

Panic Party Pizzeria

SCOPE.....	2
1. Customer records:.....	2
2. Menu:.....	2
3. Payment Processing:.....	2
4. Manager and Employee Permissions.....	2
5. Home Screen:.....	2
Schedule.....	3
Team Organization.....	4
Team Roles.....	5
Technical Description.....	6
Data Management Plan.....	7
Test Plan.....	8

SCOPE

1. Customer records:

- a. Store customer data in a database (postgres/whatever)
 - i. Phone Number, Name, and Card Type
 - ii. Address and the associated information
 - iii. Payment method tracking (Cash vs Card)
 - iv. Receipt Records (Place to sign if card)
- b. New vs Old Customer (Sign up vs Login)

2. Menu:

- a. UI for menu (JS)
- b. Pizza(Size, Crust type, topping list)
- c. Beverages
- d. Functionality
- e. Ability to check off items for order

3. Payment Processing:

- a. Card/Cash
 - i. "Pay at Store" option for carryouts
- b. Order Details
 - i. Record each order a customer makes
- c. Cost
- d. "Thank You" Screen

4. Manager and Employee Permissions

- a. Clock in/out
- b. Item inventory
- c. See Employee information
- d. Able to see previous orders and receipts
- e. Sales
 - i. Able to see information by day, week, and month
 - ii. Total profit
 - iii. Able to perform refunds

5. Home Screen:

- a. Business Address & Phone number
- b. Welcome and Thank you pages

Schedule

Work-Breakdown

Gantt Chart

Team Organization

Team Roles

Project Manager: Ahmadou Bah

Backend: Will Donovan

Frontend: Wainaina Wainaina

QA/Testing: Jaymin McCoy

Technical Description

User Interface Overview

The projected UI will consist of several key screens:

1. Opening Screen: Displays the business name, address, and phone number. Includes navigation buttons for Customer, Employee, and Manager login.
2. Login Screen: Users log in using their phone number. Separate login paths for customers, employees, and managers. New customers can sign up with their name, address, card type, and payment method.
3. Selection/Menu Screen: Customers can browse the menu, including pizza (size, crust type, toppings) and beverages. Items are selectable via checkboxes or toggles.
4. Order Summary & Payment Screen: Displays selected items, total cost, and payment options (Card, Cash, Pay at Store). Includes a signature field for card payments.
5. Thank You Screen: Confirms order and displays estimated pickup/delivery time.
6. Manager Dashboard: Access to inventory, employee info, receipts, and sales data. View sales by day, week, or month. Perform refunds and manage previous orders.

Hardware Requirements

Client Devices: Any modern smartphone, tablet, or desktop with internet access.

Software Requirements

Frontend: React.js

Backend: Java + Springboot

Database: PostgreSQL

Authentication: login using phone number (send code to phone)

Hosting: Cloud-based

Development Restrictions

Max 3 employees and 1 manager account. Login must be phone-number-based. Must support both card and cash payments. Must include a signature field for card payments.

Design & Coding Methods

Agile Development with sprints. MVC Architecture for clean separation of concerns. Responsive Design using CSS frameworks like Tailwind or Bootstrap. Version Control via GitHub.

Data Management Plan

Customer Access and Control

Customers can view and update their personal information (name, address, card type), choose and modify payment methods, view past orders and receipts, and sign up or log in.

Manager Access and Control

Managers can view and manage inventory levels, access employee information, view all customer orders and receipts, analyze sales data by day, week, or month, issue refunds, and manage payment records.

All data is stored securely in a PostgreSQL database with role-based access control to ensure privacy and integrity.

Test Plan

Testing Strategy

Unit Testing: Validate individual components (e.g., login, menu selection). Integration Testing: Ensure smooth interaction between frontend, backend, and database. User Acceptance Testing (UAT): Real users test the app for usability and functionality.

Termination Criteria

Testing will conclude when all critical bugs are resolved, all user stories are validated, and performance benchmarks are met.

User Training

Employees and managers will receive a short training guide. Customers will interact with intuitive UI and tooltips.

Automation

Automated tests using tools. CI/CD pipeline for continuous deployment and testing.

Paging, Data Storage, and Access

Paging: Menu and order history will use pagination for performance. Data Storage: PostgreSQL with encrypted fields for sensitive data. Management Access: Restricted to the manager role with audit logging.