Project 16

Maxi Fine Foods Online Grocery Management System

The Grocery Management System is developed to facilitate the work of the grocery storekeeper and the customer who buys the products. This system consists of the methods which can relate the perpetual crisis while managing the rhetoric quandaries concerning the authoritarian source. After the technological advancement in every sector, people are utilizing more and more machines which are deleterious for the environment and engendering hazardous situations for human lives.

Because of the sudden increase in the overall world's pollution index now it's not safe to buy victuals items which are making and selling in the open area or the street pabulum. This pabulum can make a person sick, he may suffer some diseases if he orally consumes the pollution-contaminated aliment.

It is rigorously exhorted to buy pabulum items either packed or from the stores which are not in an open area and following hygienic rules. We require aliment daily consequently to buy grocery is a customary task which people incline to do weekly or monthly as per their location and availability. The proposed system will avail the customer to opt for the type of grocery items he requires and forward the list to the admin then after the admin suggest a team and pack the items and then the distribution person distributes them to the doorstep of the customer.

The customer doesn't need to visit the store, cull the items from the shelves, and then to buy groceries. Now he can just get it by the avail of the system, he just needs to access his account or if he is an incipient user then first he requires registering himself into the system so that the data of him is perpetually stored in the database of the system. It will be auxiliary when the customer authenticates again then he doesn't need to lodge his details because the system can retrieve it within the database quandary.

The system is very active and covers all the necessary options which are needed by the admin and customer even if there is some error that occurs in the system then it sends an alert mail to the admin or the concerning department. The following are the concerning points of this system which are considered:

- The system is proactive and performs the task which is necessary sometimes even for the troubleshoots.
- The clause of this system handles many customers at the same time and provides smooth functioning with a better and enhanced user interface.
- The admin can perform tasks to handle the customer as well as the staff of the store.
- The authorized person can edit, delete, and modify the sustaining database as and when required.
- It monitors the actions of the customer and suggests him better products from the system interface.

Obsolete System:

People need to buy the grocery at a fine-tuned interval of time, there are some items which can be purchased for utilizing it more than a week but withal there are many items which need to be utilized on the same day or within 2-3 days like dairy products, etc. if the customer is a working person then he has to first schedule his timing to go for grocery shopping then after making the list of sundry items, he goes to the market where he visits many shops to buy items because all items are not available in the same street shop.

This process requires adscititious physical effort and time to buy the grocery, it's a very tiring and hectic method plus a person needs to reiterate it every week or month as per the requisite. Sometimes due to lack of time people buy victuals items from nearby street shops of the roadside areas then after consuming they get sick and many diseases follow it.

The owner keeps the record of the transaction on papers and then he forefends the papers punctiliously but there are many chances of error to occur while maintaining the records additionally if someone wants to find any record then he has to check all the papers. If there is any kind of error occurs afore then it is very arduous to track the error on the papers.

Apart from maintaining the record, there is so much other information which the owner needs to take care and indicted it down somewhere consequently a systematic management system is needed to clear out all the possible quandaries as well as managing the day to day transaction and data. The sundry points which are weakening the current system are:

- The information or the data recorded in the papers are less trustable because there may be so many problems occur while managing the record papers.
- This system contains many errors and there is no method through which we can track the error if any occurs.
- The owner cannot keep an eye on every transaction that happens in a day with the customer.
- This system has also required the efforts of the customer as he has to come to the store and then select items and buy them himself.
- To keep the information of the store staff, it requires great efforts and time to write down each staffs data.

Required System:

A centralized system is needed to store the data and information of every customer and staff in an automated manner so that we can retrieve the data whenever required. This system is created by keeping in mind the difficulties and problems of the owner and the customer as well. This system is error-free, reliable, and efficient enough to control the flow of data and store it in a systematic manner.

The system bifurcates the data in product type, size, variety, quantity, brand manner also there are many types of items available in the groceries like wheat, rice, pulses, sugar, and other household edible items.

It can easily run all the work simultaneously without error and technical glitches. The admin customer and the staff can work after logging into their respective accounts. The interface of all three is given according to the need and requirement of the entity it has separated options as well.

The admin controls the data of the customers and also the staff member's data along with their salary structure, the designation of every staff. It provides full control of the system to the admin.

Here the customer can select the items and save them in the online cart in the system and pays the estimated amount through online options after receiving the request the admin gathers the staff of particular racks and instruct them to pack those items, thereafter the delivery boy deliver the items at the customer's residence as per his given time and date. The following are the key points of this system:

- All the data is in an automated form where we can track any information if required.
- The employees can see the data of the respective customer if there's any error occurs and reported by the customer.
- It is a unified system that can work simultaneously and saves data according to the department and account in the database.
- There are separate sections through which the customer can find any item or product by entering the title of it.
- The admin can track the delivery person and guide him towards the customer's residence.

Module Description:

The model itself clears the relation among the entities of the system which are internal link and provide data if we access one data, the relevant files can be open through the link explained in this diagram. So this model is the hard wire which transfuses the hierarchy and flow of data internally. It helps the admin to manage the sundry accounts and provides a list of solutions if any error occurs in the system. It can also help in tracking the error from which entity it has occurred.

1. User Management Module

- Handles user registration, login and profile management.
- Supports three user roles: Admin, Customer and Delivery Staff.
- Includes email verification, password reset and session management.
- Enforces access control for different roles.

2. Product Management Module

- Allows admin to add, edit, delete and categorise products.
- Displays products with names, images, prices and descriptions.
- Automatically hides out-of-stock items and updates inventory after orders.
- Includes search and filter functionality.

3. Shopping Cart and Checkout Module

- Manages customer shopping carts and calculates totals in real time.
- Handles item additions, removals and quantity changes.
- Supports discount codes and promotional offers.
- Finalises checkout with order summary and payment selection.

4. Order Management Module

- Records and tracks customer orders from placement to delivery.
- Supports order ID generation, status tracking and cancellation.
- Updates stock levels automatically after order confirmation.
- Maintains order logs for auditing and reporting.

5. Payment Management Module

- Integrates secure payment gateways (e.g. PayPal, Stripe).
- Supports Cash on Delivery and Online Payment.
- Verifies payment completion and generates digital receipts.
- Stores all transaction data securely in the database.

6. Delivery Management Module

- Assigns and monitors delivery staff for each order.
- Allows delivery staff to log in, view assigned deliveries and update status.
- Sends real-time delivery updates to customers.
- Displays estimated delivery time and completion tracking.

7. Notification and Feedback Module

- Sends automated notifications for registration, order placement and delivery.

- Collects customer ratings and reviews after order completion.
- Displays feedback summaries for admin analysis.

8. Reporting and Analytics Module

- Generates daily, weekly, and monthly reports for sales, deliveries and performance.
- Displays visual charts for top products and active customers.
- Provides exportable reports (PDF/CSV).

9. Automation Module

- Automates tasks such as:
    - Hiding out-of-stock products.
    - Sending email confirmations.
    - Cancelling unpaid orders after a timeout.
    - Updating stock after confirmed orders.
    - Generating scheduled reports.

Functional Requirements

Functional requirements are product features or functions that developers must implement to enable users to accomplish their tasks. So, it’s important to make them clear for the stakeholders. Generally, functional requirements describe system behavior under specific conditions. The developers of this system must enhance the performance and efficiency of the system by adding 15 to 20 more functional requirements. Students need to do their own research to find how they can improve the system and which FRs need to added. The group must need a prior approval from the stakeholders/project supervisor before finalizing these Functional Requirements. These enhanced FRs must be reflected separately in Final SRS Report after the approval.

User Registration and Authentication

FR1. The system should allow users to register by entering their full name, email, contact number and password.
FR2. The system should verify the user’s email before activating the account.
FR3. The system should allow users to log in using a valid email and password.
FR4. The system should allow users to reset their password through email verification.
FR5. The system should allow administrators to view, edit or delete user accounts.
FR6. The system should manage three user roles: Admin, Customer and Delivery Staff (each with different access privileges.)

Product and Category Management

FR7. The system should allow admins to add new products with name, description, price, image and stock quantity.
FR8. The system should allow admins to categorise products (e.g. Grocery, Electronics, Fashion etc.).
FR9. The system should allow admins to edit or delete existing products.
FR10. The system should allow customers to view products with images, prices and descriptions.
FR11. The system should allow customers to search for products by name or category.
FR12. The system should automatically hide or mark products as *Out of Stock* when inventory reaches zero.

Shopping Cart and Checkout

FR13. The system should allow customers to add multiple items to their shopping cart.
FR14. The system should allow customers to modify product quantities or remove items from the cart.
FR15. The system should display subtotal, taxes, and total payable amount in real time.
FR16. The system should allow customers to proceed to checkout after reviewing their cart.

FR17. The system should store order details after checkout, including items, quantities and prices.
FR18. The system should notify customers of successful order placement through email or SMS.

Order Management

FR19. The system should allow customers to view their current and past orders in their dashboard.
FR20. The system should allow customers to cancel orders that are still in Pending status.
FR21. The system should assign a unique order ID for each order placed.
FR22. The system should allow admins to update order status (e.g. Pending, Processing, Out for Delivery, Delivered, Cancelled).
FR23. The system should allow admins to reassign orders to different delivery staff if required.
FR24. The system should automatically update inventory levels after order confirmation.
FR25. The system should maintain a log of all order-related changes (status updates, cancellations etc.)

Payment Management

FR26. The system should allow customers to choose payment methods like Cash on Delivery or Online Payment.
FR27. The system should validate payment information before confirming the order.
FR28. The system should integrate with third-party payment gateways (e.g. PayPal, Stripe).
FR29. The system should store transaction details securely for admin reference.
FR30. The system should display payment confirmation and generate an electronic receipt.

Delivery Management

FR31. The system should allow admins to assign orders to delivery staff.
FR32. The system should allow delivery staff to log in and view assigned deliveries.
FR33. The system should allow delivery staff to update delivery status in real time.
FR34. The system should allow customers to view delivery progress and estimated delivery time.
FR35. The system should allow admins to track all deliveries on a dashboard (filter by area, status, or staff).

Notifications and Feedback

FR36. The system should send automated email/SMS alerts for order confirmation, dispatch and delivery completion.
FR37. The system should allow customers to submit ratings and feedback for delivered orders.
FR38. The system should allow admins to view feedback and respond to customer complaints.

Reporting and Analytics

FR39. The system should generate daily, weekly and monthly reports on total sales, revenue and number of deliveries.
FR40. The system should display analytical charts showing top-selling products, delivery performance and customer activity.

FR41 Auto-hide out-of-stock products

FR42 Auto email confirmation

FR43 Auto stock update

FR44 Auto order timeout cancellation

FR45 Auto weekly report generation

FR46 The system should provide an automated chatbot assistant to help customers with common queries such as: Where is my order? Or How do I reset my password?

FR47 The system should analyse customer feedback sentiment (positive, neutral, negative) to help the admin monitor service satisfaction levels.

(You can use API for sentiment analysis (e.g. using TextBlob or NLTK).

Non-Functional Requirements

1. Performance Requirements

NFR1. The system should handle a minimum of 100 concurrent users without performance degradation.
NFR2. Each page or query response should load within 3 seconds under normal conditions.
NFR3. The database should process basic CRUD operations (Create, Read, Update, Delete) in under 2 seconds.
NFR4. The system should maintain efficient indexing in the MySQL database to ensure fast retrieval of product and order data.

2. Reliability Requirements

NFR5. The system should automatically back up data at least once every 24 hours.
NFR6. System downtime for maintenance should not exceed 2 hours per month.
NFR7. The system should recover from hardware or software failures within 15 minutes using backup data.

3. Security Requirements

NFR8. All passwords should be stored using encryption techniques (e.g. bcrypt or MD5 with salt).
NFR9. All data transmitted between client and server should use HTTPS for encryption.
NFR10. The system should implement input validation and sanitisation to prevent SQL injection and XSS attacks.
NFR11. User sessions should automatically expire after 15 minutes of inactivity.

NFR12. Role-based access control should restrict unauthorised access to administrative features.

4. Usability Requirements

NFR13. The user interface should be intuitive, consistent, and responsive across desktop, tablet and mobile devices.
NFR14. Navigation menus and buttons should be clearly labelled with descriptive text and tooltips.
NFR15. The system should provide confirmation messages for all key user actions (e.g. order placed, payment received).

5. Scalability Requirements

NFR16. The system architecture should allow integration of additional modules such as Vendor Portal or Multi-Store Support without major redesign.
NFR17. The MySQL database should be normalised to at least 3NF to support large datasets efficiently.
NFR18. The system should be capable of scaling horizontally (adding more servers) to handle increased traffic.

6. Maintainability Requirements

NFR19. The system should use modular PHP code, separating frontend, backend and database logic.
NFR20. All functions and classes should be documented using PHPDoc comments.
NFR21. The project should be version-controlled using Git to track changes and enable rollback when necessary.

7. Availability Requirements

NFR22. The system should be operational 24/7 maintaining at least 99% uptime.
NFR23. The system should display a maintenance message when the server is temporarily offline.

8. Portability & Compatibility Requirements

NFR24. The system should run on any platform supporting PHP 8.0+ and MySQL 8.0+.
NFR25. The system should be compatible with modern browsers including Google Chrome, Mozilla Firefox, Microsoft Edge and Safari.

9. Data Integrity & Accuracy Requirements

NFR26. The system should ensure data consistency between related tables using foreign-key constraints.
NFR27. Automated scripts should validate order totals and payment amounts to prevent calculation errors.

10. Audit & Logging Requirements

NFR28. The system should log all critical actions (user logins, payments, stock updates, deliveries) with timestamps and user IDs.
NFR29. Administrators should be able to view or export activity logs in CSV or PDF format for auditing purposes.

Hardware Requirement: Should be recommended by the developers.
Software Requirement: Should be recommended by the developers.