

# Loyalty Points System - Project Work Plan

## Project Overview

The loyalty system rewards customers with points for each transaction. Points can be redeemed for discounts or offers, helping increase customer retention and engagement.

## Core Features

Authentication & Authorization (JWT for backend)  
Customer Management (profile, history, points)  
Points System (earn, redeem, expire)  
Reward Management (custom offers, point thresholds)  
Transaction Logging (track purchases and redemptions)  
Reporting & Analytics (points summary, activity logs)  
Notifications (email/SMS/push - optional)  
React Frontend with responsive UI and admin panel

## Technical Stack

Frontend: React + Tailwind CSS or Chakra UI  
Backend: NestJS + Prisma ORM  
Database: PostgreSQL (via Prisma)  
Auth: JWT, bcrypt (role-based access)  
Deployment: Docker, Vercel/Render/Heroku

## Team Collaboration

The team consists of two members working concurrently:

- Backend Developer (NestJS + Prisma): API, logic, and DB integration
- Frontend Developer (React): UI, state management, API integration

## Timeline (Concurrent Work)

Week 1:

- Backend: Auth setup, DB schema (Prisma), customer module
- Frontend: Project setup, login/register UI, dashboard layout

Week 2:

- Backend: Transaction logic, point calculation, reward system

- Frontend: Customer dashboard, transaction UI, reward views

Week 3:

- Backend: Reporting endpoints, admin logic, notifications
- Frontend: Admin panel, reports, final UI adjustments

Week 4:

- Backend: Testing, final fixes, deployment (Docker/Render)
- Frontend: Integration testing, final UI polish, deployment

## **Project Milestones**

M1: Authentication and customer management APIs

M2: Points system (earn/redeem) with transaction tracking

M3: Fully functional admin and customer dashboards

M4: Reporting system and final deployment