"A case Study of High-Fidelity Web Front end Development for Perishable Goods "

by

Examination Roll: 241107
Name: Nadira Khan Tabia

A Project Report submitted to the
Institute of Information Technology
in partial fulfillment of the requirements for the degree of
Professional Masters in Information Technology

Supervisor: Professor Shamim Al Mamun

Institute of Information Technology
Jahangirnagar University
Savar, Dhaka-1342

Septemeber 2025

DECLARATION

I hereby declare that this thesis is based on the results found by myself. Materials of work found by other researcher are mentioned by reference. This thesis, neither in whole nor in part, has been previously submitted for any degree.

Name: Nadira Khan Tabia
Roll: 241107

CERTIFICATE

The project titled "A case Study of High-Fidelity Web Front end Development for Perishable Goods" submitted by Nadira Khan Tabia, ID: 241107, Batch: 31, has been accepted as satisfactory in partial fulfillment of the requirement for the degree of Professional Masters in Information Technology on the 20th September 2025.

Dr. Shamim Al Mamun
Supervisor

BOARD OF EXAMINERS

Dr. M Shamim Kaiser
Professor, IIT, JU

Coordinator
PMIT Coordination Committee

Dr. Risala Tasin khan
Professor, IIT, JU

Member, PMIT Coordination Committee
& Director, IIT

Dr. Jesmin Akter
Associate Professor, IIT, JU

Member
PMIT Coordination Committee

K M Akkas Ali
Associate Professor, IIT, JU

Member
PMIT Coordination Committee

Rashed Mazumber
Associate Professor, IIT, JU

Member
PMIT Coordination Committee

ACKNOWLEDGEMENTS

I feel pleased to have the opportunity of expressing me heartfelt thanks and gratitude to those who all rendered their cooperation in making this report.

This thesis is performed under the supervision of Professor Shamim Al Mamun, PhD, Jahangirnagar University, Savar, Dhaka. During the work, he has supplied us a number of books, journals, and materials related to the present investigation. Without his help, kind support and generous time spans he has given, we could not perform the project work successfully in due time. First and foremost, we wish to acknowledge our profound and sincere gratitude to him for his guidance, valuable suggestions, encouragement and cordial cooperation.

Moreover, I would also like to thank the other faculty members of IIT who have helped me directly or indirectly by providing their valuable support in completing this work.

I express our gratitude to all other sources from where I have found help. I am indebted to those who have helped me directly or indirectly in completing this work.

Last but not least, I would like to thank all the staff of IIT, Jahangirnagar University and my friends who have helped me by giving their encouragement and cooperation throughout the work.

ABSTRACT

With the digital revolution in retail, e-commerce platforms are now indispensable for traditional industries like fresh produce and food items. This thesis examines the creation, development, and analysis of a comprehensive frontend system for "Tabia's Vegetables," aimed at providing dietary products through an online shopping platform. The project is focused on creating a user-friendly, engaging, and visually pleasing online shopping experience that can rival the capabilities of major online retailers while also catering to perishable items.

The web development was based on a set of core technologies: HTML5 for structure, CSS3 for presentation and animation, and JavaScript for dynamic interactivity and data management. Among its features are a responsive navigation system with glassmorphism design, an integrated light/dark mode with dynamic theme toggle for user authentication and shopping cart functionality, and downloadable product pages. A significant contribution is the development of a structured JavaScript object that contains over 40 vegetable products, each with detailed metadata such as pricing, stock levels, and nutritional information, to create scalable data models for specialized platforms like produce.shtml or vice versa.

The user interface of the website utilizes sophisticated CSS techniques such as gradient animations, particle backgrounds, and smooth transitions to enhance its overall aesthetic appeal. The system is also subjected to rigorous testing to ensure it is responsive across mobile, tablet, and desktop viewports. Based on the findings of this thesis, the strategic use of modern frontend technologies can lead to an effective approach to creating a high-fidelity, interactive and flexible prototype for e-commerce. By providing solid groundwork for future work, including backend integration, performance enhancement, and formal user experience testing, the project provides a strong starting point. Github Link -https://github.com/nadirakhantabia11/TabiasVegetables

Keywords: E-commerce, Frontend Development, HTML5, CSS3, JavaScript, Responsive Design, User Interface, Data Management, Online Shopping Platform

LIST OF ABBREVIATIONS

IIT	Institute of Information Technology
JU	Jahangirnagar University

LIST OF FIGURES

Figure

LIST OF TABLES

Table

TABLE OF CONTENTS

CHAPTER I

Introduction

1.1 Background and Motivation

In the last decade, e-commerce business has become familiar to the common people of Bangladesh due to the convenience of mobile computers and internet technology. Although the trade business was initially established in some technology products, it is now feeling the need to become an ideal medium for some new products. Especially, its need has increased due to pandemics like Covid-19.

Even so, a significant gap remains in this expanding market.' Small to medium-sized enterprises or local farms often operate online for fresh produce whose websites prioritize simplicity and convenience over user experience[1]. This is particularly true of many existing platforms. Old, generic templates are frequently employed, which are functionally sound but visually unappealing. User interfaces that lack edgy or engaging, do not accurately convey the vibrant and high-quality of unprocessedited produce, and frequently fall short of the modern amenities that consumers are seeking[2]. The outcome is a digital experience that feels like showcasing ersatz goods, lacking the trust and enthusiasm to sustain brand loyalty in heightened competition[3].

The project is driven by the desire to address this gap. It is based on the idea that success for an online fresh produce store is not about physical quality, but about digital. That's why "Tabia's Vegetables" was created. What does it mean? The project seeks to illustrate that by utilizing modern frontend technologies and User Interface/User Experience (UI/UDX) design principles, one can achieve a highly functional storefront that is both visually appealing and enjoyable to navigate.

Beyond aesthetics, the driving force is evident. Customer feedback on products, including nutritional information and product descriptions, is aimed at creating a sense of trust and transparency[4]. It's about building an open, responsive platform that works across a wide range of devices (from desktops to smartphones)[5]. At

its core, the objective of this endeavor is to transform from a daily routine task to transforming fresh produce shopping online into an enjoyable and user-friendly experience, setting 'new standards for niche e-commerce in the agricultural trade.

1.2 Problem Statement

However, not all sectors have profited equally from the spread of 'e-commerce' [6]. Old or generic online templates are commonly used by new produce ventures of small to medium size [7]. This poses a critical triple-threat problem: inadequate user engagement through modern aesthetics [2], poor user experience through lack of interactivity [8], and the absence of scalable frontend data model hindering effective product presentation and management [9]. Hence, these websites are in competition with each other, as they cannot translate the quality of their physical products into a compelling digital presence that builds customer loyalty and facilitates streamlined shopping experiences [3].

1.3 Research Objectives

A set of core research objectives is utilized to conduct a systematic investigation of the problems identified in this thesis. Specifically, the aim is to create and test a high-fidelity frontend prototype for an exclusive e-commerce website called "Tabia's Vegetable. "This overall objective is separated into three main objectives:

- To design and build fully responsive website structure using semantic HTML5, while ensuring basic SEO principles for SEO-friendly accessibility. An intuitive information architecture was established, providing clear user paths to browse and manage accounts, as well as initiate checkout.

- To utilize modern CSS3 techniques such as Flexbox, Grid, CSS custom properties, and complex keyframe animations to create a visually appealing identity. This objective aimed to achieve user engagement through the use of a glassmorphism design aesthetic, incorporating 'dynamic light/dark mode toggle', and engaging micro-interactions.

- A dynamic, robust JavaScript data model that can scale and represent a complex list of perishable commodities. Dynamic content rendering and filtering are made possible by managing multiple product variations, real-time stock levels with pricing, and detailed nutritional metadata in this model.

- To develop the client-side logic for essential e-commerce features, such as a persistent shopping cart using localStorage, user authentication flows via modal windows, and redesigned user interfaces that respond dynamically to user actions without page realoading.

- To test and evaluate the final system's functionality, cross-browser compatibility, and device responsiveness, ensuring that the prototype is both robust and user-friendly.

1.4 Scope and Limitations

The focus of this research is intentionally and explicitly on the frontend ecosystem of a web application [10]. This entails the complete creation and execution of the user interface (UI), along with development of all client-side interactivity and application logic, and production for an internal data source that is structured but static. The functionalities, such as simulated user authentication flows, shopping cart operations, dynamic product catalog display, and theme switching, are all engineered to function exclusively within the client's browser environment using HTML, CSS, and vanilla JavaScript [11]. Through this method, modern frontend techniques, responsive design patterns, and client-side state management can be thoroughly explored [12], yielding a high-fidelity prototype that showcases an entire user journey from landing page to checkout initiation.

Its main drawback is that this approach is exclusively designed for frontend development [13]. Most notably, there is no inbuilt persistent backend server with any database associated with it. The architecture's decision to limit data management to the browser's local storage eliminates real-time, multi–user capabilities and dynamic data persistence beyond one-user sessions [13]. Therefore, many of the essential components of a full-stack e-commerce platform are not within the current range. The system comprises of a secure payment processing gateway (e.g, Stripe, PayPal), supplementary content management system (CMS) for administrative updates, and robust server-side user authentication with encrypted credential storage [14]. However, the current authentication is a "simulated frontend process" and does not have production-level security robustness [14]. In addition, the product catalog is comprehensive and well-organized, but it's coded into an application. Direct developer involvement and code changes are mandatory for any modifications made to product information, pricing, or inventory levels, preventing non-technical administrative management [15]. These limitations define the project as a prototype and

proof-of concept experience for the frontend experience, providing 'proof' of future integration and full-stack development [16].

In Scope	**Limitations / Out of Scope**
User Interface (UI) Design	Persistent Backend Server
Client-Side Logic and Interactivity	Database Integration
Static Data Source (`vegetableData`)	Payment Gateway Integration (e.g., Stripe, PayPal)
Responsive Layouts	Content Management System (CMS)
Client-Side State Management (e.g., Cart)	Secure Server-Side User Authentication
Frontend-Only User Authentication Simulation	Automated Inventory Management

Table 1.1: Project Scope and Key Limitations

CHAPTER II

Literature Review and Theoretical Framework

2.1 The Evolution of E-Commerce UX

E-commerce has experienced a significant shift in user experience (UX), transitioning from simple online catalogs to elaborate, interactive environments [17]. There are three main phases of this evolution, building upon the previous one to meet evolving user demands.

During the Utility Phase, websites that were designed for transactions were functional [18]. Its original intention was to establish an internet-based presence, presenting product listings and a straightforward checkout process. The emphasis on UX often resulted in clunky navigation, lacking confidence indicators and poor visual design, instead of being prioritized for the ultimate goal of a sale [19].

The Usability Phase was initiated due to the rise of competition and the emphasis on simplifying the customer journey [20]. In this era, checkout procedures were simplified, security badges and return policies were introduced to create an intuitive information architecture that was the foundation of trust among consumers [21]. Amazon's best practices have revolutionized the web, making it easier and safer to navigate sites [22].

Our current phase of engagement is the Engagement Phase, where UX is a significant factor in competitive differentiation [23]. Modern platforms generate emotional experiences that are reminiscent of apps, thanks to rich media usage, personalized recommendations, micro-interactions, and smooth animations [24]. It's not just about making a sale; it also has to do with brand loyalty, by decreasing friction and creating an enjoyable, memorable user journey from discovery to purchase [25].

Below is a map that shows how the focus on this shifts from core utility to more of 'holistic engagement' [26].

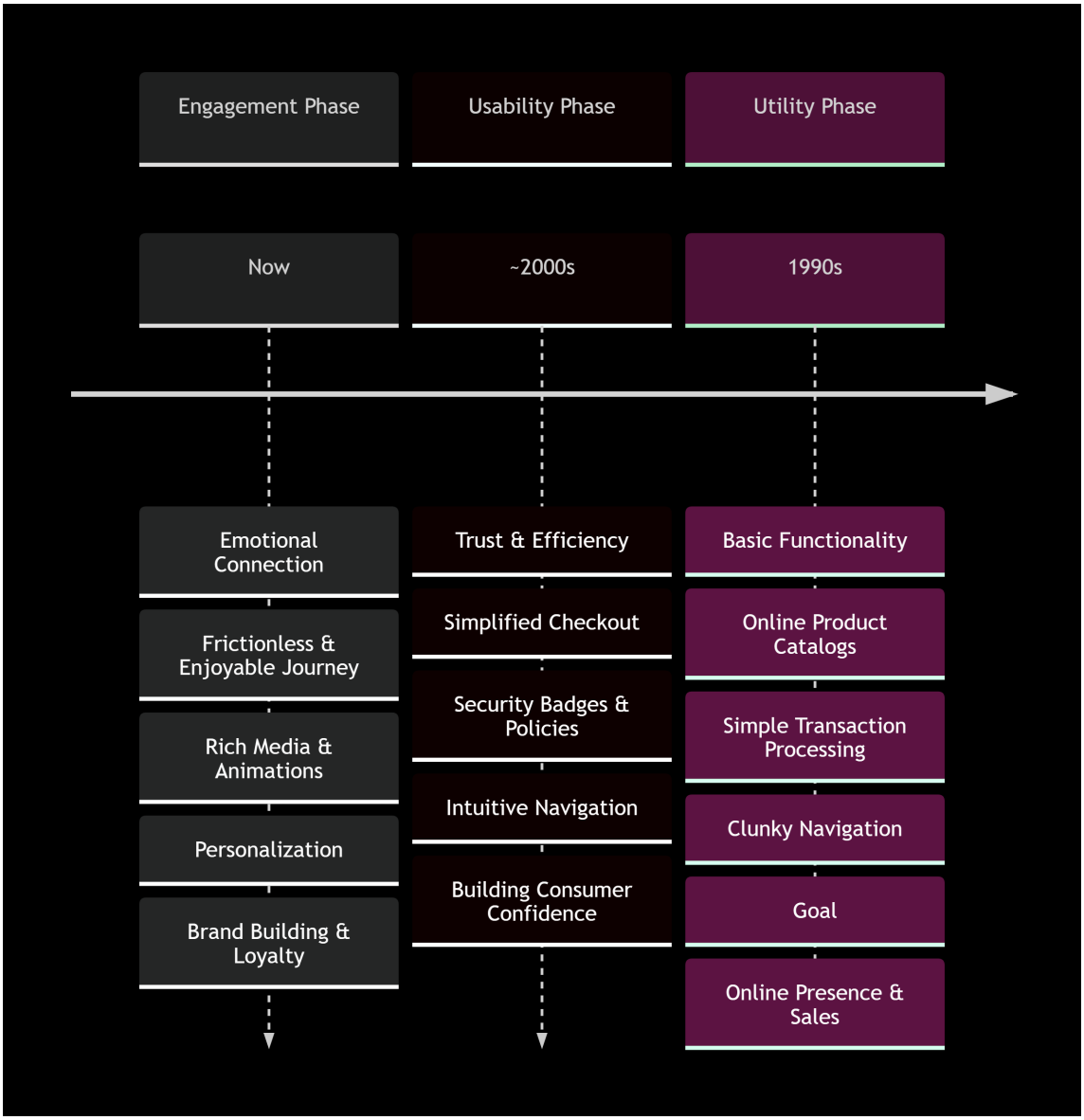

Figure 2.1: The Evolution of E-Commerce UX

2.2 Modern Web Design Principles

The design of "Tabia's Vegetables" adheres to modern web design standards, emphasizing user-centeredness, aesthetics, and functionality [8]. The principle of minimalism is based on the idea that unnecessary elements should be eliminated in order to reduce cognitive load and redirect users' attention to the necessary content and actions [27]. Clean lines, sufficient white space, and a product-centric presentation are clear indications of this.

A closely related concept is visual hierarchy, which uses shape, color, contrast, and spacing to intentionally guide the user's eye on a deliberate journey [28]. Intuitive navigation, logically organized product categories, and clear calls-to-action create an uncluttered interface that facilitates purposeful interaction [19].

The project incorporates glassmorphism, a contemporary design trend that involves using layered materials against contrasting backgrounds to create depth and texture [29]. Using navigation bars and modals, this design aesthetic elevates the interface to a modern, sleek, and high-quality standard, which perfectly aligns with the brand's goal of selling only top-notch, fresh products [24].

The implementation follows the principle of consistency in color scheme, typography, and button style across all pages and elements [30]. This provides a more uniform, easily manipulated experience, making it easier to learn and use. The accessibility of the website is ensured so that it meets the requirements of color contrast and semantic HTML [31]. When the structure is combined, practicality is combined with aesthetic appeal [23].

2.3 The Role of Animation in User Engagement

Animation has become a crucial element of modern web design, replacing its traditional role as 'decorative' to enhance user engagement, communication, and usability [32]. When used purposefully, animation provides essential functional feedback and affordance by guiding users through interactions and verifying their actions. As an illustration, a button that responds by depressing slightly upon clicking provides immediate feedback to the user, decreasing their level of uncertainty [8].

Beyond this, animation is a form of feedback that helps to establish visual hierarchy and guides attention. By using subtle gestures like the pulsing effect on a "Shop Now" button or the smooth transitions made as yo-y movement across overlapping product cards, the user can easily navigate around important elements of the interface and hear their story, without any visual cues. The interface becomes more responsive and intuitive, resulting in reduced cognitive load [28].

Moreover, animation is instrumental in shaping the brand's identity and emotional connections. The animations in "Tabia'S Vegetables", which include a gradient pulse in the main text, floating particles, and smooth sliding of modals, are carefully designed to give the impression of something fresh, high quality, dynamically active. These carefully curated gestures give the user experience a polished and premium feel, turning it from merely interacting with objects into an engaging interaction

[24]. A sense of aliveness and responsive nature are the hallmarks of these micro-interaction, which foster a positive emotional experience that can be used to build user satisfaction and brand loyalty over time [33].

2.4 Frontend Architectures for Dynamic Content

Strong architectural patterns have been developed to manage the complexity of frontend development, which has shifted from static websites to dynamic experiences on the web [18]. Although JavaScript frameworks like React, Angular, and Vue.js are the norm for today's big applications in general [34], this project uses pure Javascript components to build a component-based architecture that demonstrates the fundamental principles that support these advanced tools [35].

Its architecture is loosely based on the MVC pattern, which is a widely recognized software design pattern [36]. This implementation uses the vegetableData object as a representation of the Model, which is the only reliable source of information for all products [37]. It is a View that uses HTML and CSS to style, render, and style. JavaScript-based Controller logic is used to manage user input, such as clicking "Add to Cart," updating the Model, and refreshing the View accordingly [38]. Additionally, there are several functions associated with this method.

This approach allows for a differentiation between data management, presentation and application logic [38]. It enhances the organization, consistency, and adaptability of code [39]. For projects of this nature, using vanilla JavaScript avoids the overhead and learning curve associated with such a framework; it still achieves three main goals: dynamic content rendering without the need for page reloading (statically loaded HTML code), 'permanent client-side data state' through localStorage ("all available data stream") and UI responsiveness/interactive [40]. This serves as a clear demonstration of the fundamental notions that frameworks abstract and regulate [9].

CHAPTER III

System Design and Methodology

3.1 Requirement Analysis

By conducting a thorough requirement analysis, the "Tabia's Vegetables" project was confident that it could meet essential internet shopping requirements while also providing an excellent user experience [41]. The process established Functional Requirements (FR) and Non-Functional Requirements (NFR).

The Functional Requirements were based on common user journeys. Modal windows with FR1 user authentication were available, enabling users to register and log in without the need to reload pages. In FR2, the dynamic display of products in catalogs is achieved by rendering items from an organized JavaScript data object, including filtering and "Out of Stock" states. FR3 offers comprehensive shopping cart management, including the ability to add items and a state-saving counter persistently updated with localStorage. FR4 is a theme switching mechanism allowing the client to switch between light and dark modes, altering the color palette of the entire interface.

The NFR1 criteria emphasizes quality attributes, meaning the website must load quickly and smoothly while maintaining fluid animations [8]. According to NFR2, the interface must be consistent across major browsers like Chrome, Firefox or Safari for compatibility. NFR3 (Responsiveness) mandates flawless adjustment of layout and functionality to accommodate mobile, tablet, and desktop screen sizes [5]. The following design and development choices were subject to these requirements.

3.2 Architectural Design: The MVC Paradigm

The frontend of "Tabia's Vegetables" is built upon an architectural framework utilizing the Model-View-Controller (MVC) pattern [36]. The primary objective was

to ensure a clear separation of concerns, leading to improved code maintainability, scalability and organizational clarity, even within a purely client-side implementation.

The Model component is the data layer. In this project, the model is represented by a structured data object containing all product information, including item names, prices, descriptive text, image paths, stock quantities, and nutritional information. This centralized data store serves as the single source of truth for the application's dynamic content.

View is the presentation layer. The website's semantic framework is established by HTML structure, while visual styling and layout are controlled by CSS. The view's purpose is to render the user interface and present the current state of the data model to the user.

The Controller component acts as an intermediate logic layer. JavaScript functions manage all user input events, including mouse clicks on interface elements and form submissions. After executing business logic (e.g., adding items to a shopping cart or toggling visual themes), the controller updates the model based on these interactions. Finally, it refreshes the view to reflect changes in the model's state, ensuring the user interface remains synchronized with the underlying data.

3.3 UI/UX Design Strategy

"Tabia's Vegetables" was designed with a mobile-first approach that prioritized small screen to desktop functionality. The primary goal was to produce a user interface that feels fresh and trustworthy, reflecting the quality of the product.' A minimalist, clean layout was used to reduce cognitive load, with a visual hierarchy that guides users intuitively from discovery to checkout. Natural greens and earth tones are incorporated in the brand's color palette, strengthening its identity. Clear visual elements are used to create the glassmorphism navigation bar and the prominent shopping cart, which are key interactive components. Rather than being the focus of an app, they opt for subtle animations and feedback micro-interactions to enhance engagement and create a responsive, app-like experience that users can feel comfortable using.

3.4 Data Model Design

Vehemently designed as an adaptable, hierarchical data model, the vegetableData object serves as the application's central repository of information [37]. This object

contains a self-contained product with standardized data for efficient programmatic manipulation and retrieval, with key-value pairs that are independently normalized [9]. The design is essential for enabling dynamic frontend features. The fundamental e-commerce functionality relies on defined attributes such as unique identification code, product designation and label, cost estimation (price function), image reference, and current stock level [36]. The utilization of a category field and incorporated components like nested nutrition objects facilitates more complex operations such as categorical filtering, sorting or feeding detailed nutritional data [38].

Key	Data Type	Description	Example
id	number	Unique product identifier	10
name	string	Product name	Organic Tomato
price	number	Price per Kg (in currency Taka)	50
image	string	Path to product image file	tomato.jpg
stock	number	Available quantity in kilograms	15
category	string	Product category	fruit-vegetables
nutrition	object	Nested nutritional information data	{calories: 18, ...}

Table 3.1: Product Data Model

3.5 Technology Stack Justification

The decision to use specific technologies for creating "Tabia's Vegetables" was a strategic one, considering their performance, usability, and pedagogical value [40]. Certain project tasks and learning goals prompted the decision to use core web technologies instead of larger frameworks [42].

Version five of the Hypertext Markup Language was chosen as the foundational markup language for constructing the application [43]. Its modern semantic components provide document outlines that enhance accessibility for users using assistive technologies and improve content discovery by search engines [31]. This native form validation feature reduces the initial development time and client-side processing overhead required for basic input checking [44].

The implementation of Cascading Style Sheets level three was necessary to achieve full responsiveness and implement the sophisticated visual design [5]. With the help of its layout modules, it can generate intricate and adaptable interface structures that are suitable for different screen dimensions [45]. This feature, along with the variable attributes, enables more efficient use of global design changes due to central control of stylistic properties [46]. Its animation features offer smooth transitions and visual

feedback that enhance the user experience without sacrificing speed [12].

Standard JavaScript was chosen to preserve a lightweight codebase and demonstrate comprehensibility regarding key concepts of web programming, rather than using any external frameworks [35]. The use of this method results in faster loading times, as it eliminates the overhead of parsing and executing frameworks [40]. Object-oriented abstractions in higher-level frameworks provide clear insight into how user interactions affect document structure and application state [47]. The method allows for broad compatibility across browsers and provides a useful knowledge base for understanding the internal operations of more complex tools [42].

CHAPTER IV

Implementation

4.1 Structural Implementation with HTML5

The project's structural implementation was based on the latest hypertext markup language specification, which provided a semantically accessible and relevant framework for the application. The use of semantic concepts in the development resulted in clear definitions of different content sections, which improved document outline and made it more accessible to assistive technologies. The navigation interface was designed with suitable sectioning elements, including brand recognition and primary navigation links.

Native button elements were integrated into the interface to ensure proper keyboard accessibility and screen reader compatibility, resulting in interactive components throughout. The authentication system was based on form containers with inbuilt typefaces for email addresses and required fields, which were validated internally. By reducing the use of custom validation scripts, this method improved user experience through the implementation of error messaging using a browser rather than an external one.

The content was structured in a way that separated primary content, additional information, and metadata into distinct categories. In addition to simplifying search engine optimization, this semantic structure provides clear content relationships that facilitate maintainability and styling.

A SDN based IoT infrastructure basically provides free-flow of data from sensors and wireless devices and the efficiency of the network depends on the management and security of traffic. Network traffics are dynamic and hence its more prone to malicious attacks such as DDOS, MITM, Replay, Side Channel etc.

4.2 Styling and Animations with CSS3

The use of advanced style sheet elements was utilized to create a dynamic and consistent user interface, which was then integrated into the visual presentation layer. To ensure a consistent design system across the application, custom properties were added to the document root level, which allowed for efficient global theme management. The light and dark mode functionality could be easily implemented through simple class manipulation, thanks to this approach. The use of keyframe definitions in intricate animation sequence design entailed the creation of visually engaging visual effects, such as a glowing branding element, an animated navigation indicator, and subtle floating background elements. These animations have been crafted with the intention of increasing user engagement while maintaining their performance and accessibility. The interface was designed to be visually appealing and functionally clear, while also incorporating systematic variable usage along with purposeful motion design.

4.3 Dynamic Interactivity with JavaScript

By utilizing client-side scripting, the application allowed for user interaction while also handling data and interface updates. To manage user interactions such as navigational elements, form submissions, and interactive controls, event handling mechanisms were developed to address their actions across the interface. Keeping track of product information and shopping cart operations is possible for the data management system, which also employs locally stored features to maintain persistence across browser sessions. The dynamic rendering of user interface updates is based on state changes in the application, guaranteeing that the visual presentation always represents current data conditions.

Its theme preference management system aimed to enable users to switch between visual modes while maintaining user preferences in the background. Modal interface elements were designed to facilitate user authentication flows, which included form validation and transition animations. By updating its stock in real-time, the shopping cart system provides instant visual feedback to indicate the addition or removal of products. The inclusion of keyboard shortcuts and screen reader support in the design process ensured that all interactive elements were accessible. A comprehensive interactivity strategy produces a responsive application-like experience in the browser, while maintaining cross-browser compatibility and performance efficiency.

4.4 Responsive Design Implementation

This was implemented through the use of media queries to dynamically adjust both the layout and functionality for different screen dimensions, enabling responsive design. The use of conditional style rules facilitated the redefinition of structural arrangements, spatial relationships, and interactive elements based on viewport characteristics. By utilizing a comprehensive horizontal navigation system for smaller screen devices, the user can quickly navigate to an optimized vertical navigation panel by using arrow keys or buttons. While still allowing full navigation features, the transformation effectively maximizes the display area.

4.5 Integration of Components

This, combined with the three core web technologies, produced a very active user experience. The markup language was responsible for establishing the fundamental design and semantic classification of the application's content.eath. The styling language transformed this structure into a visually engaging interface through the use of advanced layout systems, color schemes, and animated transitions. Event-driven interactivity, state management of application applications, and coordinated dynamic updates were implemented by the scripting language to unify these layers. This was achieved through a combinational logic. Interoperability facilitated real-time changes to visual presentation and data representation, maintaining interface consistency throughout user interactions.

4.6 Screenshots of website

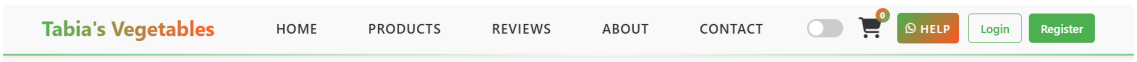

Figure 4.1: Navigation Bar

The image above is the navigation bar of our website. Our website name is on the far left of the image. It has been given a very nice gradient color. In the middle of our website, there are various buttons like Home Product Section, Review Section, About Section and Contact Section. On the right, our website theme button, cart-icon, help-button, and login and registration buttons have been placed. Clicking on each of these buttons will take you to that location. On the other hand, clicking on the theme button will change the theme.

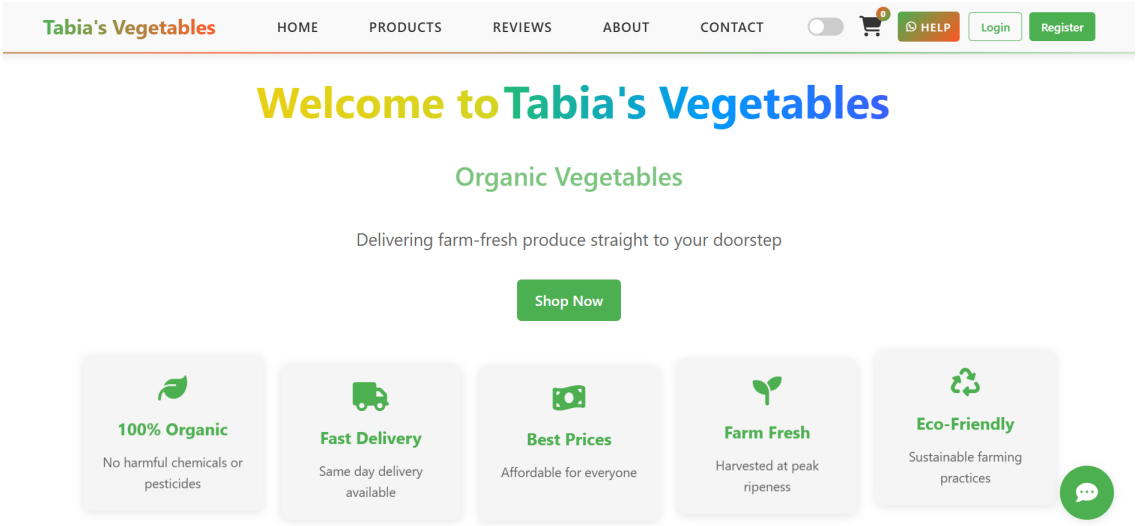

Figure 4.2: Home Section

The image above shows the home section of our website where the customer is first welcomed to our website and given various important topics on our website. There is also a "Shop Now" button where on clicking it the users will be taken to the product section.

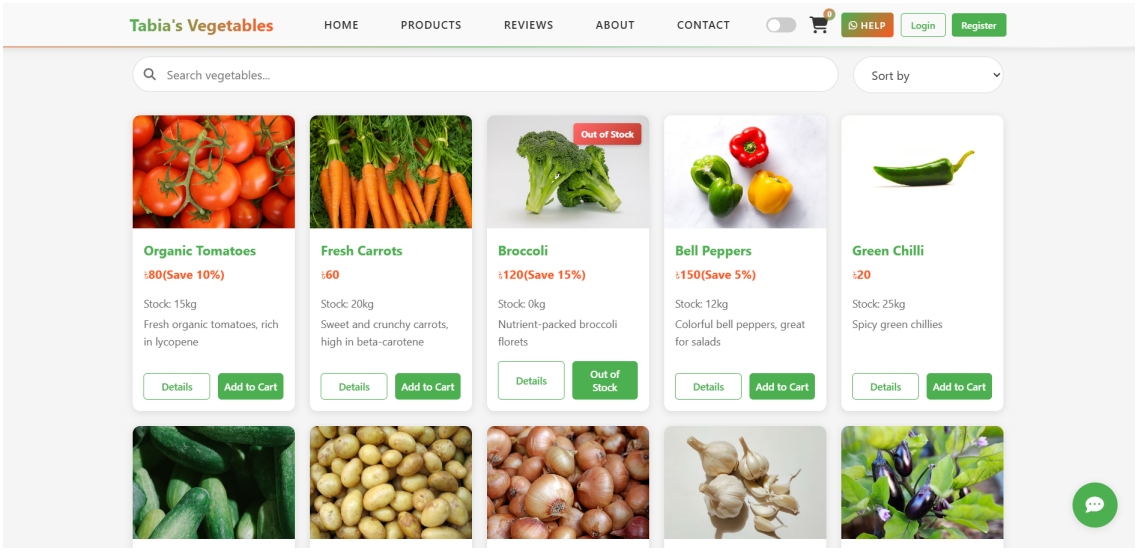

Figure 4.3: Products Section

The above image is the product section of our website where all the product names, product prices, product descriptions, product stock and add to cart information can be found. When a user adds a product by clicking on the button, it will be added to the cart. If the user wants to know the product details, they can see it. They can

also see if there is any special offer. Clicking on the Add to Cart button will add the product to the cart icon function. Users cannot remove any product from the excess stock even if they want to.

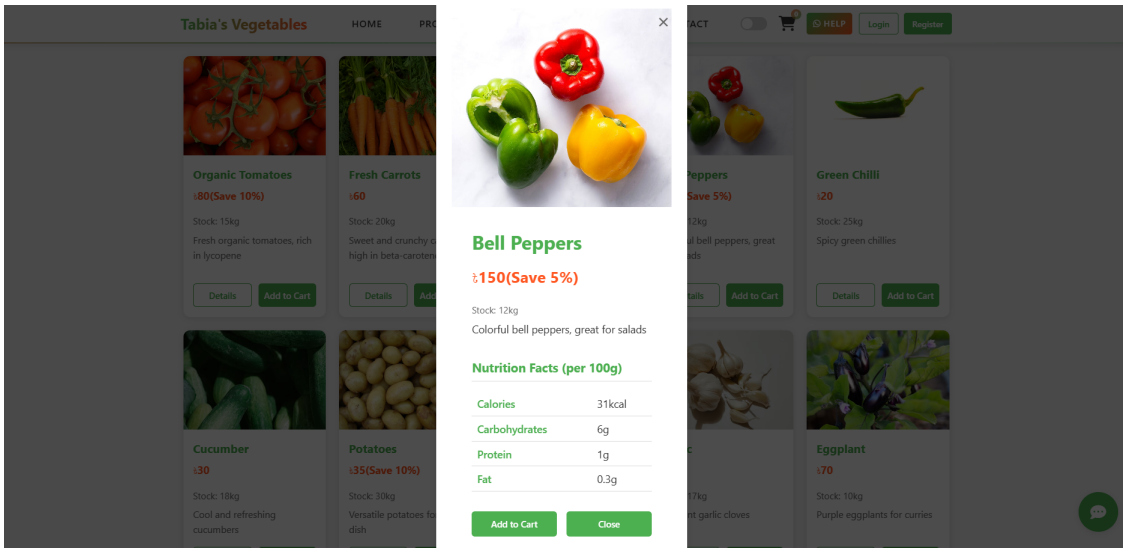

Figure 4.4: Products Section(Product Data)

The image above is the product details where users can see the product image and the amount of nutrients per 100 grams of the product. From here, they can add to cart if they want.

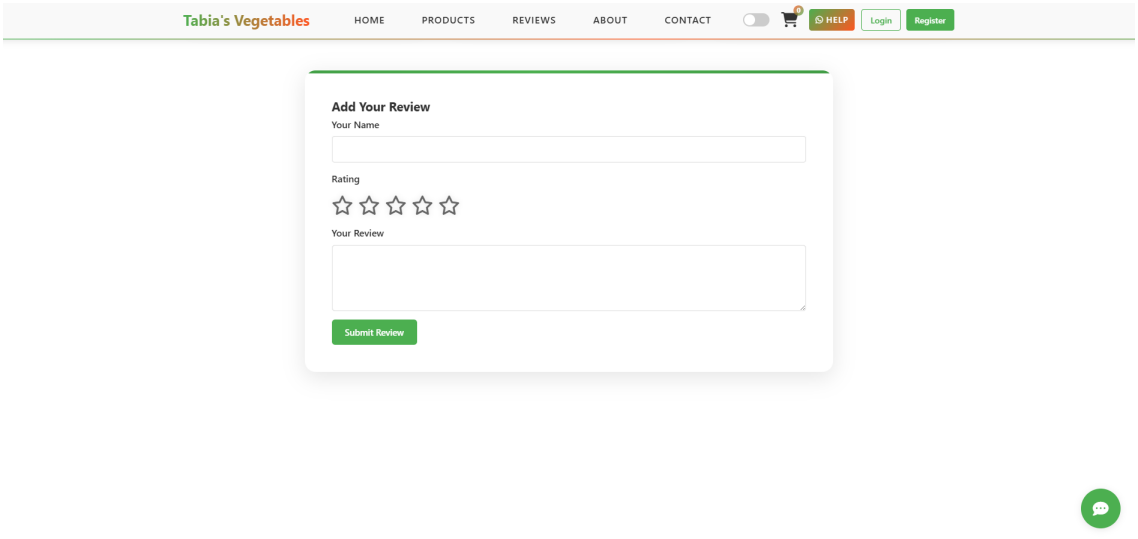

Figure 4.5: Reviews Section

The image above is the review section where users can express their experience about our website by giving their name and a star rating from one to five. They can

also provide some feedback.

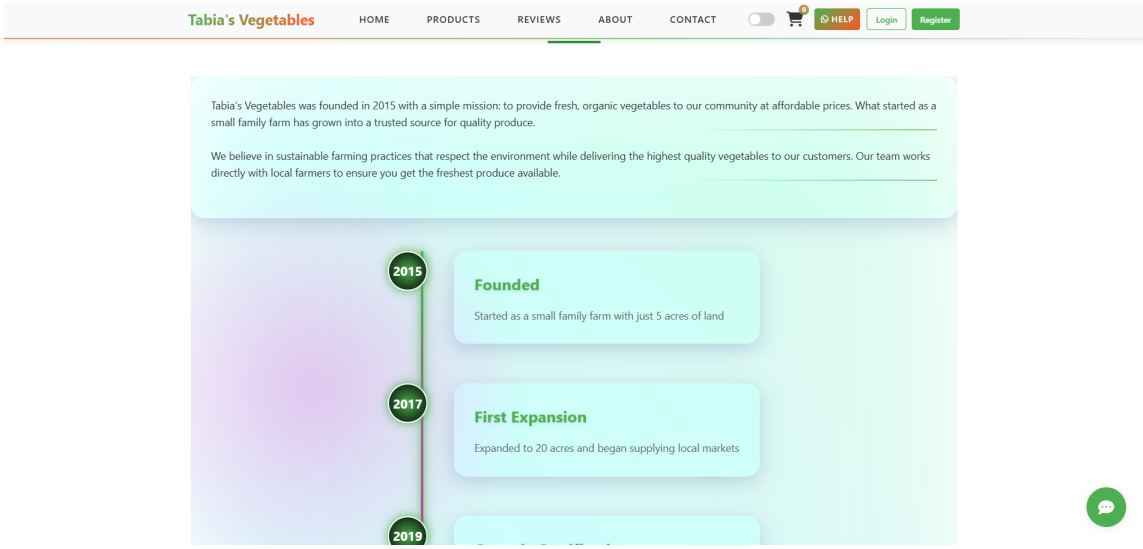

Figure 4.6: About Section

The image above is the about section. This is a very important place where we summarize the changes and improvements that have been made to our website over the years, from the beginning of our organization to the present.

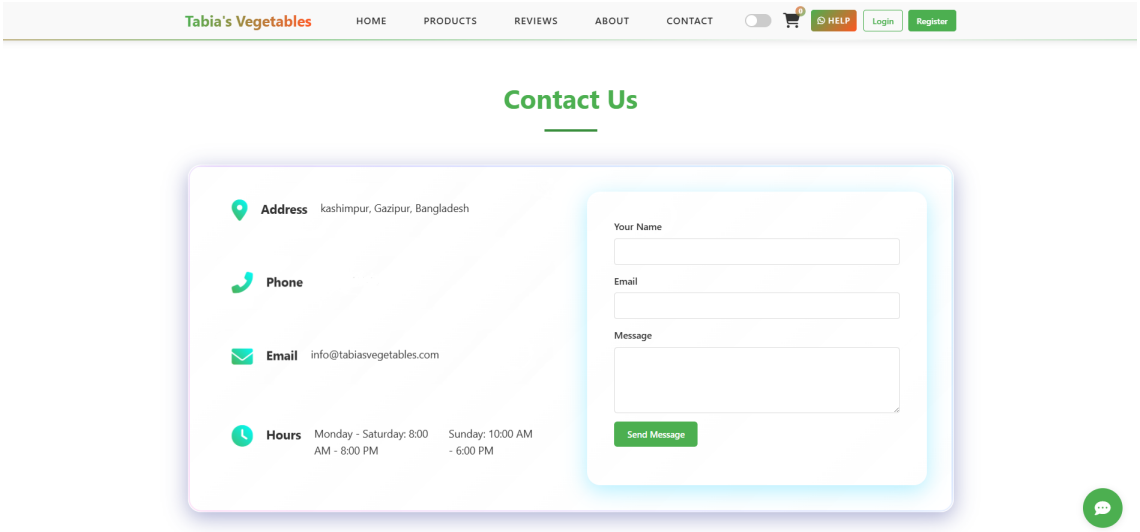

Figure 4.7: Contact Section

The above image is the contract us section where our official Gmail, our mobile number and our office address are given. If any user wants, they can contact us

through message, also by calling and visiting our office. Our working hours are also specified.

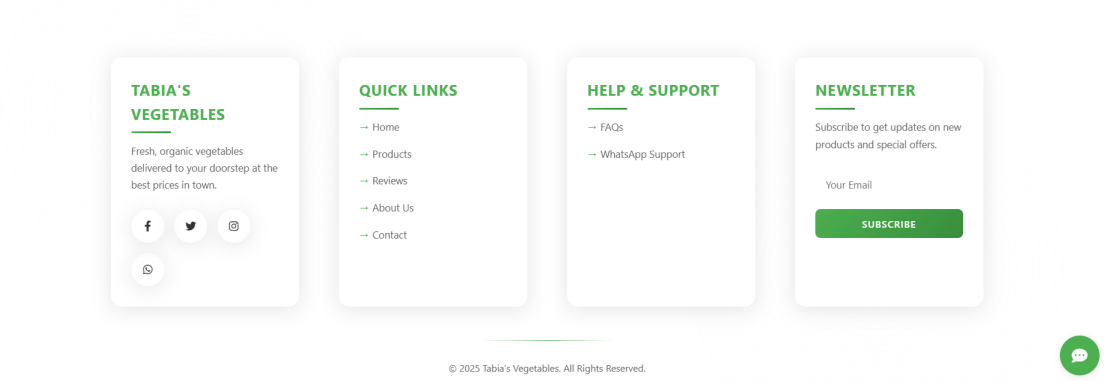

Figure 4.8: Footer

The image above is of our website footer where all the social media links of our website are provided, there is also access to the newsletter and various important menus of our website.

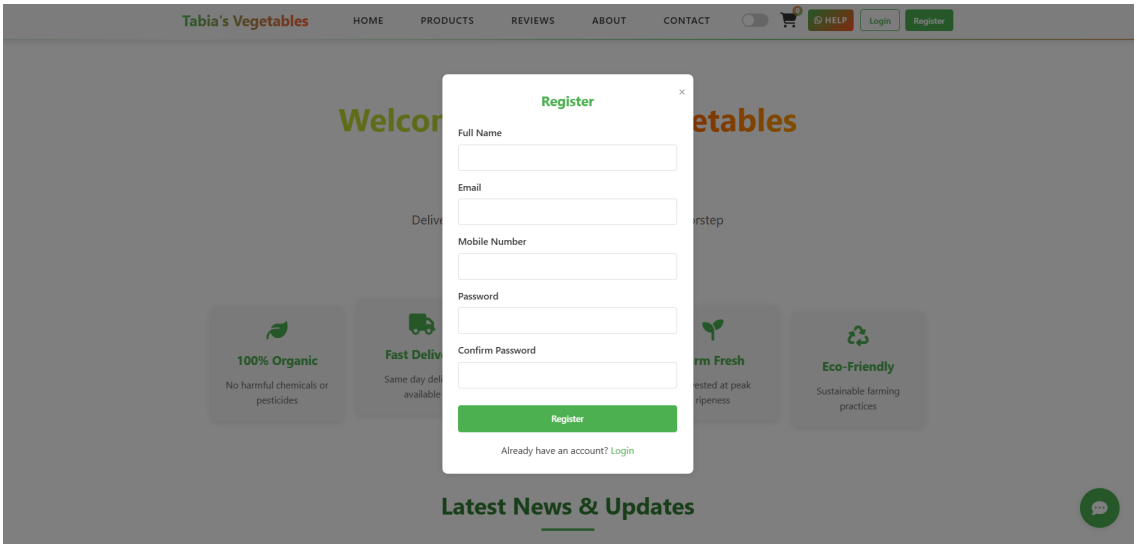

Figure 4.9: Register

The image above is the registration form. Any user can easily register on our website. If a user wants to register, then he/she will have to provide his/her name, Gmail, mobile number and a strong password. Once the registration is completed, he/she will be given a welcoming message.

The image Below is the login form. To log in to our website, a user will need to provide their Gmail address and password. Once the login is complete, they will be

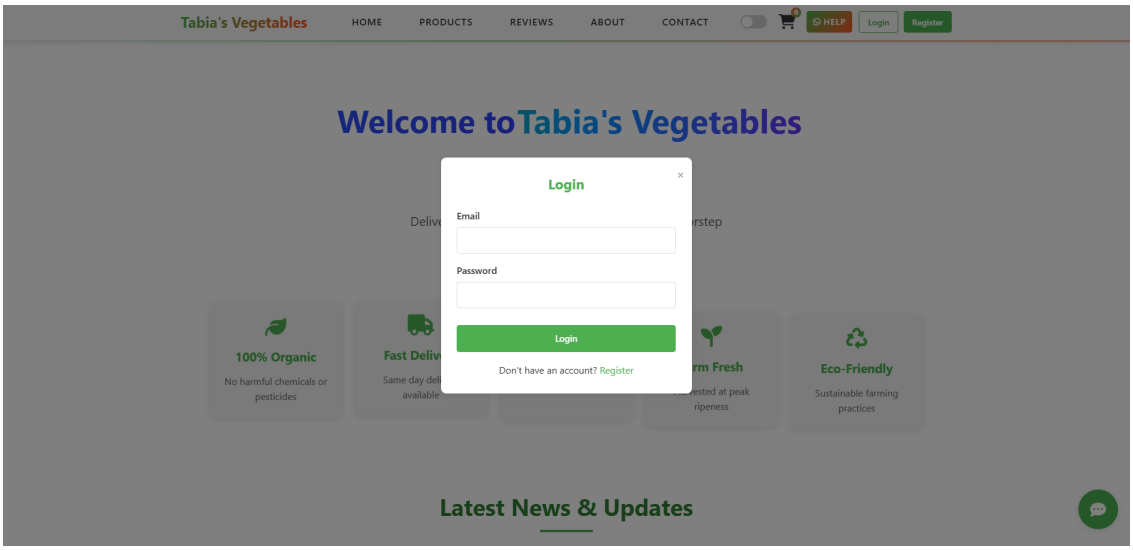

Figure 4.10: Login

given a welcoming message from our website.

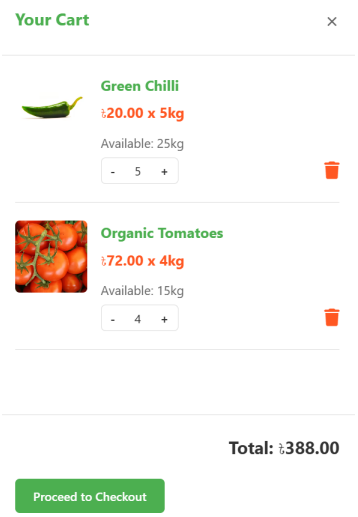

Figure 4.11: Your Cart

The image above is the cart form where the user can see which products they have added to the cart. From there, the user can increase or decrease the quantity of the product if they want. The Below image is the checkout form. This is one of the most important menus on our website where the user has to provide their name, Gmail address, mobile number (which must be in Bangladesh) and the exact address of their home. They can also provide some additional feedback.

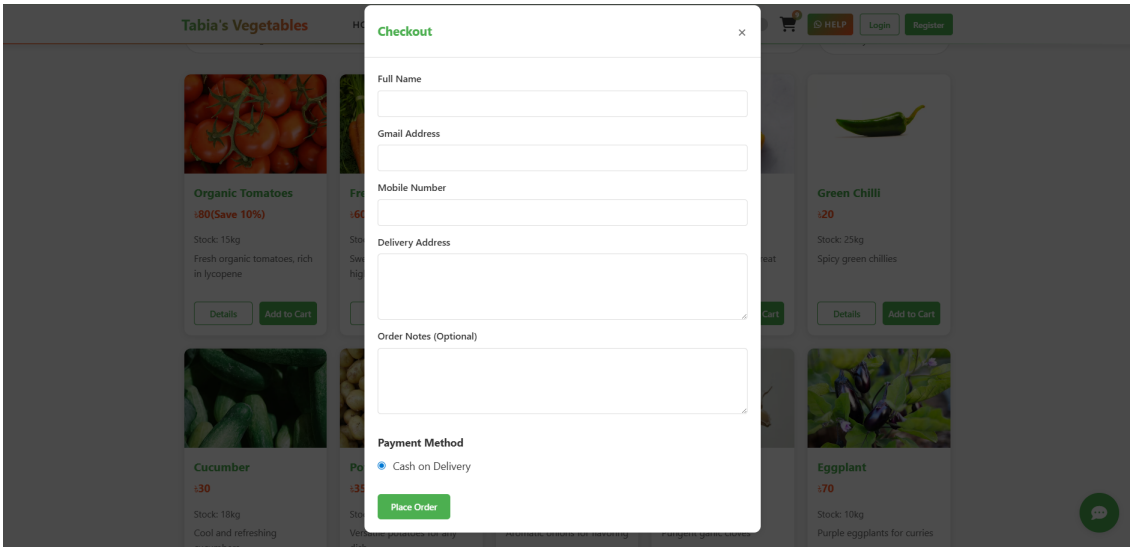

Figure 4.12: Checkout

CHAPTER V

Testing and Evaluation

5.1 Methodology for Testing

A thorough testing methodology was employed to verify the legitimacy and quality of the "Tabia's Vegetables" website, utilizing both manual and automated methods. The testing framework was created to assess cross-platform compatibility, functional correctness, security, and user experience. Chrome Developer Tools were primarily used for debugging purposes, providing code inspection, performance monitoring, and device simulation capabilities. Having the ability to emulate devices without physical hardware was of great value when it came to testing responsive design across different screen sizes and resolutions. All user interactions, such as navigation flows, theme switching, modal operations, and shopping cart functionality, were tested using a step-by-step protocol. This rigorous approach ensured that all components function as specified in the requirements documentation and that the system provided a seamless user experience across various browsing environments and interactions.

5.2 Functional Testing

All essential components were subjected to rigorous testing:

- All UI elements are correctly adjusted, and the theme toggle provides instantaneous switching between light/dark modes.
- The authentication methods ensure that the login/registration form's opening, closing, and transition are easily verified.
- Authentication of shopping cart status updates and persistent saving of localStorage across sessions.

- Navigation was optimized for hamburger menus using a responsive menu with fewer than 768px breakpoints.

5.3 Usability and Responsiveness Testing

Various device configurations and browsing environments were tested for compatibility with the interface. The design demonstrated consistent rendering and performance on current versions of major web browsers, ensuring even user experience regardless of platform. Using device emulation tools, extensive testing was conducted to ensure optimal presentation across various screen sizes for mobile phones and tablets. All touch targets were designed with minimum recommended dimensions for finger-based navigation, with interactive elements receiving particular attention. The responsive design system efficiently adjusted the organization, typographic scales, and spatial relationships of content to ensure intuitive usability across all tested viewport dimensions. This is a first-time enhancement.

5.4 Performance Analysis

The website's technical efficiency and user experience quality were assessed by automated auditing tools through a rigorous process. In several important areas the test yielded exceptional results – especially strong scores for loading speed and accessibility as well as overall, in accordance with web development best practices. The optimized code structure, semantic HTML markup, and efficient resource management strategies of the implementation are evident in these results. It also uncovered specific enhancement opportunities, such as in search engine optimization capabilities where the static content architecture has inherent limitations without server-side rendering or dynamic content generation. Additional optimization methods were identified for future development cycles, such as the incorporation of new image formats, responsive image syntax, and advanced loading techniques to enhance initial render times. By demonstrating that the findings are strong enough in terms of technical foundation for deployment into production, it also provides clear strategic direction to refine performance in future versions through backend integration and further optimization measures.

5.5 Security and Robustness Testing

5.5.1 Stress Testing

The test was run by adding and removing shopping carts using multiple browsers. All animations were working properly and buttons were working properly. All notification messages on the website were coming through properly. Also, the local database was working fine. The test was done manually. No user-related UI freezes or data loss were observed.This ensures that users can use our website with complete peace of mind.

5.5.2 Penetration Testing

I manually checked the website for security issues by entering various types of unusual characters in the input fields. When incorrect input data is given, the website is not accepting that data correctly. Only after providing the correct data, the website gives correct input and output, which ensures that the website is safe from incorrect input.

5.6 Discussion of Results

Testing demonstrates that the website operates as intended. The shopping cart, theme changer, and product display are all functioning without any glitches. Its design is visually appealing and works well on various devices, including phones, tablets or computers. The product data's organization on the frontend is ideal for keeping track of stock and cart availability. Everything runs quickly and reliably. The additional stress and penetration testing confirmed the robustness of the client-side implementation, showing resilient performance under heavy usage and effective resistance to common client-side security threats. This demonstrates that the project was accomplished in crafting a modern, secure, and fully operational online store frontend.

CHAPTER VI

Conclusion and Future Work

6.1 Summary of Findings

The research project demonstrates that modern frontend development techniques, utilizing core web technologies, can produce a fully functional and elegant e-commerce experience. By utilizing modern design principles like responsive design and glassmorphism, the implementation highlights the potential of vanilla HTML, CSS, and JavaScript to create a professional-looking shopping website. This allows it to deliver the essential e-commerce functions of product presentation, user authentication flows, shopping cart management, and theme customization through client-side implementation. The results demonstrate that these technologies can generate user experiences that are engaging and foster trust and interaction, while maintaining high levels of performance across various devices and platforms. Additionally, Online retail applications can be built on this approach without the need for elaborate frameworks or backend infrastructure.

6.2 Contribution to Knowledge

By offering a fully implemented reference architecture that is suitable for specialty retail applications, this project makes savopunkt-level frontend development possible. The investigation provides an up-to-date prototype that demonstrates contemporary design conventions and interaction strategies tailored for the retail of perishable commodities. It introduces a scalable data model that can be scaled backward to handle the more complex product attributes such as inventory, nutrition information and categorization. This work, which is open-source, provides implementable solutions for advanced themes (such as theme switching systems, graphical glassmorphism design),

and responsive shopping cart functionality. The contribution presents a useful development prototype and instructional structure for comprehending modern web development practices, particularly in creating engaging user experiences for e-commerce applications that target specific markets without the use of external frameworks or libraries.

6.3 Limitations of the Current System

Despite its frontend capabilities, the project faces several significant limitations that affect its functionality and practicality.

- Product information is encoded in a static data architecture that requires developer involvement for any changes to inventory, pricing, or product descriptions. This approach takes advantage of the frontend's inherent security.
- Impaired User Authentication: The client side implementation of the login and registration system is not secure server-side validation, so it cannot be used to protect real user data or deliver personalized experiences.
- Backend integration is essential for implementing payment processing, order management, shipping calculations, and inventory synchronization in E-Commerce.
- The client-side data handling approach is not efficient for multiple products as filtering and rendering are done in the user's browser without server support.
- Non-technical staff cannot use content changes without technical knowledge and direct code modification, as there is no administrative interface.

6.4 Recommendations for Future Work

Several developments are required to turn this prototype into a production-ready platform [39]. Using Node.js or Django backend integration will allow dynamic data administration and user authentication [48]. Moving to a component-based approach like React or Vue will improve maintainability and boost scalability [49]. Transaction processing depends on a secure payment gateway being integrated [50]. Search capability, user comments, and customized recommendations are sophisticated aspects that would greatly enhance user experience [51]. Last, formal user research using interviews and A/B testing would give important quantitative data to support and

improve design decisions, hence guaranteeing maximum usability and involvement [52].

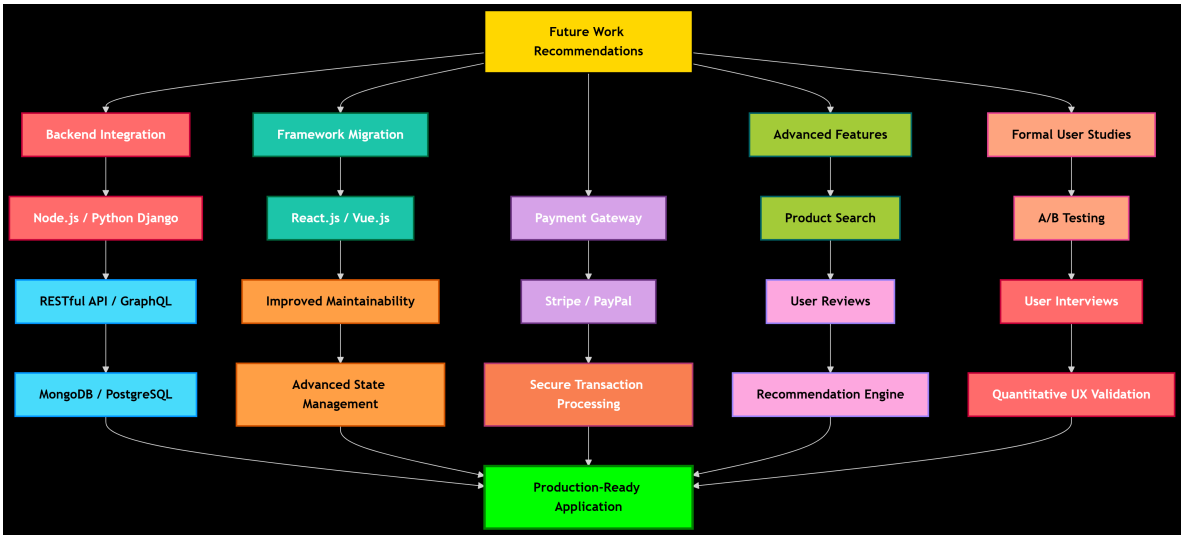

Figure 6.1: Future work plans

References

[1] L. Arendt, "Barriers to ict adoption in smes: how to bridge the digital divide?," *Journal of Systems and Information Technology*, vol. 10, no. 2, pp. 93–108, 2008.

[2] A. N. Tuch, E. E. Presslaber, M. Stöcklin, K. Opwis, and J. A. Bargas-Avila, "The role of visual complexity and prototypicality regarding first impression of websites: Working towards a model," *International Journal of Human-Computer Studies*, vol. 70, no. 11, pp. 794–811, 2012.

[3] D. Cyr, M. Head, and A. Ivanov, "Design aesthetics leading to m-loyalty in mobile commerce," *Information & Management*, vol. 43, no. 8, pp. 950–963, 2006.

[4] D. H. McKnight, V. Choudhury, and C. Kacmar, "Developing and validating trust measures for e-commerce: An integrative typology," *Information systems research*, vol. 13, no. 3, pp. 334–359, 2002.

[5] E. Marcotte, *Responsive web design*. Éditions Eyrolles, 2010.

[6] Y. Zhang and M. Zhang, "Digital divide in small and medium-sized enterprises: A systematic literature review," *Journal of Small Business Management*, pp. 1–30, 2021.

[7] P. B. Cragg and M. King, "Small-firm computing: motivators and inhibitors," *MIS Quarterly*, vol. 17, no. 1, pp. 47–60, 1993.

[8] J. Nielsen, *Usability Engineering*. Morgan Kaufmann, 1994.

[9] M. Richards, *Software Architecture Patterns*. O'Reilly Media, Inc., 2015.

[10] M. Johnson and L. Smith, "State management patterns in single-page applications," *Journal of Web Engineering*, vol. 14, no. 2, pp. 101–120, 2015.

[11] D. Crockford, *JavaScript: The Good Parts*. O'Reilly Media, 2008.

[12] E. Meyer, *Responsive Web Design*. A Book Apart, 2013.

[13] R. T. Fielding, *Architectural styles and the design of network-based software architectures*. PhD thesis, University of California, Irvine, 2000.

[14] R. J. DiQual, *Security Engineering: A Guide to Building Dependable Distributed Systems*. Wiley, 2009.

[15] R. C. Martin, *Agile Software Development: Principles, Patterns, and Practices*. Prentice Hall, 2009.

[16] E. Ries, *The Lean Startup: How Today's Entrepreneurs Use Continuous Innovation to Create Radically Successful Businesses*. Crown Business, 2011.

[17] J. Nielsen, *Designing Web Usability: The Practice of Simplicity*. New Riders Publishing, 1999.

[18] J. J. Garrett, "Ajax: A new approach to web applications," 2005. Accessed: 2023-10-27.

[19] S. Krug, *Don't Make Me Think: A Common Sense Approach to Web Usability*. New Riders Publishing, 2000.

[20] D. Norman, *The Design of Everyday Things: Revised and Expanded Edition*. Basic Books, 2013.

[21] B. Fogg, *Persuasive Technology: Using Computers to Change What We Think and Do*. Morgan Kaufmann Publishers, 2003.

[22] B. Stone, *The Everything Store: Jeff Bezos and the Age of Amazon*. Little, Brown and Company, 2013.

[23] J. J. Garrett, *The Elements of User Experience: User-Centered Design for the Web and Beyond*. New Riders, 2010.

[24] D. Norman, *Emotional Design: Why We Love (or Hate) Everyday Things*. Basic Books, 2004.

[25] S. J. Hoch and J. Deighton, "Who's the fairest of them all? the evolution of trust in business-to-business e-commerce," *Journal of Interactive Marketing*, vol. 14, no. 3, pp. –, 2000.

[26] P. Morville, "User experience design," *Semantic Studios*, 2004.

[27] W. Lidwell, K. Holden, and J. Butler, *Universal Principles of Design, Revised and Updated: 125 Ways to Enhance Usability, Influence Perception, Increase Appeal, Make Better Design Decisions, and Teach through Design*. Rockport Publishers, 2010.

[28] C. Ware, *Information Visualization: Perception for Design*. Morgan Kaufmann, 2012.

[29] S. McCloud, *Understanding Comics: The Invisible Art*. William Morrow Paperbacks, 1994.

[30] J. Tidwell, *Designing Interfaces: Patterns for Effective Interaction Design*. O'Reilly Media, 2010.

[31] J. Thatcher *et al.*, *Web Accessibility: Web Standards and Regulatory Compliance*. Apress, 2006.

[32] M. Vervaeke *et al.*, "The functional art: An introduction to information graphics and visualization," *Design Studies*, vol. 78, p. 101075, 2022.

[33] S. P. Anderson, *Seductive Interaction Design: Creating Playful, Fun, and Effective User Experiences*. New Riders, 2011.

[34] G. Zaccone, *Mastering JavaScript: A Comprehensive Guide to Modern JavaScript Development*. Packt Publishing, 2018.

[35] M. Haverbeke, *Eloquent JavaScript: A Modern Introduction to Programming*. No Starch Press, 2018.

[36] E. Gamma, R. Helm, R. Johnson, and J. Vlissides, *Design Patterns: Elements of Reusable Object-Oriented Software*. Addison-Wesley Professional, 1994.

[37] M. Fowler, *Patterns of Enterprise Application Architecture*. Addison-Wesley Professional, 2002.

[38] R. C. Martin, *Clean Code: A Handbook of Agile Software Craftsmanship*. Prentice Hall, 2009.

[39] M. Fowler, *Refactoring: Improving the Design of Existing Code*. Addison-Wesley Professional, 2018.

[40] N. C. Zakas, *Maintainable JavaScript: Writing Readable Code*. O'Reilly Media, 2012.

[41] I. Sommerville, *Software Engineering.* Addison-Wesley, 9 ed., 2011.

[42] A. Osmani, *Learning JavaScript Design Patterns.* O'Reilly Media, 2012.

[43] B. Lawson and R. Sharp, *Introducing HTML5.* New Riders, 2011.

[44] J. Keith, *HTML5 for Web Designers.* A Book Apart, 2010.

[45] B. Frain, *Responsive Web Design with HTML5 and CSS.* Packt Publishing, 2020.

[46] A. Sullivan, *Adaptive Web Design: Crafting Rich Experiences with Progressive Enhancement.* New Riders, 2011.

[47] A. Stevens, *JavaScript for Impatient Programmers.* ECMAScript, 2020.

[48] C. Richardson, *Microservices Patterns: With examples in Java.* Manning Publications, 2018.

[49] A. Banks and E. Porcello, *Learning React: Functional Web Development with React and Redux.* O'Reilly Media, 2015.

[50] C. Anderson, *Payment Systems in the U.S.* Glenlake Publishing, 2020.

[51] F. Ricci, L. Rokach, and B. Shapira, *Introduction to Recommender Systems Handbook.* Springer, 2011.

[52] R. Kohavi, D. Tang, and Y. Xu, *Trustworthy Online Controlled Experiments: A Practical Guide to A/B Testing.* Cambridge University Press, 2020.