Design Document

for

Campus Craves

Version 0.1

Prepared by

Group 19 **Group Name: Supervised Learners**

Name	Roll no.	E-mail
Aayush Gautam	220020	aayushg22@iitk.ac.in
Abhishek Kumar	210040	abhikumar21@iitk.ac.in
Anand Chutani	220130	anandc22@iitk.ac.in
Anany Dev Choudhary	220135	ananydc22@iitk.ac.in
Chatla Sowmya Sri	200293	srisowmya20@iitk.ac.in
Lokesh Mehra	220591	lokeshm22@iitk.ac.in
Pranshu Mani Tripathi	220800	pranshumt22@iitk.ac.in
Siddharth Pathak	211034	siddharthp21@iitk.ac.in
Sidharth A S	221056	sidharthas22@iitk.ac.in
Sparsh Gupta	221084	gsparsh22@iitk.ac.in
Ujjwal Kumar	211123	ujjwalk21@iitk.ac.in

Course: CS253

Mentor TA: Mr. Ashish Singh

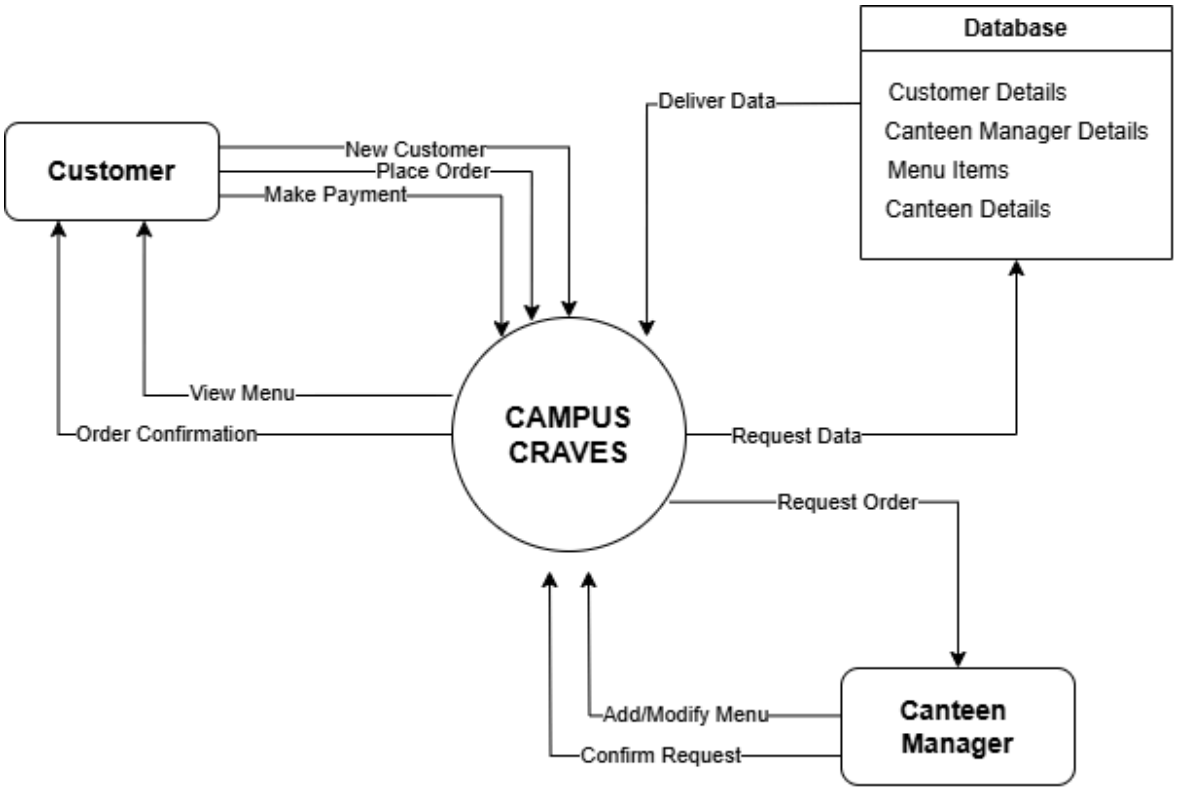
Date: 7 February, 2025

Revisions

Version	Primary Author(s)	Description of Version	Date Completed
v0.1	Abhishek Kumar Chatla Sowmya Sri Siddharth Pathak Sparsh Gupta Ujjwal Kumar	Initial Commit (First Draft)	07/02/2025

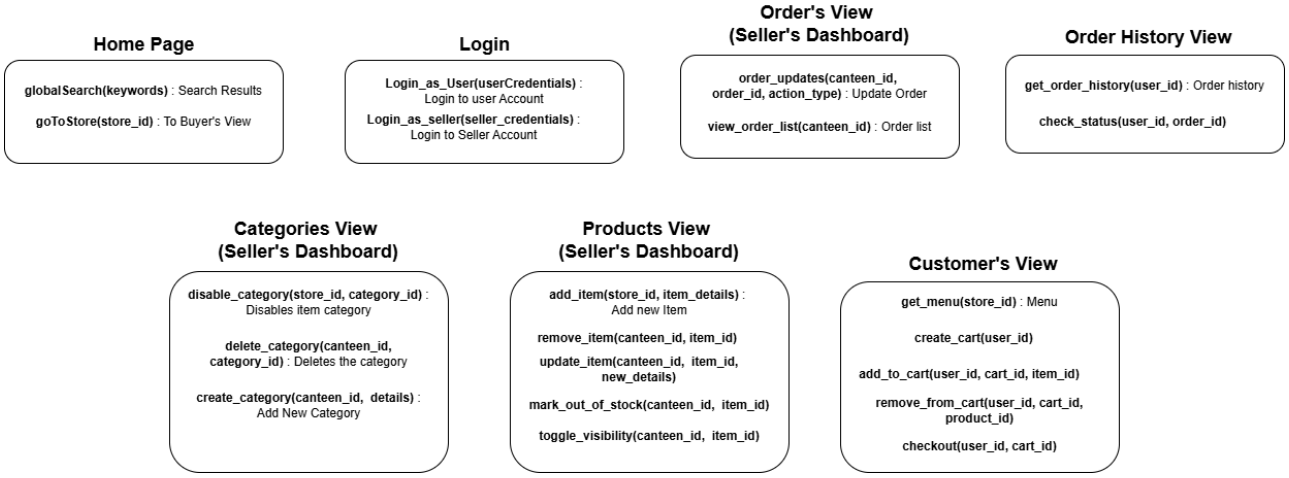
1 Context Design

1.1 Context Model

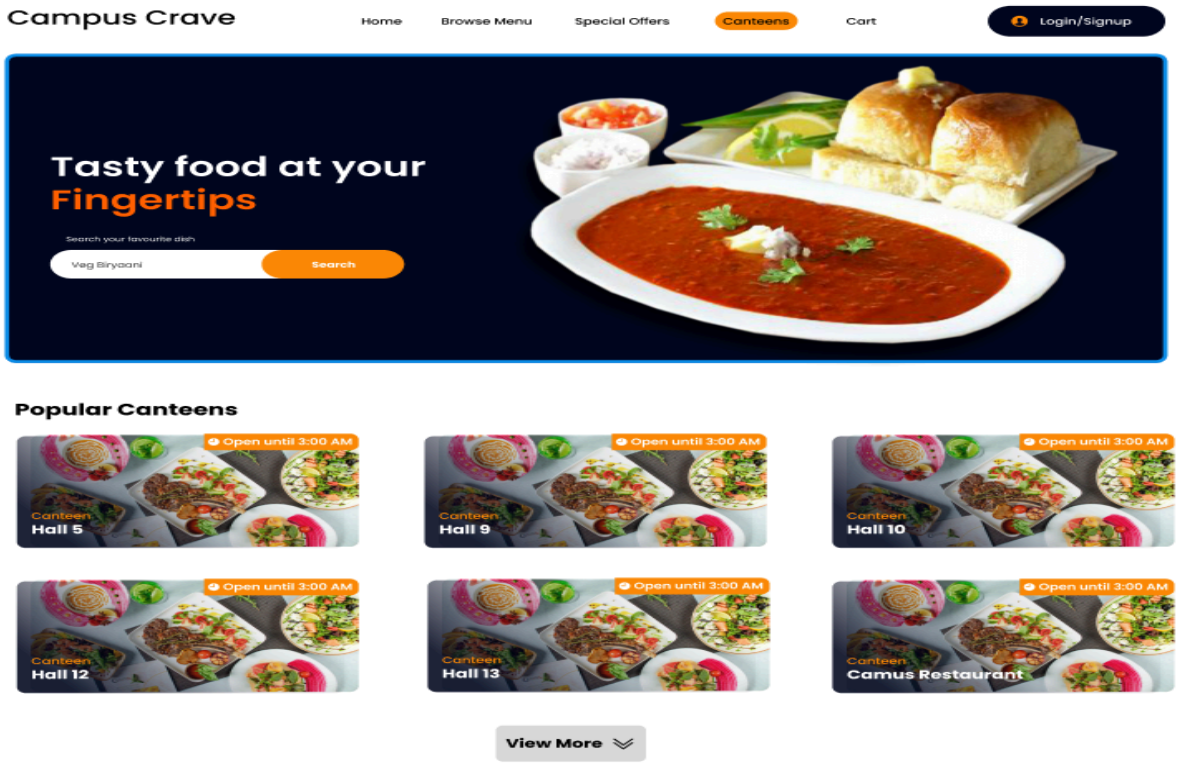


1.2 Human Interface Design

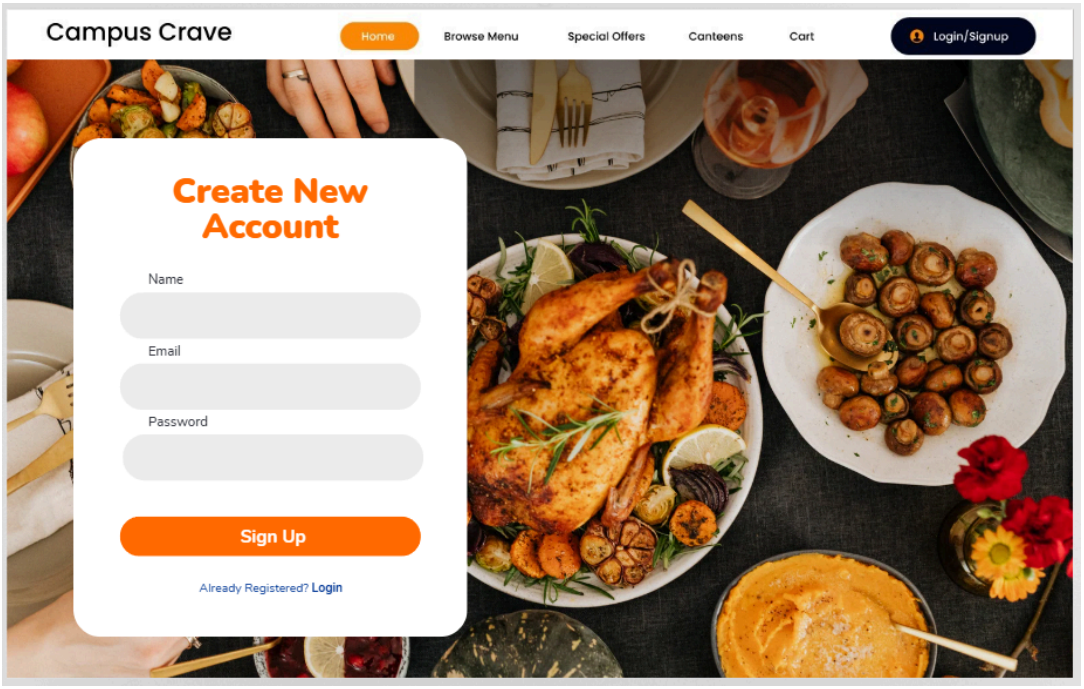
There are seven interfaces in our web-app, each having its own utility. Each interface is meant to be used for either buyers or sellers and facilitates a smooth user experience with the clients. The Software Design Document for Page 2 interface design of **Campus Craves** with all seven interfaces is given below.



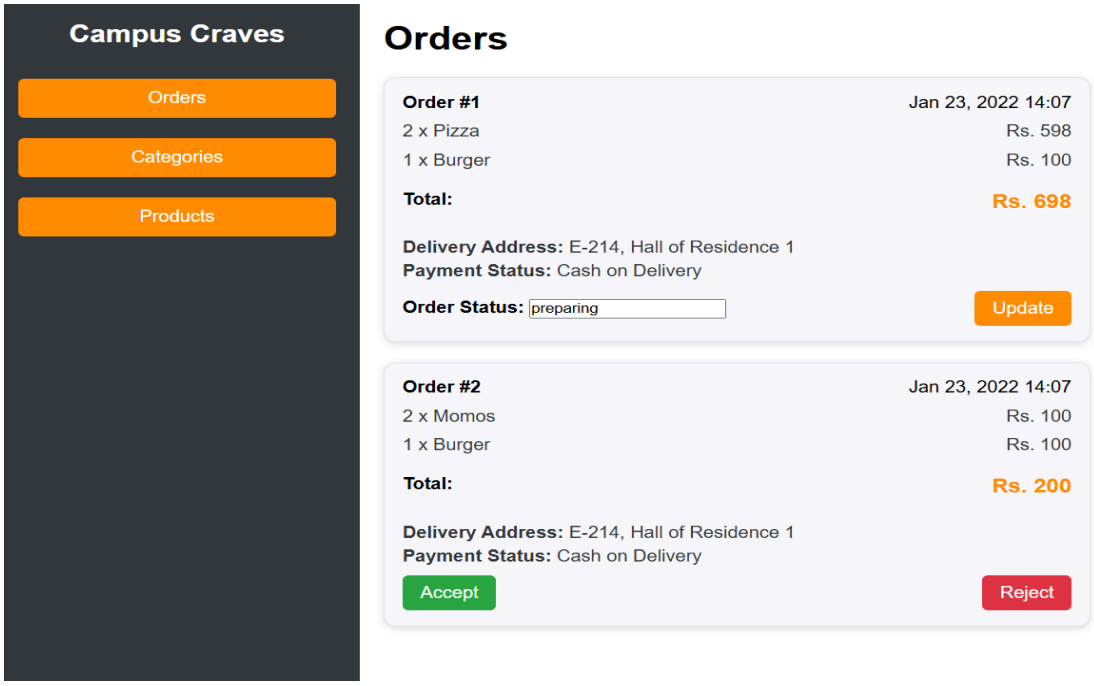
Home Page



Login Page

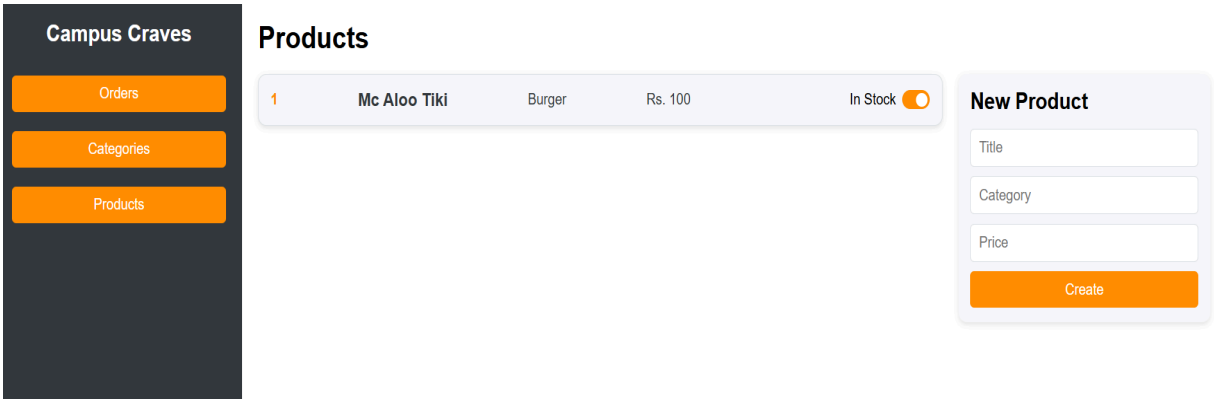


Orders' view (Seller's dashboard)

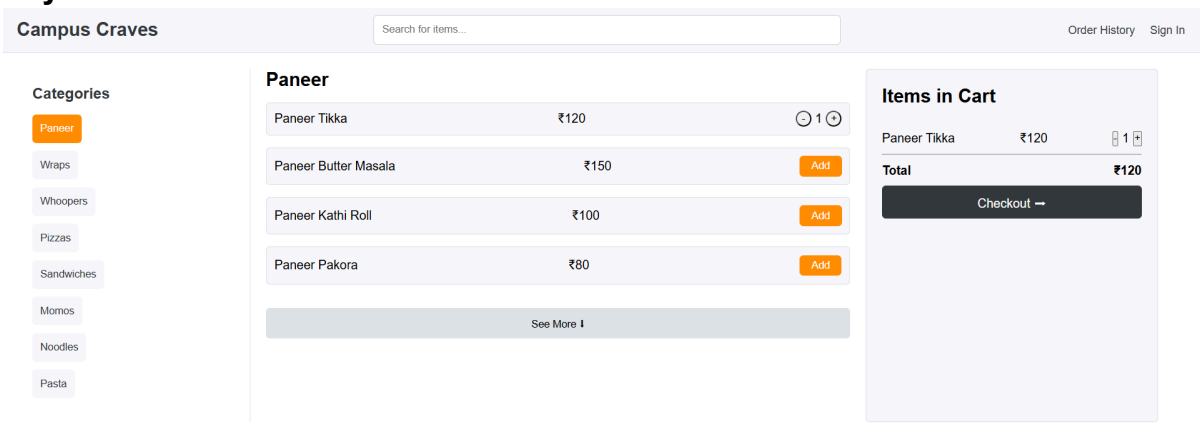


Categories' view (Seller's Dashboard)

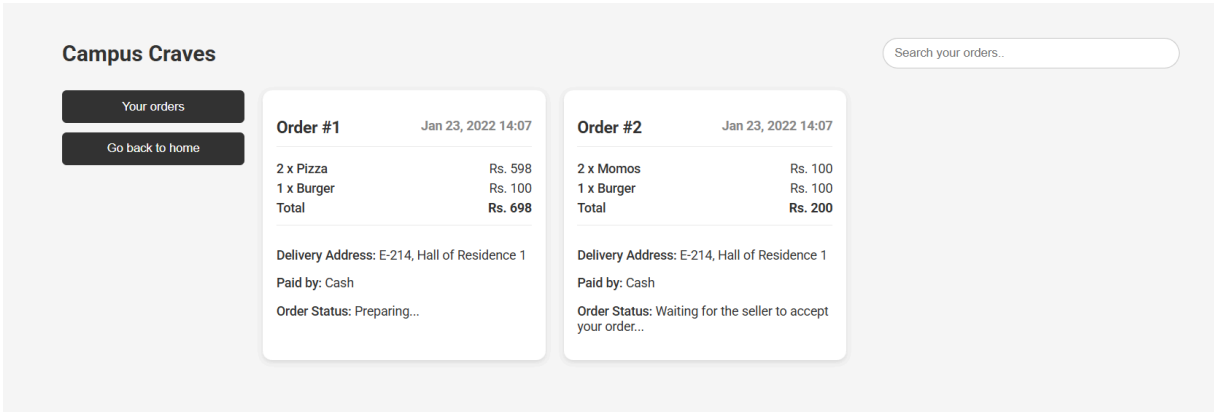
Products' view (Seller's dashboard)



Buyer's view



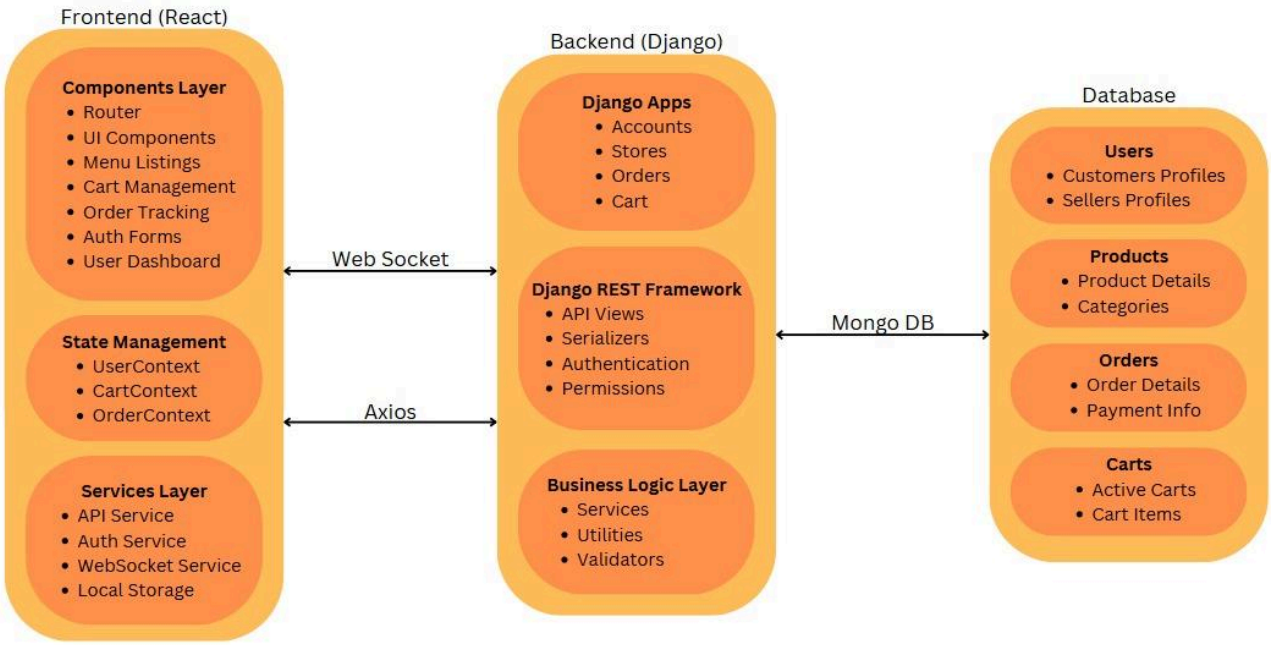
Order History view



2 Architecture Design

"Campus Craves" is a web-based application implementing a Model-View-Controller (MVC) architecture.

Architecture Diagram



Intuition for Meeting Non-Functional Attributes

The proposed MVC architecture is well-suited to handle the non-functional requirements outlined in the SRS document:

1. Performance & Scalability

- MVC manages high user traffic through:
  - Independent optimization of components (Model for data, View for rendering).
  - MongoDB's horizontal scaling supports 500+ concurrent users.
  - Response times maintained under 2 seconds through efficient data flow.

Notable Challenge: Real-time order updates need additional WebSocket integration beyond basic MVC structure.

2. Security Framework

- The architecture implements robust security through:
  - Role-based access control keeps buyers' and sellers' data secure.
  - Django’s security features (CSRF protection, password hashing) add extra safety.

Notable Challenge: Third-party payment gateway integration needs additional security measures.

3. System Reliability

- Reliability is ensured through:

- A modular design makes testing and debugging easier.
- Automated testing ensures 99.9% order processing accuracy.

Notable Challenge: Inventory updates need optimization with background tasks.

4. User Experience & Compatibility

- The system delivers optimal usability through:
  - Component-based UI design ensures smooth navigation.
  - Cross-platform compatibility (works well on both mobile and desktop).
  - Containerized deployment supporting multiple cloud platforms.

Conclusion:

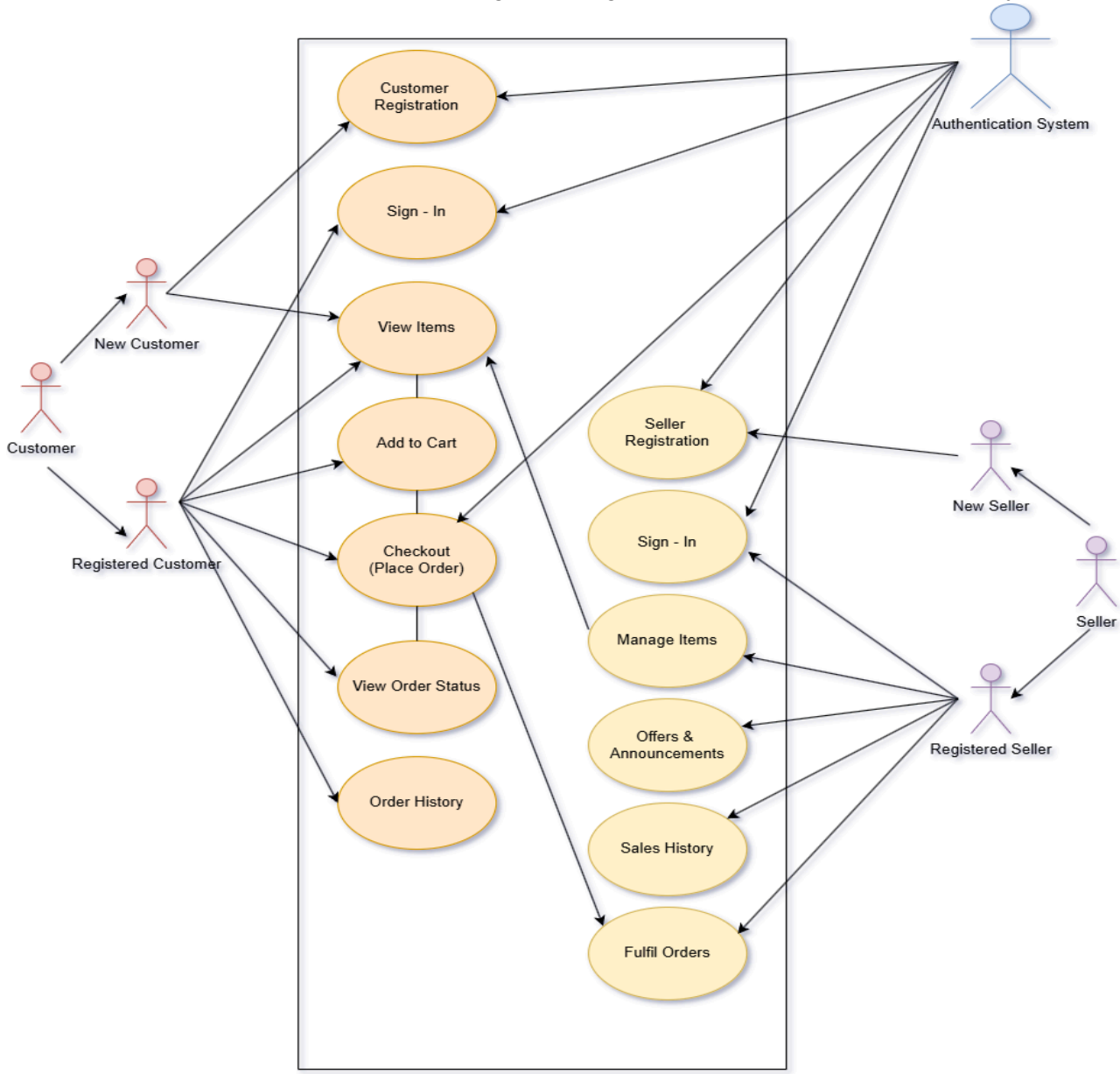
The proposed architecture effectively meets the non-functional requirements outlined in the SRS document. However, to ensure long-term performance, security, and reliability, continuous monitoring, regular updates, and proactive improvements will be essential as the system scales to accommodate a growing student population.

3 Object Oriented Design

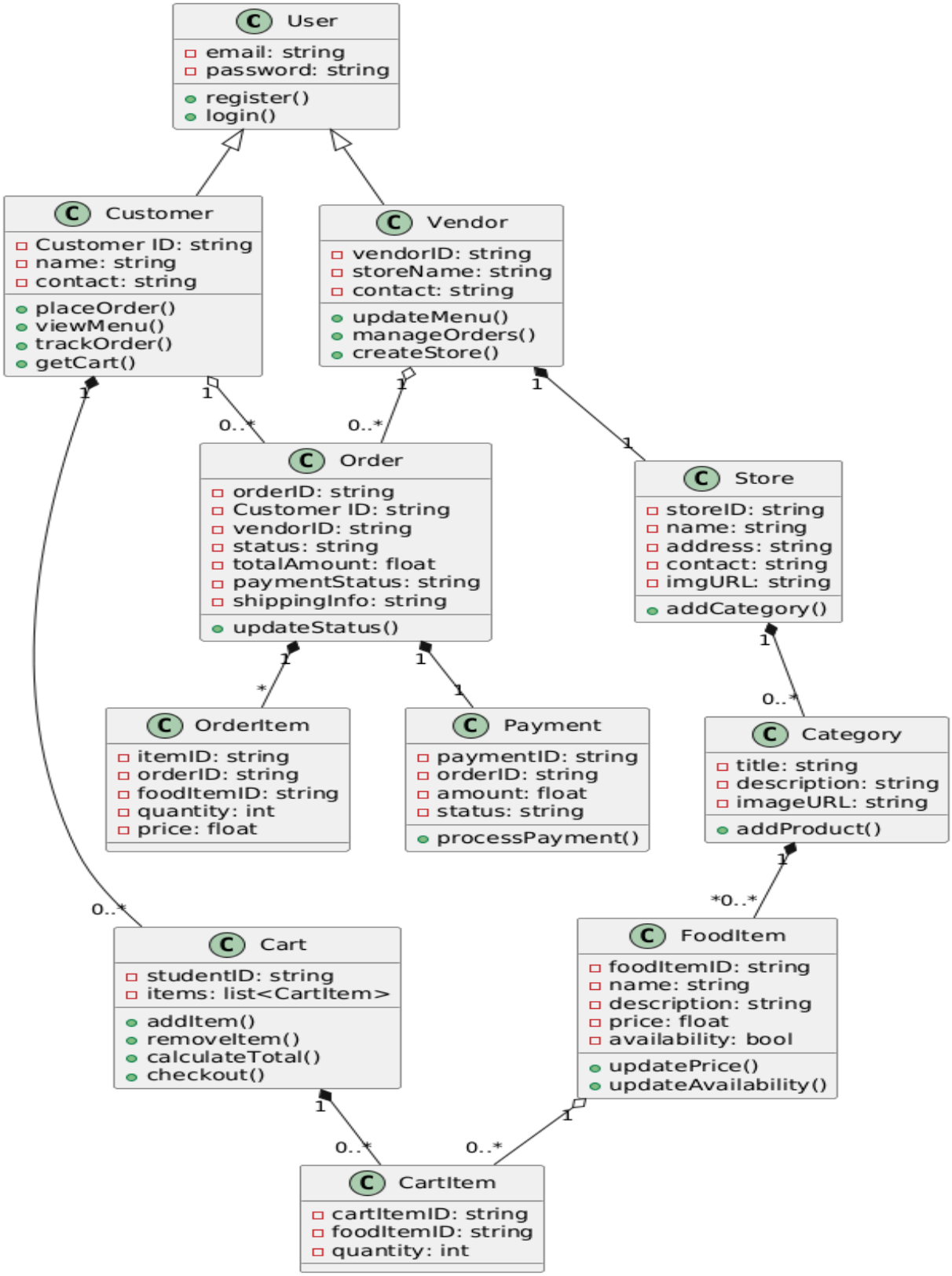
3.1 Use Case Diagrams

We have given a consolidated diagram showing different use cases provided:

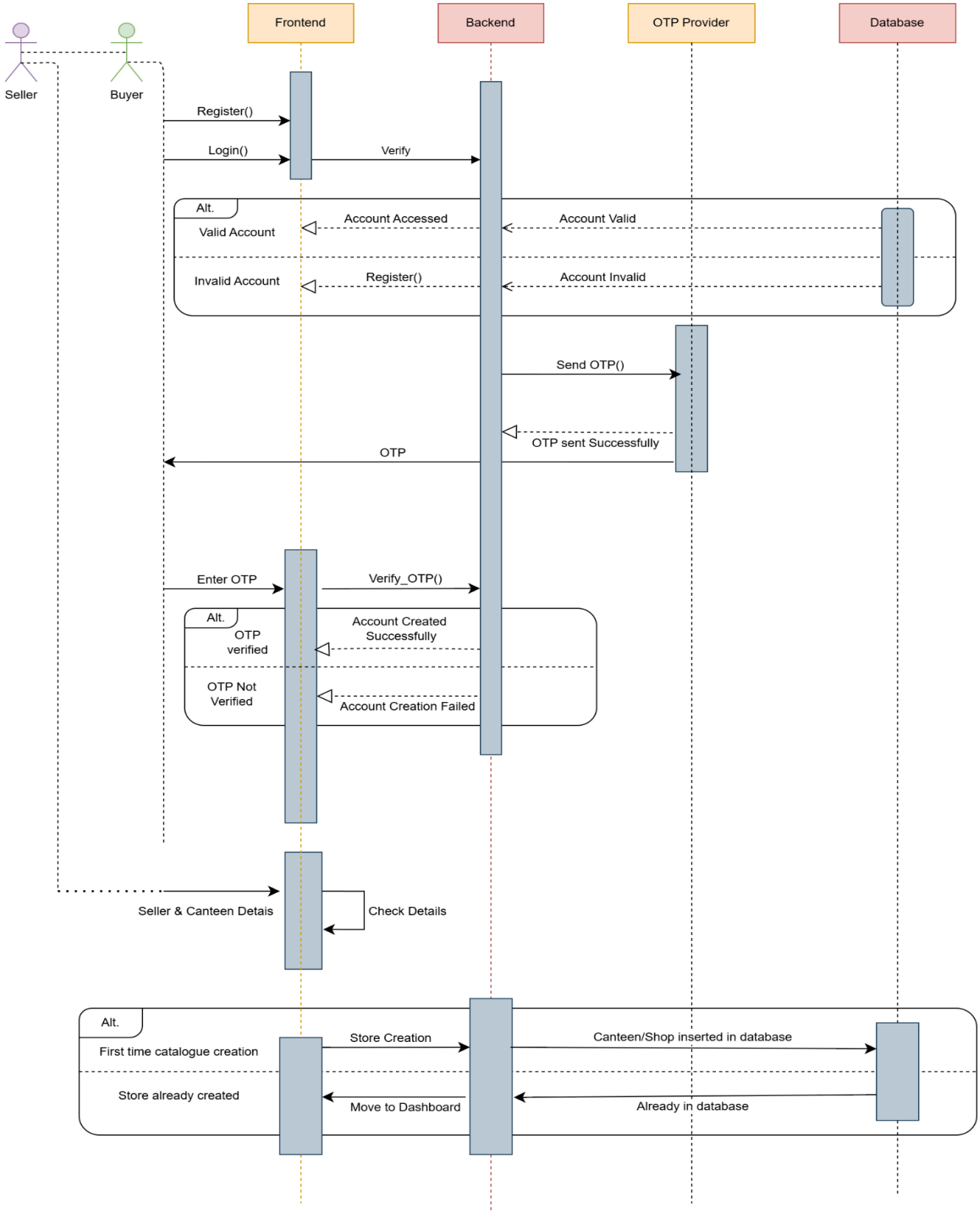
- Shopkeeper/canteen vendor registration and catalog creation.
- Browsing catalog of a particular shop/canteen.
- Items being sold by the canteen vendor and short descriptions and listed prices.
- Adding items to cart.
- Register as a customer using email and OTP verification.
- Checking out and placing orders.
- Shopkeeper/canteen vendors adding/removing items from shop/canteen inventory.

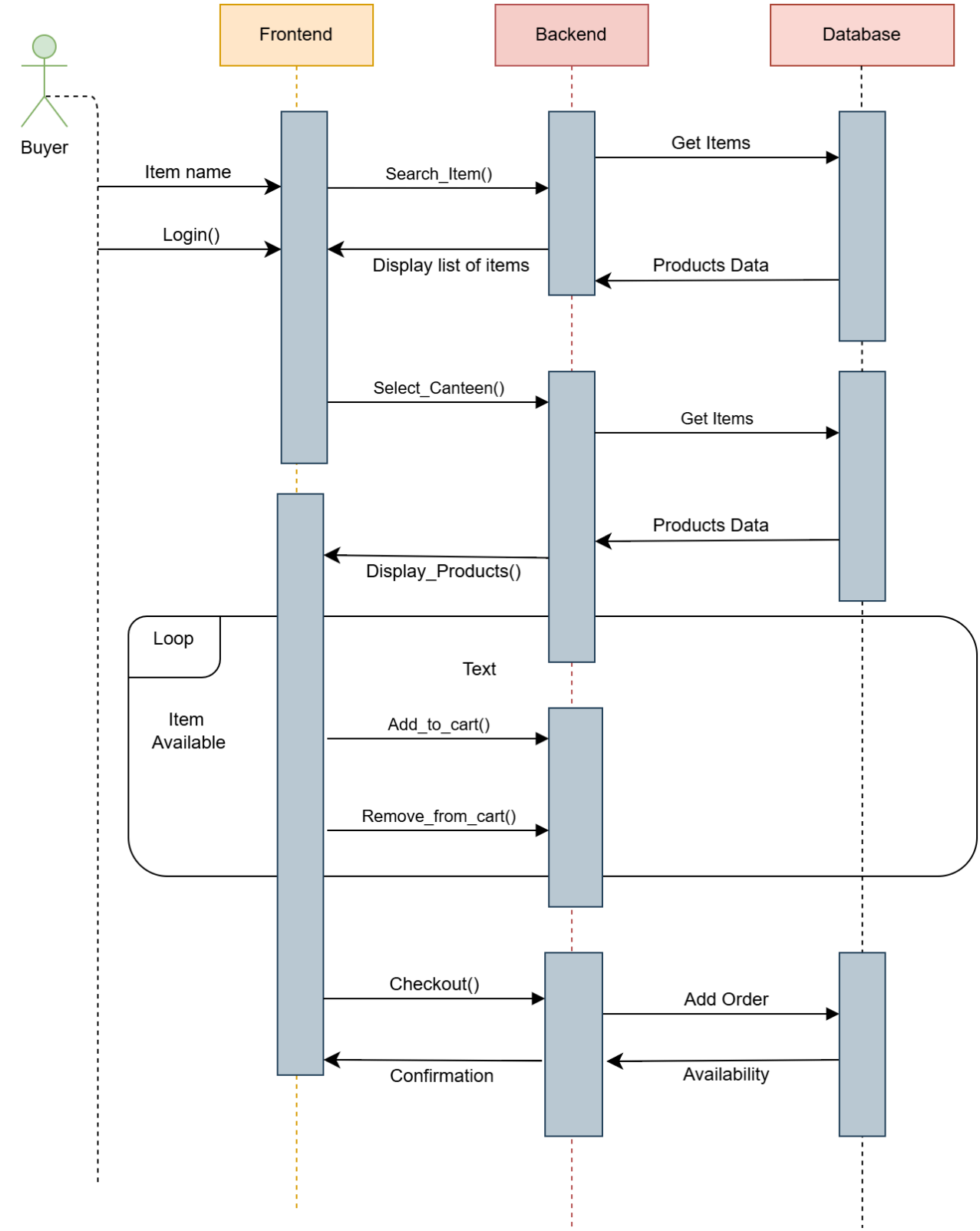


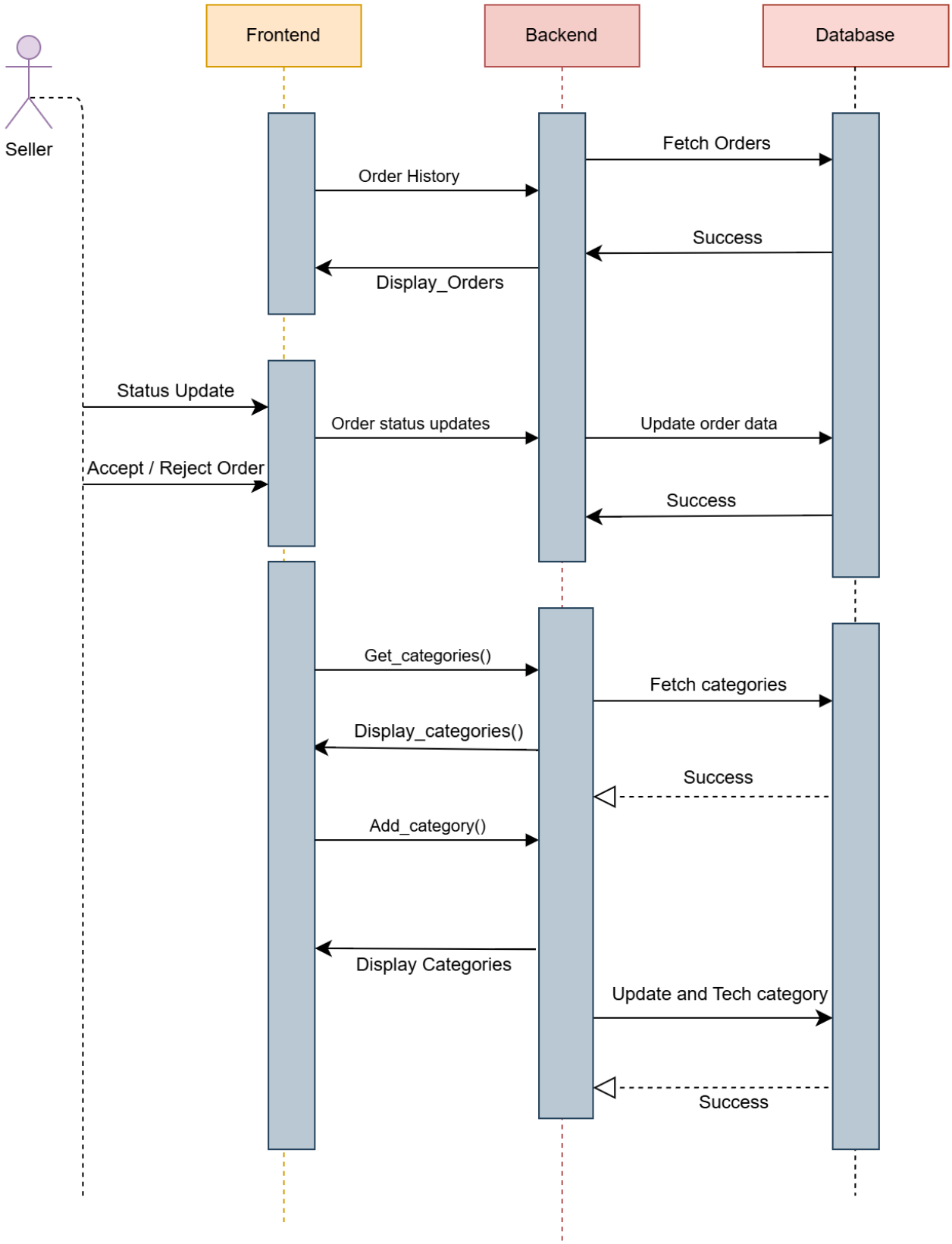
3.2 Class Diagrams

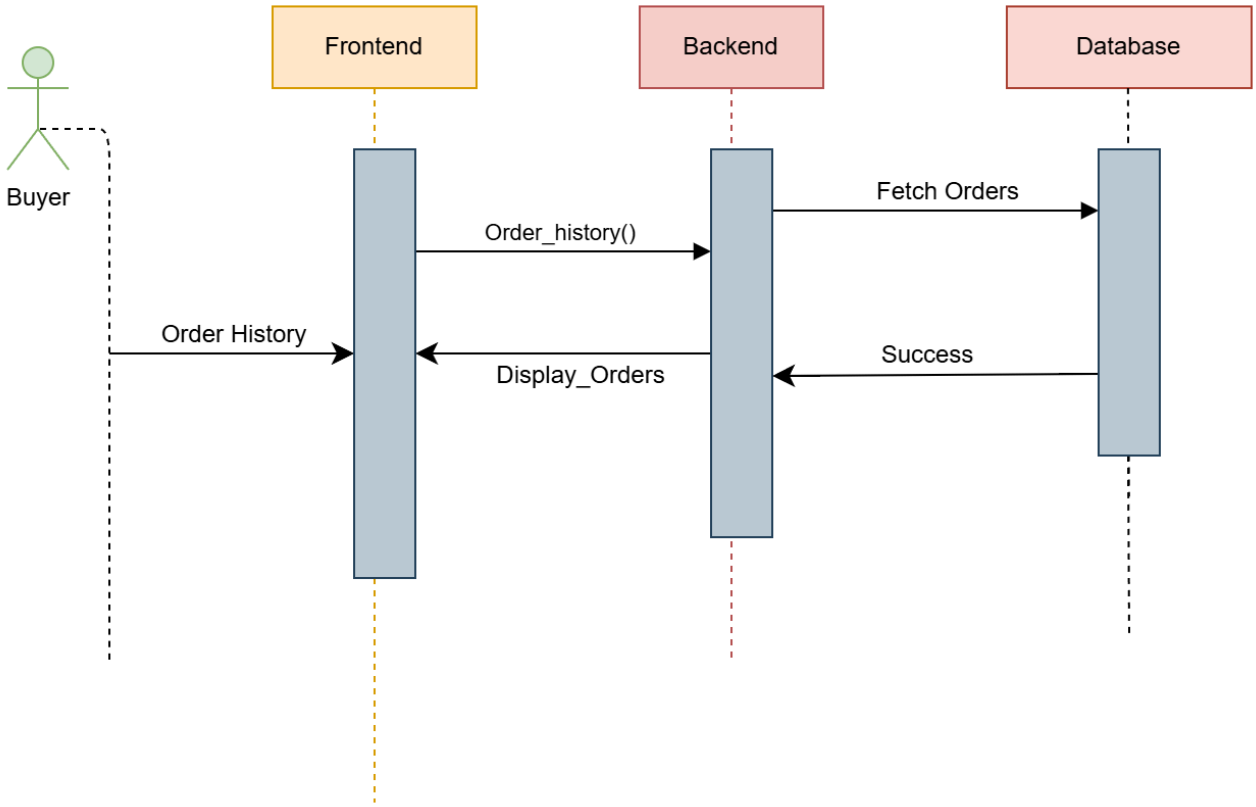


3.3 Sequence Diagrams

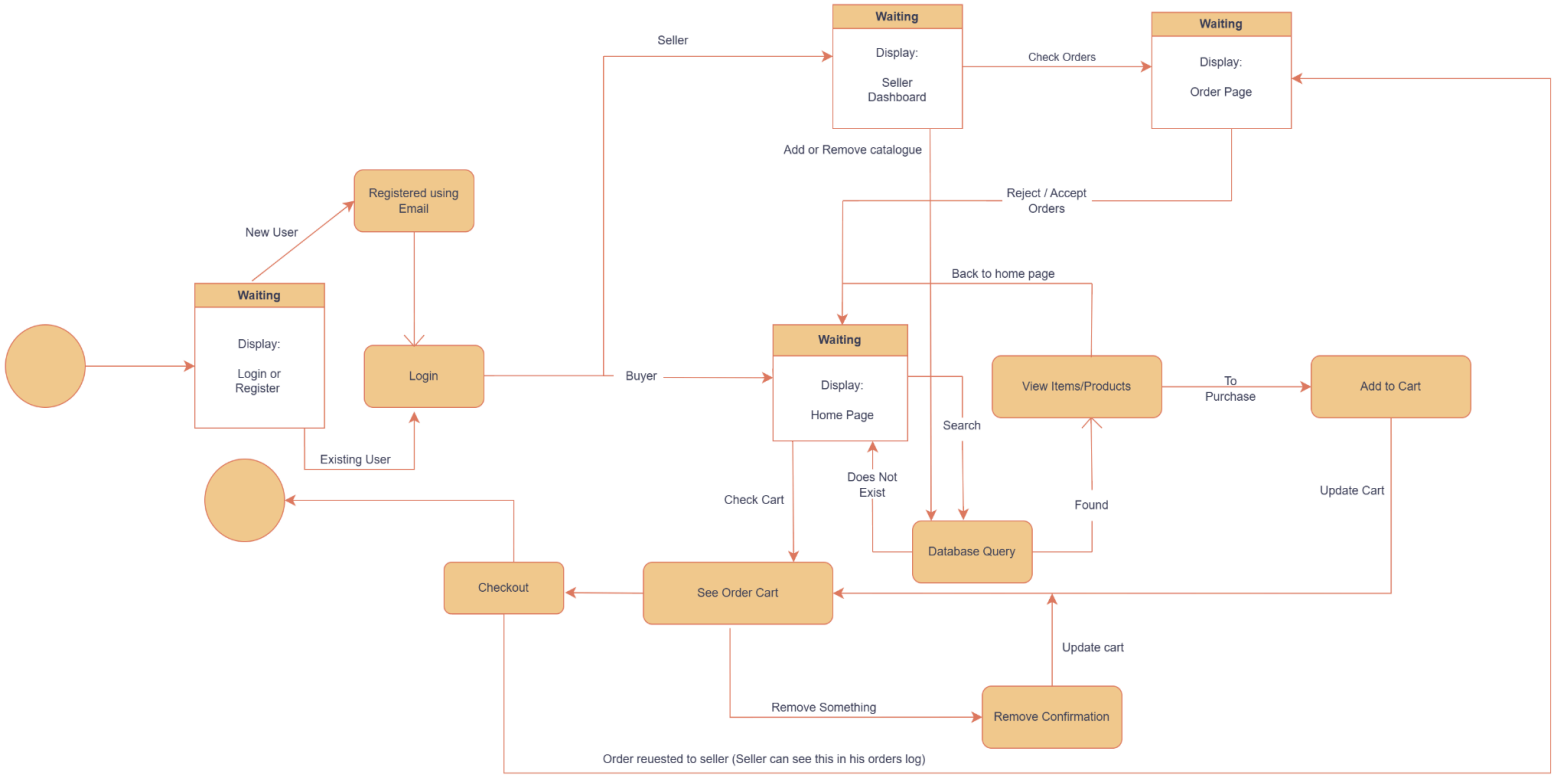



Frontend
Backend
Database
Buyer
Item name
Search_Item()
Get Items
Login()
Display list of items
Products Data
Select_Canteen()
Get Items
Products Data
Display_Products()
Loop
Text
Item
Available
Add_to_cart()
Remove_from_cart()
Checkout()
Add Order
Confirmation
Availability


Seller
Frontend
Backend
Database
Order History
Fetch Orders
Success
Display_Orders
Status Update
Order status updates
Update order data
Accept / Reject Order
Success
Get_categories()
Fetch categories
Display_categories()
Success
Add_category()
Display Categories
Update and Tech category
Success


Frontend
Backend
Database
Buyer
Order_history()
Fetch Orders
Order History
Display_Orders
Success

3.4 State Diagrams



4 Project Plan

This project plan outlines the detailed phases and tasks for developing the "Campus Craves" web application.

4.1 Implementation Plan

4.1.1 UI/UX Design Phase: (11 Jan - 27 Jan)

- **User research and Requirements Gathering:**
  - Description: Understand student/canteen needs for online food ordering at IIT Kanpur. Analyze existing menus/pricing.
  - Timeline: 3-5 days
  - Team members: Pranshu Mani Tripathi, Chatla Sowmya Sri
- **Wireframing and Prototyping:**
  - Description: Develop wireframes and interactive prototypes focusing on student ordering, canteen order management, and product catalog. Ensure prototypes are optimized for mobile use and accessibility.
  - Timeline: 3-5 days
  - Team members: Abhishek Kumar, Siddharth Pathak, Ujjwal Kumar
- **UI/UX Design Finalization:**
  - Description: Finalize UI designs. Create mockups for all screens including student home, product browsing and seller dashboards.
  - Timeline: 3-5 days
  - Team members: Abhishek Kumar, Siddharth Pathak

4.1.2 Backend Development Phase: (8 Feb - 28 Feb)

- **Database Schema:**
  - Description: Designing the structure for storing canteen, menu, user, and order data.
  - Timeline: 3-5 days
  - Team members: Sparsh Gupta, Pranshu Mani Tripathi
- **Backend Server and Database Setup:**
  - Description: Set up Django backend server and connect to MongoDB. Configure necessary security measures, including user authentication, authorization, and data encryption.
  - Timeline: 3-5 days
  - Team members: Sparsh Gupta, Pranshu Mani Tripathi
- **CRUD Implementation:**
  - Description: Implement CRUD (Create, Read, Update, Delete) operations for managing canteen data, product catalogs, user accounts, and orders. Ensure efficient data retrieval and storage.
  - Timeline: 3-5 days
  - Team members: Abhishek Kumar, Siddharth Pathak, Lokesh Mehra
- **API Endpoint Development:**
  - Description: Develop RESTful APIs for frontend interaction. Handle student requests and seller requests.
  - Timeline: 3-5 days
  - Team members: Abhishek Kumar, Siddharth Pathak
- **Business Logic:**
  - Description: Implement order processing, real-time updates, search, payment and review

calculations.
  ○ Timeline: 3-5 days
  ○ Team members: Abhishek Kumar, Siddharth Pathak
- **Handling security considerations:**
  ○ Description: Implement user authentication and authorization, data validation, password hashing, and protection against common web vulnerabilities.
  ○ Timeline: 3-5 days
  ○ Team members: Abhishek Kumar, Siddharth Pathak
- **Third-Party Integrations:**
  ○ Description: Integrate third-party services for email notifications (order confirmations, status updates), payment gateway integration, and campus directory service for user authentication.
  ○ Timeline: 3-5 days
  ○ Team members: Abhishek Kumar, Siddharth Pathak

4.1.3 Frontend Development Phase: (28 Feb - 14 March)

- **Development Environment Setup:**
  ○ Description: Set up the frontend development environment using React.js, including necessary libraries, components, and tools.
  ○ Timeline: 3-5 days
  ○ Team members: Ujjwal Kumar, Pranshu Mani Tripathi
- **Frontend Architecture Development:**
  ○ Description: Develop a modular frontend architecture for the application, focusing on reusable components for canteen listings, product displays, cart management, and order history.
  ○ Timeline: 3-5 days
  ○ Team members: Ujjwal Kumar, Chatla Sowmya Sri
- **UI Design Implementation:**
  ○ Description: Translating UI/UX designs into functional interfaces using HTML, CSS, and JavaScript.
  ○ Timeline: 3-5 days
  ○ Team members: Ujjwal Kumar, Chatla Sowmya Sri
- **Implementing Responsive Design:**
  ○ Description: Implement a responsive design to ensure the application is usable across various devices (desktops, tablets, smartphones) and screen sizes.
  ○ Timeline: 3-5 days
  ○ Team members: Ujjwal Kumar, Chatla Sowmya Sri
- **User Authentication and Authorization UI:**
  ○ Description: Develop user authentication interfaces (login, signup, password reset) and implement role-based access control to restrict access to certain features based on user roles (student, seller).
  ○ Timeline: 3-5 days
  ○ Team members: Pranshu Mani Tripathi, Chatla Sowmya Sri
- **Feature UI Implementation:**
  ○ Description: Develop the user interfaces for key features: canteen listings, product search and filtering, cart management, order placement, order tracking, seller dashboards (order management, product catalog updates).
  ○ Timeline: 3-5 days
  ○ Team members: Ujjwal Kumar, Chatla Sowmya Sri

4.1.4 Integration Phase: (14 March - 22 March)

• Integrate the frontend and backend components, ensuring seamless data flow and communication between the client and server.
• Team members: All team members

4.1.5 Testing Phase: (22 March - 28 March)

• Conduct unit testing of individual components, integration testing of frontend and backend interactions, and usability testing with representative users to identify and fix any bugs or usability issues.
• Team members: All team members

4.1.6 Final Testing Phase: (1 April - 17 April)

• Perform comprehensive testing, including beta testing with campus users, performance testing to ensure the system can handle a large number of concurrent users, and security audits to identify and address any vulnerabilities.
• Team members: All team members

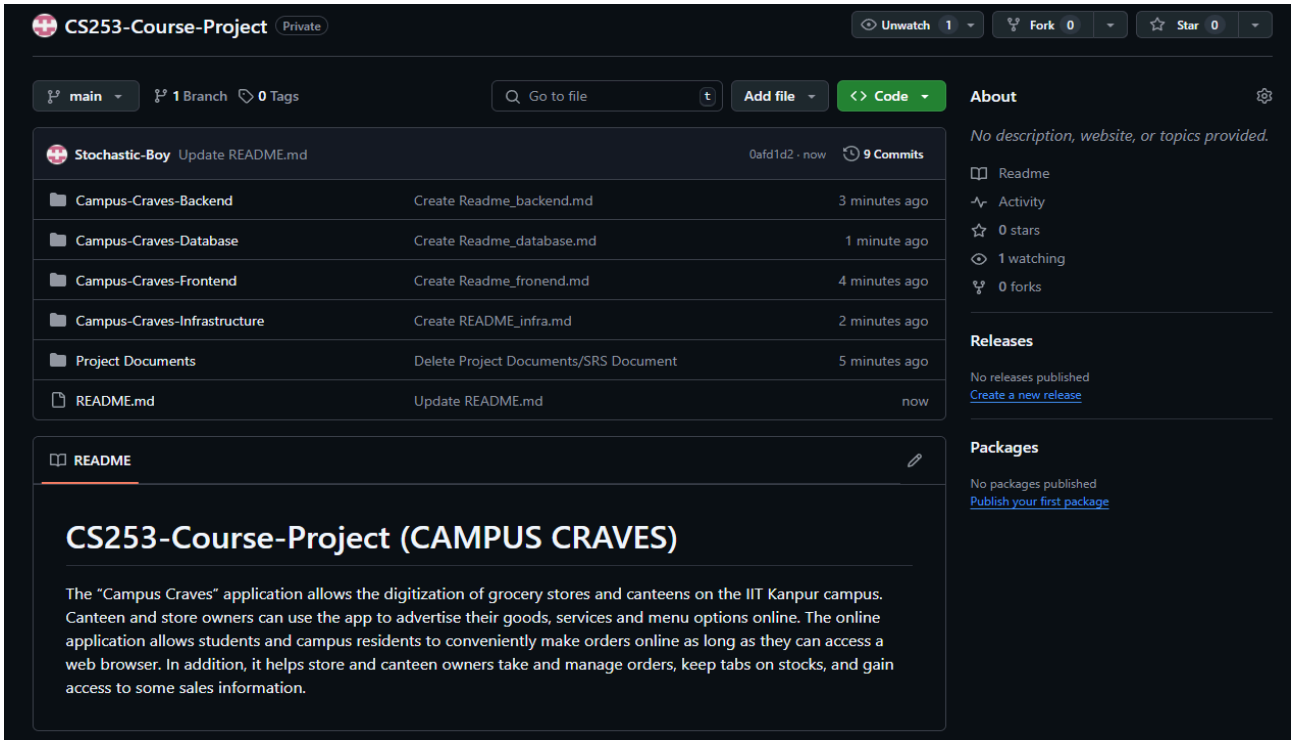
4.2 Team Collaboration and Communication for Implementation

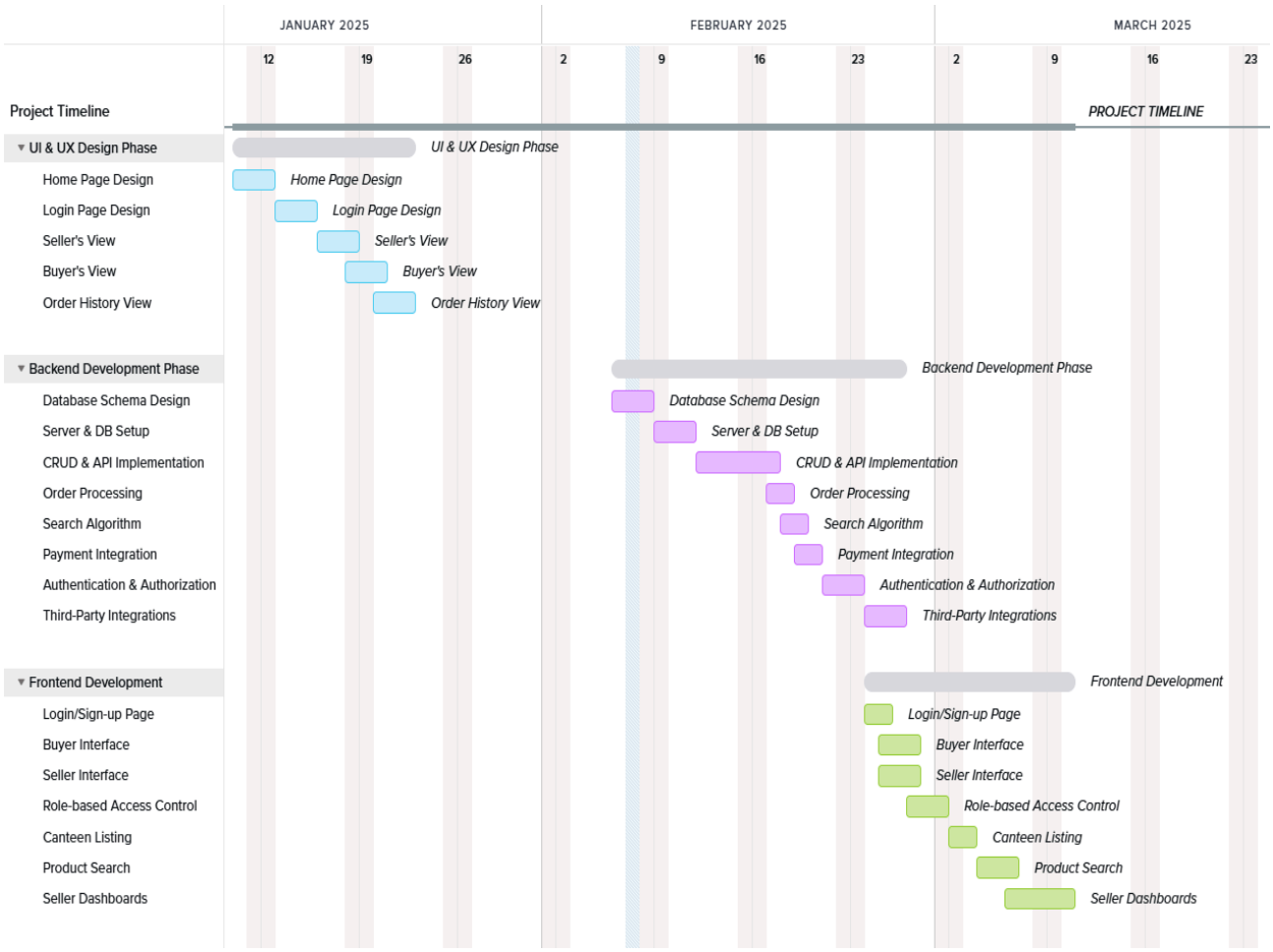
A repository has been set up on GitHub for storing and collaborating on the source code of this project.

- Currently, the repository hosts four major folders (all private access.
    - Campus-Craves-Backend
    - Campus-Craves-Frontend
    - Campus-Craves-Infrastructure
    - Campus-Craves-Database
- We aim to keep high code coverage of the codebase via unit tests.

We have set up a TeamGantt for the project planning and developing a Gantt Chart.

● Work has been divided into Epics, which is further divided into Stories.
● We will have weekly sprints where each team member will have clearly defined work.
● Every team member has internally taken the responsibility to focus majorly on either frontend or the backend.
● Our current Gantt chart looks like this:


CS253-Course-Project
Private
Unwatch 1
Fork 0
Star 0
main
1 Branch
0 Tags
Go to file
Add file
Code
Stochastic-Boy Update README.md
0afd1d2 · now
9 Commits
Campus-Craves-Backend Create Readme_backend.md 3 minutes ago
Campus-Craves-Database Create Readme_database.md 1 minute ago
Campus-Craves-Frontend Create Readme_fronend.md 4 minutes ago
Campus-Craves-Infrastructure Create README_infra.md 2 minutes ago
Project Documents Delete Project Documents/SRS Document 5 minutes ago
README.md Update README.md now
README
CS253-Course-Project (CAMPUS CRAVES)
The "Campus Craves" application allows the digitization of grocery stores and canteens on the IIT Kanpur campus. Canteen and store owners can use the app to advertise their goods, services and menu options online. The online application allows students and campus residents to conveniently make orders online as long as they can access a web browser. In addition, it helps store and canteen owners take and manage orders, keep tabs on stocks, and gain access to some sales information.
About
No description, website, or topics provided.
Readme
Activity
0 stars
1 watching
0 forks
Releases
No releases published
Create a new release
Packages
No packages published
Publish your first package


JANUARY 2025
FEBRUARY 2025
MARCH 2025
12
19
26
2
9
16
23
2
9
16
23
Project Timeline
PROJECT TIMELINE
UI & UX Design Phase
Home Page Design
Login Page Design
Seller's View
Buyer's View
Order History View
Backend Development Phase
Database Schema Design
Server & DB Setup
CRUD & API Implementation
Order Processing
Search Algorithm
Payment Integration
Authentication & Authorization
Third-Party Integrations
Frontend Development
Login/Sign-up Page
Buyer Interface
Seller Interface
Role-based Access Control
Canteen Listing
Product Search
Seller Dashboards

5 Other Details

Appendix A - Group Log

Since the beginning of the project, our entire team has been very enthusiastic. We have formed a Whatsapp group for effective communication. We are communicating via calls, messages and offline meets. We have created a private repository on GitHub for collaborative software development.

Meeting Time	**Agenda/Discussions**
28 January 2025 (21:00 - 22:00 IST)	Started working on a draft of the Software Design document.
2 February 2025 (21:00 - 22:00 IST)	Discussed the object oriented structure of the project and prepared the rough draft of sequence diagrams and state diagrams.
4 February 2025 (16:30 - 17:00 IST)	Divided the subsections among the members and completed most of them.
6 February 2025 (14:30 - 15:30 IST)	Meet with TA. Updated TA with our current progress. Discussed any issues we were facing. Planned the further steps of development
7 February 2025 (10:00 - 11:00 IST)	Made changes suggested by TA. Completed all the subsections of the document. Discussed the project plan.