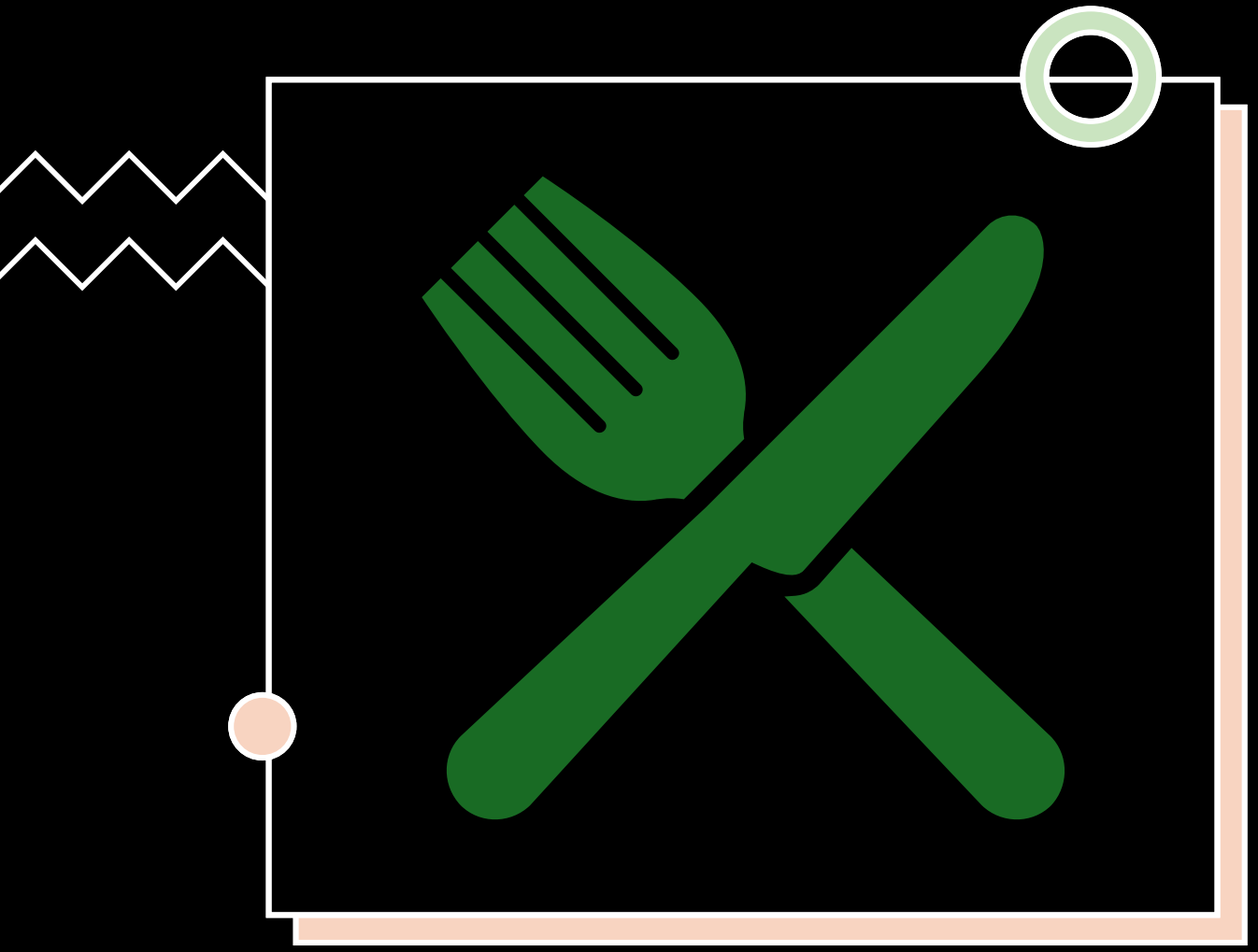


COMP 3500 Software Engineering Food Ordering and Tracking System

Group 14

Project background

Many small restaurants continue to use manual ordering which presents delays, errors and poor customer experience.

The system is designed for a chain restaurant model, requiring support for unified memberships and cross-branch.

Our project will target the enhancement of customer experience when it comes to ordering and efficiency.

Stakeholder

Customer

Register, Browse, Order, Track

Kitchen

Manage tables, Modify orders

Admin

full permission, View & update orders

Manager

Menu & staff management

Staff

System setup, Data import

Functional Requirements

Customer

- Register and login
- Browse menu by categories
- Place orders (dine-in/takeaway)
- Customize menu items
- Track order status in real-time
- View order history

Kitchen

- Manage tables
- Create and modify orders
- Search and view orders
- View kitchen orders

Admin

- View real-time order list
- Update order status
- Filter orders by status

Manager

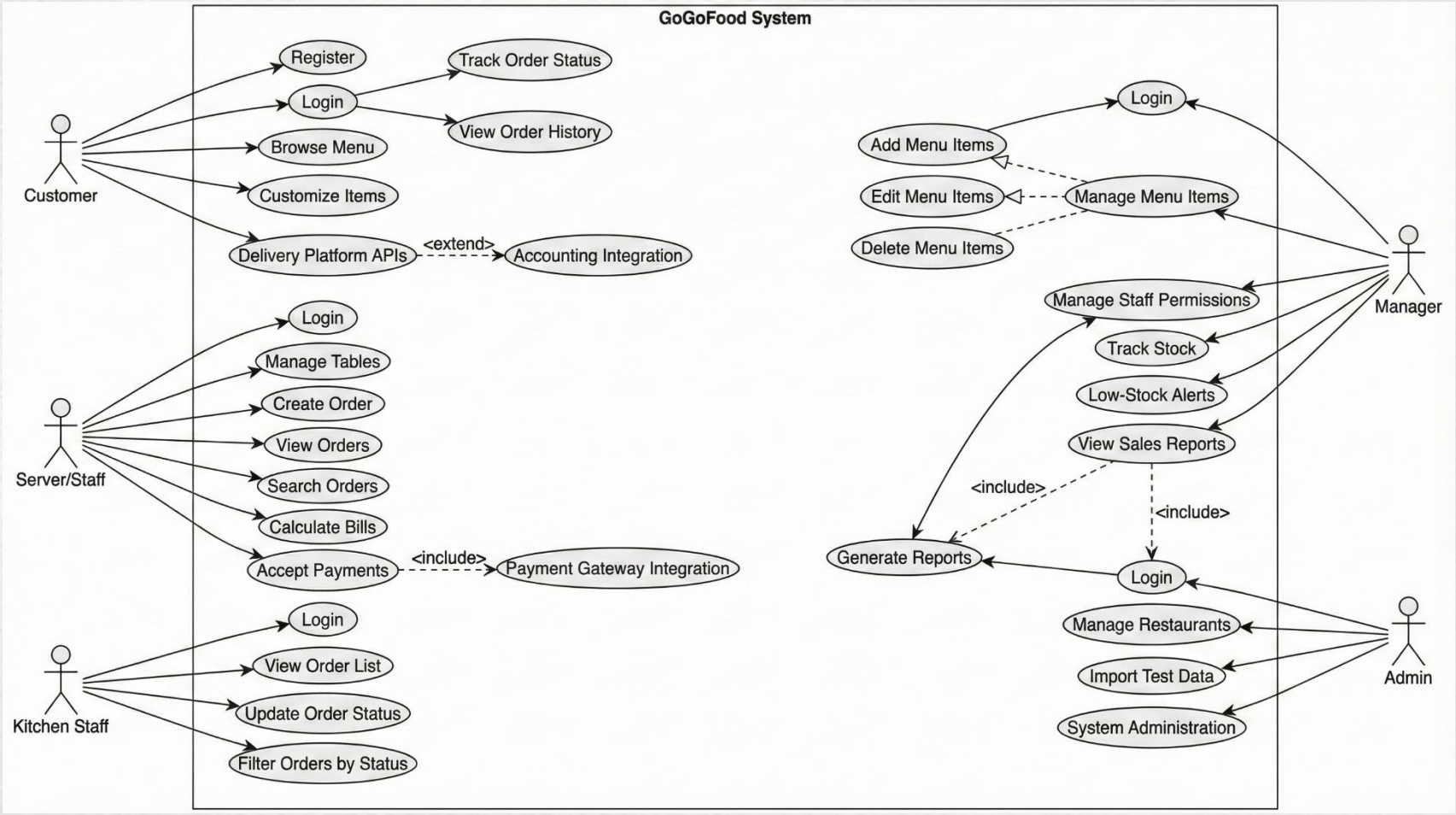
Manage menu items (add/edit/delete)

- Manage staff permissions (RBAC)
- View and generate sales reports

Staff

System administration

- Manage restaurants
- Import test data



NON Functional Requirements

Performance

- Order confirmation within 1.0 seconds (98% of the time)
- Real-time updates within 2 seconds
- Menu browsing < 500ms

Reliability

- Downtime ≤ 10 minutes/month
- 99.9% uptime target
- Data consistency across applications

Security

- Secure authentication
- Encrypted payment data
- Protected customer data

Usability

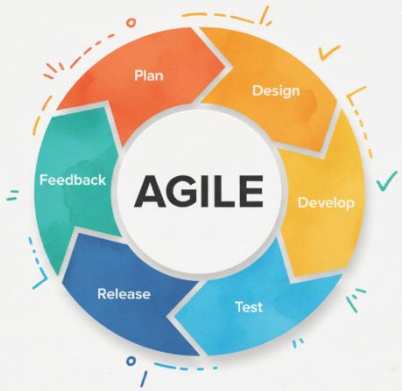
- Intuitive customer interface
- Fast interface
- Clear kitchen display

Scalability

- Handle increasing users
- Maintainable code structure

System implementation

Agile Development Plan



Sprint	Expected Duration	Process	Sprint Goal	Key Activities & User Stories to Implement	Artifacts
Sprint 0	1-2 Days	✓	**Project Setup & Planning**	- Establish project architecture (Client-Server, MVC). - Set up Android Studio and Firebase project. - Refine and prioritize the initial Product Backlog.	Initial Product Backlog
Sprint 1	2 Weeks	✓	**Core Ordering & Kitchen View**	- **UC-1:** Customer can view the menu and place an order. - **UC-7:** Kitchen staff can see a real-time list of new orders.	Sprint 1 Backlog, Potentially Shippable Increment 1
Sprint 2	2 Weeks	✓	**Order Management & Status Tracking**	- **UC-4:** Server can create a table order. - **UC-5:** Server can modify an existing order. - **UC-3:** Customer can track their order status. - **UC-8:** Kitchen staff can update an order's status.	Sprint 2 Backlog, Potentially Shippable Increment 2
Sprint 3	2 Weeks	**UC-9 developing**	**Management Features & Finalization**	- **UC-9:** Manager can add/edit menu items. - **UC-10:** Manager can generate a daily sales report. - Final testing, bug fixing, and UI polish. - Prepare for deployment.	Sprint 3 Backlog, Final Product
Future Sprint	TBD	TBD	**Payment & Billing Integration**	- **UC-2:** Customer can pay for an online order. - **UC-6:** Server can process a bill (including splits/discounts). - Implement secure payment gateway integration.	Future Sprint Backlog

System implementation

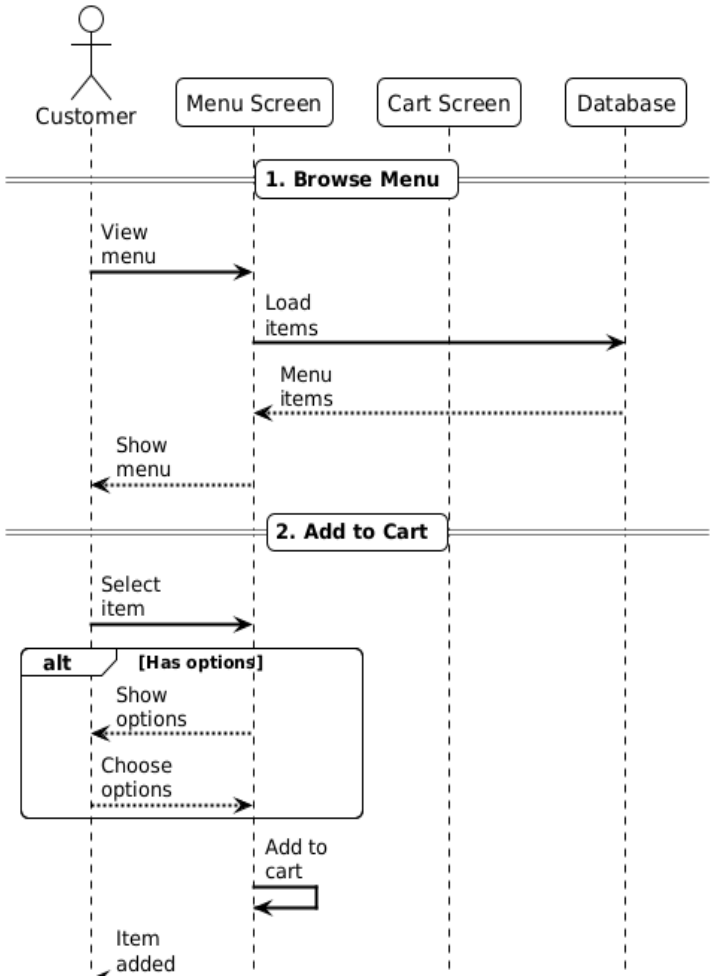
- **FRONTEND**
- **Platform:** Android Native
- **Language:** Java
- **SDK:** Min 24 | Target 36
- **IDE:** Android Studio
- **UI Framework:** Material Design Components
- **Jetpack:** AppCompat, RecyclerView, ConstraintLayout

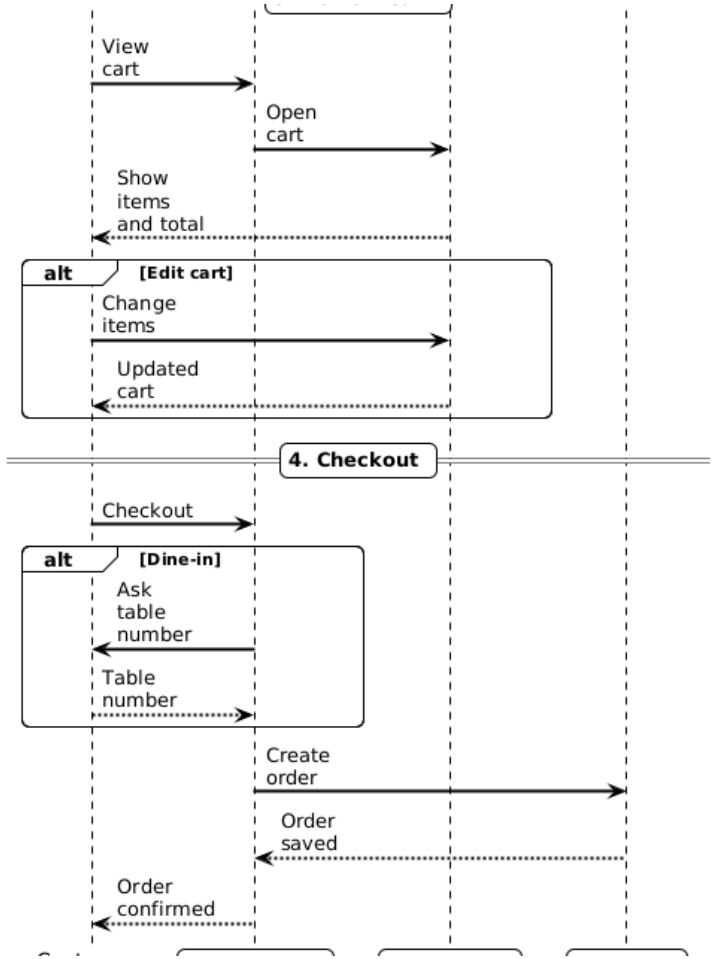
- **BACKEND & DATABASE**

Backend: Firebase (BaaS)

- **Database:** Cloud Firestore (NoSQL)
- **Authentication:** Firebase Auth
- **Real-time:** Firestore Listeners

System modelling





Use ordering function as an example:

Customer →

Browse menu →

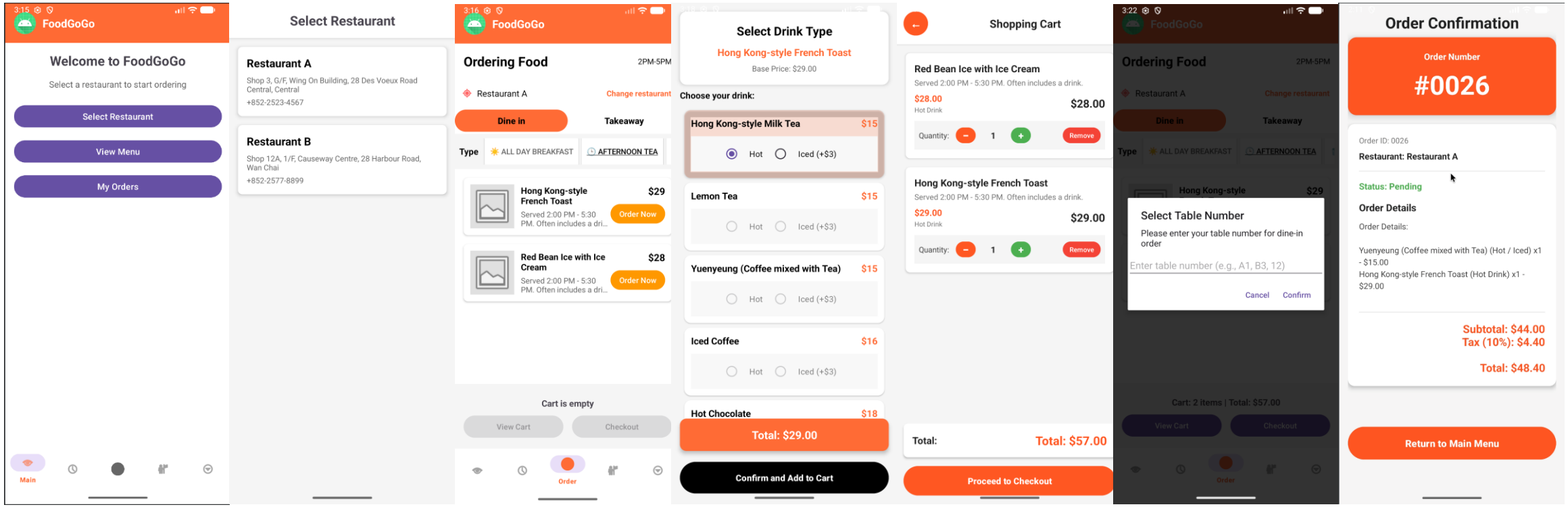
Select items →

Make order →

Payment →

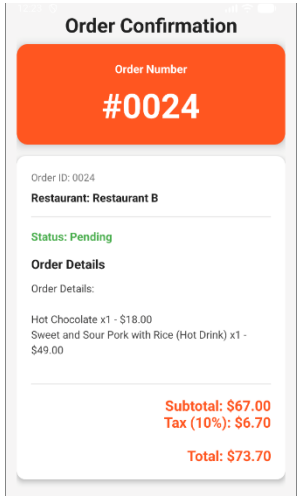
Order Confirmed

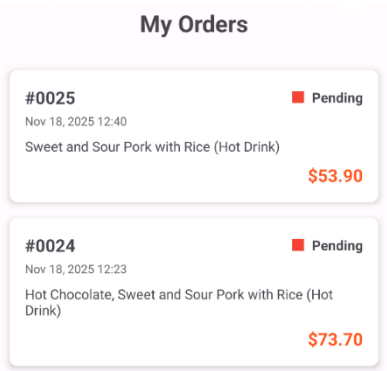
System design



Software Testing/ Validation

- Acceptance Testing: Customer place order successful
- Unit Test: Verified the price calculation
- Performance Testing: Testing the system response time and make sure the system can run every function within 2 seconds





Test Case ID	Description	Expected Result	Status
TC-01	User can select the restaurant and ordering		✅ PASS
TC-02	Verified price calculation	Correct price	✅ PASS
TC-03	Ordering status update		✅ PASS
TC-04	Generate order code & details	Show all order details	✅ PASS
TC-05	Member login function	Member/admin can login successfully	✅ PASS