ABSTRACT

**ABSTRACT**

In the modern era of digitalization, agriculture stands at the forefront of technological innovation. The Agro Based E-commerce Web App aims to revolutionize the agricultural sector by providing a seamless platform for farmers and gardening enthusiasts to procure essential supplies.This web application serves as a comprehensive marketplace offering a diverse range of high-quality seeds, fertilizers, and tools necessary for optimal crop cultivation and garden maintenance. With a user-friendly interface and intuitive navigation, users can easily browse through a vast catalog of products tailored to their specific agricultural needs.

**Product Catalog**: A rich assortment of seeds, fertilizers, and tools sourced from trusted suppliers, ensuring reliability and effectiveness in agricultural practices.

**Personalized Recommendations**: Utilizing advanced algorithms, the platform provides personalized product recommendations based on user preferences, crop types, and geographical location.

**Secure Transactions**: Built-in security measures guarantee safe and secure transactions, instilling confidence in users to conduct online purchases without apprehension.

**Expert Guidance**: Access to expert advice and informational resources to assist users in making informed decisions regarding product selection, usage techniques, and best practices in agriculture.

**Community Engagement**: Foster a thriving community of farmers and gardening enthusiasts through forums, discussions, and knowledge-sharing initiatives, promoting collaboration and mutual support.

**Order Tracking and Suppor**t: Real-time order tracking and dedicated customer support ensure a seamless shopping experience, addressing any queries or concerns promptly and effectively.

TABLE OF CONTENTS

**TABLE OF CONTENTS**

|  |  |  |
| --- | --- | --- |
| **S.No** | **CONTENTS** | **Pg.No** |
| **1** | **Introduction** | 1 |
| 1.1 | Project Definition | 2 |
| 1.2 | Project Objective | 3 |
| 1.3 | Project Overview | 3 |
| **2** | **System Specification** | 5 |
| 2.1 | Hardware Specification | 6 |
| 2.2 | Software Specification | 6 |
| 2.3 | Software Description | 6 |
| **3** | **System Study** | 9 |
| 3.1 | Existing System with limitations | 10 |
| 3.2 | Proposed System with advantages | 10 |
| **4** | **System Design** | 12 |
| 4.1 | Data Flow Diagram | 12 |
| 4.2 | Er Diagram | 15 |
| 4.3 | Input Design | 16 |
| 4.4 | Output Design | 17 |
| 4.5 | Database Design | 18 |
| **5** | **System Testing** | 20 |
| 5.1 | Unit Testing | 21 |
| 5.2 | Integration Testing | 22 |
| 5.3 | Functional Testing | 22 |
| 5.4 | System Testing | 23 |
| 5.5 | Acceptance Testing | 23 |
| 5.6 | Validation Testing | 23 |
| **6** | **System Implementation and Maintenance** | 25 |
| **7** | **Conclusion** | 28 |
| **8** | **Scope of Future Enhancements** | 30 |
|  | **Bibliography** | 33 |
|  | **Appendix** | 34 |
| A | Data Flow Diagram | 35 |
| B | ER Diagram | 37 |
| C | Database Design | 38 |
| D | Sample Screen Shots | 43 |
| E | Sample Source Code | 47 |

INTRODUCTION

**1.INTRODUCTION**

In an age where technological advancements continue to redefine various industries, agriculture remains a cornerstone of human civilization. Recognizing the pivotal role of technology in modern farming practices, we present the Agro Based E-commerce Webapp, a pioneering solution designed to bridge the gap between traditional agriculture and digital convenience.

Our web application embodies a vision of empowerment for farmers and gardening enthusiasts alike, offering a virtual marketplace where essential agricultural supplies are just a click away. Seeds, fertilizers, and tools—vital ingredients for successful cultivation—are curated and made accessible through an intuitive and user-friendly interface.

More than just a platform for transactions, our webapp fosters a sense of community and collaboration, facilitating knowledge-sharing and expert guidance to nurture the growth and prosperity of agricultural endeavors. With a commitment to security, reliability, and customer satisfaction, we strive to redefine the agricultural landscape, one digital interaction at a time.

Join us on this journey as we harness the power of technology to cultivate innovation, sustainability, and prosperity in the field of agriculture.

**1.1 PROBLEM DEFINITION:**

Despite the vital role agriculture plays in sustaining our communities and economies, farmers often face challenges in accessing high-quality seeds, fertilizers, and tools essential for crop cultivation and garden maintenance. Traditional methods of procurement are often time-consuming, inefficient, and limited by geographical constraints. Furthermore, the lack of centralized platforms for agricultural supplies leads to difficulties in product verification, price comparison, and accessing expert guidance.

In this context, the Agro Based E-commerce Webapp seeks to address these challenges by providing a comprehensive solution that empowers farmers and gardening enthusiasts to procure essential supplies conveniently and efficiently. By leveraging the capabilities of e-commerce and technology, the web application aims to streamline the procurement process, enhance product accessibility, and foster a sense of community and collaboration within the agricultural sector.

**1.2 PROJECT OBJECTIVE:**

The primary objective of the Agro Based E-commerce Webapp is to revolutionize the agricultural sector by providing a comprehensive digital platform for farmers and gardening enthusiasts to procure high-quality seeds, fertilizers, and tools with ease and convenience. By leveraging the power of technology, the web application aims to:

* Enhance accessibility: Provide farmers with access to a diverse range of agricultural supplies, irrespective of geographical constraints, thereby eliminating barriers to procurement.
* Ensure transparency: Offer transparent information regarding product quality, pricing, and usage techniques, empowering users to make informed decisions and optimize agricultural practices.
* Foster trust: Establish a secure and trustworthy e-commerce environment, instilling confidence in users to conduct online transactions and embrace digital procurement methods.
* Facilitate knowledge-sharing: Serve as a centralized hub for agricultural expertise, providing users with access to expert guidance, best practices, and informational resources to enhance crop cultivation and garden maintenance.

By achieving these objectives, the Agro Based E-commerce Webapp aims to streamline the agricultural supply chain, empower farmers with access to premium supplies and expert guidance, and contribute to the overall enhancement of agricultural productivity, sustainability, and prosperity.

**1.3 PROJECT OVERVIEW:**

**SYSTEM USERS:**

The users in this project are:

* ADMIN
* USER

**SYSTEM MODULE:**

The major modules in the project are:

* User Management Module
* Products Module
* Categories Module
* Admin Dashboard Module
* Cart Module
* Payment Gateway Module

**User Management Module:**

This module facilitates user registration, login, and profile management, ensuring secure access to the platform and personalized experiences for users.

**Products Module:**

Responsible for managing the catalog of agricultural supplies, including seeds, fertilizers, and tools, enabling users to browse, search, and purchase products conveniently.

**Categories Module:**

Organizes products into distinct categories and subcategories, simplifying navigation and enhancing user experience by facilitating intuitive browsing and product discovery.

**Admin Dashboard Module:**

Provides administrators with a centralized interface to manage users, products, orders, and other system settings, enabling efficient oversight and control of the e-commerce platform.

**Cart Module:**

Allows users to add desired products to their shopping carts, review and modify cart contents, and proceed to checkout, facilitating seamless and hassle-free transactions.

**Payment gateway Module:**

Integrates secure payment processing functionality, enabling users to make online payments using various payment methods, ensuring convenience and trust in transactions.

SYSTEM SPECIFICATION

**2. SYSTEM SPECIFICATION**

**2.1 HARDWARE SPECIFICATION**

* Operating System : Windows/Linux/Android/IOS
* Processor : ARMv7 or higher
* RAM : 1GB or more
* Internal storage : At least 8GB of free space
* Display : At least 480 x 800 pixels resolution
* Connectivity : Wi-Fi or mobile data

**2.1 SOFTWARE SPECIFICATION**

* OS : Android, IOS, Windows 11/10 x64, Linux
* CHROME VERSION : 122.0.6167.160 (64-bit)
* OS BUILD : Windows 11
* EXPERIENCE : Web Application

**DESIGN AND DEVELOPMENT ENVIRONMENT:**

* Programming language: Java Script
* Development Environment: React JS, VS Code, Node JS
* Firebase SDK: Latest version available at the time of development
* Database: Firebase Firestore, Firebase Storage, MongoDB
* Security: Encryption and decryption of sensitive user information, and proper handling of user authentication, session management, and other security features.

Our E-commerce website is designed to provide an exceptional shopping experience for customers seeking hardware components. With a user-friendly interface, comprehensive product catalog, secure checkout process, and reliable customer support, we aim to become the preferred destination for hardware enthusiasts and professionals alike.

**2.3 SOFTWARE DESCRIPTION:**

**JAVASCRIPT:**

JavaScript is a versatile and dynamic programming language primarily used for creating interactive web pages. It enables developers to add behavior to web pages, create dynamic content, and handle user interactions, making it an essential component of modern web development.

**MONGO DB:**

MongoDB is a popular NoSQL database management system known for its flexibility, scalability, and performance. It stores data in a flexible, JSON-like format, allowing for easy integration with modern web applications. MongoDB is well-suited for handling large volumes of unstructured or semi-structured data and is commonly used in environments where rapid development and scalability are paramount.

**NODE JS:**

Node.js is a runtime environment that allows developers to run JavaScript code on the server-side. It utilizes the V8 JavaScript engine from Google Chrome and provides a lightweight and efficient platform for building scalable network applications. Node.js is commonly used for developing web servers, APIs, and real-time applications, leveraging its non-blocking, event-driven architecture to handle concurrent connections efficiently.

**FIREBASE:**

Firebase is a comprehensive platform provided by Google for building mobile and web applications. It offers a suite of services, including real-time database, authentication, cloud storage, hosting, and more, all of which are seamlessly integrated and managed from a single console. Firebase's real-time database and authentication services are particularly popular for rapidly prototyping and deploying modern web applications, providing developers with the tools they need to build high-quality, engaging user experiences with minimal overhead.

SYSTEM STUDY

**3. SYSTEM STUDY**

**3.1 EXISTING SYSTEM WITH LIMITATIONS:**

The current landscape for agricultural supply procurement presents several challenges for farmers and gardening enthusiasts. One of the primary limitations lies in the accessibility of quality agricultural products. Traditional brick-and-mortar stores often offer a limited selection of seeds, fertilizers, and tools, leaving farmers with few choices. Moreover, geographical constraints exacerbate this issue, particularly for those in remote or rural areas where access to stores and transportation infrastructure is limited.

Another significant drawback of the existing system is the lack of standardized procedures for product verification. Farmers struggle to assess the quality and authenticity of agricultural supplies, leading to the risk of purchasing substandard or counterfeit products. This uncertainty further complicates an already inefficient procurement process, which involves manual tasks such as visiting multiple stores, comparing prices, and negotiating deals, resulting in productivity losses.

Furthermore, the current system fails to provide farmers with adequate information and expertise. The absence of reliable guidance on best practices, optimal product usage techniques, and crop-specific recommendations hinders farmers' ability to maximize agricultural productivity and profitability. Additionally, concerns regarding transaction security and trustworthiness of vendors deter farmers from embracing online purchasing options, perpetuating reliance on traditional procurement methods.

The fragmented ecosystem further compounds these challenges, as the absence of a centralized platform for agricultural supplies makes it difficult for farmers to discover new products, connect with suppliers, and stay updated on industry trends and innovations. These limitations underscore the pressing need for a dedicated Agro Based E-commerce Webapp that addresses the challenges faced by farmers, streamlines the procurement process, enhances product accessibility and transparency, and fosters a sense of community and collaboration within the agricultural sector.

**3.2 PROPOSED SYSTEM WITH ADVANTAGES:**

The Agro Based E-commerce Web App offers a transformative solution to the limitations inherent in traditional agricultural supply procurement systems. By migrating to an online platform, farmers and gardening enthusiasts gain unparalleled access to a diverse range of high-quality seeds, fertilizers, and tools, transcending geographical barriers and streamlining the procurement process. With transparent transactions, secure payment gateways, and personalized product recommendations, the web app instills confidence in users, enabling them to make informed decisions and embrace online purchasing options with ease.

Furthermore, the webapp serves as more than just a marketplace; it fosters a vibrant community of farmers and gardening enthusiasts, facilitating knowledge-sharing, collaboration, and support. Through expert guidance, informational resources, and community-driven initiatives, users can optimize agricultural practices, enhance productivity, and contribute to the collective growth and sustainability of the agricultural sector. The Agro Based E-commerce Webapp represents a paradigm shift in agricultural supply procurement, leveraging technology to empower users, foster innovation, and drive positive change in the agricultural landscape.

SYSTEM DESIGN

**4. SYSTEM DESIGN**

System design is "the process of studying a procedure or business in order to identify its goals, purposes and create systems and procedures that will achieve them in an efficient way". Another view sees system analysis as a problem-solving technique that breaks down a system into its component pieces for the purpose of studying how well those component parts work and interact to accomplish their purpose.

**4.1 DATA FLOW DIAGRAM:**

A data-flow diagram (DFD) is a way of representing the flow of data of a process or a system (usually an information system). DFD also provides information about the outputs and inputs of each entity and the process itself. A data-flow diagram has no control flow, there are no decision rules and no loops. Specific operations based on the data can be represented by a flowchart. The data-flow diagram is part of the structured analysis modeling tools. When using

UML, the activity diagram typically takes over the role of the data-flow diagram. A special form of data-flow plan is a site- oriented data flow plan.

A data-flow diagram is a way of representing a flow of data through a process or a system (usually an information system). The DFD also provides information about the outputs and inputs of each entity and the process itself. A data-flow diagram has no control flow, there are no decision rules and no loops. Specific operations based on the data can be represented by a flowchart.

Gane Sarson diagram templates have been used to draw this diagram. The four basic Gane Sarson symbols are mentioned

Notations:

|  |  |  |
| --- | --- | --- |
| Symbol | Name | Description |
|  | Source or destination | Also known as actors, sources or sinks, and terminators, produce and consume data that flows between the entity and the system being diagrammed. These data flows are the inputs and outputs of the DFD. |
|  | Process | An activity that changes or transforms data flows. Since they transform incoming data to outgoing data, all processes must have inputs and outputs on a DFD |
|  | Data flow | Movement of data between external entities, processes and data stores is represented with an arrow symbol, which indicates the direction of flow. |
|  | Data store | A data store does not generate any operations but simply holds data for later access. Data sores could consist of files held long term or a batch of documents stored briefly while they wait to be processed. Input flows to a data store include information or operations that change the stored data. |

Fig 4.1 DFD Notations

**Refer Appendix For DFD diagram**

**4.2 ENTITY RELATIONSHIP DIAGRAM:**

An entity relationship diagram (ERD), also known as an entity relationship model, is a graphical representation of an information system that depicts the relationships among people, objects, places, concepts or events within that system. An ERD is a data modeling technique that can help define business processes and be used as the foundation for a relational database.

Notations:

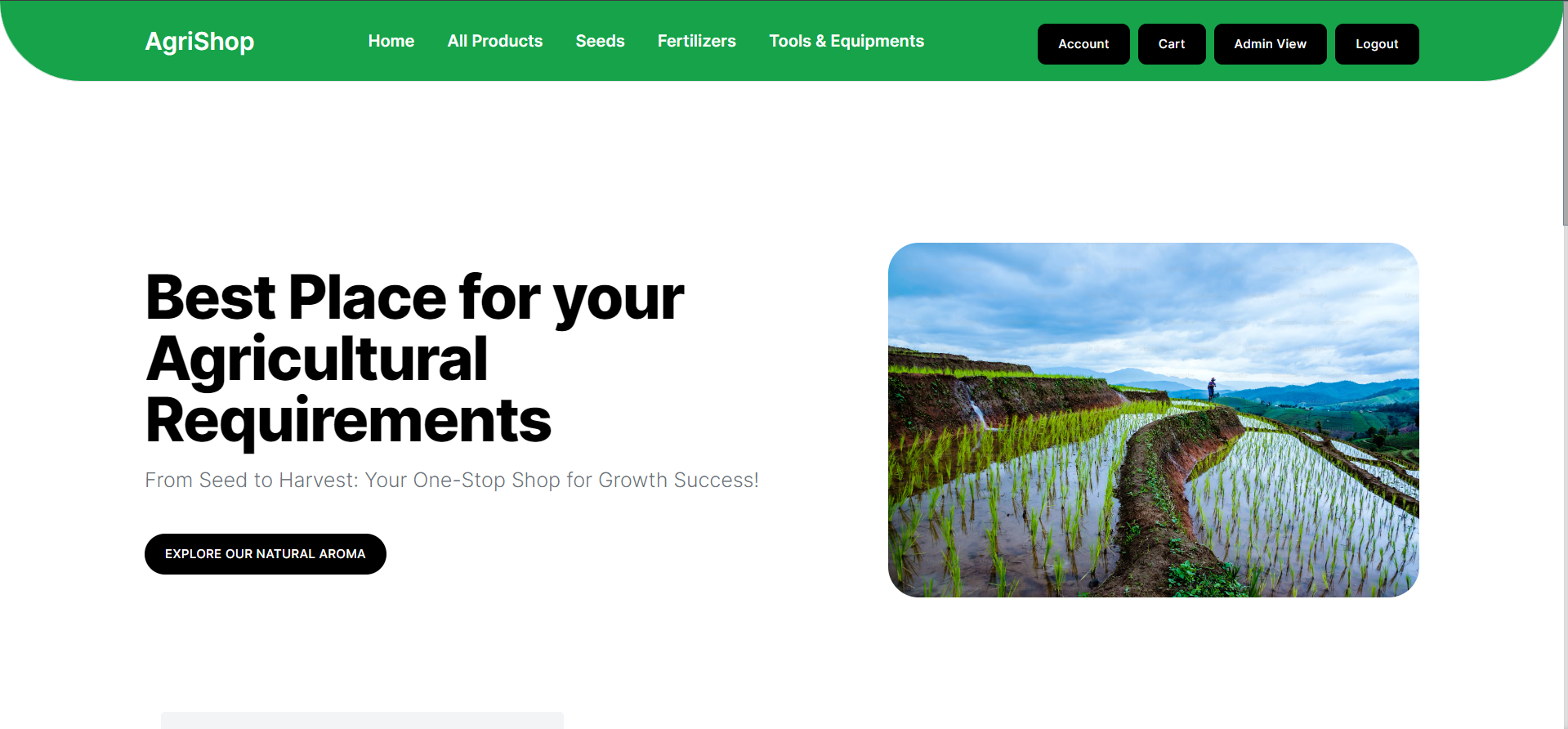
|  |  |  |
| --- | --- | --- |
| Entity Symbol | Name | Description |
|  | Strong Entity | These shapes are independent from other entities, and are often called parent entities, since they will often have weak entities that depend on them. They will also have a primary key, distinguishing each occurrence of the entity. |
|  | Associative entity | Associative entities relate the instances of several entity types. They also contain attributes specific to the relationship between those entity instances |
|  | Relationship | Relationships are associations between or among entities. |
|  | Attributes | Attributes are characteristics of an entity, a many-to-many relationship, or a one-to-one relationship. |

Fig 4.2 ERD Notation

**Refer Appendix B for ER Diagram**

**4.3 INPUT DESIGN:**

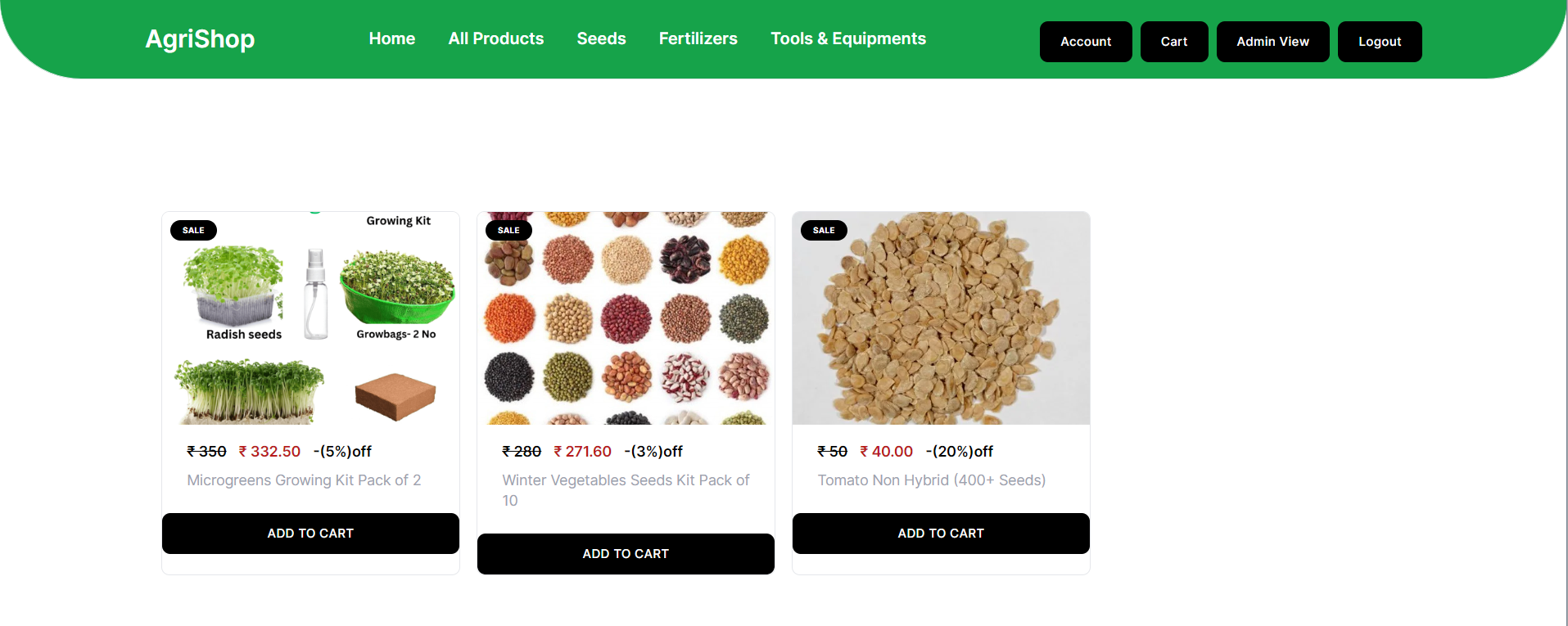
Input design is one of the most important system design phases. Input design is the process where. The input received in the system is planned and designed, so as to get only necessary information from the user, eliminating the information that is not required. The aim of the input design is to ensure the maximum possible levels of accuracy, to make data entry easier, and to be free from errors.



**4.4 OUTPUT DESIGN:**

One of the most important factors of the system is the output it produces. Output refers to the results and information that is generated by the system. Basically, the output from a computer system is used to communicate the result of the processing to the user. Output design is the process that involves designing necessary outputs which helps the user according to their requirements. Efficient output design should improve the system relationship with the user and help in decision-making.

Output of the computer is the most important and direct source of information to the user. Output design should improve the system in relationship with the user's help in decision making. Once the output document can be carried out a major form of output is the hand copy from the printer or writing the data into the file in a specified format.



**4.5 DATABASE DESIGN:**

Database is designed to manage large bodies of information. The management of data involves both the definitions of structure for the storage of information. In addition, the database system must provide the safety of the information, despite system crashes or due to attempts at unauthorized access. For developing an efficient database user has to fulfill certain conditions such as controlled redundancy.

**Data Constraints:**

All business in the world runs on business data being gathered, stored and analyzed.

**Types of Data Constraints:**

There are two types of data constraints that can be applied to data being inserted into a database table. One type of constraint is called I/O constraint. The other type of constraint is called a business rule constraint.

**I/O Constraints:**

The input /output data constraint is further divided into two distinctly different constraints.

**The Primary Key Constraint:**

Here the data constraint attached to a column ensures:

1. That the data entered in the table column is unique across the entire column.
2. That none of the cells belonging to the table column are left empty.

**Refer Appendix For Database Design**

SYSTEM TESTING

**5.SYSTEM TESTING**

In a software development project, errors can be injected at any stage during the development. Testing performs a very critical role for quality and for ensuring the reliability of software. During testing, the program to be tested is executed with a set of test cases, and the output of the program for the test cases is evaluated to determine if the program is performing as it is expected to. The main objective of testing is to uncover errors from the system. For the uncovering process users have to give proper input data to the system. So users should be more conscious of giving Input data.

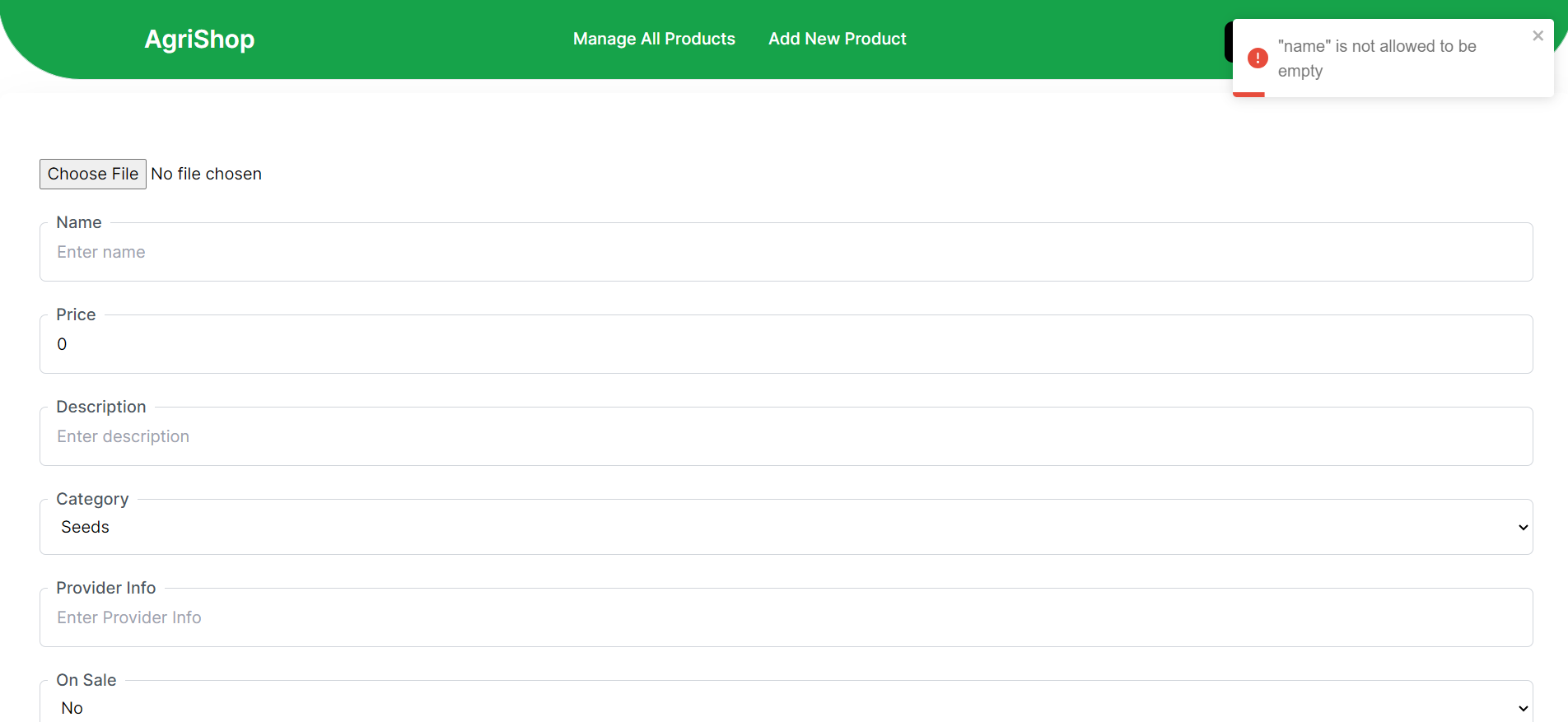
It is important to give correct inputs to efficient testing. Testing is a process of executing a program with the intent of finding an error. presents an ultimate review of specification, design and coding. System Testing is an important phase. Testing represents an interesting anomaly for the software. A good test case is one that has a high probability of finding an undiscovered error. A successful test is one that uncovers an undiscovered error.

**5.1 UNIT TESTING**

Unit testing is commenced when a unit has been created and effectively reviewed .In order to test a single module we need to provide a complete environment i.e. besides the section we would require.

* The procedures belong to other units that the unit under test calls.
* Non local data structures that module accesses.

A procedure to call the functions of the unit under test with appropriate parameters.



**5.2 INTEGRATION TESTING**

In the Integration testing we test various combinations of the project module by providing the input. The primary objective is to test the module interfaces in order to confirm that no errors are occurring when one module invokes the other module. Groups them in larger aggregates, applies tests defined in an integration test plan to those aggregates, and delivers as its output the integrated system ready for system testing.

**5.3 FUNCTIONAL TESTING**

Functional testing is a type of software testing whereby the system is tested against the functional requirements/specifications. It refers to activities that verify a specific action or function of the code. Functional tests tend to answer the questions like “can the user do this” or “does this particular feature work”. The techniques used for functional testing are often specification-based.

Testing functionality can be done from two perspectives:

**5.4 SYSTEM TESTING**

System testing is the stage of implementation, which is aimed at ensuring that the system works accurately and efficiently before live operations commences. Testing is the process of executing the program with the intent of finding errors and missing operations. The ultimate aim is quality assurance. Tests are carried out and the results are compared with the expected document. In the case of erroneous results, debugging is done. Using detailed testing strategies, a test plan is carried out on each module.

**5.5 ACCEPTANCE TESTING**

Acceptance testing involves planning and executing of functional tests, performance tests and stress tests in order to demonstrate that the implemented system satisfies its requirements. Functional test causes involve excising the code with nominal input values for which expected results are known. Giving different input values tests it.

**5.5.1 PERFORMANCE TESTING**

Performance testing determines the amount of executing time spent in various paths of the program unit, program throughput, the response time and utilization by the program unit. Performance testing is based on the maximum volume of existing data, which the system can handle with an effective throughput, and efficient utilization of the system resources. Software system developed in this manner is one that satisfies the user needs, confirms its requirement and design specification, and exhibits an absence of errors. he final process should be a software audit where the complete software project is T checked to ensure that it meets production

management requirements. This ensures that all requirement documentation has been produced, is in the correct format and is of acceptable quality.

**5.6 VALIDATION TESTING:**

Verification is a process of evaluating the intermediary work products of a software development lifecycle to check if we are on the right track of creating the final product. These can include the documents which are produced during the development phases like, requirements specification, design documents, database table design, ER diagrams, test cases. verification is a process to evaluate the mediator products of software to check whether the products satisfy the conditions imposed during the beginning of the phase.

SYSTEM IMPLEMENTATION AND MAINTENANCE

**6. SYSTEM IMPLEMENTATION AND MAINTENANCE**

The successful implementation and maintenance of the Agro Based E-commerce Webapp are crucial for its effectiveness and longevity. The implementation process begins with the development and deployment of the web application, leveraging technologies such as JavaScript, Node.js, MongoDB, and Firebase. Users, including farmers, vendors, and administrators, will undergo training sessions to familiarize themselves with the platform's features and functionalities, supported by user-friendly tutorials and documentation.

Rigorous testing and quality assurance procedures will be conducted to identify and rectify any bugs, errors, or performance issues, ensuring the webapp meets the highest standards of quality and reliability. Upon completion, the webapp will be gradually rolled out to users, starting with a pilot phase to gather feedback and refine the platform based on user input. Marketing efforts will accompany the rollout to increase awareness and encourage adoption within the target audience.

Continuous monitoring and optimization will be conducted post-launch to ensure smooth operation and performance. This includes monitoring system health, user engagement metrics, and feedback channels to identify areas for improvement and implement enhancements. Ongoing maintenance and support services will be provided, including regular software updates, security patches, and technical assistance, to ensure the long-term viability and success of the Agro Based E-commerce Webapp.

CONCLUSION

**7. CONCLUSION**

In conclusion, the Agro Based E-commerce Webapp represents a significant leap forward in the realm of agricultural supply procurement, offering a holistic solution to the challenges faced by farmers and gardening enthusiasts. By harnessing the power of technology, the webapp streamlines the procurement process, enhances product accessibility and transparency, and fosters knowledge-sharing and community building within the agricultural sector. With its user-friendly interface, personalized recommendations, and secure transactions, the webapp empowers users to make informed decisions and embrace online purchasing options with confidence. Moving forward, the Agro Based E-commerce Webapp holds immense potential to revolutionize the agricultural landscape, driving positive change, and contributing to the overall enhancement of agricultural productivity, sustainability, and prosperity.

SCOPE OF FUTURE ENHANCEMENT

**8. FUTURE ENHANCEMENT**

While the Agro Based E-commerce Webapp represents a significant advancement in agricultural supply procurement, there are several avenues for future enhancements and improvements:

* **Integration of IoT Devices**: Implementing Internet of Things (IoT) technology can enable real-time monitoring of environmental conditions such as soil moisture, temperature, and humidity. Integrating IoT devices with the webapp can provide users with valuable insights and recommendations for optimized crop cultivation and resource management.
* **Expansion of Product Range**: Continuously expanding the catalog of agricultural supplies to include a wider variety of products, such as pesticides, herbicides, and irrigation systems, can further enhance the webapp's value proposition and cater to the diverse needs of users.
* **Adoption of Machine Learning Algorithms**: Leveraging machine learning algorithms for predictive analytics can enable the webapp to anticipate user preferences, forecast demand trends, and optimize inventory management, leading to more efficient supply chain operations and improved user satisfaction.
* **Localization and Multi-language Support**: Providing localization features and multi-language support can facilitate access to the webapp for users in different regions and language preferences, thereby increasing its reach and usability on a global scale.
* **Implementation of Blockchain Technology**: Exploring the use of blockchain technology for supply chain transparency and traceability can enhance trust and accountability in transactions, ensuring the authenticity and provenance of agricultural products from farm to table.
* **Expansion to Mobile Platforms**: Developing dedicated mobile applications for iOS and Android platforms can offer users greater flexibility and convenience in accessing the web app's features and functionalities, catering to the growing trend of mobile-centric browsing and shopping habits.

By embracing these future enhancements, the Agro Based E-commerce Webapp can continue to evolve and adapt to the changing needs and expectations of users, cementing its position as a leading solution in the agricultural sector and driving sustainable growth and innovation in the industry.

BIBLIOGRAPHY

**BIBLIOGRAPHY**

**BOOK REFERENCE:**

1. Title: Revolutionizing Agriculture: A Guide to Modern Farming Techniques

Authors: John Smith, Emily Johnson, and Michael Lee

Publisher: Greenfield Publishers

Publication Year: 2023

ISBN: 978-1-234-56789-0

**WEBSITE REFERENCE:**

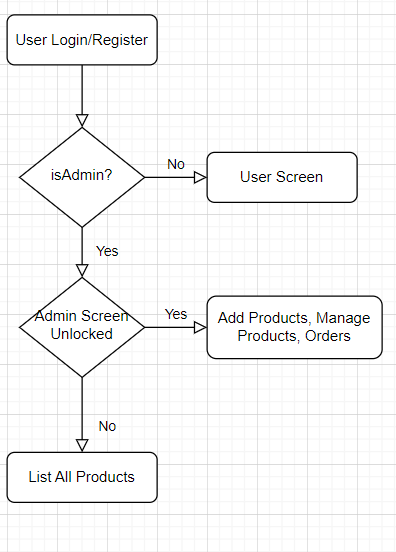
1. Title: AgroTech Solutions: Empowering Farmers through Technology

URL: [www.agrotechsolutions.com](http://www.agrotechsolutions.com)

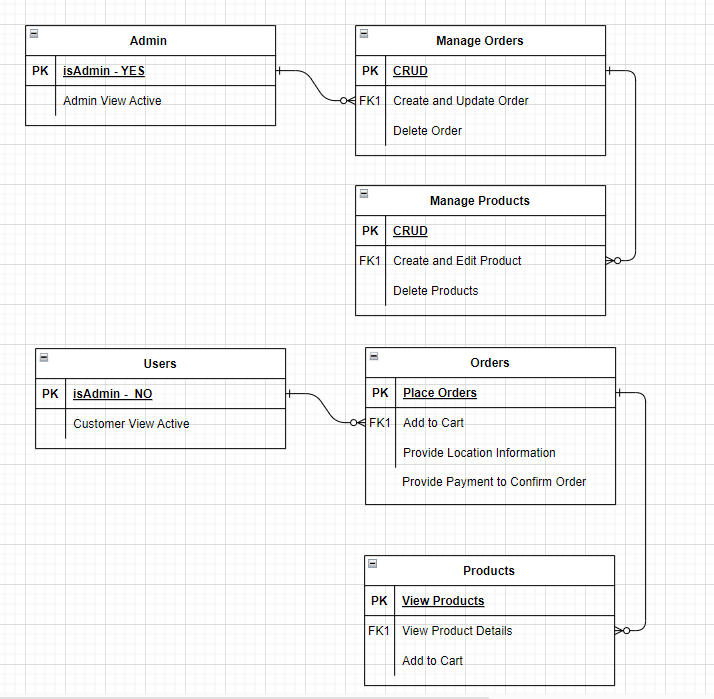
APPENDIX

**APPENDIX**

**A. DATA FLOW DIAGRAM:**



**B. ER DIAGRAM:**



**C. TABLE DESIGN:**

* **Products Table:**
  + ProductID (Primary Key)
  + Name
  + Description
  + CategoryID (Foreign Key referencing Categories Table)
  + Price
  + QuantityAvailable
  + ImageURL
  + Specifications (if applicable)
  + CreatedAt
  + UpdatedAt
* **Categories Table:**
  + CategoryID (Primary Key)
  + Name
* **Orders Table:**
  + OrderID (Primary Key)
  + UserID (Foreign Key referencing Users Table)
  + OrderDate
  + TotalAmount
  + Status (e.g., Pending, Shipped, Delivered)
  + PaymentMethod
  + PaymentStatus
  + ShippingAddress
  + CreatedAt
  + UpdatedAt
* **OrderItems Table:**
  + OrderItemID (Primary Key)
  + OrderID (Foreign Key referencing Orders Table)
  + ProductID (Foreign Key referencing Products Table)
  + Quantity
  + UnitPrice
  + TotalPrice
* **Users Table:**
  + UserID (Primary Key)
  + Name
  + Email
  + PasswordHash
  + Address
  + PhoneNumber
  + CreatedAt
  + UpdatedAt
* **Admins Table:**
  + AdminID (Primary Key)
  + Name
  + Email
  + PasswordHash
* **Cart Table (for temporary storage of user's shopping cart items):**
  + CartID (Primary Key)
  + UserID (Foreign Key referencing Users Table)
  + ProductID (Foreign Key referencing Products Table)
  + Quantity
  + CreatedAt
  + UpdatedAt

**D. SAMPLE SCREEN SHOTS:**

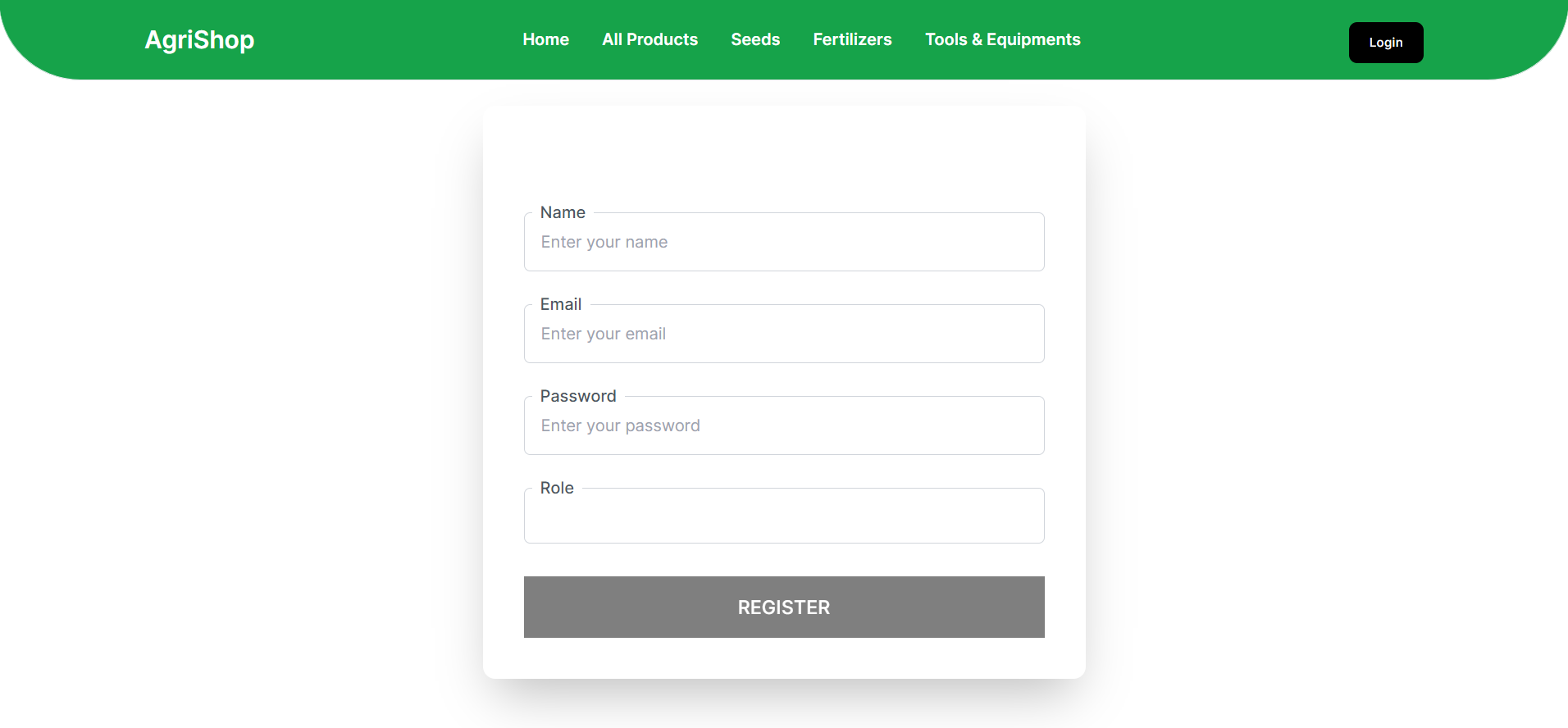
****

Fig: D.1.REGISTER SCREEN

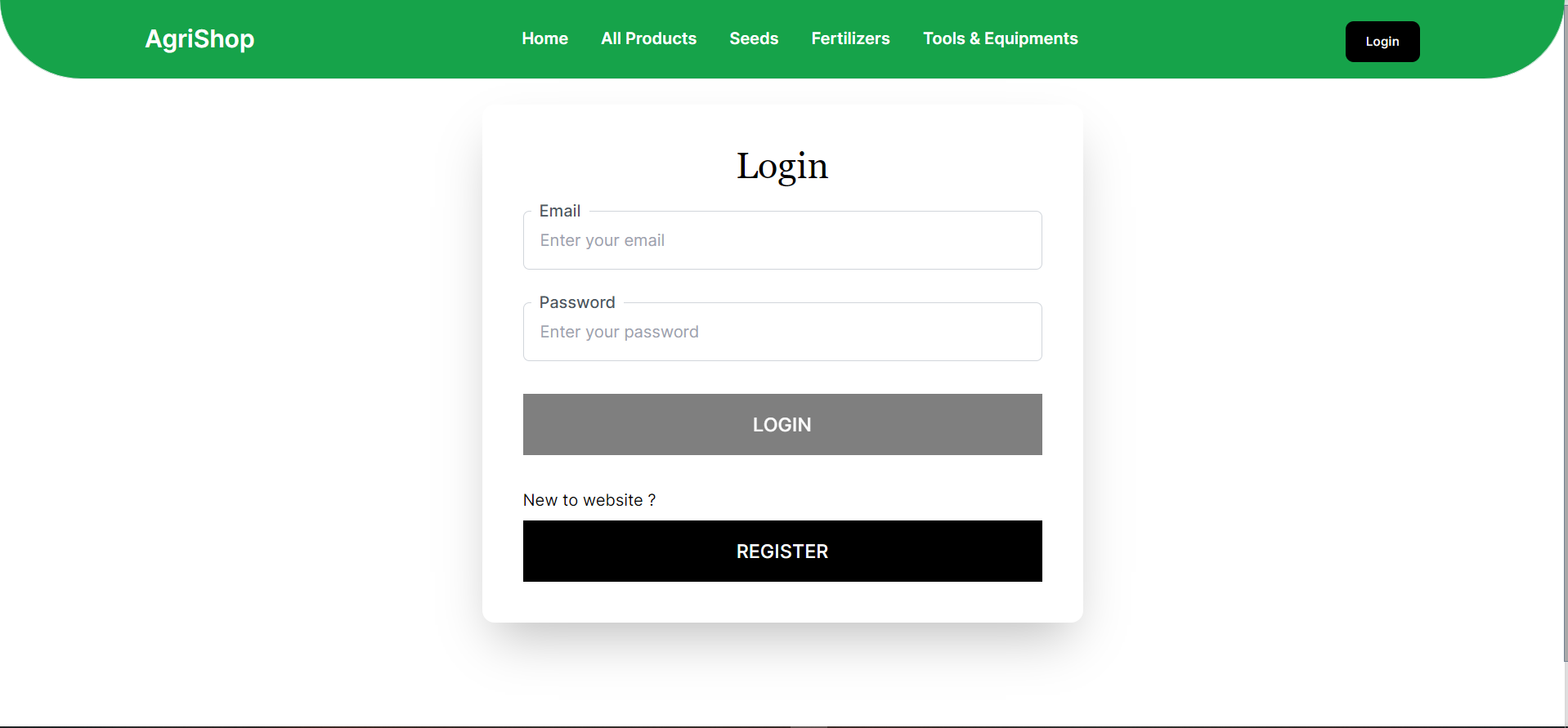


Fig: D.2.LOGIN SCREEN

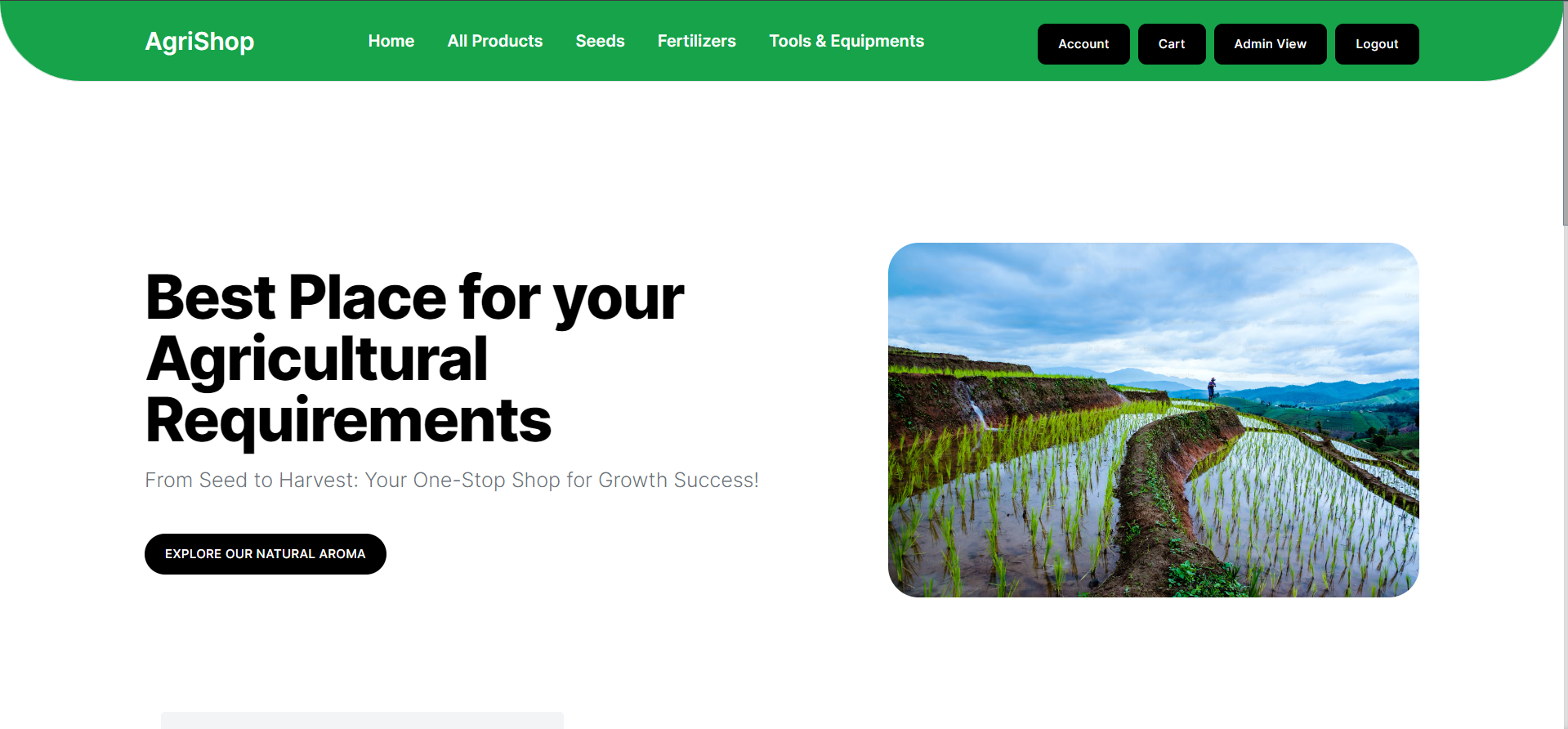


Fig: D.3.HOME SCREEN

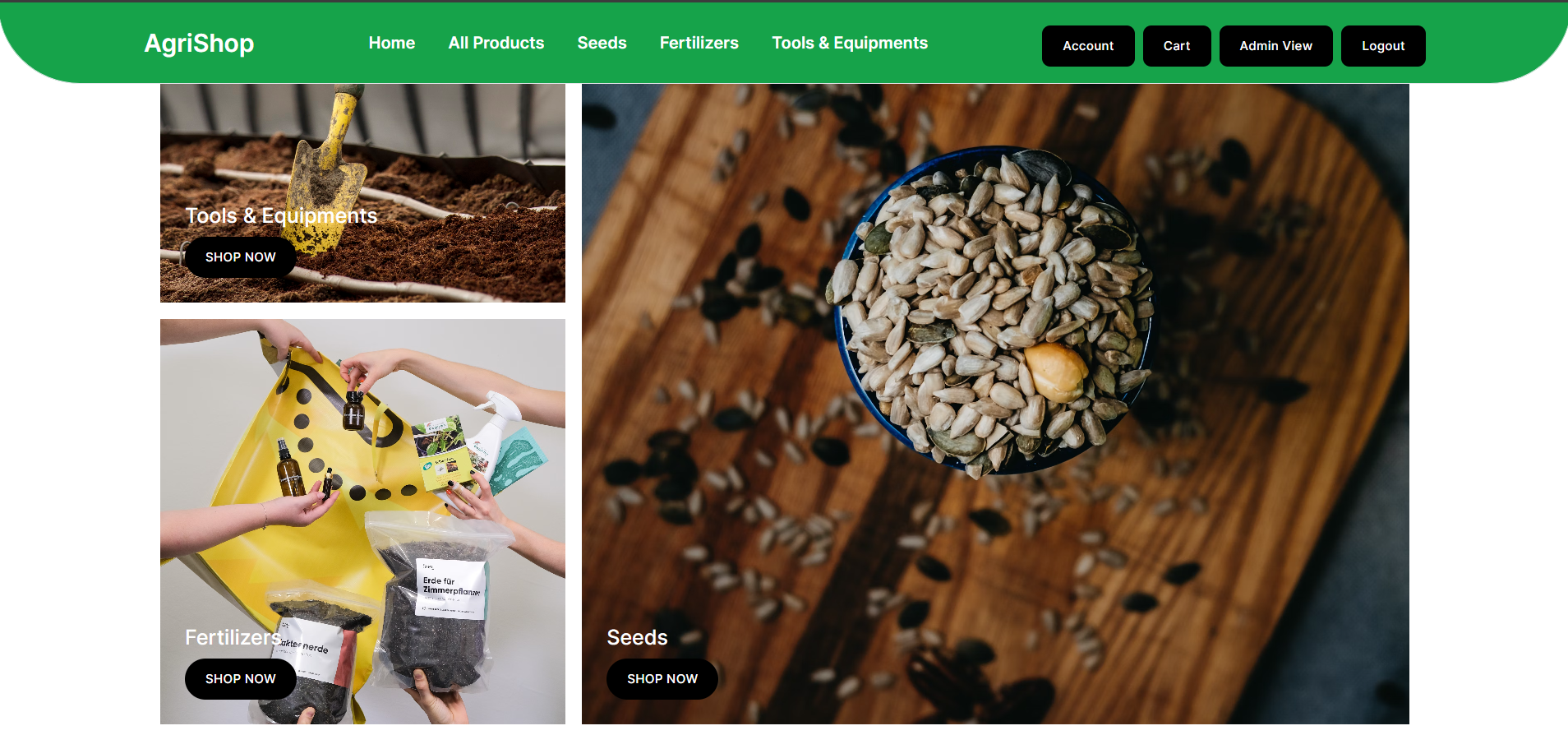


Fig: D.4.HOME SCREEN 1

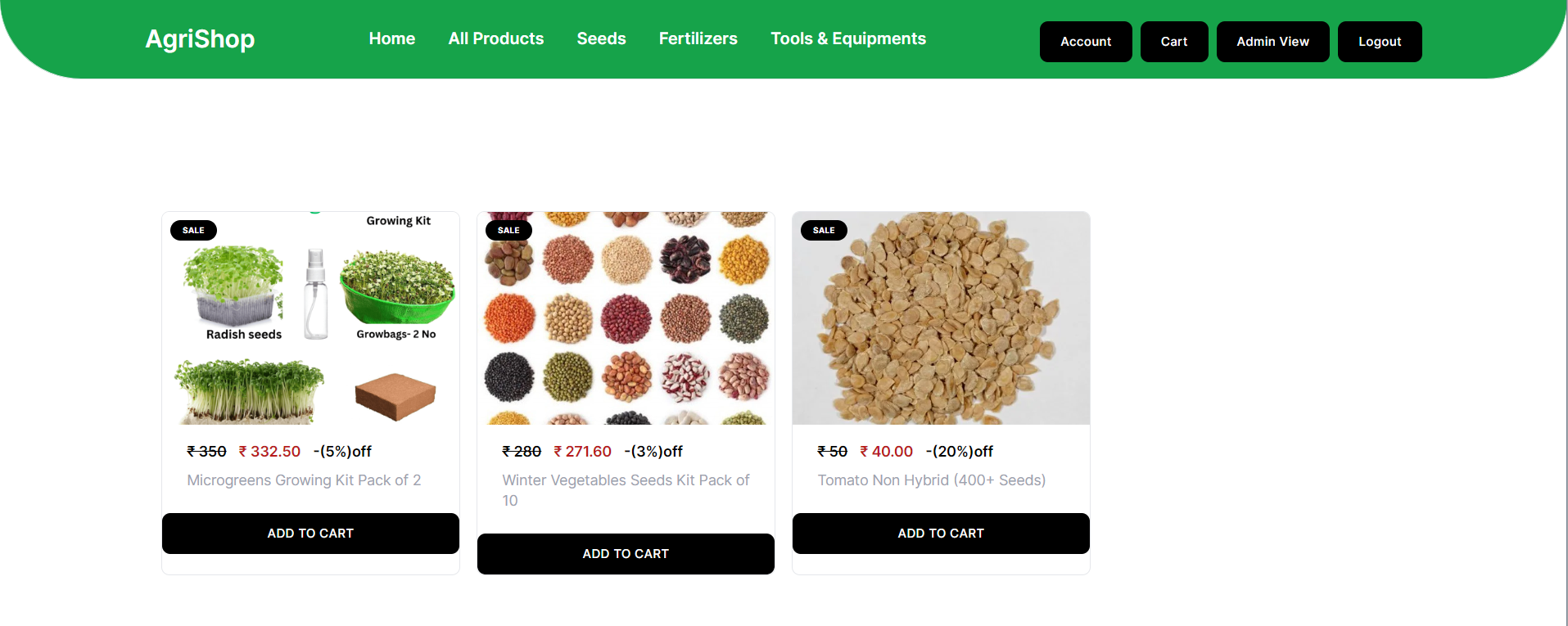


Fig: D.5.PRODUCT LIST SCREEN

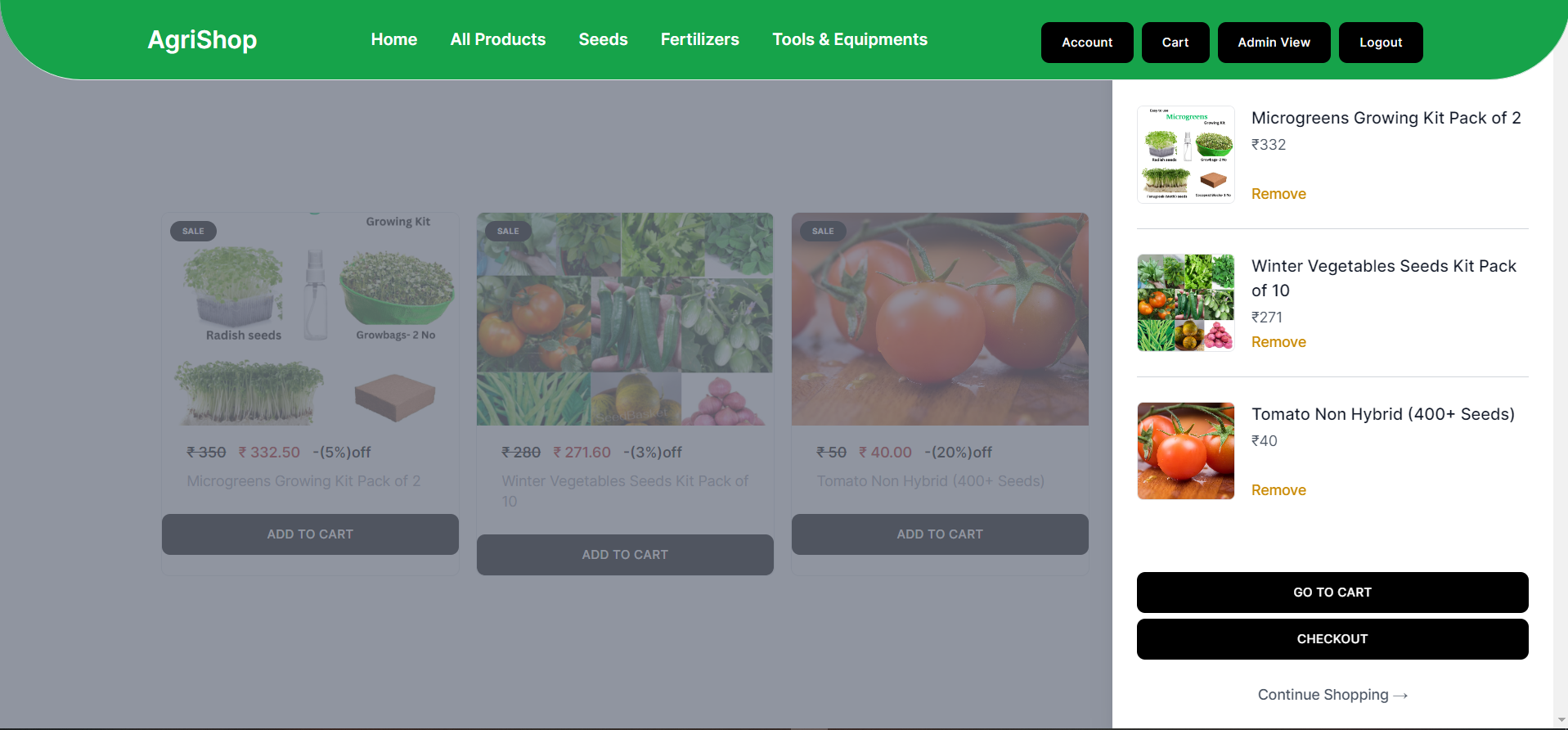


Fig: D.6.CART SCREEN

**E.SAMPLE SOURCE CODE:**

**LOGIN PAGE:**

"use client";

import InputComponent from "@/components/FormElements/InputComponent";

import ComponentLevelLoader from "@/components/Loader/componentlevel";

import Notification from "@/components/Notification";

import { GlobalContext } from "@/context";

import { login } from "@/services/login";

import { loginFormControls } from "@/utils";

import Cookies from "js-cookie";

import { useRouter } from "next/navigation";

import { useContext, useEffect, useState } from "react";

import { toast } from "react-toastify";

const initialFormdata = {

email: "",

password: "",

};

export default function Login() {

const [formData, setFormData] = useState(initialFormdata);

const {

isAuthUser,

setIsAuthUser,

user,

setUser,

componentLevelLoader,

setComponentLevelLoader,

} = useContext(GlobalContext);

const router = useRouter();

console.log(formData);

function isValidForm() {

return formData &&

formData.email &&

formData.email.trim() !== "" &&

formData.password &&

formData.password.trim() !== ""

? true

: false;

}

async function handleLogin() {

setComponentLevelLoader({ loading: true, id: "" });

const res = await login(formData);

console.log(res);

if (res.success) {

toast.success(res.message, {

position: toast.POSITION.TOP\_RIGHT,

});

setIsAuthUser(true);

setUser(res?.finalData?.user);

setFormData(initialFormdata);

Cookies.set("token", res?.finalData?.token);

localStorage.setItem("user", JSON.stringify(res?.finalData?.user));

setComponentLevelLoader({ loading: false, id: "" });

} else {

toast.error(res.message, {

position: toast.POSITION.TOP\_RIGHT,

});

setIsAuthUser(false);

setComponentLevelLoader({ loading: false, id: "" });

}

}

console.log(isAuthUser, user);

useEffect(() => {

if (isAuthUser) router.push("/");

}, [isAuthUser]);

return (

<div className="bg-white relative text-black">

<div className="flex flex-col items-center justify-between pt-0 pr-10 pb-0 pl-10 mt-8 mr-auto xl:px-5 lg:flex-row">

<div className="flex flex-col justify-center items-center w-full pr-10 pl-10 lg:flex-row">

<div className="w-full mt-10 mr-0 mb-0 ml-0 relative max-w-2xl lg:mt-0 lg:w-5/12">

<div className="flex flex-col items-center justify-start pt-10 pr-10 pb-10 pl-10 bg-white shadow-2xl rounded-xl relative z-10">

<p className="w-full text-4xl font-medium text-center font-serif">

Login

</p>

<div className="w-full mt-6 mr-0 mb-0 ml-0 relative space-y-8">

{loginFormControls.map((controlItem) =>

controlItem.componentType === "input" ? (

<InputComponent

type={controlItem.type}

placeholder={controlItem.placeholder}

label={controlItem.label}

value={formData[controlItem.id]}

onChange={(event) => {

setFormData({

...formData,

[controlItem.id]: event.target.value,

});

}}

/>

) : null

)}

<button

className="disabled:opacity-50 inline-flex w-full items-center justify-center bg-black px-6 py-4 text-lg

text-white transition-all duration-200 ease-in-out focus:shadow font-medium uppercase tracking-wide

"

disabled={!isValidForm()}

onClick={handleLogin}

>

{componentLevelLoader && componentLevelLoader.loading ? (

<ComponentLevelLoader

text={"Logging In"}

color={"#ffffff"}

loading={

componentLevelLoader && componentLevelLoader.loading

}

/>

) : (

"Login"

)}

</button>

<div className="flex flex-col gap-2">

<p>New to website ?</p>

<button

className="inline-flex w-full items-center justify-center bg-black px-6 py-4 text-lg

text-white transition-all duration-200 ease-in-out focus:shadow font-medium uppercase tracking-wide

"

onClick={() => router.push("/register")}

>

Register

</button>

</div>

</div>

</div>

</div>

</div>

</div>

<Notification />

</div>

);

}