Project Report on

VEGETABLE MARKET LOG PROJECT

Submitted in partial fulfilment of the requirement for the award of the degree of

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE & ENGINEERING

Submitted by:

SAMRIDHI DUGGAL	**2016989**
SATYAM KUSHWAHA	**2017549**
ANUKRITI SHRIVASTAVA	**2017312**
RAHUL KUMAR	**2017354**

Under the Mentorship of

Mr. Yuvraj Joshi
Assistant Professor

Department of Computer Science and Engineering
Graphic Era (Deemed to be University)
Dehradun, Uttarakhand

July-2023

Graphic Era
Deemed to be
University
DEHRADUN

CANDIDATE'S DECLARATION

I hereby certify that the work which is being presented in the project report entitled **"VEGETABLE MARKET LOG PROJECT"** in partial fulfilment of the requirements for the award of the Degree of Bachelor of Technology in Computer Science and Engineering of the Graphic Era (Deemed to be University), Dehradun shall be carried out by the under the mentorship of **Mr. Yuvraj Joshi, Assistant Professor**, Department of Computer Science and Engineering, Graphic Era (Deemed to be University), Dehradun.

Samridhi Duggal	2016989
Satyam Kushwaha	2017549
Anukriti Shrivastava	2017312
Rahul Kumar	2017354

Table of Contents

Chapter No.	Description	Page No.
HTML, CSS		
Chapter 1	Introduction	4
Chapter 2	User Interface	5
Chapter 3	Technologies Used	7
Chapter 4	Result and Discussion	10
Chapter 5	Conclusion & Future Work	13
Java Script		
Chapter 1	Problem Statement	14
Chapter 2	Dynamic Website	14
Chapter 3	Technologies Used	16
Chapter 4	Result and Discussion	17
Chapter 5	Conclusion & Future Work	20
React js		
Chapter 1	Problem Statement	21
Chapter 2	Technologies Used	21
Chapter 3	Transformation	23
Backend		
Chapter 1	Problem Statement	26
Chapter 2	Technologies Used	26
Chapter 3	Result and Discussion	28
Chapter 4	Conclusion & Future Work	29
ML		31
Chapter 1	Problem Statement	31
Chapter 2	Data & Methodology	32
Chapter 3	Libraries	32
Chapter 4	Result and Discussion	33
Chapter 5	Conclusion & Future Work	

WEEK 1

Chapter 1

Introduction

PROBLEM STATEMENT: - To create and design the User Interface of Online Vegetable Market Website using HTML and CSS.

VegEase – Farm Fresh Goodness, Every day.

In a world that's constantly on the move, finding the time to source and prepare fresh, healthy vegetables can be a challenge. But fear not, because VegEase is here to revolutionize the way you access and enjoy the goodness of nature's bounty. Welcome to a new era of convenience, nutrition, and flavour!

Introducing VegEase:

VegEase is your ultimate solution for hassle-free access to farm-fresh vegetables, conveniently delivered right to your doorstep. Our mission is simple: to bring the farm to your kitchen, ensuring you have access to the finest quality produce without compromising on your busy lifestyle.

Why Choose VegEase Express?

Unparalleled Convenience: Say goodbye to time-consuming trips to the grocery store and the uncertainty of produce quality. With VegEase, the freshest vegetables come to you, allowing you to reclaim valuable time for what truly matters.

Nutrient-Rich Selection: Our vegetables are carefully selected from local farms, ensuring they're packed with essential vitamins, minerals, and antioxidants that contribute to your overall well-being.

Variety at Your Fingertips: Discover an extensive range of vegetables, from leafy greens to vibrant bell peppers and beyond. With VegEase, you have the power to customize your order according to your preferences.

Supporting Local Agriculture: By choosing VegEase, you're not only benefiting yourself but also contributing to the growth and sustainability of local farmers and the community.

Recipe Inspiration: Stuck in a culinary rut? VegEase provides you with delightful recipe suggestions that make cooking with fresh vegetables a creative and enjoyable experience.

How VegEase Works:

Browse and Select: Explore our user-friendly online platform showcasing an assortment of fresh vegetables. Choose the ones that catch your eye and add them to your cart.

Delivered to Delight: Sit back and relax as our dedicated delivery team brings the garden-fresh goodness to your doorstep. Our meticulous packaging ensures your vegetables retain their peak freshness.

Culinary Adventures Await: Unleash your inner chef with our collection of inspired recipes. From quick weekday meals to elaborate weekend feasts, VegEase is your culinary companion.

Join the VegEase Community:

Embrace a lifestyle that celebrates both wellness and convenience. By becoming a part of VegEase, you're joining a community of health-conscious individuals who understand the importance of nurturing their bodies with the best nature has to offer.

Chapter 2

User Interface (Design Layout)

Design Overview: The website follows a contemporary design approach, combining attractive visuals, intuitive navigation, and engaging content. It consists of several sections, including a header with a logo and navigation menu, a home section with a welcoming message and a call-to-action button, an about section providing information about the company, and a products section showcasing the latest vegetable offerings. The design incorporates the use of images, icons, and appealing color schemes to enhance visual appeal and user engagement.

Navigation and User Interface: The navigation menu, located in the header, allows users to easily access different sections of the website. The use of clear and descriptive anchor links helps users navigate to specific sections such as Home, About, Products, Services, and Blog. The inclusion of icons for search and cart functionalities in the header enhances user convenience and provides a familiar interface.

Home Section: The home section aims to captivate visitors with a visually appealing layout. The heading combines an eye-catching font style and a tagline, emphasizing the freshness and organic nature of the vegetables. A brief description creates an emotional connection with the user, and a call-to-action button encourages further exploration of the website.

About Section: The about section provides an overview of the company and its values. The inclusion of an image adds visual interest and supports the content. The tagline and accompanying paragraph highlight the company's mission and establish credibility. The "Learn More" button encourages users to delve deeper into the company's story.

Products Section: The products section showcases the latest vegetable offerings. A clear heading introduces the section, and a descriptive paragraph provides additional information. Each vegetable product is presented in a visually appealing card layout, featuring an image, a tagline, and a "Learn More" button. The addition of a cart icon allows users to easily add products to their shopping cart.

Visual Elements and Branding: The website employs consistent branding elements, such as the logo and color scheme, to create a cohesive visual identity. Images are used strategically throughout the website to enhance the appeal of the vegetables and evoke a sense of freshness and quality.

Responsiveness and Compatibility: The website appears to be responsive, adapting well to different screen sizes and devices. The use of the meta viewport tag ensures proper scaling on

various devices. Additionally, external CSS and font libraries, such as Font Awesome, are properly linked, ensuring compatibility and accessibility across different browsers.

Chapter 3

Technologies Used

HTML

HTML (Hypertext Markup Language) is the standard markup language used for creating the structure and content of web pages. It is the fundamental language of the World Wide Web and provides a set of tags or elements that define the structure and semantics of a web document. Here is the list of HTML tags used in the provided HTML code:

1. **`<html>`:** The root element that wraps the entire HTML document.
2. **`<head>`:** Contains meta information and external resources for the webpage.
3. **`<meta>`:** Defines metadata such as character set and viewport settings.
4. **`<title>`:** Specifies the title of the webpage displayed in the browser's title bar.
5. **`<link>`:** Used to link external stylesheets and font libraries to the HTML document.
6. **`<body>`:** Represents the main content area of the webpage.
7. **`<nav>`:** Defines a section containing navigation elements.
8. **`<div>`:** A generic container used to group and style content.
9. **`<ul>`:** Represents an unordered list of items.
10. **`<li>`:** Represents a list item within an unordered or ordered list.
11. **`<a>`:** Creates a hyperlink to another webpage or a specific location within the same webpage.
12. **`<img>`:** Inserts an image into the webpage.
13. **`<section>`:** Defines a standalone section or area of content.
14. **`<h1>`, `<h2>`:** Headings of different levels, used to structure the content hierarchy.
15. **`<p>`:** Represents a paragraph of text.
16. **`<i>`:** Displays italicized text or icons.
17. **`<br>`:** Inserts a line break.
18. **`id="..."`:** Assigns a unique identifier to an HTML element.
19. **`class="..."`:** Assigns one or more classes to an HTML element for styling and targeting with CSS.

These tags are essential building blocks for structuring and formatting the content within an HTML document.

CSS:-

The given CSS code defines the styling attributes for various elements of a website. Here is a breakdown of the attributes used in the CSS file:

- The universal selector (*) sets the margin and padding to 0 for all elements and includes the CSS box-sizing property to ensure proper box model behaviour.
- The HTML element is styled to have smooth scrolling behaviour using scroll-behaviour: smooth.
- The navigation bar (nav) is positioned at the top of the page and has a white background with a box shadow. It contains a logo and a list of navigation links.
- The logo image within the navigation bar is styled with a width of 60px and a cursor pointer.
- The navigation links (li a) are styled with a color, and their color changes to a different shade on hover.
- The search icon within the navigation bar is styled with a font size and changes color on hover.

The CSS code also includes styling rules for different sections of the website, such as the home section, about section, products section, banner section, services section, blog section, and footer section. Each section has its own set of styling attributes defining the layout, positioning, background, fonts, colors, and other visual properties.

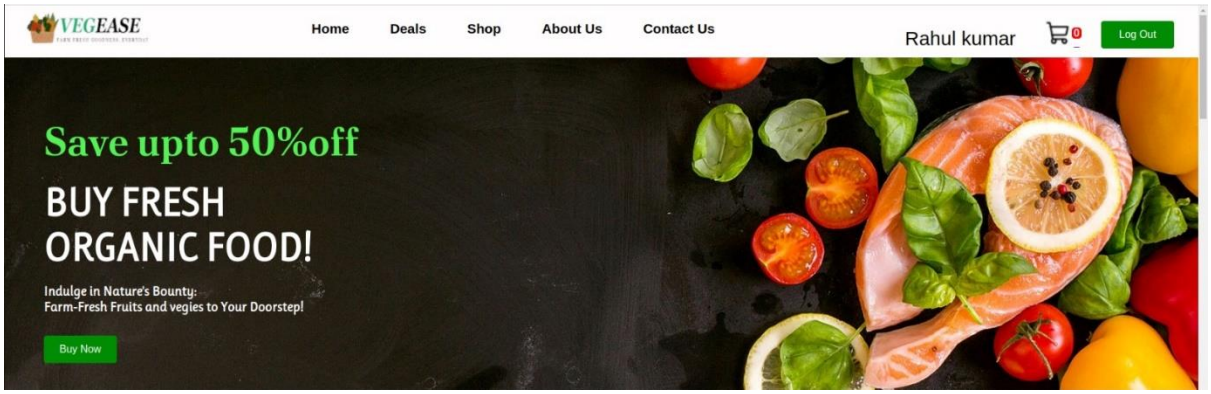
The CSS code also includes animations and transitions for certain elements.

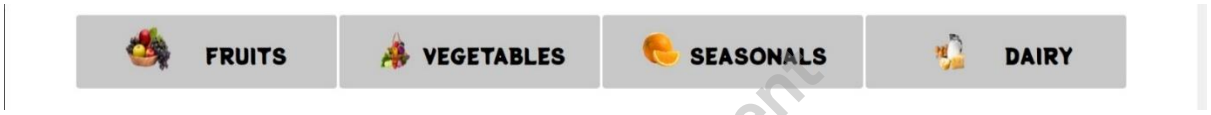
Overall, the CSS code aims to create a visually appealing and responsive website layout with consistent styling across different sections and elements.

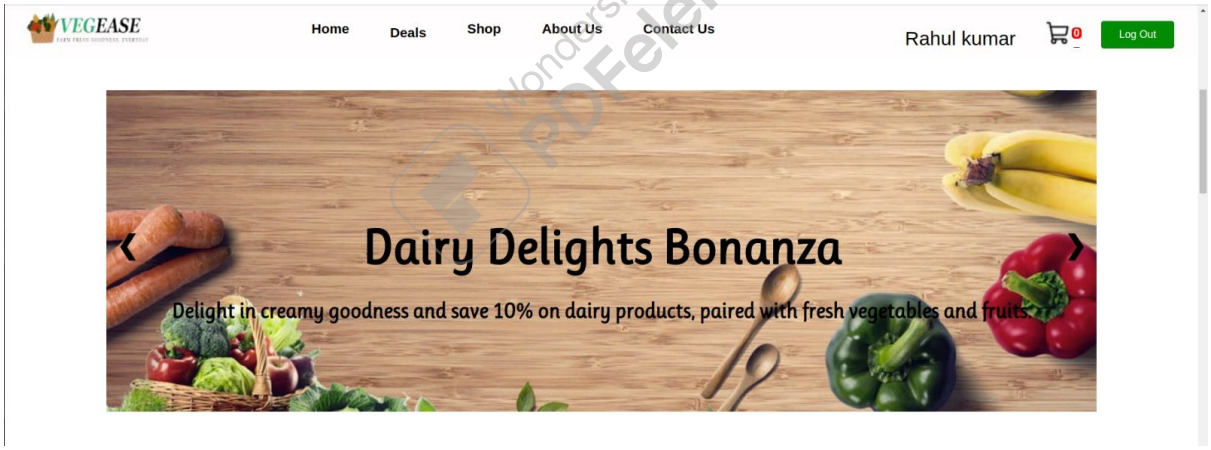
Chapter 4

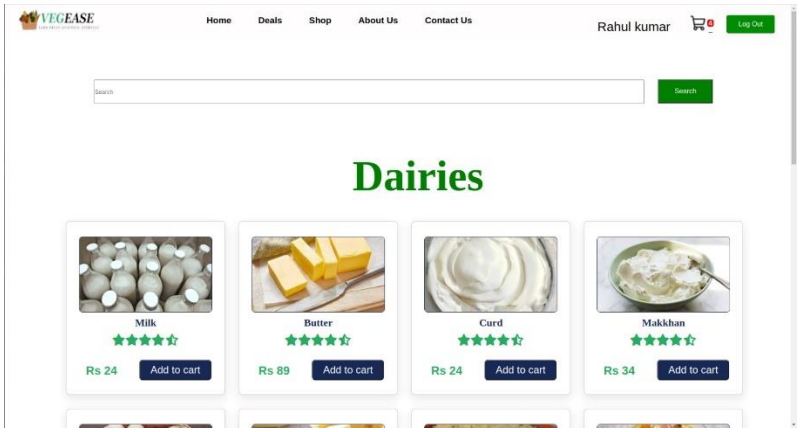
Result and Discussion

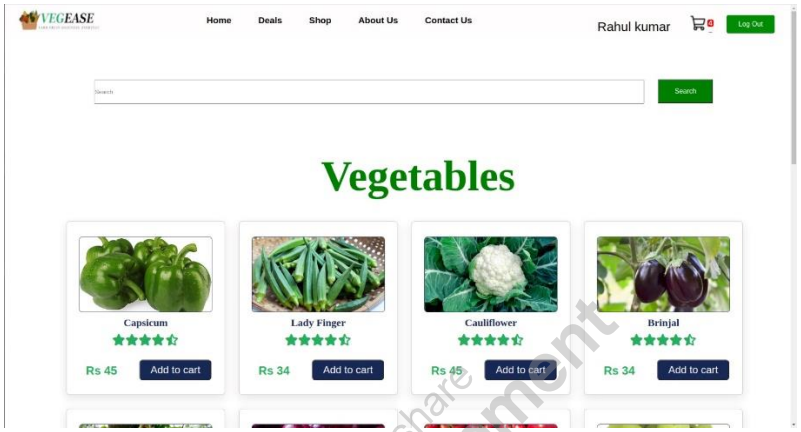
The outcome of the website is as follows: -

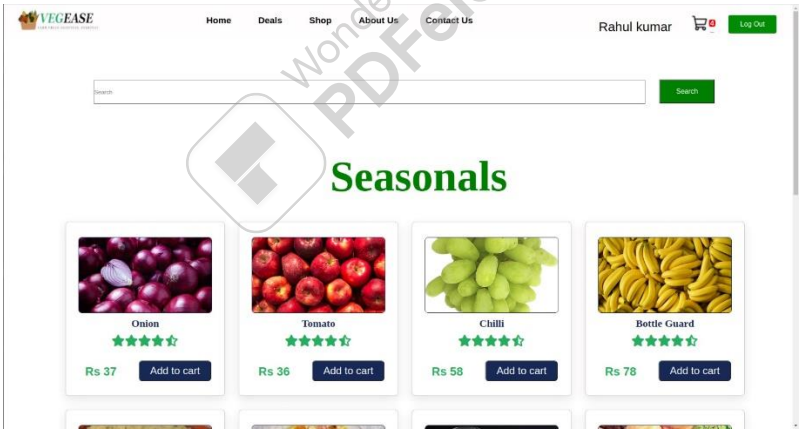


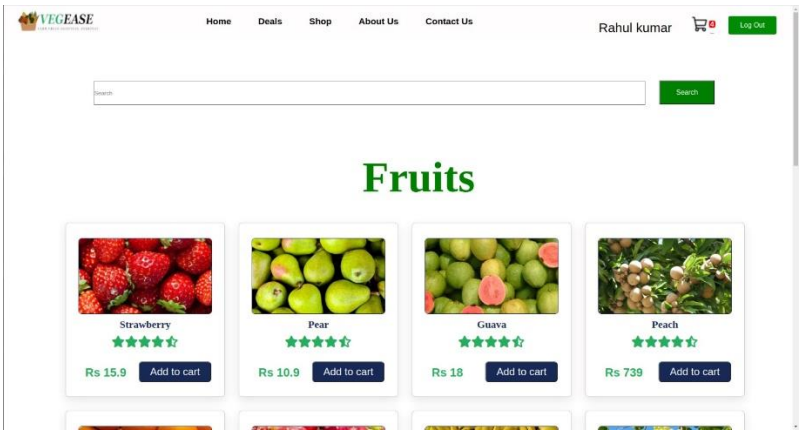








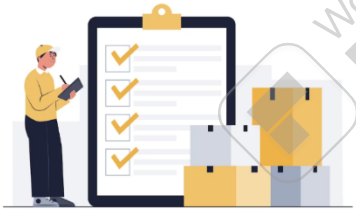




Welcome to Vegease! We are passionate about providing fresh, high-quality vegetables directly to your doorstep, making it convenient for you to access nutritious and locally sourced produce. At VegEase, we are passionate about bringing the freshest and most flavorful vegetables directly to your doorstep. Our journey began with a simple idea: to provide our community with convenient access to locally sourced, farm-fresh produce that's not only delicious but also supports our local farmers.Our mission is to promote healthier lifestyles and support local farmers by delivering the freshest and finest vegetables to our customers. We aim to make the process of buying vegetables easy, enjoyable, and sustainable. Moreover, Our mission is to make fresh, locally sourced vegetables accessible to everyone while promoting healthier living, supporting local farmers, and fostering sustainable practices. We are committed to providing our customers with the highest quality produce, delivered conveniently to their doorstep, empowering them to make nutritious choices for themselves and their families. Through our platform, we aim to strengthen communities by connecting consumers with local farmers and reducing our environmental impact by adopting eco-friendly packaging and delivery practices. Our ultimate goal is to inspire a positive change in the way people shop for vegetables and contribute to a healthier, more sustainable world.

Fast Delivery

Experience Fast delivery of fresh vegetables and fruits at your doorstep on time.

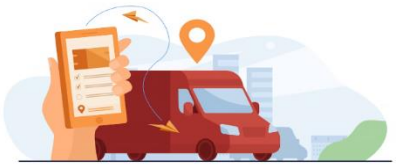


No Minimum order

Order the items according to your need without thinking about the minimum order required.

Track Location

Track your order to know in how much time the order will reach to your doorstep.



why us?

Fresh and Locally Sourced Vegetables: We partner with local farmers and suppliers to ensure that our vegetables are harvested at the peak of their freshness. By supporting local agriculture, we reduce our carbon footprint and contribute to the growth of the community.

Wide Selection: Our website offers a wide variety of vegetables to cater to all your culinary needs. Whether you're looking for common favorites or unique and exotic options, we've got you covered.

Convenience: Say goodbye to long supermarket lines and carrying heavy grocery bags. With our vegetable delivery service, you can shop from the comfort of your home and have your vegetables delivered straight to your door at a time that suits you.

Customizable Orders: We understand that everyone's needs are different. That's why we offer the option to customize your vegetable box based on your preferences. You can also choose from various subscription plans for regular deliveries.

Quality Assurance: We have a rigorous quality control process to ensure that only the freshest and best-quality vegetables make it to your doorstep. Our team carefully inspects each item before packing and shipping.

Sustainable Practices: We are committed to environmental sustainability and strive to minimize waste and packaging. Our deliveries are designed to be eco-friendly, reducing plastic usage wherever possible.

Customer Support: Our dedicated customer support team is always ready to assist you with any questions, concerns, or feedback you may have. Your satisfaction is our top priority, and we are here to make your shopping experience seamless.

Rewards and Loyalty Program: As a token of appreciation for choosing us, we offer a rewards and loyalty program that allows you to earn points with every purchase, leading to exclusive discounts and special offers.

Our Team

Behind every pixel, line of code, and design element of our website, there's a team of dedicated individuals who are passionate about creating the best possible online experience for you. The team memebers are as follows: Anukriti, Rahul, Samridhi, Satyam and Tushar. Together, the team has poured their creativity, expertise, and passion into crafting a website that embodies the essence of VegEase. We're not just delivering vegetables; we're delivering an online experience that connects you with nature, community, and the joy of fresh, wholesome produce.

Contact

Help & Support
9876******
Vegease@Gmail.Com

Company

About Us
Our Team
Our Services

Legal

Terms & Conditions
Refund & Cancellation
Privacy Policy

Shop

Fruits
Vegetables
Seasonals
Herbs
Organic

GET IT ON Google Play
Download on the App Store

Get Social With Us

By continuing, you agree to the Terms & Conditions of VegEase.
©2023 VegEase

Privacy Policy

Last Updated: 09th August 2023.

At VegEase, we are committed to protecting your privacy and ensuring the security of your personal information. This Privacy Policy outlines how we collect, use, disclose, and safeguard your information when you use our website and services. By accessing or using our website, you agree to the practices described in this Privacy Policy.

Information We Collect

We may collect various types of information from you, including but not limited to:

- Personal Information: This includes your name, email address, phone number, and shipping address. We collect this information when you create an account, place an order, or communicate with us.
- Payment Information: When you make a purchase on our website, we collect your payment details, including credit card information. This information is securely transmitted and processed by third-party payment processors.
- Usage Data: We gather information about how you interact with our website. This may include your IP address, browser type, device information, pages visited, and actions taken.

How We Use Your Information:

We use the collected information for various purposes, including:

- Order Processing: We use your personal and payment information to process and fulfill your orders, send order confirmations, and provide customer support related to your purchases.
- Account Management: Your personal information is used to create and manage your account on our website.
- Communication: We may use your contact information to send you updates about your orders, promotions, newsletters, and other relevant information. You can opt-out of these communications at any time.
- Improving Our Services: We analyze usage data to understand how our website is used, identify trends, and improve user experience.

Data Sharing and Disclosure:

We may share your information with third parties under the following circumstances:

- Service Providers:We may share your information with third-party service providers who help us with payment processing, order fulfillment, website analytics, and other operational functions.
- Legal Requirements: We may disclose your information if required by law or in response to valid legal requests, such as court orders or government investigations.
- Business Transfers: If we undergo a merger, acquisition, or sale of assets, your information may be transferred to the acquiring entity.

Data Security:

We take data security seriously and implement various measures to protect your information from unauthorized access, alteration, or disclosure. However, no method of transmission over the internet or electronic storage is completely secure, and we cannot guarantee absolute security.

Your Choices:

You have the right to access, update, or delete your personal information. You can do this by logging into your account or contacting our customer support. You can also choose to unsubscribe from marketing communications.

Cookies and Tracking Technologies:

We use cookies and similar tracking technologies to enhance your browsing experience and collect usage data. You can manage your cookie preferences through your browser settings.

Children's Privacy:

Our website is not intended for children under the age of 13. We do not knowingly collect personal information from children. If you believe we have collected information from a child, please contact us to have it removed.

Changes to this Privacy Policy:

We may update this Privacy Policy from time to time to reflect changes in our practices or legal requirements. The "Last Updated" date at the beginning of the policy will indicate when the policy was last revised.

If you have any questions or concerns about this Privacy Policy or our data practices, please contact us at [9876******].

By using our website, you acknowledge that you have read and understood this Privacy Policy and agree to its terms.

Terms And Conditions

These Terms and Conditions (the "Agreement") govern your use of the vegetable delivery website provided by VegEase (the "Company," "we," "us," or "our"). By accessing and using the Website, you agree to abide by these terms. If you do not agree with any part of these terms, you should not use the Website.

Acceptance of Terms

By accessing or using the Website, you acknowledge that you have read, understood, and agree to be bound by these Terms and Conditions. If you do not agree with any part of these terms, please do not use the Website.

Account Creation

To access certain features of the Website, you may need to create an account. You are responsible for maintaining the confidentiality of your account information and password. You agree to notify us immediately of any unauthorized use of your account.

Vegetable Orders and Deliveries

- Ordering Process: You can place orders for vegetables through the Website. By placing an order, you agree to provide accurate, current, and complete information. You are responsible for reviewing your order before submitting it.
- Delivery: We will deliver the ordered vegetables to the address you provide during the ordering process. Delivery times are approximate and subject to change. We are not responsible for delays due to unforeseen circumstances.
- Cancellation and Refunds: Orders can be cancelled within a specified time frame before the scheduled delivery. Refunds for cancelled orders will be processed

- Cancellation and Refunds: Orders can be cancelled within a specified time frame before the scheduled delivery. Refunds for cancelled orders will be processed according to our Refund Policy.

Payment

- Payment Information: You agree to provide accurate and complete payment information when placing an order. You authorize us to charge the provided payment method for the total amount of your order, including any applicable taxes and fees.
- Price Changes: Prices for vegetables and delivery services are subject to change without notice. However, the price you are charged will be the price listed at the time of your order.

User Conduct

You agree not to use the Website for any unlawful or unauthorized purpose. You shall not engage in any activities that may harm the integrity of the Website or compromise its security.

Intellectual Property

The content, graphics, logos, and other materials on the Website are protected by intellectual property laws and are owned by the Company or its licensors. You may not use, reproduce, distribute, or modify any of the content without explicit written permission.

Privacy

Your use of the Website is also governed by our Privacy Policy, which explains how we collect, use, and protect your personal information.

Disclaimer of Warranties

The Website is provided "as is" without any warranties, express or implied. We do not guarantee the accuracy, reliability, or availability of the Website.

Limitation of Liability

In no event shall the Company be liable for any indirect, incidental, special, or consequential damages arising out of or in connection with the use of the Website.

Changes to Terms

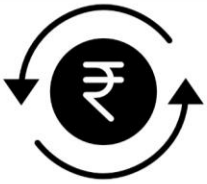
Changes to Terms

We reserve the right to modify or update these Terms and Conditions at any time. It is your responsibility to review these terms periodically.

Governing Law

This Agreement shall be governed by and construed in accordance with the laws of [Your Jurisdiction]. Any disputes arising from or related to this Agreement shall be subject to the exclusive jurisdiction of the courts in [Your Jurisdiction].

By using the Website, you acknowledge that you have read and understood these Terms and Conditions and agree to be bound by them. If you do not agree to these terms, you should not use the Website.



Refund and Cancellation Policy

Here, Customer means the person who is buying the stuff and We/Us would mean the 'VegEase' website.

Customer Cancellation

As per the rules, the Customer cannot cancel the order once it is confirmed. Customer can cancel the order only within one minute after placing it. However, after looking at the Customer's cancellation history, VegEase will decide whether to provide with a refund or not, because if a Customer is frequently doing this, then he/she is held accountable to degrade VegEase.

After one minute of placing the order, VegEase has the right to put 100% cancellation fees and no refund will be provided. If the order is prepaid, then no refund will be provided and incase the order is postpaid, then it will be adjusted in their subsequent placed order.

Other situations that may lead to Cancellation of order

VegEase will not be held responsible and has the right to apply penalty for the cases like:

- the address provided by the customer is wrong
- the customer is not replying to the phone calls or the emails while our delivery partner is trying to contact them
- the customer does not explain the exact location to the delivery partner and due to this our employee fails to deliver the order to you
- the customer books all those the items that are currently unavailable. In this case, our team will try to reach out to you through calls and 100% refund will be provided to the customer.

In case, the order gets cancelled from the merchant's side or due to some issues at VegEase, 100% cancellation will be provided to the customer.

Refunds

VegEase will provide refund on the orders that are prepaid. No refunds are provided for the postpaid orders.

Incase the order gets cancelled from the merchant's side or due to some issues at VegEase, 100% cancellation will be provided to the customer.

The refund will be provided only through online mode, no cash is being provided. The estimated time that be taken to provide with refunds is given below:

- UPI: 5-7 days
- Net Banking: 5-7 days
- PhonePe: 5-7 days
- Debit Card/Credit Card: 5-7 days

The Customer will be provided with full refund if the order takes more than two hours to be delivered.

In case of postpaid orders, the Customer does not need to pay for the order if:

- he receives an open order/ the seal of the order is broken
- the items are missing from the order that he ordered for
- wrong order is being delivered

The Customer will not have to pay for the order in these cases only if he reports about this to the Customer Care prior to the order is being delivered is marked on the site.

Important Note: VegEase is not responsible for the quality of the food that is being provided by the merchant to the Customer. We are only responsible for delivery of vegetables on time. In case any vendor is providing with extremely bad quality vegetables, you can complain to us and we will try our best to take action against them.

Chapter 5

Conclusion and Future Work

Online vegetable websites have emerged as a convenient and efficient way to connect consumers with local farmers and provide access to fresh, high-quality produce. They offer a wide range of advantages, including convenience, transparency, and supporting sustainable agriculture. While challenges exist, the potential for growth and innovation in this sector is substantial. By embracing technology and addressing consumer demands, online vegetable websites can continue to shape the future of food retail, fostering a healthier and more sustainable relationship between consumers and the agricultural ecosystem.

Future Work: -

1. Enhance Responsiveness: Ensure that the website is fully responsive across various devices and screen sizes. Test and optimize the layout and functionality for mobile devices, tablets, and larger screens.

2. Improve Visual Design: Consider enhancing the overall visual design of the website by refining the color scheme, typography, and imagery. Create a more visually appealing and cohesive look and feel.

3. Add Dynamic Content: Implement dynamic content features such as a blog management system or a content management system (CMS) to easily add, update, and manage blog posts, products, and other website content.

WEEK 2

Chapter 1

Introduction

PROBLEM STATEMENT: -Dynamic Website Enhancement with JavaScript.

Chapter 2

DYNAMIC WEBSITE

JavaScript Integration:

JavaScript is a versatile scripting language that runs in the browser, allowing us to manipulate the webpage's content and behaviour. Here are some ways we can use JavaScript to enhance the website:

Interactive Cart: Implement an interactive shopping cart that updates in real-time as users add or remove products. JavaScript will enable us to manage the cart's state and display the total price dynamically.

User Authentication: Add user registration and login functionality to create personalized accounts. JavaScript can help manage user sessions, handle authentication, and display personalized content.

Image Sliders: Create dynamic image sliders or carousels to showcase featured products or promotions. Users can navigate through images with ease.

Search Functionality: Implement a search bar with real-time suggestions as users' type. JavaScript can send search queries to the server and display results dynamically.

Feedback Forms: Add interactive feedback forms that validate user input and provide immediate feedback. JavaScript can handle form validation and submission.

Dynamic Content Loading: Load content dynamically without refreshing the page. For instance, load additional product details or blog posts when users click "Learn More" buttons.

Real-time Updates: Display real-time updates, such as stock availability or price changes, without the need for manual page refresh.

Smooth Page Transitions: Create smooth transitions between different sections of the website to enhance the user experience. JavaScript can control scrolling behaviour and animations.

Analytics Integration: Integrate JavaScript-based analytics tools to track user behaviour and gather data for website optimization.

Chapter 3

Technologies Used

JAVA SCRIPT: -

To implement JavaScript functionality on the Online Vegetable Market website, the following steps were taken:

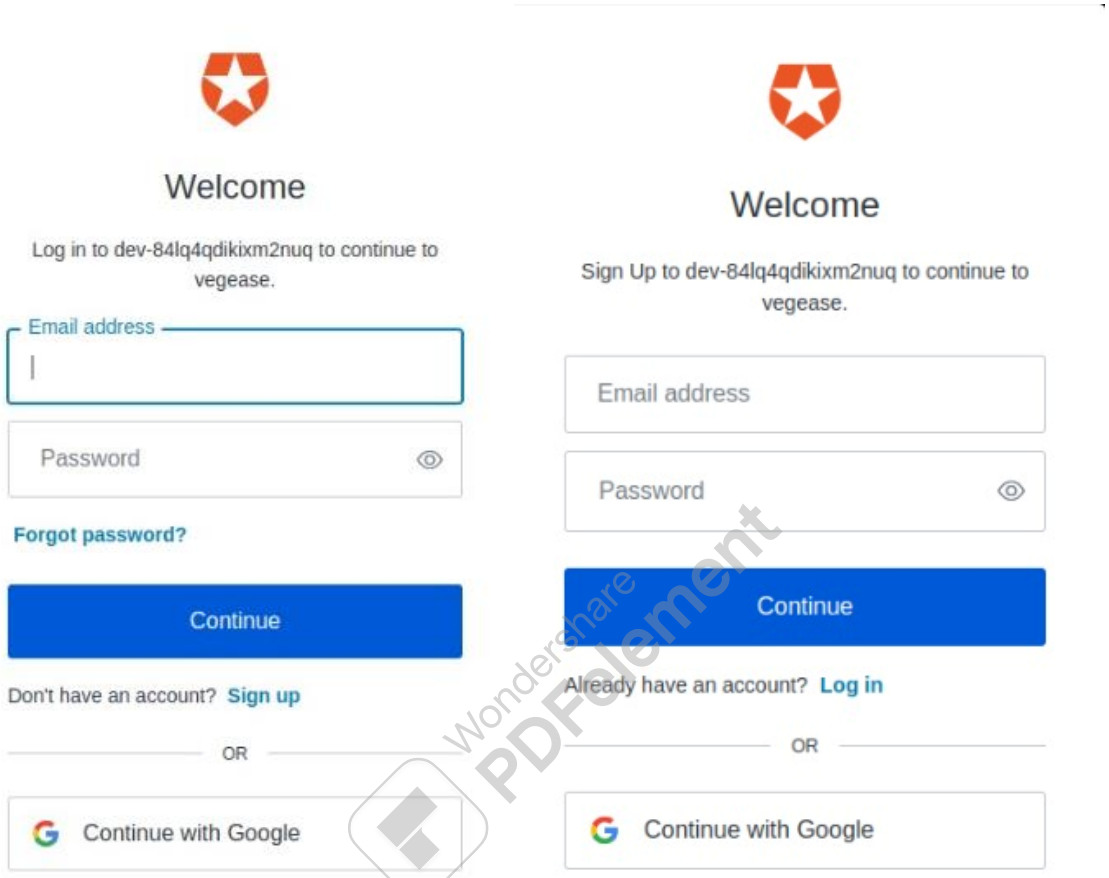
Cart Functionality: JavaScript was used to create an interactive shopping cart. When users click the "Add to Cart" button on a product card, JavaScript adds the selected item to the cart. The cart dynamically updates to show the added items and calculate the total price. Users can also remove items from the cart, which updates the cart in real-time.

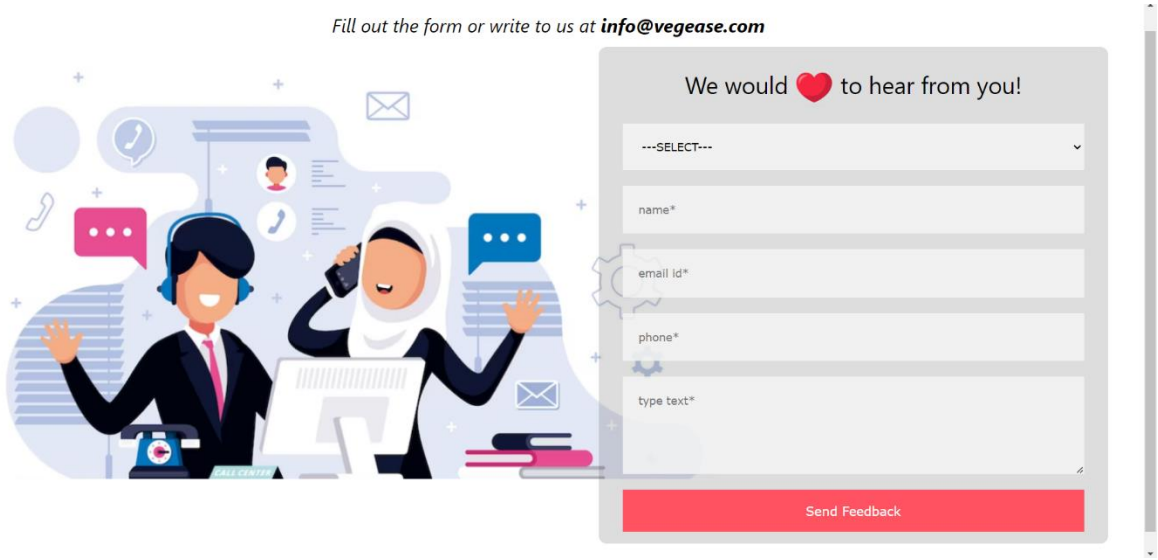
User Authentication: JavaScript was employed to handle user registration and login. Users can create accounts, log in, and stay authenticated across different pages. Their personalized information and order history are accessible through their accounts.

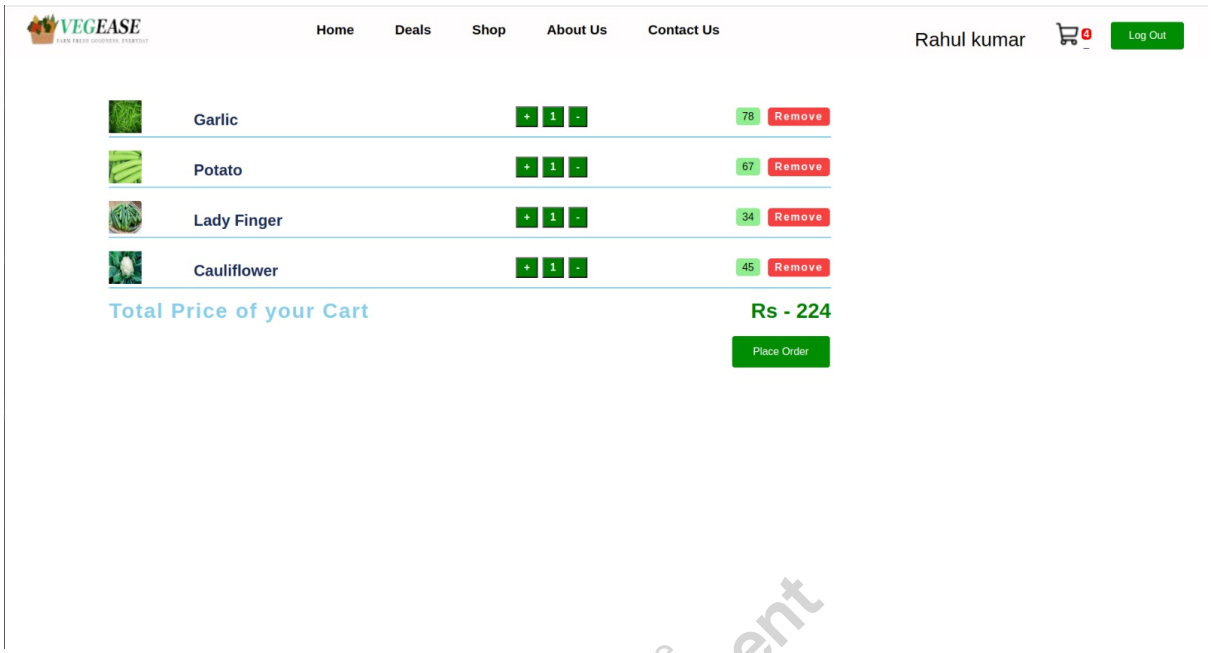
Chapter 4

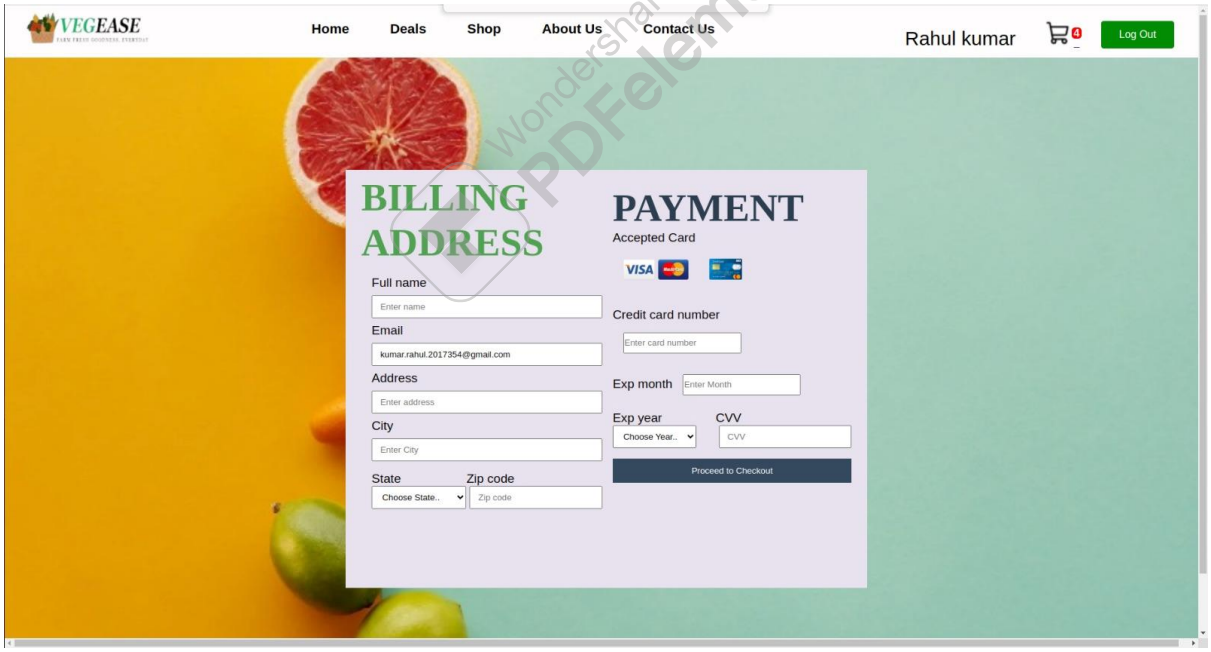
Result and Discussion

The outcome of the website is as follows: -









Chapter 5

Conclusion and Future Work

The integration of JavaScript has significantly improved the Online Vegetable Market website, making it more dynamic and user-centric. Users can now interact with the website in real-time, creating a more engaging and convenient experience.

Future Work:

Mobile App: Consider developing a mobile app using JavaScript frameworks like React Native or Flutter to reach a broader audience and provide a native mobile experience.

Push Notifications: Implement push notifications to alert users about special promotions, new arrivals, or order updates, increasing user engagement.

WEEK 3

Chapter 1

Introduction

PROBLEM STATEMENT: - The SRS outlines the requirements for an online vegetable website developed using React.js.

Chapter 2

Technologies Used

React Framework: Building Dynamic User Interfaces

React is an open-source JavaScript library for building user interfaces (UIs), specifically designed to create interactive and dynamic front-end applications. Developed and maintained by Facebook, react has gained widespread adoption due to its component-based architecture and efficient rendering capabilities.

Key Concepts and Features:

Component-Based Architecture: React revolves around the concept of components, which are self-contained, reusable building blocks for UI elements. Components encapsulate both the UI and the logic associated with it, making development modular and maintainable.

Virtual DOM (Document Object Model): React employs a virtual DOM, an in-memory representation of the actual DOM. When data changes occur, React compares the virtual DOM with the previous version, identifies differences, and then updates only the necessary parts of the actual DOM. This approach minimizes unnecessary DOM manipulation and enhances performance.

Declarative Syntax: React uses a declarative approach, where developers specify the desired outcome of UI components, and React takes care of the underlying updates and rendering. This leads to more predictable code and easier debugging.

Reconciliation: React's reconciliation algorithm efficiently updates the DOM by minimizing changes and avoiding unnecessary re-renders. This contributes to faster rendering and improved performance.

Component Lifecycle Methods: React components have lifecycle methods that allow developers to perform actions at specific stages of a component's existence, such as when it is created, updated, or removed. This enables tasks like data fetching, updating state, and cleanup operations.

State Management: React allows components to manage their internal state, making it easier to handle dynamic data and interactive behaviour.

Context API: The Context API enables data to be passed down the component tree without manually passing props at each level. This is particularly useful for managing global data and themes across an application.

React Router: React Router is a popular library for handling navigation and routing in React applications. It allows for creating single-page applications with multiple views while maintaining a seamless user experience.

Component Reusability: React components are highly reusable, which leads to efficient development and codebase maintenance. Developers can build a library of components to be used across different projects.

Transforming a Static HTML/CSS Website into a React-Powered Application

In the dynamic realm of web development, the evolution of technologies has led to new and more efficient ways to build interactive and responsive websites. Converting a website from traditional HTML/CSS to React, a powerful JavaScript library, marks a significant step towards enhancing user experiences, streamlining development, and embracing modern web practices.

Understanding the Transition:

1. **Component-Based Paradigm:** The fundamental shift when migrating to React lies in its component-based architecture. Instead of monolithic HTML files, the website is

broken down into reusable components. These components encapsulate UI elements and their behaviours, fostering modularity, reusability, and easier maintenance.

2. **Declarative Approach:** React employs a declarative approach to building interfaces. Developers describe the desired UI outcome, and React takes care of efficiently updating the DOM. This approach simplifies the process of handling dynamic content, user interactions, and state changes.
3. **Virtual DOM Optimization:** React's virtual DOM mechanism minimizes direct manipulation of the actual DOM, resulting in faster rendering and better performance. It intelligently updates only the necessary parts of the DOM, reducing overhead and enhancing the user experience.
4. **Data Flow:** React enforces a unidirectional data flow. This clear flow of data ensures predictability and makes it easier to track changes, debug issues, and maintain codebases as the application grows.

Migration Process:

1. **Component Identification:** The HTML structure is analysed to identify logical components. Elements that repeat or have shared functionality can be turned into React components.
2. **Component Creation:** Each identified component is transformed into a React component. HTML markup becomes JSX, which allows embedding JavaScript logic within the markup.
3. **State Management:** Elements that require dynamic updates are integrated with React's state management. This enables seamless handling of changes and real-time interactions.
4. **CSS Adaptation:** Existing CSS styles can be retained, but React's approach often involves component-specific styling methodologies like CSS Modules or Styled Components. This encapsulates styles and reduces the risk of class name collisions.
5. **Event Handling:** Event handling is achieved using React's event system. Inline JavaScript event attributes in HTML are replaced with event listeners attached to React components.
6. **Data Binding:** Any data binding in HTML is replaced with JSX expressions, which render dynamic content within React components.

7. **Refactoring:** Refactor the codebase to align with React's best practices, including component naming conventions, prop handling, and state management.

WEEK 4

Chapter 1

Introduction

Problem Statement: -To create Front End and Backend of Online Vegetable Market Website using MongoDB.

Chapter 2

Technologies Used

Backend (MongoDB)

1.User Authentication and Authorization: Implement user registration and login functionality to allow users to create accounts and log in. Use sessions or JSON Web Tokens (JWT) for authentication and ensure that only authenticated users can access certain features of the website, such as placing orders or managing their profiles.

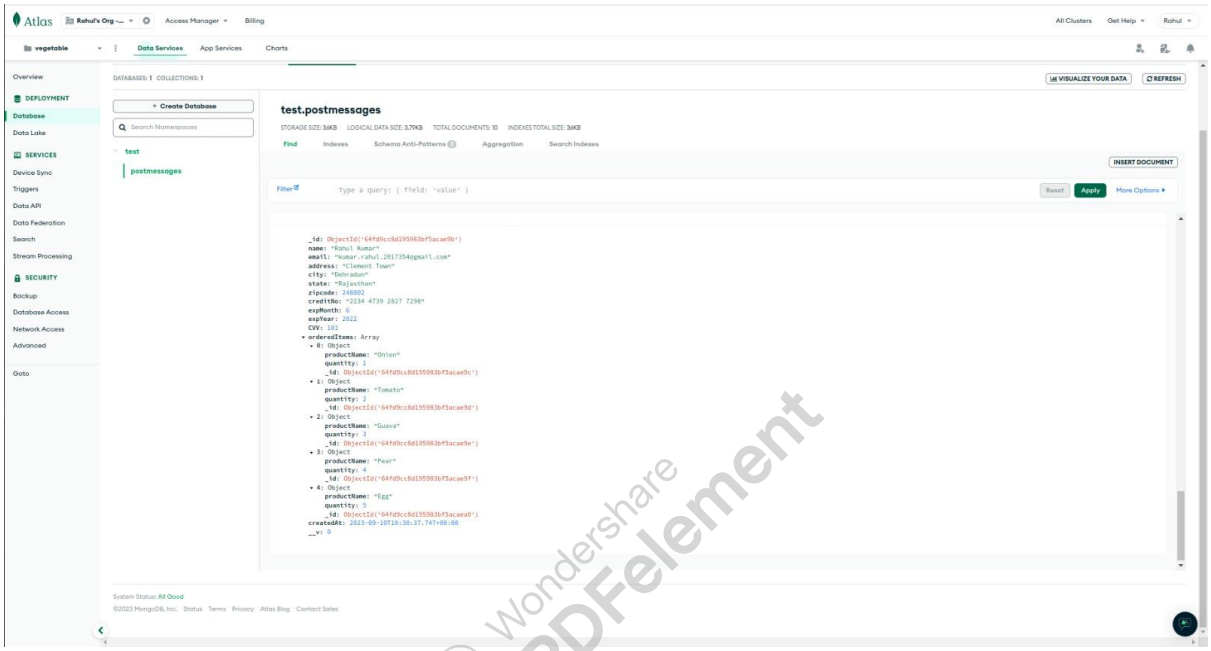
2.Product Listings: Create APIs to display a list of available vegetables with their prices, descriptions, and images. Users should be able to browse through the list and filter vegetables based on categories or search terms.

3.Shopping Cart: Implement a shopping cart functionality that allows users to add vegetables to their cart, adjust quantities, and view the total cost of their selections. The cart should persist between user sessions.

4.Order Management: Enable users to place orders for the vegetables in their cart. After placing an order, users should receive confirmation emails, and the order details should be stored in the database for future reference.

Chapter 3

Result and Discussion



Chapter 5

Backend Future Work:

1. **User Profiles:**

   Create user profiles where users can view and update their personal information, track order history, and manage delivery addresses.

2. **Wishlist:**

Give users the option to add vegetables to their Wishlist, allowing them to save products for future reference or purchases.

3. **Order Tracking:**

Provide users with the ability to track the status of their orders. You can use real-time updates or periodic notifications to keep users informed about their order progress.

WEEK 5

Chapter 1

Introduction

LSTM models for vegetable price prediction are a specialized application of deep learning in the agricultural sector. These models utilize recurrent neural networks with LSTM cells to analyze historical data, including vegetable prices, weather conditions, and other relevant factors. By capturing long-term dependencies and temporal patterns, LSTM models can forecast future vegetable prices or availability. The process involves data preparation, feature engineering, model training, validation, and testing. Once trained, the LSTM model can provide valuable insights for stakeholders in agriculture and supply chain management, aiding in decision-making regarding production, distribution, and pricing strategies. Regular model monitoring and updates are necessary to adapt to changing market conditions and ensure accurate predictions.

Problem Statement: - To detect the future prices of vegetables based on the database.

Chapter 2

Data and Methodology

Data Collection: We sourced a comprehensive dataset containing historical records of vegetable prices from various markets and regions. The dataset includes information on factors such as type of vegetable, market location, date, supply, demand, weather conditions, and more.

Methodology: The methodology for LSTM-based vegetable price prediction involves several key steps. First, historical data on vegetable prices, weather conditions, and other relevant factors are collected and cleaned. Next, feature engineering is conducted to select and create meaningful input features for the LSTM model. The LSTM neural network architecture is then designed and configured, typically with one or more layers to capture temporal dependencies in the data.

The model is trained using historical data, and its parameters are optimized to minimize prediction errors. Validation and testing are critical to assessing the model's performance using separate datasets and appropriate evaluation metrics. Once the model is trained and validated, it can be deployed for making future vegetable price predictions based on current input data.

Continuous monitoring of the model's performance is essential, as vegetable prices and market conditions can change rapidly. Regular updates or retraining may be required to ensure the model adapts to evolving trends and factors affecting vegetable markets, thus maintaining its accuracy and relevance. This methodology, rooted in deep learning and time series analysis, enables stakeholders to make informed decisions in agriculture, supply chain management, and economic planning.

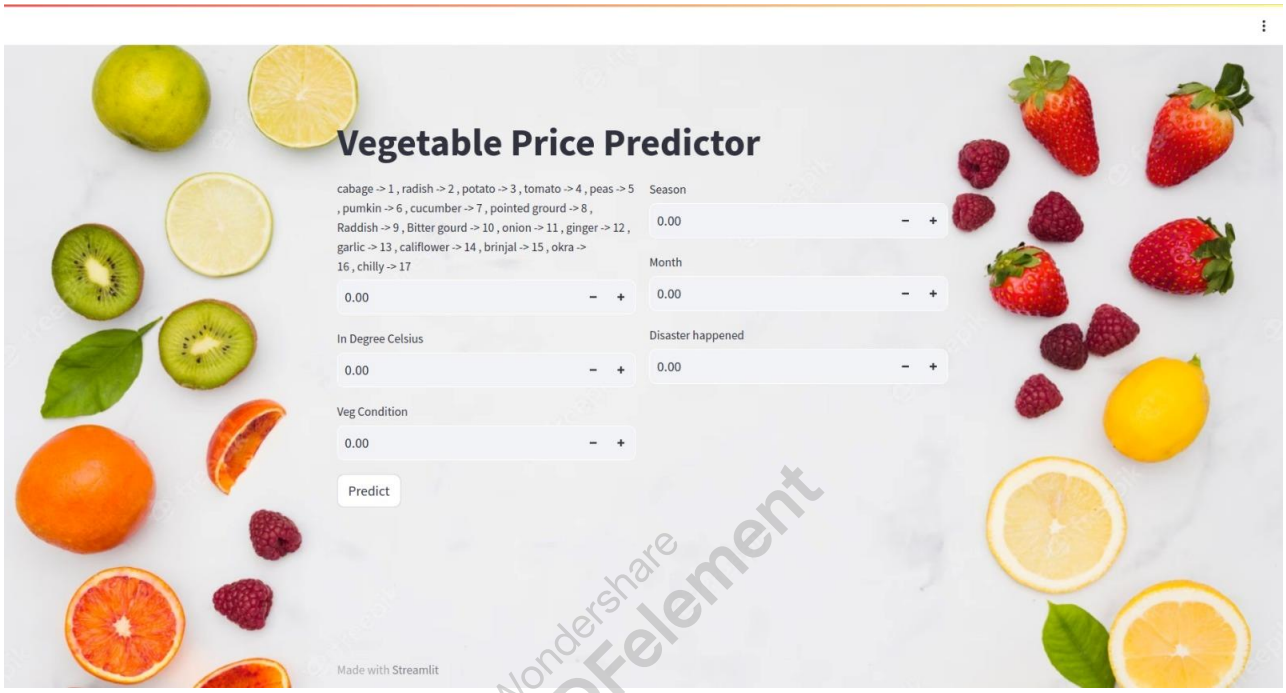
Chapter 3

Libraries Used

In LSTM-based vegetable price prediction, commonly used Python libraries include TensorFlow or PyTorch for implementing the LSTM model and scikit-learn for data preprocessing, feature engineering, and model evaluation. Additionally, libraries like pandas are used for data manipulation and numpy for numerical operations. These libraries collectively enable efficient development and deployment of LSTM models for forecasting vegetable prices.

Chapter 4

Result



Chapter 5

Future Work

In the future, advancements in LSTM-based vegetable price prediction models could encompass several key areas. One avenue involves enriching the feature set by incorporating additional variables like economic indicators, political events, and transportation data to enhance prediction accuracy. Hybrid models, combining LSTM with traditional time series forecasting techniques, might emerge to capture both linear and non-linear data patterns effectively. Interpretable models could also gain traction, enabling stakeholders to comprehend the underlying factors influencing predictions. Real-time data integration may become more prevalent, offering up-to-the-minute insights into market dynamics. Furthermore, uncertainty estimation techniques could provide confidence intervals around predictions, ensuring users gauge prediction reliability. Other areas of development include user-friendly interfaces, global market integration, environmental considerations, ethical assessments, supply chain integration, blockchain technology utilization, and longer forecasting horizons, all geared toward delivering more accurate, adaptable, and insightful vegetable price predictions to various industries and stakeholders.

WEEK 6

Chapter 1

Introduction

Netlify is a cloud-based platform and service that is primarily used for web development and hosting. It offers a variety of features and tools designed to simplify the process of building, deploying, and managing websites and web applications. Here are some key aspects of Netlify:

1. **Web Hosting**: Netlify provides hosting for websites and web applications. Users can deploy their sites to Netlify's infrastructure, which includes a global content delivery network (CDN) to ensure fast and reliable content delivery to users around the world.
2. **Continuous Integration and Deployment (CI/CD)**: Netlify integrates with version control systems like Git and offers seamless CI/CD pipelines. This means that whenever you push changes to your code repository, Netlify can automatically build and deploy your site, ensuring that the latest version is always live.
3. **Serverless Functions**: Netlify supports serverless computing, allowing developers to run serverless functions as part of their applications. This enables the creation of dynamic, serverless APIs and other backend functionality without managing traditional server infrastructure.
4. **Edge Logic**: Netlify provides tools for implementing edge logic, allowing developers to run code at the edge of the network (i.e., close to the end-users). This can be used for tasks like A/B testing, personalization, and security.
5. **Global Scalability**: With its CDN and global network of servers, Netlify can serve websites and applications quickly to users all over the world, improving performance and reducing latency.
6. **SSL/TLS Certificates**: Netlify offers free SSL/TLS certificates, making it easy to secure your website with HTTPS.
7. **Branch Deployments**: Developers can create and deploy multiple branches of their site for testing and preview purposes. This is useful for collaboration and testing new features without affecting the production site.
8. **Forms Handling**: Netlify includes form handling capabilities, making it easy to capture and process form submissions on your website.
9. **Analytics and Monitoring**: Netlify provides analytics and monitoring tools to track website performance, visitor statistics, and other relevant data.

10. **Git Integration**: Netlify is closely integrated with Git repositories (GitHub, GitLab, Bitbucket), making it easy to set up automated deployments and workflows.

Chapter 2

Technologies Used in Netlify

Here are some of the key technologies used in Netlify:

1. **Content Delivery Network (CDN)**: Netlify relies on a global CDN to distribute website content to users worldwide. CDNs use multiple data centers strategically located around the world to reduce latency and ensure fast content delivery.
2. **Git Integration**: Netlify integrates with Git version control systems, such as GitHub, GitLab, and Bitbucket, to automatically build and deploy websites and applications when changes are pushed to a repository.
3. **Continuous Integration and Continuous Deployment (CI/CD)**: Netlify employs CI/CD pipelines to automate the building and deployment of websites. It uses build tools and scripts to transform source code into a production-ready site.
4. **Serverless Functions**: Netlify allows developers to create and deploy serverless functions, often using technologies like AWS Lambda or Azure Functions. These functions can be used for dynamic server-side operations and API endpoints.
5. **Static Site Generators**: Netlify works well with static site generators like Hugo, Jekyll, Gatsby, and others. These generators help create static HTML files from source code, which can be efficiently served and cached.
6. **JavaScript Frameworks and Libraries**: Netlify supports various JavaScript frameworks and libraries for building dynamic web applications. Common ones include React, Vue.js, Angular, and Svelte.
7. **Edge Logic**: Netlify offers tools for implementing edge logic, which is executed at the edge of the network (close to the end-users). This can be done using serverless functions or Netlify Edge Handlers, which allow developers to run custom JavaScript code at the CDN edge nodes.
8. **SSL/TLS Encryption**: Netlify provides SSL/TLS certificates for securing websites with HTTPS. It uses technologies like Let's Encrypt to automate certificate provisioning and renewal.

9. **Authentication and Identity Services**: Netlify offers authentication and identity services, often using JSON Web Tokens (JWT) and OAuth 2.0, to handle user authentication and authorization.
10. **Monitoring and Analytics**: Netlify integrates with various monitoring and analytics services to provide insights into website performance, visitor behavior, and error tracking.
11. **Containerization**: Netlify can containerize applications using Docker, enabling more complex application deployment scenarios.
12. **Form Handling**: Netlify includes form handling capabilities, often using serverless functions to process form submissions and send notifications.
13. **DNS Management**: Netlify offers DNS management services, allowing users to configure custom domains and handle DNS records.
14. **Markdown and Content Management**: Netlify CMS, a content management system developed by Netlify, uses Markdown for content authoring and provides an intuitive interface for managing content.
15. **Version Control**: Netlify relies on version control systems (primarily Git) to manage code changes and trigger automated deployments.

Chapter 3

Methodology

Prerequisites:

1. **Website Code**: Make sure you have your website's code ready. This could be a static website generated using tools like Hugo, Jekyll, or a JavaScript framework like React.
2. **Git Repository**: If your website's code is not already on a Git repository (e.g., GitHub, GitLab, Bitbucket), create one and push your code to it. Netlify will use this repository to build and deploy your website.
3. **Netlify Account**: Sign up for a Netlify account if you don't have one. It's free to get started, and they offer various pricing plans depending on your needs.
4.

Methodology:

1. **Login to Netlify**: Go to the Netlify website and log in to your Netlify account.
2. **Create a New Site**:
    - Once logged in, click the "New site from Git" button on the Netlify dashboard.

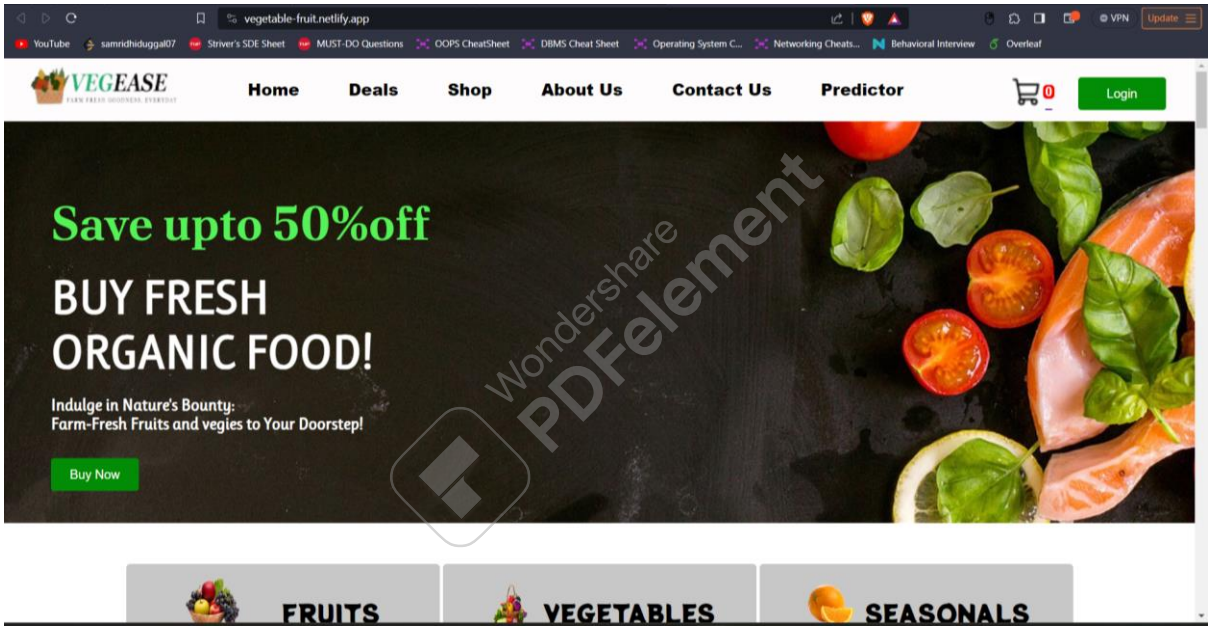
3. **Connect to Git Repository**:
    - Select your Git hosting service (GitHub, GitLab, Bitbucket, or others).
    - Authorize Netlify to access your repository.
    - Choose the repository where your website code is hosted.
4. **Configure Build Settings**:
    - Specify the branch you want to deploy (typically the main branch, like **master** or **main**).
    - Set the build command. For most static site generators, this would be something like **npm run build**, **yarn build**, or a command specific to your generator.
    - Specify the publish directory. This is the directory where your built website files are located after the build process. Common choices include **public**, **dist**, or **build**.
5. **Advanced Build Settings** (Optional):
    - You can configure environment variables, cache settings, and other advanced build options if needed.
6. **Deploy Site**:
    - Click the "Deploy site" button to start the deployment process. Netlify will fetch your code from the Git repository, build your website, and deploy it automatically.
7. **Domain and DNS Configuration**:
    - Netlify assigns a temporary domain to your site (e.g., **yoursite-name.netlify.app**). You can use this domain immediately.
    - To use your custom domain, go to your domain registrar's website (e.g., GoDaddy, Namecheap) and update your DNS settings to point to Netlify's servers. Netlify provides detailed instructions for this step.
8. **HTTPS Configuration**:
    - Netlify provides free SSL/TLS certificates. Once your DNS settings are updated and propagated, Netlify will automatically configure HTTPS for your custom domain.
9. **Testing**:
    - Test your website using the provided Netlify domain or your custom domain if DNS changes have propagated.

10. **Continuous Deployment**:
    - With Netlify, whenever you push changes to your Git repository, Netlify will automatically trigger a new build and deploy your updated website.

Chapter 4

Result

Link to the website: - https://vegetable-fruit.netlify.app/



REFERENCES

- ✓ https://react.dev/
- ✓ Introduction to react by Cory Gackenheimer
- ✓ https://www.mongodb.com/
- ✓ Data Migration from Relational Database to MongoDB By Ajit Singh
- ✓ https://www.w3schools.com/
- ✓ https://www.geeksforgeeks.org/web-development/
- ✓ Web Development with MongoDB and NodeJS by Mithun Satheesh, Bruno Joseph D'mello, Jason Krol
- ✓ User-centered Web Development By Jonathan Lazar
- ✓ https://www.netlify.com/
- ✓ Karimi, Amir-Hossein, Gilles Barthe, Borja Balle and Isabel Valera. “Model-agnostic counterfactual explanations for consequential decisions.” AISTATS (2020). (For machine learning)