DEPARTMENT OF SOFTWARE ENGINEERING
Faculty of Computing & Information Technology (FCIT)
University of the Punjab, Lahore – PAKISTAN

# FYDP-DSE Proposal

# Smart Shopping Cart

*Bachelors of Science in Software Engineering (2022-2026)*

**By**

Laiba Khalid BSEF22M504
Haiqa Khan BSEF22M507
Ayesha Shahid BSEF22M528

**Supervised by:**

Dr. Madiha Khalid
Assistant Professor

**FACULTY OF COMPUTING AND INFORMATION TECHNOLOGY, UNIVERSITY OF THE PUNJAB, LAHORE.**

DEPARTMENT OF SOFTWARE ENGINEERING
Faculty of Computing & Information Technology (FCIT)
University of the Punjab, Lahore – PAKISTAN

# Proposal Approval Signatures (FAC)

We, the undersigned members of the assigned Faculty Advisory Committee (FAC), have reviewed the submitted project proposal. Based on our initial assessment, we acknowledge the group's submission and confirm that the proposal has been evaluated for completeness and alignment with FYDP objectives.

**Group ID:** FPSEF25X62

**Project title:** Smart Shopping Cart

Primary supervisor: Dr. Madiha Khalid

*(Signature)*

**Signatures (Faculty Advisory Committee (FAC)**

| | **FAC member#1** | **FAC member#2** |
|---|---|---|
| **Name** | Dr. Amina Mustansir | Mrs. Mehwish Kiyani |
| **Signature** | | |

Head of FYDP Coordination Office: ________________

*(Signature)*

# Contents

## 1. Project Title and Abstract

Smart Shopping Cart: A Real-Time Retail Experience Enhancer

### Abstract:

The Smart Shopping Cart is a software-based solution designed to transform the traditional retail shopping experience by integrating real-time purchase tracking with efficient inventory management. The system addresses key retail challenges such as long checkout queues, customer overspending issues, and inefficient stock handling. By equipping shopping carts with RFID/barcode scanning and a live display system, customers can track their purchases, view updated totals, and enjoy faster checkout. For retailers, the system synchronizes inventory automatically, reduces manual staff workload, and provides timely alerts for low-stock items. This innovative solution enhances customer convenience while improving operational efficiency for store management.

## 2. Team Details

| Name | Roll Number | Email | Role<br>(Scrum Role / Module Owner) |
|---|---|---|---|
| Laiba Khalid | bsef22m504 | bsef22m504@pucit.edu.pk | Team Member |
| Haiqa Khan | bsef22m507 | bsef22m507@pucit.edu.pk | Team Leader |
| Ayesha Shahid | bsef22m528 | bsef22m528@pucit.edu.pk | Team Member |

## 3. Problem Statement / Motivation

In conventional retail environments, customers face long queues at billing counters due to manual checkout processes, often leading to frustration and wasted time. Additionally, many customers end up overspending and are forced to remove essential items at the counter, creating inconvenience for both buyers and store staff. Store employees also face the added burden of billing delays, restocking returned products, and manually updating inventory records. For store owners, these inefficiencies contribute to higher labor costs and reduced customer satisfaction.

The motivation behind this project is to create a smarter, technology-driven retail solution that minimizes these challenges. By integrating real-time inventory management, automated billing, and customer-friendly features, the Smart Shopping Cart aims to streamline the shopping process. Solving this problem is important as it improves the overall retail experience, increases customer loyalty, and helps retailers optimize operations in a competitive marketplace.

## 4. Project Objectives

1. **Enhance Customer Experience** – Provide customers with a live catalogue of items, real-time price updates, and automatic total calculation on the screen to make shopping faster, easier, and more convenient.
2. **Reduce Checkout Delays** – Eliminate long queues at billing counters by integrating the billing process into smart carts, thereby reducing average checkout time significantly.
3. **Enable Real-Time Inventory Management** – Synchronize customer purchases directly with the store's database to update stock levels instantly and accurately.
4. **Support Retail Staff and Management** – Minimize manual workload for staff by restocking alerts, and inventory reports.
5. **Ensure Financial Transparency for Customers** – Help customers monitor their spending in real time, reducing chances of overspending or last-minute item removal.
6. **Improve Store Profitability and Efficiency** – Lower labor costs, reduce errors, and optimize inventory restocking with automated notifications to suppliers when stock falls below minimum thresholds.

## 5. System Overview

The Smart Shopping Cart is a software-driven solution that integrates smart shopping carts with real-time inventory management to enhance the retail shopping experience. Each cart will be equipped with a touch-enabled screen connected to a backend system. The cart allows customers to scan products using barcode or RFID technology, instantly updating the cart's interface with product details, prices, and a running total of the bill.

The system will include the following core modules:

1. **Customer Interface –** Displays a live catalogue, total purchase cost, and real-time updates as items are added or removed.
2. **Total Calculation Module** – Automatically updates total items and prices on screen.
3. **Inventory Management Module** – Updates stock levels in real time, synchronizing directly with the store's database.
4. **Notification & Alerts Module** – Sends automated alerts/emails to inventory managers and suppliers when items fall below minimum stock levels.

   When a customer scans an item, the cart system communicates with the backend database to fetch item details and update the totals. The inventory module instantly reduces stock count and triggers alerts if needed. Together, these components solve major retail challenges by reducing queues, minimizing manual staff burden, ensuring accurate inventory tracking, and enhancing customer satisfaction.

## 6. Development Methodology

The development of the Smart Shopping Cart will follow the Agile (Scrum) methodology to ensure flexibility, continuous improvement, and efficient progress. Agile is well-suited to this project due to its iterative nature, adaptability to changes, and ability to deliver working prototypes quickly.

1. **Sprint-Based Iterations**
   - The project will be divided into **7 short sprints**, each lasting **2–3 weeks**.
   - Every sprint will focus on developing specific modules such as inventory management, total calculating system, or customer interface.

2. **Backlog Management**

### 2.1. Customer Interface Module

- **Touch-enabled UI setup** A smart, touch-friendly cart screen that allows customers to easily view, add, or remove items with simple taps.
- **Display product details (name, price, stock availability)** Every scanned item instantly shows its name, price, and available stock, ensuring transparency for customers.
- **Running bill display (items, price)** The cart automatically updates the item count and total bill in real time, helping customers track spending**.**

### 2.2. Total Calculation Module

- **Automatic calculation of total items** The cart auto-counts all added products, giving customers a clear view of their total items without errors.
- **Real-time total updates** The total updates instantly as items are added or removed, ensuring the customer always sees the correct payable amount.
- **Budget limit indicator** The system alerts customers when they approach or exceed a set budget, helping them manage spending effectively.

### 2.3. Inventory Management Module

- **Automatic inventory updates** Inventory adjusts instantly when items are added or removed from the cart, keeping counts accurate.
- **Real-time synchronization with database** Stock updates reflect immediately in the central system, ensuring accurate data for staff and managers.
- **Inventory staff dashboard** Staff get a live dashboard to track low-stock items and manage timely shelf replenishment.
- **Restocking history for inventory manager** Managers can review detailed records of alerts and restocking actions to improve planning and efficiency.

### 2.4. Notification & Alerts Module

- **Minimum stock threshold logic** Each item has a set minimum level; when stock falls below it, the system triggers an automatic alert.
- **Automated supplier notifications** Critical stock shortages generate instant emails or SMS alerts to suppliers for timely replenishment.
- **Inventory manager dashboard alerts** Managers receive highlighted alerts on their dashboard to make quick, informed restocking decisions.

### 2.5. Integration of Scanning System

- **Barcode scanner integration** A built-in barcode scanner quickly identifies products and displays details instantly on the cart screen.

- **RFID tag reader integration** Supports RFID tags, letting customers place items in the cart for automatic detection without manual scanning.
- **Accurate mapping of scanned codes** Each scanned product is verified against the database to ensure correct pricing and stock details.
- **Error handling for invalid scans** Invalid codes trigger on-screen error messages and staff alerts to prevent billing mistakes.

### 2.6. Checkout and Billing Module

- **Manual bill generation** At checkout, the cart provides a summary of items, and staff generate the final bill manually.
- **Cashless checkout (QR code/digital payment)** Customers can pay via wallets, credit cards, or QR scanning, reducing waiting time at counters.
- 
- **Receipt generation** Receipts are provided either as a printed copy or sent via email/SMS after payment.
- **Purchase history tracking (future scope)** Future versions can store purchase history for loyalty programs, offers, and customer insights

### 2.7. Testing & Quality Assurance

- **Unit testing of modules** Each core module (UI, billing, inventory, notifications) will be tested individually to verify functionality and correctness.
- **Integration testing** Modules will be integrated step by step, ensuring smooth communication between the shopping cart, backend database, and scanning systems.
- **Load testing for multiple carts** The system will be tested for simultaneous usage across many carts to ensure performance stability during peak shopping hours.
- **User acceptance testing** Final testing will be conducted with real customers and staff to confirm usability, efficiency, and satisfaction before deployment.

## 3.Scrum Roles:

- **Product Owner – Dr. Madiha Khalid**
  Responsible for defining project requirements, prioritizing features, and ensuring that the final solution meets business objectives.
- **Scrum Master – Haiqa Khan**
  Facilitates the Agile process, removes obstacles, and ensures collaboration.
- **Full Stack Developers – Laiba Khalid & Ayesha Shahid**
  Handle end-to-end development, including coding, system integration, and testing, to deliver a reliable and efficient product.

## 4. Scrum Ceremonies

### 4.1. Sprint Planning

These sprints are organized **priority-wise from start to end**, ensuring a logical flow from core functionality development to final documentation.

**Sprint 1: Basic Interfaces**

- Design basic shopping cart UI layout.
- Add placeholders for items, price, and total.

**Sprint 2: Barcode/RFID Module**

- Integrate barcode and RFID scanning.
- Display scanned product details (name, price).
- Enable add/remove items on cart screen.

**Sprint 3: Total Calculation Module**

- Auto-update total items and price.
- Adjust bill correctly on item removal.

**Sprint 4: Inventory Management**

- Reduce stock count when items are scanned.
- Sync stock across multiple carts in real time.

**Sprint 5: Notification & Alerts Module**

- Detect low-stock items using thresholds
- Send automated alerts to managers/staff.

**Sprint 6: Final Integration & Testing**

- Combine all modules into one system.
- Test with multiple products and carts.

**Sprint 7: Documentation**

- Prepare user manuals for customers.
- Write technical docs for developers/admins

### 4.2. Daily Standups

Conducted two or more times per week to discuss progress, highlight challenges, and ensure alignment within the team.

### 4.3. Sprint Review

Held once a week with the supervisor to demonstrate completed features, gather feedback, and ensure the project is on track.

### 4.4 Sprint Retrospective

A reflective session at the end of each sprint to analyze achievements, identify challenges, and implement process improvements for future sprints.

This methodology ensures that the system evolves in collaboration with users and stakeholders, allowing adjustments in requirements without delaying overall project progress. By the end of multiple sprints, a fully functional Smart Shopping Cart prototype will be delivered, tested, and ready for deployment.

## 7. Technology Stack

| **Layer** | **Technologies/Tools** |
|---|---|
| Frontend | HTML+CSS(Bootstrap),React, Java script |
| Backend | C# (Asp.Net) |
| Database | MySQL |
| Dev Tools | Git, GitHub, Docker, Postman, Visual Studio |

## 8. Project Timeline / Milestones

### Project Timeline for D1 Evaluation & Submission

- 1st – 10th Sept, 2025 → Define product vision, identify epics, and set up Git repository.
- 11th – 20th Sept, 2025 → Break down epics into user stories and prioritize backlog.
- 21st – 28th Sept, 2025 → Draft Definition of Done (DoD) and refine backlog.
- 29th Sept – 5th Oct, 2025 → Prepare D1 document using official template and review with supervisor.
- 6th – 10th Oct, 2025 → Final supervisor evaluation, corrections, and submission (hard + soft copy).

### Project Timeline for D2 Evaluation & Submission

- 10th – 31st Oct, 2025 → Complete core sprint development (main modules working with backlog stories).
- 1st – 15th Nov, 2025 → Verify completed stories against acceptance criteria & Definition of Done (DoD).
- 16th – 30th Nov, 2025 → Prepare working build/demo version and test for smooth execution.
- 1st – 10th Dec, 2025 → Finalize GitHub repository (feature branches, commits, README.md, screenshots/exports).
- 11th – 20th Dec, 2025 → Supervisor evaluation, corrections, and final submission (hard + soft copy).

### Project Timeline for D3 Evaluation & Submission

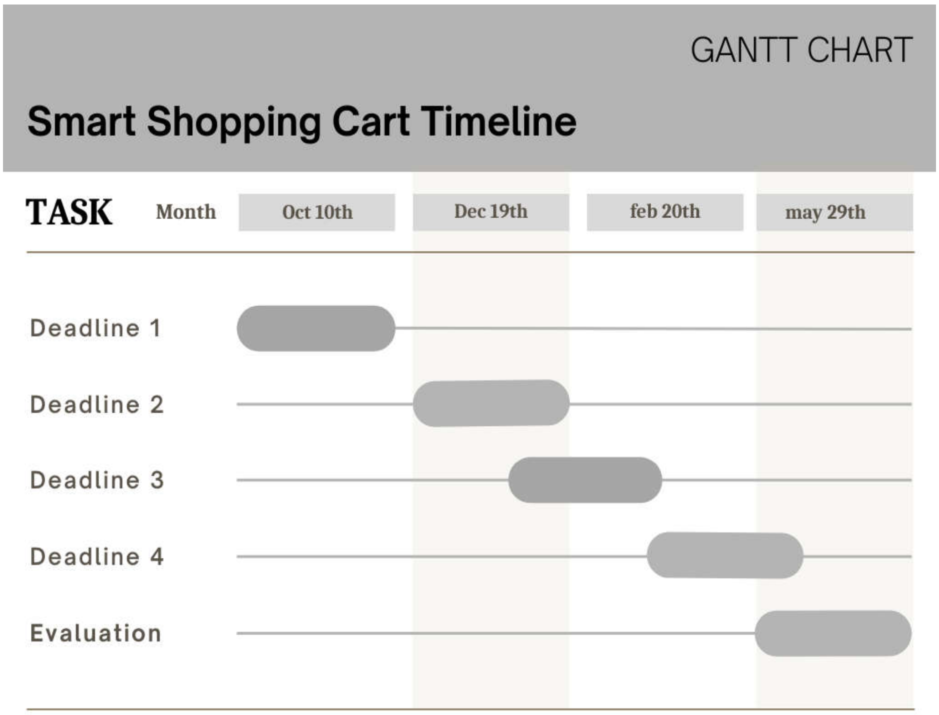
- Dec 20 – Dec 31, 2025 → Resume dev, complete coding & start integration.
- Jan 1 – Jan 15, 2026 → Final testing, verify stories with DoD.
- Jan 16 – Jan 31, 2026 → Deployment scripts & update README.
- Feb 1 – Feb 10, 2026 → Finalize GitHub (branches, docs, packaging).
- Feb 11 – Feb 20, 2026 → Demo, supervisor review & submission.

### Project Timeline (D4 → Final Submission)

- Feb 20 – Mar 10, 2026 → Final deployment setup & testing on fresh environment.
- Mar 11 – Mar 31, 2026 → Code cleanup & optimization.
- Apr 1 – Apr 15, 2026 → Documentation (user guide, setup manual, API docs).
- Apr 16 – May 10, 2026 → Demo preparation (record video, finalize packaging).
- May 11 – May 29, 2026 → Review, polish, and submit.

### Final Evaluation Timeline (Post-D4)

- May 29 – May 31, 2026 → Deployment verification & bug fixes.
- Jun 1 – Jun 2, 2026 → Final project documentation.
- Jun 3, 2026 → Poster preparation.
- Jun 4, 2026 → Presentation & demo rehearsal.
- Jun 5, 2026 → Final Evaluation.



## 9. Market Fit / Commercial Potential

The Smart Shopping Cart is designed to address the growing demand for faster, more convenient, and technology-driven shopping experiences. With the rise of smart retail and digital transformation in supermarkets, malls, and grocery stores, the proposed system has significant market potential.

**Usefulness:**

**For Customers** – Reduces waiting time at counters, provides real-time cost tracking, and makes shopping stress-free.

**For Retailers** – Automates total calculation, reduces staff workload, minimizes inventory errors, and provides sales insights for better decision-making.

**Uniqueness:**

Unlike traditional billing systems, this solution directly integrates smart carts with real-time inventory synchronization, eliminating the gap between customer checkout and store stock records. The system's alert mechanism for low stock, offers features that most existing retail solutions lack.

**Commercial Potential:**

- Can be licensed or sold as software + hardware packages to supermarkets, shopping malls, and retail chains.
- Potential partnerships with POS system providers and retail hardware manufacturers.
- Long-term scalability to online-to-offline (O2O) retail integration, expanding revenue streams.

This combination of customer convenience and retailer efficiency positions the Smart Shopping Cart as both a profitable and sustainable product in the evolving retail technology market.

## 10. Competitor Analysis

Currently, retailers use solutions like traditional POS systems, self-checkout machines, and advanced smart store models such as Amazon Go. However, each has limitations that the Smart Shopping System addresses more effectively.

**Current solutions already present:**

Traditional POS Systems rely on manual checkout. This leads to long queues, delayed billing, and no real-time integration with customer carts.

Self-Checkout Machines reduce dependency on cashiers but are expensive to install and maintain. They are prone to errors or misuse by customers and still require some manual supervision.

Amazon Go (Smart Stores) represent a fully automated shopping experience, but they demand highly advanced infrastructure (AI, sensors, cameras) and are extremely costly, making them impractical for mid-level or smaller retailers.

**Our Smart Shopping Cart is:**

1. Affordable compared to self-checkout and Amazon Go.
2. Fully integrated, as it links the cart system directly with billing and inventory updates.
3. Scalable, making it suitable for both small shops and large supermarkets.
4. Unique in providing real-time stock alerts and sales analytics, which competitors either lack or offer at a very high cost.
5. This gives the Smart Shopping Cart a clear competitive edge by combining affordability, efficiency, and innovation in a way that existing solutions do not.