Farm2Market App – System Documentation

# Introduction

Farm2Market is a web-based platform designed to connect farmers with customers directly, allowing for the sale of agricultural products without intermediaries. The system features a role-based access model where farmers can register and manage their products, customers can place orders, and administrators oversee approvals and platform integrity. The application uses a Django-based backend connected to a MySQL database, and a fully developed frontend (HTML, CSS, JavaScript).

# Problem Statement

In many agricultural regions, farmers face difficulty reaching buyers due to the presence of middlemen and lack of digital tools. Farm2Market solves this problem by allowing farmers to sign up, get approved, and list their products for direct sale to customers. The platform ensures secure registration, farmer verification, and product management while streamlining the order and delivery process.

# User Roles and Access Control

## 1. Farmer

- Registers through the sign-up form (username and email only).  
- Waits for admin approval.  
- Upon approval, receives an auto-generated password via email.  
- Logs in using username and password.  
- Manages own products (add/edit/delete).  
- Views orders containing their products.

## 2. Customer

- Registers and logs in using username and password.  
- Browses available products.  
- Adds products to cart or directly places orders.  
- Views past orders and their statuses.

## 3. Administrator

- Reviews and approves/rejects farmer applications.  
- Triggers password generation and email on approval.  
- Manages users and product listings if needed.

# Product Management

- Products are managed by farmers.  
- Attributes include name, description, price, stock, and optionally an image.  
- Products are visible to customers only if the farmer is approved and stock is available.

# Order Management

- Customers place orders with specified product quantities.  
- Orders contain: customer info, product list, total price, date/time, and status.  
- Uses a join table (OrderItem) to handle many-to-many product-order relationships with quantities.  
- Farmers can view orders involving their products.

# Authentication and Security

- Farmers do not set passwords on sign-up.  
- Passwords are securely generated and emailed post-approval.  
- Login uses username and password (not email).  
- All sensitive actions are CSRF-protected and validated.

# Email Notification System

- Configured via SMTP.  
- Emails are sent upon farmer approval with credentials.  
- Optional future notifications include order updates and password reset.

# System Architecture

- Frontend: HTML, CSS, JavaScript  
- Backend: Django  
- Database: MySQL  
- Email: SMTP (e.g., Gmail)  
- Auth: Session-based with password hashing

# Data Relationships

- Farmer 1─< Product  
- Customer 1─< Order  
- Order M─< OrderItem >─M Product

# Optional Future Features

- Product categories and search  
- Online payment integration  
- Delivery tracking  
- Reviews and ratings  
- Admin dashboards and analytics