system's functions. The relation is explained in the table below.

ID	Use Case Title	Summary	Alternative Sequence	Actor
UC_01	Register	User need to register themselves to the system to access the system.	N/A	Unregistered User
UC_02	Login	Administrative and regular users' login is done.	In case of incorrect password a message pop ups.	Admin, User
UC_03	Verify password	The system verifies the password entered with the password on the database	N/A	The System
UC_04	Display dashboard	Both users land firstly here.	N/A	Admin, User
UC_05	Edit Profile	Both users can edit their profile.	Message is displayed after editing the profile.	Admin, User
UC_06	Add Category	Admin adds new category.	Message is displayed after a category is added.	Admin
UC_07	Update Category	Admin updates the category.	Message is displayed when the category is updated.	Admin
UC_08	Delete Category	Only admin can delete the category.	Message is displayed when the category is deleted.	Admin
UC_09	Add Product	Admin can add new products.	N/A	Admin
UC_10	Update Product	Admin can update the existing product.	N/A	Admin
UC_11	Delete Product	Admin can delete the existing product.	N/A	Admin
UC_12	Add to cart	Users can add the product to their cart	N/A	User
UC_13	Delete from cart	Users can delete the product they added to their cart	Message will be displayed when a product from the cart is deleted.	User
UC_14	View order details	Admin will view all of the orders made by the users.	N/A	Admin
UC_15	Approve Order	Admin is capable of approving the orders.	N/A	Admin
UC_16	Delete Order	Admin can delete the	N/A	Admin

		orders.		
UC_17	Add Message	Users can send message to the admin.	N/A	Users
UC_18	View Message	Admin will see the message send by the users.	N/A	Admin
UC_19	Delete Message	Admin can delete the message.	A message saying message deleted will be displayed	Admin
UC_20	Manage Access	Only admin is capable of controlling the access provided to the users.	N/A	Admin
UC_21	Logout	Both users and admin can logout of the system as per their choice.	Logged out successfully message will be displayed.	Users, Admin

Table 4: Use Case

## 2.7 System Architecture

The system architecture describes how a product is composed of components and how the components are related to each other via interfaces. (System Architecture, 2020)

### 2.7.1 National Language Architecture [NLA]

Salaka Vegetables and fruits is a farm where fresh, local fruits and vegetables are grown and are sold. It is located at Salleri, Solukhumbu and is at service since the early 2000. Here a large amount of fresh vegetables and fruits are grown and sold at the local market. All the works in this farm are done in a traditional method. They sell their products only at local markets. Since it produces vegetables and fruits at large number the consumption of these goods at local market is lower. Due to this these products are going to waste.

So, the owner of this farm intends to expand its service to other places. So, he has asked to make a web application that enables users from other areas to be able to buy their products.

Here are his requirements:

There are many farmers and people who wish to sell their fruits and vegetables online. These people who produce fresh vegetables and fruits sometimes have to let their fruits and vegetables go to waste because the fruits and vegetables they produce are in high quantity and the number of consumption is low. So the owner wants me to make a web application that will help these people. The web application should be able to help user register to the system. It should be user-friendly, easily accessible. It should also take users message so that they can elevate the level of application to another level. Customers should be able to order the product they want to purchase and admin can add as many product as they want to provide the information about products to the customers.

List of nouns (potential class) are:

Users, Customer, Order, Farmers, Fruits, Vegetables, Product, Admin.

Final candidate class has been finalized following the rules given below:

- Removed the matching, irrelevant, synonyms and outside the scope nouns.
- Removed the architectural level action noun. Also, the noun that could be design during development were skipped.

Final Candidate class

- Customer
- Order
- Product
- Admin

## 2.7.2 Initial Class Diagram

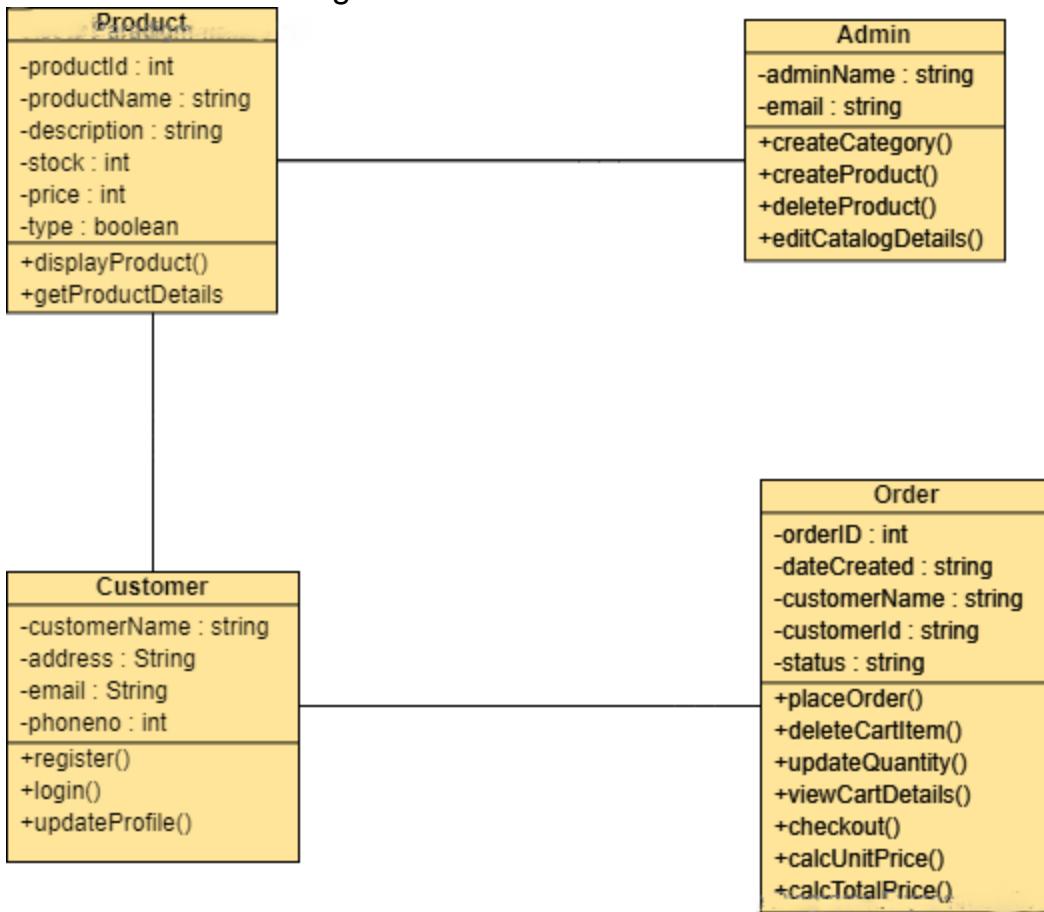


Figure 2: Initial Class Diagram

### 2.7.3 System Architecture

3-Tier system architecture has been used which holds the user interface, data storage, isolated platform and business logic. It includes client-server relation like client tier, application and database tier. It is used because all the tasks are separated which are handled by each layer in client-server application. Interface services are displayed to user in client tier whereas all the business logic is maintained in application tier. All the information are received and preserved in the data tier independently from other tiers. It also helps to keep the availability, scalability and performance of the overall system.

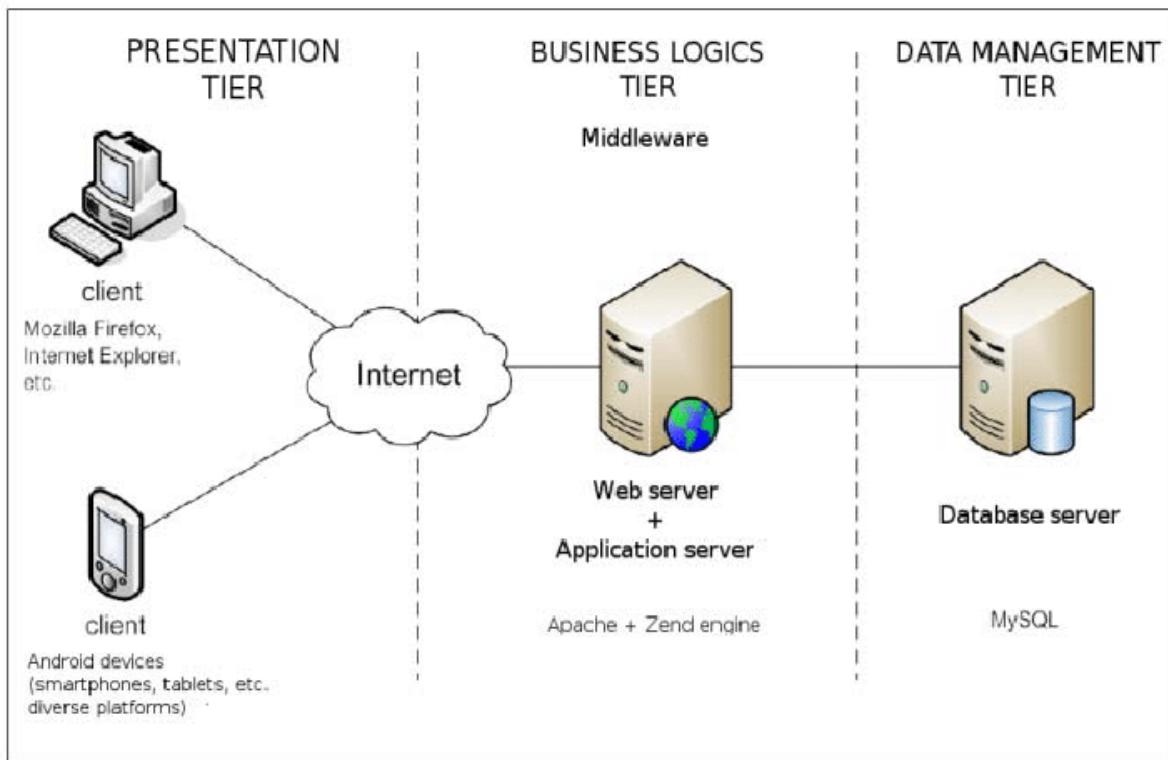


Figure 3: System Architecture

# Chapter 3: Design

## Introduction

The stage of development where all the decision regarding the final product are made to make the product much better and smooth is known as design in programming. User interface, logical function are planned before the coding of an application in the design phase. It is executed based on the requirements of the application.

### 3.1 Structural Modelling

Structural Modelling is an analysis technique which presents the logical organization of the objects without indicating how they are stored, created, or manipulated so that analysts can focus on the business, without being distracted by technical details. (STRUCTURAL MODELING, 2020)

#### 3.1.1 Class Diagram

Class diagram is a static diagram which represents the static view of an application. It describes the attributes and operations of a class and also the constraints imposed on the system. It is a part of unified modelling language. (UML - Class Diagram, 2020)

The importance of class diagram are:

- To show the classes and how it interacts within the system.
- To show the relationship between class and object as it is an object oriented application.
- To outline all the object which the system contains and finalize relationship between them.
- To reduce maintenance time as it provides the structural diagram of the application.

Notation of Class Diagram:

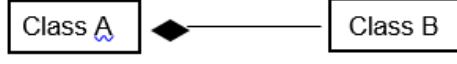
S.N	Notation	Description	Remarks
1.	 Association	It represents that there is relation between 2 class in association relation	-
2	 Aggregation	It represents special association relation as aggregation with a parent and child classes where child class is determined as type of parent class.	For example many of class B can be related to Class A.
3	 Composition	It represents aggregation where the types of classes are destroyed when the whole system is destroyed to which it depend on.	When class B depends upon class A, class B gets destroy as of no dependency left.
4	 Dependency	It represents the dependency of one class to other.	Any edit in the class A may effect on the class B for an instance.

Table 5: Class Diagram Notation

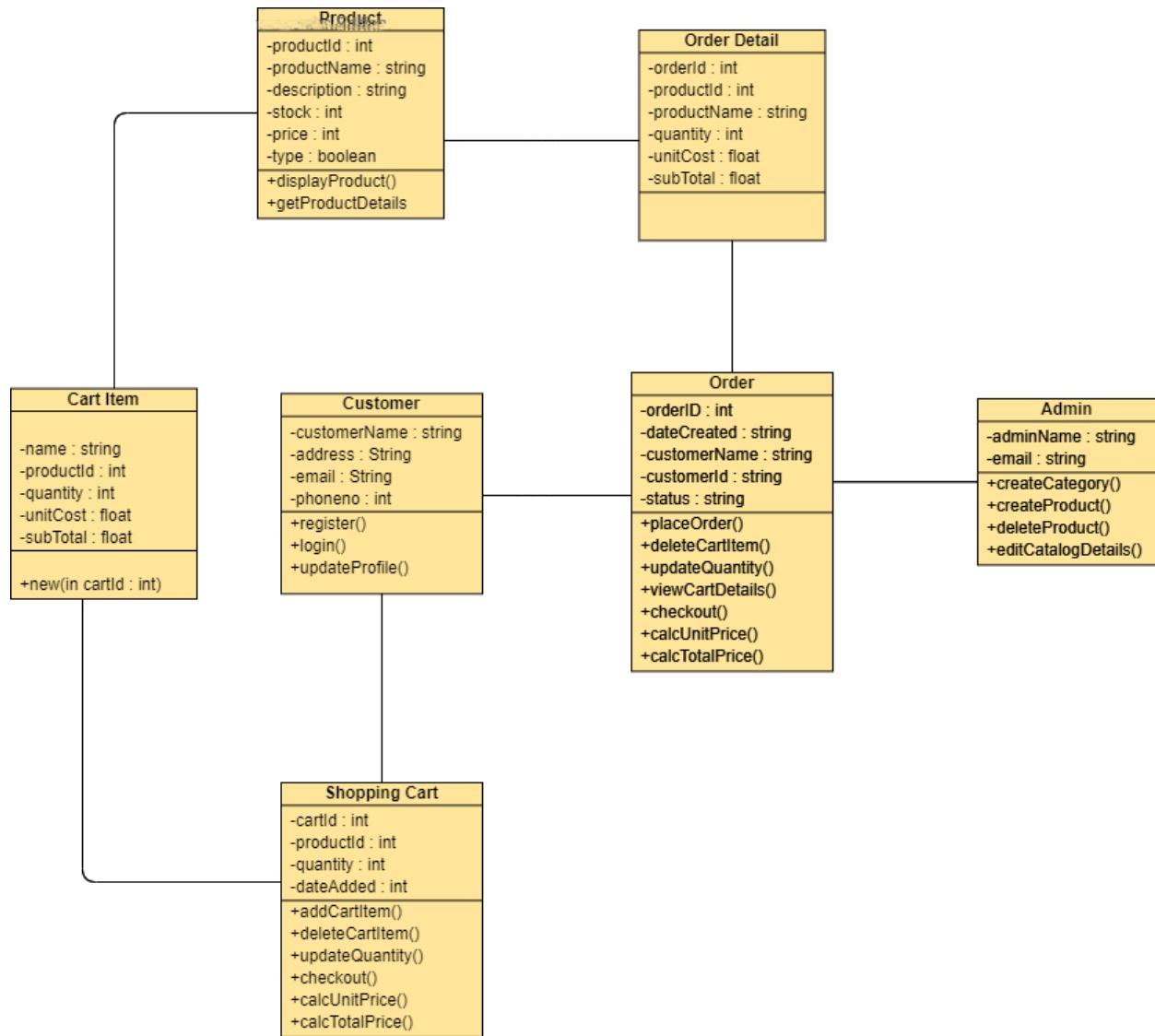


Figure 4: Class Diagram

### 3.1.2 Data Flow Diagram [DFD]

Data Flow Diagram (DFD) shows the flow of the system through a business procedure where it signifies all the inputs, output, data storage and other process which occurs in the website.

The importance of the data flow diagram are:

- Helps to narrate the end lines of the system.
- Can be included as a report in pre-implementing plans.
- Reduces data redundancy.
- Its structural diagram shows the execution of the website.

Notation of Data Flow Diagram (DFD):

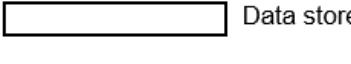
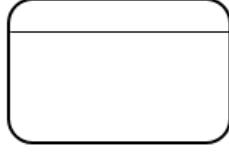
S.N	Notation	Description	Remarks
1.		External entities are executioner, source and actors which informs how they enter and leave in a system.	Both symbols can be used to represent them.
2		They are the channel which the information flow from one place to another. Represent the arrow the moves with the data through it with a name.	
3		They are where all the data are stored of the system. It represents data storage.	Both symbols can be used to represent them.
4		It is a function where changes and transformation of information takes place. It can be change to more detailed form for describing the flow of data.	

Table 6: Notation Diagram of Data Flow Diagram

Admin DFD:

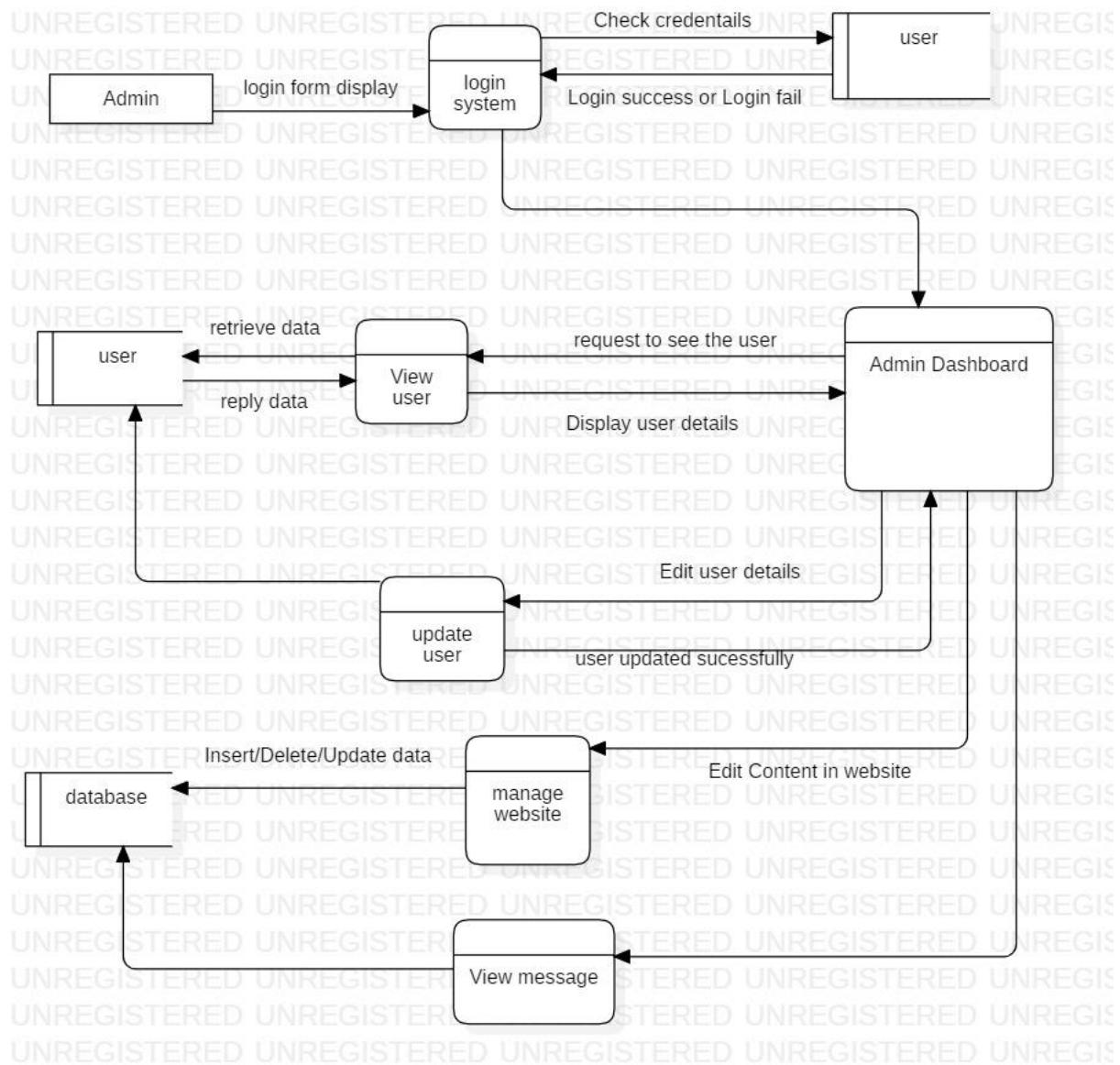


Figure 5: Admin DFD

## 3.2 Behavioral Modelling

Behavioral models describe the internal dynamic aspects of an information system that supports the business processes in an organization. During analysis, behavioral models describe what the internal logic of the processes is without specifying how the processes are to be implemented. Later, in the design and implementation phases, the detailed design of the operations contained in the object is fully specified. (BEHAVIORAL MODELING, 2020)

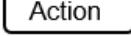
### 3.2.1 Activity Diagram

Activity Diagrams describe how activities are coordinated to provide a service which can be at different levels of abstraction. It describes the flow of methods to another method in a system.

Importance of activity diagram are:

- Helps to finalize the use case diagram.
- Views the flow of function in the system.
- It simplifies and improves complicated methods.
- It describes all logic of the system in particular diagrams.

Notations of activity diagram are:

S.N	Notation	Description	Remarks
1.	 Initial node	The initial point of an activity where it takes place is denoted by initial node.	Both symbols can be used to represent them.
2	 Final Node	The end point of an activity or the system in specific process is denoted by final node.	
3	 Action	Action is represented by round cornered rectangle where an activity is been implementing an action with the object.	
4	 Swim lane	It denotes the grouping of activities done by the same actor in a diagram.	
5	 Decision and End Selection/Iteration	It denotes the decision made with an alternative path of flow.	
6	 Send Signal	It denotes that a signal is send to a receiving activity.	It doesn't wait for any responses from receiver, it ends and passes the control to the next.

7		Control flow show the flow of an object from one activity to another activity (action).	-
8		Splits a single activity flow into two concurrent activities. Symbolized with multiple arrowed lines from a join.	A fork node is a control node that splits a flow into multiple concurrent flows.
9		Combines two concurrent activities and re-introduces them to a flow where only one activity occurs at a time. Represented with a thick vertical or horizontal line.	A join node has multiple incoming edges and one outgoing edge.
10		Demonstrates the acceptance of an event. After the event is received, the flow that comes from this action is completed.	-
11		Allows the diagram creators or collaborators to communicate additional messages that don't fit within the diagram itself. Leave notes for added clarity and specification.	A comment carries no semantic force, but may contain information that is useful to a modeler.
12		Time event flows when the time expression is true, while event signal flows when an event occurs. An hourglass symbol can be used to represent the acceptance of a time event.	-

Table 7: Notation of activity diagram

Activity diagram of login and registration:

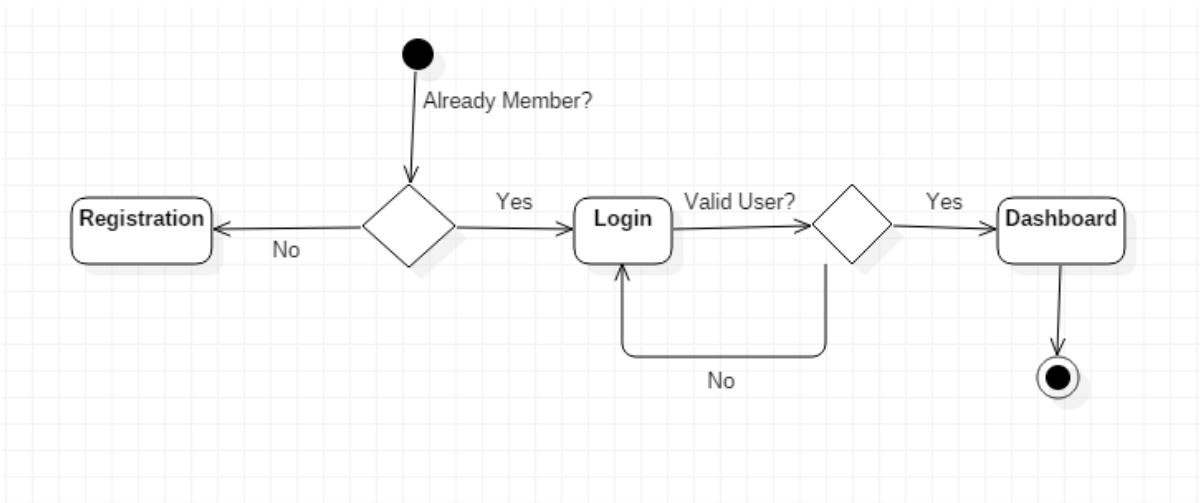


Figure 6: Activity Diagram of login and registration

### 3.2.2 Sequence Diagram

Sequence Diagrams are interaction diagrams that detail how operations are carried out. They capture the interaction between objects in the context of a collaboration. Sequence Diagrams are time focus and they show the order of the interaction visually by using the vertical axis of the diagram to represent time what messages are sent and when. (Sequence Diagram, 2020)

The importance of Sequence diagram are:

- It shows the reaction with object and its operations.
- It helps to plan and understand the function of object for future reference.
- It helps to adjust the operation of the system.
- It identifies problem, interface and architectural problems.

Notations of sequence diagram are:

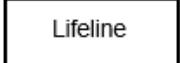
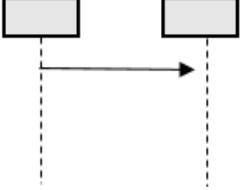
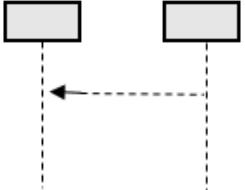
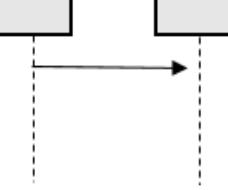
S.N	Notation	Description	Remarks
1.	 Life Line	It denotes singular participant in activity.	
2	 Synchronous Message	It denotes that a sender is waiting for a data to return from the receiver. Message request is denoted by solid line with arrow.	
3	 Response Message	It denotes communication between lifelines of activity, which means to return the information back to the sender of a relative message.	
4	 Asynchronous Communication	It denotes that none is waiting for any return message. It is shown by solid line arrow.	
5	 Recursive message	It denote communication between lifelines of activity that illustrates the message of request of the same life line	

Table 8: Notation of Sequence Diagram

Sequence diagram of login system:

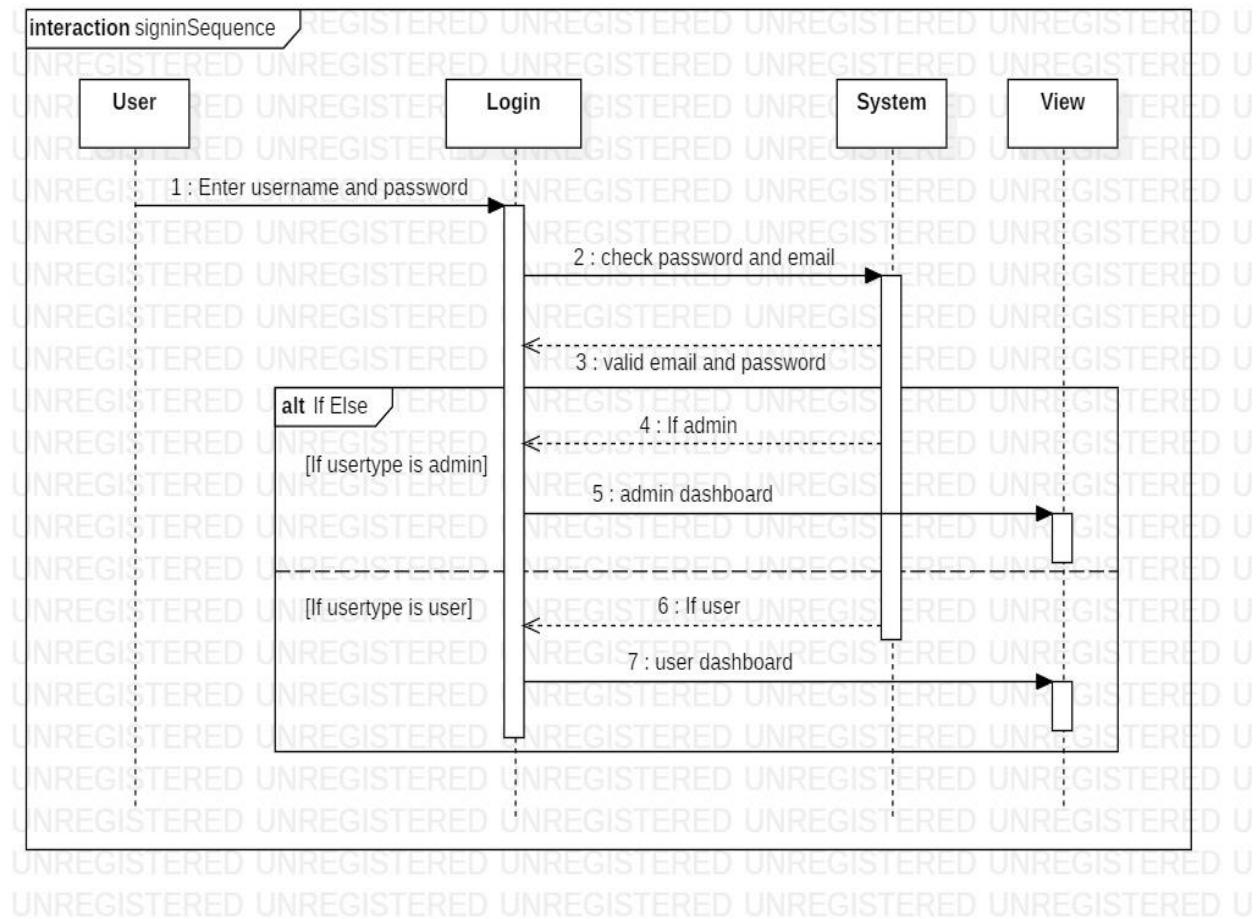


Figure 7: Sequence Diagram of login system

### 3.3 Database Modelling

Database modelling (data modelling) is the process of creating a data model for the data to be stored in a Database. This data model is a conceptual representation of Data objects, the associations between different data objects and the rules. Data modeling helps in the visual representation of data and enforces business rules, regulatory compliances, and government policies on the data. Data Models ensure consistency in naming conventions, default values, semantics, security while ensuring quality of the data. (Data Modelling, 2020)

#### 3.3.1 ER Diagram

ER (Entity Relationship) diagram is a pictorial diagram illustrating the relationship between class, objects, function, etc. as per the requirement of the system and given by the client.

Importance of ER diagram:

- Object and attributes can be determined.
- Show the relation with one entity to another.
- It shows the number of relations implemented in the entity.

*Table 9: ER Diagram*

### 3.3.2 Data Dictionary

It is a set of data and information which contains metadata such as table name, column name, attributes, and related information.

S.N	Table Name
1	Admin Table
2	Customer Table
3	Product Table
4	Order Table
5	Shopping Cart Table
6	Order Details Table
7	Cart Item Table

Table 10: Table Names

#### Admin Table

Column Name	Type	Length	Nullable	Key
ID	Int	10	Not nullable	Primary Key
First_name	Varchar	255	Not nullable	
Last_name	Varchar	255	Not nullable	
Email	Varchar	255	Not nullable	
Password	Varchar	100	Not nullable	

Table 11: Table Names

*Table 12: Data Dictionary*

### 3.4 UI Modelling

The process of developing front end layout of the website or an application before implementing in the back end of the system is called UI (User Interface) modeling. UI is very important nowadays as every people trust in a friendly user interface which help them to use it in an easy and smooth order. As my project is a website it is the key component of the project. Here, I have designed my layout in an external application called Balsamic.

#### 3.4.1 Prototyping

Prototype is rough sketch of the layout of the front end design of any application or a website to sort out future problems. It helps to show the client how the system work in a user interface module.

A Web Page

https://www.vegetabledeliverysystem.com

Vegetable Store

HOME ABOUT CONTACT  LOGIN/SIGNUP

Name

Enter your email

Enter your phone number

Enter your address

Enter password

Confirm password

Create Account



Figure 6: Registration Form

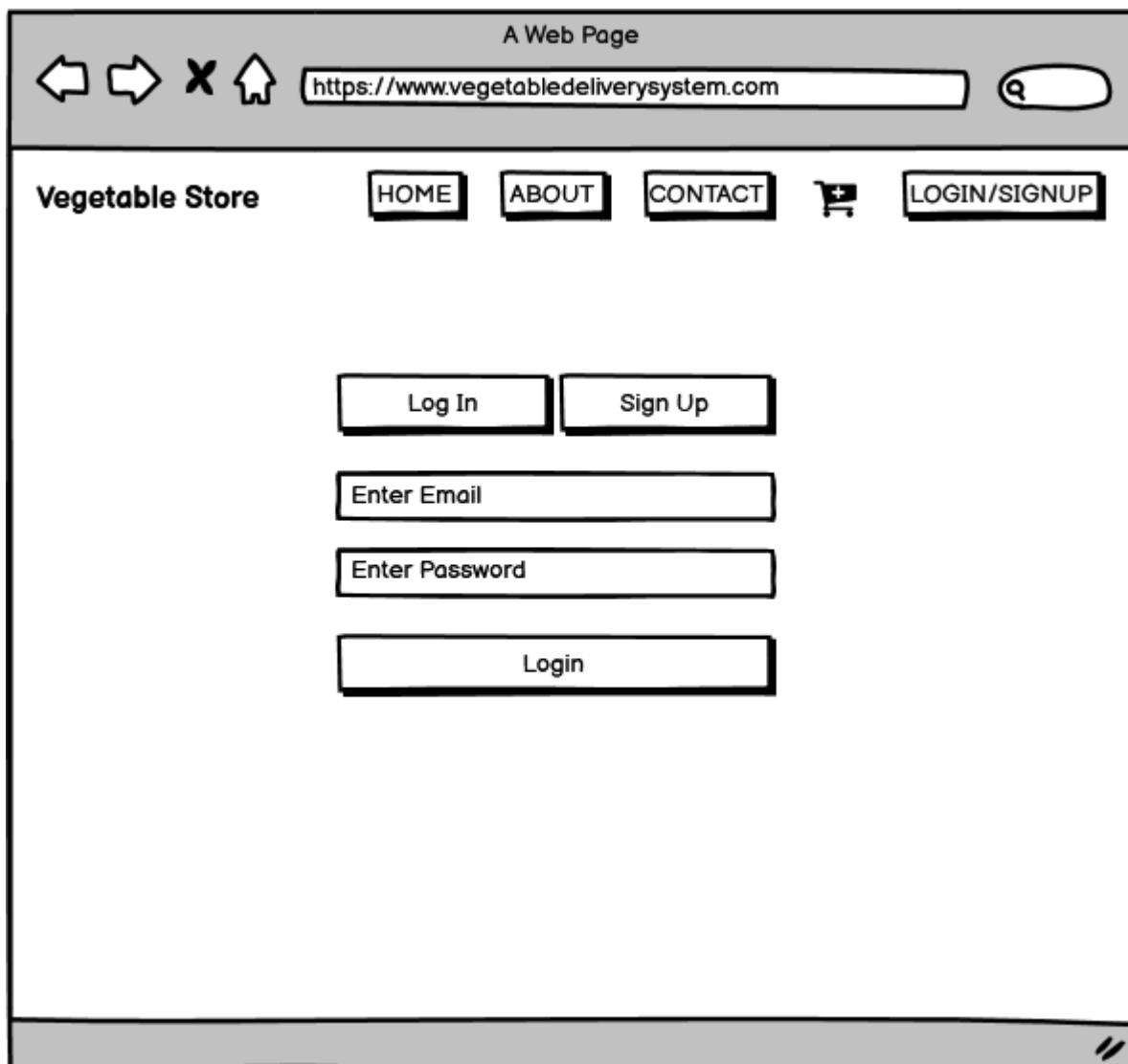


Figure 7: Login Form

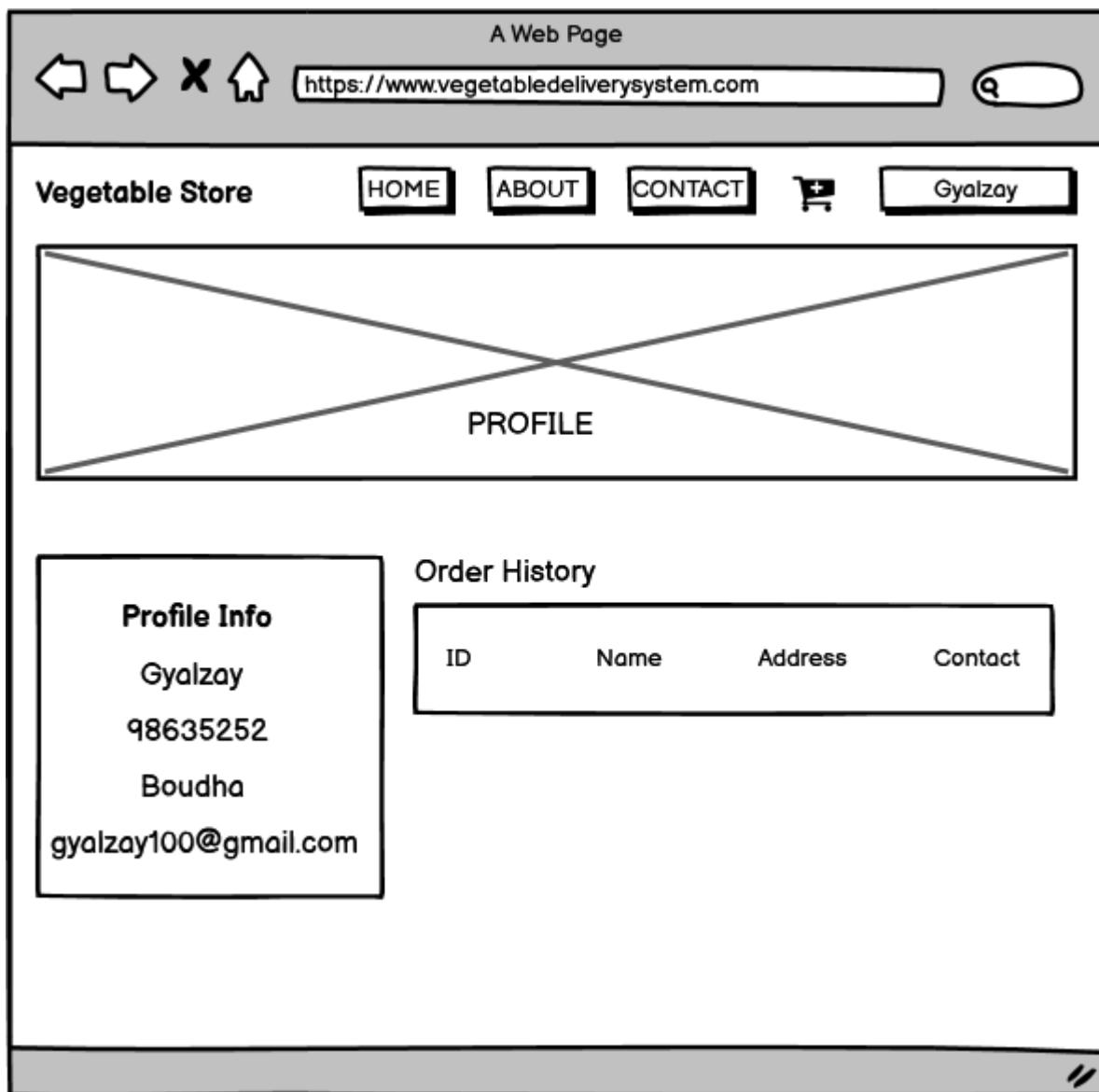


Figure 8: User's Profile

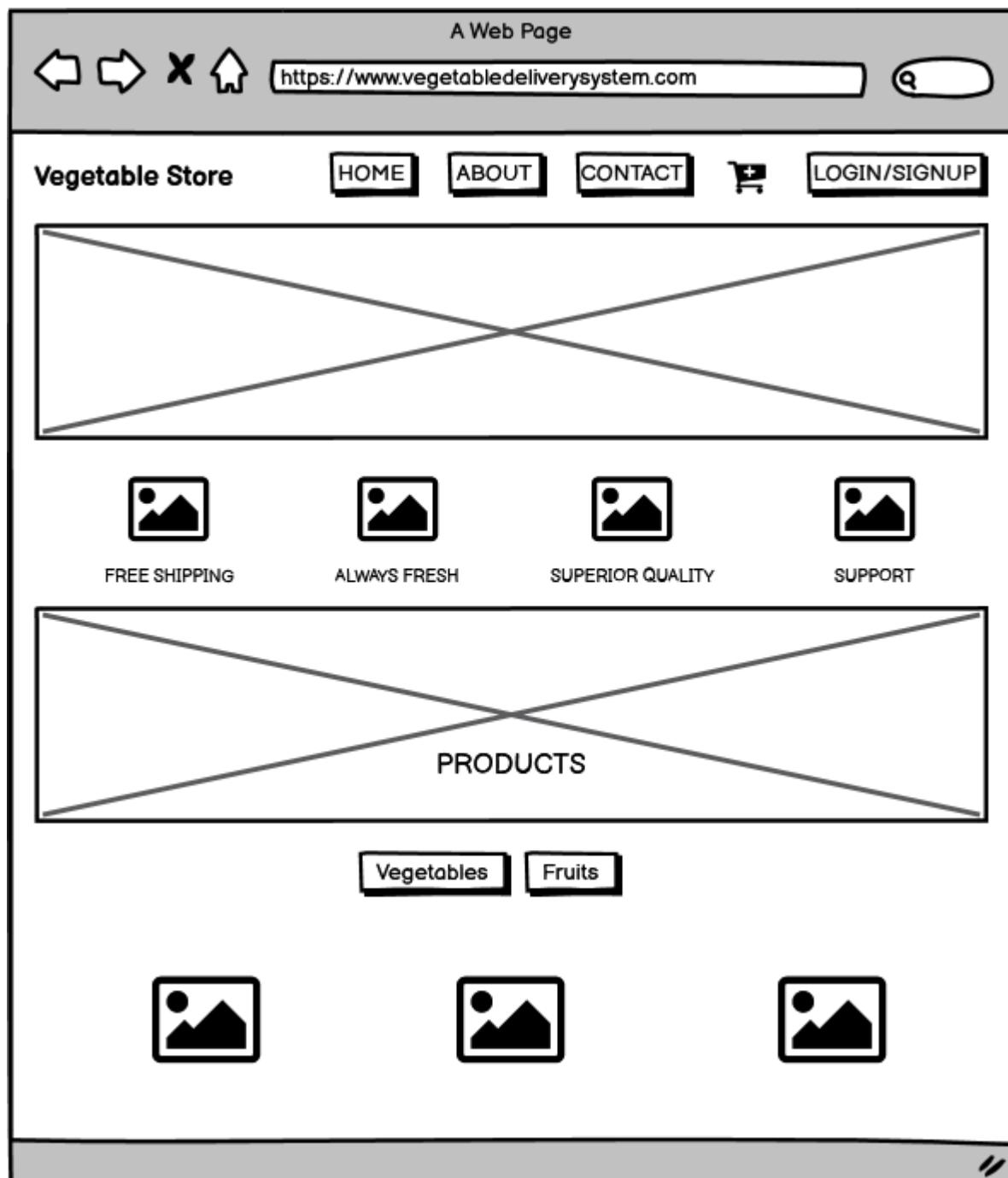


Figure 9: User's Homepage

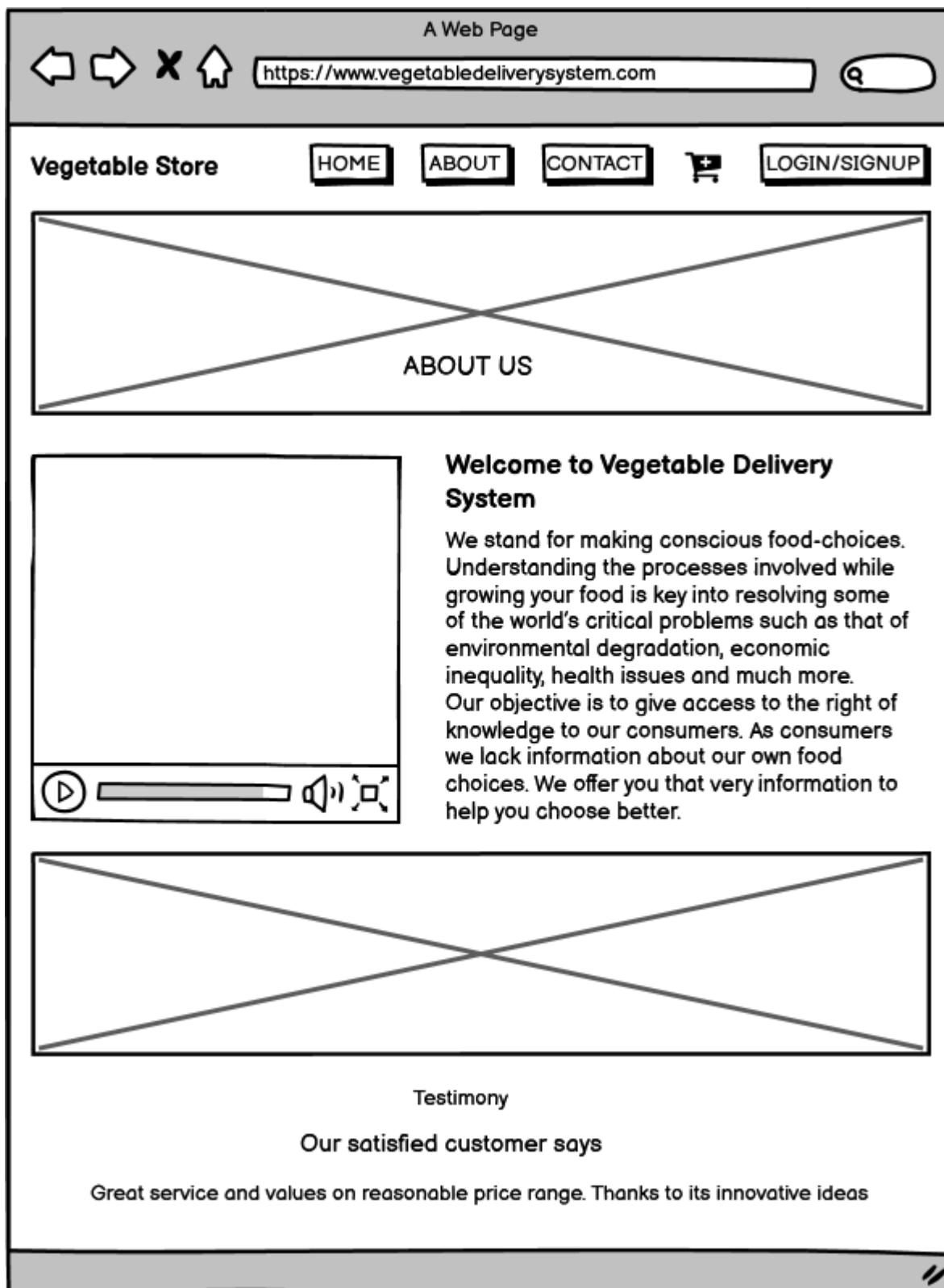


Figure 10: About Us

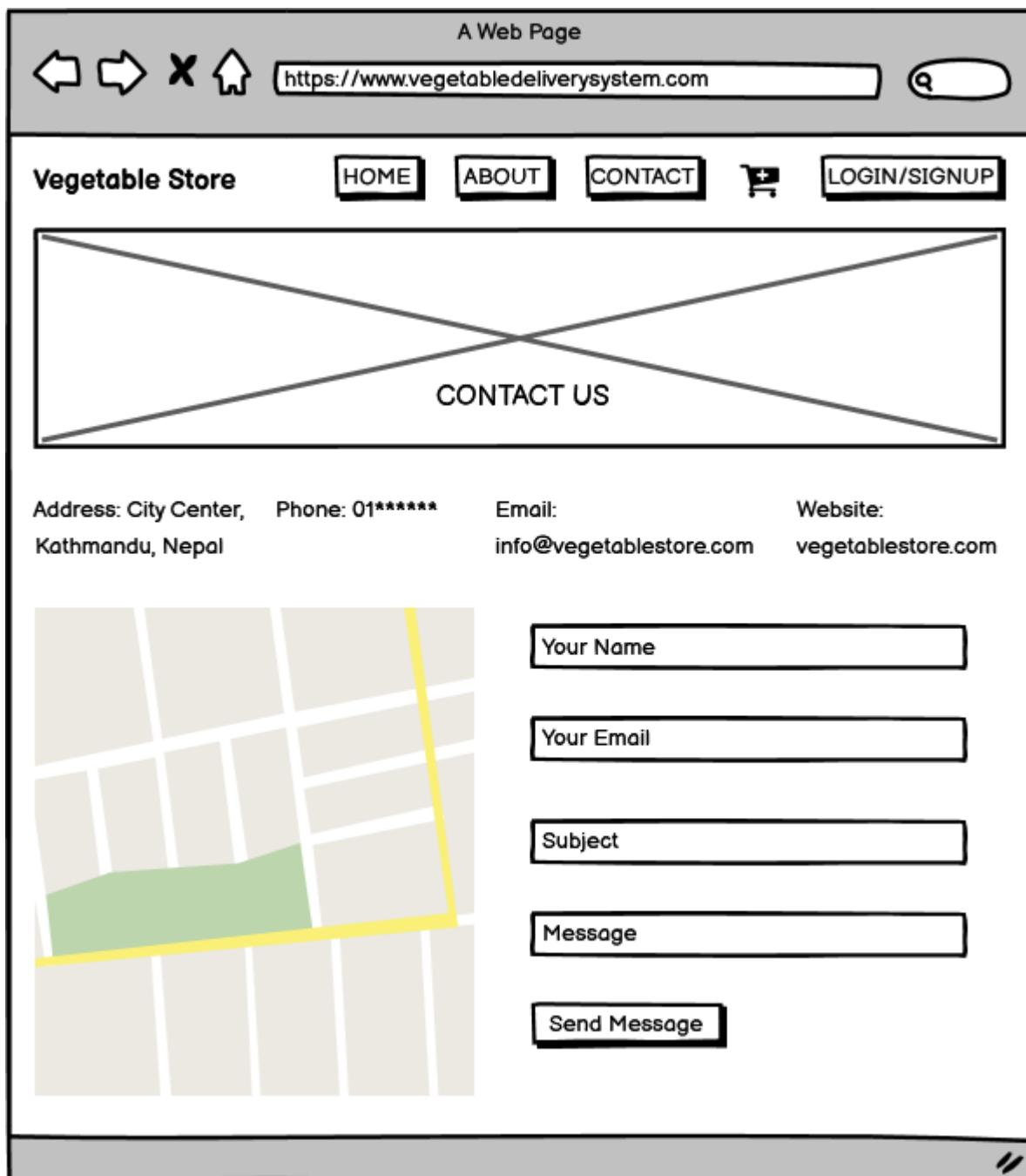


Figure 11: Contact Page

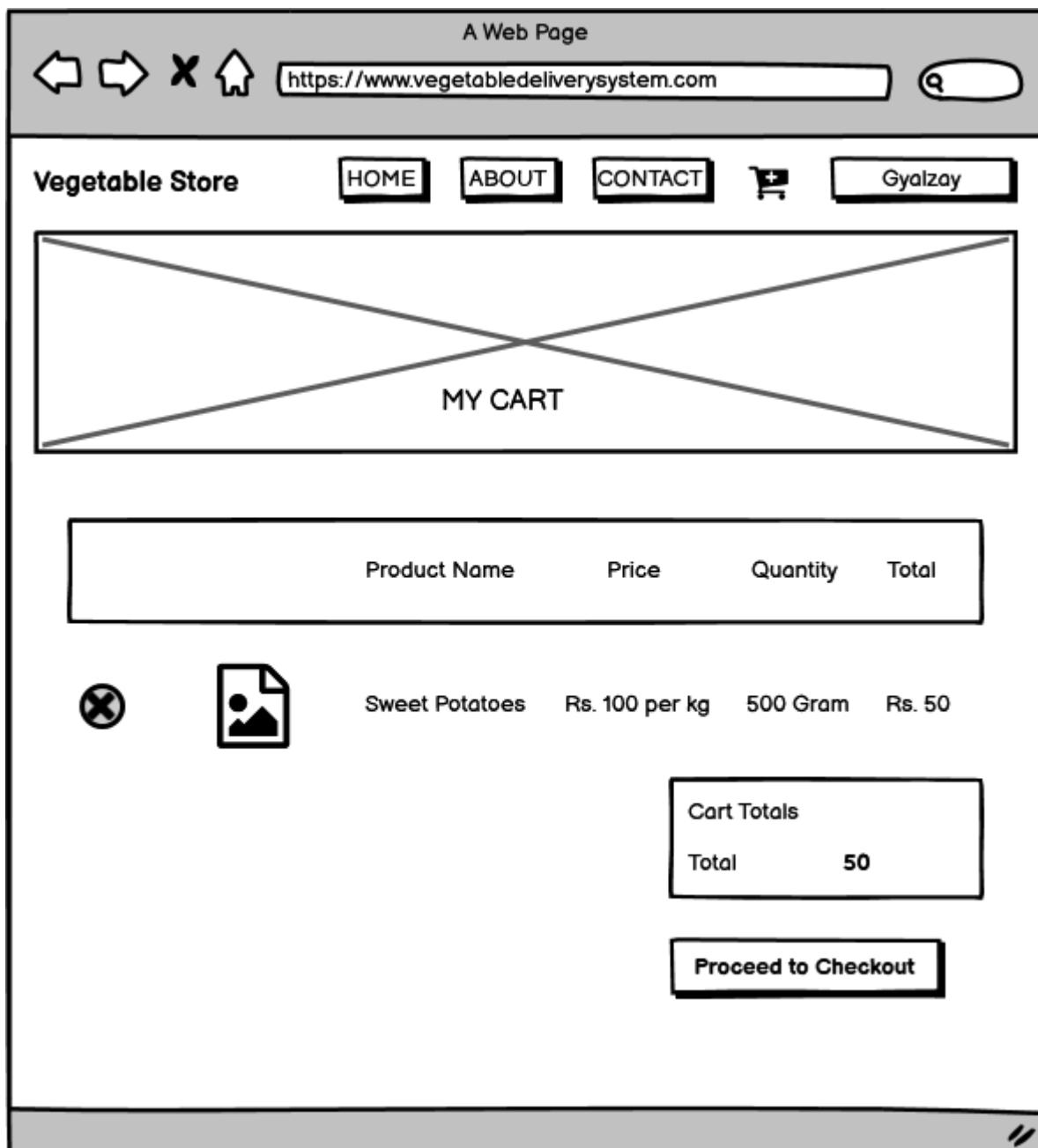


Figure 12: Cart Details

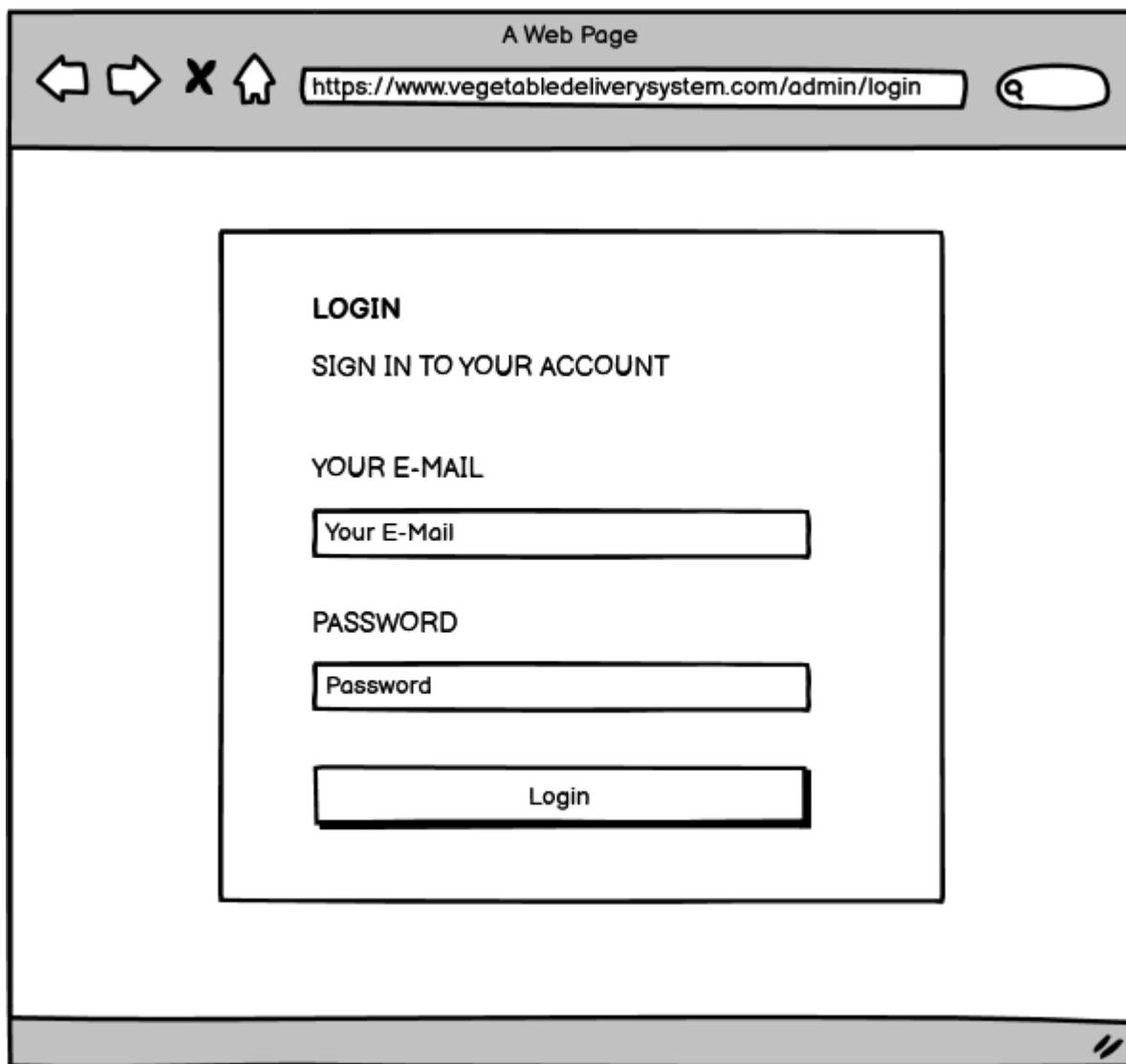


Figure 13: Admin Login Page

A Web Page

[https://www.vegetabledeliverysystem.com/admin/profile](#)

Vegetables Delivery Dashboard  First Name Last Name

CONTENT

Categories  
Products  
Orders  
Testimonials  
Messages

SETTINGS

Manage Access

Edit Profile

First Name   
Last Name   
Email   
Language

 SAVE

Figure 14: Admin Profile

A Web Page

[https://www.vegetabledeliverysystem.com/admin/profile](#)

Vegetables Delivery Dashboard  First Name Last Name

CONTENT

Categories  
Products  
Orders  
Testimonials  
Messages

SETTINGS

Manage Access

Edit Password

Password   
Password Confirmation

SAVE

Figure 15: Admin Password Change

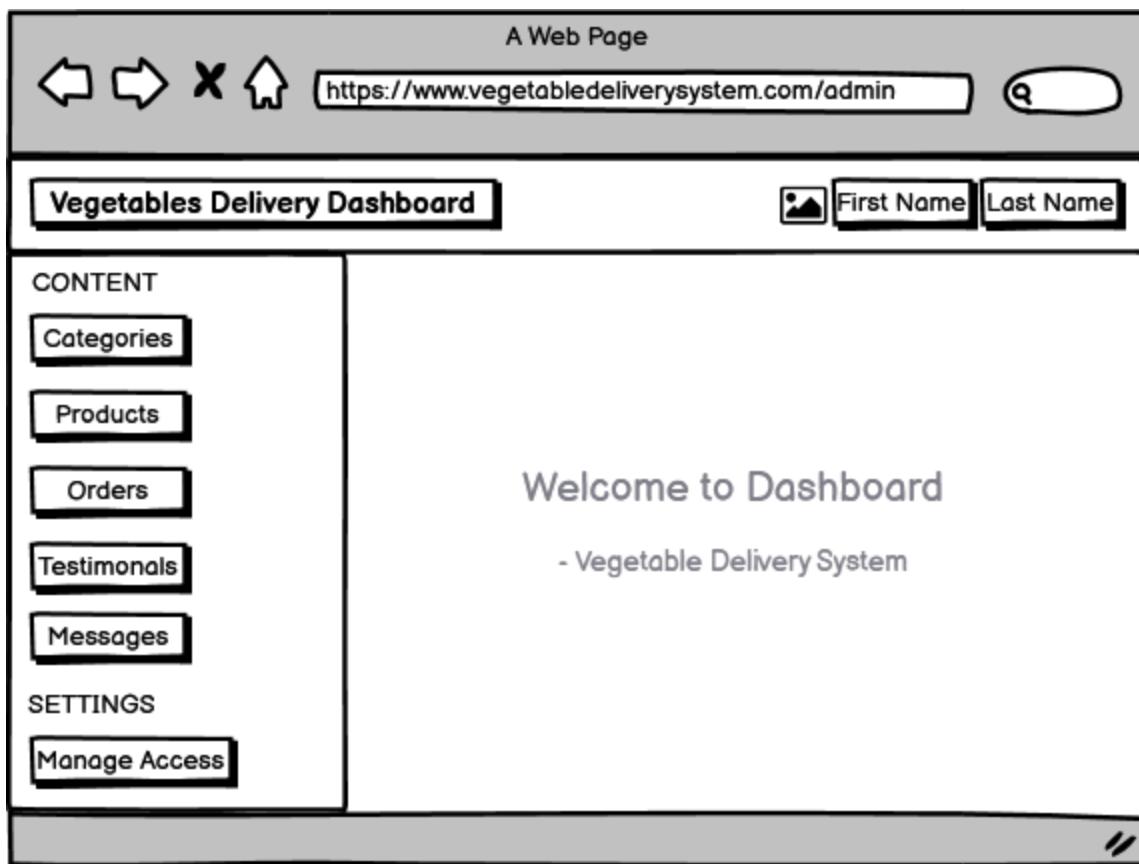


Figure 16: Admin Dashboard

A Web Page

https://www.vegetabledeliverysystem.com/admin

Vegetables Delivery Dashboard

First Name Last Name

CONTENT

Categories Products Orders Testimonials Messages

SETTINGS

Manage Access

Categories

+ New Category

search Search 10 ▾

ID	Category	Category Slug		
1	Vegetables	Vegetables	<input type="checkbox"/>	<input checked="" type="checkbox"/>
2	Fruits	Fruits	<input checked="" type="checkbox"/>	<input type="checkbox"/>

Displaying items from 1 to 2 of total 2 items

Figure 17: Manage Category

A Web Page  
https://www.vegetabledeliverysystem.com/admin

## Vegetables Delivery Dashboard

First Name Last Name

CONTENT

- Categories
- Products
- Orders
- Testimonials
- Messages

SETTINGS

- Manage Access

### Product

+ New Product

search Search 10 ▾

ID	Title	Product Slug	Category	Stock	Price	New
1	Cabbage	Cabbage	Vegetables	1000	85	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>
2	Mango	Mango	Fruits	120	210	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>

Displaying items from 1 to 2 of total 2 items

Figure 18: Manage Products

A Web Page  
https://www.vegetabledeliverysystem.com/admin

## Vegetables Delivery Dashboard

First Name Last Name

CONTENT

- Categories
- Products
- Orders
- Testimonials
- Messages

SETTINGS

- Manage Access

### Orders

search Search 10 ▾

ID	Order	Name	Address	Contact	Order Approve
1	7OMAJYU	Gyalzen Sherpa	Boudha	9876543235	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>

Displaying items from 1 to 1 of total 1 item.

Figure 19: Manage Orders

A Web Page

https://www.vegetabledeliverysystem.com/admin

Vegetables Delivery Dashboard First Name Last Name

CONTENT

Categories Products Orders Testimonials Messages

SETTINGS

Manage Access

Testimonials + New Testimonial

search Search 10 ▾

ID Name Message

Could not find any items Try changing the filter or add a new one + New Testimonial

This screenshot shows the 'Manage Testimonials' section of the admin dashboard. It includes a sidebar with links for Categories, Products, Orders, Testimonials, and Messages. The main area has a search bar, a table header with columns for ID, Name, and Message, and a message stating 'Could not find any items' with a note to change the filter or add a new one.

Figure 20: Manage Testimonials

A Web Page

https://www.vegetabledeliverysystem.com/admin/messages

Vegetables Delivery Dashboard First Name Last Name

CONTENT

Categories Products Orders Testimonials Messages

SETTINGS

Manage Access

Message + New Testimonial

search Search 10 ▾

ID Name Email Subject Message

Could not find any items Try changing the filter or add a new one

This screenshot shows the 'Manage Messages' section of the admin dashboard. It includes a sidebar with links for Categories, Products, Orders, Testimonials, and Messages. The main area has a search bar, a table header with columns for ID, Name, Email, Subject, and Message, and a message stating 'Could not find any items' with a note to change the filter or add a new one.

Figure 21: Manage Messages

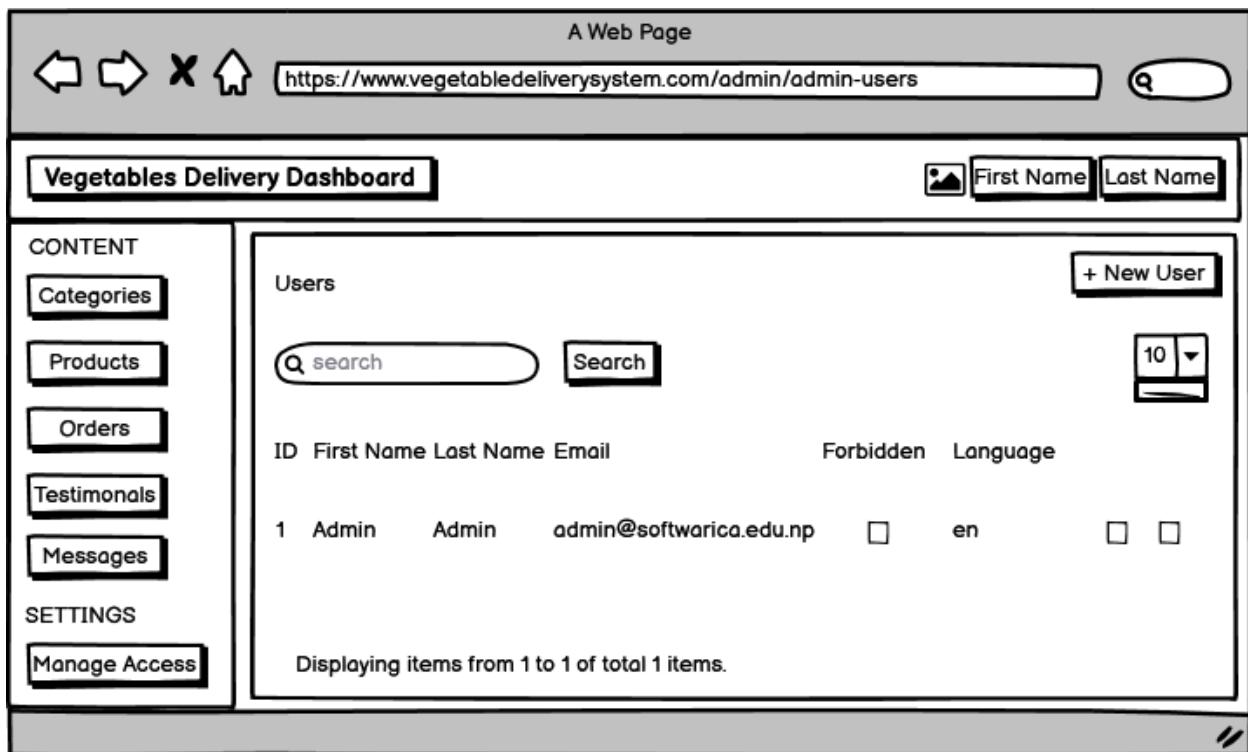


Figure 22: Manage Access

## Chapter 4: Implementation

### Introduction

In this stage, all the information gathered before in the previous stages are implemented. We implement all the function for the project to be developed. We start the implementation process by creating the site from the back end of the topic.

### Tools used for project

Online vegetables and fruits is a web based application where users can buy fresh vegetables and fruits which can be delivered at their home if preferred. To make this project, I have used PHP language, Javascript, HTML, CSS, jQuery. And the framework used to make the website is Laravel. For the database I have used MySQL and for the server I have used Xampp a cross platform application which enables Apache server and database functionality.

### Development Environment

Integrated Development Environment (IDE)

The code editor used to develop this website is Visual Studio Code. Visual Studio Code is refined and optimized for building and debugging modern web and cloud applications.

#### Development platforms

I have windows 10 on my computer system. It has 64-bit operating system, x64-based processor, 8GB RAM

#### Conclusion

The proper implementation of the system is done with the help of the analysis and design methods in the earlier section. To avoid some problems in the future a proper testing of the application will be done in the next section.

## Chapter 5: Testing

### Testing

Software testing is a process to evaluate the functionality of a software application with an intent to find whether the developed software/application met the specified requirements or not and to identify the defects to ensure that the product is defect-free in order to produce the quality product. (Software testing, 2020) For my project, I have used black box and white box testing.

#### Unit Testing (White Box Testing)

Unit Testing or White Box Testing refers to a kind of testing where individual units or components of a software are tested. (Unit Testing Tutorial, 2020) Unit Testing is done to validate that every unit of the software code performs as expected.

## Black Box Testing

Black Box Testing is a software testing method where all the functionalities of the software applications are tested. During the test there is no proper knowledge of the internal code structure, details of the implementations, and the internal parts of the software system. Black Box Testing mainly emphasizes on the input and the output of the system. (Black Box Testing, 2020)

Case Name	Data Passed	Expected	Result
BBT1	Invalid	Fail	Fail

Log In      Sign Up

Dawa Gyalzen Sherpa

fkjadfka]

Please enter valid email

Enter your phone number

Enter your address

Enter password

Confirm password

Create Account

Figure 23: Validation

Case Name	Data Passed	Expected	Result
BBT2	Valid	Pass	Pass

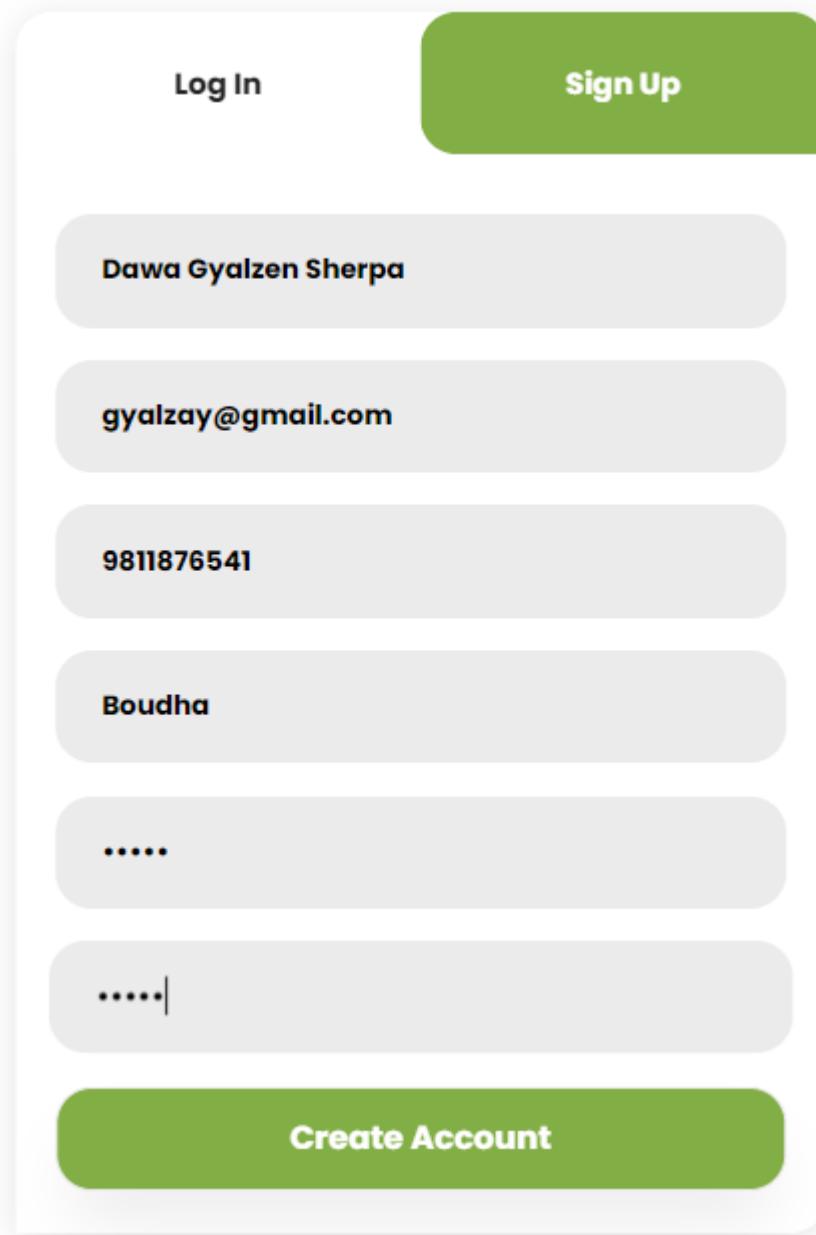
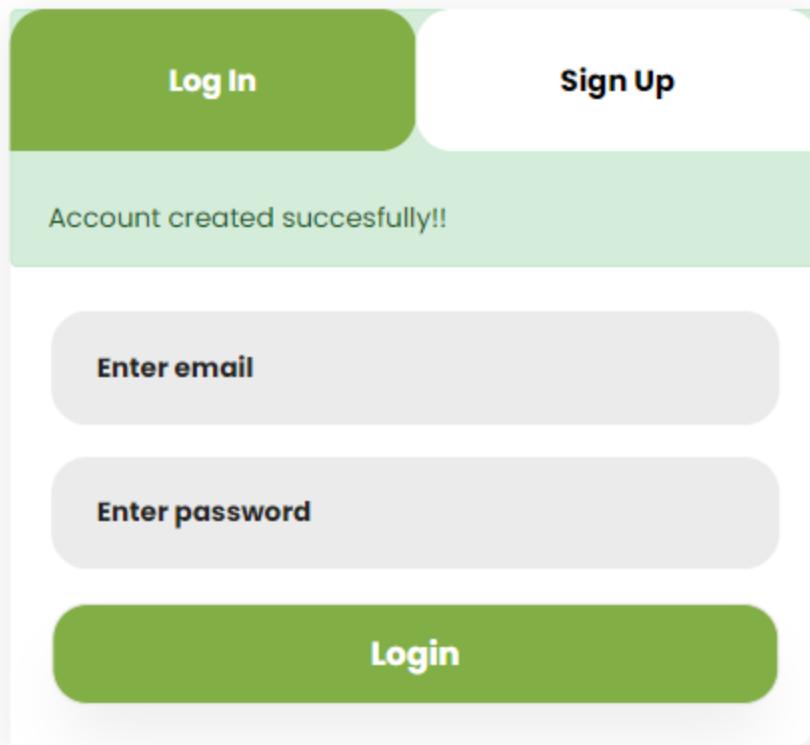


Figure 24: Valid Details



*Figure 25: Account Created Successfully*

Case Name	Data Passed	Expected	Result
BBT3	Valid Info	Successful Login	Pass

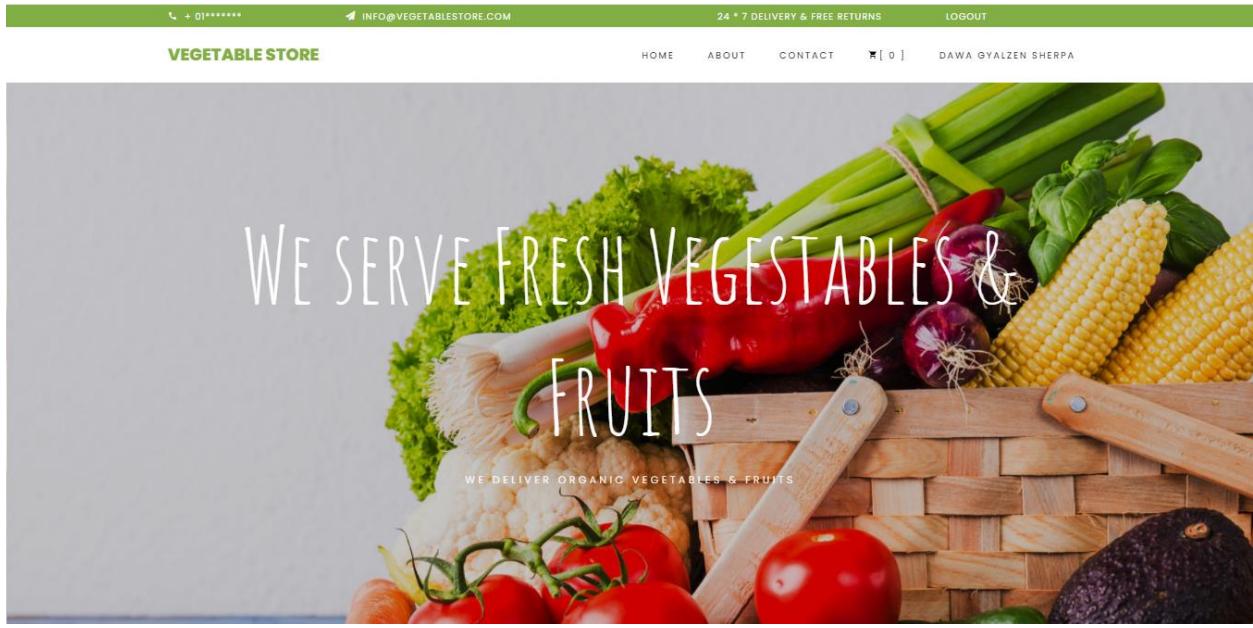


Figure 26: Successful Login

Case Name	Data Passed	Expected	Result
BBT4	Invalid Info	Successful Login	Fail

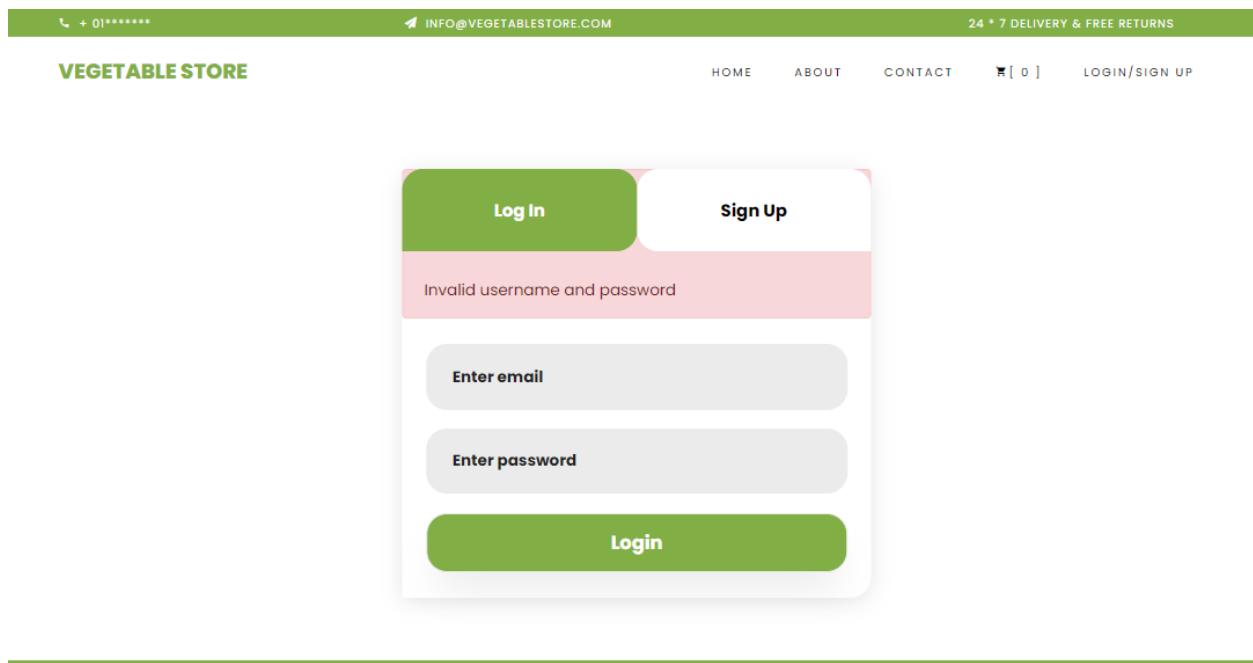


Figure 27: Invalid Login

Case Name	Data Passed	Expected	Result
BBT5	Invalid Info of Admin	Successful Login	Fail

# LOGIN

SIGN IN TO YOUR ACCOUNT

---

YOUR E-MAIL

×

The email field must be a valid email

PASSWORD

>Password

**Login**

Figure 28: Admin Login Validation

Case Name	Data Passed	Expected	Result
BBT6	Valid Info of Admin	Successful Login	Pass

The image shows a login screen with the following elements:

- LOGIN**: The main title at the top.
- SIGN IN TO YOUR ACCOUNT**: Sub-instruction below the title.
- YOUR E-MAIL**: Label for the first input field.
- admin@softwarica.edu.np**: The value entered into the email input field.
- PASSWORD**: Label for the second input field.
- .....**: The value entered into the password input field, represented by dots.
- Login**: A large blue button at the bottom of the form.

Figure 29: Valid Info of Admin

The screenshot shows a web browser window with the URL 127.0.0.1:8000/admin. The title bar reads "Vegetables Delivery Dashboard". On the right, there is a user profile for "Administrator Administrator" with a green profile picture. The left sidebar has a dark blue background with white text. It includes sections for "CONTENT" (Categories, Products, Orders, Testimonials, Messages) and "SETTINGS" (Manage access). The main content area has a light gray background. At the top, it says "Welcome to Dashboard" with a subtitle "— Vegetable Delivery System".

Figure 30: Admin Login Successful

Case Name	Data Passed	Expected	Result
BBT7	Add to Cart	Successful	Pass



### Sweet Potatoes

Rs. 100 per kg

Fresh juicy sweet potatoes..

1 kg available

250 GM ▾

Add to Cart

Figure 31: Add to Cart

Product name	Price	Quantity	Total
	Sweet Potatoes Rs. 100 Per KG	250 Gram	Rs. 25

◀ ▶

Cart Totals

TOTAL	<b>25</b>
-------	-----------

**Proceed to Checkout**

Figure 32: Add to Cart Successful

Case Name	Data Passed	Expected	Result
BBT8	Checkout	Successful	Pass

**Billing Details**

Full Name

Cart Total

TOTAL **25**

Address

Contact

Payment Method

Cash On Delivery

**Place an order**

Figure 33: Checkout

Vegetables Delivery Dashboard

#	ID	Order	Name	Address	Contact	Order approve
<input type="checkbox"/>	2	86NNDY60GAHW9M0	Dawa Gyalzen Sherpa	Boudha	9876543223	

Displaying items from 1 to 1 of total 1 items.

Figure 34: Checkout Order Successful

Case Name	Data Passed	Expected	Result
BBT9	Message to Admin	Successful	Pass

① 127.0.0.1:8000/contact.html

VEGETABLE STORE

HOME ABOUT CONTACT [ 0 ] DAWA GYALZEN SHERPA

Address: City Center, Kathmandu, Nepal

Phone: 01\*\*\*\*\*

Email: info@vegetablestore.com

Website: vegetablestore.com

Kathmandu 44600  
View larger map

HATTISAR GYANESHWOR KAMALADI PUTALISADAK BHRIKUTI MANDAP

Dawa Gyalzen Sherpa

gyalzay@gmail.com

Fast Delivery

Maintain fast Delivery

Send Message

Figure 35: Message to Admin

The screenshot shows a web-based administrative dashboard for a vegetable delivery service. The left sidebar has a dark blue background with white text and icons. It includes sections for 'CONTENT' (Categories, Products, Orders, Testimonials, Messages), which is currently selected and highlighted in blue. Below that is a 'SETTINGS' section with 'Manage access'. The main content area has a light gray background. At the top, it says 'Vegetables Delivery Dashboard' and shows a user profile for 'Administrator Administrator'. A search bar and a dropdown menu are also at the top. The main area is titled 'Message' and contains a table with one row of data. The table columns are '#', 'ID', 'Name', 'Email', 'Subject', and 'Message'. The data row shows: # 1, Name Dawa Gyalzen Sherpa, Email gyalzay@gmail.com, Subject Fast Delivery, Message Maintain fast delivery. There is a small red icon next to the message. Below the table, a message says 'Displaying items from 1 to 1 of total 1 items.'

Figure 36: Message from Users

Case Name	Data Passed	Expected	Result
BBT10	Adding Product	Added successful	Pass

The screenshot shows a website for a vegetable store. At the top, there is a navigation bar with links for 'HOME', 'ABOUT', 'CONTACT', and a green button labeled 'DAWA GYALZEN SHERPA'. Below the navigation is a large, colorful image of various vegetables and fruits. The main content area features two products: a mango and a papaya. Each product has a 'New' badge. The mango is labeled 'MANGO' and 'Rs. 210 per kg'. The papaya is labeled 'PAPAYA' and 'Rs. 325 per kg'. There are also tabs for 'Vegetables' and 'Fruits'.

Figure 37: Before Updating Products

**Vegetables Delivery Dashboard**

 Administrator Administrator ▾

**CONTENT**

-  Categories
-  Products
-  Orders
-  Testimonials
-  Messages

**SETTINGS**

-  Manage access

**+ New Product**

Title	<input type="text" value="Banana"/> 
Select Category	<input type="text" value="Choose Category"/> 
  <b>Images</b> (max no. of files: 1 files)	
Choose Images	
Stock	<input type="text" value="20000"/> 

**Vegetables Delivery Dashboard**

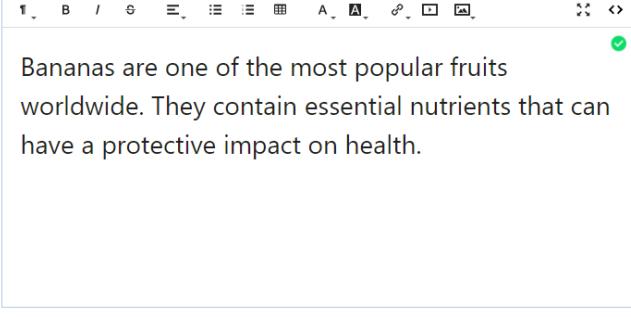
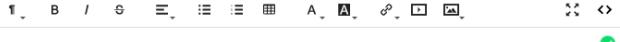
 Administrator Administrator ▾

**CONTENT**

-  Categories
-  Products
-  Orders
-  Testimonials
-  Messages

**SETTINGS**

-  Manage access

Stock	<input type="text" value="20000"/> 
Description	 
Bananas are one of the most popular fruits worldwide. They contain essential nutrients that can have a protective impact on health.	
Price	<input type="text" value="175"/> 
<input checked="" type="checkbox"/> New	

**Save**

Figure 38: Adding Product by Admin

**Vegetables Delivery Dashboard**

#	ID	Title	Product Slug	Category	Stock	Price	New
1	Sweet Potatoes	sweet-potatoes	Vegetables	750	100	<input checked="" type="checkbox"/>	
2	Mango	mango	Fruits	19750	210	<input checked="" type="checkbox"/>	
3	Papaya	papaya	Fruits	10000	325	<input checked="" type="checkbox"/>	
4	Tomatoes	tomatoes	Vegetables	3500	125	<input checked="" type="checkbox"/>	
5	Pumpkin	pumpkin	Vegetables	12000	120	<input checked="" type="checkbox"/>	
8	Banana	banana	Fruits	20000	175	<input checked="" type="checkbox"/>	

Displaying items from 1 to 6 of total 6 items.

Figure 39: Product Added On Product Details

**VEGETABLE STORE**

HOME ABOUT CONTACT DAWA GYALZEN SHERPA

Vegetables



MANGO

Rs. 210 per kg



PAPAYA

Rs. 325 per kg



BANANA

Rs. 175 per kg

Figure 40: Updated Products

## Identification of Risk

Risk management is the operation of controlling, identifying and evaluating risks to the application or program. These risks can damage the organization's data, information, prestige, etc. Nowadays, people are more concerned about their personal data and billing details. Those risk impact on management, financial status and legal liabilities of the organization directly or indirectly. Risk management is usually evaluated in three areas like risk consequences, probability and affect due to risk.

## Risk management table

This table evaluates the risk that may damage the program in the future days. This is to maintain the application for longer period of time with no attacks. Organization may lose their prestige, integrity, and etc. due to lack of maintenance.

Likelihood	Value
Low	1
Medium	2
High	3

Consequences	Value
Very Low	1
Low	2
Medium	3
High	4
Very High	5

Topic	Likelihood	Consequences	Impact	Remarks
Hardware failure	3	3	9	Data cloud backup. Repairing the hardware. Get a proper quality hardware
Loss of digital data	1	4	4	Back up data frequently. Two flash drives.
No access of electricity	1	5	5	Power supply devices should be installed.
Malware	2	4	8	Using of anti-malware software and antivirus.
Incorrect input of data	3	3	9	Testing the inputted data frequently.
Hard disk failure	1	5	5	Preparing the duplicate of local hard disk before initiating the project.
Scope is poorly defined	2	3	6	Holding meetings with employee and users. Conducting requirement specifications walkthroughs and technical inspections.
Natural Calamities	1	5	5	Security measures
Personal Injury	2	4	8	First Aid training Medical attention

### Other issues

Other issues are all the other unimplemented issues that may occur to the project due to lack of time, resources and technology. Different problems may occur even if it is not expected in the technical world.

### Future Works

Many of the functions was not been able to add to the project due to lack of time and resources. Some of the features that may be added in the near future are as follows.

- Internet payment, E-banking could be implemented via various third party like PayPal in the coming days for simplicity of payment to the users.
- Various kind of time details of same course can be included.
- Search function by categorizing the classes was complicated may be useful while updating the system.

### Configuration Management

It is a method for maintaining execution, cost, physical quality and plan similarly reliability in data for the life duration of the project. It also help to maintain complexity of the system.

File management of the project is effectively followed. For no threat to occur in any situation the file are manage in corporate order. It contains all the suitable document related with the organization of the system.

### User training

User training is necessary to people who aren't familiar to website an online booking. All the necessary interface are user friendly and easy to use. All the system of website are very simple made for student over 16 years so would not be a problem to book a course. If any of the user faces any difficulty they can concern to the institute itself and can help the needy. Like normal website, user can easily find booking if they are logged in. Even for the not user they can easily access register page and book a course.

#### Limitation

Vegetables and fruit delivery system consist of some limitation which includes online payment, inefficiency in delivering the services on proper time all 365 days, etc

## Conclusion

The project was developed to make an easy way for people to buy fresh vegetables and fruits online. And as for the farm it was developed to help sellers sell their products in a huge quantity which helps them to maximize their profit level as per their performance. The system is simple, user-friendly which requires user to register to the system and can have access to the system by simply login to the system.

The system will function efficiently since it has been developed under the Laravel framework. The main ambition of the project was to help sellers to sell their products online and buyers to buy products as per their needs.

## Appendix

The screenshot shows the homepage of a vegetable store website. At the top, there is a green header bar with the text "INFO@VEGETABLESTORE.COM", "24 \* 7 DELIVERY & FREE RETURNS", and "LOGOUT". Below the header is a navigation bar with links for "HOME", "ABOUT", "CONTACT", a shopping cart icon showing "0", and "DAWA GYALZEN SHERPA". The main visual is a vibrant collage of various fresh fruits and vegetables, including onions, bananas, apples, carrots, bell peppers, and tomatoes. Overlaid on this image is the text "100% FRESH & ORGANIC FOODS" in a large, white, serif font, and below it, "WE DELIVER ORGANIC VEGETABLES & FRUITS" in a smaller, sans-serif font. Below the main image is another header bar identical to the one at the top, featuring the "VEGETABLE STORE" logo, navigation links, and user information. Underneath this second header are four circular icons representing service features: "FREE SHIPPING" (truck icon), "ALWAYS FRESH" (fridge icon), "SUPERIOR QUALITY" (award star icon), and "SUPPORT" (phone icon). Below these icons is a large image of various fresh produce like lettuce, corn, and onions.

Figure 41: Home Page

Vegetables

Fruits

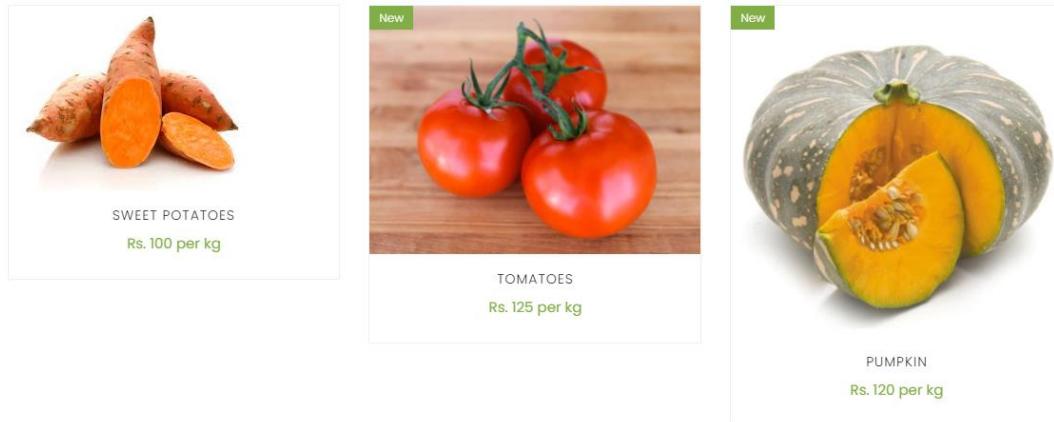
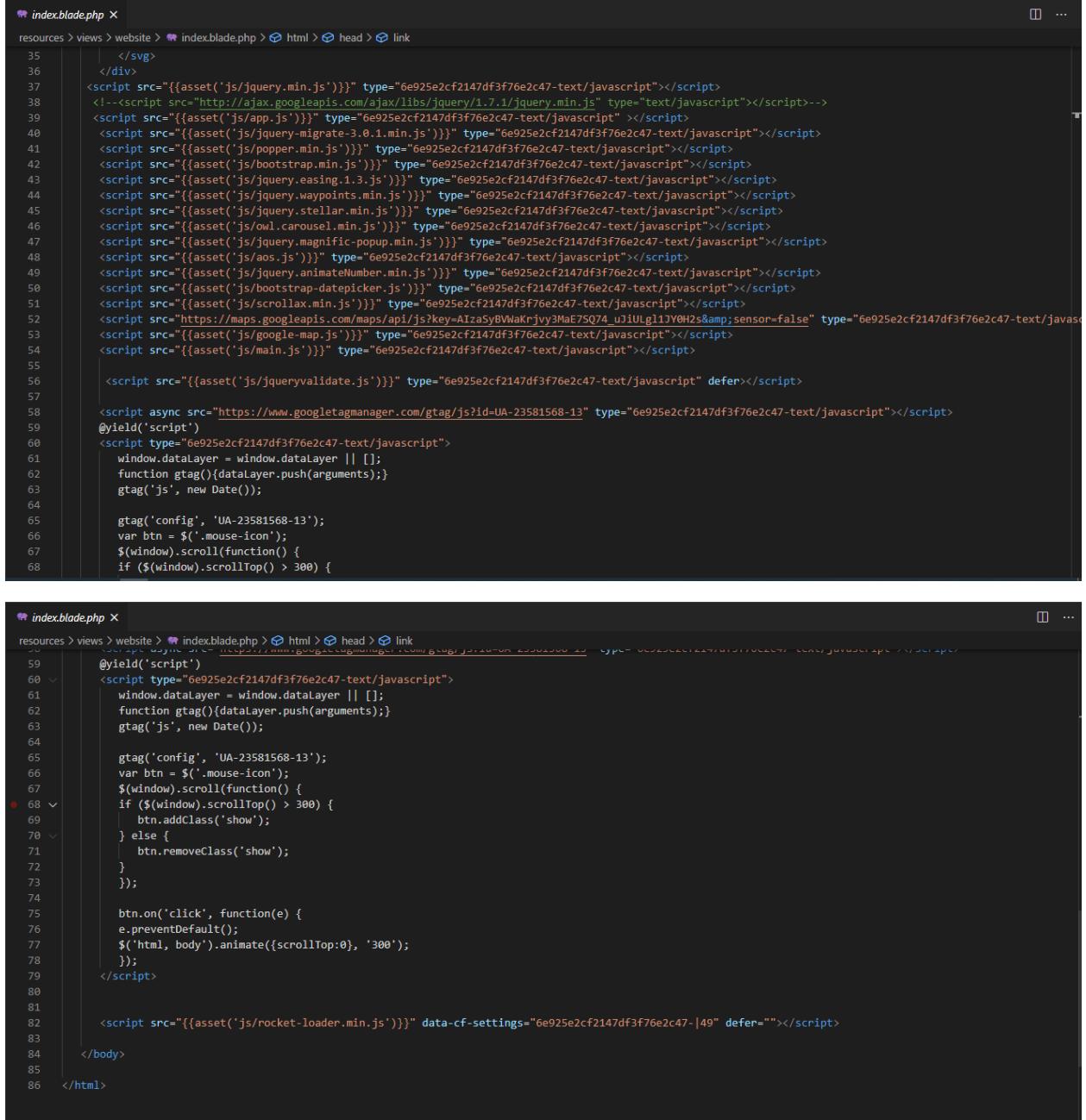


Figure 42: Products Page

```

index.blade.php X
resources > views > website > index.blade.php > html > head > link
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <title>Vegetable Delivery System</title>
5      <meta charset="utf-8">
6      <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
7      <link href="https://fonts.googleapis.com/css?family=Poppins:200,300,400,500,600,700,800&display=swap" rel="stylesheet">
8      <link href="https://fonts.googleapis.com/css?family=Lora:400,400i,700,700i&display=swap" rel="stylesheet">
9      <link href="https://fonts.googleapis.com/css?family=Amatic+SC:400,700&display=swap" rel="stylesheet">
10     <link rel="stylesheet" href="{{asset('css/open-iconic-bootstrap.min.css')}}"/>
11     <link rel="stylesheet" href="{{asset('css/animate.css')}}"/>
12     <link rel="stylesheet" href="{{asset('css/owl.carousel.min.css')}}"/>
13     <link rel="stylesheet" href="{{asset('css/owl.theme.default.min.css')}}"/>
14     <link rel="stylesheet" href="{{asset('css/magnific-popup.css')}}"/>
15     <link rel="stylesheet" href="{{asset('css/aos.css')}}"/>
16     <link rel="stylesheet" href="{{asset('css/ionicons.min.css')}}"/>
17     <link rel="stylesheet" href="{{asset('css/bootstrap-datepicker.css')}}"/>
18     <link rel="stylesheet" href="{{asset('css/jquery.timepicker.css')}}"/>
19     <link rel="stylesheet" href="{{asset('css/flaticon.css')}}"/>
20     <link rel="stylesheet" href="{{asset('css/icomoon.css')}}"/>
21     <link rel="stylesheet" href="{{asset('css/style.css')}}"/>
22     <meta name="csrf-token" content="{{ csrf_token() }} />
23 </head>
24 <body class="goto-here">
25     @include('website.header')
26
27     @yield('body')
28
29
30     @include('website.footer')
31     <div id="ftco-loader" class="show fullscreen">
32         <svg class="circular" width="48px" height="48px">
33             <circle class="path-bg" cx="24" cy="24" r="22" fill="none" stroke-width="4" stroke="#eeeeee" />
34             <circle class="path" cx="24" cy="24" r="22" fill="none" stroke-width="4" stroke-miterlimit="10" stroke="#F96000" />

```



```

resources > views > website > index.blade.php > html > head > link
35 |     </svg>
36 |   </div>
37 |   <script src="{{asset('js/jquery.min.js')}}" type="6e925e2cf2147df3f76e2c47-text/javascript"></script>
38 |   <!--<script src="https://ajax.googleapis.com/ajax/libs/jquery/1.7.1/jquery.min.js" type="text/javascript"></script-->
39 |   <script src="{{asset('js/app.js')}}" type="6e925e2cf2147df3f76e2c47-text/javascript"></script>
40 |   <script src="{{asset('js/jquery-migrate-3.0.1.min.js')}}" type="6e925e2cf2147df3f76e2c47-text/javascript"></script>
41 |   <script src="{{asset('js/popper.min.js')}}" type="6e925e2cf2147df3f76e2c47-text/javascript"></script>
42 |   <script src="{{asset('js/bootstrap.min.js')}}" type="6e925e2cf2147df3f76e2c47-text/javascript"></script>
43 |   <script src="{{asset('js/jquery.easing.1.3.js')}}" type="6e925e2cf2147df3f76e2c47-text/javascript"></script>
44 |   <script src="{{asset('js/jquery.waypoints.min.js')}}" type="6e925e2cf2147df3f76e2c47-text/javascript"></script>
45 |   <script src="{{asset('js/jquery.stellar.min.js')}}" type="6e925e2cf2147df3f76e2c47-text/javascript"></script>
46 |   <script src="{{asset('js/owl.carousel.min.js')}}" type="6e925e2cf2147df3f76e2c47-text/javascript"></script>
47 |   <script src="{{asset('js/jquery.magnific-popup.min.js')}}" type="6e925e2cf2147df3f76e2c47-text/javascript"></script>
48 |   <script src="{{asset('js/aos.js')}}" type="6e925e2cf2147df3f76e2c47-text/javascript"></script>
49 |   <script src="{{asset('js/jquery.animateNumber.min.js')}}" type="6e925e2cf2147df3f76e2c47-text/javascript"></script>
50 |   <script src="{{asset('js/bootstrap-datepicker.js')}}" type="6e925e2cf2147df3f76e2c47-text/javascript"></script>
51 |   <script src="{{asset('js/scrollax.min.js')}}" type="6e925e2cf2147df3f76e2c47-text/javascript"></script>
52 |   <script src="https://maps.googleapis.com/maps/api/js?key=AIzaSyBVWkrjvyyMaE75Q74_juiulg11Y0H2s&sensor=false" type="6e925e2cf2147df3f76e2c47-text/javascript"></script>
53 |   <script src="{{asset('js/google-map.js')}}" type="6e925e2cf2147df3f76e2c47-text/javascript"></script>
54 |   <script src="{{asset('js/main.js')}}" type="6e925e2cf2147df3f76e2c47-text/javascript"></script>
55 |
56 |   <script src="{{asset('js/jquery.validate.js')}}" type="6e925e2cf2147df3f76e2c47-text/javascript" defer></script>
57 |
58 |   <script async src="https://www.googletagmanager.com/gtag/js?id=UA-23581568-13" type="6e925e2cf2147df3f76e2c47-text/javascript"></script>
59 |   @yield('script')
60 |   <script type="6e925e2cf2147df3f76e2c47-text/javascript">
61 |     window.dataLayer = window.dataLayer || [];
62 |     function gtag(){dataLayer.push(arguments);}
63 |     gtag('js', new Date());
64 |
65 |     gtag('config', 'UA-23581568-13');
66 |     var btn = $('.mouse-icon');
67 |     $(window).scroll(function() {
68 |       if ($(window).scrollTop() > 300) {
69 |         btn.addClass('show');
70 |       } else {
71 |         btn.removeClass('show');
72 |       }
73 |     });
74 |
75 |     btn.on('click', function(e) {
76 |       e.preventDefault();
77 |       $('html, body').animate({scrollTop:0}, '300');
78 |     });
79 |   </script>
80 |
81 |   <script src="{{asset('js/rocket-loader.min.js')}}" data-cf-settings="6e925e2cf2147df3f76e2c47-|49" defer=""></script>
82 |
83 | </body>
84 |
85 | </html>

```

Figure 43: Index.php

```
home.blade.php ×
resources > views > website > home.blade.php > ...
1 @extends('website.index')
2
3 @section('body')
4     <section id="home-section" class="hero">
5         <div class="home-slider owl-carousel">
6             <div class="slider-item" style="background-image: url(images/bg_1.jpg);>
7                 <div class="overlay"></div>
8                 <div class="container">
9                     <div class="row slider-text justify-content-center align-items-center" data-scrollax-parent="true">
10                         <div class="col-md-12 ftco-animate text-center">
11                             <h1 class="mb-2">We serve Fresh Vegetables & Fruits</h1>
12                             <h2 class="subheading mb-4">We deliver organic vegetables & fruits</h2>
13
14                         </div>
15                     </div>
16                 </div>
17             </div>
18             <div class="slider-item" style="background-image: url(images/bg_2.jpg);>
19                 <div class="overlay"></div>
20                 <div class="container">
21                     <div class="row slider-text justify-content-center align-items-center" data-scrollax-parent="true">
22                         <div class="col-sm-12 ftco-animate text-center">
23                             <h1 class="mb-2">100% Fresh & Organic Foods</h1>
24                             <h2 class="subheading mb-4">We deliver organic vegetables & fruits</h2>
25
26                         </div>
27                     </div>
28                 </div>
29             </div>
30         </section>
31         <section class="ftco-section">
32             <div class="container">
33                 <div class="row no-gutters ftco-services">
34                     <div class="col-md-3 text-center d-flex align-self-stretch ftco-animate">
35                         <div class="media block-6 services mb-md-0 mb-4">
36                             <div class="icon bg-color-1 active d-flex justify-content-center align-items-center mb-2">
37                                 <span class="flaticon-shipped"></span>
38                             </div>
39                             <div class="media-body">
40                                 <h3 class="heading">Free Shipping</h3>
41                                 <span>On order over Rs1000</span>
42                             </div>
43                         </div>
44                     </div>
45                 </div>
46                 <div class="col-md-3 text-center d-flex align-self-stretch ftco-animate">
47                     <div class="media block-6 services mb-md-0 mb-4">
48                         <div class="icon bg-color-2 d-flex justify-content-center align-items-center mb-2">
49                             <span class="flaticon-diet"></span>
50                         </div>
51                         <div class="media-body">
52                             <h3 class="heading">Always Fresh</h3>
53                             <span>Product well package</span>
54                         </div>
55                     </div>
56                 </div>
57                 <div class="col-md-3 text-center d-flex align-self-stretch ftco-animate">
58                     <div class="media block-6 services mb-md-0 mb-4">
59                         <div class="icon bg-color-3 d-flex justify-content-center align-items-center mb-2">
60                             <span class="flaticon-award"></span>
61                         </div>
62                         <div class="media-body">
63                             <h3 class="heading">Superior Quality</h3>
64                             <span>Quality Products</span>
65                         </div>
66                     </div>
67                 </div>
68             </div>
69         </section>
```

```
home.blade.php ×
resources > views > website > home.blade.php > ...
35             <div class="col-md-3 text-center d-flex align-self-stretch ftco-animate">
36                 <div class="media block-6 services mb-md-0 mb-4">
37                     <div class="icon bg-color-1 active d-flex justify-content-center align-items-center mb-2">
38                         <span class="flaticon-shipped"></span>
39                     </div>
40                     <div class="media-body">
41                         <h3 class="heading">Free Shipping</h3>
42                         <span>On order over Rs1000</span>
43                     </div>
44                 </div>
45             </div>
46             <div class="col-md-3 text-center d-flex align-self-stretch ftco-animate">
47                 <div class="media block-6 services mb-md-0 mb-4">
48                     <div class="icon bg-color-2 d-flex justify-content-center align-items-center mb-2">
49                         <span class="flaticon-diet"></span>
50                     </div>
51                     <div class="media-body">
52                         <h3 class="heading">Always Fresh</h3>
53                         <span>Product well package</span>
54                     </div>
55                 </div>
56             </div>
57             <div class="col-md-3 text-center d-flex align-self-stretch ftco-animate">
58                 <div class="media block-6 services mb-md-0 mb-4">
59                     <div class="icon bg-color-3 d-flex justify-content-center align-items-center mb-2">
60                         <span class="flaticon-award"></span>
61                     </div>
62                     <div class="media-body">
63                         <h3 class="heading">Superior Quality</h3>
64                         <span>Quality Products</span>
65                     </div>
66                 </div>
67             </div>
68         </div>
```

```
home.blade.php X
resources > views > website > home.blade.php > ...
69 |     <div class="media block-6 services mb-md-0 mb-4">
70 |         <div class="icon bg-color-4 d-flex justify-content-center align-items-center mb-2">
71 |             <span class="flaticon-customer-service"></span>
72 |         </div>
73 |         <div class="media-body">
74 |             <h3 class="heading">Support</h3>
75 |             <span>24/7 Support</span>
76 |         </div>
77 |     </div>
78 | </div>
79 | </div>
80 | </div>
81 | </section>
82 | <div class="hero-wrap hero-bread" style="background-image: url('images/bg_1.jpg');">
83 |     <div class="container">
84 |         <div class="row no-gutters slider-text align-items-center justify-content-center">
85 |             <div class="col-md-9 ftco-animate text-center">
86 |                 <p class="breadcrumbs"><span class="mr-2"><a href="index-2.html">Home</a></span> <span>Products</span></p>
87 |                 <h1 class="mb-0 bread">Products</h1>
88 |             </div>
89 |         </div>
90 |     </div>
91 | </div>
92 | <section class="ftco-section">
93 |     <div class="container">
94 |         <div class="row justify-content-center">
95 |             <div class="col-md-12 mb-5 text-center">
96 |
97 |                 <ul class="product-category nav nav-pills justify-content-center mb-3 id="pills-tab" role="tablist">
98 |                     <?php if(count($category) > 0): ?>
99 |
100 |                         <?php $i = 0; ?>
101 |
102 |                         <?php foreach($category as $c): ?>
```

```
home.blade.php X
resources > views > website > home.blade.php > ...
103 |             <li><a class="<?php echo $i == 0 ? 'active' : ''; ?>" id="pills-<?php echo $c->category; ?>-tab" data-toggle="pill" href="#pills-<?php echo $c->category; ?>">
104 |                 <?php $i++; endforeach; ?>
105 |
106 |             </ul>
107 |         </div>
108 |     </div>
109 |     <div class="row">
110 |         <div class="tab-content id="pills-tabContent">
111 |             <?php $i = 0; ?>
112 |             <?php foreach($category as $c): ?>
113 |                 <div class="tab-pane fade <?php echo $i == 0 ? 'show active' : ''; ?>" id="pills-<?php echo $c->category; ?>" role="tabpanel">
114 |                     <?php foreach($c->products as $product): ?>
115 |                         <div class="col-md-4 ftco-animate">
116 |                             <div class="product">
117 |                                 <a href="{{route('product',['product'=>$product['slug']])}}" class="img-prod">
118 |                                     <?php
119 |                                         $mediaItems = $product->getMedia('images');
120 |                                         $publicUrl = $mediaItems[0]->getFullUrl();
121 |                                     ?>
122 |                                     
123 |                                     @if($product->new)
124 |                                         <span class="status">New</span>
125 |                                     @endif
126 |                                     <div class="overlay"></div>
127 |                                 </a>
128 |                                 <div class="text py-3 pb-4 px-3 text-center">
129 |                                     <h3><a href="{{route('product',['product'=>$product['slug']])}}"><?php echo $product['title']; ?></a></h3>
130 |                                     <div class="d-flex">
131 |                                         <div class="pricing">
132 |                                             <p class="price"><span>Rs. <?php echo $product['price']; ?> per kg</span></p>
133 |                                         </div>
```

```

home.blade.php X
resources > views > website > home.blade.php > ...
134 |     <div class="pricing">
135 |         <p class="price"><span>Rs. <?php echo $product['price']; ?> per kg</span></p>
136 |     </div>
137 |     <div class="bottom-area d-flex px-3">
138 |         <div class="m-auto d-flex">
139 |             <a href="{{route('product',['product'=>$product['slug']])}}" class="buy-now d-flex justify-content-center align-items-c
140 |                 <span><i class="ion-ios-cart"></i></span>
141 |             </a>
142 |         </div>
143 |     </div>
144 |     </div>
145 |     </div>
146 |     <?php endforeach; ?>
147 | </div>
148 | <?php $i++; endforeach; ?>
149 | <?php
150 |     if(count($category) > 0){
151 |         foreach($category[0]->products as $product){
152 |             }
153 |         }
154 |     }
155 | <?>
156 | </div>
157 | </div>
158 | </section>
159 |
160 |
161 | @endsection

```

Figure 44: Home.php

```

header.blade.php X
resources > views > website > header.blade.php > ...
1 | <div class="py-1 bg-primary">
2 |     <div class="container">
3 |         <div no-gutters d-flex align-items-start align-items-center px-md-0">
4 |             <div class="col-lg-12 d-block">
5 |                 <div class="row d-flex">
6 |                     <div class="col-md pr-4 d-flex topper align-items-center">
7 |                         <div class="icon mr-2 d-flex justify-content-center align-items-center"><span class="icon-phone2"></span></div>
8 |                         <span class="text">+ 01*****</span>
9 |                     </div>
10 |                     <div class="col-md pr-4 d-flex topper align-items-center">
11 |                         <div class="icon mr-2 d-flex justify-content-center align-items-center"><span class="icon-paper-plane"></span></div>
12 |                         <span class="text"><a href="#" style="color:white" >info@vegetablestore.com</a></span>
13 |                     </div>
14 |                     <div class="col-md-5 pr-4 d-flex topper align-items-center text-lg-right">
15 |                         <span class="text">24 * 7 delivery & Free Returns</span>
16 |                     </div>
17 |
18 |                     @if(Auth::check())
19 |
20 |                         <div class="col-md pr-4 d-flex topper align-items-center">
21 |                             <div class="icon mr-2 d-flex justify-content-center align-items-center"><span class="icon-logout"></span></div>
22 |                             <span class="text"><a href="/logout" style="color:white" >Logout</a></span>
23 |                         </div>
24 |
25 |                     @endif
26 |                 </div>
27 |             </div>
28 |         </div>
29 |     </div>
30 | </div>
31 | <nav class="navbar navbar-expand-lg navbar-dark ftco-navbar bg-dark ftco-navbar-light" id="ftco-navbar">
32 |     <div class="container">
33 |         <a class="navbar-brand" href="/">Vegetable Store</a>
34 |         <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#ftco-nav" aria-controls="ftco-nav" aria-expanded="false" aria-label=""

```

```
header.blade.php X
resources > views > website > header.blade.php > ...
35   <span class="oi oi-menu"></span> Menu
36   </button>
37   <div class="collapse navbar-collapse" id="ftco-nav">
38     <ul class="navbar-nav ml-auto">
39       <li class="nav-item"><a href="/" class="nav-link">Home</a></li>
40       <li class="nav-item"><a href="/about.html" class="nav-link">About</a></li>
41       <li class="nav-item"><a href="/contact.html" class="nav-link">Contact</a></li>
42
43       <li class="nav-item cta cta-colored"><a href="/cart.html" class="nav-link"><span class="icon-shopping_cart"></span><span id="total-item">[<?php if(Session::has('cart')) {<br/> echo count(Session::get('cart'));<br/>} else {<br/> echo '0';<br/>} ?></span></a></li>
44
45       @if(Auth::check())
46         <li class="nav-item"><a href="/profile.html" class="nav-link">{{Auth::user()->name}}</a></li>
47       @else
48
49       <li class="nav-item"><a href="/login.html" class="nav-link">Login/Sign up</a></li>
50       @endif
51
52     </ul>
53   </div>
54 </nav>
```

Figure 45: Header.php

```
footer.blade.php X
resources > views > website > footer.blade.php > footer
1 <footer style="padding-top:20px; background:#82ae46;">
2   <div class="container">
3     <div class="row">
4       <div class="col-md-12 text-center">
5         <p style="color:white">
6           Copyright &copy; Vegetable Store <script type="6e925e2cf2147df3f76e2c47-text/javascript">document.write(new Date().getFullYear());</script>
7         </p>
8       </div>
9     </div>
10   </div>
11 </footer>
12
13 <div class="mouse" id="scroll-btn">
14   <a href="#" class="mouse-icon">
15     <div class="mouse-wheel"><span class="ion-ios-arrow-up"></span></div>
16   </a>
17 </div>
```

Figure 46: Footer.php

C ① 127.0.0.1:8000/about.html

INFO@VEGETABLESTORE.COM

24 \* 7 DELIVERY & FREE RETURNS

LOGOUT

VEGETABLE STORE

HOME ABOUT CONTACT [ 0 ] DAWA GYALZEN SHERPA



## Welcome to Vegetable Delivery System

C ① 127.0.0.1:8000/about.html

VEGETABLE STORE

HOME ABOUT CONTACT [ 0 ] DAWA GYALZEN SHERPA



## Welcome to Vegetable Delivery System

We stand for making conscious food-choices. Understanding the processes involved while growing your food is key into resolving some of the world's critical problems such as that of environmental degradation, economic inequality, health issues and much more. Creating a place to connect growers who are in the organic and wholesome path to provide you with fresh, sustainable food is a start for us to take care of our earth, our mind and body.

Our objective is to give access to the right of knowledge to our consumers. As consumers we lack information about our own food choices. We offer you that very information to help you choose better. This choice will ultimately help protect nature, biodiversity and also empower communities by becoming self-reliant through the sustainable use of natural resources.



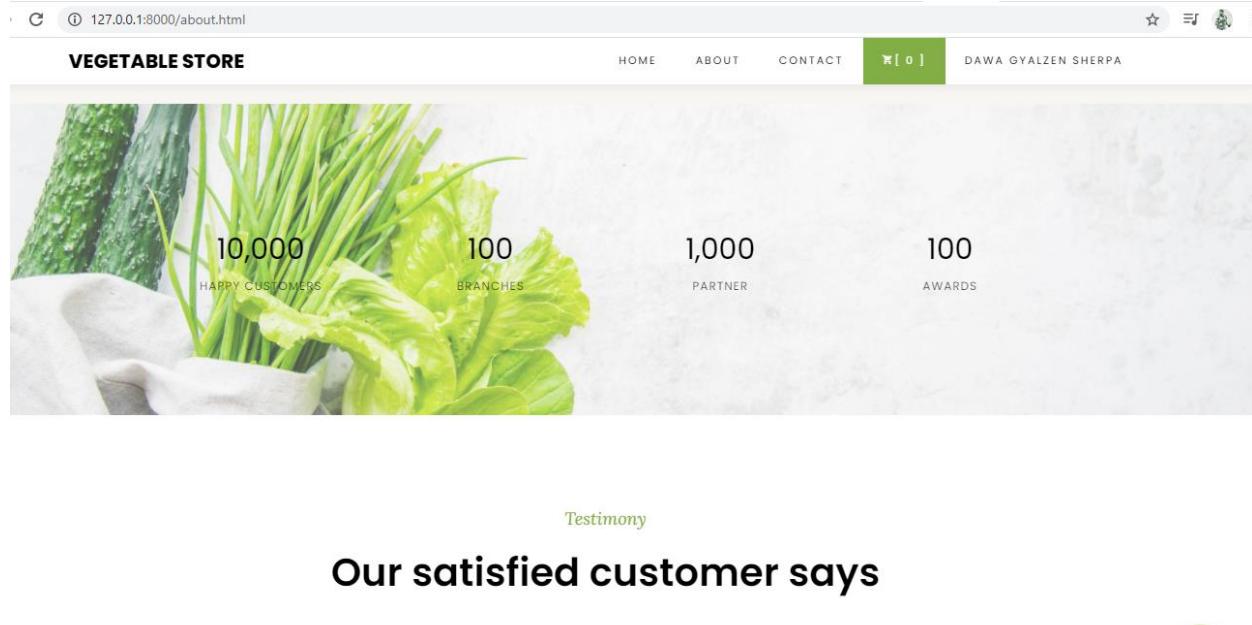


Figure 47: About Page

```

about.blade.php X
resources > views > website > about.blade.php > ...
1 @extends('website.index')
2
3  @section('body')
4  <div class="hero-wrap hero-bread" style="background-image: url('images/bg_1.jpg');">
5    <div class="container">
6      <div class="row no-gutters slider-text align-items-center justify-content-center">
7        <div class="col-md-9 ftco-animate text-center">
8          <p class="breadcrumbs"><span class="mr-2"><a href="#">Home</a></span> <span>About us</span></p>
9          <h1 class="mb-0 bread">About us</h1>
10         </div>
11       </div>
12     </div>
13   </div>
14 <section class="ftco-section ftco-no-pb ftco-no-pt bg-light">
15   <div class="container">
16     <div class="row">
17       <div class="col-md-5 p-md-5 img img-2 d-flex justify-content-center align-items-center">
18         <iframe width="560" height="515" src="https://www.youtube.com/embed/0Ix8hIqB_8k" allowfullscreen></iframe>
19       </div>
20       <div class="col-md-7 py-5 wrap-about pb-md-5 ftco-animate">
21         <div class="heading-section-bold mb-4 mt-md-5">
22           <div class="ml-md-0">
23             <h2 class="mb-4">Welcome to Vegetable Delivery System</h2>
24           </div>
25         </div>
26         <div class="pb-md-5">
27           <p>We stand for making conscious food-choices. Understanding the processes involved while growing your food is key into resolving some of the world's most pressing issues. Our objective is to give access to the right of knowledge to our consumers. As consumers we lack information about our own food choices. We offer a platform where you can learn about the food you eat and make informed decisions.</p>
28           <!-- <p><a href="#" class="btn btn-primary">Shop now</a></p> -->
29         </div>
30       </div>
31     </div>
32   </div>
33 </section>
34

```

```
⌚ about.blade.php X
resources > views > website > ⌚ about.blade.php > ...
35  <section class="ftco-section ftco-counter img" id="section-counter" style="background-image: url(images/bg_3.jpg);>
36  <div class="container">
37  <div class="row justify-content-center py-5">
38  <div class="col-md-10">
39  <div class="row">
40  <div class="col-md-3 d-flex justify-content-center counter-wrap ftco-animate">
41  <div class="block-18 text-center">
42  <div class="text">
43  <strong class="number" data-number="10000">0</strong>
44  <span>Happy Customers</span>
45  </div>
46  </div>
47 </div>
48 <div class="col-md-3 d-flex justify-content-center counter-wrap ftco-animate">
49  <div class="block-18 text-center">
50  <div class="text">
51  <strong class="number" data-number="100">0</strong>
52  <span>Branches</span>
53  </div>
54 </div>
55 <div class="col-md-3 d-flex justify-content-center counter-wrap ftco-animate">
56  <div class="block-18 text-center">
57  <div class="text">
58  <strong class="number" data-number="1000">0</strong>
59  <span>Partner</span>
60  </div>
61 </div>
62 </div>
63 <div class="col-md-3 d-flex justify-content-center counter-wrap ftco-animate">
64  <div class="block-18 text-center">
65  <div class="text">
66  <strong class="number" data-number="100">0</strong>
67  <span>Awards</span>
68 </div>
```

```
⌚ about.blade.php X
resources > views > website > ⌚ about.blade.php > ...
69  </div>
70  </div>
71  </div>
72  </div>
73  </div>
74 </div>
75 </div>
76 </div>
77 <section class="ftco-section testimony-section">
78  <div class="container">
79  <div class="row justify-content-center mb-5 pb-3">
80  <div class="col-md-7 heading-section ftco-animate text-center">
81  <span class="subheading">Testimony</span>
82  <h2 class="mb-4">Our satisfied customer says</h2>
83  <p>Great service and values on reasonable price range.Thanks to its innovative ideas.</p>
84 </div>
85 </div>
86 <div class="row ftco-animate">
87  <div class="col-md-12">
88  <div class="carousel-testimony owl-carousel">
89  @if(sizeof($data)>0)
90  @foreach($data as $testimonial)
91  <div class="item">
92  <div class="testimony-wrap p-4 pb-5">
93  @php
94  $mediaItems = $testimonial->getMedia('image');
95  $publicUrl = $mediaItems[0]->getUrl();
96  @endphp
97  <div class="user-img mb-5" style="background-image: url({{$publicUrl}})">
98  <span class="quote d-flex align-items-center justify-content-center">
99  <i class="icon-quote-left"></i>
100 </span>
101 </div>
102 <div class="text text-center">
```

```
aboutblade.php X
resources > views > website > aboutblade.php > ...
103 |     <p class="mb-5 pl-4 line">{$testimonial->message}</p>
104 |     <p class="name">{$testimonial->name}</p>
105 |
106 |         </div>
107 |     </div>
108 |     </div>
109 | @endforeach
110 @endif
111 <!-- <div class="item">
112     <div class="testimony-wrap p-4 pb-5">
113         <div class="user-img mb-5" style="background-image: url(images/person_2.jpg)">
114             <span class="quote d-flex align-items-center justify-content-center">
115                 <i class="icon-quote-left"></i>
116             </span>
117         </div>
118         <div class="text text-center">
119             <p class="mb-5 pl-4 line">Great service and values on reasonable price range.Thanks to its innovative ideas.</p>
120             <p class="name">Garreth Smith</p>
121             <span class="position">Interface Designer</span>
122         </div>
123     </div>
124 </div>
125 <div class="item">
126     <div class="testimony-wrap p-4 pb-5">
127         <div class="user-img mb-5" style="background-image: url(images/person_3.jpg)">
128             <span class="quote d-flex align-items-center justify-content-center">
129                 <i class="icon-quote-left"></i>
130             </span>
131         </div>
132         <div class="text text-center">
133             <p class="mb-5 pl-4 line">Love them! They are very helpful.</p>
134             <p class="name">Garreth Smith</p>
135             <span class="position">UI Designer</span>
136         </div>

```

```
aboutblade.php X
resources > views > website > aboutblade.php > ...
137     </div>
138 </div>
139 <div class="item">
140     <div class="testimony-wrap p-4 pb-5">
141         <div class="user-img mb-5" style="background-image: url(images/person_1.jpg)">
142             <span class="quote d-flex align-items-center justify-content-center">
143                 <i class="icon-quote-left"></i>
144             </span>
145         </div>
146         <div class="text text-center">
147             <p class="mb-5 pl-4 line">Good service good Quality.</p>
148             <p class="name">Garreth Smith</p>
149             <span class="position">Web Developer</span>
150         </div>
151     </div>
152 </div>
153 <div class="item">
154     <div class="testimony-wrap p-4 pb-5">
155         <div class="user-img mb-5" style="background-image: url(images/person_1.jpg)">
156             <span class="quote d-flex align-items-center justify-content-center">
157                 <i class="icon-quote-left"></i>
158             </span>
159         </div>
160         <div class="text text-center">
161             <p class="mb-5 pl-4 line">Excellent and great idea and business. Highly recommended to all.</p>
162             <p class="name">Garreth Smith</p>
163             <span class="position">System Analyst</span>
164         </div>
165     </div>
166     </div> -->
167 </div>
168 </div>
169 </div>
170 </section>
171 @endsection
172
```

Figure 48: About.php

→ C ① 127.0.0.1:8000/contact.html

+ 01\*\*\*\*\* INFO@VEGETABLESTORE.COM 24 \* 7 DELIVERY & FREE RETURNS LOGOUT

**VEGETABLE STORE**

HOME ABOUT CONTACT [ 0 ] DAWA GYALZEN SHERPA

HOME CONTACT US  
**CONTACT US**

Address: City Center,  
Kathmandu ,Nepal

Phone: 01\*\*\*\*\*

Email:  
info@vegetablestore.com

Website  
vegetablestore.com

---

→ C ① 127.0.0.1:8000/contact.html

**VEGETABLE STORE**

HOME ABOUT CONTACT [ 0 ] DAWA GYALZEN SHERPA

Your Name

Your Email

Subject

Message

Send Message

Copyright © Vegetable Store 2020 All rights reserved

Figure 49: Contact Page

```
about.blade.php contact.blade.php X
resources > views > website > contact.blade.php > ...
1 @extends('website.index')
2 @section('body')
3     <div class="hero-wrap hero-bread" style="background-image: url('images/bg_1.jpg');">
4         <div class="container">
5             <div class="row no-gutters slider-text align-items-center justify-content-center">
6                 <div class="col-md-9 ftco-animate text-center">
7                     <p class="breadcrumbs"><span class="mr-2"><a href="index-2.html">Home</a></span> <span>Contact us</span></p>
8                     <h1 class="mb-0 bread">Contact us</h1>
9                 </div>
10            </div>
11        </div>
12    </div>
13    <section class="ftco-section contact-section bg-light">
14        <div class="container">
15            <div class="row d-flex mb-5 contact-info">
16                <div class="w-100"></div>
17                <div class="col-md-3 d-flex">
18                    <div class="info bg-white p-4">
19                        <p><span>Address:</span> City Center, Kathmandu ,Nepal</p>
20                    </div>
21                </div>
22                <div class="col-md-3 d-flex">
23                    <div class="info bg-white p-4">
24                        <p><span>Phone:</span> <a href="tel://1234567920">01*****</a></p>
25                    </div>
26                </div>
27                <div class="col-md-3 d-flex">
28                    <div class="info bg-white p-4">
29                        <p><span>Email:</span> <a href="#"><span class="__cf_email__" >info@vegetablestore.com</span></a></p>
30                    </div>
31                </div>
32                <div class="col-md-3 d-flex">
33                    <div class="info bg-white p-4">
34                        <p><span>Website:</span> <a href="#">vegetablestore.com</a></p>
35                    </div>
36                </div>
37            </div>
38        <div class="row block-9">
39
40            <div class="col-md-6 order-md-last d-flex">
41
42                <form action="{{route('message')}}" class="bg-white p-5 contact-form" id="contact-form" method="POST">
43                    @if(Session::has('success'))
44                        <div class="alert alert-success">{{ Session::get('success') }}</div>
45                    @endif
46
47                    @if(Session::has('error'))
48                        <div class="alert alert-success">{{ Session::get('error') }}</div>
49                    @endif
50
51                    @csrf
52
53                    <div class="form-group">
54                        <input type="text" class="form-control" name="name" id="name" placeholder="Your Name">
55                    </div>
56                    <div class="form-group">
57                        <input type="text" class="form-control" name="email" id="email" placeholder="Your Email">
58                    </div>
59                    <div class="form-group">
60                        <input type="text" class="form-control" name="subject" id="subject" placeholder="Subject">
61                    </div>
62                    <div class="form-group">
63                        <input type="text" class="form-control" name="message" id="message" placeholder="Message" >
64                    </div>
65
66                </form>
67            </div>
68        <div class="col-md-6 d-flex">
```

```
about.blade.php contact.blade.php X
resources > views > website > contact.blade.php > ...
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
```

```
about.blade.php contact.blade.php x
resources > views > website > contact.blade.php > ...
69     <div>
70         <div>
71             </div>
72         </div>
73     </section>
74     @endsection
75     @section('script')
76     <script type="6e925e2cf2147df3f76e2c47-text/javascript">
77
78
79     $(document).ready(function(){
80
81         $("#contact-form").validate({
82
83             rules: {
84                 name: {
85                     required: true,
86                     maxlength: 50
87                 },
88
89                 email: {
90                     required: true,
91                     maxlength: 50,
92                     email: true,
93                     minlength: 5,
94                 },
95                 subject: {
96                     required: true,
97                     maxlength: 50,
98                     minlength: 5,
99                 },
100                message: {
101                    required: true,
102                    maxlength: 50,
```

```
about.blade.php X contact.blade.php X
resources > views > website > contact.blade.php > script
103         minlength: 5,
104     },
105     ],
106   },
107   messages: {
108     name: {
109       required: "Please enter your name",
110       maxlength: "The name should less than or equal to 50 characters",
111     },
112     email: {
113       required: "Please enter valid email",
114       email: "Please enter valid email",
115       maxlength: "The email name should less than or equal to 50 characters",
116       minlength: "The email name should greater than or equal to 15 characters",
117     },
118     subject: {
119       required: "Subject field is required",
120       maxlength: "The subject should less than or equal to 50 characters",
121       minlength: "The subject should greater than 5 characters",
122     },
123     message: [
124       {
125         required: "Message field is required",
126         maxlength: "The address should less than or equal to 50 characters",
127         minlength: "The address should greater than or equal to 5 characters",
128       },
129     ],
130   },
131 },
132 },
133 },
134 });
135 </script>
136 @endsection
```

*Figure 50: Contact.php*

① 127.0.0.1:8000/login.html

star

+ 0]\*\*\*\*\*

INFO@VEGETABLESTORE.COM

24 \* 7 DELIVERY & FREE RETURNS

VEGETABLE STORE

HOME ABOUT CONTACT [ 0 ] LOGIN/SIGN UP

Log In Sign Up

Enter email

Enter password

Login

Copyright © Vegetable Store 2020 All rights reserved

Figure 51: Login Form

+ 0]\*\*\*\*\*

INFO@VEGETABLESTORE.COM

24 \* 7 DELIVERY & FREE RETURNS

VEGETABLE STORE

HOME ABOUT CONTACT [ 0 ] LOGIN/SIGN UP

Log In Sign Up

Name

Enter your email

Enter your phone number

Enter your address

Enter password

Confirm password

Create Account

Figure 52: SignUp Form

```
(contact.blade.php) (login.blade.php) ...
resources > views > website > login.blade.php > ...
1  @extends('website.index')
2
3  @section('body')
4
5  <div class="form-modal">
6      <div class="form-toggle">
7          <button id="login-toggle" onclick="toggleLogin()">log in</button>
8          <button id="signup-toggle" onclick="toggleSignup()">sign up</button>
9      </div>
10
11     <div id="login-form">
12         @if (Session::has('success'))
13             <div class="alert alert-success">{{ Session::get('success') }}</div>
14         @endif
15
16         @if (Session::has('error'))
17             <div class="alert alert-danger">{{ Session::get('error') }}</div>
18         @endif
19
20         <form method="POST" action="{{route("ulogin")}}" autocomplete="off" id="login">
21             {{csrf_field()}}
22             <input name="email" class="{{ $errors->has('email') ? ' is-invalid' : '' }}" type="email" placeholder="Enter email" required/>
23             @if ($errors->has('email'))
24                 <span class="help-block text-danger">
25                     <strong>{{ $errors->first('email') }}</strong>
26                 </span>
27             @endif
28
29             <input name="password" class="{{ $errors->has('password') ? ' is-invalid' : '' }}" type="password" placeholder="Enter password" required/>
30
31             @if ($errors->has('password'))
32                 <span class="help-block text-danger">
33                     <strong>{{ $errors->first('password') }}</strong>
34                 </span>
35         @endif
36
37         <input type="submit" class="btn login" value="Login">
38     </form>
39
40     <div id="signup-form">
41         <div class="alert alert-success d-none" id="msg_div">
42             <span id="res_message"></span>
43         </div>
44         <form autocomplete="off" method="POST" action="{{route('userregister')}}" id="register" >
45             {{csrf_field()}}
46             <input type="text" class="{{ $errors->has('name') ? ' is-invalid' : '' }}" name="name" placeholder="Name" id="name" />
47
48             @if ($errors->has('name'))
49                 <span class="help-block text-danger">
50                     <strong>{{ $errors->first('name') }}</strong>
51                 </span>
52                 <span class="text-danger">{{ $errors->first('name') }}</span>
53             @endif
54             <input type="email" class="{{ $errors->has('email') ? ' is-invalid' : '' }}" name="email" placeholder="Enter your email" id="email" />
55
56             @if ($errors->has('email'))
57                 <span class="help-block text-danger">
58                     <strong>{{ $errors->first('email') }}</strong>
59                 </span>
60             @endif
61
62             <input type="text" class="{{ $errors->has('phone') ? ' is-invalid' : '' }}" name="phone" placeholder="Enter your phone number" id="phone" oninput="this.value = this.value.replace(/\D/g, '')" />
63             @if ($errors->has('phone'))
64                 <span class="help-block text-danger">
65                     <strong>{{ $errors->first('phone') }}</strong>
66                 </span>
67             @endif
68         @endif
69     </div>
70 
```

```
(contact.blade.php) (login.blade.php) ...
resources > views > website > login.blade.php > ...
35     @endif
36     <input type="submit" class="btn login" value="Login">
37
38 </div>
39
40 <div id="signup-form">
41     <div class="alert alert-success d-none" id="msg_div">
42         <span id="res_message"></span>
43     </div>
44     <form autocomplete="off" method="POST" action="{{route('userregister')}}" id="register" >
45         {{csrf_field()}}
46         <input type="text" class="{{ $errors->has('name') ? ' is-invalid' : '' }}" name="name" placeholder="Name" id="name" />
47
48         @if ($errors->has('name'))
49             <span class="help-block text-danger">
50                 <strong>{{ $errors->first('name') }}</strong>
51             </span>
52             <span class="text-danger">{{ $errors->first('name') }}</span>
53         @endif
54         <input type="email" class="{{ $errors->has('email') ? ' is-invalid' : '' }}" name="email" placeholder="Enter your email" id="email" />
55
56         @if ($errors->has('email'))
57             <span class="help-block text-danger">
58                 <strong>{{ $errors->first('email') }}</strong>
59             </span>
60         @endif
61
62         <input type="text" class="{{ $errors->has('phone') ? ' is-invalid' : '' }}" name="phone" placeholder="Enter your phone number" id="phone" oninput="this.value = this.value.replace(/\D/g, '')" />
63         @if ($errors->has('phone'))
64             <span class="help-block text-danger">
65                 <strong>{{ $errors->first('phone') }}</strong>
66             </span>
67         @endif
68     </div>
69 
```

```
(contact.blade.php) login.blade.php ...
resources > views > website > login.blade.php > ...
69      <input type="text" class="{{ $errors->has('address') ? ' is-invalid' : '' }}" name="address" placeholder="Enter your address" id="address" />
70
71      @if ($errors->has('address'))
72          <span class="help-block text-danger">
73              | <strong>{{ $errors->first('address') }}</strong>
74          </span>
75      @endif
76      <input type="password" class="{{ $errors->has('password') ? ' is-invalid' : '' }}" name="password" placeholder="Enter password" id="password" />
77
78      @if ($errors->has('password'))
79          <span class="help-block text-danger">
80              | <strong>{{ $errors->first('password') }}</strong>
81          </span>
82      @endif
83      <input type="password" class="{{ $errors->has('password_confirmation') ? ' is-invalid' : '' }}" name="password_confirmation" placeholder="Confirm password" id="password_confirmation" />
84
85      @if ($errors->has('password_confirmation'))
86          <span class="help-block text-danger">
87              | <strong>{{ $errors->first('password_confirmation') }}</strong>
88          </span>
89      @endif
90      <button type="submit" class="btn signup">Create account</button>
91  </form>
92 </div>
93
94
95
96 @endsection
97 @section('script')
98 <script type="6e925e2cf2147df3f76e2c47-text/javascript">
99
100
101     $(document).ready(function(){
102         $("#login").validate({

```

```
(contact.blade.php) login.blade.php ...
resources > views > website > login.blade.php > ...
103     rules: {
104         email: {
105             required: true,
106             maxlength: 50,
107             email: true,
108         },
109         password: {
110             required: true,
111             maxlength: 50,
112             minlength: 5,
113         },
114     },
115     messages: {
116         email: {
117             required: "Please enter valid email",
118             email: "Please enter valid email",
119             maxlength: "The email name should less than or equal to 50 characters",
120         },
121         password: {
122             required: "Password field is required",
123             maxlength: "The password should less than or equal to 15 characters",
124             minlength: "The password should greater than or equal to 5 characters",
125         },
126     },
127 },
128 });
129
130 });
131
132     $("#register").validate({
133
134         rules: {
135             name: {
136                 required: true,
```

```
resources > views > website > contact.blade.php X
resources > views > website > login.blade.php X ...
```

```
137     },
138     },
139     },
140     },
141     },
142     },
143     },
144     },
145     },
146     },
147     },
148     },
149     },
150     },
151     },
152     },
153     },
154     },
155     },
156     },
157     },
158     },
159     },
160     },
161     },
162     },
163     },
164     },
165     },
166     },
167     },
168     },
169     },
170     },
```

```
resources > views > website > contact.blade.php X
resources > views > website > login.blade.php X ...
```

```
171     },
172     },
173     },
174     },
175     },
176     },
177     },
178     },
179     },
180     },
181     },
182     },
183     },
184     },
185     },
186     },
187     },
188     },
189     },
190     },
191     },
192     },
193     },
194     },
195     },
196     },
197     },
198     },
199     },
200     },
201     },
202     },
203     },
204     },
205     
```

```
1.0
```

Figure 53: Login and Signup.php

← → ⌂ ① 127.0.0.1:8000/cart.html

+ 0\*\*\*\*\* INFO@VEGETABLESTORE.COM 24 \* 7 DELIVERY & FREE RETURNS

**VEGETABLE STORE** HOME ABOUT CONTACT ⚡ [ 0 ] LOGIN/SIGN UP

HOME CART  
**MY CART**

Product name	Price	Quantity	Total
Sweet Potatoes	Rs. 100 Per KG	250 Gram	Rs. 25

**VEGETABLE STORE** HOME ABOUT CONTACT ⚡ [ 1 ] DAWA GYALZEN SHERPA

Product name	Price	Quantity	Total
Sweet Potatoes	Rs. 100 Per KG	250 Gram	Rs. 25

**Cart Totals**

TOTAL	<b>25</b>
-------	-----------

**Proceed to Checkout**

Figure 54: Add to Cart Page

```
login.blade.php X cart.blade.php X
resources > views > website > cart.blade.php > ...
1 @extends('website.index')
2 @section('body')
3 <div class="hero-wrap hero-bread" style="background-image: url('images/bg_1.jpg');">
4 <div class="container">
5 <div class="row no-gutters slider-text align-items-center justify-content-center">
6 <div class="col-md-9 ftco-animate text-center">
7 <p class="breadcrumbs"><span class="mr-2"><a href="index-2.html">Home</a></span> <span>Cart</span></p>
8 <h1 class="mb-0 bread">My Cart</h1>
9 </div>
10 </div>
11 </div>
12 </div>
13 <section class="ftco-section ftco-cart">
14 <div class="container">
15 <div class="row">
16 <div class="col-md-12 ftco-animate">
17 <div class="cart-list">
18 <table class="table">
19 <thead class="thead-primary">
20 <tr class="text-center">
21 <th>&nbsp;</th>
22 <th>&nbsp;</th>
23 <th>Product name</th>
24 <th>Price</th>
25 <th>Quantity</th>
26 <th>Total</th>
27 </tr>
28 </thead>
29 <tbody>
30 <?php if ($session::has("cart")): ?>
31 <?php $i = 0; ?>
32 <?php foreach ($session::get("cart") as $cart): ?>
33 <tr class="text-center">
34 <td class="product-remove"><a href="#" data-index="<?php echo $i; ?>" class="delete-product"><span class="ion-ios-close"></span></a>
```

```
login.blade.php X cart.blade.php X
resources > views > website > cart.blade.php > ...
35 <td class="image-prod">
36 <div class="img" style="background-image:url('{{ $cart->image }}');"></div>
37 </td>
38 <?php $i++; ?>
39 <td class="product-name">
40 <h3><?php echo $cart->title; ?></h3>
41 <p><?php echo $cart->description; ?></p>
42 </td>
43 <td class="price">Rs. <?php echo $cart->price; ?> Per KG</td>
44 <td class="quantity">
45 <?php
46
47 if($cart->qty/1000>=1){
48 echo $cart->qty/1000 ." KG";
49 }else{
50 echo "$cart->qty Gram";
51 }
52 ?>
53 </td>
54 <td class="total">Rs. <?php echo $cart->total; ?></td>
55 </tr>
56 <?php endforeach; ?>
57 <?php endif; ?>
58 </tbody>
59 </table>
60 </div>
61 </div>
62 </div>
63 <div class="row justify-content-end">
64 <div class="col-lg-4 mt-5 cart-wrap ftco-animate">
65 <div class="cart-total mb-3">
66 <h3>Cart Totals</h3>
```

```

login.blade.php X cart.blade.php X
resources > views > website > cart.blade.php > ...
69   <p class="d-flex total-price">
70     <span>Total</span>
71     <span class="price">
72       @if(Session::has("cart"))
73         @php
74           $sum = array_reduce(Session::get("cart"),function($i,$obj){
75             return $i += $obj->total;
76           });
77           echo $sum;
78         @endphp
79       @endif
80     </span>
81   </p>
82   <p><a href="checkout.html" class="btn btn-primary py-3 px-4 <?php echo Session::get('cart') ? '' : 'disabled'; ?>">Proceed to Checkout</a></p>
83 </div>
84 </div>
85 </div>
86 </div>
87 </div>
88 </div>
89 </div>
90 </div>
91 </section>
92
93 <div class="toast bg-success" style="position: fixed; top: 50px; right: 0; z-index: 999999;">
94   <div class="toast-header">
95     <p class="mb-0">Item removed successfully.</p>
96     <button type="button" class="ml-2 mb-1 close" data-dismiss="toast" aria-label="Close">
97       <span aria-hidden="true">&times;</span>
98     </button>
99   </div>
100 </div>
101
102 <div class="toast toast-success bg-success" style="position: fixed; top: 50px; right: 0; z-index: 999999;">
103   <div class="toast-header">
104     <p class="mb-0">Item removed successfully.</p>
105     <button type="button" class="ml-2 mb-1 close" data-dismiss="toast" aria-label="Close">
106       <span aria-hidden="true">&times;</span>
107     </button>
108   </div>
109 </div>
110
111 <div class="toast toast-error bg-danger" style="position: fixed; top: 50px; right: 0; z-index: 999999;">
112   <div class="toast-header">
113     <p class="mb-0">Something went wrong.</p>
114     <button type="button" class="ml-2 mb-1 close" data-dismiss="toast" aria-label="Close">
115       <span aria-hidden="true">&times;</span>
116     </button>
117   </div>
118 </div>
119 @endsection

```

Figure 55: Cart.php





```
login.blade.php profile.blade.php
resources > views > website > profile.blade.php > ...
35     <th>Address</th>
36     <th>Contact</th>
37     <th>/</th>
38     </tr>
39   </thead>
40   <tbody>
41
42
43   <?php foreach($orders as $order): ?>
44
45     <tr>
46       <td>{$order->order_id}</td>
47       <td>{$order->name}</td>
48       <td>{$order->address}</td>
49       <td>{$order->contact}</td>
50       <td><span type="button" data-order="{$json_encode($order)}" data-id="<?php echo $order['order_id']; ?>" data-toggle="modal" data-target="#order-modal">View</span></td>
51     </tr>
52   <?php endforeach; ?>
53 </tbody>
54 </table>
55 </div>
56 </div>
57 </div>
58 </div>
59
60
61 <div class="modal fade" id="order-modal" tabindex="-1" role="dialog">
62   <div class="modal-dialog" role="document">
63     <div class="modal-content">
64       <div class="modal-header">
65         <h5 class="modal-title"></h5>
66         <button type="button" class="close" data-dismiss="modal" aria-label="Close">
67           <span aria-hidden="true">&times;</span>
68         </button>
69       </div>
70       <div class="modal-body">
71
72         <table id="items" class="table w-100" style="min-width: 100% !important">
73           <thead class="thead-primary">
74             <tr class="text-center">
75               <th>Item</th>
76               <th>Quantity</th>
77               <th>Rate/Kg</th>
78               <th>Price</th>
79             </tr>
80           </thead>
81           <tbody>
82
83             </tbody>
84           </table>
85         </div>
86       </div>
87     </div>
88   </div>
89 </div>
90
91 @endsection
92 @section('script')
93 <script type="6e925e2cf2147df3f76e2c47-text/javascript">
94 $(document).ready(function() {
95
96   $(".icon-eye").click(function(e) {
97     e.preventDefault();
98     $("#items > tbody").html("");
99
100    var order = JSON.parse($(this).attr('data-order'));
101    var id= $(this).attr('data-id');
102    $('.modal').attr('order-id', $(this).attr('data-id'))
```

```
login.blade.php profile.blade.php
resources > views > website > profile.blade.php > ...
69   </div>
70   <div class="modal-body">
71
72     <table id="items" class="table w-100" style="min-width: 100% !important">
73       <thead class="thead-primary">
74         <tr class="text-center">
75           <th>Item</th>
76           <th>Quantity</th>
77           <th>Rate/Kg</th>
78           <th>Price</th>
79         </tr>
80       </thead>
81       <tbody>
82
83         </tbody>
84       </table>
85     </div>
86   </div>
87 </div>
88 </div>
89 </div>
90
91 @endsection
92 @section('script')
93 <script type="6e925e2cf2147df3f76e2c47-text/javascript">
94 $(document).ready(function() {
95
96   $(".icon-eye").click(function(e) {
97     e.preventDefault();
98     $("#items > tbody").html("");
99
100    var order = JSON.parse($(this).attr('data-order'));
101    var id= $(this).attr('data-id');
102    $('.modal').attr('order-id', $(this).attr('data-id'))
```

```

resources > views > website > profile.blade.php > ...
103     console.log(order.orderitems);
104     for(var i=0;i<order.orderitems.length;i++){
105         var unit = order.orderitems[i].quantity/1000<1?"Gram":"KG"
106         var qty= order.orderitems[i].quantity/1000<1?order.orderitems[i].quantity:order.orderitems[i].quantity/1000;
107         $("#items").append(
108             `<tr>
109                 <td>
110                     ${order.orderitems[i].product.title}
111                 </td>
112                 <td>
113                     ${qty} ${unit}
114                 </td>
115                 <td>
116                     Rs. ${order.orderitems[i].product.price}
117                 </td>
118                 <td>
119                     Rs . ${order.orderitems[i].price}
120                 </td>
121             </tr>
122         );
123     );
124 });
125 });
126 </script>
127 @endsection

```

Figure 57: Profile.php

#	ID	Category	Category slug	Action
<input type="checkbox"/>	1	Vegetables	vegetables	
<input type="checkbox"/>	2	Fruits	fruits	

Figure 58: Category Page

```
index.blade.php X
resources > views > admin > category > index.blade.php > ...
1  @extends("brackets/admin-ui::admin.layout.default")
2
3  @section('title', trans('admin.category.actions.index'))
4
5  @section('body')
6
7      <category-listing
8          :data="{!! $data->toJson() !!}"
9          :url="{!! url('admin/categories') !!}"
10         inline-template>
11
12         <div class="row">
13             <div class="col">
14                 <div class="card">
15                     <div class="card-header">
16                         <i class="fa fa-align-justify"></i> {{ trans('admin.category.actions.index') }}
```

```
index.blade.php X
resources > views > admin > category > index.blade.php > ...
17     <a class="btn btn-primary btn-spinner btn-sm pull-right m-b-0" href="{!! url('admin/categories/create') !!}" role="button"><i class="fa fa-plus"></i>&ampnbsp {{ trans('admin.category.actions.create') }} </a>
18 </div>
19 <div class="card-body" v-cloak>
20     <div class="card-block">
21         <form @submit.prevent="">
22             <div class="row justify-content-md-between">
23                 <div class="col col-lg-7 col-xl-5 form-group">
24                     <div class="input-group">
25                         <input class="form-control" placeholder="{!! trans('brackets/admin-ui::admin.placeholder.search') !!}" v-model="search" @keyup.enter="filter" type="text">
26                         <span class="input-group-append">
27                             <button type="button" class="btn btn-primary" @click="filter('search', search)"><i class="fa fa-search"></i>&ampnbsp {{ trans('brackets/admin-ui::admin.button.filter') }} </button>
28                         </span>
29                     </div>
30                 </div>
31                 <div class="col-sm-auto form-group ">
32                     <select class="form-control" v-model="pagination.state.per_page">
33                         <option value="10">10</option>
34                         <option value="25">25</option>
35                         <option value="100">100</option>
36                     </select>
37                 </div>
38             </div>
39         </form>
40
41         <table class="table table-hover table-listing">
42             <thead>
43                 <tr>
44                     <th class="bulk-checkbox">
45                         <input class="form-check-input" id="enabled" type="checkbox" v-model="isClickedAll" v-validate="" data-vv-name="enabled" name="enabled" checked="">
46                         <label class="form-check-label" for="enabled">
47                             #
48                         </label>
49                     </th>
50                 </tr>
```

```

index.blade.php X
resources > views > admin > category > index.blade.php > ...
51
52     <th is='sortable' :column="'id'">{{ trans('admin.category.columns.id') }}</th>
53     <th is='sortable' :column="'category'">{{ trans('admin.category.columns.category') }}</th>
54     <th is='sortable' :column="'category_slug'">{{ trans('admin.category.columns.category_slug') }}</th>
55
56     <th></th>
57 </tr>
58 <tr v-show="(clickedBulkItemsCount > 0) || isClickedAll">
59     <td class="bg-bulk-info d-table-cell text-center" colspan="5">
60         <span class="align-middle font-weight-light text-dark">{{ trans('brackets/admin-ui::admin.listing.selected_items') }} @{{ clickedBulkItemsCount }}</span>
61         <a href="#" class="text-primary" @click="onBulkItemsClickedAllUncheck()">{{ trans('brackets/admin-ui::admin.listing.uncheck_all_items') }}</a>
62
63         <span class="pull-right pr-2">
64             <button class="btn btn-sm btn-danger pr-3 pl-3" @click="bulkDelete('/admin/categories/bulk-destroy')">{{ trans('brackets/admin-ui::admin.btn.delete') }}</button>
65         </span>
66
67     </td>
68 </tr>
69 </thead>
70 <tbody>
71     <tr v-for="(item, index) in collection" :key="item.id" :class="bulkItems[item.id] ? 'bg-bulk' : ''">
72         <td class="bulk-checkbox">
73             <input class="form-check-input" :id="`enabled` + item.id" type="checkbox" v-model="bulkItems[item.id]" v-validate="" :data-vv-name="`enabled` + item.id" checked="">
74             <label class="form-check-label" :for="`enabled` + item.id">{{ item.enabled }}</label>
75         </td>
76
77         <td>@{{ item.id }}</td>
78         <td>@{{ item.category }}</td>
79         <td>@{{ item.category_slug }}</td>
80
81         <td>
82             <div class="row no-gutters">
83                 <div class="col-auto">
84
85
index.blade.php X
resources > views > admin > category > index.blade.php > ...
85
86
87         <div class="col" style="text-align: right; padding-right: 10px;">
88             <a class="btn btn-sm btn-spinner btn-info" :href="item.resource_url + '/edit'" title="{{ trans('brackets/admin-ui::admin.btn.edit') }}>{{ item.name }}</a>
89         </div>
90         <form class="col" @submit.prevent="deleteItem(item.resource_url)">
91             <button type="submit" class="btn btn-sm btn-danger" title="{{ trans('brackets/admin-ui::admin.btn.delete') }}>{{ item.name }}</button>
92         </form>
93     </td>
94 </tr>
95 </tbody>
96 </table>
97
98 <div class="row" v-if="pagination.state.total > 0">
99     <div class="col-sm">
100         <span class="pagination-caption">{{ trans('brackets/admin-ui::admin.pagination.overview') }}</span>
101     </div>
102     <div class="col-sm-auto">
103         <pagination></pagination>
104     </div>
105 </div>
106 <div class="no-items-found" v-if="!collection.length > 0">
107     <i class="icon-magnifier"></i>
108     <h3>{{ trans('brackets/admin-ui::admin.index.no_items') }}</h3>
109     <p>{{ trans('brackets/admin-ui::admin.index.try_changing_items') }}</p>
110     <a class="btn btn-primary btn-spinner" href="{{ url('admin/categories/create') }}" role="button"><i class="fa fa-plus"></i>&ampnbsp {{ trans('brackets/admin-ui::admin.button.create') }}</a>
111 </div>
112 </div>
113 </div>
114 </div>
115 </div>
116 </category-listing>
117
118 @endsection

```

Figure 59: Add Category.php

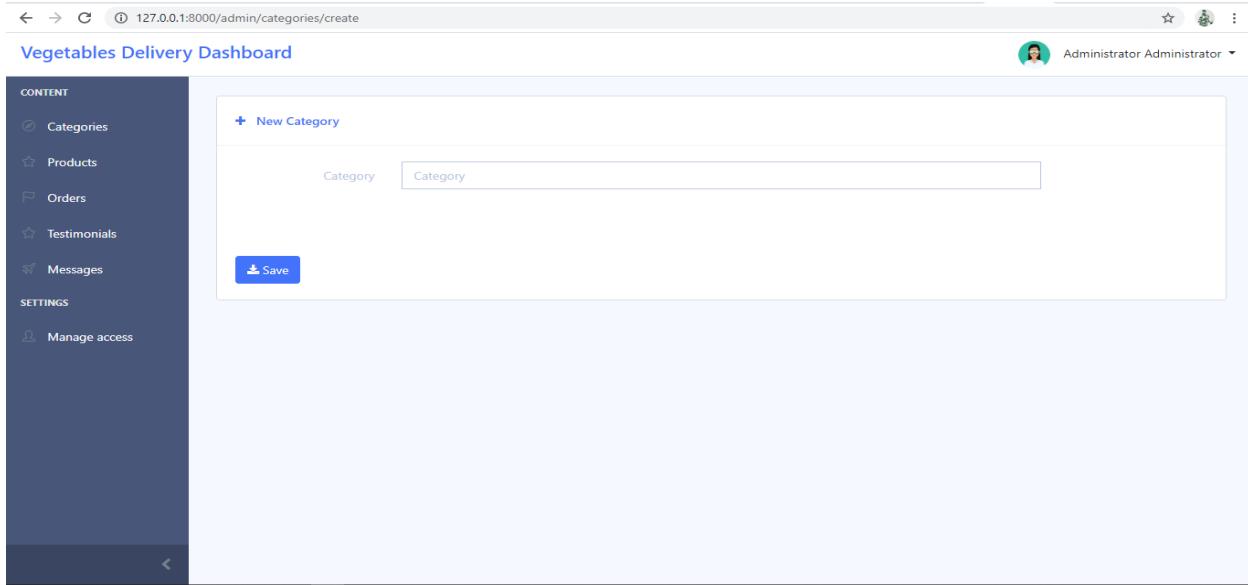
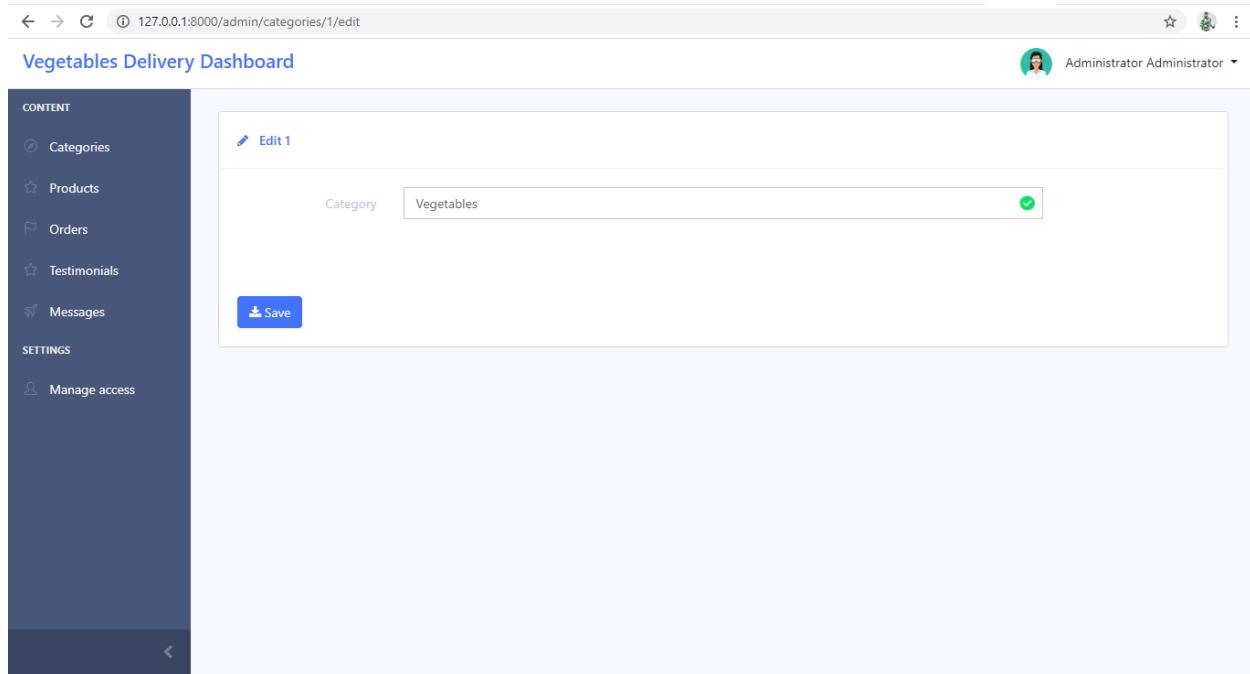


Figure 60: New Category Form Page

```
create.blade.php X
resources > views > admin > category > create.blade.php > ...
1  @extends('brackets/admin-ui::admin.layout.default')
2
3  @section('title', trans('admin.category.actions.create'))
4
5  @section('body')
6
7      <div class="container-xl">
8
9          |     <div class="card">
10
11             <category-form
12                 :action="{!! url('admin/categories') !!}"
13                 v-cloak
14                 inline-template>
15
16                 <form class="form-horizontal form-create" method="post" @submit.prevent="onSubmit" :action="action" novalidate>
17
18                     <div class="card-header">
19                         |     <i class="fa fa-plus"></i> {{ trans('admin.category.actions.create') }}
20                     </div>
21
22                     <div class="card-body">
23                         |     @include('admin.category.components.form-elements')
24                     </div>
25
26                     <div class="card-footer">
27                         <button type="submit" class="btn btn-primary" :disabled="submiting">
28                             |     <i class="fa" :class="submiting ? 'fa-spinner' : 'fa-download'"></i>
29                             |     {{ trans('brackets/admin-ui::admin.btn.save') }}
30                         </button>
31                     </div>
32
33             </form>
34
35         </category-form>
36
37     </div>
38
39     </div>
40
41
42 @endsection
```

```
create.blade.php X
resources > views > admin > category > create.blade.php > ...
33     </form>
34
35     </category-form>
36
37     </div>
38
39     </div>
40
41
42 @endsection
```

Figure 61: Add New Category.php

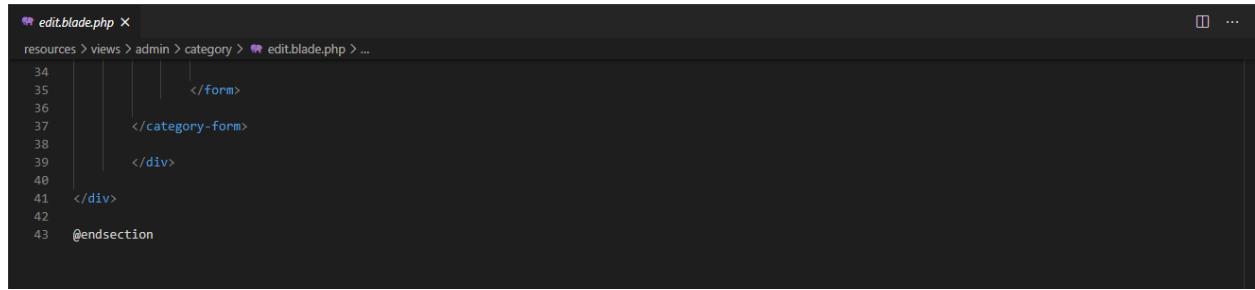


*Figure 62: Edit Category*

```
edit.blade.php x

resources > views > admin > category > edit.blade.php > ...
1  @extends('brackets/admin-ui::admin.layout.default')
2
3  @section('title', trans('admin.category.actions.edit', ['name' => $category->id]))
4
5  @section('body')
6
7      <div class="container-xl">
8          <div class="card">
9
10             <category-form
11                 :action="{{ $category->resource_url }}"
12                 :data="{{ $category->toJson() }}"
13                 v-cloak
14                 inline-template>
15
16                 <form class="form-horizontal form-edit" method="post" @submit.prevent="onSubmit" :action="action" novalidate>
17
18                     <div class="card-header">
19                         <i class="fa fa-pencil"></i> {{ trans('admin.category.actions.edit', ['name' => $category->id]) }}
20                     </div>
21
22
23                     <div class="card-body">
24                         @include('admin.category.components.form-elements')
25                     </div>
26
27
28                     <div class="card-footer">
29                         <button type="submit" class="btn btn-primary" :disabled="submiting">
30                             <i class="fa" :class="submiting ? 'fa-spinner' : 'fa-download'"></i>
31                             {{ trans('brackets/admin-ui::admin.btn.save') }}
32                         </button>
33                     </div>
34

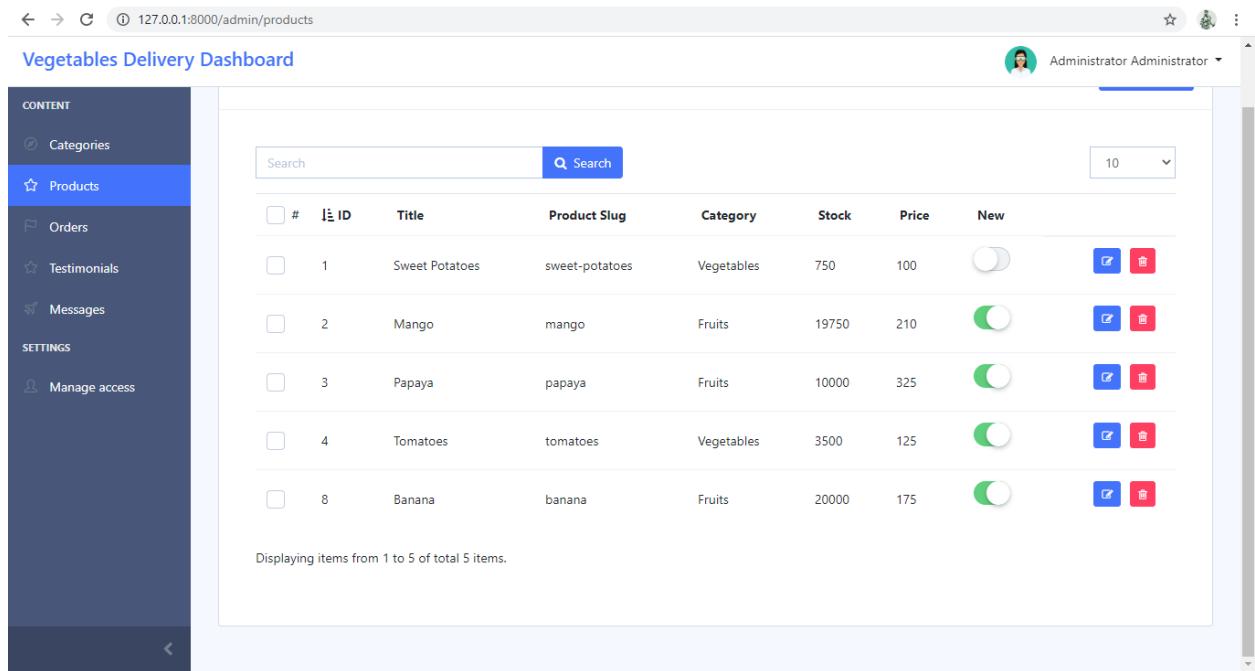
```



The screenshot shows a code editor window with the file 'edit.blade.php' open. The code is a Blade template for editing a category. It includes sections for form opening, category form, and closing divs, followed by an end section. The code is numbered from 34 to 43.

```
34 | </form>
35 |
36 |     </category-form>
37 |
38 |     </div>
39 |
40 | </div>
41 |
42 | @endsection
```

Figure 63: Edit Category.php



The screenshot shows a web browser displaying the 'Vegetables Delivery Dashboard'. The URL is 127.0.0.1:8000/admin/products. The left sidebar has 'CONTENT' and 'SETTINGS' sections with links for Categories, Products (which is selected), Orders, Testimonials, Messages, Manage access, and a back arrow. The main area is titled 'Vegetables Delivery Dashboard' and shows a table of products. The table columns are #, ID, Title, Product Slug, Category, Stock, Price, and New. The products listed are:

#	ID	Title	Product Slug	Category	Stock	Price	New
1	1	Sweet Potatoes	sweet-potatoes	Vegetables	750	100	<input checked="" type="checkbox"/> <input type="button" value="Edit"/>
2	2	Mango	mango	Fruits	19750	210	<input checked="" type="checkbox"/> <input type="button" value="Edit"/>
3	3	Papaya	papaya	Fruits	10000	325	<input checked="" type="checkbox"/> <input type="button" value="Edit"/>
4	4	Tomatoes	tomatoes	Vegetables	3500	125	<input checked="" type="checkbox"/> <input type="button" value="Edit"/>
8	8	Banana	banana	Fruits	20000	175	<input checked="" type="checkbox"/> <input type="button" value="Edit"/>

Below the table, a message says 'Displaying items from 1 to 5 of total 5 items.'

Figure 64: Product Page

```
  index.blade.php X
resources > views > admin > product >   index.blade.php > ...
1  @extends('brackets/admin-ui::admin.layout.default')
2
3  @section('title', trans('admin.product.actions.index'))
4
5  @section('body')
6
7      <product-listing
8          :data="{!! $data->toJson() !!}"
9          :url="{!! url('admin/products') !!}"
10         inline-template>
11
12      <div class="row">
13          <div class="col">
14              <div class="card">
15                  <div class="card-header">
16                      <i class="fa fa-align-justify"></i> {{ trans('admin.product.actions.index') }}
17                      <a class="btn btn-primary btn-spinner btn-sm pull-right m-b-0" href="{!! url('admin/products/create') !!}" role="button"><i class="fa fa-plus"></i> {{ trans('admin.add_new') }}
18                  </div>
19                  <div class="card-body" v-cloak>
20                      <div class="card-block">
21                          <form @submit.prevent="">
22                              <div class="row justify-content-md-between">
23                                  <div class="col col-lg-7 col-xl-5 form-group">
24                                      <div class="input-group">
25                                          <input class="form-control" placeholder="{!! trans('brackets/admin-ui::admin.placeholder.search') !!}" v-model="search" @input="filter('search', search)" type="text" />
26                                          <span class="input-group-append">
27                                              <button type="button" class="btn btn-primary" @click="filter('search', search)"><i class="fa fa-search"></i>&ampnbsp</button>
28                                          </span>
29                                      </div>
30                                  <div class="col-sm-auto form-group">
31                                      <select class="form-control" v-model="pagination.state.per_page">
32                                          <option value="10">10</option>
33                                          <option value="25">25</option>
34                                          <option value="100">100</option>
35                                      </select>
36                                  </div>
37                              </div>
38                          </div>
39                      </div>
40                  </div>
41
42                  <table class="table table-hover table-listing">
43                      <thead>
44                          <tr>
45                              <th class="bulk-checkbox">
46                                  <input class="form-check-input" id="enabled" type="checkbox" v-model="isClickedAll" v-validate="" data-vv-name="enabled" name="enabled_fake" checked="" />
47                                  <label class="form-check-label" for="enabled">
48                                      # {{ trans('admin.product.columns.enabled') }}
49                                  </label>
50                              </th>
51
52                              <th is="sortable" :column="'id'">{{ trans('admin.product.columns.id') }}</th>
53                              <th is="sortable" :column="'title'">{{ trans('admin.product.columns.title') }}</th>
54                              <th is="sortable" :column="'slug'">Product Slug</th>
55                              <th is="sortable" :column="'category_id'">{{ trans('admin.product.columns.category_id') }}</th>
56                              <th is="sortable" :column="'stock'">{{ trans('admin.product.columns.stock') }}</th>
57                              <th is="sortable" :column="'price'">{{ trans('admin.product.columns.price') }}</th>
58
59                              <!-- <th is="sortable" :column="'without_discount_price'">{{ trans('admin.product.columns.without_discount_price') }}</th> -->
60                              <th is="sortable" :column="'new'">{{ trans('admin.product.columns.new') }}</th>
61
62                          <th></th>
63                      </tr>
64                      <tr v-show="(ClickedBulkItemsCount > 0) || isClickedAll">
65                          <td class="bg-bulk-info d-table-cell text-center" colspan="9">
66                              <span class="align-middle font-weight-light text-dark">{{ trans('brackets/admin-ui::admin.listing.selected_items') }} @{{ ClickedBulkItemsCount }}</span>
67                              <span href="#" class="text-primary" @click="onBulkItemsClickedAllUncheck()">{{ trans('brackets/admin-ui::admin.listing.uncheck_all_items') }}</span>
68                          </td>
69                      </tr>
70                  </thead>
71                  <tbody>
72                      <tr v-for="product in products" :key="product.id">
73                          <td>{{ product.id }}</td>
74                          <td>{{ product.title }}</td>
75                          <td>{{ product.slug }}</td>
76                          <td>{{ product.category_id }}</td>
77                          <td>{{ product.stock }}</td>
78                          <td>{{ product.price }}</td>
79                          <td>{{ product.without_discount_price }}</td>
80                          <td>{{ product.new }}</td>
81                          <td>{{ product.enabled }}</td>
82                      </tr>
83                  </tbody>
84              </table>
85          </div>
86      </div>
87  </div>
88
89  <script>
90      window.onload = function() {
91          var table = document.querySelector('.table-listing');
92          var tbody = table.querySelector('tbody');
93          var rows = tbody.querySelectorAll('tr');
94
95          for (var i = 0; i < rows.length; i++) {
96              var row = rows[i];
97              var cells = row.querySelectorAll('td');
98
99              for (var j = 0; j < cells.length; j++) {
100                  var cell = cells[j];
101
102                  if (cell.classList.contains('bulk-checkbox')) {
103                      var checkbox = cell.querySelector('input');
104
105                      checkbox.addEventListener('change', function() {
106                          var checked = this.checked;
107
108                          var allCheckboxes = table.querySelectorAll('input[type="checkbox"]');
109
110                          for (var k = 0; k < allCheckboxes.length; k++) {
111                              var checkbox = allCheckboxes[k];
112
113                              if (checkbox !== this) {
114                                  checkbox.checked = checked;
115                              }
116                          }
117
118                          var selectedCount = 0;
119
120                          for (var l = 0; l < allCheckboxes.length; l++) {
121                              var checkbox = allCheckboxes[l];
122
123                              if (checkbox.checked) {
124                                  selectedCount++;
125                              }
126                          }
127
128                          var selectedText = selectedCount + ' ' + trans('admin.product.columns.selected_items');
129
130                          var uncheckAllText = trans('admin.product.actions.uncheck_all_items');
131
132                          var uncheckAllLink = document.createElement('a');
133                          uncheckAllLink.href = '#';
134                          uncheckAllLink.className = 'text-primary';
135                          uncheckAllLink.textContent = uncheckAllText;
136
137                          var uncheckAllSpan = document.createElement('span');
138                          uncheckAllSpan.className = 'text-primary';
139                          uncheckAllSpan.textContent = ' ' + uncheckAllText;
140
141                          var uncheckAllDiv = document.createElement('div');
142                          uncheckAllDiv.className = 'd-table-cell align-middle';
143                          uncheckAllDiv.textContent = '';
144
145                          uncheckAllDiv.appendChild(uncheckAllLink);
146                          uncheckAllDiv.appendChild(uncheckAllSpan);
147
148                          var uncheckAllTd = document.createElement('td');
149                          uncheckAllTd.className = 'bg-bulk-info d-table-cell text-center';
150                          uncheckAllTd.textContent = '';
151
152                          uncheckAllTd.appendChild(uncheckAllDiv);
153
154                          var uncheckAllTr = document.createElement('tr');
155                          uncheckAllTr.className = 'bg-bulk-info';
156                          uncheckAllTr.textContent = '';
157
158                          uncheckAllTr.appendChild(uncheckAllTd);
159
160                          table.appendChild(uncheckAllTr);
161
162                          var uncheckAllTdText = document.createElement('td');
163                          uncheckAllTdText.textContent = '';
164
165                          var uncheckAllTrText = document.createElement('tr');
166                          uncheckAllTrText.textContent = '';
167
168                          uncheckAllTrText.appendChild(uncheckAllTdText);
169
170                          table.appendChild(uncheckAllTrText);
171
172                          var uncheckAllText = document.createElement('div');
173                          uncheckAllText.className = 'd-table-cell align-middle';
174                          uncheckAllText.textContent = '';
175
176                          uncheckAllText.appendChild(uncheckAllLink);
177                          uncheckAllText.appendChild(uncheckAllSpan);
178
179                          var uncheckAllTdText2 = document.createElement('td');
180                          uncheckAllTdText2.textContent = '';
181
182                          var uncheckAllTrText2 = document.createElement('tr');
183                          uncheckAllTrText2.textContent = '';
184
185                          uncheckAllTrText2.appendChild(uncheckAllTdText2);
186
187                          table.appendChild(uncheckAllTrText2);
188
189                          var uncheckAllText2 = document.createElement('div');
190                          uncheckAllText2.className = 'd-table-cell align-middle';
191                          uncheckAllText2.textContent = '';
192
193                          uncheckAllText2.appendChild(uncheckAllLink);
194                          uncheckAllText2.appendChild(uncheckAllSpan);
195
196                          var uncheckAllTdText3 = document.createElement('td');
197                          uncheckAllTdText3.textContent = '';
198
199                          var uncheckAllTrText3 = document.createElement('tr');
200                          uncheckAllTrText3.textContent = '';
201
202                          uncheckAllTrText3.appendChild(uncheckAllTdText3);
203
204                          table.appendChild(uncheckAllTrText3);
205
206                          var uncheckAllText3 = document.createElement('div');
207                          uncheckAllText3.className = 'd-table-cell align-middle';
208                          uncheckAllText3.textContent = '';
209
210                          uncheckAllText3.appendChild(uncheckAllLink);
211                          uncheckAllText3.appendChild(uncheckAllSpan);
212
213                          var uncheckAllTdText4 = document.createElement('td');
214                          uncheckAllTdText4.textContent = '';
215
216                          var uncheckAllTrText4 = document.createElement('tr');
217                          uncheckAllTrText4.textContent = '';
218
219                          uncheckAllTrText4.appendChild(uncheckAllTdText4);
220
221                          table.appendChild(uncheckAllTrText4);
222
223                          var uncheckAllText4 = document.createElement('div');
224                          uncheckAllText4.className = 'd-table-cell align-middle';
225                          uncheckAllText4.textContent = '';
226
227                          uncheckAllText4.appendChild(uncheckAllLink);
228                          uncheckAllText4.appendChild(uncheckAllSpan);
229
230                          var uncheckAllTdText5 = document.createElement('td');
231                          uncheckAllTdText5.textContent = '';
232
233                          var uncheckAllTrText5 = document.createElement('tr');
234                          uncheckAllTrText5.textContent = '';
235
236                          uncheckAllTrText5.appendChild(uncheckAllTdText5);
237
238                          table.appendChild(uncheckAllTrText5);
239
240                          var uncheckAllText5 = document.createElement('div');
241                          uncheckAllText5.className = 'd-table-cell align-middle';
242                          uncheckAllText5.textContent = '';
243
244                          uncheckAllText5.appendChild(uncheckAllLink);
245                          uncheckAllText5.appendChild(uncheckAllSpan);
246
247                          var uncheckAllTdText6 = document.createElement('td');
248                          uncheckAllTdText6.textContent = '';
249
250                          var uncheckAllTrText6 = document.createElement('tr');
251                          uncheckAllTrText6.textContent = '';
252
253                          uncheckAllTrText6.appendChild(uncheckAllTdText6);
254
255                          table.appendChild(uncheckAllTrText6);
256
257                          var uncheckAllText6 = document.createElement('div');
258                          uncheckAllText6.className = 'd-table-cell align-middle';
259                          uncheckAllText6.textContent = '';
260
261                          uncheckAllText6.appendChild(uncheckAllLink);
262                          uncheckAllText6.appendChild(uncheckAllSpan);
263
264                          var uncheckAllTdText7 = document.createElement('td');
265                          uncheckAllTdText7.textContent = '';
266
267                          var uncheckAllTrText7 = document.createElement('tr');
268                          uncheckAllTrText7.textContent = '';
269
270                          uncheckAllTrText7.appendChild(uncheckAllTdText7);
271
272                          table.appendChild(uncheckAllTrText7);
273
274                          var uncheckAllText7 = document.createElement('div');
275                          uncheckAllText7.className = 'd-table-cell align-middle';
276                          uncheckAllText7.textContent = '';
277
278                          uncheckAllText7.appendChild(uncheckAllLink);
279                          uncheckAllText7.appendChild(uncheckAllSpan);
280
281                          var uncheckAllTdText8 = document.createElement('td');
282                          uncheckAllTdText8.textContent = '';
283
284                          var uncheckAllTrText8 = document.createElement('tr');
285                          uncheckAllTrText8.textContent = '';
286
287                          uncheckAllTrText8.appendChild(uncheckAllTdText8);
288
289                          table.appendChild(uncheckAllTrText8);
290
291                          var uncheckAllText8 = document.createElement('div');
292                          uncheckAllText8.className = 'd-table-cell align-middle';
293                          uncheckAllText8.textContent = '';
294
295                          uncheckAllText8.appendChild(uncheckAllLink);
296                          uncheckAllText8.appendChild(uncheckAllSpan);
297
298                          var uncheckAllTdText9 = document.createElement('td');
299                          uncheckAllTdText9.textContent = '';
300
301                          var uncheckAllTrText9 = document.createElement('tr');
302                          uncheckAllTrText9.textContent = '';
303
304                          uncheckAllTrText9.appendChild(uncheckAllTdText9);
305
306                          table.appendChild(uncheckAllTrText9);
307
308                          var uncheckAllText9 = document.createElement('div');
309                          uncheckAllText9.className = 'd-table-cell align-middle';
310                          uncheckAllText9.textContent = '';
311
312                          uncheckAllText9.appendChild(uncheckAllLink);
313                          uncheckAllText9.appendChild(uncheckAllSpan);
314
315                          var uncheckAllTdText10 = document.createElement('td');
316                          uncheckAllTdText10.textContent = '';
317
318                          var uncheckAllTrText10 = document.createElement('tr');
319                          uncheckAllTrText10.textContent = '';
320
321                          uncheckAllTrText10.appendChild(uncheckAllTdText10);
322
323                          table.appendChild(uncheckAllTrText10);
324
325                          var uncheckAllText10 = document.createElement('div');
326                          uncheckAllText10.className = 'd-table-cell align-middle';
327                          uncheckAllText10.textContent = '';
328
329                          uncheckAllText10.appendChild(uncheckAllLink);
330                          uncheckAllText10.appendChild(uncheckAllSpan);
331
332                          var uncheckAllTdText11 = document.createElement('td');
333                          uncheckAllTdText11.textContent = '';
334
335                          var uncheckAllTrText11 = document.createElement('tr');
336                          uncheckAllTrText11.textContent = '';
337
338                          uncheckAllTrText11.appendChild(uncheckAllTdText11);
339
340                          table.appendChild(uncheckAllTrText11);
341
342                          var uncheckAllText11 = document.createElement('div');
343                          uncheckAllText11.className = 'd-table-cell align-middle';
344                          uncheckAllText11.textContent = '';
345
346                          uncheckAllText11.appendChild(uncheckAllLink);
347                          uncheckAllText11.appendChild(uncheckAllSpan);
348
349                          var uncheckAllTdText12 = document.createElement('td');
350                          uncheckAllTdText12.textContent = '';
351
352                          var uncheckAllTrText12 = document.createElement('tr');
353                          uncheckAllTrText12.textContent = '';
354
355                          uncheckAllTrText12.appendChild(uncheckAllTdText12);
356
357                          table.appendChild(uncheckAllTrText12);
358
359                          var uncheckAllText12 = document.createElement('div');
360                          uncheckAllText12.className = 'd-table-cell align-middle';
361                          uncheckAllText12.textContent = '';
362
363                          uncheckAllText12.appendChild(uncheckAllLink);
364                          uncheckAllText12.appendChild(uncheckAllSpan);
365
366                          var uncheckAllTdText13 = document.createElement('td');
367                          uncheckAllTdText13.textContent = '';
368
369                          var uncheckAllTrText13 = document.createElement('tr');
370                          uncheckAllTrText13.textContent = '';
371
372                          uncheckAllTrText13.appendChild(uncheckAllTdText13);
373
374                          table.appendChild(uncheckAllTrText13);
375
376                          var uncheckAllText13 = document.createElement('div');
377                          uncheckAllText13.className = 'd-table-cell align-middle';
378                          uncheckAllText13.textContent = '';
379
380                          uncheckAllText13.appendChild(uncheckAllLink);
381                          uncheckAllText13.appendChild(uncheckAllSpan);
382
383                          var uncheckAllTdText14 = document.createElement('td');
384                          uncheckAllTdText14.textContent = '';
385
386                          var uncheckAllTrText14 = document.createElement('tr');
387                          uncheckAllTrText14.textContent = '';
388
389                          uncheckAllTrText14.appendChild(uncheckAllTdText14);
390
391                          table.appendChild(uncheckAllTrText14);
392
393                          var uncheckAllText14 = document.createElement('div');
394                          uncheckAllText14.className = 'd-table-cell align-middle';
395                          uncheckAllText14.textContent = '';
396
397                          uncheckAllText14.appendChild(uncheckAllLink);
398                          uncheckAllText14.appendChild(uncheckAllSpan);
399
400                          var uncheckAllTdText15 = document.createElement('td');
401                          uncheckAllTdText15.textContent = '';
402
403                          var uncheckAllTrText15 = document.createElement('tr');
404                          uncheckAllTrText15.textContent = '';
405
406                          uncheckAllTrText15.appendChild(uncheckAllTdText15);
407
408                          table.appendChild(uncheckAllTrText15);
409
410                          var uncheckAllText15 = document.createElement('div');
411                          uncheckAllText15.className = 'd-table-cell align-middle';
412                          uncheckAllText15.textContent = '';
413
414                          uncheckAllText15.appendChild(uncheckAllLink);
415                          uncheckAllText15.appendChild(uncheckAllSpan);
416
417                          var uncheckAllTdText16 = document.createElement('td');
418                          uncheckAllTdText16.textContent = '';
419
420                          var uncheckAllTrText16 = document.createElement('tr');
421                          uncheckAllTrText16.textContent = '';
422
423                          uncheckAllTrText16.appendChild(uncheckAllTdText16);
424
425                          table.appendChild(uncheckAllTrText16);
426
427                          var uncheckAllText16 = document.createElement('div');
428                          uncheckAllText16.className = 'd-table-cell align-middle';
429                          uncheckAllText16.textContent = '';
430
431                          uncheckAllText16.appendChild(uncheckAllLink);
432                          uncheckAllText16.appendChild(uncheckAllSpan);
433
434                          var uncheckAllTdText17 = document.createElement('td');
435                          uncheckAllTdText17.textContent = '';
436
437                          var uncheckAllTrText17 = document.createElement('tr');
438                          uncheckAllTrText17.textContent = '';
439
440                          uncheckAllTrText17.appendChild(uncheckAllTdText17);
441
442                          table.appendChild(uncheckAllTrText17);
443
444                          var uncheckAllText17 = document.createElement('div');
445                          uncheckAllText17.className = 'd-table-cell align-middle';
446                          uncheckAllText17.textContent = '';
447
448                          uncheckAllText17.appendChild(uncheckAllLink);
449                          uncheckAllText17.appendChild(uncheckAllSpan);
450
451                          var uncheckAllTdText18 = document.createElement('td');
452                          uncheckAllTdText18.textContent = '';
453
454                          var uncheckAllTrText18 = document.createElement('tr');
455                          uncheckAllTrText18.textContent = '';
456
457                          uncheckAllTrText18.appendChild(uncheckAllTdText18);
458
459                          table.appendChild(uncheckAllTrText18);
460
461                          var uncheckAllText18 = document.createElement('div');
462                          uncheckAllText18.className = 'd-table-cell align-middle';
463                          uncheckAllText18.textContent = '';
464
465                          uncheckAllText18.appendChild(uncheckAllLink);
466                          uncheckAllText18.appendChild(uncheckAllSpan);
467
468                          var uncheckAllTdText19 = document.createElement('td');
469                          uncheckAllTdText19.textContent = '';
470
471                          var uncheckAllTrText19 = document.createElement('tr');
472                          uncheckAllTrText19.textContent = '';
473
474                          uncheckAllTrText19.appendChild(uncheckAllTdText19);
475
476                          table.appendChild(uncheckAllTrText19);
477
478                          var uncheckAllText19 = document.createElement('div');
479                          uncheckAllText19.className = 'd-table-cell align-middle';
480                          uncheckAllText19.textContent = '';
481
482                          uncheckAllText19.appendChild(uncheckAllLink);
483                          uncheckAllText19.appendChild(uncheckAllSpan);
484
485                          var uncheckAllTdText20 = document.createElement('td');
486                          uncheckAllTdText20.textContent = '';
487
488                          var uncheckAllTrText20 = document.createElement('tr');
489                          uncheckAllTrText20.textContent = '';
490
491                          uncheckAllTrText20.appendChild(uncheckAllTdText20);
492
493                          table.appendChild(uncheckAllTrText20);
494
495                          var uncheckAllText20 = document.createElement('div');
496                          uncheckAllText20.className = 'd-table-cell align-middle';
497                          uncheckAllText20.textContent = '';
498
499                          uncheckAllText20.appendChild(uncheckAllLink);
500                          uncheckAllText20.appendChild(uncheckAllSpan);
501
502                          var uncheckAllTdText21 = document.createElement('td');
503                          uncheckAllTdText21.textContent = '';
504
505                          var uncheckAllTrText21 = document.createElement('tr');
506                          uncheckAllTrText21.textContent = '';
507
508                          uncheckAllTrText21.appendChild(uncheckAllTdText21);
509
510                          table.appendChild(uncheckAllTrText21);
511
512                          var uncheckAllText21 = document.createElement('div');
513                          uncheckAllText21.className = 'd-table-cell align-middle';
514                          uncheckAllText21.textContent = '';
515
516                          uncheckAllText21.appendChild(uncheckAllLink);
517                          uncheckAllText21.appendChild(uncheckAllSpan);
518
519                          var uncheckAllTdText22 = document.createElement('td');
520                          uncheckAllTdText22.textContent = '';
521
522                          var uncheckAllTrText22 = document.createElement('tr');
523                          uncheckAllTrText22.textContent = '';
524
525                          uncheckAllTrText22.appendChild(uncheckAllTdText22);
526
527                          table.appendChild(uncheckAllTrText22);
528
529                          var uncheckAllText22 = document.createElement('div');
530                          uncheckAllText22.className = 'd-table-cell align-middle';
531                          uncheckAllText22.textContent = '';
532
533                          uncheckAllText22.appendChild(uncheckAllLink);
534                          uncheckAllText22.appendChild(uncheckAllSpan);
535
536                          var uncheckAllTdText23 = document.createElement('td');
537                          uncheckAllTdText23.textContent = '';
538
539                          var uncheckAllTrText23 = document.createElement('tr');
540                          uncheckAllTrText23.textContent = '';
541
542                          uncheckAllTrText23.appendChild(uncheckAllTdText23);
543
544                          table.appendChild(uncheckAllTrText23);
545
546                          var uncheckAllText23 = document.createElement('div');
547                          uncheckAllText23.className = 'd-table-cell align-middle';
548                          uncheckAllText23.textContent = '';
549
550                          uncheckAllText23.appendChild(uncheckAllLink);
551                          uncheckAllText23.appendChild(uncheckAllSpan);
552
553                          var uncheckAllTdText24 = document.createElement('td');
554                          uncheckAllTdText24.textContent = '';
555
556                          var uncheckAllTrText24 = document.createElement('tr');
557                          uncheckAllTrText24.textContent = '';
558
559                          uncheckAllTrText24.appendChild(uncheckAllTdText24);
560
561                          table.appendChild(uncheckAllTrText24);
562
563                          var uncheckAllText24 = document.createElement('div');
564                          uncheckAllText24.className = 'd-table-cell align-middle';
565                          uncheckAllText24.textContent = '';
566
567                          uncheckAllText24.appendChild(uncheckAllLink);
568                          uncheckAllText24.appendChild(uncheckAllSpan);
569
570                          var uncheckAllTdText25 = document.createElement('td');
571                          uncheckAllTdText25.textContent = '';
572
573                          var uncheckAllTrText25 = document.createElement('tr');
574                          uncheckAllTrText25.textContent = '';
575
576                          uncheckAllTrText25.appendChild(uncheckAllTdText25);
577
578                          table.appendChild(uncheckAllTrText25);
579
580                          var uncheckAllText25 = document.createElement('div');
581                          uncheckAllText25.className = 'd-table-cell align-middle';
582                          uncheckAllText25.textContent = '';
583
584                          uncheckAllText25.appendChild(uncheckAllLink);
585                          uncheckAllText25.appendChild(uncheckAllSpan);
586
587                          var uncheckAllTdText26 = document.createElement('td');
588                          uncheckAllTdText26.textContent = '';
589
590                          var uncheckAllTrText26 = document.createElement('tr');
591                          uncheckAllTrText26.textContent = '';
592
593                          uncheckAllTrText26.appendChild(uncheckAllTdText26);
594
595                          table.appendChild(uncheckAllTrText26);
596
597                          var uncheckAllText26 = document.createElement('div');
598                          uncheckAllText26.className = 'd-table-cell align-middle';
599                          uncheckAllText26.textContent = '';
600
601                          uncheckAllText26.appendChild(uncheckAllLink);
602                          uncheckAllText26.appendChild(uncheckAllSpan);
603
604                          var uncheckAllTdText27 = document.createElement('td');
605                          uncheckAllTdText27.textContent = '';
606
607                          var uncheckAllTrText27 = document.createElement('tr');
608                          uncheckAllTrText27.textContent = '';
609
610                          uncheckAllTrText27.appendChild(uncheckAllTdText27);
611
612                          table.appendChild(uncheckAllTrText27);
613
614                          var uncheckAllText27 = document.createElement('div');
615                          uncheckAllText27.className = 'd-table-cell align-middle';
616                          uncheckAllText27.textContent = '';
617
618                          uncheckAllText27.appendChild(uncheckAllLink);
619                          uncheckAllText27.appendChild(uncheckAllSpan);
620
621                          var uncheckAllTdText28 = document.createElement('td');
622                          uncheckAllTdText28.textContent = '';
623
624                          var uncheckAllTrText28 = document.createElement('tr');
625                          uncheckAllTrText28.textContent = '';
626
627                          uncheckAllTrText28.appendChild(uncheckAllTdText28);
628
629                          table.appendChild(uncheckAllTrText28);
630
631                          var uncheckAllText28 = document.createElement('div');
632                          uncheckAllText28.className = 'd-table-cell align-middle';
633                          uncheckAllText28.textContent = '';
634
635                          uncheckAllText28.appendChild(uncheckAllLink);
636                          uncheckAllText28.appendChild(uncheckAllSpan);
637
638                          var uncheckAllTdText29 = document.createElement('td');
639                          uncheckAllTdText29.textContent = '';
640
641                          var uncheckAllTrText29 = document.createElement('tr');
642                          uncheckAllTrText29.textContent = '';
643
644                          uncheckAllTrText29.appendChild(uncheckAllTdText29);
645
646                          table.appendChild(uncheckAllTrText29);
647
648                          var uncheckAllText29 = document.createElement('div');
649                          uncheckAllText29.className = 'd-table-cell align-middle';
650                          uncheckAllText29.textContent = '';
651
652                          uncheckAllText29.appendChild(uncheckAllLink);
653                          uncheckAllText29.appendChild(uncheckAllSpan);
654
655                          var uncheckAllTdText30 = document.createElement('td');
656                          uncheckAllTdText30.textContent = '';
657
658                          var uncheckAllTrText30 = document.createElement('tr');
659                          uncheckAllTrText30.textContent = '';
660
661                          uncheckAllTrText30.appendChild(uncheckAllTdText30);
662
663                          table.appendChild(uncheckAllTrText30);
664
665                          var uncheckAllText30 = document.createElement('div');
666                          uncheckAllText30.className = 'd-table-cell align-middle';
667                          uncheckAllText30.textContent = '';
668
669                          uncheckAllText30.appendChild(uncheckAllLink);
670                          uncheckAllText30.appendChild(uncheckAllSpan);
671
672                          var uncheckAllTdText31 = document.createElement('td');
673                          uncheckAllTdText31.textContent = '';
674
675                          var uncheckAllTrText31 = document.createElement('tr');
676                          uncheckAllTrText31.textContent = '';
677
678                          uncheckAllTrText31.appendChild(uncheckAllTdText31);
679
680                          table.appendChild(uncheckAllTrText31);
681
682                          var uncheckAllText31 = document.createElement('div');
683                          uncheckAllText31.className = 'd-table-cell align-middle';
684                          uncheckAllText31.textContent = '';
685
686                          uncheckAllText31.appendChild(uncheckAllLink);
687                          uncheckAllText31.appendChild(uncheckAllSpan);
688
689                          var uncheckAllTdText32 = document.createElement('td');
690                          uncheckAllTdText32.textContent = '';
691
692                          var uncheckAllTrText32 = document.createElement('tr');
693                          uncheckAllTrText32.textContent = '';
694
695                          uncheckAllTrText32.appendChild(uncheckAllTdText32);
696
697                          table.appendChild(uncheckAllTrText32);
698
699                          var uncheckAllText32 = document.createElement('div');
700                          uncheckAllText32.className = 'd-table-cell align-middle';
701                          uncheckAllText32.textContent = '';
702
703                          uncheckAllText32.appendChild(uncheckAllLink);
704                          uncheckAllText32.appendChild(uncheckAllSpan);
705
706                          var uncheckAllTdText33 = document.createElement('td');
707                          uncheckAllTdText33.textContent = '';
708
709                          var uncheckAllTrText33 = document.createElement('tr');
710                          uncheckAllTrText33.textContent = '';
711
712                          uncheckAllTrText33.appendChild(uncheckAllTdText33);
713
714                          table.appendChild(uncheckAllTrText33);
715
716                          var uncheckAllText33 = document.createElement('div');
717                          uncheckAllText33.className = 'd-table-cell align-middle';
718                          uncheckAllText33.textContent = '';
719
720                          uncheckAllText33.appendChild(uncheckAllLink);
721                          uncheckAllText33.appendChild(uncheckAllSpan);
722
723                          var uncheckAllTdText34 = document.createElement('td');
724                          uncheckAllTdText34.textContent = '';
725
726                          var uncheckAllTrText34 = document.createElement('tr');
727                          uncheckAllTrText34.textContent = '';
728
729                          uncheckAllTrText34.appendChild(uncheckAllTdText34);
730
731                          table.appendChild(uncheckAllTrText34);
732
733                          var uncheckAllText34 = document.createElement('div');
734                          uncheckAllText34.className = 'd-table-cell align-middle';
735                          uncheckAllText34.textContent = '';
736
737                          uncheckAllText34.appendChild(uncheckAllLink);
738                          uncheckAllText34.appendChild(uncheckAllSpan);
739
740                          var uncheckAllTdText35 = document.createElement('td');
741                          uncheckAllTdText35.textContent = '';
742
743                          var uncheckAllTrText35 = document.createElement('tr');
744                          uncheckAllTrText35.textContent = '';
745
746                          uncheckAllTrText35.appendChild(uncheckAllTdText35);
747
748                          table.appendChild(uncheckAllTrText35);
749
750                          var uncheckAllText35 = document.createElement('div');
751                          uncheckAllText35.className = 'd-table-cell align-middle';
752                          uncheckAllText35.textContent = '';
753
754                          uncheckAllText35.appendChild(uncheckAllLink);
755                          uncheckAllText35.appendChild(uncheckAllSpan);
756
757                          var uncheckAllTdText36 = document.createElement('td');
758                          uncheckAllTdText36.textContent = '';
759
760                          var uncheckAllTrText36 = document.createElement('tr');
761                          uncheckAllTrText36.textContent = '';
762
763                          uncheckAllTrText36.appendChild(uncheckAllTdText36);
764
765                          table.appendChild(uncheckAllTrText36);
766
767                          var uncheckAllText36 = document.createElement('div');
768                          uncheckAllText36.className = 'd-table-cell align-middle';
769                          uncheckAllText36.textContent = '';
770
771                          uncheckAllText36.appendChild(uncheckAllLink);
772                          uncheckAllText36.appendChild(uncheckAllSpan);
773
774                          var uncheckAllTdText37 = document.createElement('td');
775                          uncheckAllTdText37.textContent = '';
776
777                          var uncheckAllTrText37 = document.createElement('tr');
778                          uncheckAllTrText37.textContent = '';
779
780                          uncheckAllTrText37.appendChild(uncheckAllTdText37);
781
782                          table.appendChild(uncheckAllTrText37);
783
784                          var uncheckAllText37 = document.createElement('div');
785                          uncheckAllText37.className = 'd-table-cell align-middle';
786                          uncheckAllText37.textContent = '';
787
788                          uncheckAllText37.appendChild(uncheckAllLink);
789                          uncheckAllText37.appendChild(uncheckAllSpan);
790
791                          var uncheckAllTdText38 = document.createElement('td');
792                          uncheckAllTdText38.textContent = '';
793
794                          var uncheckAllTrText38 = document.createElement('tr');
795                          uncheckAllTrText38.textContent = '';
796
797                          uncheckAllTrText38.appendChild(uncheckAllTdText38);
798
799                          table.appendChild(uncheckAllTrText38);
800
801                          var uncheckAllText38 = document.createElement('div');
802                          uncheckAllText38.className = 'd-table-cell align-middle';
803                          uncheckAllText38.textContent = '';
804
805                          uncheckAllText38.appendChild(uncheckAllLink);
806                          uncheckAllText38.appendChild(uncheckAllSpan);
807
808                          var uncheckAllTdText39 = document.createElement('td');
809                          uncheckAllTdText39.textContent = '';
810
811                          var uncheckAllTrText39 = document.createElement('tr');
812                          uncheckAllTrText39.textContent = '';
813
814                          uncheckAllTrText39.appendChild(uncheckAllTdText39);
815
816                          table.appendChild(uncheckAllTrText39);
817
818                          var uncheckAllText39 = document.createElement('div');
819                          uncheckAllText39.className = 'd-table-cell align-middle';
820                          uncheckAllText39.textContent = '';
821
822                          uncheckAllText39.appendChild(uncheckAllLink);
823                          uncheckAllText39.appendChild(uncheckAllSpan);
824
825                          var uncheckAllTdText40 = document.createElement('td');
826                          uncheckAllTdText40.textContent = '';
827
828                          var uncheckAllTrText40 = document.createElement('tr');
829                          uncheckAllTrText40.textContent = '';
830
831                          uncheckAllTrText40.appendChild(uncheckAllTdText40);
832
833                          table.appendChild(uncheckAllTrText40);
834
835                          var uncheckAllText40 = document.createElement('div');
836                          uncheckAllText40.className = 'd-table-cell align-middle';
837                          uncheckAllText40.textContent = '';
838
839                          uncheckAllText40.appendChild(uncheckAllLink);
840                          uncheckAllText40.appendChild(uncheckAllSpan);
841
842                          var uncheckAllTdText41 = document.createElement('td');
843                          uncheckAllTdText41.textContent = '';
844
845                          var uncheckAllTrText41 = document.createElement('tr');
846                          uncheckAllTrText41.textContent = '';
847
848                          uncheckAllTrText41.appendChild(uncheckAllTdText41);
849
850                          table.appendChild(uncheckAllTrText41);
851
852                          var uncheckAllText41 = document.createElement('div');
853                          uncheckAllText41.className = 'd-table-cell align-middle';
854                          uncheckAllText41.textContent = '';
855
856                          uncheckAllText41.appendChild(uncheckAllLink);
857                          uncheckAllText41.appendChild(uncheckAllSpan);
858
859                          var uncheckAllTdText42 = document.createElement('td');
860                          uncheckAllTdText42.textContent = '';
861
862                          var uncheckAllTrText42 = document.createElement('tr');
863                          uncheckAllTrText42.textContent = '';
864
865                          uncheckAllTrText42.appendChild(uncheckAllTdText42);
866
867                          table.appendChild(uncheckAllTrText42);
868
869                          var uncheckAllText42 = document.createElement('div');
870                          uncheckAllText42.className = 'd-table-cell align-middle';
871                          uncheckAllText42.textContent = '';
872
873                          uncheckAllText42.appendChild(uncheckAllLink);
874                          uncheckAllText42.appendChild(uncheckAllSpan);
875
876                          var uncheckAllTdText43 = document.createElement('td');
877                          uncheckAllTdText43.textContent = '';
878
879                          var uncheckAllTrText43 = document.createElement('tr');
880                          uncheckAllTrText43.textContent = '';
881
882                          uncheckAllTrText43.appendChild(uncheckAllTdText43);
883
884                          table.appendChild(uncheckAllTrText43);
885
886                          var uncheckAllText43 = document.createElement('div');
887                          uncheckAllText43.className = 'd-table-cell align-middle';
888                          uncheckAllText43.textContent = '';
889
890                          uncheckAllText43.appendChild(uncheckAllLink);
891                          uncheckAllText43.appendChild(uncheckAllSpan);
892
893                          var uncheckAllTdText44 = document.createElement('td');
894                          uncheckAllTdText44.textContent = '';
895
896                          var uncheckAllTrText44 = document.createElement('tr');
897                          uncheckAllTrText44.textContent = '';
898
899                          uncheckAllTrText44.appendChild(uncheckAllTdText44);
900
901                          table.appendChild(uncheckAllTrText44);
902
903                          var uncheckAllText44 = document.createElement('div');
904                          uncheckAllText44.className = 'd-table-cell align-middle';
905                          uncheckAllText44.textContent = '';
906
907                          uncheckAllText44.appendChild(uncheckAllLink);
908                          uncheckAllText44.appendChild(uncheckAllSpan);
909
910                          var uncheckAllTdText45 = document.createElement('td');
911                          uncheckAllTdText45.textContent = '';
912
913                          var uncheckAllTrText45 = document.createElement('tr');
914                          uncheckAllTrText45.textContent = '';
915
916                          uncheckAllTrText45.appendChild(uncheckAllTdText45);
917
918                          table.appendChild(uncheckAllTrText45);
919
920                          var uncheckAllText45 = document.createElement('div');
921                          uncheckAllText45.className = 'd-table-cell align-middle';
922                          uncheckAllText45.textContent = '';
923
924                          uncheckAllText45.appendChild(uncheckAllLink);
925                          uncheckAllText45.appendChild(uncheckAllSpan);
926
927                          var uncheckAllTdText46 = document.createElement('td');

```

```

index.blade.php X
resources > views > admin > product > index.blade.php > ...
69      <span class="pull-right pr-2">
70        <button class="btn btn-sm btn-danger pr-3 pl-3" @click="bulkDelete('/admin/products/bulk-destroy')">{{ trans('brackets/admin-ui::admin.btn.bulk-delete') }}</button>
71      </span>
72    </td>
73  </tr>
74</thead>
75<tbody>
76
77  <tr v-for="(item, index) in collection" :key="item.id" :class="bulkItems[item.id] ? 'bg-bulk' : ''">
78    <td class="bulk-checkbox">
79      <input class="form-check-input" :id="`enabled` + item.id" type="checkbox" v-model="bulkItems[item.id]" v-validate="" :data-vv-name="`enabled` + item.id">
80      <label class="form-check-label" :for="`enabled` + item.id"></label>
81    </td>
82
83    <td>{{ item.id }}</td>
84    <td>{{ item.title }}</td>
85    <td>{{ item.slug }}</td>
86    <td>{{ item.category.category }}</td>
87    <td>{{ item.stock }}</td>
88    <td>{{ item.price }}</td>
89    <td>{{ item.without_discount_price }}</td> -->
90
91    <td>
92      <label class="switch switch-3d switch-success">
93        <input type="checkbox" class="switch-input" v-model="collection[index].new" @change="toggleSwitch(item.resource_url, 'new', collection[index].new)">
94        <span class="switch-slider"></span>
95      </label>
96    </td>
97
98    <td>
99      <div class="row no-gutters">
100        <div class="col-auto">
101          <a class="btn btn-sm btn-spinner btn-info" :href="item.resource_url + '/edit'" title="{{ trans('brackets/admin-ui::admin.btn.edit') }}"></a>
102        </div>
103      </div>
104    </td>
105  </tr>
106</tbody>
107</table>
108
109<div class="row" v-if="pagination.state.total > 0">
110  <div class="col-sm">
111    <span class="pagination-caption">{{ trans('brackets/admin-ui::admin.pagination.overview') }}</span>
112  </div>
113  <div class="col-sm-auto">
114    <pagination></pagination>
115  </div>
116</div>
117
118<div class="no-items-found" v-if="!collection.length > 0">
119  <i class="icon-magnifier"></i>
120  <h3>{{ trans('brackets/admin-ui::admin.index.no_items') }}</h3>
121  <p>{{ trans('brackets/admin-ui::admin.index.try_changing_items') }}</p>
122  <a class="btn btn-primary btn-spinner" href="{{ url('admin/products/create') }}" role="button"><i class="fa fa-plus"></i>&ampnbsp {{ trans('brackets/admin-ui::admin.button.create') }}</a>
123</div>
124</div>
125</div>
126</div>
127</div>
128</div>
129</div>
130</div>
131</div>
132</div>
133</product-listing>
134 @endsection

```

Figure 65: Product.php

← → ⌂ ① 127.0.0.1:8000/admin/products/create

Vegetables Delivery Dashboard

Administrator Administrator

**CONTENT**

- Categories
- Products
- Orders
- Testimonials
- Messages

**SETTINGS**

- Manage access

**+ New Product**

Title

Select Category

(max no. of files: 1 files)

Choose Images

Stock

← → ⌂ ① 127.0.0.1:8000/admin/products/create

Vegetables Delivery Dashboard

Administrator Administrator

**CONTENT**

- Categories
- Products
- Orders
- Testimonials
- Messages

**SETTINGS**

- Manage access

Description

Price

New

Figure 66: New Product Form

```

create.blade.php X
resources > views > admin > product > create.blade.php >  div.container-xl
1  @extends('brackets/admin-ui::admin.layout.default')
2
3  @section('title', trans('admin.product.actions.create'))
4
5  @section('body')
6
7      <div class="container-xl">
8          |      <div class="card">
9
10         <product-form
11             :action="'{ url('admin/products') }'"
12             v-cloak
13             inline-template>
14
15             <form class="form-horizontal form-create" method="post" @submit.prevent="onSubmit" :action="action" >
16
17                 <div class="card-header">
18                     |                     <i class="fa fa-plus"></i> {{ trans('admin.product.actions.create') }}
19                 </div>
20
21                 <div class="card-body">
22                     |                     @include('admin.product.components.form-elements')
23                 </div>
24
25                 <div class="card-footer">
26                     <button type="submit" class="btn btn-primary" :disabled="submiting">
27                         |                         <i class="fa" :class="submiting ? 'fa-spinner' : 'fa-download'"></i>
28                         {{ trans('brackets/admin-ui::admin.btn.save') }}
29                     </button>
30                 </div>
31
32             </form>
33
34     </div>

```

```

create.blade.php X
resources > views > admin > product > create.blade.php >  div.container-xl
35     </product-form>
36
37     </div>
38
39     </div>
40 @endsection

```

Figure 67: New Product.php

The screenshot shows a web-based administration interface for a 'Vegetables Delivery Dashboard'. The top navigation bar includes a back arrow, forward arrow, refresh button, and a URL field showing '127.0.0.1:8000/admin/products/1/edit'. The top right corner shows a user profile for 'Administrator'.

The main content area is titled 'Edit Sweet Potatoes'. It contains the following fields:

- Title:** Sweet Potatoes
- Select Category:** Vegetables
- Choose Images:** An input field showing a thumbnail of several sweet potatoes.
- Stock:** 750

A sidebar on the left, titled 'CONTENT', lists navigation items: Categories, Products, Orders, Testimonials, and Messages. Below that, under 'SETTINGS', is 'Manage access'.

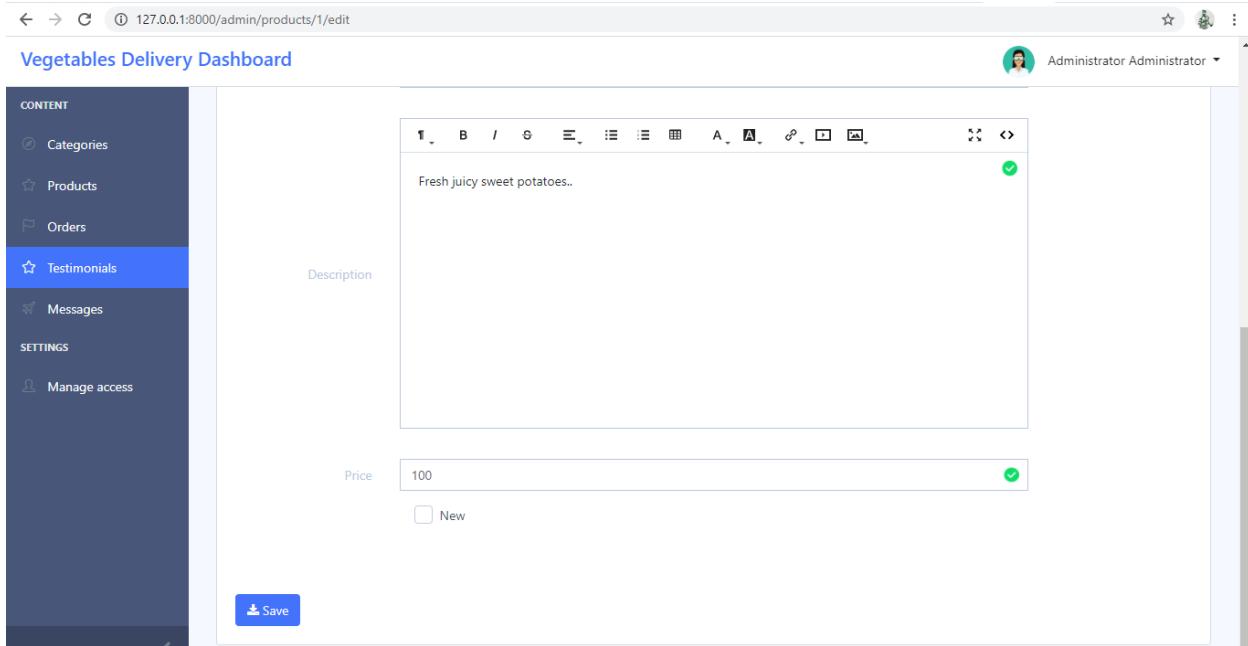


Figure 68: Edit Product Form

```

edit.blade.php X
resources > views > admin > product > edit.blade.php > ...
1  @extends('brackets/admin-ui::admin.layout.default')
2
3  @section('title', trans('admin.product.actions.edit', ['name' => $product->title]))
4
5  @section('body')
6
7      <div class="container-xl">
8          <div class="card">
9
10             <product-form
11                 :action="{{ $product->resource_url }}"
12                 :data="{{ $product->toJson() }}"
13                 v-cloak
14                 inline-template>
15
16                 <form class="form-horizontal form-edit" method="post" @submit.prevent="onSubmit" :action="action" novalidate>
17
18                     <div class="card-header">
19                         | <i class="fa fa-pencil"></i> {{ trans('admin.product.actions.edit', ['name' => $product->title]) }}
20                     </div>
21
22                     <div class="card-body">
23                         | @include('admin.product.components.edit-form-elements')
24                     </div>
25
26
27                     <div class="card-footer">
28                         <button type="submit" class="btn btn-primary" :disabled="submiting">
29                             | <i class="fa" :class="submiting ? 'fa-spinner' : 'fa-download'"></i>
30                             | {{ trans('brackets/admin-ui::admin.btn.save') }}
31                         </button>
32                     </div>
33
34

```

*Figure 69: Edit Product.php*

The screenshot shows a web-based administrative dashboard for a vegetable delivery service. The URL in the address bar is 127.0.0.1:8000/admin/orders. The top navigation bar includes a back arrow, forward arrow, a refresh icon, and a user profile for 'Administrator'. The main header is 'Vegetables Delivery Dashboard'.

The left sidebar, titled 'CONTENT', contains the following menu items:

- Categories
- Products
- Orders** (selected)
- Testimonials
- Messages

Below the sidebar, under 'SETTINGS', is the 'Manage access' option.

The central content area is titled 'Orders' and features a search bar with a placeholder 'Search' and a blue 'Search' button. To the right of the search bar is a dropdown menu set to '10' items per page. A table lists one order entry:

#	ID	Order	Name	Address	Contact	Order approve
<input type="checkbox"/>	2	86NNDY60GAHW9M0	Dawa Gyalzen Sherpa	Boudha	9876543223	<input checked="" type="checkbox"/>

Below the table, a message states 'Displaying items from 1 to 1 of total 1 items.' On the far right of the table row are three small icons: a blue square with a white checkmark, a blue square with a white eye, and a red square with a white minus sign.

*Figure 70: Orders Page*

```
index.blade.php ...\\product index.blade.php ...\\order X
resources > views > admin > order > index.blade.php > ...
1  @extends('brackets/admin-ui::admin.layout.default')
2
3  @section('title', trans('admin.order.actions.index'))
4
5  @section('body')
6
7      <order-listing
8          :data="{!! $data->toJson() !!}"
9          :url="{!! url('admin/orders') !!}"'
10         inline-template>
11
12         <div class="row">
13             <div class="col">
14                 <div class="card">
15                     <div class="card-header">
16                         <i class="fa fa-align-justify"></i> {{ trans('admin.order.actions.index') }}
17                         <!-- <a class="btn btn-primary btn-spinner btn-sm pull-right m-b-0" href="{!! url('admin/orders/create') !!}" role="button"><i class="fa fa-plus"></i> Create </a> -->
18                     </div>
19                     <div class="card-body" v-cloak>
20                         <div class="card-block">
21                             <form @submit.prevent="">
22                                 <div class="row justify-content-md-between">
23                                     <div class="col col-lg-7 col-xl-5 form-group">
24                                         <div class="input-group">
25                                             <input class="form-control" placeholder="{!! trans('brackets/admin-ui::admin.placeholder.search') !!}" v-model="search" @input="filterSearch($event)" type="text" />
26                                             <span class="input-group-append">
27                                                 <button type="button" class="btn btn-primary" @click="filter('search', search)"><i class="fa fa-search"></i>&ampnbsp</button>
28                                             </span>
29                                         </div>
30                                         <div class="col-sm-auto form-group">
31                                             <select class="form-control" v-model="pagination.state.per_page">
32                                                 <option value="10">10</option>
33                                                 <option value="25">25</option>
34                                                 <option value="100">100</option>
35                                             </select>
36                                         </div>
37                                     </div>
38                                 </div>
39                             </form>
40                         </div>
41                     </div>
42                     <table class="table table-hover table-listing">
43                         <thead>
44                             <tr>
45                                 <th class="bulk-checkbox">
46                                     <input class="form-check-input" id="enabled" type="checkbox" v-model="isClickedAll" v-validate="" data-vv-name="enabled" name="enabled_fake_el" checked="" />
47                                     <label class="form-check-label" for="enabled">
48                                         #
49                                     </label>
50                                 </th>
51
52                                 <th is='sortable' :column='id'>{{ trans('admin.order.columns.id') }}</th>
53                                 <th is='sortable' :column='order_id'>{{ trans('admin.order.columns.order_id') }}</th>
54                                 <!-- <th is="sortable" :column="user_id">{{ trans('admin.order.columns.user_id') }}</th> -->
55                                 <th is='sortable' :column='name'>{{ trans('admin.order.columns.name') }}</th>
56                                 <th is='sortable' :column='address'>{{ trans('admin.order.columns.address') }}</th>
57                                 <th is='sortable' :column='contact'>{{ trans('admin.order.columns.contact') }}</th>
58                                 <th is='sortable' :column='order_approve'>{{ trans('admin.order.columns.order_approve') }}</th>
59
60                             <th></th>
61                         </tr>
62                     </thead>
63                     <tbody>
64                         <tr v-show="(clickedBulkItemsCount > 0) || isClickedAll">
65                             <td class="bg-bulk-info d-table-cell text-center" colspan="9">
66                                 <span class="align-middle font-weight-light text-dark">{{ trans('brackets/admin-ui::admin.listing.selected_items') }} @{{ clickedBulkItemsCount }}</span>
67                                 <a href="#" class="text-primary" @click="onBulkItemsClickedAllUncheck()">{{ trans('brackets/admin-ui::admin.listing.uncheck_all_items') }}</a>
68
69                                 <span class="pull-right pr-2">
70                                     <button class="btn btn-sm btn-danger pr-3 pl-3" @click="bulkDelete('/admin/orders/bulk-destroy')">{{ trans('brackets/admin-ui::admin.btnDelete') }}</button>
71                                 </span>
72                             </td>
73                         </tr>
74                     </tbody>
75                 </table>
76             </div>
77         </div>
78     </div>
79 
```

```
index.blade.php ...\\product index.blade.php ...\\order X
resources > views > admin > order > index.blade.php > ...
35             <option value="25">25</option>
36             <option value="100">100</option>
37         </select>
38     </div>
39 </div>
40 </form>
41
42 <table class="table table-hover table-listing">
43     <thead>
44         <tr>
45             <th class="bulk-checkbox">
46                 <input class="form-check-input" id="enabled" type="checkbox" v-model="isClickedAll" v-validate="" data-vv-name="enabled" name="enabled_fake_el" checked="" />
47                 <label class="form-check-label" for="enabled">
48                     #
49                 </label>
50             </th>
51
52             <th is='sortable' :column='id'>{{ trans('admin.order.columns.id') }}</th>
53             <th is='sortable' :column='order_id'>{{ trans('admin.order.columns.order_id') }}</th>
54             <!-- <th is="sortable" :column="user_id">{{ trans('admin.order.columns.user_id') }}</th> -->
55             <th is='sortable' :column='name'>{{ trans('admin.order.columns.name') }}</th>
56             <th is='sortable' :column='address'>{{ trans('admin.order.columns.address') }}</th>
57             <th is='sortable' :column='contact'>{{ trans('admin.order.columns.contact') }}</th>
58             <th is='sortable' :column='order_approve'>{{ trans('admin.order.columns.order_approve') }}</th>
59
60         <th></th>
61     </tr>
62     <tr v-show="(clickedBulkItemsCount > 0) || isClickedAll">
63         <td class="bg-bulk-info d-table-cell text-center" colspan="9">
64             <span class="align-middle font-weight-light text-dark">{{ trans('brackets/admin-ui::admin.listing.selected_items') }} @{{ clickedBulkItemsCount }}</span>
65             <a href="#" class="text-primary" @click="onBulkItemsClickedAllUncheck()">{{ trans('brackets/admin-ui::admin.listing.uncheck_all_items') }}</a>
66
67             <span class="pull-right pr-2">
68                 <button class="btn btn-sm btn-danger pr-3 pl-3" @click="bulkDelete('/admin/orders/bulk-destroy')">{{ trans('brackets/admin-ui::admin.btnDelete') }}</button>
69             </span>
70         </td>
71     </tr>
72 </thead>
73 <tbody>
74     <tr v-show="(clickedBulkItemsCount > 0) || isClickedAll">
75         <td class="bg-bulk-info d-table-cell text-center" colspan="9">
76             <span class="align-middle font-weight-light text-dark">{{ trans('brackets/admin-ui::admin.listing.selected_items') }} @{{ clickedBulkItemsCount }}</span>
77             <a href="#" class="text-primary" @click="onBulkItemsClickedAllUncheck()">{{ trans('brackets/admin-ui::admin.listing.uncheck_all_items') }}</a>
78
79             <span class="pull-right pr-2">
80                 <button class="btn btn-sm btn-danger pr-3 pl-3" @click="bulkDelete('/admin/orders/bulk-destroy')">{{ trans('brackets/admin-ui::admin.btnDelete') }}</button>
81             </span>
82         </td>
83     </tr>
84 </tbody>
85 </table>
86 
```

```

index.blade.php ...\\product index.blade.php ...\\order X
resources > views > admin > order > index.blade.php > ...
69 |         </span>
70 |     </td>
71 |   </tr>
72 | </tbody>
73 | <tbody v-for="(item, index) in collection" :key="item.id" :class="bulkItems[item.id] ? 'bg-bulk' : ''">
74 |   <tr class="bulk-checkbox">
75 |     <td>@{{ item.id }}</td>
76 |     <td>@{{ item.order_id }}</td>
77 |     <!-- <td>@{{ item.user_id }}</td> -->
78 |     <td>@{{ item.name }}</td>
79 |     <td>@{{ item.address }}</td>
80 |     <td>@{{ item.contact }}</td>
81 |     <td>
82 |       <label class="switch switch-3d switch-success">
83 |         <input type="checkbox" class="switch-input" v-model="collection[index].order_approve" @change="toggleSwitch(item.resource_url, 'order_approve')"/>
84 |         <span class="switch-slider"></span>
85 |       </label>
86 |     </td>
87 |     <!-- <td>@{{ item.order_approve }}</td> -->
88 |
89 |   <td>
90 |     <div class="row no-gutters">
91 |       <div class="col-auto">
92 |         <a class="btn btn-sm btn-spinner btn-info" :href="item.resource_url + '/edit'" title="{{ trans('brackets/admin-ui::admin.btn.edit') }}" role="button">
93 |           Edit
94 |         </a>
95 |       </div>
96 |       <div class="col">
97 |         <a class="btn btn-sm btn-spinner btn-info" :href="item.order_item" title="{{ trans('brackets/admin-ui::admin.btn.view') }}" role="button">
98 |           View
99 |         </a>
100 |       </div>
101 |     </div>
102 |   </td>
103 | </tr>
104 | </tbody>
105 | </table>
106 |
107 | <!-- <div class="row" v-if="pagination.state.total > 0">
108 |   <div class="col-sm">
109 |     <form class="col" @submit.prevent="deleteItem(item.resource_url)">
110 |       <button type="submit" class="btn btn-sm btn-danger" title="{{ trans('brackets/admin-ui::admin.btn.delete') }}">&ampnbspDelete
111 |     </button>
112 |   </form>
113 | </div>
114 | </div>
115 | <!-- <div class="no-items-found" v-if="!collection.length > 0">
116 |   <i class="icon-magnifier"></i>
117 |   <h3>{{ trans('brackets/admin-ui::admin.index.no_items') }}</h3>
118 |   <p>{{ trans('brackets/admin-ui::admin.index.try_changing_items') }}</p>
119 |   <a class="btn btn-primary btn-spinner" href="{{ url('admin/orders/create') }}" role="button"><i class="fa fa-plus"></i>&ampnbsp {{ trans('brackets/admin-ui::admin.add_new') }} </a>
120 | </div>
121 | </div>
122 | </div>
123 | </div>
124 | </div>
125 | </div>
126 | </div>
127 | </div>
128 | </div>
129 | </div>
130 | </div>
131 | </div>
132 | </div>
133 | </div>
134 | </div>
135 | @endsection

```

Figure 71: Orders.php

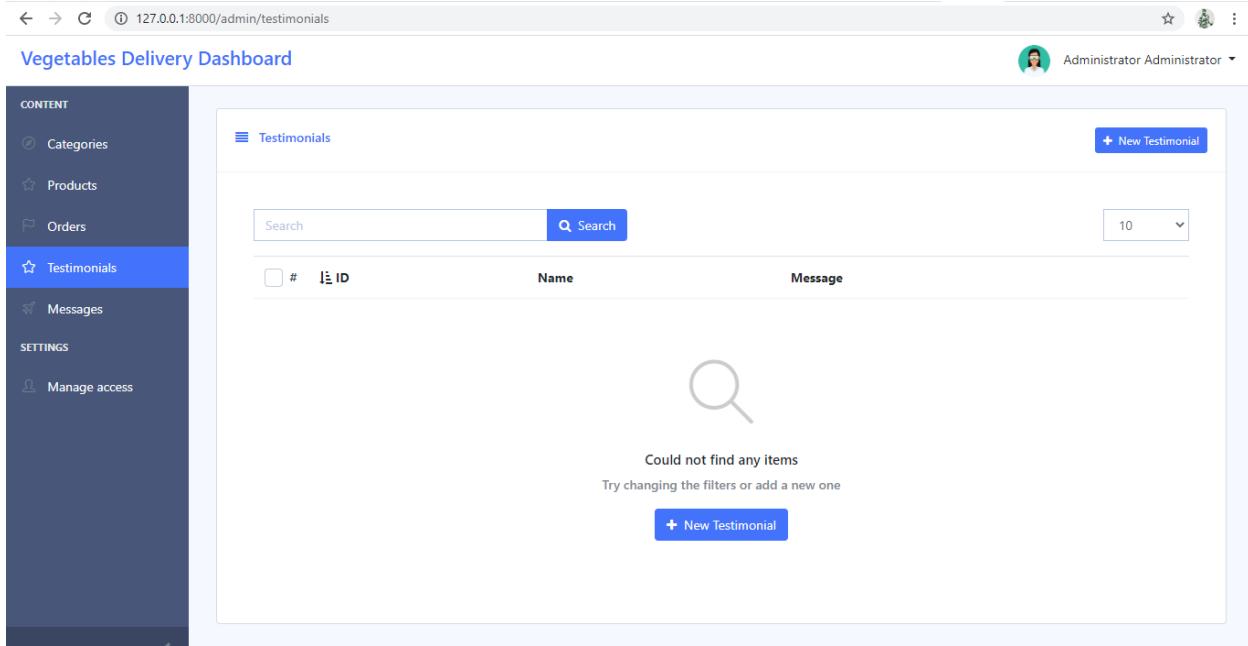


Figure 72: Testimonials Page

```
index.blade.php X
resources > views > admin > testimonial > index.blade.php > ...
1  @extends('brackets/admin-ui::admin.layout.default')
2
3  @section('title', trans('admin.testimonial.actions.index'))
4
5  @section('body')
6
7      <testimonial-listing
8          :data="{!! $data->toJson() !!}"
9          :url="{!! url('admin/testimonials') !!}"'
10         inline-template>
11
12         <div class="row">
13             <div class="col">
14                 <div class="card">
15                     <div class="card-header">
16                         <i class="fa fa-align-justify"></i> {{ trans('admin.testimonial.actions.index') }}
17                         <a class="btn btn-primary btn-spinner btn-sm pull-right m-b-0" href="{!! url('admin/testimonials/create') !!}" role="button"><i class="fa fa-plus"></i> New Testimonial</a>
18                     </div>
19                     <div class="card-body" v-cloak>
20                         <div class="card-block">
21                             <form @submit.prevent="">
22                                 <div class="row justify-content-md-between">
23                                     <div class="col col-lg-7 col-xl-5 form-group">
24                                         <div class="input-group">
25                                             <input class="form-control" placeholder="{!! trans('brackets/admin-ui::admin.placeholder.search') !!}" v-model="search" @change="filter('search', search)" type="text">
26                                             <span class="input-group-append">
27                                                 <button type="button" class="btn btn-primary" @click="filter('search', search)"><i class="fa fa-search"></i>&ampnbsp</button>
28                                             </span>
29                                         </div>
30                                         <div class="col-sm-auto form-group">
31                                             <select class="form-control" v-model="pagination.state.per_page">
32                                                 <option value="10">10</option>
33                                             </select>
34                                         </div>
35                                     </div>
36                                 </div>
37                             </form>
38                         </div>
39                     </div>
40                 </div>
41             </div>
42         </div>
43     </testimonial-listing>
```

```
  index.blade.php X
resources > views > admin > testimonial >   index.blade.php > ...
35     <option value="25">25</option>
36     <option value="100">100</option>
37   </select>
38 </div>
39 </form>
40
41 <table class="table table-hover table-listing">
42   <thead>
43     <tr>
44       <th class="bulk-checkbox">
45         <input class="form-check-input" id="enabled" type="checkbox" v-model="isClickedAll" v-validate="" :data-vv-name="enabled" name="enabled_fake_ele
46         <label class="form-check-label" for="enabled">
47           #
48         </label>
49       </th>
50     </tr>
51     <th is='sortable' :column="'id'">{{ trans('admin.testimonial.columns.id') }}</th>
52     <th is='sortable' :column="'name'">{{ trans('admin.testimonial.columns.name') }}</th>
53     <th is='sortable' :column="'message'">{{ trans('admin.testimonial.columns.message') }}</th>
54
55     <th></th>
56   </thead>
57   <tr v-show="(clickedBulkItemsCount > 0) || isClickedAll">
58     <td class="bg-bulk-info d-table-cell text-center" colspan="5">
59       <span class="align-middle font-weight-light text-dark">{{ trans('brackets/admin-ui::admin.listing.selected_items') }} @{{ clickedBulkItemsCount }}<
60       <a href="#" class="text-primary" @click="onBulkItemsClickedAllUncheck()">{{ trans('brackets/admin-ui::admin.listing.uncheck_all_items') }}<
61       <span class="pull-right pr-2">
62         <button class="btn btn-sm btn-danger pr-3 pl-3" @click="bulkDelete('/admin/testimonials/bulk-destroy')">{{ trans('brackets/admin-ui::admin.btr
63       </span>
64     </td>
65   </tr>
66 </table>
```

```
  index.blade.php X
resources > views > admin > testimonial >   index.blade.php > ...
69   </thead>
70   <tbody>
71     <tr v-for="(item, index) in collection" :key="item.id" :class="bulkItems[item.id] ? 'bg-bulk' : ''">
72       <td class="bulk-checkbox">
73         <input class="form-check-input" :id="`enabled` + item.id" type="checkbox" v-model="bulkItems[item.id]" v-validate="" :data-vv-name="`enabled` +
74         <label class="form-check-label" :for="`enabled` + item.id">
75       </td>
76
77       <td>@{{ item.id }}</td>
78       <td>@{{ item.name }}</td>
79       <td>@{{ item.message }}</td>
80
81       <td>
82         <div class="row no-gutters">
83           <div class="col-auto">
84             <a class="btn btn-sm btn-spinner btn-info" :href="item.resource_url + '/edit'" title="{{ trans('brackets/admin-ui::admin.btn.edit') }}" r
85           </div>
86           <form class="col" @submit.prevent="deleteItem(item.resource_url)">
87             <button type="submit" class="btn btn-sm btn-danger" title="{{ trans('brackets/admin-ui::admin.btn.delete') }}><i class="fa fa-trash-o"><
88           </form>
89         </div>
90       </td>
91     </tr>
92   </tbody>
93 </table>
94
95 <div class="row" v-if="pagination.state.total > 0">
96   <div class="col-sm">
97     <span class="pagination-caption">{{ trans('brackets/admin-ui::admin.pagination.overview') }}</span>
98   </div>
99   <div class="col-sm-auto">
100     <pagination></pagination>
101   </div>
```

```

index.blade.php X
resources > views > admin > testimonial > index.blade.php ...
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118 @endsection

```

Figure 73: Testimonials.php

← → ⌂ ⓘ 127.0.0.1:8000/admin/testimonials/create

Vegetables Delivery Dashboard

Administrator Administrator ▾

**CONTENT**

- Categories
- Products
- Orders
- Testimonials
- Messages

**SETTINGS**

- Manage access

+ New Testimonial

Name

Message

Image (max no. of files: 1 files)

Choose Image

Drop files here to upload

**Save**

Figure 74: Add Testimonial Form

The image shows two code snippets in a code editor, likely Laravel Blade files, for creating a testimonial.

**Top Snippet:**

```
resources > views > admin > testimonial > create.blade.php > ...
1  @extends('brackets/admin-ui::admin.layout.default')
2
3  @section('title', trans('admin.testimonial.actions.create'))
4
5  @section('body')
6
7      <div class="container-xl">
8          |      <div class="card">
9
10         <testimonial-form
11             :action="{{ url('admin/testimonials') }}"
12             v-cloak
13             inline-template>
14
15             <form class="form-horizontal form-create" method="post" @submit.prevent="onSubmit" :action="action" novalidate>
16
17                 <div class="card-header">
18                     |                     <i class="fa fa-plus"></i> {{ trans('admin.testimonial.actions.create') }}
19                 </div>
20
21                 <div class="card-body">
22                     |                     @include('admin.testimonial.components.form-elements')
23                 </div>
24
25                 <div class="card-footer">
26                     <button type="submit" class="btn btn-primary" :disabled="submiting">
27                         |                         <i class="fa" :class="submiting ? 'fa-spinner' : 'fa-download'"></i>
28                         {{ trans('brackets/admin-ui::admin.btn.save') }}
29                     </button>
30                 </div>
31
32             </form>
33
34     </div>
```

**Bottom Snippet:**

```
resources > views > admin > testimonial > create.blade.php > ...
34
35     </testimonial-form>
36
37     </div>
38
39     </div>
40
41
42 @endsection
```

Figure 75: Add Testimonial.php

Figure 76: Messages Page

```

index.blade.php X
resources > views > admin > message > index.blade.php > ...
1  @extends('brackets/admin-ui::admin.layout.default')
2
3  @section('title', trans('admin.message.actions.index'))
4
5  @section('body')
6
7      <message-listing
8          :data="{!! $data->toJson() !!}"
9          :url="{!! url('admin/messages') !!}"
10         inline-template>
11
12      <div class="row">
13          <div class="col">
14              <div class="card">
15                  <div class="card-header">
16                      <i class="fa fa-align-justify"></i> {{ trans('admin.message.actions.index') }}
17                      <!-- <a class="btn btn-primary btn-spinner btn-sm pull-right m-b-0" href="{!! url('admin/messages/create') !!}" role="button"><i class="fa fa-plus"></i> Create New </a> -->
18                  </div>
19                  <div class="card-body" v-cloak>
20                      <div class="card-block">
21                          <form @submit.prevent="">
22                              <div class="row justify-content-md-between">
23                                  <div class="col col-lg-7 col-xl-5 form-group">
24                                      <div class="input-group">
25                                          <input class="form-control" placeholder="{!! trans('brackets/admin-ui::admin.placeholder.search') !!}" v-model="search" @change="filter(search)" type="text">
26                                          <span class="input-group-append">
27                                              <button type="button" class="btn btn-primary" @click="filter('search', search)"><i class="fa fa-search"></i>&ampnbsp</button>
28                                          </span>
29                                      </div>
30                                  </div>
31                                  <div class="col-sm-auto form-group">
32                                      <select class="form-control" v-model="pagination.state.per_page">
33                                          <option value="10">10</option>
34

```

```
index.blade.php X
resources > views > admin > message > index.blade.php > ...
34 |         <option value="10">10</option>
35 |         <option value="25">25</option>
36 |         <option value="100">100</option>
37 |     </select>
38 |   </div>
39 | </div>
40 | </form>
41 |
42 <table class="table table-hover table-listing">
43   <thead>
44     <tr>
45       <th class="bulk-checkbox">
46         <input class="form-check-input" id="enabled" type="checkbox" v-model="isClickedAll" v-validate="" data-vv-name="enabled" name="enabled_fake_e
47         <label class="form-check-label" for="enabled">
48           # ...
49         </label>
50       </th>
51
52       <th is="sortable" :column="`id`">{{ trans('admin.message.columns.id') }}</th>
53       <th is="sortable" :column="`name`">{{ trans('admin.message.columns.name') }}</th>
54       <th is="sortable" :column="`email`">{{ trans('admin.message.columns.email') }}</th>
55       <th is="sortable" :column="`subject`">{{ trans('admin.message.columns.subject') }}</th>
56       <th is="sortable" :column="`message`">{{ trans('admin.message.columns.message') }}</th>
57
58     <th></th>
59   </tr>
60   <tr v-show="(clickedBulkItemCount > 0) || isClickedAll">
61     <td class="bg-bulk-info d-table-cell text-center" colspan="7">
62       <span class="align-middle font-weight-light text-dark">{{ trans('brackets/admin-ui::admin.listing.selected_items') }} @{{ clickedBulkItemCount
63       | href="#" class="text-primary" @click="onBulkItemsClickedAllUncheck()">{{ trans('brackets/admin-ui::admin.listing.uncheck_all_items')
64
65       <span class="pull-right pr-2">
66         <button class="btn btn-sm btn-danger pr-3 pl-3" @click="bulkDelete('/admin/messages/bulk-destroy')">{{ trans('brackets/admin-ui::admin.btn.d
67       </span>
```

```
index.blade.php X
resources > views > admin > message > index.blade.php > ...
67 |         </span>
68 |       </td>
69 |     </tr>
70   </thead>
71   <tbody>
72     <tr v-for="(item, index) in collection" :key="item.id" :class="bulkItems[item.id] ? 'bg-bulk' : ''">
73       <td class="bulk-checkbox">
74         <input class="form-check-input" :id="`enabled` + item.id" type="checkbox" v-model="bulkItems[item.id]" v-validate="" :data-vv-name="`enabled` + item.id"
75         <label class="form-check-label" for="`enabled` + item.id">
76       </td>
77
78       <td>@{{ item.id }}</td>
79       <td>@{{ item.name }}</td>
80       <td>@{{ item.email }}</td>
81       <td>@{{ item.subject }}</td>
82       <td>@{{ item.message }}</td>
83
84       <td>
85         <div class="row no-gutters">
86           <!-- <div class="col-auto">
87             <a class="btn btn-sm btn-spinner btn-info" :href="item.resource_url + '/edit'" title="{{ trans('brackets/admin-ui::admin.btn.edit') }}"
88           </div> -->
89           <form class="col" @submit.prevent="deleteItem(item.resource_url)">
90             <button type="submit" class="btn btn-sm btn-danger" title="{{ trans('brackets/admin-ui::admin.btn.delete') }}"><i class="fa fa-trash-o">
91           </button>
92         </form>
93       </div>
94
95     </td>
96   </tr>
97 </tbody>
98 </table>
99
100 <div class="row" v-if="pagination.state.total > 0">
```

```

index.blade.php X
resources > views > admin > message > index.blade.php > ...
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122 @endsection

```

Figure 77: Messages.php

← → ⌂ ① 127.0.0.1:8000/admin/profile

Administrator Administrator ▾

**Vegetables Delivery Dashboard**

**CONTENT**

- Categories
- Products
- Orders
- Testimonials
- Messages

**SETTINGS**

- Manage access

**Edit Profile**

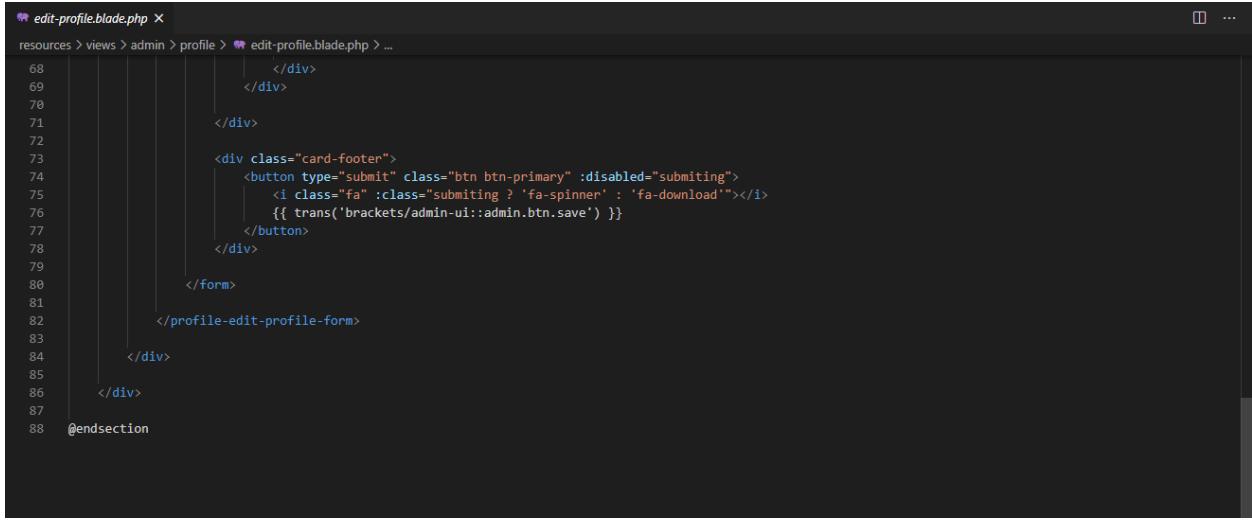
	First name	Administrator	✓
	Last name	Administrator	✓
	Email	admin@softwarica.edu.np	✓
	Language	en	▼

**Save**

Figure 78: Admin Profile Edit Form

```
edit-profile.blade.php x
resources > views > admin > profile > edit-profile.blade.php > ...
1  @extends('brackets/admin-ui::admin.layout.default')
2
3  @section('title', trans('admin.admin-user.actions.edit_profile'))
4
5  @section('body')
6
7      <div class="container-xl">
8
9          <div class="card">
10
11              <profile-edit-profile-form
12                  :action="{{ url('admin/profile') }}"
13                  :data="{{ $adminUser->toJson() }}"
14
15                  inline-template>
16
17                  <form class="form-horizontal form-edit" method="post" @submit.prevent="onSubmit" :action="action">
18
19                      <div class="card-header">
20                          <i class="fa fa-pencil"></i> {{ trans('admin.admin-user.actions.edit_profile') }}
21                      </div>
22
23                      <div class="card-body">
24
25                          <div class="row">
26                              <div class="col-md-4 text-center">
27                                  <div class="avatar-upload">
28                                      @include('brackets/admin-ui::admin.includes.avatar-uploader', [
29                                          'mediaCollection' => app(\Brackets\AdminAuth\Models\AdminUser::class)->getMediaCollection('avatar'),
30                                          'media' => $adminUser->getThumbs200ForCollection('avatar')
31                                      ])
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
229
230
231
232
233
234
235
236
237
238
239
239
240
241
242
243
244
245
246
247
248
249
249
250
251
252
253
254
255
256
257
257
258
259
259
260
261
262
263
264
265
266
267
267
268
268
269
269
270
271
272
273
273
274
274
275
275
276
276
277
277
278
278
279
279
280
280
281
281
282
282
283
283
284
284
285
285
286
286
287
287
288
288
289
289
290
290
291
291
292
292
293
293
294
294
295
295
296
296
297
297
298
298
299
299
300
300
301
301
302
302
303
303
304
304
305
305
306
306
307
307
308
308
309
309
310
310
311
311
312
312
313
313
314
314
315
315
316
316
317
317
318
318
319
319
320
320
321
321
322
322
323
323
324
324
325
325
326
326
327
327
328
328
329
329
330
330
331
331
332
332
333
333
334
334
335
335
336
336
337
337
338
338
339
339
340
340
341
341
342
342
343
343
344
344
345
345
346
346
347
347
348
348
349
349
350
350
351
351
352
352
353
353
354
354
355
355
356
356
357
357
358
358
359
359
360
360
361
361
362
362
363
363
364
364
365
365
366
366
367
367
368
368
369
369
370
370
371
371
372
372
373
373
374
374
375
375
376
376
377
377
378
378
379
379
380
380
381
381
382
382
383
383
384
384
385
385
386
386
387
387
388
388
389
389
390
390
391
391
392
392
393
393
394
394
395
395
396
396
397
397
398
398
399
399
400
400
401
401
402
402
403
403
404
404
405
405
406
406
407
407
408
408
409
409
410
410
411
411
412
412
413
413
414
414
415
415
416
416
417
417
418
418
419
419
420
420
421
421
422
422
423
423
424
424
425
425
426
426
427
427
428
428
429
429
430
430
431
431
432
432
433
433
434
434
435
435
436
436
437
437
438
438
439
439
440
440
441
441
442
442
443
443
444
444
445
445
446
446
447
447
448
448
449
449
450
450
451
451
452
452
453
453
454
454
455
455
456
456
457
457
458
458
459
459
460
460
461
461
462
462
463
463
464
464
465
465
466
466
467
467
468
468
469
469
470
470
471
471
472
472
473
473
474
474
475
475
476
476
477
477
478
478
479
479
480
480
481
481
482
482
483
483
484
484
485
485
486
486
487
487
488
488
489
489
490
490
491
491
492
492
493
493
494
494
495
495
496
496
497
497
498
498
499
499
500
500
501
501
502
502
503
503
504
504
505
505
506
506
507
507
508
508
509
509
510
510
511
511
512
512
513
513
514
514
515
515
516
516
517
517
518
518
519
519
520
520
521
521
522
522
523
523
524
524
525
525
526
526
527
527
528
528
529
529
530
530
531
531
532
532
533
533
534
534
535
535
536
536
537
537
538
538
539
539
540
540
541
541
542
542
543
543
544
544
545
545
546
546
547
547
548
548
549
549
550
550
551
551
552
552
553
553
554
554
555
555
556
556
557
557
558
558
559
559
560
560
561
561
562
562
563
563
564
564
565
565
566
566
567
567
568
568
569
569
570
570
571
571
572
572
573
573
574
574
575
575
576
576
577
577
578
578
579
579
580
580
581
581
582
582
583
583
584
584
585
585
586
586
587
587
588
588
589
589
590
590
591
591
592
592
593
593
594
594
595
595
596
596
597
597
598
598
599
599
600
600
601
601
602
602
603
603
604
604
605
605
606
606
607
607
608
608
609
609
610
610
611
611
612
612
613
613
614
614
615
615
616
616
617
617
618
618
619
619
620
620
621
621
622
622
623
623
624
624
625
625
626
626
627
627
628
628
629
629
630
630
631
631
632
632
633
633
634
634
635
635
636
636
637
637
638
638
639
639
640
640
641
641
642
642
643
643
644
644
645
645
646
646
647
647
648
648
649
649
650
650
651
651
652
652
653
653
654
654
655
655
656
656
657
657
658
658
659
659
660
660
661
661
662
662
663
663
664
664
665
665
666
666
667
667
668
668
669
669
670
670
671
671
672
672
673
673
674
674
675
675
676
676
677
677
678
678
679
679
680
680
681
681
682
682
683
683
684
684
685
685
686
686
687
687
688
688
689
689
690
690
691
691
692
692
693
693
694
694
695
695
696
696
697
697
698
698
699
699
700
700
701
701
702
702
703
703
704
704
705
705
706
706
707
707
708
708
709
709
710
710
711
711
712
712
713
713
714
714
715
715
716
716
717
717
718
718
719
719
720
720
721
721
722
722
723
723
724
724
725
725
726
726
727
727
728
728
729
729
730
730
731
731
732
732
733
733
734
734
735
735
736
736
737
737
738
738
739
739
740
740
741
741
742
742
743
743
744
744
745
745
746
746
747
747
748
748
749
749
750
750
751
751
752
752
753
753
754
754
755
755
756
756
757
757
758
758
759
759
760
760
761
761
762
762
763
763
764
764
765
765
766
766
767
767
768
768
769
769
770
770
771
771
772
772
773
773
774
774
775
775
776
776
777
777
778
778
779
779
780
780
781
781
782
782
783
783
784
784
785
785
786
786
787
787
788
788
789
789
790
790
791
791
792
792
793
793
794
794
795
795
796
796
797
797
798
798
799
799
800
800
801
801
802
802
803
803
804
804
805
805
806
806
807
807
808
808
809
809
810
810
811
811
812
812
813
813
814
814
815
815
816
816
817
817
818
818
819
819
820
820
821
821
822
822
823
823
824
824
825
825
826
826
827
827
828
828
829
829
830
830
831
831
832
832
833
833
834
834
835
835
836
836
837
837
838
838
839
839
840
840
841
841
842
842
843
843
844
844
845
845
846
846
847
847
848
848
849
849
850
850
851
851
852
852
853
853
854
854
855
855
856
856
857
857
858
858
859
859
860
860
861
861
862
862
863
863
864
864
865
865
866
866
867
867
868
868
869
869
870
870
871
871
872
872
873
873
874
874
875
875
876
876
877
877
878
878
879
879
880
880
881
881
882
882
883
883
884
884
885
885
886
886
887
887
888
888
889
889
890
890
891
891
892
892
893
893
894
894
895
895
896
896
897
897
898
898
899
899
900
900
901
901
902
902
903
903
904
904
905
905
906
906
907
907
908
908
909
909
910
910
911
911
912
912
913
913
914
914
915
915
916
916
917
917
918
918
919
919
920
920
921
921
922
922
923
923
924
924
925
925
926
926
927
927
928
928
929
929
930
930
931
931
932
932
933
933
934
934
935
935
936
936
937
937
938
938
939
939
940
940
941
941
942
942
943
943
944
944
945
945
946
946
947
947
948
948
949
949
950
950
951
951
952
952
953
953
954
954
955
955
956
956
957
957
958
958
959
959
960
960
961
961
962
962
963
963
964
964
965
965
966
966
967
967
968
968
969
969
970
970
971
971
972
972
973
973
974
974
975
975
976
976
977
977
978
978
979
979
980
980
981
981
982
982
983
983
984
984
985
985
986
986
987
987
988
988
989
989
990
990
991
991
992
992
993
993
994
994
995
995
996
996
997
997
998
998
999
999
1000
1000
1001
1001
1002
1002
1003
1003
1004
1004
1005
1005
1006
1006
1007
1007
1008
1008
1009
1009
1010
1010
1011
1011
1012
1012
1013
1013
1014
1014
1015
1015
1016
1016
1017
1017
1018
1018
1019
1019
1020
1020
1021
1021
1022
1022
1023
1023
1024
1024
1025
1025
1026
1026
1027
1027
1028
1028
1029
1029
1030
1030
1031
1031
1032
1032
1033
1033
1034
1034
1035
1035
1036
1036
1037
1037
1038
1038
1039
1039
1040
1040
1041
1041
1042
1042
1043
1043
1044
1044
1045
1045
1046
1046
1047
1047
1048
1048
1049
1049
1050
1050
1051
1051
1052
1052
1053
1053
1054
1054
1055
1055
1056
1056
1057
1057
1058
1058
1059
1059
1060
1060
1061
1061
1062
1062
1063
1063
1064
1064
1065
1065
1066
1066
1067
1067
1068
1068
1069
1069
1070
1070
1071
1071
1072
1072
1073
1073
1074
1074
1075
1075
1076
1076
1077
1077
1078
1078
1079
1079
1080
1080
1081
1081
1082
1082
1083
1083
1084
1084
1085
1085
1086
1086
1087
1087
1088
1088
1089
1089
1090
1090
1091
1091
1092
1092
1093
1093
1094
1094
1095
1095
1096
1096
1097
1097
1098
1098
1099
1099
1100
1100
1101
1101
1102
1102
1103
1103
1104
1104
1105
1105
1106
1106
1107
1107
1108
1108
1109
1109
1110
1110
1111
1111
1112
1112
1113
1113
1114
1114
1115
1115
1116
1116
1117
1117
1118
1118
1119
1119
1120
1120
1121
1121
1122
1122
1123
1123
1124
1124
1125
1125
1126
1126
1127
1127
1128
1128
1129
1129
1130
1130
1131
1131
1132
1132
1133
1133
1134
1134
1135
1135
1136
1136
1137
1137
1138
1138
1139
1139
1140
1140
1141
1141
1142
1142
1143
1143
1144
1144
1145
1145
1146
1146
1147
1147
1148
1148
1149
1149
1150
1150
1151
1151
1152
1152
1153
1153
1154
1154
1155
1155
1156
1156
1157
1157
1158
1158
1159
1159
1160
1160
1161
1161
1162
1162
1163
1163
1164
1164
1165
1165
1166
1166
1167
1167
1168
1168
1169
1169
1170
1170
1171
1171
1172
1172
1173
1173
1174
1174
1175
1175
1176
1176
1177
1177
1178
1178
1179
1179
1180
1180
1181
1181
1182
1182
1183
1183
1184
1184
1185
1185
1186
1186
1187
1187
1188
1188
1189
1189
1190
1190
1191
1191
1192
1192
1193
1193
1194
1194
1195
1195
1196
1196
1197
1197
1198
1198
1199
1199
1200
1200
1201
1201
1202
1202
1203
1203
1204
1204
1205
1205
1206
1206
1207
1207
1208
1208
1209
1209
1210
1210
1211
1211
1212
1212
1213
1213
1214
1214
1215
1215
1216
1216
1217
1217
1218
1218
1219
1219
1220
1220
1221
1221
1222
1222
1223
1223
1224
1224
1225
1225
1226
1226
1227
1227
1228
1228
1229
1229
1230
1230
1231
1231
1232
1232
1233
1233
1234
1234
1235
1235
1236
1236
1237
1237
1238
1238
1239
1239
1240
1240
1241
1241
1242
1242
1243
1243
1244
1244
1245
1245
1246
1246
1247
1247
1248
1248
1249
1249
1250
1250
1251
1251
1252
1252
1253
1253
1254
1254
1255
1255
1256
1256
1257
1257
1258
1258
1259
1259
1260
1260
1261
1261
1262
1262
1263
1263
1264
1264
1265
1265
1266
1266
1267
1267
1268
1268
1269
1269
1270
1270
1271
1271
1272
1272
1273
1273
1274
1274
1275
1275
1276
1276
1277
1277
1278
1278
1279
1279
1280
1280
1281
1281
1282
1282
1283
1283
1284
1284
1285
1285
1286
1286
1287
1287
1288
1288
1289
1289
1290
1290
1291
1291
1292
1292
1293
1293
1294
1294
1295
1295
1296
1296
1297
1297
1298
1298
1299
1299
1300
1300
1301
1301
1302
1302
1303
1303
1304
1304
1305
1305
1306
1306
1307
1307
1308
1308
1309
1309
1310
1310
1311
1311
1312
1312
1313
1313
1314
1314
1315
1315
1316
1316
1317
1317
1318
1318
1319
1319
1320
1320
1321
1321
1322
1322
1323
1323
1324
1324
1325
1325
1326
1326
1327
1327
1328
1328
1329
1329
1330
1330
1331
1331
1332
1332
1333
1333
1334
1334
1335
1335
1336
1336
1337
1337
1338
1338
1339
1339
1340
1340
1341
1341
1342
1342
1343
1343
1344
1344
1345
1345
1346
1346
1347
1347
1348
1348
1349
1349
1350
1350
1351
1351
1352
1352
1353
1353
1354
1354
1355
1355
1356
1356
1357
1357
1358
1358
1359
1359
1360
1360
1361
1361
1362
1362
1363
1363
1364
1364
1365
1365
1366
1366
1367
1367
1368
1368
1369
1369
1370
1370
1371
1371
1372
1372
1373
1373
1374
1374
1375
1375
1376
1376
1377
1377
1378
1378
1379
1379
1380
1380
1381
1381
1382
1382
1383
1383
1384
1384
1385
1385
1386
1386
1387
1387
1388
1388
1389
1389
1390
1390
1391
1391
1392
1392
1393
1393
1394
1394
1395
1395
1396
1396
1397
1397
1398
1398
1399
1399
1400
1400
1401
1401
1402
1402
1403
1403
1404
1404
1405
1405
1406
1406
1407
1407
1408
1408
1409
1409
1410
1410
1411
1411
1412
1412
1413
1413
1414
1414
1415
1415
1416
1416
1417
1417
1418
1418
1419
1419

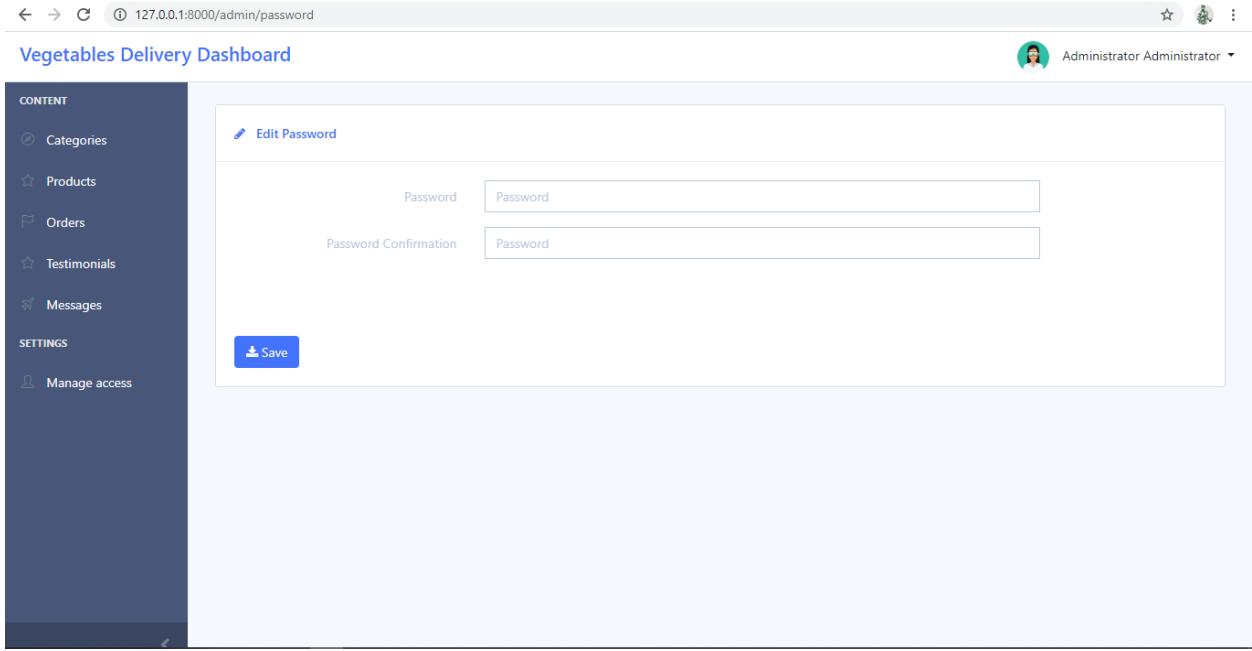
```



The screenshot shows a code editor with the file `edit-profile.blade.php` open. The code is written in Blade PHP template syntax. It includes sections for header and footer, a form for editing profile information, and a card footer with a submit button. The submit button is disabled if the submission is in progress.

```
resources > views > admin > profile > edit-profile.blade.php > ...
68 |     </div>
69 |     </div>
70 |
71 |     </div>
72 |
73 |     <div class="card-footer">
74 |         <button type="submit" class="btn btn-primary" :disabled="submiting">
75 |             <i class="fa" :class="submiting ? 'fa-spinner' : 'fa-download'"></i>
76 |             {{ trans('brackets/admin-ui::admin.btn.save') }}
77 |         </button>
78 |     </div>
79 |
80 |     </form>
81 |
82 |     </profile-edit-profile-form>
83 |
84 | </div>
85 |
86 | </div>
87 |
88 @endsection
```

Figure 79: Edit Profile.php



The screenshot shows a web application dashboard titled "Vegetables Delivery Dashboard". The left sidebar contains navigation links for "CONTENT" (Categories, Products, Orders, Testimonials, Messages) and "SETTINGS" (Manage access). The main content area is titled "Edit Password" and contains two input fields: "Password" and "Password Confirmation", both currently empty. A blue "Save" button is located at the bottom of the form. The top right corner shows a user profile for "Administrator Administrator".

Figure 80: Admin Password Edit Form

```

edit-password.blade.php X
resources > views > admin > profile > edit-password.blade.php > ...
1  @extends('brackets/admin-ui::admin.layout.default')
2
3  @section('title', trans('admin.admin-user.actions.edit_password'))
4
5  @section('body')
6
7      <div class="container-xl">
8
9          <div class="card">
10
11              <profile-edit-password-form
12                  :action="'{ url('admin/password') }'"
13                  :data="{!! $adminUser->toJson() !!}"
14                  inline-template>
15
16                  <form class="form-horizontal form-edit" method="post" @submit.prevent="onSubmit" :action="action">
17
18                      <div class="card-header">
19                          <i class="fa fa-pencil"></i> {{ trans('admin.admin-user.actions.edit_password') }}
20                      </div>
21
22                      <div class="card-body">
23

```

```

edit-password.blade.php X
resources > views > admin > profile > edit-password.blade.php > ...
24      <div class="form-group row align-items-center" :class="{'has-danger': errors.has('password'), 'has-success': fields.password && fields.password.valid }">
25          <label for="password" class="col-form-label text-md-right" :class="isFormLocalized ? 'col-md-4' : 'col-md-3'">{{ trans('admin.admin-user.columns.password') }}</label>
26          <div :class="isFormLocalized ? 'col-md-4' : 'col-md-9 col-xl-7'">
27              <input type="password" v-model="form.password" v-validate="required|min:7" @input="validate($event)" class="form-control" :class="{'form-control-dirty': errors.has('password')}">
28              <div v-if="errors.has('password')" class="form-control-feedback form-text" v-cloak>{!! errors.first('password') !!}</div>
29          </div>
30      </div>
31      <div class="form-group row align-items-center" :class="{'has-danger': errors.has('password_confirmation'), 'has-success': fields.password_confirmation && fields.password_confirmation.valid }">
32          <label for="password_confirmation" class="col-form-label text-md-right" :class="isFormLocalized ? 'col-md-4' : 'col-md-3'">{{ trans('admin.admin-user.columns.password_confirmation') }}</label>
33          <div :class="isFormLocalized ? 'col-md-4' : 'col-md-9 col-xl-7'">
34              <input type="password" v-model="form.password_confirmation" v-validate="required|confirmed:password|min:7" @input="validate($event)" class="form-control" :class="{'form-control-dirty': errors.has('password_confirmation')}">
35              <div v-if="errors.has('password_confirmation')" class="form-control-feedback form-text" v-cloak>{!! errors.first('password_confirmation') !!}</div>
36          </div>
37      </div>
38
39      <div class="card-footer">
40          <button type="submit" class="btn btn-primary" :disabled="submiting">
41              <i class="fa" :class="submiting ? 'fa-spinner' : 'fa-download'"></i>
42              {{ trans('brackets/admin-ui::admin.btn.save') }}
43          </button>
44      </div>
45
46
47      </div>
48
49      </form>
50
51      </profile-edit-password-form>
52
53      </div>
54
55      </div>
56
57  @endsection

```

Figure 81: Edit Password.php

## References

- Analysis.* (2020, 10 10). Retrieved from businessdictionary.com:  
<http://www.businessdictionary.com/definition/analysis.html>
- BEHAVIORAL MODELING.* (2020, 10 03). Retrieved from O'Reilly:  
[https://www.oreilly.com/library/view/systems-analysis-and-9781118037423/11\\_chapter006.html](https://www.oreilly.com/library/view/systems-analysis-and-9781118037423/11_chapter006.html)
- Black Box Testing.* (2020, 10 10). Retrieved from guru99: <https://www.guru99.com/black-box-testing.html>
- Data Modelling.* (2020, 10 05). Retrieved from guru99: <https://www.guru99.com/data-modelling-conceptual-logical.html>
- feasibility study.* (2020, 10 02). Retrieved from Investopedia:  
<https://www.investopedia.com/terms/f/feasibility-study.asp>
- Feasibility Study.* (2020, 10 04). Retrieved from Simplilearn: <https://www.simplilearn.com/feasibility-study-article>
- Feasibility Study.* (2020, 10 09). Retrieved from simplilearn: <https://www.simplilearn.com/feasibility-study-article>
- Functional Requirements.* (2020, 09 23). Retrieved from ofnisystems.com:  
<http://www.ofnisystems.com/services/validation/functional-requirements/>
- Interviewing in Information Gathering.* (2020, 10 06). Retrieved from System Analysis:  
<https://www.w3computing.com/systemsanalysis/interviewing/>
- MoSCoW Prioritization.* (2020, 10 04). Retrieved from ProductPlan:  
<https://www.productplan.com/glossary/moscow-prioritization/>
- Sequence Diagram.* (2020, 9 28). Retrieved from Visual Paradigm: <https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-sequence-diagram/>
- Software testing.* (2020, 09 29). Retrieved from Software testing material:  
<https://www.softwaretestingmaterial.com/software-testing/>
- STRUCTURAL MODELING.* (2020, 09 30). Retrieved from O'Reilly:  
[https://www.oreilly.com/library/view/systems-analysis-and-9781118037423/10\\_chapter005.html](https://www.oreilly.com/library/view/systems-analysis-and-9781118037423/10_chapter005.html)
- System Analysis.* (2020, 09 28). Retrieved from Using Questionnaires in Information Gathering:  
<https://www.w3computing.com/systemsanalysis/using-questionnaires-information-gathering/>
- System Architecture.* (2020, 09 27). Retrieved from Johner Institute: <https://www.johner-institute.com/articles/product-development/system-architecture/>
- System Requirements Specifications.* (2020, 09 30). Retrieved from inflectra:  
<https://www.inflectra.com/ideas/topic/requirements-definition.aspx>

*UML - Class Diagram.* (2020, 10 01). Retrieved from tutorialspoint:  
[https://www.tutorialspoint.com/uml/uml\\_class\\_diagram.htm](https://www.tutorialspoint.com/uml/uml_class_diagram.htm)

*Unit Testing Tutorial.* (2020, 10 10). Retrieved from guru99: <https://www.guru99.com/unit-testing-guide.html>

*Use Case Diagram.* (2020, 07 31). Retrieved from Visual Paradigm: <https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-use-case-diagram/>