

B.Tech AI&DS
Anna University
MIT Campus
Chromepet



DBMS - Mini Project

Agro Nexum Live

Grower's Marketplace Management System

January 02, 2024

Done by

Mirsha A K -2022510044
Blessed Nevin -2022510036
Rameez Akther M -2022510025

AgroNexumLive

Transforming agricultural landscapes, bringing together farmers, livestock experts, and consumers, our platform simplifies produce listings, delivering bespoke news and security with transparent transactions. Join our thriving, trusted community in cultivating a more connected and vibrant agricultural future.

Table of Contents

1. Executive Summary	4
1.1 Purpose	
1.2 Target Users	
1.3 Unique Value Proposition	
2. Introduction	5
2.1 Connecting Agriculture to consumer	
2.2 Motivation and Purpose	
2.3 Filling the Market Void	
2.4 Main Objectives	
3. Market Analysis	7
3.1 Understanding Today's Agricultural Landscape	
3.2 Addressing the Disconnect	
3.3 Evidencing the Demand	
3.4 Filling the Market Void	
4. Unique Selling Points	8
4.1 Agrarian-Centric USPs	
4.2 Consumer-Centric USPs	
5. Technical Stack	9
5.1 Backend Framework	
5.2 Frontend Technologies	
5.3 Database	
5.4 Third-Party Integrations	
6. Software Requirements	10
7. Hardware Requirements	10
8. Features and Functionality	11
8.1 For Farmers	
8.2 For Consumers	
8.3 Mobile-Friendly Experience	
8.4 Login through OTP	
9. User Experience	13
9.1 Farmer-Centric Experience	
9.2 Consumer-Focused Design	
10. Contents and working of AgroNexumLive	14
10.1 Home Page	
10.2 User Entities	
10.3 Sign Up Page	
10.4 Login Page	
10.5 Agrarian Register Page	
10.6 Add Produce Type Page	
10.7 Agrarian Details Page	
10.8 Available Products Page	
10.9 Communication Channel for AgroNexumLive	

11.Database	17
11.1 Modeling Language	
11.2 Data Structures	
11.3 Query Language and Report Writer	
11.4 Data Security	
11.5 Interactive Database Management	
11.6 Transaction Mechanism (ACID Properties)	
11.7 Redundancy Avoidance	
11.8 Adaptability to Information Changes	
12.SQL (Structured Query Language)	19
12.1 Data Definition (DDL)	
12.2 Data Manipulation (DML)	
12.3 Data Retrieval	
12.4 Data Filtering and Sorting	
12.5 Data Aggregation	
12.6 Data Joins	
12.7 Data Integrity Enforcement	
12.8 Transactional Control	
12.9 Index Management	
12.10 User Authorization and Security	
12.12 Stored Procedures and Functions	
12.13 Views	
12.14 Triggers	
12.15 Data Backup and Recovery	
13.An Overview of our database design	21
13.1 Tables	
13.2 Auto-increment Information	
13.3 Index Information	
13.4 Views	
13.5 Functions	
13.6 Stored Procedures	
13.7 Triggers	
13.8 Summary	
14.Conceptual ER Diagram	24
15.Schema diagram	25
16.Future Development and Expansion	26
16.1 Scalability and Technological Adaptability	
16.2 Enhanced User Experience and Features	
16.3 Market Expansion and Community Building	
16.4 Security and Privacy Enhancements	
17.Conclusion	28

Executive Summary

AgroNexumLive emerges as a pioneering platform designed to foster symbiotic relationships between growers, livestock luminaries, and consumers. Our primary objective is to revolutionize the agricultural landscape by integrating these crucial stakeholders, thereby enhancing efficiency and transparency across the supply chain.

Purpose

AgroNexumLive serves as the nexus where growers, livestock luminaries, and consumers converge, facilitating seamless interactions, knowledge sharing, and transactions. By leveraging technology, our platform aims to transcend barriers and bridge the gap between traditional farming practices and modern consumer demands.

Target Users

Our platform caters to a diverse audience encompassing farmers seeking innovative cultivation methods, livestock luminaries striving for industry insights, and discerning consumers eager for sustainable, quality agricultural products.

Unique Value Proposition

AgroNexumLive stands out through its holistic approach, offering a comprehensive suite of tools and resources. It empowers growers with data-driven insights, connects livestock luminaries with valuable market trends, and provides consumers access to ethically sourced, high-quality produce.

In essence, AgroNexumLive represents not just a platform but a transformative force driving agricultural evolution. It embodies collaboration, knowledge exchange, and a commitment to sustainability, positioning itself as a pivotal player in shaping the future of agriculture.

Introduction

AgroNexumLive, a visionary digital platform, serves as the dynamic bridge knitting together farmers, agrarians, and consumers in an interconnected ecosystem. Our venture stems from a profound motivation to address the longstanding gaps within the agricultural market, driven by the imperative need for cohesion and transparency across its diverse stakeholders.

Connecting Agriculture to Consumers

At its core, AgroNexumLive redefines the agricultural landscape by propelling the convergence of farmers and consumers onto a unified platform. This synergy seeks to establish a direct, symbiotic relationship, eliminating intermediaries and fostering an environment where agricultural producers can directly engage with and serve the end consumers.

Motivation and Purpose

The genesis of AgroNexumLive arises from the recognition of fragmented interactions within the agricultural sector. Traditional models often obscure transparency and impede direct connections between growers and consumers. Our motivation is to dismantle these barriers, empowering farmers with broader market reach while providing consumers access to ethically sourced, farm-fresh produce.

Filling the Market Void

The project emerges as a response to a conspicuous void in the market – the absence of a centralized, accessible platform that not only promotes agricultural products but also encapsulates the essence of sustainable farming practices, fostering a more informed and conscientious consumer base.

AgroNexumLive stands poised at the forefront of this transformative journey, pioneering a paradigm shift in agricultural interactions. By seamlessly interlinking agrarians and consumers, it strives to cultivate a more transparent, sustainable, and mutually beneficial agricultural ecosystem.

Main Objective

- Design and develop a user-friendly system for improved accessibility and ease of use.
- Create an efficient computerized system that streamlines processes for enhanced productivity.
- Develop an accurate and flexible system to eliminate data redundancy, ensuring data integrity.
- Study the functioning of the Farm Management System for insights into agricultural processes.

- Design software with fast processing capabilities and a user-friendly interface for optimal usability.
- Ensure the software's longevity with a robust user interface, allowing for easy modifications and extended use without errors.
- Establish a synchronized and centralized farmer and seller database for efficient data management.
- Utilize computerization to save time and costs, contributing to resource efficiency.
- Implement a visually appealing Graphical User Interface (GUI) for enhanced user experience.
- Enhance data security through login and password methods, minimizing the risk of information leakage.
- Facilitate immediate storage and retrieval of data, contributing to real-time decision-making.
- Improve arrangements for farmer coordination, fostering better collaboration and communication.
- Mitigate losses through efficient data management and process optimization.

Market Analysis

Understanding Today's Agricultural Landscape

The agricultural market currently operates amidst fragmented interactions, posing challenges for both farmers and consumers. Farmers often face obstacles in reaching consumers directly, relying on intermediaries that limit their profitability. Similarly, consumers encounter difficulties in accessing fresh, ethically sourced produce due to these disconnected channels. This disconnect results in inefficiencies, higher prices, and limited choices for both parties involved.

Addressing the Disconnect

AgroNexumLive emerges as a solution to bridge this gap. By serving as a digital platform connecting farmers directly with consumers, it streamlines the distribution process, ensuring a more efficient and transparent pathway for agricultural products. Through this direct connection, consumers gain access to a wider array of farm-fresh goods while empowering farmers to showcase their produce and practices directly to end-users.

Evidencing the Demand

The genesis of AgroNexumLive arises from the recognition of fragmented interactions within the agricultural sector. Traditional models often obscure transparency and impede direct connections between growers and consumers. Our motivation is to dismantle these barriers, empowering farmers with broader market reach while providing consumers access to ethically sourced, farm-fresh produce.

Filling the Market Void

Demand for a platform like AgroNexumLive is substantiated by market trends and studies. Consumer preferences are shifting towards knowing the origins of their food and supporting sustainable practices. Surveys highlight a growing desire for transparency and trust in the food supply chain, where consumers prioritize ethical sourcing and farm-to-table experiences.

Moreover, farmers express a need to expand their market reach beyond traditional avenues, seeking fairer compensation for their efforts and more direct connections with consumers. AgroNexumLive aligns with these evolving needs by facilitating transparent, direct interactions, promoting sustainable farming, and ensuring fairer returns for farmers.

AgroNexumLive represents a pivotal solution poised to revolutionize the agricultural landscape. By alleviating existing market challenges and responding to the burgeoning demand for transparency and direct-to-consumer connections, it promises to reshape how agricultural goods are exchanged and experienced.

Unique Selling Points

Agrarian-Centric USPs

Efficient Listing and Management

AgroNexumLive redefines the way agrarians showcase their produce. The platform offers a streamlined process, empowering farmers to efficiently list and manage their products. Simplified interfaces and tools allow for easy updates, ensuring their offerings are consistently up-to-date for consumers.

Networking Opportunities

Beyond a mere marketplace, AgroNexumLive functions as a digital ecosystem fostering connections among farmers. It serves as a nexus where agrarians collaborate, exchange knowledge, and form partnerships. By facilitating these interactions, the platform enables a community-driven approach to agriculture, promoting innovation and best practices.

Customized Profiles

Each agrarian on AgroNexumLive benefits from a comprehensive profile. These profiles serve as detailed showcases, highlighting not only their produce but also their expertise and practices. Through these personalized profiles, consumers gain insights into the journey of the products they purchase, fostering a deeper connection with the farmers.

Consumer-Centric USPs

Diverse Product Range

AgroNexumLive offers consumers an unparalleled variety of agricultural products sourced directly from verified growers. From fresh seasonal produce to artisanal goods, the platform provides access to a diverse and curated selection, ensuring consumers find exactly what they seek while discovering new, high-quality options.

Streamlined Communication

With the simplicity of a single click, consumers and buyers gain immediate access to the contact details of the farmer or grower. This user-friendly feature facilitates swift and direct communication, enhancing the overall convenience of connecting with agricultural producers. Whether inquiring about specific produce or establishing partnerships, this one-click accessibility ensures a seamless and efficient exchange between buyers and farmers, fostering a more connected and dynamic agricultural marketplace.

Technical Stack

Backend Framework

- **Python Flask**

Employed as the primary backend framework, Flask provides a robust and flexible foundation for developing the application's server-side logic. Its lightweight nature and extensive libraries make it an ideal choice for building scalable web applications.



Frontend Technologies

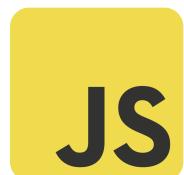
- **HTML, CSS, JavaScript (JS)**

These fundamental web technologies form the core of the user interface. HTML structures the content, CSS styles the presentation, and JavaScript enables dynamic interactions and functionalities.



- **Bootstrap**

Leveraging the power of Bootstrap, a front-end framework, ensures a responsive and visually appealing design. Its grid system and pre-built components expedite the development process while maintaining consistency across different devices.



- **MySQL**

Chosen as the database management system, MySQL efficiently stores and manages the application's data. Its reliability, speed, and scalability make it a popular choice for web-based applications.



- **phpMyAdmin**

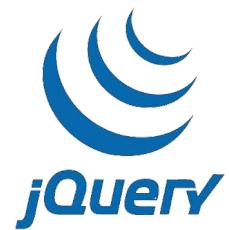
Used as a graphical interface to manage MySQL databases, phpMyAdmin simplifies database operations such as querying, managing tables, and performing backups.



Third-Party Integrations

- **jQuery**

Integrated for simplifying client-side scripting and enhancing the interactivity of the web pages. jQuery streamlines tasks like DOM manipulation and event handling.



- **XAMPP (Apache and MySQL)**

Utilized for local development environments, XAMPP provides Apache as the web server and MySQL as the database server, ensuring a unified platform for testing and development.



Software Requirements

Frontend: HTML, CSS, Java Script, Bootstrap

Backend: Python flask (Python 3.7) , SQLAlchemy

- Operating System: Windows 10
- Google Chrome/Internet Explorer
- XAMPP (Version-3.7)
- Python main editor (user interface): PyCharm Community
- workspace editor: Sublime text 3

Hardware Requirements

- Computer with a 1.1 GHz or faster processor
- Minimum 2GB of RAM or more
- 2.5 GB of available hard-disk space
- 5400 RPM hard drive
- 1366 × 768 or higher-resolution display
- DVD-ROM drive

Features and Functionality

For Farmers

1. Efficient Listing System

- Streamlined product listing process using HTML, CSS, and JavaScript for user-friendly interfaces.
- Database integration (MySQL) for managing and updating product information.

2. Customized Profiles

- Detailed agrarian profiles using HTML, CSS, and Bootstrap to highlight expertise and produce specifics.
- Python Flask for managing profile details and integrating with the database.

3. Inventory Management

- Tools using PHP for tracking inventory and managing product availability.
- Integration with MySQL database and PHPMyAdmin for handling inventory data.

4. Product Showcase

- Bootstrap for creating visually appealing product displays and showcasing produce details.
- JavaScript for interactive product displays and image galleries.

5. Password Encryption

- The system implements password encryption mechanisms using PHP and encryption libraries for heightened security. This feature is common for both consumers and growers and is applied during the signup process.

For Consumers

1. Diverse Product Range

- HTML, CSS, and JavaScript to display and categorize a wide variety of agricultural products.
- Bootstrap for responsive and visually engaging product presentation.

2. Secure Transactions

- PHP for implementing a secure payment gateway.
- MySQL database integration for handling transaction data securely.

3. Product Information

- Detailed product information using HTML, CSS, and JavaScript.
- Utilization of jQuery for dynamic content updates and enhanced user experience.

4. Search and Filtering

- Implementing search functionalities using JavaScript and MySQL for product filtering.
- Utilization of PHP for handling search queries and displaying relevant results.

Mobile-Friendly Experience

AgroNexumLive prioritizes mobile usability, offering a seamless interface tailored for smartphones and tablets. This ensures consistent access and functionality across devices.

Key Elements

- Responsive Design: Adapts effortlessly to various screen sizes, guaranteeing content clarity and functionality.
- Intuitive Navigation: Simplified touch-based controls for effortless browsing on smaller screens.
- Efficient Performance: Swift loading times and optimized data usage for faster access to information.
- Visual Consistency: Uniform branding and layout for easy recognition on any device.

Benefits

- Enhanced Accessibility: Convenient engagement and product management on mobile devices.
- Improved Engagement: Consistent functionality fosters continued usage and interaction.
- Expanded Reach: Connects users from diverse settings, ensuring access from anywhere.

Login through OTP

- Enable quick account access by recovering forgotten passwords with OTP login.
- Send a one-time password (OTP) to the user's registered email for secure verification.
- Users receive the OTP in their email inbox, simplifying the login process.
- Introduce an additional security layer to mitigate risks associated with forgotten passwords.
- Provide a swift and secure pathway for users to log in, ensuring an efficient user experience.

User Experience

Farmer-Centric Experience

1. Effortless Product Listing

- Simplified listing process employing HTML, CSS, and JavaScript for a user-friendly interface.
- Intuitive form layouts and validation to ensure smooth data entry and product details submission.

2. Streamlined Profile Management

- Customized farmer profiles designed with HTML, CSS, and Bootstrap, emphasizing usability and clarity.
- Python Flask framework for managing profile data, ensuring ease of updating and editing.

3. Inventory Management Simplified

- Utilization of PHP and MySQL for inventory tracking, enhancing accessibility and control.
- Integration with PHPMyAdmin for efficient database management.

Consumer-Focused Design

1. Visually Engaging Product Displays

- Responsive and visually appealing product showcases utilizing HTML, CSS, and JavaScript.
- Bootstrap for creating attractive layouts, improving user engagement and product visibility.

2. Enhanced Product Information

- Detailed and well-organized product descriptions using HTML, CSS, and JavaScript for easy comprehension.
- Dynamic content updates using jQuery, enhancing user interaction and understanding.

3. Efficient Search and Filtering

- Seamless search functionalities facilitated by JavaScript and MySQL, simplifying product exploration.
- PHP integration for robust search handling, ensuring quick and relevant results.

Contents and working of AgroNexumLive

Home Page

- Default Page: Home
 - A visually appealing landing page to welcome users.
 - Navigation Links:
 - Home
 - Agrarian Register
 - Add Produce Type
 - Agrarian Details
 - Records Created

User Entities

1.Admin

- Full access to the web app.
- Can view detailed records and available products.
- Ability to edit agrarian details and add listings.

2.Farmer

- Access to view personal agrarian details.
- Capability to add detailed listings for agricultural products.
- Flexibility to edit their own profile information.

3.Consumer

- Limited access, allowing the viewing of available product listings.
- Functionality to contact farmers for specific products.

4.Unlogged User

- Basic access to view available products.
- Required to sign up or log in for contacting farmers.

Sign Up Page

- User details collection includes:
 - Username
 - Email address
 - Password

- Checkbox for identifying whether the user is a farmer or consumer.
- Differential privileges granted based on the selected checkbox.
- A personalized greeting email sent upon successful signup.

Login Page

- Users input their email address and password for authentication.
- A welcome-back email is dispatched after every successful login.
- Option for OTP login in case of a forgotten password.

Agrarian Register Page

- Farmers provide the following details:
 - Name
 - Aadhar number
 - Age
 - Gender
 - Contact number
 - Mail address (matching the signup details)
 - Registration for each type of produce they handle.

Add Produce Type Page

- Allows farmers to specify their agricultural focus.
- Provides the ability to add custom product types.

Agrarian Details Page

- For Logged-In Agrarian:
 - Comprehensive view of personal details.
 - Capability to add, edit, or delete product listings.
 - Options to edit or delete their profile.
- For Admin:
 - Access to view all agrarian details.

Available Products Page

- Products displayed in an aesthetically pleasing manner.
- Listing order based on the time of insertion.
- The "Purchase" button redirects to an email with the farmer's contact address for further communication.

Communication Channels for AgroNexumLive

Email Communication



- An authentic email ID(agonexumlive@gmail.com) has been established for AgroNexumLive, facilitating official communication. This email is utilized for sending notifications, updates, and other relevant information to users.

WhatsApp Groups



AgroNexumLive Prospects' Portal:

- A dedicated WhatsApp group for prospects and potential users of AgroNexumLive. This group serves as a platform for communication, inquiries, and discussions related to the application.

AgroNexumLive Agrarian's Portal:

- Another WhatsApp group specifically designed for agrarians associated with AgroNexumLive. This group fosters communication among farmers, allowing them to discuss, share experiences, and coordinate on agricultural matters.

Announcement Group:

- An announcement-focused WhatsApp group serves as a channel for posting updates, announcements, and important information regarding the AgroNexumLive application. This ensures that users stay informed about any developments or changes in the app.

Database

A Database Management System (DBMS) is specialized computer software designed to efficiently manage large sets of structured data and execute operations on the data as requested by multiple users. Prominent examples of DBMSs include Oracle, DB2, Microsoft Access, Microsoft SQL Server, Firebird, PostgreSQL, MySQL, SQLite, FileMaker, and Sybase Adaptive Server Enterprise. Database administrators commonly utilize DBMSs to create and maintain Database systems, with applications spanning areas such as accounting, human resources, and customer support systems.

DBMSs, once exclusive to large enterprises with the necessary hardware, have now become a standard component in the back-office operations of companies. The complexity of a DBMS lies in its comprehensive set of software programs that govern the organization, storage, management, and retrieval of data within a database. Key components of a DBMS include:

Modeling Language

- Defines the schema of each database within the DBMS, adhering to the DBMS data model.
- The prevailing model, often embedded in SQL, is widely used despite criticisms from relational model purists.

Data Structures

- Optimized structures such as fields, records, files, and objects designed to handle large datasets stored on permanent storage devices.
- Access to data in these structures may be relatively slower compared to volatile main memory.

Query Language and Report Writer

- Employs a database query language to interactively interrogate, analyze, and update the database.
- Supports user privileges on data, ensuring secure and controlled access.

DBMSs also frequently support the Open Database Connectivity API, offering a standardized way for programmers to access the DBMS. This adaptability enhances practicality and performance, even though it may deviate from some fundamental relational model principles. The inclusion of a robust data query language and report writer allows users to interact with the database efficiently, facilitating analysis and updates in accordance with their data access privileges.

Data Security

- Prevents unauthorized users from accessing or modifying the database.
- Access is controlled through passwords, with users having permission for the entire database or specific subsets known as sub-schemas.
- Example: An employee database may contain all employee data, but certain users are restricted to viewing only payroll information, while others have access to work history and student data.

Interactive Database Management

- If the DBMS supports interactive data entry, updates, and queries, it allows for managing personal databases.
- However, lacking an audit trail or necessary controls, this capability might not be suitable for multi-user organizations.
- Comprehensive controls are typically achieved through customized application programs for each data entry and updating function.

Transaction Mechanism (ACID Properties)

- Ensures data integrity, even with concurrent user accesses (concurrency control) and faults (fault tolerance).
- Maintains the integrity of the database by preventing multiple users from updating the same record simultaneously.
- Unique index constraints, such as avoiding duplicate records, contribute to data integrity.

Redundancy Avoidance

- The DBMS helps maintain database integrity by enforcing constraints, such as unique indexes, to prevent the entry of duplicate records.
- For example, it ensures that no two customers with the same customer numbers (key fields) can be entered into the database.

Adaptability to Information Changes

- Organizations can easily adapt information systems to changing requirements with the use of a DBMS.
- Different types of DBMS may be employed for daily transaction processing and analytical tasks, allowing for flexibility in system design.
- Data administrators and systems analysts make overarching systems design decisions, while detailed database design is handled by database administrators.

SQL (Structured Query Language)

Structured Query Language (SQL) serves as the primary language for manipulating relational databases, closely aligned with the relational model. In the relational model, data is organized into structures known as relations or tables. SQL statements are crafted for various purposes within the context of the Farm Management System:

Data Definition (DDL)

- Involves defining tables and structures in the database.
- DDL is used to create, alter, and drop schema objects such as tables and indexes.

Data Manipulation (DML)

- Encompasses actions that interact with data within the tables.
- Common DML operations include SELECT (retrieving data), INSERT (adding new records), UPDATE (modifying existing records), and DELETE (removing records).

Data Retrieval

- SQL facilitates efficient data retrieval using SELECT statements, allowing users to query and retrieve specific information from the database.

Data Filtering and Sorting

- SQL enables the application of filters and sorting criteria to refine query results.
- WHERE clause filters data based on specified conditions, and ORDER BY clause sorts the results.

Data Aggregation

- SQL supports aggregate functions like SUM, AVG, COUNT, MIN, and MAX for summarizing and analyzing data across multiple records.

Data Joins

- Enables the combination of data from multiple tables using JOIN operations.
- Different types of joins, such as INNER JOIN and LEFT JOIN, allow for merging data based on specified relationships.

Data Integrity Enforcement

- SQL ensures data integrity through constraints like PRIMARY KEY, FOREIGN KEY, UNIQUE, and CHECK, maintaining the accuracy and consistency of the data.

Transactional Control

- SQL includes commands like COMMIT and ROLLBACK to manage transactions, ensuring the integrity of the database during complex operations.

Index Management

- Allows the creation and management of indexes to enhance query performance by providing faster access to specific data.

User Authorization and Security

- SQL provides mechanisms for defining user roles, granting permissions, and enforcing security measures to control access to the database.

Stored Procedures and Functions

- Supports the creation of stored procedures and functions, allowing the encapsulation of logic for reuse and efficiency.

Views

- SQL allows the creation of views, which are virtual tables derived from the result of a SELECT query. Views simplify complex queries and provide an abstraction layer for users.

Triggers

- Triggers in SQL are predefined actions that are automatically executed in response to specific events, such as INSERT, UPDATE, or DELETE operations. Triggers enhance data integrity and enforce business rules.

Data Backup and Recovery

- SQL offers mechanisms for backing up and restoring databases, ensuring data resilience and recoverability in case of failures.

SQL, as the cornerstone of relational database management, empowers the Grower's Marketplace Management System with a versatile set of tools for effective data manipulation, retrieval, and system management.

An Overview of our database design

Tables

1.addagroproducts

- Columns: 6 columns (username, email, pid, productname, productdesc, price)
- Primary Key: pid
- Auto-incremented: pid
- Index: idx_username_addagroproducts on username

2.farming

- Columns: 2 columns (fid, farmingtype)
- Primary Key: fid
- Auto-incremented: fid

3.register

- Columns: 8 columns (rid, farmername, adharnumber, age, gender, phonenumbers, address, farming)
- Primary Key: rid
- Auto-incremented: rid
- Index: idx_adharnumber_register on adharnumber

4.test

- Columns: 2 columns (id, name)
- Primary Key: id
- Auto-incremented: id

5.trig

- Columns: 4 columns (id, fid, action, timestamp)
- Primary Key: id
- Auto-incremented: id
- Index: idx_fid_trig on fid

6.user

- Columns: 6 columns (id, username, email, password, is_agrarian, is_admin)
- Primary Key: id
- Auto-incremented: id

Auto-increment Information

- addagroproducts: pid is auto-incremented.
- farming: fid is auto-incremented.
- register: rid is auto-incremented.
- test: id is auto-incremented.
- trig: id is auto-incremented.
- user: id is auto-incremented.

Index Information

- addagroproducts: Index on username (idx_username_addagroproducts).
- farming: No additional index mentioned.
- register: Index on adharnumber (idx_adharnumber_register).
- test: No additional index mentioned.
- trig: Index on fid (idx_fid_trig).
- user: No additional index mentioned.

Views

- addagroproducts_view: View for addagroproducts selecting columns pid, productname, productdesc, and price.
- register_view: View for register selecting columns farmername, adharnumber, phonenumbers, and farming.

Functions

- farming_count: Function that retrieves the total number of registered farmers for a given farming type.
- average_age_for_farming_type: Function that calculates the average age of farmers based on a given farming type.

Stored Procedures

- calculate_total_value: Procedure that calculates and returns the total value of all products for a given user in addagroproducts.
- get_farmers_by_age_and_type: Procedure that retrieves the details of farmers who have a specified minimum age and are associated with a specific farming type.

Triggers

Deletion Trigger

- BEFORE DELETE ON register
- Logs "Agrarian DELETED" with timestamp in trig.

Insertion Trigger

- AFTER INSERT ON register
- Logs "Agrarian Inserted" with timestamp in trig.

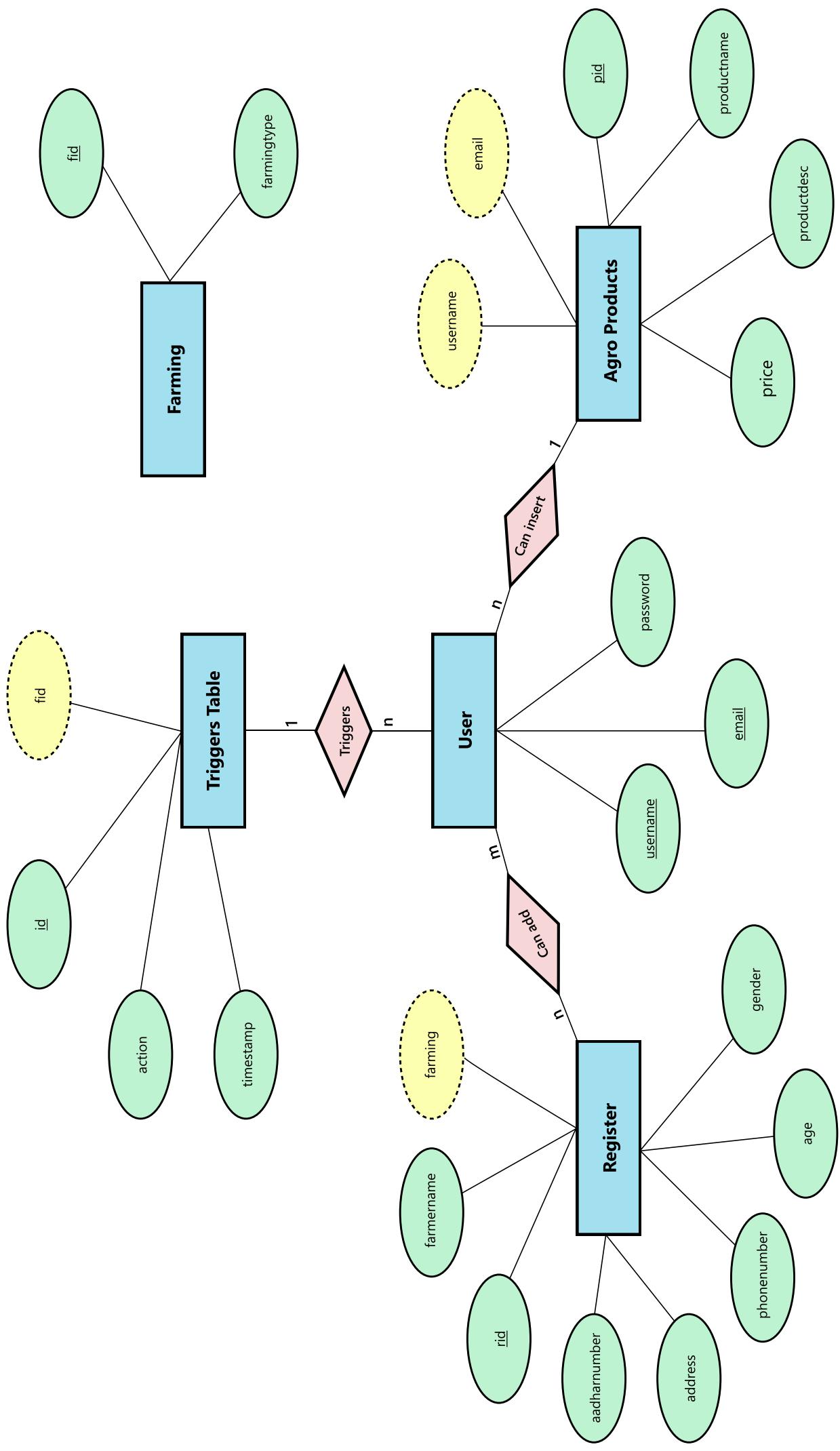
Updation Trigger

- AFTER UPDATE ON register
- Logs "Agrarian UPDATED" with timestamp in trig.

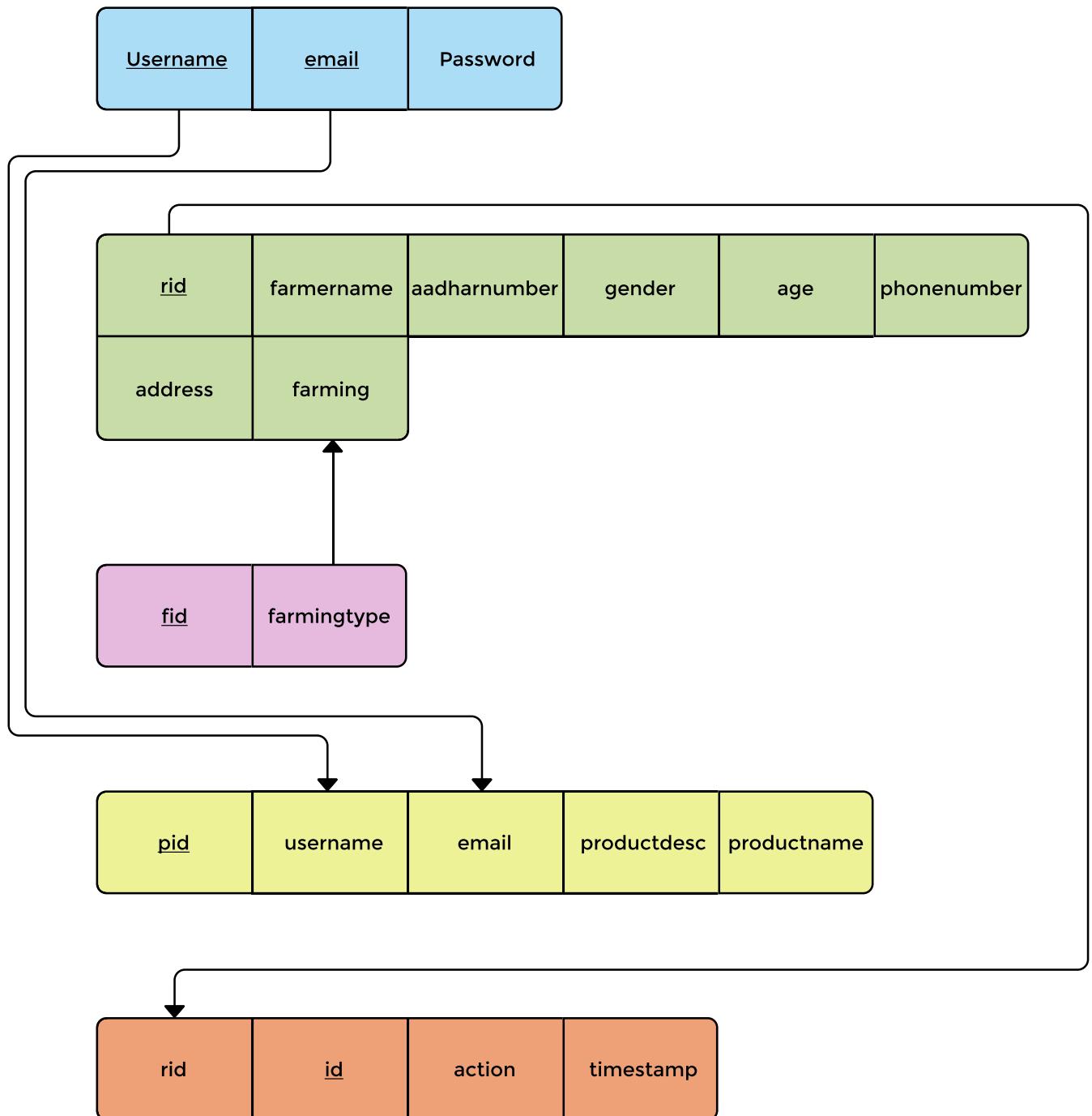
Summary

- Number of Tables: 6
- Number of Views: 2
- Number of Functions: 2
- Number of Stored Procedures: 2
- Number of Triggers: 3

Conceptual Design: ER Diagram



Schema Diagram



Future Development and Expansion

Scalability and Technological Adaptability

1. Scalable Infrastructure

- Consideration of cloud-based solutions (AWS, Azure) for enhanced scalability and performance.
- Integration of Docker containers for streamlined deployment and scaling based on demand.

2. Adoption of Advanced Frameworks

- Explore the integration of React.js or Vue.js for frontend enhancements, ensuring better interactivity and responsiveness.
- Consider transitioning backend functionalities to Node.js for a more unified JavaScript environment.

Enhanced User Experience and Features

1. Mobile Application Development

- Plan and develop mobile applications (iOS, Android) to expand market reach and cater to on-the-go users.
- Ensure feature parity with the web platform to maintain a consistent user experience across devices.

2. Real-time Analytics and Reporting

- Implement analytics tools (Google Analytics, custom dashboards) for data-driven insights.
- Provide agrarian users with detailed sales reports, aiding in decision-making and market analysis.

Market Expansion and Community Building

1. Regional Expansion

- Strategize market expansion to encompass different geographical regions or global markets.
- Localize the platform by offering multi-language support to engage a wider user base.

2. Community Forums or Knowledge Hub

- Introduce a knowledge-sharing section for farmers, enabling discussions, tips, and best practices.
- Foster a sense of community by organizing webinars or virtual events centered around agricultural advancements.

Security and Privacy Enhancements

Advanced Security Measures

- Implement additional layers of security protocols such as OAuth for user authentication and authorization.
- Regular security audits and updates to ensure robust protection against emerging threats.

Privacy-Centric Features

- Offer granular privacy controls to farmers, enabling them to manage their data visibility.
- Ensure compliance with data protection regulations (GDPR, CCPA) for consumer data handling.

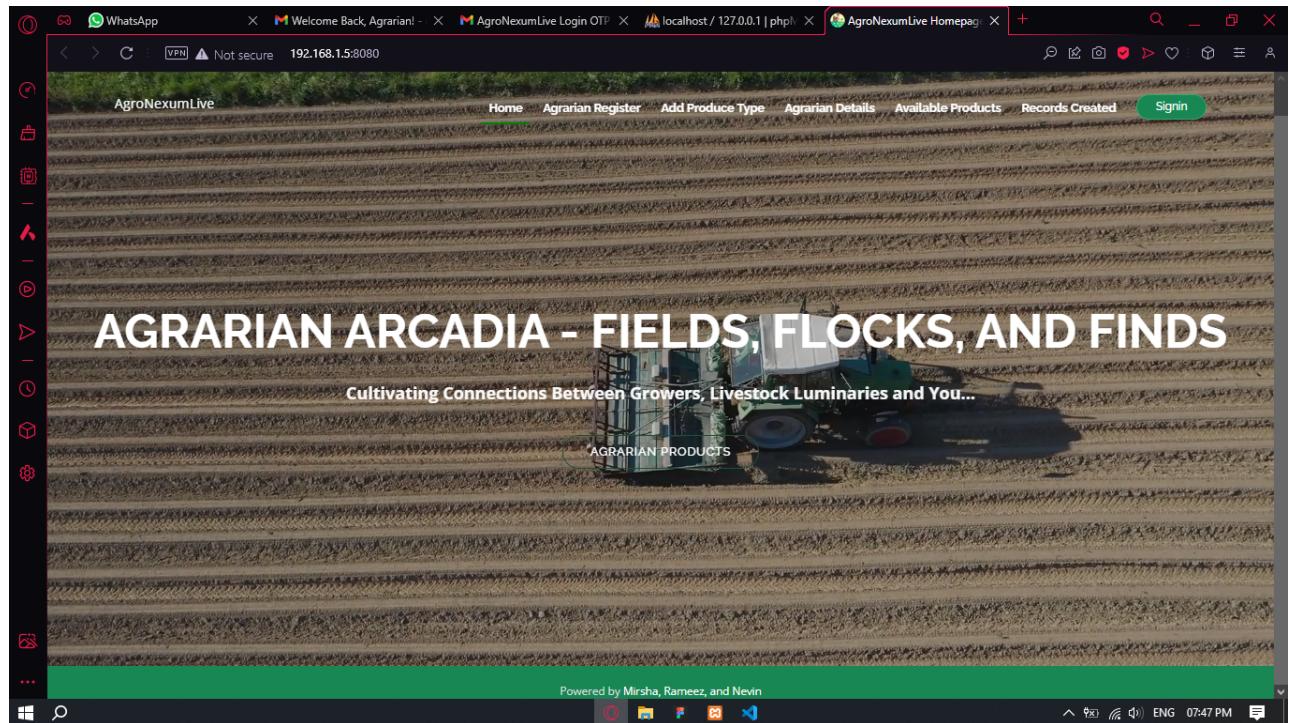
Conclusion

The trajectory of AgroNexumLive propels towards a future steeped in innovation and adaptability. Our roadmap aligns with the relentless pursuit of technological advancements, ensuring that our platform remains at the forefront of agricultural digitalization. We are committed to refining user experience, streamlining accessibility across devices, and fortifying our security infrastructure to uphold trust and reliability.

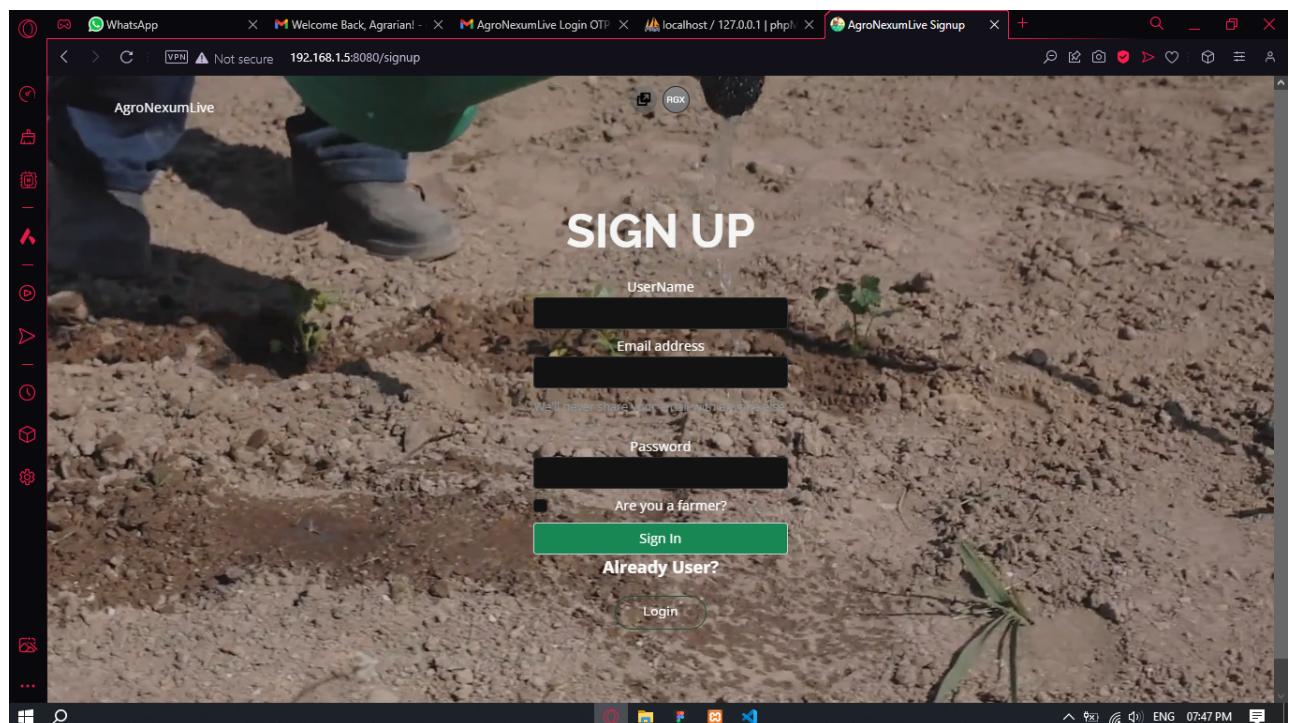
Our vision extends beyond the present, focusing on market expansion while catering to the evolving demands of farmers and consumers. We envisage AgroNexumLive not just as a solution but as a dynamic, scalable entity poised to revolutionize agricultural ecosystems. With a strategic blend of technological sophistication and user-centric design, we aspire to cement our position as the go-to platform in the ever-evolving agricultural landscape, offering unparalleled value and resilience.

Output

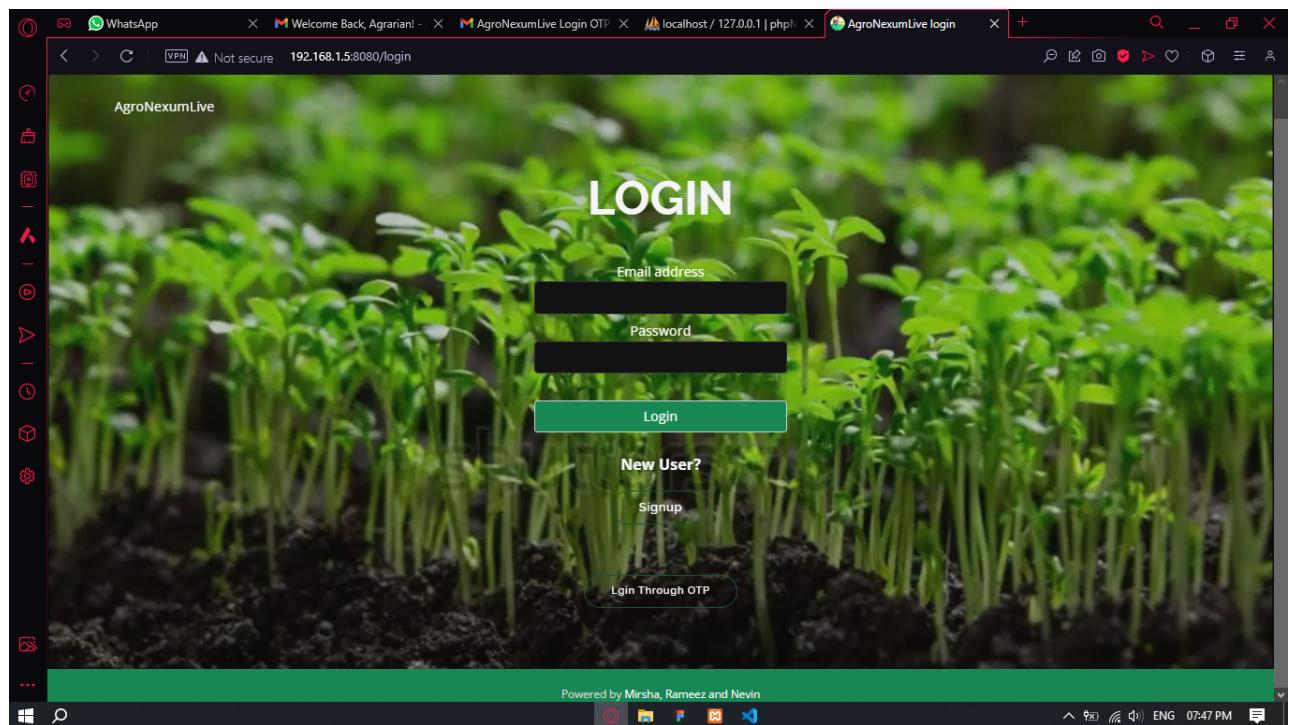
Home page



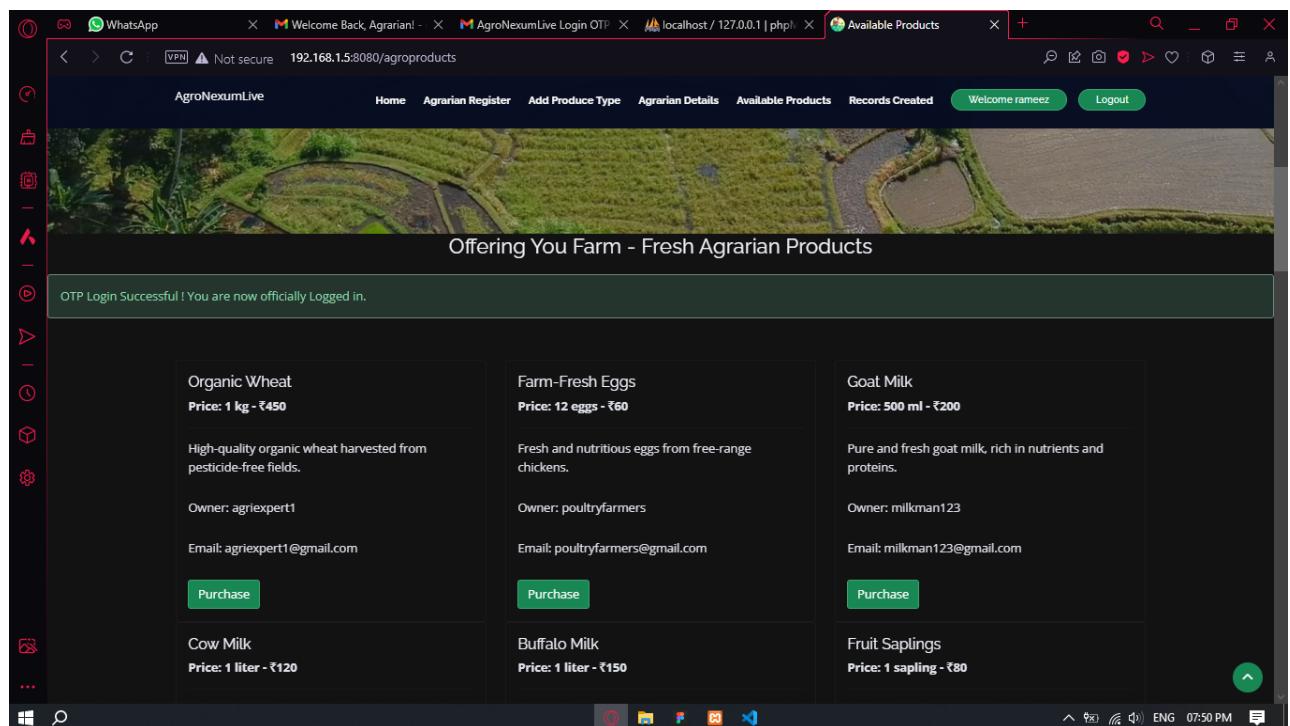
Sign Up Page



Login page



Available Products page



Add Produce Type page

The screenshot shows a web browser window with the URL 192.168.1.5:8080/addfarming. The page title is "Add Agricultural Focus". The header includes links for Home, Agrarian Register, Add Produce Type (which is highlighted in green), Agrarian Details, Available Products, and Records Created, along with a "Logout" button. The main content features a large image of a tractor in a field with the text "AGRARIAN ARCADIA - FIELDS, FLOCKS, AND FINDS" and "Cultivating Connections Between Growers, Livestock Luminaries and You...". Below this is a green button labeled "AGRARIAN PRODUCTS". A form field labeled "Enter Agriculture Focus" is present, with a placeholder "Agriculture Focus" and a "Save Record" button. The status bar at the bottom shows the URL again and the system time as 07:48 PM.

Register Agrarian Details Page

The screenshot shows a web browser window with the URL 192.168.1.5:8080/register. The page title is "Register Agrarian Details". The header includes links for Home, Agrarian Register, Add Produce Type, Agrarian Details, Available Products, and Records Created, along with a "Logout" button. The main content features a "Login Success" message in a blue box. Below it is a form with fields for "Agrarian Name", "Agrarian Adhaar Number", "Agrarian Age", "Gender Selection", "Contact Number", "Agrarian's Address", and "Specify Agricultural Focus". A "Save Record" button is located at the bottom of the form. The status bar at the bottom shows the URL again and the system time as 07:48 PM.

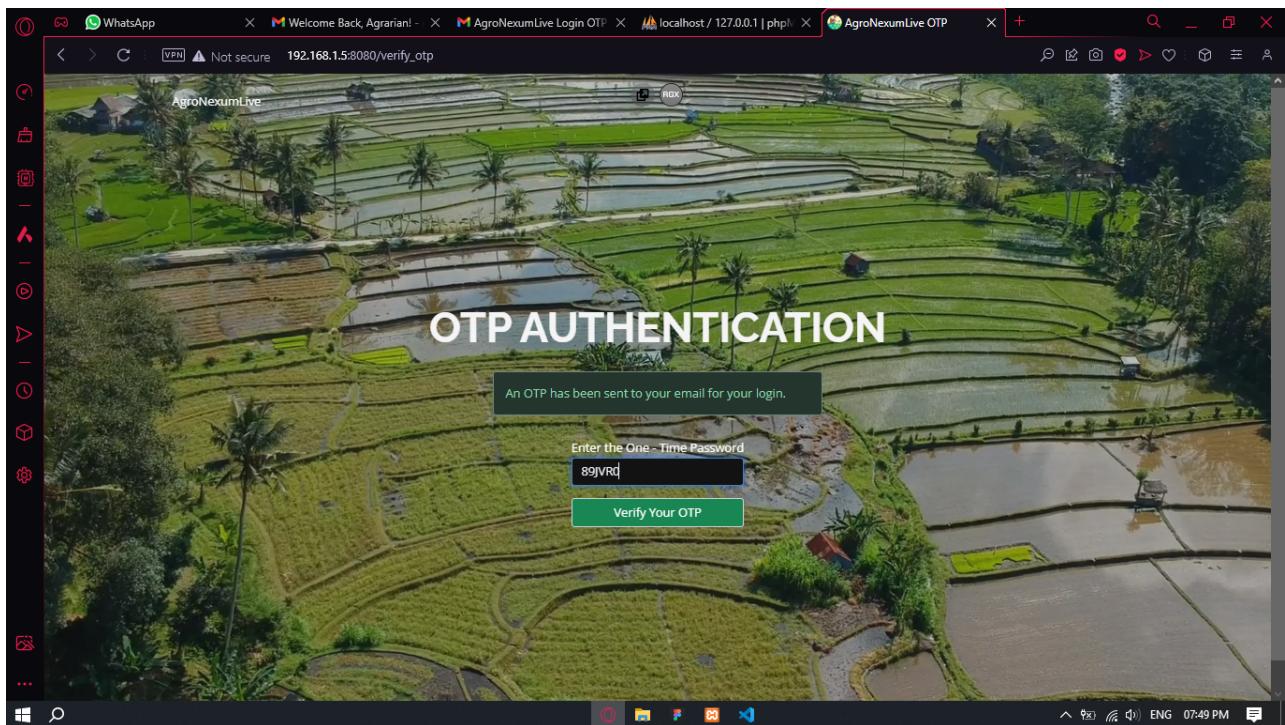
Agrarian Details Page

The screenshot shows a web browser window with the URL 192.168.1.5:8080/farmerdetails. The page title is "Agrarian Details". The header includes links for "Home", "Agrarian Register", "Add Produce Type", "Agrarian Details", "Available Products", and "Records Created", along with "Welcome rameez2" and "Logout" buttons. The main content features a large image of a tractor in a field with the text "AGRARIAN ARCADIA - FIELDS, FLOCKS, AND FINDS" and "Cultivating Connections Between Growers, Livestock Luminaries and You...". Below this is a section titled "AGRARIAN PRODUCTS". A table displays the following data:

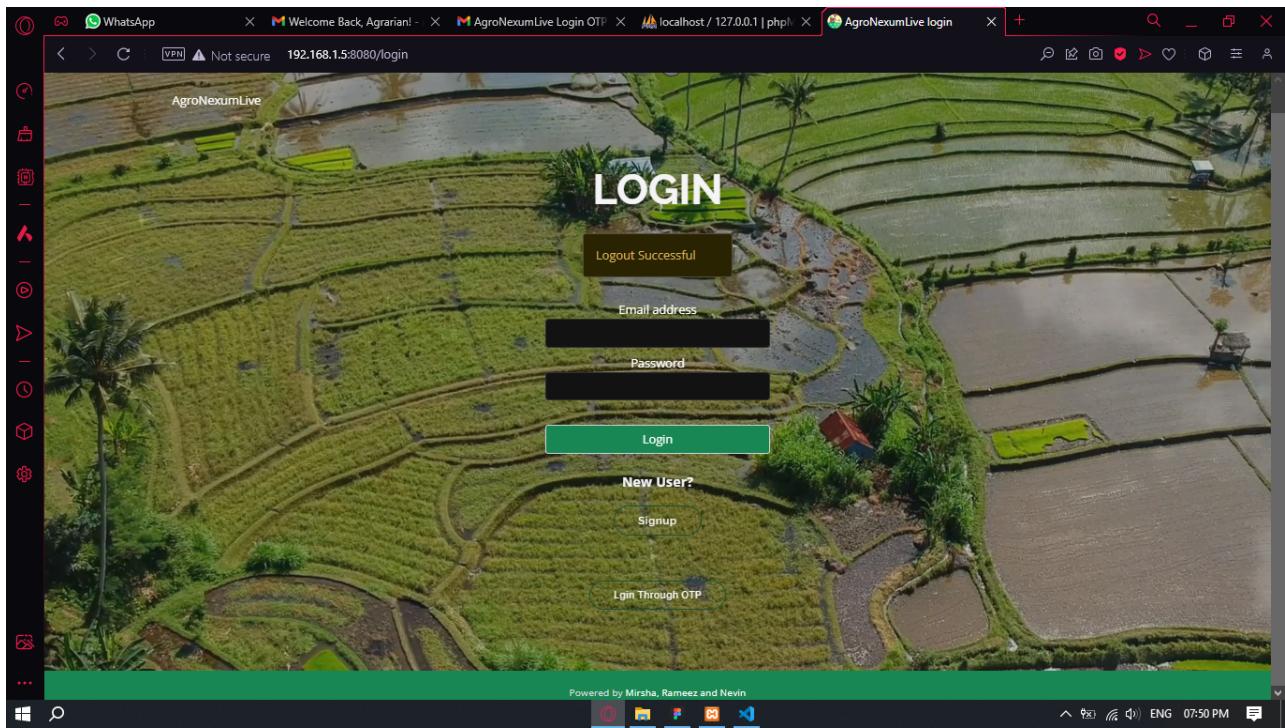
REGISTERED NUMBER	AGRARIAN NAME	ADHAAR	AGE	GENDER	CONTACT	ADDRESS	AGRICULTURAL FOCUS	EDIT	DELETE	ADD AGRO PRODUCT
31	rameez2	495834872903	20	Male	9384783243	ramzymit@gmail.com	Tapioca	<button>Edit</button>	<button>Delete</button>	<button>ADD</button>

Login Through OTP Pages

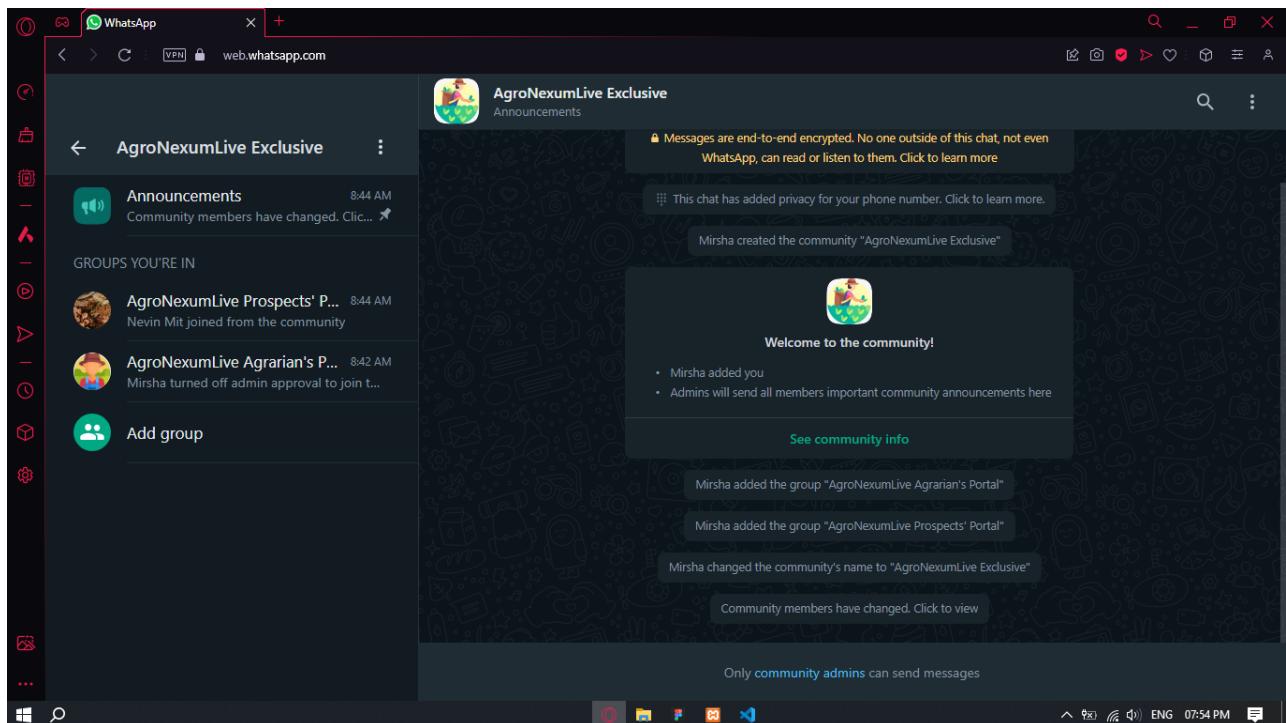
The screenshot shows a web browser window with the URL 192.168.1.5:8080/forgot_password. The page title is "AgroNexumLive OTP". The header includes links for "Home", "Agrarian Register", "Add Produce Type", "Agrarian Details", "Available Products", and "Records Created", along with "Welcome rameez2" and "Logout" buttons. The main content features a large image of cows grazing in a field with the text "LOGIN THROUGH OTP". Below this is a form with an "Email address" input field and a "Login Through OTP" button. At the bottom, it says "Powered by Mirsha, Rameez and Nevin".



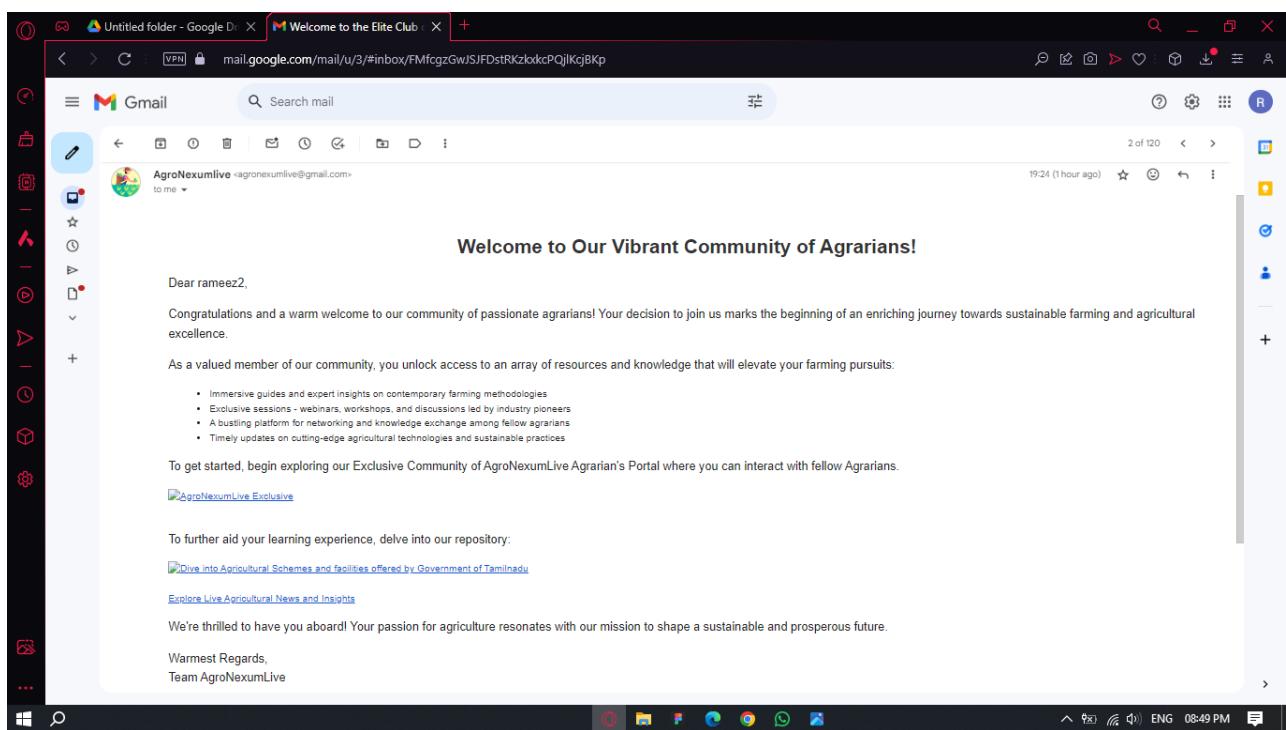
After Logout



What's app community



Mail received for farmer at the time of signup



Mail Received at the time of login as agrarian

The screenshot shows a Gmail inbox with one unread email. The subject of the email is "Welcome Back, Agrarian!". The sender is "AgroNexumlive <agronexumlive@gmail.com>" and it was sent "to me" 19-24 (1 hour ago). The email body starts with "Glad to See You Once Again!" and expresses appreciation for the recipient's unwavering commitment to agriculture. It includes links to "Partake in Agricultural Visuals" and "Explore Live Agricultural News". The message concludes with "Kindest Regards With Warmth, Team AgroNexumLive".

Mail received at the time of sign up as consumer

The screenshot shows a Gmail inbox with one unread email. The subject of the email is "Welcome to the Elite Club!". The sender is "AgroNexumlive <agronexumlive@gmail.com>" and it was sent "to me" Sun, 31 Dec 2023, 11:42 (1 day ago). The email body starts with "Welcome to Our Vibrant Agrarian Community!" and congratulates the recipient on joining the community. It states that as a valued member, they unlock access to an array of resources and knowledge that will elevate their farming pursuits. A bulleted list follows, detailing the benefits:

- Immersive guides and expert insights on contemporary farming methodologies
- Exclusive sessions - webinars, workshops, and discussions led by industry pioneers
- A bustling platform for networking and knowledge exchange among fellow agrarians
- Timely updates on cutting-edge agricultural technologies and sustainable practices

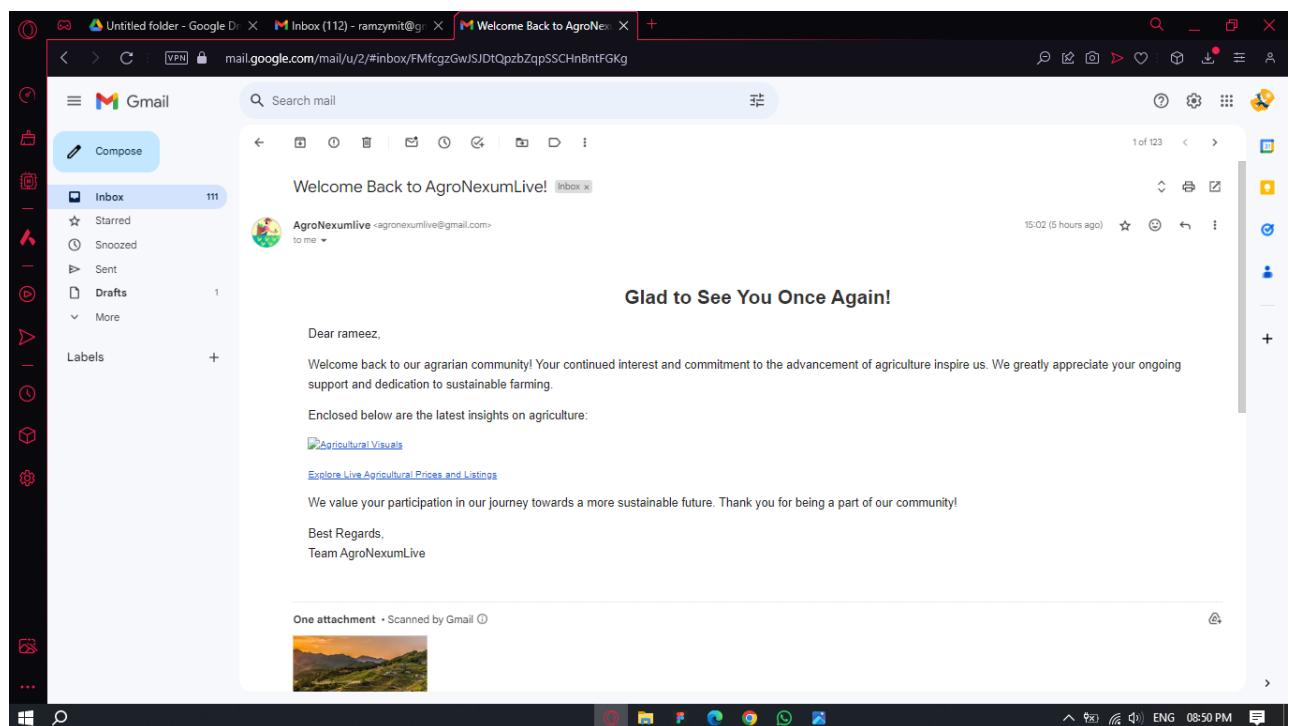
To further aid your learning experience, delve into our repository:

[Dive into Agricultural Schemes and facilities offered by Government of Tamilnadu](#)
[Explore Live Agricultural News and Insights](#)

We're thrilled to have you aboard! Your passion for agriculture resonates with our mission to shape a sustainable and prosperous future.

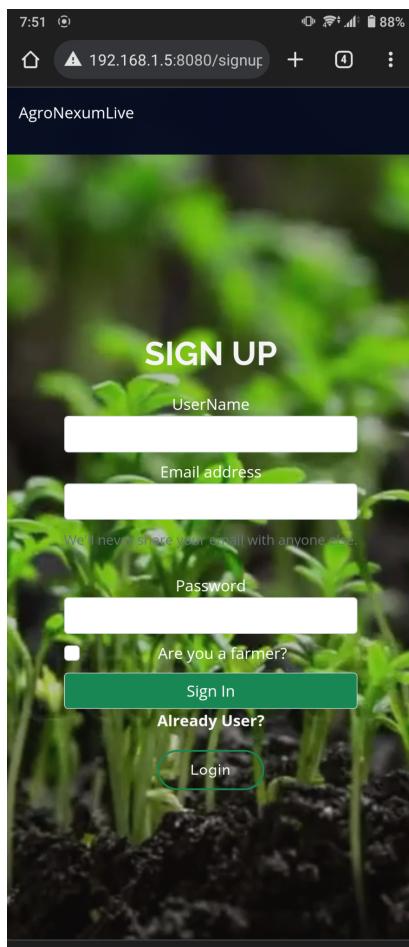
Warmest Regards,
Team AgroNexumLive

Mail received at the time of login as consumer

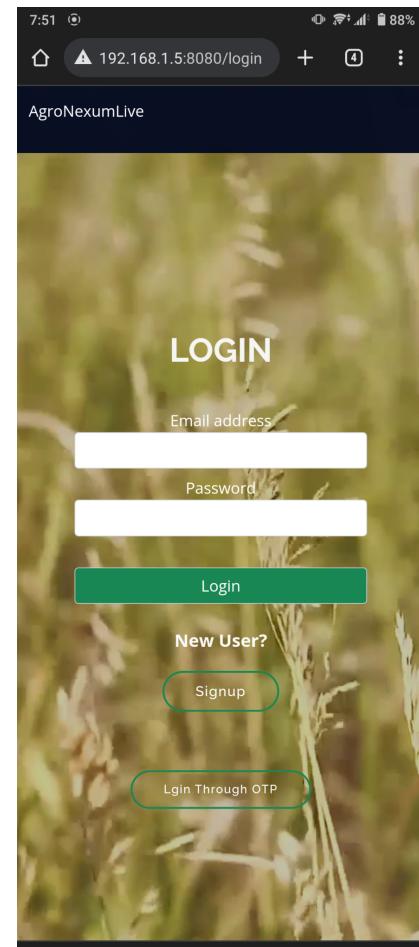


Mobile View

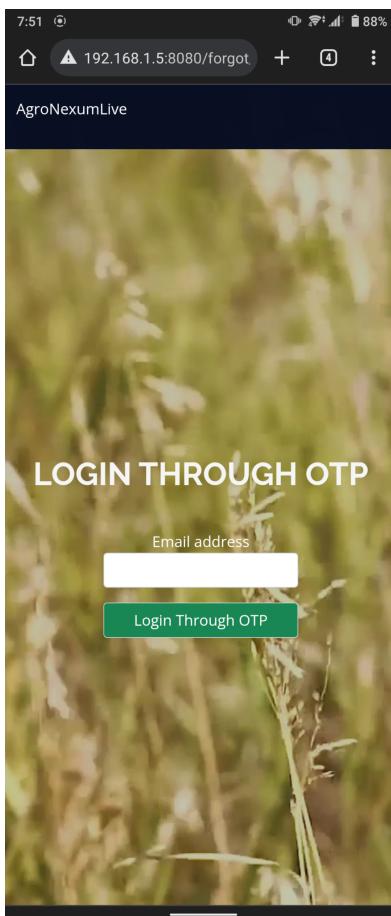
Sign Up Page



Login Page



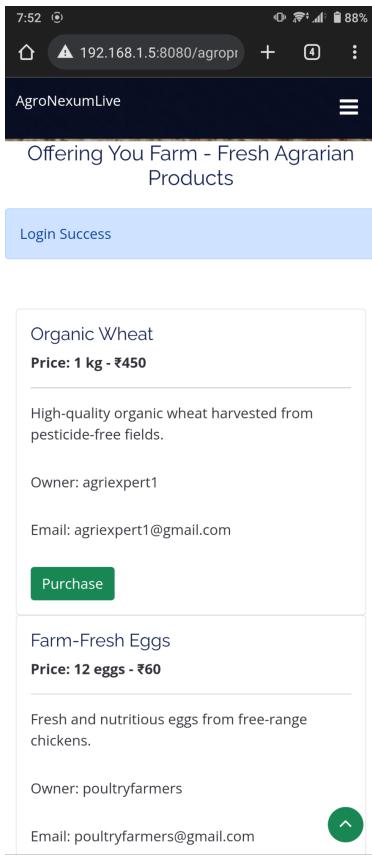
Login through OTP



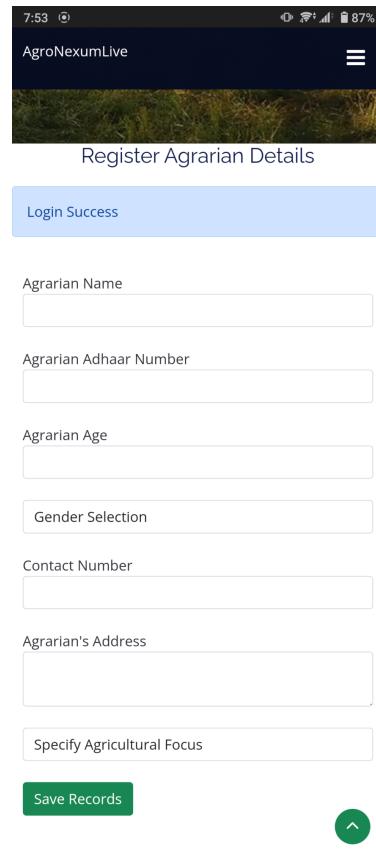
Home page



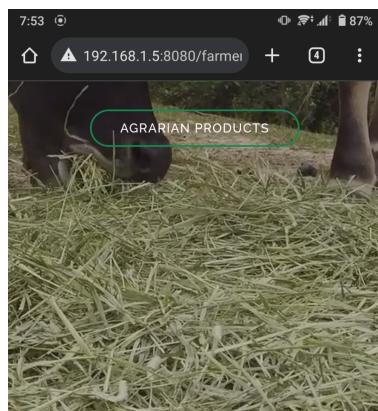
Available product



Agrarian Register



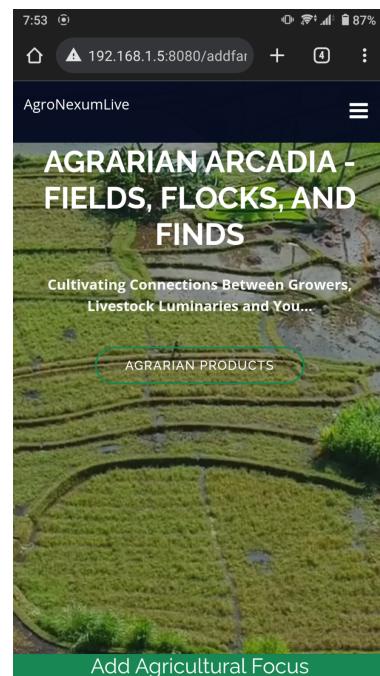
Agrarian's Details



Agrarian Details

REGISTERED NUMBER	AGRARIAN NAME	ADHAAR	AGE	G
31	rameez2	495834872903	20	M

Add product type



Add Agricultural Focus

Enter Agriculture Focus

Add Agriculture Category

^

Logout



Source Code



Output video



Source Code

HTML

about.html

```
{% extends 'basic.html' %}  
{% block title %}  
About Us  
{% endblock title %}  
{% block about %}  
menu-active  
{% endblock about %}  
  
{% block body %}  
<h2 class="text-center text-danger">About Us</h2>  
{% endblock body %}
```

addagroproducts.html

```
{% extends 'base.html' %}  
{% block title %}  
Add AgroProducts  
{% endblock title %}  
  
{% block body %}  
<h3 class="text-center"><span>Add Agro Products</span> </h3>  
  
{% with messages=get_flashed_messages(with_categories=true) %}  
{% if messages %}  
{% for category, message in messages %}  
  
<div class="alert alert-{{category}} alert-dismissible fade show" role="alert">  
    {{message}}  
  
</div>  
    {% endfor %}  
    {% endif %}  
    {% endwith %}  
<br>  
<div class="container">  
  
<div class="row">
```

```

<div class="col-md-4"></div>
<div class="col-md-4">

<form action="/addagroproduct" method="post">
<div class="form-group">

<label for="username">Farmer Name</label>
<input type="text" class="form-control" name="username"
value="{{current_user.username}}" id="{{current_user.username}}" required>
</div>
<br>
<div class="form-group">

<label for="username">Farmer Email</label>
<input type="email" class="form-control" name="email"
value="{{current_user.email}}" id="{{current_user.email}}" required>
</div>
<br>
<div class="form-group">

<label for="rollno">Product Name</label>
<input type="text" class="form-control" name="productname" id="productname"
required>
</div>
<br>

<div class="form-group">
<label for="productdesc">Product Description</label>
<textarea class="form-control" name="productdesc" id="productdesc"
required></textarea>
</div>
<br>

<div class="form-group">

<label for="price">Price</label>
<input type="text" class="form-control" name="price" id="price" required>
</div>
<br>

    <button type="submit" class="btn btn-success btn-block">Add Product</button>
</form>
<br>
<br>

</div>

<div class="col-md-4"></div>

```

```
</div></div>
```

```
{% endblock body %}
```

```
agroproducts.html
```

```
{% extends 'base.html' %}  
{% block title %}  
Available Products  
{% endblock title %}
```

```
{% block body %}  
    <h3 class="text-center"><span>Offering You Farm - Fresh Agrarian  
Products</span> </h3>
```

```
{% with messages=get_flashed_messages(with_categories=true) %}  
{% if messages %}  
{% for category, message in messages %}
```

```
<div class="alert alert-{{category}} alert-dismissible fade show" role="alert">  
    {{message}}
```

```
</div>  
    {% endfor %}  
    {% endif %}  
    {% endwith %}  
<br>
```

```
<div class="container mt-3">  
    <div class="row">  
        {% for p in query %}  
            <div class="col-sm-4">  
                <div class="card">  
                    <div class="card-body">  
                        <b><h5 class="card-title">{{p.productname}}</h5></b>  
                        <b>Price: {{p.price}}</b>  
                        <hr>  
                        <p class="card-text">{{p.productdesc}}</p>  
                        <p>Owner: {{p.username}}</p>  
                        <p>Email: {{p.email}}</p>  
                        {% if session.get('logged_in') %}  
                            <a  
                                href="https://mail.google.com/mail/?view=cm&fs=1&tf=1&to={{p.email}}"  
                                target="_blank" class="btn btn-success">Purchase</a>
```

```

        {% else %}
            <a href="{{ url_for('login') }}" class="btn
btn-primary">Log In to Purchase</a>
        {% endif %}
    </div>
</div>
</div>
{% endfor %}
</div>
</div>
{% endblock body %}

```

auth.html

```

<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="utf-8">
    <meta content="width=device-width, initial-scale=1.0" name="viewport">

    <title>{% block title %}</title>
    {% endblock title %}</title>
    <meta content="" name="description">
    <meta content="" name="keywords">

    <!-- Favicon -->
    <link rel="icon" type="image/png" href="/static/images/farmer.png">

    {% block style %}
    {% endblock style %}
    <link
    href="https://fonts.googleapis.com/css?family=Open+Sans:300,300i,400,400i,700,7
00i|Raleway:300,400,500,700,800" rel="stylesheet">

    <!-- Vendor CSS Files -->
    <link href="static/assets/vendor/bootstrap/css/bootstrap.min.css"
rel="stylesheet">
    <link href="static/assets/vendor/venobox/venobox.css" rel="stylesheet">
    <link href="static/assets/vendor/font-awesome/css/font-awesome.min.css"
rel="stylesheet">
    <link href="static/assets/vendor/owl.carousel/assets/owl.carousel.min.css"
rel="stylesheet">
    <link href="static/assets/vendor/aos/aos.css" rel="stylesheet">

    <!-- Template Main CSS File -->
    <link href="static/assets/css/style.css" rel="stylesheet">

```

```

</head>

<body style="margin: 0; padding: 0;"> <!-- Reset margin and padding to remove
white space -->

<!-- ===== Header ===== -->
<header id="header">
  <div class="container">

    <div id="logo" class="pull-left">

      <a href="/" class="scrollto">AgroNexumLive</a>
    </div>

  </div>
</header><!-- End Header -->

<section id="intro" style="position: relative;">
  <video autoplay muted loop id="video-background" width="100%" height="auto"
style="position: absolute; top: 0; left: 0;">
    <source src="/static/images/output.mp4" type="video/mp4">
    Your browser does not support the video tag.
  </video>
  <div class="intro-container" data-aos="zoom-in" data-aos-delay="100">
    {% block content %}
    {% endblock content %}
  </div>
</section>

<div class="main">

  <!-- ===== Footer ===== -->
  <footer id="footer">
    <div class="footer-top">
      <div class="container">

        </div>
      <div class="container">

        <div class="credits">
          <br>
          Powered by <a href="/">Mirsha, Rameez and Nevin</a>
        </div>
      </div>
    </div>
</footer><!-- End Footer -->

  <a href="#" class="back-to-top"><i class="fa fa-angle-up"></i></a>

```

```

<!-- Vendor JS Files -->
<script src="static/assets/vendor/jquery/jquery.min.js"></script>
<script
src="static/assets/vendor/bootstrap/js/bootstrap.bundle.min.js"></script>
<script
src="static/assets/vendor/jquery.easing/jquery.easing.min.js"></script>
<script src="static/assets/vendor/php-email-form/validate.js"></script>
<script src="static/assets/vendor/venobox/venobox.min.js"></script>
<script src="static/assets/vendor/owl.carousel/owl.carousel.min.js"></script>
<script src="static/assets/vendor/superfish/superfish.min.js"></script>
<script src="static/assets/vendor/hoverIntent/hoverIntent.js"></script>
<script src="static/assets/vendor/aos/aos.js"></script>

<!-- Template Main JS File -->
<script src="static/assets/js/main.js"></script>

<!--<script>
    // To ensure the video background covers the entire section
    const videoBackground = document.getElementById('video-background');
    const introSection = document.getElementById('intro');
    videoBackground.style.width = introSection.offsetWidth + 'px';
    videoBackground.style.height = introSection.offsetHeight + 'px';
</script>
-->
<script>
    const video = document.getElementById('video-background');

    // Check if there's a stored time for the video
    if (sessionStorage.getItem('videoTime')) {
        // Set the video time to the stored value
        video.currentTime = parseFloat(sessionStorage.getItem('videoTime'))+7;
    }

    // Event listener to store the current time when the video is paused
    video.addEventListener('pause', function() {
        sessionStorage.setItem('videoTime', this.currentTime);
    });

    // Event listener to handle video end
    video.addEventListener('ended', function() {
        this.currentTime = 0;
        sessionStorage.removeItem('videoTime');
    });
</script>
</body>

</html>

```

base.html

```
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="utf-8">
    <meta content="width=device-width, initial-scale=1.0" name="viewport">

    <title>{% block title %}</title>
    {% endblock title %}
    <meta content="" name="description">
    <meta content="" name="keywords">

    <!-- Favicon -->
    <link rel="icon" type="image/png" href="/static/images/farmer.png">

    {% block style %}<br/>
    {% endblock style %}
    <link href="https://fonts.googleapis.com/css?family=Open+Sans:300,300i,400,400i,700,700i|Raleway:300,400,500,700,800" rel="stylesheet">

    <!-- Vendor CSS Files -->
    <link href="static/assets/vendor/bootstrap/css/bootstrap.min.css" rel="stylesheet">
    <link href="static/assets/vendor/venobox/venobox.css" rel="stylesheet">
    <link href="static/assets/vendor/font-awesome/css/font-awesome.min.css" rel="stylesheet">
    <link href="static/assets/vendor/owl.carousel/assets/owl.carousel.min.css" rel="stylesheet">
    <link href="static/assets/vendor/aos/aos.css" rel="stylesheet">

    <!-- Template Main CSS File -->
    <link href="static/assets/css/style.css" rel="stylesheet">

</head>

<body style="margin: 0; padding: 0;"> <!-- Reset margin and padding to remove extra space -->

    <!-- ===== Header ===== -->
    <header id="header">
        <div class="container">

            <div id="logo" class="pull-left">
                <a href="/" class="scrollto">AgroNexumLive</a>
            </div>
```

```

<nav id="nav-menu-container">
  <ul class="nav-menu">
    <li class="{% block home %}"
        {% endblock home %}"><a href="/">Home</a></li>

    <li><a href="/register">Agrarian Register</a></li>
    <li><a href="/addfarming">Add Produce Type</a></li>
    <li><a href="/farmerdetails">Agrarian Details</a></li>
    <li><a href="/agroproducts">Available Products</a></li>
    <li><a href="/triggers">Records Created</a></li>

    {% if current_user.is_authenticated  %}
      <li class="buy-tickets"><a href="">Welcome
      {{current_user.username}}</a></li>
      <li class="buy-tickets"><a href="/logout">Logout</a></li>
    {% else %}
      <li class="buy-tickets"><a href="/signup">Signin</a></li>

    {% endif %}
  </ul>
</nav><!-- #nav-menu-container -->
</div>
</header><!-- End Header -->

<!-- ====== Intro Section ====== -->
<section id="intro">
  <video autoplay muted loop id="video-background" width="100%" height="auto"
  style="position: absolute; top: 0; left: 0;">
    <source src="/static/images/output.mp4" type="video/mp4">
    Your browser does not support the video tag.
  </video>
  <div class="intro-container" data-aos="zoom-in" data-aos-delay="100">
    <h1 class="mb-4 pb-0">Agrarian Arcadia - Fields, Flocks, and Finds</h1>
    <p class="mb-4 pb-0">Cultivating Connections Between Growers, Livestock
    Luminaries and You...</p>
    <a href="/agroproducts" class="about-btn scrollto">AGRARIAN
    PRODUCTS</a>
  </div>
</section><!-- End Intro Section -->
<main id="main">

  {% block body %}

  {% with messages=get_flashed_messages(with_categories=true) %}
```

```

{% if messages %}
{% for category, message in messages %}

    <div class="alert alert-{{category}} alert-dismissible fade show"
role="alert">
        {{message}}


</div>

{% endfor %}
{% endif %}
{% endwith %}
{% endblock body %}

<i class="fa fa-angle-up"></i></a>

<!-- Vendor JS Files -->
<script src="static/assets/vendor/jquery/jquery.min.js"></script>
<script
src="static/assets/vendor/bootstrap/js/bootstrap.bundle.min.js"></script>
<script
src="static/assets/vendor/jquery.easing/jquery.easing.min.js"></script>
<script src="static/assets/vendor/php-email-form/validate.js"></script>
<script src="static/assets/vendor/venobox/venobox.min.js"></script>
<script src="static/assets/vendor/owl.carousel/owl.carousel.min.js"></script>
<script src="static/assets/vendor/superfish/superfish.min.js"></script>
<script src="static/assets/vendor/hoverIntent/hoverIntent.js"></script>
<script src="static/assets/vendor/aos/aos.js"></script>

<!-- Template Main JS File -->
<script src="static/assets/js/main.js"></script>
<script>
    const video = document.getElementById\('video-background'\);

    // Check if there's a stored time for the video
    if \(sessionStorage.getItem\('videoTime'\)\) {
        // Set the video time to the stored value
        video.currentTime = parseFloat\(sessionStorage.getItem\('videoTime'\)\);
    }

    // Event listener to store the current time when the video is paused
    video.addEventListener\('pause', function\(\) {
        sessionStorage.setItem\('videoTime', this.currentTime\);
    }\);

```

```

// Event listener to handle video end
video.addEventListener('ended', function() {
  this.currentTime = 0;
  sessionStorage.removeItem('videoTime');
});
</script>

</body>

</html>

edit.html

<!doctype html>
<html lang="en">
  <head>
    <!-- Required meta tags -->
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1,
shrink-to-fit=no">

    <!-- Bootstrap CSS -->
    <link rel="stylesheet"
href="https://cdn.jsdelivr.net/npm/bootstrap@4.5.3/dist/css/bootstrap.min.css"
integrity="sha384-TX8t27EcRE3e/ihU7zmQxVncDAy5uIKz4rEkgIXeMed4M0jlfIDPvg6uqKI2x
Xr2" crossorigin="anonymous">

    <title>Edit Registered Agrarian Details</title>
  </head>
  <body class="bg-success">

    <h3 class="text-center"><span>Edit Agrarian Details</span> </h3>

    {% with messages=get_flashed_messages(with_categories=true) %}
    {% if messages %}
    {% for category, message in messages %}

      <div class="alert alert-{{category}} alert-dismissible fade show" role="alert">
        {{message}}
      </div>
    {% endfor %}
    {% endif %}
    {% endwith %}
    <br>
    <div class="container">

```

```

<div class="row">

<div class="col-md-4"></div>
<div class="col-md-4">

<form action="/edit/{{posts.rid}}" method="post">
<div class="form-group">

<label for="rollno">Agrarian Name</label>
<input type="text" class="form-control" name="farmername" id="farmername"
value="{{posts.farmername}} required>
</div>
<br>
<div class="form-group">

<label for="adharnumber">Adhaar Number</label>
<input type="number" class="form-control" name="adharnumber" id="adharnumber"
value="{{posts.adharnumber}} required>
</div>
<br>
<div class="form-group">

<label for="age">Age</label>
<input type="number" class="form-control" name="age" id="age"
value="{{posts.age}} required>
</div>
<br>
<div class="form-group">
<select class="form-control" id="gender" name="gender" required>
    <option selected>{{posts.gender}} </option>

        <option value="Male">Male</option>
        <option value="Female">Female</option>
        <option value="Prefer Not to Say">Prefer Not to Say</option>
        <option value="Non-Binary">Non-Binary</option>
        <option value="Other">Other</option>

    </select>
</div>
<br>

<div class="form-group">
<label for="num">Contact Number</label>
<input type="number" class="form-control" name="phonenumbers" id="phonenumbers"
value="{{posts.phonenumbers}} required>
</div>
<br>

<div class="form-group">

```

```

<label for="address">Agrarian's Address</label>
<input class="form-control" name="address" id="address" value="{{posts.address}}"
required/>
</div>
<br>

<div class="form-group">
<select class="form-control" id="farmingtype" name="farmingtype" required>
<option selected>{{posts.farming}}</option>
{%
  for d in farming %
    <option value="{{d.farmingtype}}">{{d.farmingtype}}</option>
  {% endfor %}
</select>
</div>
<br>

<button type="submit" class="btn btn-light btn-sm btn-block">Submit
Record</button>
</form>
<br>
<br>

</div>

<div class="col-md-4"></div>

</div>
</div>
</div>

<!-- Optional JavaScript; choose one of the two! -->

<!-- Option 1: jQuery and Bootstrap Bundle (includes Popper) -->
<script src="https://code.jquery.com/jquery-3.5.1.slim.min.js"
integrity="sha384-DfXdz2htPH0lsSSs5nCTpuj/zy4C+OGpamoFVy38MVBnE+IbbVYUew+OrCXaR
kfj" crossorigin="anonymous"></script>
<script
src="https://cdn.jsdelivr.net/npm/bootstrap@4.5.3/dist/js/bootstrap.bundle.min.
js"
integrity="sha384-ho+j7jyWK8fNQe+A12Hb8AhRq26LrZ/JpcUGGOn+Y7RsweNrtN/tE3MoK7ZeZ
Dyx" crossorigin="anonymous"></script>

<!-- Option 2: jQuery, Popper.js, and Bootstrap JS
<script src="https://code.jquery.com/jquery-3.5.1.slim.min.js"
integrity="sha384-DfXdz2htPH0lsSSs5nCTpuj/zy4C+OGpamoFVy38MVBnE+IbbVYUew+OrCXaR
kfj" crossorigin="anonymous"></script>
<script
src="https://cdn.jsdelivr.net/npm/popper.js@1.16.1/dist/umd/popper.min.js"

```

```
integrity="sha384-9/reFTGAW83EW2RDu2S0VKaIzap3H66lZH81PoY1FhbGU+6BZp6G7niu735Sk
71N" crossorigin="anonymous">></script>
<script
src="https://cdn.jsdelivr.net/npm/bootstrap@4.5.3/dist/js/bootstrap.min.js"
integrity="sha384-w1Q4orYjBQndcko6MimVbzY0tgp4pWB4lZ7lr30WKz0vr/aWKhXdBNmNb5D92
v7s" crossorigin="anonymous">></script>
-->
</body>
</html>
```

email_template.html

```
<!DOCTYPE html>
<html>

<head>
    <title>Welcome Back, Agrarian!</title>
    <style>
        body {
            font-family: Arial, sans-serif;
            background-color: #f4f4f4;
            padding: 20px;
        }

        .email-container {
            background-color: #ffffff;
            padding: 20px;
            border-radius: 10px;
            box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1);
        }

        h1 {
            color: #333;
            text-align: center;
        }

        p {
            font-size: 16px;
            line-height: 1.6;
        }

        img {
            max-width: 100%;
            height: auto;
            margin-bottom: 20px;
        }

        iframe {
            width: 100%;
```

```

        height: 315px;
        margin-bottom: 20px;
    }

```

</style>

</head>

<body>

<div class="email-container">

<header>

<h1>Glad to See You Once Again!</h1>

</header>

<main>

<p>Dear {{ username }},</p>

<p>Your unwavering commitment to the advancement of agriculture and farming is truly remarkable. Your tireless dedication continues to yield substantial and invaluable contributions, shaping a profound impact on our world. Thank you for your relentless efforts and invaluable contributions.</p>

<p>Enclosed below is a dynamic array of agricultural insights, carefully curated to illuminate the intricate tapestry of farming. Dive into this mosaic of knowledge, where the cultivation of crops and the artistry of sustainable practices converge to nurture our planet and feed the world.</p>

Explore Live Agricultural News

<p>Kindest Regards With Warmth,
Team AgroNexumLive</p>

</main>

</div>

</body>

email_template2.html

```

<!DOCTYPE html>
<html>

<head>
    <title>Welcome to the Agrarian Community!</title>
    <style>
        body {
            font-family: Arial, sans-serif;
            background-color: #f4f4f4;
            padding: 20px;
        }

        .email-container {
            background-color: #ffffff;

```

```

        padding: 20px;
        border-radius: 10px;
        box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1);
    }

    h1 {
        color: #333;
        text-align: center;
    }

    p {
        font-size: 16px;
        line-height: 1.6;
    }

    img {
        max-width: 100%;
        height: auto;
        margin-bottom: 20px;
    }

    iframe {
        width: 100%;
        height: 315px;
        margin-bottom: 20px;
    }

```

</style>

</head>

<body>

Welcome to Our Vibrant Community of Agrarians!

Dear {{ username }},

Congratulations and a warm welcome to our community of passionate agrarians! Your decision to join us marks the beginning of an enriching journey towards sustainable farming and agricultural excellence.

As a valued member of our community, you unlock access to an array of resources and knowledge that will elevate your farming pursuits:

- Immersive guides and expert insights on contemporary farming methodologies
- Exclusive sessions - webinars, workshops, and discussions led by industry pioneers
- A bustling platform for networking and knowledge exchange among fellow agrarians
- Timely updates on cutting-edge agricultural technologies and sustainable practices

```

        </ul>
        <p>To get started, begin exploring our Exclusive Community of
AgroNexumLive Agrarian's Portal where you can interact with fellow
Agrarians.</p>
        <a href="https://chat.whatsapp.com/HLwSnmHaQWk6MjMmsdbUWj"
target="_blank"></a>
        <br>
        <p>To further aid your learning experience, delve into our
repository:</p>
        <a href="{{ image_url }}" target="_blank"></a>
        <br>
        <a href="{{ video_url }}" target="_blank">Explore Live Agricultural
News and Insights</a>
        <p>We're thrilled to have you aboard! Your passion for agriculture
resonates with our mission to shape a sustainable and prosperous future.</p>
        <p>Warmest Regards,<br>Team AgroNexumLive</p>
    </main>
</div>
</body>

</html>

```

email_template3.html

```

<!DOCTYPE html>
<html>

<head>
    <title>Welcome to AgroNexumLive</title>
    <style>
        body {
            font-family: Arial, sans-serif;
            background-color: #f4f4f4;
            padding: 20px;
        }

        .email-container {
            background-color: #ffffff;
            padding: 20px;
            border-radius: 10px;
            box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1);
        }

        h1 {
            color: #333;
            text-align: center;

```

```

    }

    p {
        font-size: 16px;
        line-height: 1.6;
    }

    img {
        max-width: 100%;
        height: auto;
        margin-bottom: 20px;
    }

    iframe {
        width: 100%;
        height: 315px;
        margin-bottom: 20px;
    }

```

</style>

</head>

<body>

<div class="email-container">

<header>

<h1>Welcome to the World's First Ever Revolutionary Digital Agrarian Marketplace!</h1>

</header>

<main>

<p>Dear {{ username }},</p>

<p>A heartfelt welcome to our thriving agrarian marketplace! Your entry into our platform marks the start of an exciting journey in exploring and supporting sustainable farming practices.</p>

<p>As a newly registered buyer, here's what you can expect:</p>

- Access to a diverse selection of high-quality farm produce direct from agrarians
- Exclusive deals and offers on fresh, locally sourced products
- A seamless and secure platform for your agricultural procurement needs
- Regular updates on seasonal produce and special offerings

<p>To get started, begin exploring our Exclusive Community of AgroNexumLive Prospects' Portal where you can interact with fellow consumers.</p>


```

        <p>Now, Go through the current rates of some common products
besides us:</p>
        <a href="{{ image_url }}" target="_blank"></a>
        <br>
        <a href="{{ video_url }}" target="_blank">Feel Free to compare with
our MarketPlace...</a>
        <p>We're thrilled to have you join us on this journey of promoting
sustainable agriculture and supporting our agrarian community!</p>
        <p>Warm Regards,<br>Team AgroNexumLive</p>
    </main>
</div>
</body>

</html>

```

email_template4.html

```

<!DOCTYPE html>
<html>

<head>
    <title>Welcome Back to AgroNexumLive!</title>
    <style>
        body {
            font-family: Arial, sans-serif;
            background-color: #f4f4f4;
            padding: 20px;
        }

        .email-container {
            background-color: #ffffff;
            padding: 20px;
            border-radius: 10px;
            box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1);
        }

        h1 {
            color: #333;
            text-align: center;
        }

        p {
            font-size: 16px;
            line-height: 1.6;
        }

        img {
            max-width: 100%;
        }
    </style>
</head>
<body>
    <div class="email-container">
        <h1>Welcome Back to AgroNexumLive!</h1>
        <p>We're thrilled to have you join us on this journey of promoting
sustainable agriculture and supporting our agrarian community!</p>
        <p>Warm Regards,<br>Team AgroNexumLive</p>
    </div>
</body>

```

```

        height: auto;
        margin-bottom: 20px;
    }

    iframe {
        width: 100%;
        height: 315px;
        margin-bottom: 20px;
    }

```

</style>

</head>

<body>

<div class="email-container">

<header>

<h1>Glad to See You Once Again!</h1>

</header>

<main>

<p>Dear {{ username }},</p>

<p>Welcome back to our agrarian community! Your continued interest and commitment to the advancement of agriculture inspire us. We greatly appreciate your ongoing support and dedication to sustainable farming.</p>

<p>Enclosed below are the latest insights on agriculture:</p>

Explore Live Agricultural Prices and Listings

<p>We value your participation in our journey towards a more sustainable future. Thank you for being a part of our community!</p>

<p>Best Regards,
Team AgroNexumLive</p>

</main>

</div>

</body>

</html>

farmer.html

```

{% extends 'base.html' %}

{% block title %}
Register Agrarian Details
{% endblock title %}

{% block body %}
<h3 class="text-center"><span>Register Agrarian Details</span> </h3>

```

```

{% with messages=get_flashed_messages(with_categories=true) %}
{% if messages %}
{% for category, message in messages %}

<div class="alert alert-{{category}} alert-dismissible fade show" role="alert">
    {{message}}


</div>
    {% endfor %}
    {% endif %}
    {% endwith %}
<br>
<div class="container">

<div class="row">

<div class="col-md-4"></div>
<div class="col-md-4">

<form action="/register" method="post">
<div class="form-group">

<label for="rollno">Agrarian Name</label>
<input type="text" class="form-control" name="farmername" id="farmername" required>
</div>
<br>
<div class="form-group">

<label for="adharnumber">Agrarian Adhaar Number</label>
<input type="number" class="form-control" name="adharnumber" id="adharnumber" required>
</div>
<br>
<div class="form-group">

<label for="age">Agrarian Age</label>
<input type="number" class="form-control" name="age" id="age" required>
</div>
<br>
<div class="form-group">
<select class="form-control" id="gender" name="gender" required>
    <option selected>Gender Selection</option>

    <option value="Male">Male</option>
    <option value="Female">Female</option>
    <option value="Prefer Not to Say">Prefer Not to Say</option>
    <option value="Non-Binary">Non-Binary</option>
    <option value="Other">Other</option>

```

```

        </select>
    </div>
    <br>

    <div class="form-group">
        <label for="num">Contact Number</label>
        <input type="number" class="form-control" name="phonenumbers" id="phonenumbers" required>
    </div>
    <br>

    <div class="form-group">
        <label for="address">Agrarian's Address</label>
        <textarea class="form-control" name="address" id="address" required></textarea>
    </div>
    <br>

    <div class="form-group">
        <select class="form-control" id="farmingtype" name="farmingtype" required>
            <option selected>Specify Agricultural Focus</option>
            {% for d in farming %}
            <option value="{{d.farmingtype}}>{{d.farmingtype}}</option>
            {% endfor %}
        </select>
    </div>
    <br>

        <button type="submit" class="btn btn-success btn-block">Save
Records</button>
    </form>
    <br>
    <br>

</div>

<div class="col-md-4"></div>

</div></div>

    {% endblock body %}
```

farmerdetails.html

```
{% extends 'base.html' %}
{% block title %}
```

```

Agrarian Details
{%
    endblock title %}

{%
    block body %
        <h3 class="text-center"><span>Agrarian Details</span> </h3>

        {% with messages=get_flashed_messages(with_categories=true) %}
        {% if messages %}
            {% for category, message in messages %}

                <div class="alert alert-{{category}} alert-dismissible fade show" role="alert">
                    {{message}}


                </div>
            {% endfor %}
            {% endif %}
            {% endwith %}
        <br>

        <div class="container mt-3">
            <table class="table">
                <thead class="thead-light">
                    <tr>
                        <th scope="col">REGISTERED NUMBER</th>
                        <th scope="col">AGRARIAN NAME</th>
                        <th scope="col">ADHAAR</th>
                        <th scope="col">AGE</th>
                        <th scope="col">GENDER</th>
                        <th scope="col">CONTACT</th>
                        <th scope="col">ADDRESS</th>
                        <th scope="col">AGRICULTURAL FOCUS</th>
                        <th scope="col">EDIT</th>
                        <th scope="col">DELETE</th>
                        <th scope="col">ADD AGRO PRODUCT</th>

                    </tr>
                </thead>
                <tbody>
                    {% for post in query %}
                        <tr>
                            <th scope="row">{{post.rid}}</th>
                            <td>{{post.farmernname}}</td>
                            <td>{{post.adharnumber}}</td>
                            <td>{{post.age}}</td>
                            <td>{{post.gender}}</td>
                            <td>{{post.phonenumber}}</td>
                            <td>{{post.address}}</td>
                        </tr>
                    {% endfor %}
                </tbody>
            </table>
        </div>
    {%
        endblock body %}

```

```

<td>{{post.farming}}</td>

<td><a href="/edit/{{post.rid}}"><button class="btn btn-success">Edit
</button> </a> </td>

<td><a href="/delete/{{post.rid}}"><button onclick="return confirm('Are you
sure to Delete data');" class="btn btn-success">Delete </button> </a> </td>
<td><a href="/addagroproduct"><button class="btn btn-success">ADD </button>
</a> </td>
</tr>
{% endfor %}

</tbody>
</table>

</div>

```

{% endblock body %}

farming.html

```

{% extends 'base.html' %}
{% block title %}
Add Agricultural Focus
{% endblock title %}

{% block body %}
<h3 class="text-center bg-success text-white"><span>Add Agricultural
Focus</span> </h3>

{% with messages=get_flashed_messages(with_categories=true) %}
{% if messages %}
{% for category, message in messages %}

<div class="alert alert-{{category}} alert-dismissible fade show" role="alert">
{{message}}


</div>
{% endfor %}
{% endif %}
{% endwith %}
<br>
<div class="container">

```

```

<div class="row">

<div class="col-md-4"></div>
<div class="col-md-4">

<form action="/addfarming" method="post">

<div class="form-group">
<label for="dept">Enter Agriculture Focus</label>
<input type="text" class="form-control" name="farming" id="farming">
</div>
<br>

<button type="submit" class="btn btn-success btn-sm btn-block">Add
Agriculture Category</button>
</form>
<br>
<br>

</div>

<div class="col-md-4"></div>

</div></div>

{%
  endblock body %}


```

forgot_password.html

```

{% extends 'auth.html' %}
{% block title %}
AgroNexumLive OTP
{% endblock title %}
{% block content %}
<h1 class="mb-4 pb-0"><span>Login through OTP</span> </h1>

{% with messages=get_flashed_messages(with_categories=true) %}
{% if messages %}
{% for category, message in messages %}

<div class="alert alert-{{category}} alert-dismissible fade show" role="alert">
  {{message}}
</div>


```

```

    {% endfor %}
    {% endif %}
    {% endwith %}
<form class="text-white py-3 px-3" action="/forgot_password" method="post">
<div class="form-group">
    <label for="exampleInputEmail1">Email address</label>
    <input type="email" class="form-control mb-3" id="exampleInputEmail1"
name="email" aria-describedby="emailHelp" required>

    <button type="submit" class="form-control bg-success text-white">Login
Through OTP</button>
</form>

{% endblock content %}

```

index.html

```

{% extends 'base.html' %}

{% block title %}
    AgroNexumLive Homepage
{% endblock %}

{% block home %}
    menu-active
{% endblock %}

{% block body %}
    <!-- Add this section for the pop-up notification -->
    {% if redirect_message %}
        <div class="flash-message">
            <div class="alert alert-warning animated fadeInDown">
                {{ redirect_message }}
            </div>
        </div>
    {% endif %}
    <!-- End of pop-up notification -->

    <!-- ===== Footer ===== -->
    <footer id="footer">
        <div class="footer-top">
            <!-- ... (existing content) ... -->
        </div>

        <div class="container">
            <div class="credits">
                <br>
                Powered by <a href="/">Mirsha, Rameez, and Nevin</a>
            </div>
        </div>
    </footer>
{% endblock %}

```

```

        </div>
    </div>
</footer><!-- End Footer -->

<!-- jQuery and script for hiding flash message -->
<script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
<script>
$(document).ready(function() {
    // Automatically hide flash message after 5 seconds
    $('.flash-message').delay(3000).fadeOut('slow');
});
</script>
{% endblock %}

```

login.html

```

{% extends 'auth.html' %}

{% block title %}
AgroNexumLive login
{% endblock title %}

{% block content %}
<h1 class="mb-4 pb-0"><span>Login</span></h1>

{% with messages=get_flashed_messages(with_categories=true) %}
    {% if messages %}
        {% for category, message in messages %}
            <div class="alert alert-{{category}} alert-dismissible fade show"
role="alert">
                {{message}}
            </div>
        {% endfor %}
    {% endif %}
    {% endwith %}

<form class="text-white py-3 px-3" action="/login" method="post">
    <div class="form-group">
        <label for="exampleInputEmail1">Email address</label>
        <br>
        <input type="email" class="form-control mb-2" id="exampleInputEmail1"
name="email" aria-describedby="emailHelp" required style="width: 300px;">
    </div>
    <div class="form-group">
        <label for="exampleInputPassword1">Password</label>
        <input type="password" class="form-control mb-2" id="exampleInputPassword1"
name="password" required>
    </div>
    <br>

```

```

    <button type="submit" class="form-control bg-success
text-white">Login</button>
    <br> <!-- Line break added -->

    <p class="mb-2 pb-0" style="font-size: 18px;">New User?</p>
    <a href="/signup" class="about-btn scrollto">Signup</a>
</form>

<!-- Reduced size for "Forgot Password?" button -->
<br>
<a href="/forgot_password" class="about-btn scrollto" style="font-size:
75%;">Lgin Through OTP</a>

{% endblock content %}

```

signup.html

```

{% extends 'auth.html' %}
{% block title %}
AgroNexumLive Signup
{% endblock title %}
{% block content %}
<h1 class="mb-1"><span>Sign Up</span> </h1>

{% with messages=get_flashed_messages(with_categories=true) %}
{% if messages %}
{% for category, message in messages %}
<div class="alert alert-{{category}} alert-dismissible fade show" role="alert">
    {{message}}
</div>
{% endfor %}
{% endif %}
{% endwith %}
<form class="text-white py-3 px-3" action="/signup" method="post">
    <div class="form-group">
        <label for="exampleInputPassword1">UserName</label>
        <input type="text" class="form-control mb-2" name="username" id="username"
required>
    </div>
    <div class="form-group">
        <label for="exampleInputEmail1">Email address</label>
        <input type="email" class="form-control mb-2" id="exampleInputEmail1"
name="email" aria-describedby="emailHelp" required>
        <small id="emailHelp" class="form-text text-muted">We'll never share your
email with anyone else.</small>
    </div>
    <br>
    <div class="form-group">
        <label for="exampleInputPassword1">Password</label>

```

```

        <input type="password" class="form-control mb-2" name="password"
id="exampleInputPassword1" required>
    </div>
    <div class="form-check mb-2">
        <input type="checkbox" class="form-check-input" id="isAgrarian"
name="is_agrarian">
        <label class="form-check-label" for="isAgrarian">Are you a farmer?</label>
    </div>
    <button type="submit" class="form-control bg-success text-white">Sign
In</button>
    <p class="mb-2 pb-0">Already User?</p>
    <a href="/login" class="about-btn scrollto">Login</a>
</form>
{% endblock content %}
```

triggers.html

```

{% extends 'base.html' %}
{% block title %}
Agrarian Triggers Recorded
{% endblock title %}

{% block body %}
<h3 class="text-center"><span>Agrarian Triggers Recorded</span> </h3>

{% with messages=get_flashed_messages(with_categories=true) %}
{% if messages %}
{% for category, message in messages %}

<div class="alert alert-{{category}} alert-dismissible fade show" role="alert">
    {{message}}
```

```

</div>
    {% endfor %}
    {% endif %}
    {% endwith %}
<br>
```

```

<div class="container mt-4">
<table class="table">
    <thead class="thead-light">
        <tr>
            <th scope="col">AGRARIAN REF ID</th>
            <th scope="col">ACTION TAKEN</th>
            <th scope="col">OBSERVED TIMESTAMP</th>
```

```

        </tr>
    </thead>
    <tbody>
        {% for post in query %}
            <tr>

                <td>{{post.fid}}</td>
                <td>{{post.action}}</td>
                <td>{{post.timestamp}}</td>

            </tr>
        {% endfor %}

    </tbody>
</table>
</div>

{% endblock body %}

```

verify_otp.html

```

{% extends 'auth.html' %}
{% block title %}
AgroNexumLive OTP
{% endblock title %}
{% block content %}
<h1 class="mb-4 pb-0"><span>OTP AUTHENTICATION</span> </h1>

{% with messages=get_flashed_messages(with_categories=true) %}
{% if messages %}
{% for category, message in messages %}

<div class="alert alert-{{category}} alert-dismissible fade show" role="alert">
    {{message}}


</div>

{% endfor %}
{% endif %}
{% endwith %}
<form class="text-white py-3 px-3" action="/verify_otp" method="post">
    <div class="form-group">

```

```

        <label for="exampleInputOTP">Enter the One - Time Password</label>
        <input type="text" class="form-control mb-3" id="exampleInputOTP"
name="otp" required>
    </div>
    <button type="submit" class="form-control bg-success text-white">Verify Your
OTP</button>
</form>

{% endblock content %}

<!--
    !DOCTYPE html
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Verify OTP</title>
</head>
<body>
    <h1>Verify OTP</h1>
    <form class="text-white py-3 px-3" action="/verify_otp" method="post">
        <div class="form-group">
            <label for="exampleInputOTP">Enter OTP</label>
            <input type="text" class="form-control mb-3" id="exampleInputOTP"
name="otp" required>
        </div>
        <button type="submit" class="form-control bg-success text-white">Verify
OTP</button>
    </form>
</body>
</html>
-->
```

CSS

style.css

```

body {
    background: #fff;
    color: #2f3138;
    font-family: "Open Sans", sans-serif;
}

a {
    color: #fff!important;
    ;
    text-decoration: none;
    transition: 0.5s;
}

a:hover,
a:active,
```

```

a:focus {
    color: #fff!important;
    outline: none;
    text-decoration: none;
}

p {
    padding: 0;
    margin: 0 0 30px 0;
}

h1,
h2,
h3,
h4,
h5,
h6 {
    font-family: "Raleway", sans-serif;
    font-weight: 400;
    margin: 0 0 20px 0;
    padding: 0;
    color: #0e1b4d;
}

.main-page {
    margin-top: 70px;
}

@media (prefers-reduced-motion: no-preference) {
    :root {
        scroll-behavior: auto;
    }
}

/* Preloader */

#preloader {
    position: fixed;
    left: 0;
    top: 0;
    z-index: 999;
    width: 100%;
    height: 100%;
    overflow: visible;
    background: #fff url("../img/preloader.svg") no-repeat center center;
}

/* Back to top button */

.back-to-top {
    position: fixed;
    display: none;
    background: #198754!important;
}

```

```

color: #fff;
width: 40px;
height: 40px;
text-align: center;
border-radius: 50px;
right: 15px;
bottom: 15px;
transition: background 0.5s ease-in-out;
}

.back-to-top i {
    font-size: 24px;
    padding-top: 6px;
}

@media (max-width: 768px) {
    .back-to-top {
        bottom: 15px;
    }
}

.back-to-top:focus {
    background: #e0072f;
    color: #fff;
    outline: none;
}

.back-to-top:hover {
    background: #e0072f;
    color: #fff;
}

/* Sections Header
-----*/
.section-header {
    margin-bottom: 60px;
    position: relative;
    padding-bottom: 20px;
}

.section-header::before {
    content: '';
    position: absolute;
    display: block;
    width: 60px;
    height: 5px;
    background: #f82249;
    bottom: 0;
    left: calc(50% - 25px);
}

.section-header h2 {
    font-size: 36px;
}

```

```

        text-transform: uppercase;
        text-align: center;
        font-weight: 700;
        margin-bottom: 10px;
    }

.section-header p {
    text-align: center;
    padding: 0;
    margin: 0;
    font-size: 18px;
    font-weight: 500;
    color: #9195a2;
}

.section-with-bg {
    background-color: #f6f7fd;
}

/*
# Header
-----*/
#header {
    height: 90px;
    padding: 25px 0;
    position: fixed;
    left: 0;
    top: 0;
    right: 0;
    transition: all 0.5s;
    z-index: 997;
}

@media (max-width: 991px) {
    #header {
        background: rgba(6, 12, 34, 0.98);
        height: 70px;
        padding: 15px 0;
        transition: all 0.5s;
    }
}

@media (max-width: 1199px) {
    #header .container {
        max-width: 100%;
    }
}

#header.header-scrolled,
#header.header-fixed {
    background: rgba(6, 12, 34, 0.98);
    height: 70px;
    padding: 15px 0;
}

```

```

        transition: all 0.5s;
    }

#header #logo h1 {
    font-size: 36px;
    margin: 0;
    padding: 6px 0;
    line-height: 1;
    font-family: "Raleway", sans-serif;
    font-weight: 700;
    letter-spacing: 3px;
    text-transform: uppercase;
}

#header #logo h1 span {
    color: #f82249;
}

#header #logo h1 a,
#header #logo h1 a:hover {
    color: #fff;
}

#header #logo img {
    padding: 0;
    margin-top: -8px;
    max-height: 54px;
    border-radius: 8px 25px;
}

/*
-----#
# Navigation Menu
-----*/
/* Nav Menu Essentials */

.nav-menu,
.nav-menu * {
    margin: 0;
    padding: 0;
    list-style: none;
}

.nav-menu ul {
    position: absolute;
    display: none;
    top: 100%;
    left: 0;
    z-index: 99;
}

.nav-menu li {
    position: relative;

```

```

        white-space: nowrap;
    }

.nav-menu>li {
    float: left;
}

.nav-menu li:hover>ul,
.nav-menu li.sfHover>ul {
    display: block;
}

.nav-menu ul ul {
    top: 0;
    left: 100%;
}

.nav-menu ul li {
    min-width: 180px;
}

/* Nav Menu Arrows */

.sf-arrows .sf-with-ul {
    padding-right: 30px;
}

.sf-arrows .sf-with-ul:after {
    content: "\f107";
    position: absolute;
    right: 15px;
    font-family: FontAwesome;
    font-style: normal;
    font-weight: normal;
}

.sf-arrows ul .sf-with-ul:after {
    content: "\f105";
}

/* Nav Meu Container */

#nav-menu-container {
    float: right;
    margin: 0;
}

@media (max-width: 991px) {
    #nav-menu-container {
        display: none;
    }
}

```

```

/* Nav Meu Styling */

.nav-menu a {
    padding: 8px;
    text-decoration: none;
    display: inline-block;
    color: rgba(202, 206, 221, 0.8);
    font-family: "Raleway", sans-serif;
    font-weight: 600;
    font-size: 14px;
    outline: none;
}

@media (max-width: 1199px) {
    .nav-menu a {
        padding: 8px 4px;
    }
}

.nav-menu .menu-active a,
.nav-menu a:hover {
    color: #fff;
}

.nav-menu>li {
    margin-left: 8px;
}

.nav-menu>li>a:before {
    content: "";
    position: absolute;
    width: 0;
    height: 2px;
    bottom: 0;
    left: 0;
    background-color: green;
    visibility: hidden;
    transition: all 0.3s ease-in-out 0s;
}

.nav-menu a:hover:before,
.nav-menu li:hover>a:before,
.nav-menu .menu-active>a:before {
    visibility: visible;
    width: 100%;
}

.nav-menu li.buy-tickets a {
    color: #fff;
    background: #198754!important;
    padding: 7px 22px;
    border-radius: 50px;
    border: 2px solid #198754!important;
    transition: all ease-in-out 0.3s;
}

```

```

    font-weight: 500;
    margin-left: 8px;
    margin-top: 2px;
    line-height: 1;
    font-size: 13px;
}

.nav-menu li.buy-tickets a:hover {
    background: none;
}

.nav-menu li.buy-tickets:hover a:before,
.nav-menu li.buy-tickets.menu-active a:before {
    visibility: hidden;
}

.nav-menu ul {
    margin: 4px 0 0 0;
    padding: 10px;
    box-shadow: 0px 0px 30px rgba(127, 137, 161, 0.25);
    background: #fff;
    border-radius: 3px;
}

.nav-menu ul li {
    transition: 0.3s;
}

.nav-menu ul li a {
    padding: 10px;
    color: #060c22;
    transition: 0.3s;
    display: block;
    font-size: 13px;
    text-transform: none;
    border-radius: 3px;
}

.nav-menu ul li:hover>a {
    background: #f82249;
    color: #fff;
}

.nav-menu ul ul {
    margin: 0;
}

/* Mobile Nav Toggle */

#mobile-nav-toggle {
    position: fixed;
    right: 0;
    top: 0;
    z-index: 999;
}

```

```
margin: 15px 15px 0 0;
border: 0;
background: none;
font-size: 24px;
display: none;
transition: all 0.4s;
outline: none;
cursor: pointer;
}

#mobile-nav-toggle i {
    color: #fff;
}

@media (max-width: 991px) {
    #mobile-nav-toggle {
        display: inline;
    }
}

/* Mobile Nav Styling */

#mobile-nav {
    position: fixed;
    top: 0;
    padding-top: 18px;
    bottom: 0;
    z-index: 998;
    background: rgba(6, 12, 34, 0.9);
    left: -260px;
    width: 260px;
    overflow-y: auto;
    transition: 0.4s;
}

#mobile-nav ul {
    padding: 0;
    margin: 0;
    list-style: none;
}

#mobile-nav ul li {
    position: relative;
}

#mobile-nav ul li a {
    color: #fff;
    font-size: 17px;
    overflow: hidden;
    padding: 10px 22px 10px 15px;
    position: relative;
    text-decoration: none;
    width: 100%;
    display: block;
}
```

```

        outline: none;
    }

#mobile-nav ul li a:hover {
    color: #f82249;
}

#mobile-nav ul li li {
    padding-left: 30px;
}

#mobile-nav ul .menu-has-children i {
    position: absolute;
    right: 0;
    z-index: 99;
    padding: 15px;
    cursor: pointer;
    color: #fff;
}

#mobile-nav ul .menu-has-children i.fa-chevron-up {
    color: #f82249;
}

#mobile-nav ul .menu-item-active {
    color: #f82249;
}

#mobile-body-overly {
    width: 100%;
    height: 100%;
    z-index: 997;
    top: 0;
    left: 0;
    position: fixed;
    background: rgba(255, 255, 255, 0.8);
    display: none;
}

/*
 * Mobile Nav body classes */
body.mobile-nav-active {
    overflow: hidden;
}

body.mobile-nav-active #mobile-nav {
    left: 0;
}

body.mobile-nav-active #mobile-nav-toggle {
    color: #fff;
}

```

```

/*
# Intro Section
*/
#intro {
    width: 100%;
    height: 100vh;
    position: relative;
    overflow: hidden;
}

#intro video {
    position: absolute;
    top: 0;
    left: 0;
    width: 100%;
    height: 100%;
    object-fit: cover;
    z-index: -1; /* Ensures the video stays behind other content */
}

#intro:before {
    content: "";
    background: rgba(33,33,33, 0.274);
    position: absolute;
    bottom: 0;
    top: 0;
    left: 0;
    right: 0;
}

#intro .intro-container {
    position: absolute;
    bottom: 0;
    left: 0;
    top: 90px;
    right: 0;
    display: flex;
    justify-content: center;
    align-items: center;
    flex-direction: column;
    text-align: center;
    padding: 0 15px;
}

@media (max-width: 991px) {
    #intro .intro-container {
        top: 70px;
    }
}

#intro h1 {
    color: #ffffff;
    font-family: "Raleway", sans-serif;
    font-size: 56px;
}

```

```

        font-weight: 600;
        text-transform: uppercase;
    }

#intro h1 span {
    color: rgba(245, 245, 245, 0.997);
}

@media (max-width: 991px) {
    #intro h1 {
        font-size: 34px;
    }
}

#intro p {
    color: #ffffffff5;
    font-weight: 700;
    font-size: 20px;
}

@media (max-width: 991px) {
    #intro p {
        font-size: 16px;
    }
}

#intro .play-btn {
    width: 94px;
    height: 94px;
    background: radial-gradient(#f82249 50%, rgba(101, 111, 150, 0.15) 52%);
    border-radius: 50%;
    display: block;
    position: relative;
    overflow: hidden;
}

#intro .play-btn::after {
    content: '';
    position: absolute;
    left: 50%;
    top: 50%;
    transform: translateX(-40%) translateY(-50%);
    width: 0;
    height: 0;
    border-top: 10px solid transparent;
    border-bottom: 10px solid transparent;
    border-left: 15px solid #ffffff;
    z-index: 100;
    transition: all 400ms cubic-bezier(0.55, 0.055, 0.675, 0.19);
}

#intro .play-btn:before {
    content: '';
    position: absolute;
    width: 120px;
}

```

```

height: 120px;
-webkit-animation-delay: 0s;
animation-delay: 0s;
-webkit-animation: pulsate-btn 2s;
animation: pulsate-btn 2s;
-webkit-animation-direction: forwards;
animation-direction: forwards;
-webkit-animation-iteration-count: infinite;
animation-iteration-count: infinite;
-webkit-animation-timing-function: steps;
animation-timing-function: steps;
opacity: 1;
border-radius: 50%;
border: 2px solid rgb(247, 247, 247);
top: -15%;
left: -15%;
background: rgba(198, 16, 0, 0);
}

#intro .play-btn:hover::after {
    border-left: 15px solid #f82249;
    transform: scale(20);
}

#intro .play-btn:hover::before {
    content: '';
    position: absolute;
    left: 50%;
    top: 50%;
    transform: translateX(-40%) translateY(-50%);
    width: 0;
    height: 0;
    border: none;
    border-top: 10px solid transparent;
    border-bottom: 10px solid transparent;
    border-left: 15px solid #ffffff;
    z-index: 200;
    -webkit-animation: none;
    animation: none;
    border-radius: 0;
}

#intro .about-btn {
    font-family: "Raleway", sans-serif;
    font-weight: 500;
    font-size: 14px;
    letter-spacing: 1px;
    display: inline-block;
    padding: 12px 32px;
    border-radius: 50px;
    transition: 0.5s;
    line-height: 1;
    margin: 10px;
    color: #ffffff;
    -webkit-animation-delay: 0.8s;
}

```

```

        animation-delay: 0.8s;
        border: 2px solid #198754!important;
    }

#intro .about-btn:hover {
    background: black;
    color: #ffffff;
}

@-webkit-keyframes pulsate-btn {
    0% {
        transform: scale(0.6, 0.6);
        opacity: 1;
    }
    100% {
        transform: scale(1, 1);
        opacity: 0;
    }
}

@keyframes pulsate-btn {
    0% {
        transform: scale(0.6, 0.6);
        opacity: 1;
    }
    100% {
        transform: scale(1, 1);
        opacity: 0;
    }
}

/*
-----#
# About Section
-----*/
#about {
    background: url("../img/about-bg.jpg");
    background-size: cover;
    overflow: hidden;
    position: relative;
    color: #ffffff;
    padding: 60px 0 40px 0;
}

@media (min-width: 1024px) {
    #about {
        background-attachment: fixed;
    }
}

#about:before {
    content: "";
    background: rgba(13, 20, 41, 0.8);
    position: absolute;

```

```

        bottom: 0;
        top: 0;
        left: 0;
        right: 0;
    }

#about h2 {
    font-size: 36px;
    font-weight: bold;
    margin-bottom: 10px;
    color: #ffff;
}

#about h3 {
    font-size: 18px;
    font-weight: bold;
    text-transform: uppercase;
    margin-bottom: 10px;
    color: #ffff;
}

#about p {
    font-size: 14px;
    margin-bottom: 20px;
    color: #ffff;
}

/*
-----#
# Speakers Section
-----*/
#speakers {
    padding: 60px 0 30px 0;
}

#speakers .speaker {
    position: relative;
    overflow: hidden;
    margin-bottom: 30px;
}

#speakers .speaker .details {
    background: rgba(6, 12, 34, 0.76);
    position: absolute;
    left: 0;
    bottom: -30px;
    right: 0;
    text-align: center;
    padding-top: 10px;
    transition: all 300ms cubic-bezier(0.645, 0.045, 0.355, 1);
}

#speakers .speaker .details h3 {
    color: #ffff;
}

```

```
    font-size: 22px;
    font-weight: 600;
    margin-bottom: 5px;
}

#speakers .speaker .details p {
    color: #ffff;
    font-size: 15px;
    margin-bottom: 10px;
    font-style: italic;
}

#speakers .speaker .details .social {
    height: 30px;
}

#speakers .speaker .details a {
    color: #ffff;
}

#speakers .speaker .details a:hover {
    color: #f82249;
}

#speakers .speaker:hover .details {
    bottom: 0;
}

#speakers-details {
    padding: 60px 0;
}

#speakers-details .details h2 {
    color: #112363;
    font-size: 28px;
    font-weight: 700;
    margin-bottom: 10px;
}

#speakers-details .details .social {
    margin-bottom: 15px;
}

#speakers-details .details .social a {
    background: #e9edfb;
    color: #112363;
    line-height: 1;
    display: inline-block;
    text-align: center;
    border-radius: 50%;
    width: 36px;
    height: 36px;
    padding-top: 9px;
}
```

```

#speakers-details .details .social a:hover {
    background: #f82249;
    color: #fff;
}

#speakers-details .details .social a i {
    font-size: 18px;
}

#speakers-details .details p {
    color: #112363;
    font-size: 15px;
    margin-bottom: 10px;
}

/*
-----#
# Schedule Section
-----*/
#schedule {
    padding: 60px 0 60px 0;
}

#schedule .nav-tabs {
    text-align: center;
    margin: auto;
    display: block;
    border-bottom: 0;
    margin-bottom: 30px;
}

#schedule .nav-tabs li {
    display: inline-block;
    margin-bottom: 0;
}

#schedule .nav-tabs a {
    border: none;
    border-radius: 50px;
    font-weight: 600;
    background-color: #0e1b4d;
    color: #fff;
    padding: 10px 100px;
}

@media (max-width: 991px) {
    #schedule .nav-tabs a {
        padding: 8px 60px;
    }
}

@media (max-width: 767px) {
    #schedule .nav-tabs a {
        padding: 8px 50px;
}

```

```

        }
    }

@media (max-width: 480px) {
    #schedule .nav-tabs a {
        padding: 8px 30px;
    }
}

#schedule .nav-tabs a.active {
    background-color: #f82249;
    color: #fff;
}

#schedule .sub-heading {
    text-align: center;
    font-size: 18px;
    font-style: italic;
    margin: 0 auto 30px auto;
}

@media (min-width: 991px) {
    #schedule .sub-heading {
        width: 75%;
    }
}

#schedule .tab-pane {
    transition: ease-in-out .2s;
}

#schedule .schedule-item {
    border-bottom: 1px solid #cad4f6;
    padding-top: 15px;
    padding-bottom: 15px;
    transition: background-color ease-in-out 0.3s;
}

#schedule .schedule-item:hover {
    background-color: #fff;
}

#schedule .schedule-item time {
    padding-bottom: 5px;
    display: inline-block;
}

#schedule .schedule-item .speaker {
    width: 60px;
    height: 60px;
    overflow: hidden;
    border-radius: 50%;
    float: left;
    margin: 0 10px 10px 0;
}

```

```

#schedule .schedule-item .speaker img {
    height: 100%;
    transform: translateX(-50%);
    margin-left: 50%;
    transition: all ease-in-out 0.3s;
}

#schedule .schedule-item h4 {
    font-size: 18px;
    font-weight: 600;
    margin-bottom: 5px;
}

#schedule .schedule-item h4 span {
    font-style: italic;
    color: #19328e;
    font-weight: normal;
    font-size: 16px;
}

#schedule .schedule-item p {
    font-style: italic;
    color: #152b79;
    margin-bottom: 0;
}

/*
-----#
# Venue Section
-----*/
#venue {
    padding: 60px 0;
}

#venue .container-fluid {
    margin-bottom: 3px;
}

#venue .venue-map iframe {
    width: 100%;
    height: 100%;
    min-height: 300px;
}

#venue .venue-info {
    background: url("../img/venue-info-bg.jpg") top center no-repeat;
    background-size: cover;
    position: relative;
    padding-top: 60px;
    padding-bottom: 60px;
}

#venue .venue-info:before {

```

```

        content: "";
        background: rgba(13, 20, 41, 0.8);
        position: absolute;
        bottom: 0;
        top: 0;
        left: 0;
        right: 0;
    }

#venue .venue-info h3 {
    font-size: 36px;
    font-weight: 700;
    color: #fff;
}

@media (max-width: 574px) {
    #venue .venue-info h3 {
        font-size: 24px;
    }
}

#venue .venue-info p {
    color: #fff;
    margin-bottom: 0;
}

#venue .venue-gallery-container {
    padding-right: 12px;
}

#venue .venue-gallery {
    overflow: hidden;
    border-right: 3px solid #fff;
    border-bottom: 3px solid #fff;
}

#venue .venue-gallery img {
    transition: all ease-in-out 0.4s;
}

#venue .venue-gallery:hover img {
    transform: scale(1.1);
}

/*
-----#
# Hotels Section
-----*/
#hotels {
    padding: 60px 0;
}

#hotels .hotel {
    border: 1px solid #e0e5fa;
}

```

```

        background: #fff;
        margin-bottom: 30px;
    }

#hotels .hotel:hover .hotel-img img {
    transform: scale(1.1);
}

#hotels .hotel-img {
    overflow: hidden;
    margin-bottom: 15px;
}

#hotels .hotel-img img {
    transition: 0.3s ease-in-out;
}

#hotels h3 {
    font-weight: 600;
    font-size: 20px;
    margin-bottom: 5px;
    padding: 0 20px;
}

#hotels a {
    color: #791515;
}

#hotels a:hover {
    color: #f82249;
}

#hotels .stars {
    padding: 0 20px;
    margin-bottom: 5px;
}

#hotels .stars i {
    color: #ffc31d;
}

#hotels p {
    padding: 0 20px;
    margin-bottom: 20px;
    color: #220606;
    font-style: italic;
    font-size: 15px;
}

/*
-----#
# Gallery Section
-----*/
#gallery {

```

```

padding: 60px;
overflow: hidden;
}

#gallery .owl-nav,
#gallery .owl-dots {
    margin-top: 25px;
    text-align: center;
}

#gallery .owl-item {
    border-left: 2px solid #fff;
    border-right: 2px solid #fff;
}

#gallery .owl-dot {
    display: inline-block;
    margin: 0 5px;
    width: 12px;
    height: 12px;
    border-radius: 50%;
    background-color: #ddd;
}

#gallery .owl-dot.active {
    background-color: #f82249;
}

#gallery .gallery-carousel .owl-stage-outer {
    overflow: visible;
}

#gallery .gallery-carousel .center {
    border: 6px solid #f82249;
    margin: -10px;
    box-sizing: content-box;
    padding: 4px;
    background: #fff;
    z-index: 1;
}

/*
-----#
# Sponsors Section
-----*/
#supporters {
    padding: 60px 0;
}

#supporters .supporters-wrap {
    border-top: 1px solid #e0e5fa;
    border-left: 1px solid #e0e5fa;
    margin-bottom: 30px;
}

```

```

#supporters .supporter-logo {
    padding: 30px;
    display: flex;
    justify-content: center;
    align-items: center;
    border-right: 1px solid #e0e5fa;
    border-bottom: 1px solid #e0e5fa;
    overflow: hidden;
    background: rgba(255, 255, 255, 0.292);
    height: 160px;
}

#supporters .supporter-logo:hover img {
    transform: scale(1.2);
}

#supporters img {
    transition: all 0.4s ease-in-out;
}

/*
-----#
# F.A.Q Section
-----*/
#faq {
    padding: 60px 0;
}

#faq #faq-list {
    padding: 0;
    list-style: none;
}

#faq #faq-list li {
    border-bottom: 1px solid #ddd;
}

#faq #faq-list a {
    padding: 18px 0;
    display: block;
    position: relative;
    font-family: "Raleway", sans-serif;
    font-size: 16px;
    line-height: 24px;
    font-weight: 600;
    padding-right: 20px;
    cursor: pointer;
}

#faq #faq-list i {
    font-size: 24px;
    position: absolute;
    right: 0;
}

```

```

        top: 16px;
    }

#faq #faq-list p {
    margin-bottom: 20px;
}

@media (max-width: 768px) {
    #faq #faq-list a {
        font-size: 18px;
    }
    #faq #faq-list i {
        top: 13px;
    }
}

#faq #faq-list a.collapse {
    color: #f82249;
}

#faq #faq-list a.collapsed {
    color: #000;
}

#faq #faq-list a.collapsed i::before {
    content: "\f055" !important;
}

/*
-----#
# Subscribe Section
-----*/
#subscribe {
    padding: 60px;
    background: url(..../img/subscribe-bg.jpg) center center no-repeat;
    background-size: cover;
    overflow: hidden;
    position: relative;
}

#subscribe:before {
    content: "";
    background: rgba(6, 12, 34, 0.6);
    position: absolute;
    bottom: 0;
    top: 0;
    left: 0;
    right: 0;
}

@media (min-width: 1024px) {
    #subscribe {
        background-attachment: fixed;
    }
}

```

```

}

#subscribe .section-header h2,
#subscribe p {
    color: #fff;
}

#subscribe input {
    background: #fff;
    color: #060c22;
    border: 0;
    outline: none;
    margin: 0;
    padding: 9px 20px;
    border-radius: 50px;
    font-size: 14px;
}

@media (min-width: 767px) {
    #subscribe input {
        min-width: 400px;
    }
}

#subscribe button {
    border: 0;
    padding: 9px 25px;
    cursor: pointer;
    background: #f82249;
    color: #fff;
    transition: all 0.3s ease;
    outline: none;
    font-size: 14px;
    border-radius: 50px;
}

#subscribe button:hover {
    background: #e0072f;
}

@media (max-width: 460px) {
    #subscribe button {
        margin-top: 10px;
    }
}
}

/*
-----#
# Buy Tickets Section
-----*/
#buy-tickets {
    padding: 60px 0;
}

```

```
#buy-tickets .card {
    border: none;
    border-radius: 5px;
    transition: all 0.3s ease-in-out;
    box-shadow: 0 10px 25px 0 rgba(34, 6, 6, 0.1);
}

#buy-tickets .card:hover {
    box-shadow: 0 10px 35px 0 rgba(34, 6, 6, 0.2);
}

#buy-tickets .card hr {
    margin: 25px 0;
}

#buy-tickets .card .card-title {
    margin: 10px 0;
    font-size: 14px;
    letter-spacing: 1px;
    font-weight: bold;
}

#buy-tickets .card .card-price {
    font-size: 48px;
    margin: 0;
}

#buy-tickets .card ul li {
    margin-bottom: 20px;
}

#buy-tickets .card .text-muted {
    opacity: 0.7;
}

#buy-tickets .card .btn {
    font-size: 15px;
    border-radius: 50px;
    padding: 10px 40px;
    transition: all 0.2s;
    background-color: #f82249;
    border: 0;
    color: #fff;
}

#buy-tickets .card .btn:hover {
    background-color: #e0072f;
}

#buy-tickets #buy-ticket-modal input,
#buy-tickets #buy-ticket-modal select {
    border-radius: 0;
}

#buy-tickets #buy-ticket-modal .btn {
```

```

        font-size: 15px;
        border-radius: 50px;
        padding: 10px 40px;
        transition: all 0.2s;
        background-color: #f82249;
        border: 0;
        color: #fff;
    }

#buy-tickets #buy-ticket-modal .btn:hover {
    background-color: #e0072f;
}

/*
-----#
# Contact Section
-----*/
#contact {
    padding: 60px 0;
}

#contact .contact-info {
    margin-bottom: 20px;
    text-align: center;
}

#contact .contact-info i {
    font-size: 48px;
    display: inline-block;
    margin-bottom: 10px;
    color: #f82249;
}

#contact .contact-info address,
#contact .contact-info p {
    margin-bottom: 0;
    color: #112363;
}

#contact .contact-info h3 {
    font-size: 18px;
    margin-bottom: 15px;
    font-weight: bold;
    text-transform: uppercase;
    color: #631111;
}

#contact .contact-info a {
    color: #4869df;
}

#contact .contact-info a:hover {
    color: #f82249;
}

```

```
#contact .contact-address,  
#contact .contact-phone,  
#contact .contact-email {  
    margin-bottom: 20px;  
}  
  
@media (min-width: 768px) {  
    #contact .contact-address,  
    #contact .contact-phone,  
    #contact .contact-email {  
        padding: 20px 0;  
    }  
}  
  
@media (min-width: 768px) {  
    #contact .contact-phone {  
        border-left: 1px solid #ddd;  
        border-right: 1px solid #ddd;  
    }  
}  
  
#contact .php-email-form .validate {  
    display: none;  
    color: red;  
    margin: 0 0 15px 0;  
    font-weight: 400;  
    font-size: 13px;  
}  
  
#contact .php-email-form .error-message {  
    display: none;  
    color: #fff;  
    background: #ed3c0d;  
    text-align: left;  
    padding: 15px;  
    font-weight: 600;  
}  
  
#contact .php-email-form .error-message br+br {  
    margin-top: 25px;  
}  
  
#contact .php-email-form .sent-message {  
    display: none;  
    color: #fff;  
    background: #18d26e;  
    text-align: center;  
    padding: 15px;  
    font-weight: 600;  
}  
  
#contact .php-email-form .loading {  
    display: none;  
    background: #fff;
```

```

        text-align: center;
        padding: 15px;
    }

#contact .php-email-form .loading:before {
    content: "";
    display: inline-block;
    border-radius: 50%;
    width: 24px;
    height: 24px;
    margin: 0 10px -6px 0;
    border: 3px solid #18d26e;
    border-top-color: #eee;
    -webkit-animation: animate-loading 1s linear infinite;
    animation: animate-loading 1s linear infinite;
}

#contact .php-email-form input,
#contact .php-email-form textarea {
    border-radius: 0;
    box-shadow: none;
    font-size: 14px;
}

#contact .php-email-form input::focus,
#contact .php-email-form textarea::focus {
    background-color: #f82249;
}

#contact .php-email-form input {
    padding: 20px 15px;
}

#contact .php-email-form textarea {
    padding: 12px 15px;
}

#contact .php-email-form button[type="submit"] {
    background: #f82249;
    border: 0;
    padding: 10px 40px;
    color: #fff;
    transition: 0.4s;
    border-radius: 50px;
    cursor: pointer;
}

#contact .php-email-form button[type="submit"]:hover {
    background: #e0072f;
}

@-webkit-keyframes animate-loading {
    0% {
        transform: rotate(0deg);
    }
}

```

```

    100% {
      transform: rotate(360deg);
    }
  }

@keyframes animate-loading {
  0% {
    transform: rotate(0deg);
  }
  100% {
    transform: rotate(360deg);
  }
}

/*
# Footer
-----*/
#footer {
  background: #198754!important;
  color: #eee;
  font-size: 14px;
}

#footer .footer-top {
  background: #198754!important;
}

#footer .footer-top .footer-info {
  margin-bottom: 30px;
}

#footer .footer-top .footer-info h3 {
  font-size: 26px;
  margin: 0 0 20px 0;
  padding: 2px 0 2px 0;
  line-height: 1;
  font-family: "Raleway", sans-serif;
  font-weight: 700;
  color: #fff;
}

#footer .footer-top .footer-info img {
  height: 40px;
  margin-bottom: 10px;
}

#footer .footer-top .footer-info p {
  font-size: 14px;
  line-height: 24px;
  margin-bottom: 0;
  font-family: "Raleway", sans-serif;
  color: #fff;
}

```

```
#footer .footer-top .social-links a {
    font-size: 18px;
    display: inline-block;
    background: #362222;
    color: #eee;
    line-height: 1;
    padding: 8px 0;
    margin-right: 4px;
    border-radius: 50%;
    text-align: center;
    width: 36px;
    height: 36px;
    transition: 0.3s;
}

#footer .footer-top .social-links a:hover {
    background: #f82249;
    color: #fff;
}

#footer .footer-top h4 {
    font-size: 14px;
    font-weight: bold;
    color: #fff;
    text-transform: uppercase;
    position: relative;
    padding-bottom: 12px;
    border-bottom: 2px solid #f82249;
}

#footer .footer-top .footer-links {
    margin-bottom: 30px;
}

#footer .footer-top .footer-links ul {
    list-style: none;
    padding: 0;
    margin: 0;
}

#footer .footer-top .footer-links ul i {
    padding-right: 5px;
    color: #f82249;
    font-size: 18px;
}

#footer .footer-top .footer-links ul li {
    border-bottom: 1px solid #7f3e3e;
    padding: 10px 0;
}

#footer .footer-top .footer-links ul li:first-child {
    padding-top: 0;
}
```

```
#footer .footer-top .footer-links ul a {
    color: #eee;
}

#footer .footer-top .footer-links ul a:hover {
    color: #f82249;
}

#footer .footer-top .footer-contact {
    margin-bottom: 30px;
}

#footer .footer-top .footer-contact p {
    line-height: 26px;
}

#footer .footer-top .footer-newsletter {
    margin-bottom: 30px;
}

#footer .footer-top .footer-newsletter input[type="email"] {
    border: 0;
    padding: 6px 8px;
    width: 65%;
}

#footer .footer-top .footer-newsletter input[type="submit"] {
    background: #f82249;
    border: 0;
    width: 35%;
    padding: 6px 0;
    text-align: center;
    color: #fff;
    transition: 0.3s;
    cursor: pointer;
}

#footer .footer-top .footer-newsletter input[type="submit"]:hover {
    background: #e0072f;
}

#footer .copyright {
    text-align: center;
    padding-top: 30px;
}

#footer .credits {
    text-align: center;
    font-size: 13px;
    color: #ddd;
}
```

JavaScript

main.js

```
/**  
 * Template Name: TheEvent - v3.0.0  
 * Template URL:  
 https://bootstrapmade.com/theevent-conference-event-bootstrap-template/  
 * Author: BootstrapMade.com  
 * License: https://bootstrapmade.com/license/  
 */  
!(function($) {  
    "use strict";  
  
    // Back to top button  
    $(window).scroll(function() {  
        if ($(this).scrollTop() > 100) {  
            $('.back-to-top').fadeIn('slow');  
        } else {  
            $('.back-to-top').fadeOut('slow');  
        }  
    });  
    $('.back-to-top').click(function() {  
        $('html, body').animate({  
            scrollTop: 0  
        }, 1500, 'easeInOutExpo');  
        return false;  
    });  
  
    // Header fixed on scroll  
    $(window).scroll(function() {  
        if ($(this).scrollTop() > 100) {  
            $('#header').addClass('header-scrolled');  
        } else {  
            $('#header').removeClass('header-scrolled');  
        }  
    });  
  
    if ($(window).scrollTop() > 100) {  
        $('#header').addClass('header-scrolled');  
    }  
  
    // Initialize Venobox  
    $(window).on('load', function() {  
        $('.venobox').venobox({  
            bgcolor: '',  
            overlayColor: 'rgba(6, 12, 34, 0.85)',  
            closeBackground: '',  
            closeColor: '#fff',  
        });  
    });  
});
```

```

        share: false
    });
});

// Initiate superfish on nav menu
$('.nav-menu').superfish({
    animation: {
        opacity: 'show'
    },
    speed: 400
});

// Mobile Navigation
if ($('#nav-menu-container').length) {
    var $mobile_nav = $('#nav-menu-container').clone().prop({
        id: 'mobile-nav'
    });
    $mobile_nav.find('> ul').attr({
        'class': '',
        'id': ''
    });
    $('body').append($mobile_nav);
    $('body').prepend('<button type="button" id="mobile-nav-toggle"><i class="fa fa-bars"></i></button>');
    $('body').append('<div id="mobile-body-overly"></div>');
    $('#mobile-nav').find('.menu-has-children').prepend('<i class="fa fa-chevron-down"></i>');
}

$(document).on('click', '.menu-has-children i', function(e) {
    $(this).next().toggleClass('menu-item-active');
    $(this).nextAll('ul').eq(0).slideToggle();
    $(this).toggleClass("fa-chevron-up fa-chevron-down");
});

$(document).on('click', '#mobile-nav-toggle', function(e) {
    $('body').toggleClass('mobile-nav-active');
    $('#mobile-nav-toggle i').toggleClass('fa-times fa-bars');
    $('#mobile-body-overly').toggle();
});

$(document).click(function(e) {
    var container = $("#mobile-nav, #mobile-nav-toggle");
    if (!container.is(e.target) && container.has(e.target).length === 0) {
        if ($('body').hasClass('mobile-nav-active')) {
            $('body').removeClass('mobile-nav-active');
            $('#mobile-nav-toggle i').toggleClass('fa-times fa-bars');
            $('#mobile-body-overly').fadeOut();
        }
    }
});
}
);

```

```

} else if ($("#mobile-nav, #mobile-nav-toggle").length) {
    $("#mobile-nav, #mobile-nav-toggle").hide();
}

// Smooth scroll for the navigation menu and links with .scrollto classes
var scrolltoOffset = $('#header').outerHeight() - 21;
if (window.matchMedia("(max-width: 991px)").matches) {
    scrolltoOffset += 20;
}
$(document).on('click', '.nav-menu a, #mobile-nav a, .scrollto', function(e)
{
    if (location.pathname.replace(/^\//, '') == this.pathname.replace(/^\//,
'') && location.hostname == this.hostname) {
        var target = $(this.hash);
        if (target.length) {
            e.preventDefault();

            var scrollto = target.offset().top - scrolltoOffset;

            if ($(this).attr("href") == '#header') {
                scrollto = 0;
            }

            $('html, body').animate({
                scrollTop: scrollto
            }, 1500, 'easeInOutExpo');

            if ($(this).parents('.nav-menu').length) {
                $('.nav-menu .menu-active').removeClass('menu-active');
                $(this).closest('li').addClass('menu-active');
            }
        }

        if($('body').hasClass('mobile-nav-active')) {
            $('body').removeClass('mobile-nav-active');
            $('#mobile-nav-toggle i').toggleClass('fa-times fa-bars');
            $('#mobile-body-overly').fadeOut();
        }
        return false;
    }
});
});

// Activate smooth scroll on page load with hash links in the url
$(document).ready(function() {
    if (window.location.hash) {
        var initial_nav = window.location.hash;
        if ($(initial_nav).length) {
            var scrollto = $(initial_nav).offset().top - scrolltoOffset;
            $('html, body').animate({
                scrollTop: scrollto
            });
        }
    }
});

```

```

        }, 1500, 'easeInOutExpo');
    }
});
});

// Navigation active state on scroll
var nav_sections = $('section');
var main_nav = $('.nav-menu, #mobile-nav');

$(window).on('scroll', function() {
    var cur_pos = $(this).scrollTop() + 200;

    nav_sections.each(function() {
        var top = $(this).offset().top,
            bottom = top + $(this).outerHeight();

        if (cur_pos >= top && cur_pos <= bottom) {
            if (cur_pos <= bottom) {
                main_nav.find('li').removeClass('menu-active');
            }
            main_nav.find('a[href="#' + $(this).attr('id') +
'"']).parent('li').addClass('menu-active');
        }
        if (cur_pos < 300) {
            $(".nav-menu li:first").addClass('menu-active');
        }
    });
});

// Gallery carousel (uses the Owl Carousel library)
$(".gallery-carousel").owlCarousel({
    autoplay: true,
    dots: true,
    loop: true,
    center: true,
    responsive: {
        0: {
            items: 1
        },
        768: {
            items: 3
        },
        992: {
            items: 4
        },
        1200: {
            items: 5
        }
    }
});

```

```

// Buy tickets select the ticket type on click
$('#buy-ticket-modal').on('show.bs.modal', function(event) {
    var button = $(event.relatedTarget);
    var ticketType = button.data('ticket-type');
    var modal = $(this);
    modal.find('#ticket-type').val(ticketType);
});

// Init AOS
function aos_init() {
    AOS.init({
        duration: 1000,
        once: true
    });
}
$(window).on('load', function() {
    aos_init();
});
})(jQuery);

```

Python

agroNexumLive.py

```

from flask import Flask, render_template, request, session, redirect, url_for, flash, session
from flask_sqlalchemy import SQLAlchemy
from flask_login import UserMixin, login_user, logout_user, login_manager, LoginManager, login_required, current_user
from werkzeug.security import generate_password_hash, check_password_hash
from flask_mail import Mail, Message
from flask import render_template
import time
import requests
import random
import string
import secrets

from email.mime.multipart import MIME Multipart
from email.mime.image import MIMEImage
from email.mime.text import MIMEText

# MY db connection
local_server= True

```

```

app = Flask(__name__)
print(app.name)
app.secret_key='harshithbhaskar'

# Configure Flask-Mail settings
app.config['MAIL_SERVER'] = 'smtp.gmail.com'
app.config['MAIL_PORT'] = 465 # Use port 465 for SSL
app.config['MAIL_USERNAME'] = 'agronexumlive@gmail.com'
app.config['MAIL_PASSWORD'] = 'vgntssbcnbrdgjhf'
app.config['MAIL_USE_TLS'] = False
app.config['MAIL_USE_SSL'] = True
app.config['MAIL_DEFAULT_SENDER'] = ('AgroNexumLive',
'agronexumlive@gmail.com') # Set a default sender

# Initialize Flask-Mail
mail = Mail(app)

# An extensive blocking for sending automated mails
# Function to send email with dynamic content
def send_dynamic_email(recipient, subject, username, image_url, video_url,
html_url):
    msg = Message(subject=subject,
recipients=[recipient],sender=('AgroNexumlive', 'agronexumlive@gmail.com'))
    msg.html = render_template(
        html_url,
        username=username,
        image_url=image_url,
        video_url=video_url
    )
    mail.send(msg)

"""def embedded_send_dynamic_email(recipient, subject, username, image_path,
video_url, html_template, mail):
    msg = Message(subject=subject, recipients=[recipient],
sender=('AgroNexumlive', 'agronexumlive@gmail.com'))

    with open(image_path, 'rb') as img_file:
        image = MIMEImage(img_file.read())
        image.add_header('Content-ID', '<image>')
    msg.attach(image)

    msg.html = render_template(html_template, username=username,
image_url='cid:image', video_url=video_url)

    mail.send(msg)"""

"""def send_email(recipient, subject, body):
    msg = Message(subject=subject, recipients=[recipient],
sender=('AgroNexumlive', 'agronexumlive@gmail.com'))
```

```

        msg.body = body
        mail.send(msg)
    """

def generate_otp(length=6):
    alphabet = string.ascii_uppercase + string.digits
    return ''.join(secrets.choice(alphabet) for _ in range(length))

@app.route('/forgot_password', methods=['GET', 'POST'])
def forgot_password():
    if request.method == "POST":
        email = request.form.get('email')
        # Check if email is provided
        if not email:
            flash("Please enter your email to proceed with the OTP Login.", "warning")
            return render_template('forgot_password.html')

        # Generate OTP
        otp = generate_otp()
        # Here, you can update the user's record in the database with the
        hashed value of the OTP
        # Update the 'user' table accordingly using SQL or an ORM method like
        SQLAlchemy
        # Store the OTP and email in session
        session['otp'] = otp
        session['email'] = email

        # Send OTP to the user's email
        msg = Message('AgroNexumLive Login OTP', recipients=[email],
        sender='your_email@gmail.com')
        msg.body = f'Hello There, Your One Time Password for logging into
        AgroNexum Live is: {otp}'
        mail.send(msg)

        flash("An OTP has been sent to your email for your login.", "success")
        return redirect(url_for('verify_otp'))

    return render_template('forgot_password.html')

@app.route('/verify_otp', methods=['GET', 'POST'])
def verify_otp():
    if request.method == "POST":
        entered_otp = request.form.get('otp')
        # Check if entered OTP matches the stored OTP in session
        if 'otp' in session and session['otp'] == entered_otp:

```

```

# If OTP matches, proceed to login with the email provided
email = session['email']
user = MyUser.query.filter_by(email=email).first()
if user:
    # Login the user using the login_user function (assuming you're
    using Flask-Login)
    login_user(user)
    flash("OTP Login Successful ! You are now officially Logged
    in.", "success")

    # Set the session variable to indicate the user is logged in
    session['logged_in'] = True
    # Sending a login acknowledgment email for agrarians
    if user.is_agrarian:
        subject = "Welcome Back, Agrarian!"
        image_url =
"https://drive.google.com/file/d/1FdDaL6D4mVy_jZbJ4YwzeM-0Y7oBwzWD/view?usp=dri
ve_link" # Replace with the actual image URL
        news_url =
"https://timesofindia.indiatimes.com/topic/agriculture" # Replace with the
actual video URL
            send_dynamic_email(user.email, subject, user.username,
image_url, news_url,'email_template.html')
            #embedded_send_dynamic_email(recipient_email, subject,
username, image_path, video_url, html_template, mail)
            #embedded_send_dynamic_email('recipient@example.com',
'Welcome Back!', 'John', '/path/to/image.jpg', 'https://example.com/video',
'email_template.html')

    else:
        subject = "Welcome Back to AgroNexumLive!"
        image_url =
"https://drive.google.com/file/d/1xDcrhp1Dfdfn-z1Rj-jhMT4g0FMAeQy8/view?usp=dri
ve_link" # Replace with the actual image URL
        news_url = "https://www.oneindia.com/vegetables-price.html"
# Replace with the actual video URL
        send_dynamic_email(user.email, subject, user.username,
image_url, news_url,'email_template4.html')

    return redirect(url_for('index')) # Redirect to the main page
after successful login
else:
    flash("OOPS! User not found! SIGN UP to Continue", "danger")
    return redirect(url_for('signup')) # Redirect to login page if
user not found

    flash("Invalid OTP. Please try again.", "warning")
    return render_template('verify_otp.html')

```

```

    return render_template('verify_otp.html')

# this is for getting unique user access
login_manager=LoginManager(app)
login_manager.login_view='login'

@login_manager.user_loader
def load_user(user_id):
    return MyUser.query.get(int(user_id))

#
app.config['SQLALCHEMY_DATABASE_URL']='mysql://username:password@localhost/data
base_name'
app.config['SQLALCHEMY_DATABASE_URI']='mysql://root:@localhost/farmers'
db=SQLAlchemy(app)

# here we will create db models that is tables
class Test(db.Model):
    id=db.Column(db.Integer,primary_key=True)
    name=db.Column(db.String(100))

class Farming(db.Model):
    fid=db.Column(db.Integer,primary_key=True)
    farmingtype=db.Column(db.String(100))

class Addagroproducts(db.Model):
    username=db.Column(db.String(50))
    email=db.Column(db.String(50))
    pid=db.Column(db.Integer,primary_key=True)
    productname=db.Column(db.String(100))
    productdesc=db.Column(db.String(300))
    price=db.Column(db.Integer)

class Trig(db.Model):
    id=db.Column(db.Integer,primary_key=True)
    fid=db.Column(db.String(100))
    action=db.Column(db.String(100))
    timestamp=db.Column(db.String(100))

class MyUser(UserMixin, db.Model):
    __tablename__ = 'user'  # Explicitly specifying the table name
    id = db.Column(db.Integer, primary_key=True)
    username = db.Column(db.String(50))
    email = db.Column(db.String(50), unique=True)
    password = db.Column(db.String(1000))

```

```

is_agrarian = db.Column(db.Boolean, default=False)
is_admin = db.Column(db.Boolean, default=False)

class Register(db.Model):
    rid=db.Column(db.Integer,primary_key=True)
    farmername=db.Column(db.String(50))
    adharnumber=db.Column(db.String(50))
    age=db.Column(db.Integer)
    gender=db.Column(db.String(50))
    phonenumer=db.Column(db.String(50))
    address=db.Column(db.String(50))
    farming=db.Column(db.String(50))

@app.route('/')
@app.route('/')
def index():
    redirect_message = session.pop('redirect_message', None)
    return render_template('index.html', redirect_message=redirect_message)

@app.route('/farmerdetails')
@login_required
def farmerdetails():
    if current_user.is_agrarian or current_user.is_admin:
        if current_user.is_admin:
            query = Register.query.all()
            return render_template('farmerdetails.html', query=query)
        if current_user.is_agrarian:
            # Fetch the current logged-in user's username
            current_username = current_user.username

            # Query to retrieve agrarian details for the current user
            agrarian_details = (
                db.session.query(Register)
                .join(MyUser, MyUser.username == Register.farmername)
                .filter(MyUser.username == current_username)
                .all()
            )
            return render_template('farmerdetails.html',
query=agrarian_details)
        else:
            #flash("You don't have permission to view Agrarian Details page.",
            "warning")
            session['redirect_message'] = "You don't have permission to view
Agrarian Details page."
            return redirect(url_for('index')) # Redirect to the homepage or
another suitable route

```

```

@app.route('/agroproducts')
def agroproducts():
    # query=db.engine.execute(f"SELECT * FROM `addagroproducts`")
    query=Addagroproducts.query.all()
    return render_template('agroproducts.html',query=query)

@app.route('/addagroproduct',methods=[ 'POST','GET'])
@login_required
def addagroproduct():

    if request.method=="POST":
        username=request.form.get('username')
        email=request.form.get('email')
        productname=request.form.get('productname')
        productdesc=request.form.get('productdesc')
        price=request.form.get('price')

    products=Addagroproducts(username=username,email=email,productname=productname,
    productdesc=productdesc,price=price)
        db.session.add(products)
        db.session.commit()
        flash("Product Added Successfully","info")
        return redirect('/agroproducts')

    return render_template('addagroproducts.html')

@app.route('/triggers')
@login_required
def triggers():
    if current_user.is_admin:
        # query=db.engine.execute(f"SELECT * FROM `trig`")
        query=Trig.query.all()
        return render_template('triggers.html',query=query)
    else:
        session['redirect_message'] = "Only Administrators can view 'Records Created' page."
        return redirect(url_for('index')) # Redirect to the homepage or another suitable route

@app.route('/addfarming',methods=[ 'POST','GET'])
@login_required
def addfarming():
    if current_user.is_agrarian or current_user.is_admin:
        if request.method=="POST":
            farmingtype=request.form.get('farming')
            query=Farming.query.filter_by(farmingtype=farmingtype).first()
            if query:

```

```

        flash("Agricultural Focus Type Already Exists","warning")
        return redirect('/addfarming')
    dep=Farming(farmingtype=farmingtype)
    db.session.add(dep)
    db.session.commit()
    flash("Agricultural Type Added Successfully... ","success")
    return redirect('/addfarming')
    return render_template('farming.html')
else:
    flash("You don't have permission to access 'Add Produce type' page.", "warning")
    return redirect(url_for('agroproducts')) # Redirect to the homepage or another suitable route

```

```

@app.route("/delete/<string:rid>",methods=[ 'POST','GET'])
@login_required
def delete(rid):
    # db.engine.execute(f"DELETE FROM `register` WHERE `register`.`rid`={rid}")
    post=Register.query.filter_by(rid=rid).first()
    db.session.delete(post)
    db.session.commit()
    flash("Slot Deleted Successful","warning")
    return redirect('/farmerdetails')

```

```

@app.route("/edit/<string:rid>",methods=[ 'POST','GET'])
@login_required
def edit(rid):
    user = MyUser.query.filter_by(username=rid).first()

    # farming=db.engine.execute("SELECT * FROM `farming`")
    if request.method=="POST":
        farmername=request.form.get('farmername')
        adharnumber=request.form.get('adharnumber')
        age=request.form.get('age')
        gender=request.form.get('gender')
        phonenumbers=request.form.get('phonenumbers')
        address=request.form.get('address')
        farmingtype=request.form.get('farmingtype')
        # query=db.engine.execute(f"UPDATE `register` SET
        `farmername`='{farmername}',`adharnumber`='{adharnumber}',`age`='{age}',`gender`='
        '{gender}',`phonenumbers`='{phonenumbers}',`address`='{address}',`farming`='
        {farmingtype}'")
        post=Register.query.filter_by(rid=rid).first()
        print(post.farmername)
        post.farmername=farmername
        post.adharnumber=adharnumber

```

```

post.age=age
post.gender=gender
post.phonenumber=phonenumber
post.address=address
post.farming=farmingtype
db.session.commit()
flash("Slot is Updated","success")
return redirect('/farmerdetails')
posts=Register.query.filter_by(rid=rid).first()
farming=Farming.query.all()
return render_template('edit.html',posts=posts,farming=farming)

@app.route('/signup', methods=['POST', 'GET'])
def signup():
    if request.method == "POST":
        username = request.form.get('username')
        email = request.form.get('email')
        password = request.form.get('password')
        is_agrarian = bool(request.form.get('is_agrarian'))

        user = MyUser.query.filter_by(email=email).first()
        if user:
            flash("Kindly Verify! Email Already Exists!", "warning")
            return render_template('/signup.html')

        enc_password = generate_password_hash(password)
        new_user = MyUser(username=username, email=email,
password=enc_password, is_agrarian=is_agrarian)
        db.session.add(new_user)
        db.session.commit()

        flash("Signup Successful... Please Login to continue.", "success")
        # Sending email after successful signup
        if new_user.is_agrarian:
            subject = "Welcome to the Elite Club of Agrarian!"
            image_url =
"https://agritech.tnau.ac.in/agriculture/agri_nutrientmgt_priceoffertilizers.ht
ml" # Replace with the actual image URL
            news_url = "https://timesofindia.indiatimes.com/topic/agriculture"
# Replace with the actual video URL
            send_dynamic_email(new_user.email, subject, new_user.username,
image_url, news_url,'email_template2.html')
        else:
            subject = "Welcome to the World's First Ever Revolutionary Digital
Agrarian Marketplace!"
            image_url =
"https://rates.goldenchennai.com/vegetable-price/chennai-vegetable-price-today"
# Replace with the actual image URL

```

```

        news_url = "http://127.0.0.1:8080/agroproducts" # Replace with the
actual video URL
        send_dynamic_email(new_user.email, subject, new_user.username,
image_url, news_url,'email_template3.html')

    return render_template('login.html')

return render_template('signup.html')

@app.route('/login',methods=['POST','GET'])
def login():
    if request.method == "POST":
        email=request.form.get('email')
        password=request.form.get('password')
        user=MyUser.query.filter_by(email=email).first()

        if user and check_password_hash(user.password,password):
            login_user(user)
            flash("Login Success","primary")
            # Set the session variable to indicate the user is logged in
            session['logged_in'] = True
            # Sending a login acknowledgment email for agrarians
            if user.is_agrarian:
                subject = "Welcome Back, Agrarian!"
                image_url =
"https://drive.google.com/file/d/1FdDaL6D4mVy_jZbJ4YwzeM-0Y7oBwzWD/view?usp=dri
ve_link" # Replace with the actual image URL
                news_url =
"https://timesofindia.indiatimes.com/topic/agriculture" # Replace with the
actual video URL
                send_dynamic_email(user.email, subject, user.username,
image_url, news_url,'email_template.html')
                #embedded_send_dynamic_email(recipient_email, subject,
username, image_path, video_url, html_template, mail)
                #embedded_send_dynamic_email('recipient@example.com', 'Welcome
Back!', 'John', '/path/to/image.jpg', 'https://example.com/video',
'email_template.html')

            else:
                subject = "Welcome Back to AgroNexumLive!"
                image_url =
"https://drive.google.com/file/d/1xDcrhp1Dfdfn-z1Rj-jhMT4g0FMAeQy8/view?usp=dri
ve_link" # Replace with the actual image URL
                news_url = "https://www.oneindia.com/vegetables-price.html" #
Replace with the actual video URL
                send_dynamic_email(user.email, subject, user.username,
image_url, news_url,'email_template4.html')
                return redirect(url_for('index'))

        elif user:

```

```

        flash("Please Enter The valid Password !","warning")
        return render_template('login.html')
    else:
        flash("Please Sign Up to continue !","warning")
        return render_template('signup.html')

    return render_template('login.html')

@app.route('/logout')
@login_required
def logout():
    logout_user()
    flash("Logout Successful","warning")
    # Clear the 'logged_in' session variable
    session.pop('logged_in', None)
    return redirect(url_for('login'))


@app.route('/register',methods=['POST','GET'])
@login_required
def register():
    if current_user.is_agrarian or current_user.is_admin:
        farming=Farming.query.all()
        if request.method=="POST":
            farmername=request.form.get('farmername')
            adharnumber=request.form.get('adharnumber')
            age=request.form.get('age')
            gender=request.form.get('gender')
            phonenumber=request.form.get('phonenumber')
            address=request.form.get('address')
            farmingtype=request.form.get('farmingtype')

            query=Register(farmername=farmername,adharnumber=adharnumber,age=age,gender=gender,phonenumber=phonenumber,address=address,farming=farmingtype)
            db.session.add(query)
            db.session.commit()
            # query=db.engine.execute(f"INSERT INTO `register`(`farmername`, `adharnumber`, `age`, `gender`, `phonenumber`, `address`, `farming`) VALUES ('{farmername}', '{adharnumber}', '{age}', '{gender}', '{phonenumber}', '{address}', '{farmingtype}')")
            # flash("Your Record Has Been Saved","success")
            return redirect('/farmerdetails')
        return render_template('farmer.html',farming=farming)
    else:
        session['redirect_message'] = "You are not an Agrarian !"
        return redirect(url_for('index')) # Redirect to the homepage or another suitable route

```

```

@app.route('/test')
def test():
    try:
        Test.query.all()
        return 'My database is Connected'
    except:
        return 'My db is not Connected'

app.run(host='0.0.0.0', port=8080, debug=True)

```

SQL

DDLcommands.sql

```

-- phpMyAdmin SQL Dump
-- version 5.0.2
-- https://www.phpmyadmin.net/
--
-- Host: 127.0.0.1
-- Generation Time: Jan 20, 2021 at 06:31 AM
-- Server version: 10.4.11-MariaDB
-- PHP Version: 7.2.29

SET SQL_MODE = "NO_AUTO_VALUE_ON_ZERO";
START TRANSACTION;
SET time_zone = "+00:00";

/*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
/*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
/*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
/*!40101 SET NAMES utf8mb4 */;

--
-- Database: `farmers`


CREATE TABLE `addagroproducts` (
  `username` varchar(50) NOT NULL,
  `email` varchar(50) NOT NULL,
  `pid` int(11) NOT NULL,
  `productname` varchar(100) NOT NULL,
  `productdesc` text NOT NULL,
  `price` varchar(100) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

```

```
CREATE TABLE `farming` (
  `fid` int(11) NOT NULL,
  `farmingtype` varchar(200) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

```
CREATE TABLE `register` (
  `rid` int(11) NOT NULL,
  `farmername` varchar(50) NOT NULL,
  `adharnumber` varchar(20) NOT NULL,
  `age` int(100) NOT NULL,
  `gender` varchar(50) NOT NULL,
  `phonenumer` varchar(12) NOT NULL,
  `address` varchar(50) NOT NULL,
  `farming` varchar(50) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

```
CREATE TABLE `test` (
  `id` int(11) NOT NULL,
  `name` varchar(50) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

```
CREATE TABLE `trig` (
  `id` int(11) NOT NULL,
  `fid` varchar(50) NOT NULL,
  `action` varchar(50) NOT NULL,
  `timestamp` datetime NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

```
CREATE TABLE `user` (
  `id` int(11) NOT NULL,
  `username` varchar(50) NOT NULL,
  `email` varchar(50) NOT NULL,
  `password` varchar(500) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

```
ALTER TABLE `user`
ADD COLUMN `is_agrarian` BOOLEAN NOT NULL DEFAULT FALSE AFTER `password`;
```

```
ALTER TABLE `user`
ADD COLUMN `is_admin` BOOLEAN NOT NULL DEFAULT FALSE AFTER `is_agrarian`;
```

```
-- Indexes for dumped tables
```

```
--
```

```

--  

-- Indexes for table `addagroproducts`  

--  

ALTER TABLE `addagroproducts`  

  ADD PRIMARY KEY (`pid`);  

--  

-- Indexes for table `farming`  

--  

ALTER TABLE `farming`  

  ADD PRIMARY KEY (`fid`);  

--  

-- Indexes for table `register`  

--  

ALTER TABLE `register`  

  ADD PRIMARY KEY (`rid`);  

--  

-- Indexes for table `test`  

--  

ALTER TABLE `test`  

  ADD PRIMARY KEY (`id`);  

--  

-- Indexes for table `trig`  

--  

ALTER TABLE `trig`  

  ADD PRIMARY KEY (`id`);  

--  

-- Indexes for table `user`  

--  

ALTER TABLE `user`  

  ADD PRIMARY KEY (`id`);  

--  

-- AUTO_INCREMENT for dumped tables
--  

--  

-- AUTO_INCREMENT for table `addagroproducts`  

--  

ALTER TABLE `addagroproducts`  

  MODIFY `pid` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=31;  

--  

-- AUTO_INCREMENT for table `farming`  

--  

ALTER TABLE `farming`  


```

```

MODIFY `fid` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=32;

-- 
-- AUTO_INCREMENT for table `register`
-- 

ALTER TABLE `register`
MODIFY `rid` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=31;

-- 
-- AUTO_INCREMENT for table `test`
-- 

ALTER TABLE `test`
MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=2;

-- 
-- AUTO_INCREMENT for table `trig`
-- 

ALTER TABLE `trig`
MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=41;

-- 
-- AUTO_INCREMENT for table `user`
-- 

ALTER TABLE `user`
MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=2;
COMMIT;

```

--Indexes

```

--For the addagroproducts table:
CREATE INDEX idx_username_addagroproducts ON addagroproducts(username);

--For the register table:
CREATE INDEX idx_adharnumber_register ON register(adharnumber);

--For the trig table:
CREATE INDEX idx_fid_trig ON trig(fid);

```

--Views

```

-- Create a view for the addagroproducts table
CREATE VIEW addagroproducts_view AS
SELECT pid, productname, productdesc, price
FROM addagroproducts;

-- Create a view for the register table
CREATE VIEW register_view AS
SELECT farmername, adharnumber, phonenumbers, farming

```

```
FROM register;
```

DMLcommands.sql

```
INSERT INTO `addagroproducts`(`username`, `email`, `pid`, `productname`, `productdesc`, `price`) VALUES  
('agriexpert1', 'agriexpert1@gmail.com', 1, 'Organic Wheat', 'High-quality organic wheat harvested from pesticide-free fields.', '1 kg - ₹450'),  
('poultryfarmers', 'poultryfarmers@gmail.com', 2, 'Farm-Fresh Eggs', 'Fresh and nutritious eggs from free-range chickens.', '12 eggs - ₹60'),  
('milkman123', 'milkman123@gmail.com', 3, 'Goat Milk', 'Pure and fresh goat milk, rich in nutrients and proteins.', '500 ml - ₹200'),  
('dairyfarms', 'dairyfarms@gmail.com', 4, 'Cow Milk', 'Farm-fresh cow milk, pasteurized and free from additives.', '1 liter - ₹120'),  
('farmerlife', 'farmerlife@gmail.com', 5, 'Buffalo Milk', 'Naturally sourced buffalo milk with high cream content.', '1 liter - ₹150'),  
('plantnursery', 'plantnursery@gmail.com', 6, 'Fruit Saplings', 'Healthy saplings of various fruit trees for home gardens or farms.', '1 sapling - ₹80'),  
('indianveggies', 'indianveggies@gmail.com', 7, 'Organic Tomatoes', 'Locally grown organic tomatoes, rich in flavor and nutrients.', '1 kg - ₹40'),  
('greenfieldfarm', 'greenfieldfarm@gmail.com', 8, 'Fresh Spinach', 'Nourishing and pesticide-free spinach, handpicked from the farm.', '500g - ₹30'),  
('spicegrove', 'spicegrove@gmail.com', 9, 'Aromatic Spices Set', 'Assorted pack of aromatic Indian spices for culinary use.', '250g - ₹150'),  
('flowercultivator', 'flowercultivator@gmail.com', 10, 'Marigold Flowers', 'Freshly plucked marigold flowers, ideal for religious ceremonies.', '25 flowers - ₹25'),  
('honeyharvesters', 'honeyharvesters@gmail.com', 11, 'Pure Honey', 'Raw and unprocessed honey, collected from local bee farms.', '500g - ₹180'),  
('ricefarmers', 'ricefarmers@gmail.com', 12, 'Basmati Rice', 'Premium quality Basmati rice grains, known for their aroma.', '1 kg - ₹500'),  
('organicgrains', 'organicgrains@gmail.com', 13, 'Quinoa', 'Nutritious and gluten-free quinoa grains, perfect for a healthy diet.', '500g - ₹350'),  
('freshveggieshub', 'freshveggieshub@gmail.com', 14, 'Green Capsicum', 'Fresh and crunchy green capsicums grown using organic methods.', '500g - ₹60'),  
('tealeafplanters', 'tealeafplanters@gmail.com', 15, 'Assam Tea Leaves', 'High-quality Assam tea leaves, perfect for a refreshing cup of tea.', '250g - ₹250'),  
('spicychillies', 'spicychillies@gmail.com', 16, 'Red Chillies', 'Sun-dried spicy red chillies, essential for Indian cooking.', '100g - ₹45'),  
('organicfarmerindia', 'organicfarmerindia@gmail.com', 17, 'Healthy Millets Mix', 'A mix of nutrient-rich millets for a balanced diet.', '1 kg - ₹180'),  
('herbsharvest', 'herbsharvest@gmail.com', 18, 'Fresh Coriander', 'Fragrant and fresh coriander leaves, essential in Indian cuisine.', '50g - ₹20'),  
('nutrientrichfarms', 'nutrientrichfarms@gmail.com', 19, 'Chickpeas', 'High-protein and fiber-rich chickpeas for healthy meals.', '1 kg - ₹90'),  
('tumericgrowers', 'tumericgrowers@gmail.com', 20, 'Turmeric Powder', 'Ground turmeric powder, known for its medicinal properties.', '250g - ₹70'),
```

('mangoorchards', 'mangoorchards@gmail.com', 21, 'Alphonso Mangoes', 'Juicy and flavorful Alphonso mangoes, a summer delight.', '1 dozen - ₹80'),
 ('cashewplantations', 'cashewplantations@gmail.com', 22, 'Cashew Nuts', 'Premium-quality cashew nuts, handpicked and processed.', '250g - ₹300'),
 ('bengalifarmfresh', 'bengalifarmfresh@gmail.com', 23, 'Bengal Gram', 'Fresh Bengal gram, ideal for Indian culinary preparations.', '1 kg - ₹120'),
 ('localvegetablevendor', 'localvegetablevendor@gmail.com', 24, 'Potatoes', 'Locally sourced and fresh potatoes for various cooking needs.', '1 kg - ₹30'),
 ('grapevineyard', 'grapevineyard@gmail.com', 25, 'Seedless Grapes', 'Sweet and seedless grapes, perfect for snacking or desserts.', '500g - ₹100'),
 ('artisanalcheesery', 'artisanalcheesery@gmail.com', 26, 'Artisanal Cheese Set', 'Assorted set of artisanal cheeses for gourmet experiences.', '500g - ₹350'),
 ('localmeatshop', 'localmeatshop@gmail.com', 27, 'Fresh Lamb Meat', 'Tender and fresh lamb meat, sourced from local farms.', '1 kg - ₹400'),
 ('fisherydock', 'fisherydock@gmail.com', 28, 'Fresh Water Fish', 'Freshly caught local fish from freshwater sources.', '500g - ₹250'),
 ('poultryzone', 'poultryzone@gmail.com', 29, 'Dressed Chicken', 'Freshly dressed and cleaned chicken for cooking purposes.', '1 kg - ₹180'),
 ('dairyproductsindia', 'dairyproductsindia@gmail.com', 30, 'Butter', 'Homemade butter churned from farm-fresh cow milk.', '500g - ₹120');

```
INSERT INTO `farming` (`fid`, `farmingtype`) VALUES
(1, 'Seed Farming'),
(2, 'Cocoon Farming'),
(3, 'Silk Farming'),
(4, 'Goat Farming'),
(5, 'Cow Farming'),
(6, 'Buffalo Farming'),
(7, 'Sapling Cultivation'),
(8, 'Tomato Farming'),
(9, 'Spinach Farming'),
(10, 'Spice Cultivation'),
(11, 'Flower Cultivation'),
(12, 'Honey Harvesting'),
(13, 'Rice Farming'),
(14, 'Grain Cultivation'),
(15, 'Capsicum Farming'),
(16, 'Tea Leaf Cultivation'),
(17, 'Chilli Farming'),
(18, 'Millets Farming'),
(19, 'Coriander Farming'),
(20, 'Chickpea Farming'),
(21, 'Turmeric Farming'),
(22, 'Mango Orchard'),
(23, 'Cashew Plantation'),
(24, 'Bengal Gram Farming'),
(25, 'Potato Farming'),
(26, 'Grape Farming'),
```

```

(27, 'Cheese Production'),
(28, 'Lamb Farming'),
(29, 'Fishery'),
(30, 'Poultry Farming'),
(31, 'Butter Production');

INSERT INTO `register` (`rid`, `farmername`, `adhnarnumber`, `age`, `gender`,
`phonenumer`, `address`, `farming`) VALUES
(1, 'agriexpert1', '123456789012', 35, 'Male', '9876543210', 'Address 1', 'Crop
Cultivation'),
(2, 'poultryfarmers', '234567890123', 28, 'Female', '8765432109', 'Address 2',
'Poultry Farming'),
(3, 'milkman123', '345678901234', 40, 'Male', '7654321098', 'Address 3', 'Dairy
Farming'),
(4, 'dairyfarms', '456789012345', 32, 'Female', '6543210987', 'Address 4',
'Dairy Farming'),
(5, 'farmerlife', '567890123456', 45, 'Male', '5432109876', 'Address 5', 'Dairy
Farming'),
(6, 'plantnursery', '678901234567', 50, 'Female', '4321098765', 'Address 6',
'Horticulture'),
(7, 'indianveggies', '789012345678', 38, 'Male', '3210987654', 'Address 7',
'Crop Cultivation'),
(8, 'greenfieldfarm', '890123456789', 26, 'Female', '2109876543', 'Address 8',
'Crop Cultivation'),
(9, 'spicegrove', '901234567890', 31, 'Male', '1098765432', 'Address 9',
'Horticulture'),
(10, 'flowercultivator', '012345678901', 42, 'Female', '9876543210', 'Address
10', 'Horticulture'),
(11, 'honeyharvesters', '123456789012', 33, 'Male', '8765432109', 'Address 11',
'Apiculture'),
(12, 'ricefarmers', '234567890123', 29, 'Female', '7654321098', 'Address 12',
'Crop Cultivation'),
(13, 'organicgrains', '345678901234', 44, 'Male', '6543210987', 'Address 13',
'Organic Farming'),
(14, 'freshveggieshub', '456789012345', 36, 'Female', '5432109876', 'Address
14', 'Crop Cultivation'),
(15, 'tealeafplanters', '567890123456', 41, 'Male', '4321098765', 'Address 15',
'Horticulture'),
(16, 'spicychillies', '678901234567', 27, 'Female', '3210987654', 'Address 16',
'Crop Cultivation'),
(17, 'organicfarmerindia', '789012345678', 39, 'Male', '2109876543', 'Address
17', 'Organic Farming'),
(18, 'herbsharvest', '890123456789', 30, 'Female', '1098765432', 'Address 18',
'Horticulture'),
(19, 'nutrientrichfarms', '901234567890', 43, 'Male', '0987654321', 'Address
19', 'Crop Cultivation'),
(20, 'tumericgrowers', '012345678901', 34, 'Female', '9876543210', 'Address
20', 'Crop Cultivation'),
(21, 'mangoorchards', '123456789012', 37, 'Male', '8765432109', 'Address 21',
'Horticulture'),

```

```

(22, 'cashewplantations', '234567890123', 25, 'Female', '7654321098', 'Address
22', 'Agroforestry'),
(23, 'bengalifarmfresh', '345678901234', 48, 'Male', '6543210987', 'Address
23', 'Crop Cultivation'),
(24, 'localvegetablevendor', '456789012345', 31, 'Female', '5432109876',
'Address 24', 'Crop Cultivation'),
(25, 'grapevineyard', '567890123456', 46, 'Male', '4321098765', 'Address 25',
'Horticulture'),
(26, 'artisanalcheesery', '678901234567', 29, 'Female', '3210987654', 'Address
26', 'Dairy Farming'),
(27, 'localmeatshop', '789012345678', 35, 'Male', '2109876543', 'Address 27',
'Livestock Rearing'),
(28, 'fisherydock', '890123456789', 40, 'Female', '1098765432', 'Address 28',
'Aquaculture'),
(29, 'poultryzone', '901234567890', 33, 'Male', '0987654321', 'Address 29',
'Poultry Farming'),
(30, 'dairyproductsindia', '012345678901', 42, 'Female', '9876543210', 'Address
30', 'Dairy Farming');

```

DQLcommands.sql

```

--DQL with where conditions

--Select all products with a specific username from the addagroproducts table:
SELECT * FROM addagroproducts WHERE username = 'example_username';

--Retrieve farming types with a specific ID from the farming table:
SELECT * FROM farming WHERE fid = 1;

--Find farmers with a specific Aadhar number from the register table:
SELECT * FROM register WHERE adharnumber = '123456789012';

--Get information about a specific test from the test table:
SELECT * FROM test WHERE id = 1;

--Retrieve triggers associated with a specific farmer ID from the trig table:
SELECT * FROM trig WHERE fid = 'example_farmer_id';

--Select user information for a specific username from the user table:
SELECT * FROM user WHERE username = 'example_user';

--Retrieve users who are associated with agriculture (is_agrarian = TRUE) from
the user table:
SELECT * FROM user WHERE is_agrarian = TRUE;

--Find users who have admin privileges (is_admin = TRUE) from the user table:
SELECT * FROM user WHERE is_admin = TRUE;

```

```
--DQL with aggregate functions

--Calculate the average price of products in the addagroproducts table:
SELECT AVG(price) AS average_price FROM addagroproducts;

--Count the number of farming types in the farming table:
SELECT COUNT(*) AS farming_type_count FROM farming;

--Find the maximum age of farmers in the register table:
SELECT MAX(age) AS max_age FROM register;

--Calculate the total number of tests in the test table:
SELECT COUNT(*) AS total_tests FROM test;

--Find the earliest timestamp among triggers in the trig table:
SELECT MIN(timestamp) AS earliest_timestamp FROM trig;

--Calculate the total number of users in the user table:
SELECT COUNT(*) AS total_users FROM user;
```

```
--Calculate the sum of ages for all farmers in the register table:
SELECT SUM(age) AS total_age FROM register;
```

```
--Find the product with the highest price in the addagroproducts table:
SELECT * FROM addagroproducts WHERE price = (SELECT MAX(price) FROM
addagroproducts);
```

```
--DQL of simple join operations
```

```
--Join addagroproducts and register tables to get products along with farmer
details:
SELECT addagroproducts.*, register.farmername, register.age
FROM addagroproducts
JOIN register ON addagroproducts.username = register.username;
```

```
--Join farming and register tables to get farmers along with their associated
farming types:
SELECT register.*, farming.farmingtype
FROM register
JOIN farming ON register.farming = farming.fid;
```

```
--Join trig and register tables to get triggers along with farmer details:
SELECT trig.*, register.farmername
FROM trig
JOIN register ON trig.fid = register.rid;
```

```
--Join user and register tables to get user details along with farmer
registration information:
SELECT user.*, register.address
```

```

FROM user
JOIN register ON user.username = register.username;

--Join addagroproducts, register, and farming tables to get products along with
farmer details and farming types:
SELECT addagroproducts.*, register.farmername, farming.farmingtype
FROM addagroproducts
JOIN register ON addagroproducts.username = register.username
JOIN farming ON register.farming = farming.fid;

--DQL of Subqueries

--Find farmers who have registered and are associated with agriculture in the
register table:
SELECT *
FROM register
WHERE farming IN (SELECT farmingtype FROM farming);

--Find the products with a price higher than the average price in the
addagroproducts table:
SELECT *
FROM addagroproducts
WHERE price > (SELECT AVG(price) FROM addagroproducts);

--Retrieve the farming type associated with the youngest farmer in the register
table:
SELECT farmingtype
FROM farming
WHERE fid = (SELECT rid FROM register ORDER BY age ASC LIMIT 1);

--Get the number of products listed by each user in the addagroproducts table:
SELECT username, COUNT(pid) AS product_count
FROM addagroproducts
GROUP BY username;

--Find users who are not associated with agriculture in the user table:
SELECT *
FROM user
WHERE username NOT IN (SELECT username FROM register WHERE farming IS NOT
NULL);

--Functions

--function that retrieves the total number of registered farmers for a given
farming type from the provided tables (register and farming tables). The
function will take the farming type as an input parameter and return the count.

-- Create the farming_count function

```

```

DELIMITER //

CREATE FUNCTION farming_count(farming_type_input varchar(200))
RETURNS INT
BEGIN
    DECLARE count_result INT;

    -- Use a subquery to count the number of farmers with the given farming
    type
    SELECT COUNT(*) INTO count_result
    FROM register
    WHERE farming = (SELECT fid FROM farming WHERE farmingtype =
    farming_type_input);

    -- Return the count
    RETURN count_result;
END //

DELIMITER ;

-- Example usage of the farming_count function
SELECT farming_count('Type1') AS farmer_count_for_type1;

--function that calculates the average age of farmers based on a given farming
type.

-- Create the average_age_for_farming_type function
DELIMITER //

CREATE FUNCTION average_age_for_farming_type(farming_type_input varchar(200))
RETURNS DECIMAL(10,2)
BEGIN
    DECLARE avg_age_result DECIMAL(10,2);

    -- Use a subquery to calculate the average age of farmers with the given
    farming type
    SELECT AVG(age) INTO avg_age_result
    FROM register
    WHERE farming = (SELECT fid FROM farming WHERE farmingtype =
    farming_type_input);

    -- Return the average age
    RETURN avg_age_result;
END //

DELIMITER ;

-- Example usage of the average_age_for_farming_type function
SELECT average_age_for_farming_type('Type1') AS avg_age_for_type1;

```

```

--Stored procedures

--a stored procedure that calculates and returns the total value of all
products for a given user in the addagroproducts table

-- Create the calculate_total_value stored procedure
DELIMITER //

CREATE PROCEDURE calculate_total_value(IN username_input VARCHAR(50), OUT
total_value DECIMAL(10,2))
BEGIN
    -- Calculate the total value of all products for the specified user
    SELECT SUM(price) INTO total_value
    FROM addagroproducts
    WHERE username = username_input;
END //

DELIMITER ;

-- Example usage of the calculate_total_value stored procedure
CALL calculate_total_value('example_username', @total_value);
SELECT @total_value AS total_product_value;

--a stored procedure that retrieves the details of farmers who have a specified
minimum age and are associated with a specific farming type. The procedure will
take the minimum age and farming type as input parameters and return the
matching farmer details.

-- Create the get_farmers_by_age_and_type stored procedure
DELIMITER //

CREATE PROCEDURE get_farmers_by_age_and_type(IN min_age INT, IN
farming_type_input VARCHAR(200))
BEGIN
    -- Retrieve farmers who meet the specified criteria
    SELECT *
    FROM register
    WHERE age >= min_age AND farming = (SELECT fid FROM farming WHERE
farmingtype = farming_type_input);
END //

DELIMITER ;

-- Example usage of the get_farmers_by_age_and_type stored procedure
CALL get_farmers_by_age_and_type(30, 'Type1');

```