Chapter 3

Requirements and Analysis

3.1 Problem Definition

The traditional college bookstore mainly depends on physical stores where students must visit to buy textbooks, notes, stationery, and other study materials. This manual approach causes various issues:

Limited Availability

- Students often face stock shortages, and there is no real-time way tocheck availability.

Time Consuming

- Students need to visit the store physically, stand in queues, andsometimes leave without getting the required book.

Lack of Centralization

- No proper record of sales, inventory, or customer details makesmanagement difficult.

Payment Issues

- Cash-only transactions are inconvenient for students who prefer digitalpayments.

Delayed Updates

- Price changes, new arrivals, or discount offers cannot be instantlycommunicated to students.

The proposed College Bookstore E-commerce Platform provides a centralized online solution where students and staff can browse, search, and purchase required materials anytime. It improves accessibility, saves time, provides secure transactions, and ensures real-time updates of inventory and offers.

3.2 Requirement Specification

The proposed College Bookstore E-commerce Platform replaces manual operations with a digital, centralized system accessible via web and mobile. The requirements are classified as Functional and Non-Functional.

Functional Requirements

- **User Management:** Students register/login, update profiles, and view order history.
- **Product Management:** Admin adds/edits/deletes books & stationery; students search, filter, and view availability in real time.
- **Shopping Cart & Checkout:** Add/remove items, apply discounts, and confirm orders before payment.
- **Order & Payment:** Track order status, pay via UPI, cards, net banking, or COD with secure gateway.
- **Inventory Control:** Automatic stock updates; low-stock alerts prevent overselling.
- **Reports & Analytics:** Admin views sales, revenue, and inventory reports for decision-making.

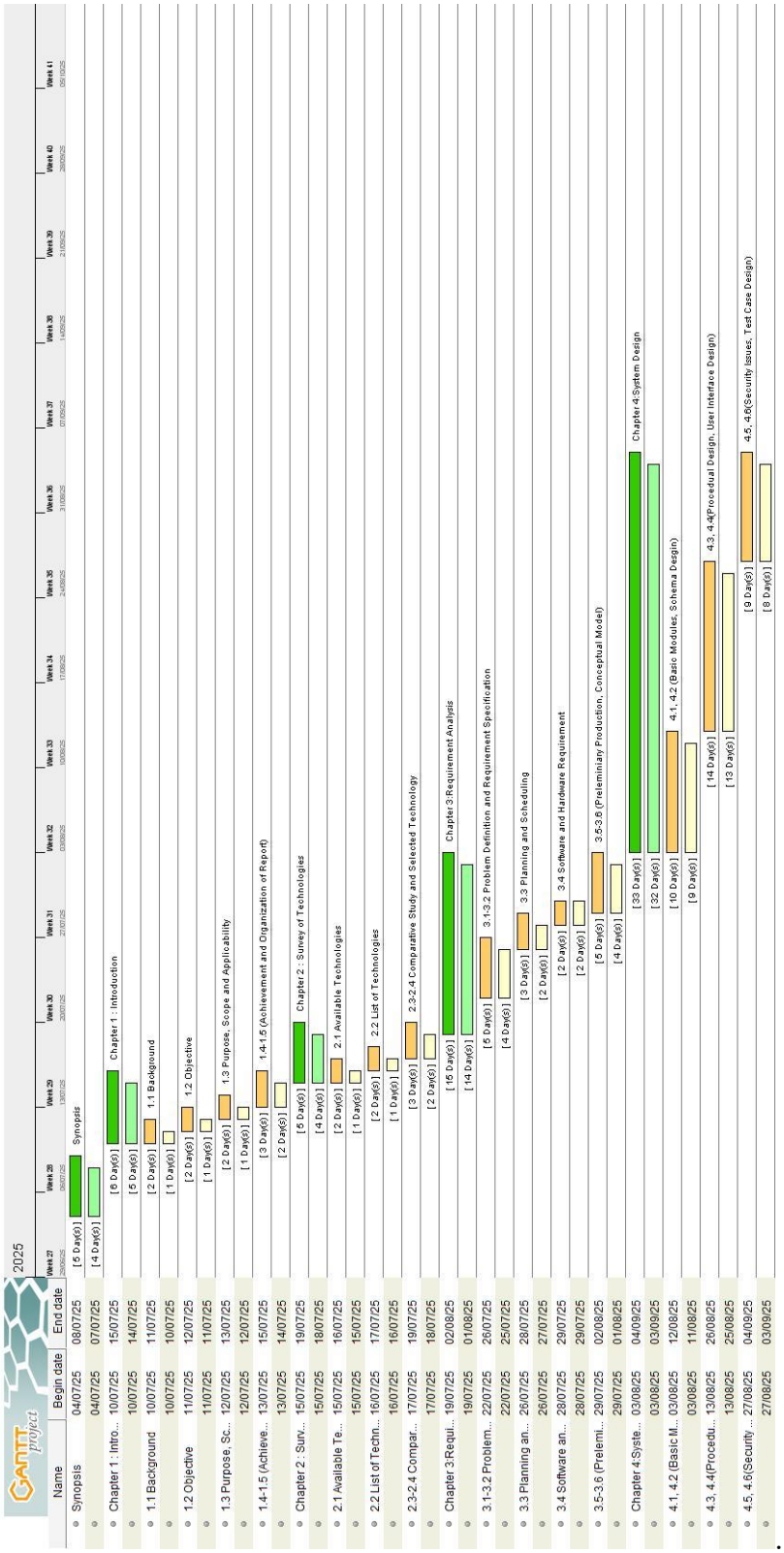
Non-Functional Requirements

- **Performance:** Fast loading (2–3 seconds) and support for multiple users.
- **Security:** Encrypted data, role-based access, and protection from attacks.
- **Usability:** Simple UI, mobile-friendly, and smooth checkout process.
- **Scalability:** Handles growing users, products, and transactions.
- **Reliability:** 99% uptime with backup and recovery options.
- **Maintainability:** Modular code, easy updates, and GitHub version control.

3.3 Planning and Scheduling

3.3.1 Gantt Chart

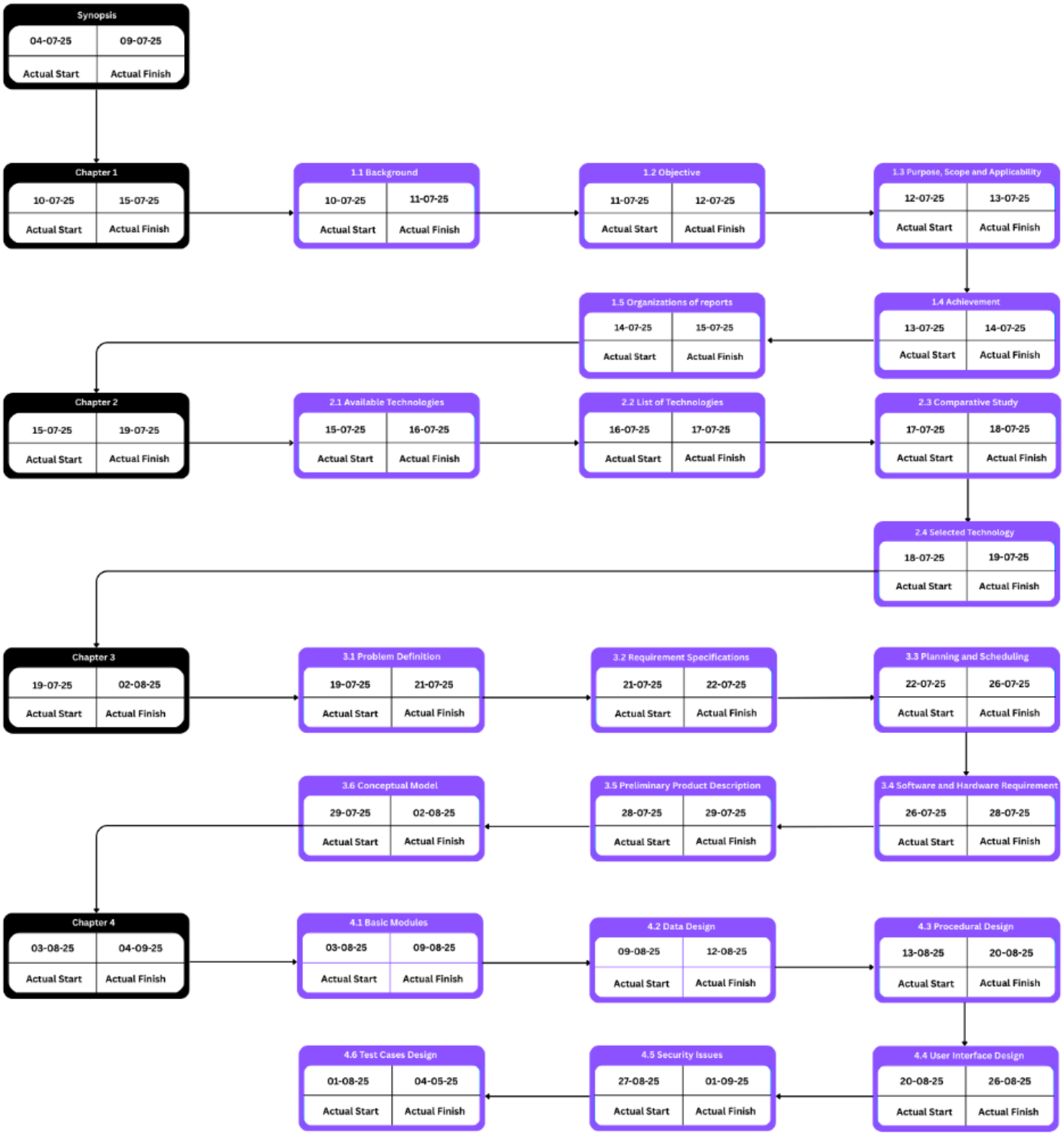
A Gantt chart similar to the attendance system can be prepared showing phases like: Requirement Gathering → Design → Development → Testing → Deployment → Maintenance



.

3.3.2 Pert Chart

A Pert chart will show dependencies between activities such as UI design, backend setup, database integration, payment gateway integration, etc.



3.4 Software and Hardware Requirements

3.4.1 Developer Requirements

Hardware

- Laptop/PC with Intel i3 or above (i5 recommended)
- 8 GB RAM preferred
- 500 MB free storage for project files & database

Software

- OS: Windows/Linux
- Code Editor: VS Code
- Frontend: HTML, CSS, JavaScript
- Backend: PHP
- Database: MySQL
- Version Control: GitHub

3.4.2 User Requirements

Hardware

- Smartphone or PC with internet connectivity.

Software

- Any modern web browser (Chrome, Firefox, Edge).
- Email for registration and authentication.

3.5 Preliminary Product Description

The College Bookstore E-commerce Platform is a web and mobile-based system where:

Students

- Register/Login
- Browse/Search books & stationery
- Add to cart & checkout
- Make secure payments

Store Manager

- Add/Edit/Delete books & products
- Manage categories & stock levels
- View and process customer orders
- Generate sales & revenue reports
- Manage discounts/offers

This system ensures real-time inventory management, secure transactions, and a user-friendly shopping experience.

3.6 Conceptual Model

3.6.1 Use Case Diagram

A use case diagram is the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system. It is a visual representation of the functional requirements of a system, showing how users (actors) interact with the system to achieve specific goals. It's a tool used in software development, systems engineering, and business analysis to identify, clarify, and organize system requirements.

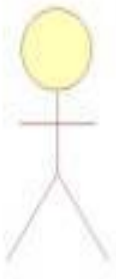
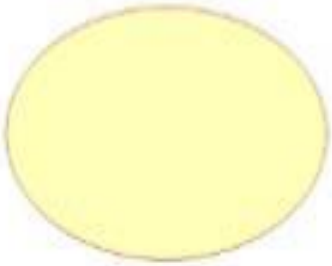
Symbol	Name	Description
	Actor	An actor specifies a role played by a user or any other system that interacts with the subject. It could be a user, another system, or an external hardware device.
	Use Case	Represent the different uses that a user might have. These describe the actions performed by actors to achieve a specific goal.
	Association	Indicates that an actor participates in or interacts with a use case. It shows the interaction or communication between the actor and the system's functionality.

Table 3.1 Use Case Diagram

Actors with respect to project:-

1. User (Student/Faculty)

2. Administrator (Admin)

3. System (Database)

USE CASE :-

1. User registration
2. User login
3. Search book
4. Buy book
5. sell book
6. Upload notes
7. Download notes
8. Manage users
9. Remove inappropriate contant
10. Generate report
11. Maintain defaulter list
12. View announcement/Notification
13. Database management

User (Student/Faculty)

1. User Registration
2. User Login
3. Search Book
4. Buy Book
5. Sell Book
6. Upload Notes
7. Download Notes
8. View Announcement / Notification

2. Administrator (Admin)

1. Manage Users
2. Remove Inappropriate Content
3. Generate Report
4. Maintain Defaulter List

3. System / Database

Database Management (Data store, retrieve, update)

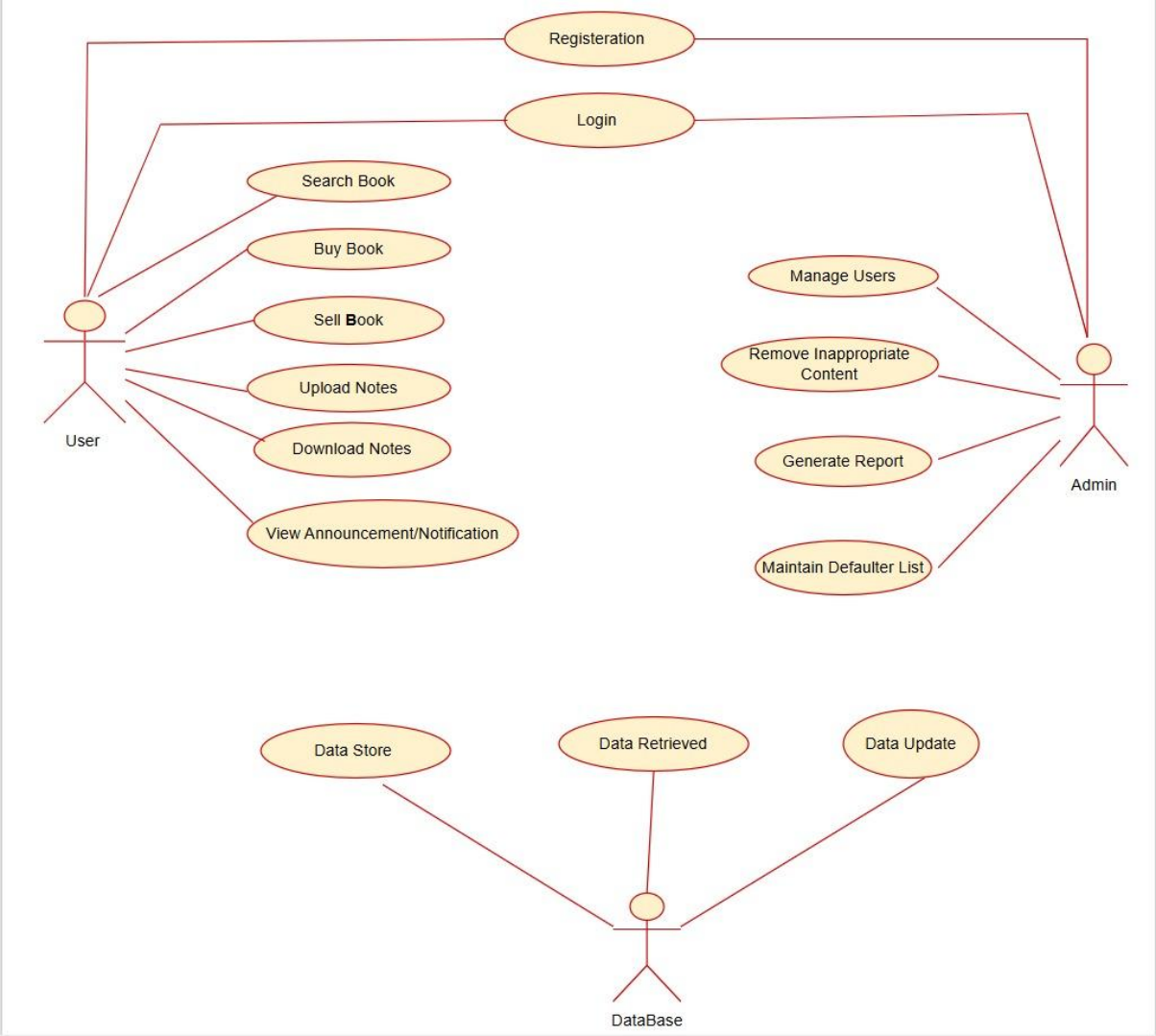


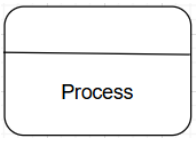
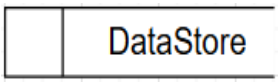
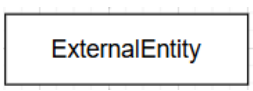
Fig: 3.2 Use Case Diagram

3.6.2 Data Flow Diagram

Definition

A Data Flow Diagram (DFD) is a visual representation of the flow of data through a system, highlighting the inputs, processing, and outputs. A neat and clear DFD can depict a good amount of the system requirements graphically, it makes easy for the business requirements of applications by representing the sequence of process steps by step. It shows how data enters and leaves the system, what changes the information, and where data is stored. The DFD is also called as a data flow graph or bubble chart.

Components

Component Names	Symbol	Description
Process		Processes represent activities or functions that transform input data into output data. They are the actions performed on data within the system.
Data Source		Data Stores represent repositories where data is stored. They can be databases, files, or any other storage location. Data flows in and out of these stores through processes.
External Entity		External Entities are sources or destinations of data that interact with the system but are outside of it. They represent users, other systems, or external data sources.

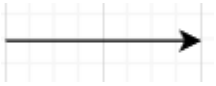
Data Flow		Data Flows depict the movement of data between processes, data stores, and external entities. They indicate how data is transferred, processed, or stored within the system.

Table 3.3 Data Flow Diagram.

Entities Used

1. Student (User)
2. Faculty (User)
3. Administrator (Admin)

Processes

1. User Registration
2. User Login
3. Search Book
4. Buy Baok
5. Sell Book
6. Upload Notes
7. Download Notes
8. Manage Users (Admin)
9. Remove Inappropriate Content (Admin)
10. Generate Reports (Admin)
11. Maintain Defaulter List (Admin)
12. View Announcements/Notifications
13. Database Management

Data source used

1. User Database
2. Book Database
3. Notes Repository
4. Defaulter Records
5. Report Logs