



Fashion Store with **AI Recommendation** **system**

20% Off on Selected Items!



Department of Computer Science

University of Gujrat

Fashion Store with AI Recommendations System



Session: BSCS (Morning)

Project Advisor: Mr. Muhammad Abrar

Members

Ali Iftikhar	Reg# 20011519-054
Muhammad Adil	Reg# 20011519-046
Danyal Yousaf	Reg# 20011519-012

Department of Computer Science

University of Gujrat

STATEMENT OF SUBMISSION

This is certify that _____ Roll No. _____ and _____ Roll No. _____ has successfully completed the final year project named as _____ at the Department of Information Technology, University of Gujrat, to fulfill the requirement of the degree of _____ **in Computer Science.**

Project Supervisor

Project Coordination Office

Faculty of C&IT -UOG

Head of the Department

Acknowledgement

We truly acknowledge the cooperation and help make by _____, Chairman, Department of Computer Science, University of Gujrat. He has been a constant source of guidance throughout the course of this project. We would also like to thank _____ for his help and guidance throughout this project. We are also thankful to our friends and families whose silent support led us to complete our project.

Date:

Abstract

The Fashion Store website with an AI-powered personalized recommendation system is designed to enhance user experience and boost sales by leveraging artificial intelligence. The project provides an extensive historical context, tracking the development of e-commerce and the growing need for customized suggestions. The proposed modules include a user-friendly interface, a good product catalog, and an advanced recommendation engine driven by machine learning algorithms. By analyzing user behavior and preferences, the system personalizes product recommendations, creating a more engaging and relevant shopping experience. Pros of the system include improved customer satisfaction, increased sales, and enhanced user engagement. However, there are some challenges such as data privacy concerns and algorithmic biases that should be carefully addressed. The project employs innovative technologies like machine learning frameworks for recommendation algorithms, secure payment gateways. Future directions involve refining recommendation algorithms, implementing real-time personalization, and incorporating feedback loops for continuous improvement.

Contents



	2
CHAPTER 1:	9
PROJECT FEASIBILITY REPORT	9
1.1. Introduction.....	10
1.2. Project/Product Feasibility Report	10
1.3. Project/Product Scope.....	11
1.4. Project/Product Costing	12
1.5. Task Dependency Table.....	17
1.6. CPM - Critical Path Method	19
1.7. Gantt chart.....	21
1.8. Introduction to Team member and their skill set	21
1.9. Task, Duration and Member Assignment Table	22
1.10. Tools and Technology with reasoning	23
1.11. Vision Document	24
1.12. Risk List.....	25
1.13. Product Features/ Product Decomposition.....	26
CHAPTER 2:	28
2.1 Introduction:.....	29
2.1.1 Systems Specifications	29
2.1.2. Identifying External Entities	31
2.1.3. Context Level Data Flow Diagram:	32
2.1.4. "Shall" Statements:	33
2.1.5. Allocate Requirements:.....	33
2.1.6. Prioritize Requirements:.....	34
2.1.7. Requirements Trace-ability Matrix:	34
2.1.8. Use Case Diagram	36
2.1.9. Use case Description!.....	38
CHAPTER 3:	50
3.1. Introduction:	51
3.2. Domain Model.....	52
3.3. Sequence Diagram	53
3.4. Operation Contracts.....	57
3.5. Collaboration Diagram.....	61

3.2. Entity Relationship Diagram:	65
3.3. Data flow diagram (Functional Model).....	66
3.5. Architectural design	68
3.6. Component Level Design.....	69
CHAPTER 4:	73
4.1. Introduction.....	74
4.2. Site Maps.....	74
4.3. Story boards	75
4.5 Trace-ability Matrix.....	77
CHAPTER 5:	78
5.1 Introduction:.....	79
5.2. Test plan.....	79
5.4. Test Case Specification	95
5.5. Test procedure specification	98
5.6. Test item transmittal report.....	99
5.7. Test log.....	99
5.8. Test incident report	100
5.9. Test summary report.....	100

Chapter 1:

Project Feasibility Report

1.1. Introduction

1.2. Project/Product Feasibility Report

1.2.1. Technical Feasibility

The fashion store can work efficiently for every person by having a personalized recommendation system. The project utilizes technologies like HTML, CSS, JavaScript, Bootstrap, Python, MySQL, and AI for recommendation systems. The technical expertise required for these technologies is assumed to be available, indicating technical feasibility.

1.2.2. Operational Feasibility

The work division among group members ensures that there is consideration for the operational aspects, and it implies that the team can operate the project, ensuring operational feasibility.

1.2.3. Economic Feasibility

The cost estimation, including Total Effort, Cost per unit of effort, and Equipment Cost, indicates an economic analysis. However, the Total Cost per unit of effort is mentioned as 0 Rs/-, the only cost which is required is just for availing the resources for the development of project, which might need clarification for economic feasibility assessment.

1.2.4. Schedule Feasibility

The Gantt Chart and work division among group members provide a detailed timeline for the project, suggesting consideration for completing the project within a specific time frame, indicating schedule feasibility.

1.2.5. Specification Feasibility

The project plan includes phases like Requirement Analysis and Planning, User Interface Design, and Database Design, indicating a structured approach to ensure clear and definite specifications.

1.2.6. Information Feasibility

Information feasibility is implicit in phases like Requirement Analysis and Planning, where gathering functional requirements is emphasized. The project plan indicates consideration for the completeness and reliability of information.

1.2.7. Motivational Feasibility

However, the work breakdown structure and planned tasks suggest a structured approach that can contribute to maintaining motivation among team members. The motivational feasibility of the project lies in its potential to enhance the online shopping experience, drive sales, and attract a broader user base through personalized recommendations. The project aims to create a user-centric

platform that aligns with evolving consumer expectations, fostering engagement and satisfaction. By providing a seamless and tailored shopping journey, the website seeks to motivate users to explore and make purchases, contributing to increased customer loyalty and business success.

1.2.8. Legal & Ethical Feasibility

The legal and ethical feasibility of the proposed e-commerce website with an AI-powered recommendation system involves ensuring compliance with data protection regulations such as GDPR, transparent communication of data usage policies, and the prevention of bias in recommendations. Robust security measures should be implemented for the protection of user data, and users should have clear control over their preferences and consent options. Comprehensive terms of service and user agreements, as well as ethical AI practices, should be established to guide platform interactions. Handling user feedback and promptly addressing concerns, along with ongoing legal compliance efforts, will contribute to a platform that prioritizes user privacy, security, and ethical considerations. Regular updates and adaptations to evolving legal standards are crucial for maintaining a sound legal and ethical foundation.

1.3. Project/Product Scope

The scope of the project encompasses the development of an E-commerce website for a Fashion Store with an AI-powered personalized recommendation system. The boundaries of the system include all functionalities and features necessary to deliver a seamless and personalized shopping experience to users while ensuring efficient utilization of available resources (time, people, and money).

Key Components within the Scope:

1. User Registration and Authentication:
 - Implement secure user registration and authentication mechanisms to ensure user privacy and data security.
2. User Profile Management:
 - Develop functionality for users to create and manage their profiles, including preferences and purchase history.
3. Product Catalog Display:
 - Design and implement a user-friendly interface for browsing and exploring the extensive product catalog.
4. Search and Filters Functionality:
 - Provide advanced search and filtering options to help users find products based on various criteria such as category, price range, and brand.
5. Shopping Cart Management:
 - Enable users to add products to their shopping cart, update quantities, and proceed to checkout seamlessly.
6. Checkout Process:
 - Develop a smooth and intuitive checkout process with multiple payment options and order summary display.
7. AI-Powered Recommendation Engine Integration:
 - Integrate an AI-powered recommendation system to analyse user behaviour and preferences, offering personalized product suggestions.
8. User Feedback and Ratings:

- Allow users to provide feedback and ratings on products, enhancing the overall shopping experience and facilitating decision-making for other users.
9. Order Management:
 - Implement functionality for users to track their orders, view order history, and manage returns or exchanges.
 10. Responsive Design Implementation:
 - Ensure the website is responsive across various devices and screen sizes, providing a consistent user experience.
 11. Admin Panel for System Management:
 - Develop an admin panel to manage products, orders, users, and other system settings efficiently.
 12. Reporting and Analytics Tools:
 - Integrate reporting and analytics tools to track user behaviour, sales performance, and other key metrics for informed decision-making.
 13. Security Measures Implementation:
 - Implement robust security measures to protect user data, prevent unauthorized access, and ensure compliance with industry standards and regulations.
 14. Notification System:
 - Develop a notification system to keep users informed about order updates, promotions, and other relevant information.

1.4. Project/Product Costing

1.4.1. Project Cost Estimation by Function Point Analysis

In FPA, there are three main components to consider: External Inputs (EI), External Outputs (EO), and External Inquiries (EQ). Each of these components is assigned a weight based on complexity, and the total weighted value is used for further calculations.

Let's analyse the functional points for our project:

External Inputs (EI):

1. User Registration and Authentication
2. User Profile Management
3. Product Catalog Display
4. Search and Filters Functionality
5. Shopping Cart Management
6. Checkout Process
7. AI-Powered Recommendation Engine Integration
8. User Feedback and Ratings
9. Order Management
10. Responsive Design Implementation
11. Admin Panel for System Management
12. Reporting and Analytics Tools

13. Security Measures Implementation

14. Notification System

External Outputs (EO):

1. AI-Powered Recommendation Engine Output
2. User Feedback and Ratings Output
3. Order Management Output
4. Notification System Output

External Inquiries (EQ):

1. User Inquiries about Product Information
2. User Inquiries about Order Status

Internal Logic Files (ILF):

1. Product Information
2. Transaction information
3. User's Information

External Interfaces (EIF):

no

Complexity Weights:

Measurement Parameter	Low	Average	High
1. Number of external inputs (EI)	7	10	15
2. Number of external outputs (EO)	5	7	10
3. Number of external inquiries (EQ)	3	4	6
4. Number of internal files (ILF)	4	5	7
5. Number of external interfaces (EIF)	3	4	6

Weighted Values:

External Inputs (EI):

- 14 EI * Average Complexity (10) = 140

External Outputs (EO):

- 4 EO * Average Complexity (7) = 28

External Inquiries (EQ):

- 2 EQ * Average Complexity (4) = 8

Internal Logic Files (ILF):

3 ILF * Average Complexity (5) = 15

External Logic Files (ELF):

- 0 ELF * Average Complexity (4) = 0

Total Unadjusted Function Points (UFP):

- $UFP = EI + EO + EQ + ILF + EIF$
- $UFP = 140 + 28 + 8 + 15 = 191$

Complexity Adjustment Factor (CAF):

- CAF is based on 14 general system characteristics, each rated on a scale from 0.65 to 1.35.
- $CAF = [0.65 + 0.01 * \sum(f_i)]$ ---- $\sum(f_i) = 14 * 3 = 42$
- $CAF = [0.65 + 0.01 * 42]$
- **CAF = 1.07**

Adjusted Function Points (AFP):

- $AFP = UFP * CAF$
- **AFP = 191 * 1.07 = 204.37**

Total Effort = Functional Points * Hours/functional points

Total Effort = 204 * 8

Total Effort = 1632

Cost per unit of effort = 300 Rs/- per unit of effort

Total Cost per unit of effort = 1632 * 0

Total Cost per unit of effort = 0 Rs/-

1.4.2. Project Cost Estimation by using COCOMO Model

KLOC of all modules is required to find out the estimated cost using COCOMO model.

Front-end Development (2,000 LOC)

- User Interface Design (1,000 LOC)
 - Homepage Design (200 LOC)
 - Product Pages (300 LOC)
 - User Profile Interface (200 LOC)
 - View Order (100 LOC)
 - Payment Page (100 LOC)
 - Search Option (100 LOC)
 - Mega Menu (50 LOC)

- Admin Interface (1,000 LOC)
 - Admin Home Page (200 LOC)
 - Products Management Page (300 LOC)
 - Orders Management Page (300 LOC)
 - User's Management Page (200 LOC)

Back-end Development (5,900 LOC)

- User Authentication (800 LOC)
- Product Management System (800 LOC)
- Checkout Process (800 LOC)
- Order Management System (1,000 LOC)
- User's Management System (1,000 LOC)
- Recommendation Engine Integration (1,500 LOC)
 - Algorithm Development (750 LOC)
 - Integration with Product Catalog (750 LOC)

Total Lines of code = 7900LOC \approx 8KLOC

let's use the Basic COCOMO model to estimate the cost of the project with a size of 10,000 lines of code (KLOC). We'll use the provided COCOMO parameters for an organic project:

Given data:

- Size of the project (LOC) = 8 KLOC
- Effort multiplier (a) = 2.4
- Exponent for size (b) = 1.05
- Effort duration multiplier (c) = 2.5
- Exponent for effort (d) = 0.38
- Average monthly salary per person = 30,000

Step 1: Size Estimation

LOC = 8 KLOC

Step 2: Effort Estimation

$$\text{Effort (PM)} = 2.4 \times (LOC)^{1.05}$$

$$\text{Effort (PM)} = 2.4 \times (8)^{1.05}$$

$$\text{Effort (PM)} \approx 21.30 \text{ Person-Months}$$

Step 3: Duration Estimation

$$\text{Duration (Months)} = 2.5 \times (\text{Effort (PM)})^{0.38}$$

$$\text{Duration (Months)} = 2.5 \times (21.30)^{0.38}$$

$$\text{Duration (Months)} \approx 7.99 \text{ Months}$$

Step 4: Staffing Estimation

$$\text{Number of People} = \text{Effort (PM)} / \text{Duration (Months)}$$

$$\text{Number of People} = 21.30 / 7.99$$

$$\text{Number of People} \approx 2.66 \text{ People}$$

Step 5: Cost Estimation

Assuming an average monthly salary per person, you can calculate the total cost:

$$\text{Cost} = \text{Number of People} \times \text{Average Monthly Salary} \times \text{Duration (Months)}$$

$$\text{Cost} = 2.66 \times 0 \times 7.99$$

$$\text{Cost} = 0 \text{ Rs/-}$$

1.4.3. Activity Based Costing

1. Project Initiation and Planning:

- Activities: Define project scope, develop project plan, team formation, requirement analysis, and planning.
- Costs: Personnel salaries (project manager, team members), project management tools.

2. Design Phase:

- Activities: User interface design, database design, system architecture.
- Costs: Design software/tools, personnel salaries.

3. Development:

- Activities: Front-end development, back-end development, AI recommendation system integration.
- Costs: Software development tools, AI-related libraries, salaries.

4. Testing:

- Activities: Unit testing, integration testing, system testing, user acceptance testing (UAT).
- Costs: Testing tools, personnel salaries.

5. Optimization and Refinement:

- Activities: Performance optimization, security audits, iterative improvements based on feedback.
- Costs: Tools for optimization, security tools, salaries.

6. Deployment:

- Activities: Initial deployment for internal testing, full system deployment.
- Costs: Deployment tools, server costs, salaries.

7. Launch and Marketing:

- Activities: Marketing campaign launch, user communication plan.
- Costs: Marketing materials, advertising costs, salaries.

8. Documentation:

- Activities: Project documentation compilation, final project report.
- Costs: Documentation tools, salaries.

Basic Cost Drivers:

For Heroku basic server, it cost almost 7 dollars per month. And we must get server for 8 months which makes about 56 dollars. In Pakistani Currency, it costs about 16 thousand.

1.5. Task Dependency Table

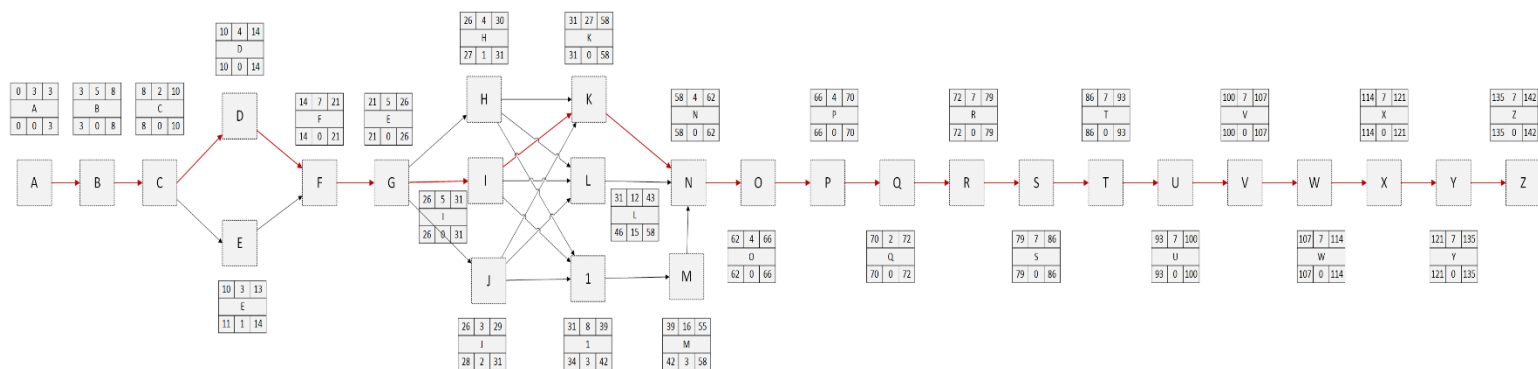
Task	Representation	Predecessor
Project Initiation		
Define Project Scope	A	-
Develop Project Plan	B	A
Team Formation and Roles	C	B
Requirement Analysis and Planning		
Gather Functional Requirements	D	C
Define Non-functional Requirements	E	C
User Stories and Use Cases	F	D, E
Project Management Document	G	F
Design Phase		
User Interface Design	H	G
Database Design	I	G
System Architecture	J	G
Development		
User Module Development	K	H, I, J
Homepage Design		
Product Pages		
User Profile Interface		
View Order		
Payment Page		
Search Option		
Mega Menu		
Admin Module Development	L	H, I, J
Admin Home Page		
Products Management Page		

Orders Management Page		
User's Management System		
Merchant Module Development	1	H, I, J
Merchant page		
Product Management Page		
Order Management Page		
Recommendation Engine Integration	M	1
Algorithm Development		
Integration with Product Catalog		
Testing		
Unit Testing	N	K, L, M
Integration Testing	O	N
System Testing	P	O
User Acceptance Testing (UAT)	Q	P
Optimization and Refinement		
Performance Optimization	R	Q
Security Audits and Enhancements	S	R
Iterative Improvements based on Feedback	T	S
Deployment		
Initial Deployment for Internal Testing	U	T
Full System Deployment	V	U
Launch And Marketing		
Marketing Campaign Launch	W	V
User Communication Plan	X	W
Documentation		
Project Documentation Compilation	Y	X
Final Project Report	Z	Y

1.6. CPM - Critical Path Method

Task	Representation	Duration (Days)	Predecessor
Project Initiation		10	
Define Project Scope	A	3	-
Develop Project Plan	B	5	A
Team Formation and Roles	C	2	B
Requirement Analysis and Planning		13	
Gather Functional Requirements	D	4	C
Define Non-functional Requirements	E	3	C
User Stories and Use Cases	F	7	D, E
Project Management Document	G	5	F
Design Phase		5	
User Interface Design	H	4	G
Database Design	I	5	G
System Architecture	J	3	G
Development		37	
User Module Development	K	27	H, I, J
Homepage Design		5	
Product Pages		5	
User Profile Interface		5	
View Order		5	
Payment Page		5	
Search Option		5	
Mega Menu		5	
Admin Module Development	L	12	H, I, J
Admin Home Page		5	
Products Management Page		7	
Orders Management Page		5	
User's Management System		5	

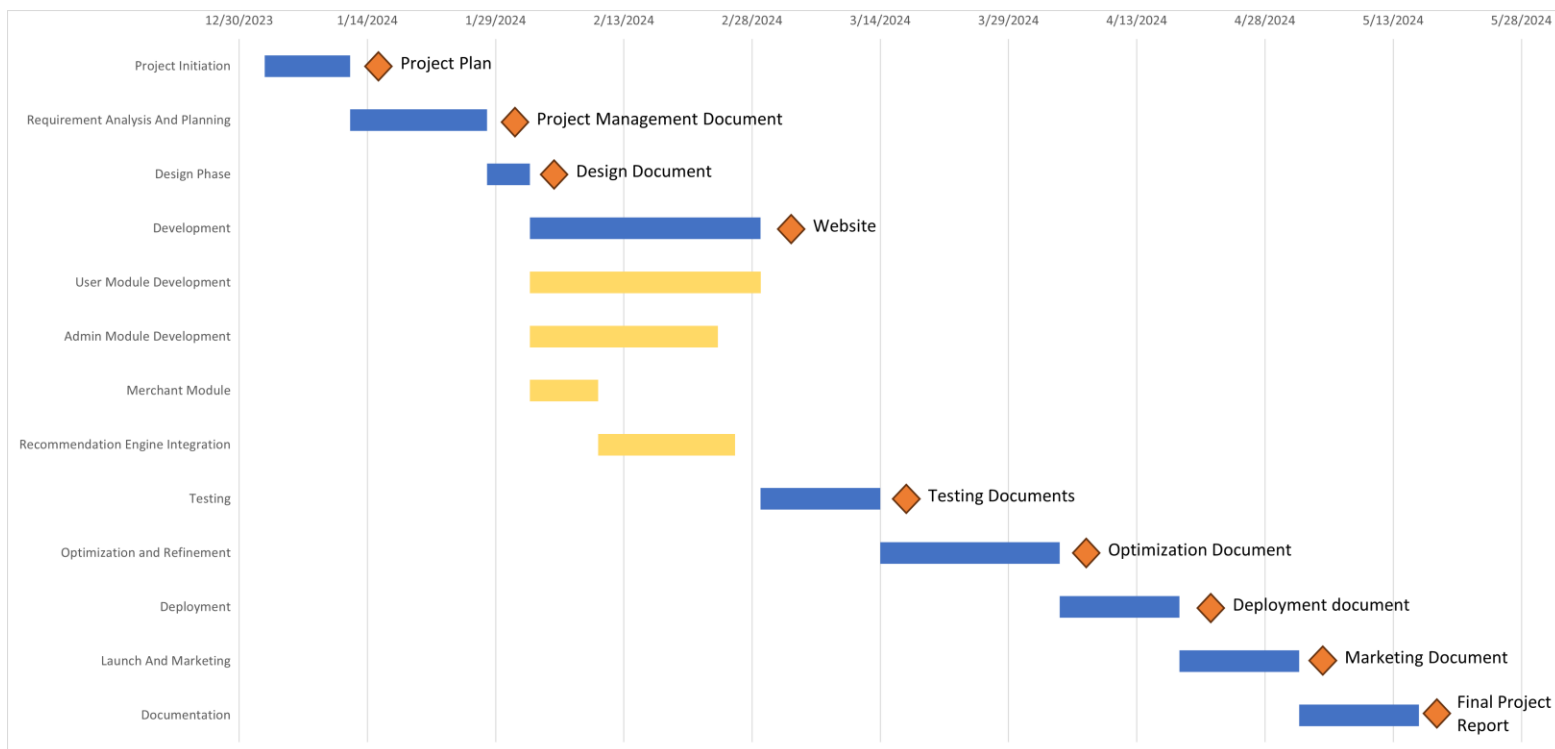
Merchant Module Development	1	8	H, I, J
Merchant page		2	
Product Management Page		3	
Order Management Page		3	
Recommendation Engine Integration	M	16	1
Algorithm Development		8	
Integration with Product Catalog		8	
Testing		14	
Unit Testing	N	4	K, L, M
Integration Testing	O	4	N
System Testing	P	4	O
User Acceptance Testing (UAT)	Q	2	P
Optimization and Refinement		21	
Performance Optimization	R	7	Q
Security Audits and Enhancements	S	7	R
Iterative Improvements based on Feedback	T	7	S
Deployment		14	
Initial Deployment for Internal Testing	U	7	T
Full System Deployment	V	7	U
Launch And Marketing		14	
Marketing Campaign Launch	W	7	V
User Communication Plan	X	7	W
Documentation		14	
Project Documentation Compilation	Y	7	X
Final Project Report	Z	7	Y



Critical Path:

A → B → C → D → F → G → I → K → N → O → P → Q → R → S → T → U → V → W → X → Y → Z

1.7. Gantt chart



1.8. Introduction to Team member and their skill set

Ali Iftikhar:

Have the knowledge and skills of machine learning and coding.

Muhammad Adil:

Have the skills of web development and documentation.

Danyal Yousaf:

Have the skills of web development and documentation.

1.9. Task, Duration and Member Assignment Table

Task	Member Name	Duration (Days)
Project Initiation		10
Define Project Scope	All	3
Develop Project Plan	All	5
Team Formation and Roles	All	2
Requirement Analysis and Planning		13
Gather Functional Requirements	Danyal	4
Define Non-functional Requirements	Adil	3
User Stories and Use Cases	Ali	7
Project Management Document	Adil	5
Design Phase		5
User Interface Design	Adil	4
Database Design	Danyal	5
System Architecture	Ali	3
Development		37
User Module Development		27
Homepage Design	Adil	5
Product Pages	Adil	5
User Profile Interface	Adil	5
View Order	Adil	5
Payment Page	Adil	5
Search Option	Adil	5
Mega Menu	Adil	5
Admin Module Development		22
Admin Home Page	Danyal	5
Products Management Page	Danyal	7
Orders Management Page	Danyal	5
User's Management System	Danyal	5
Merchant Module Development		8
Merchant page	Ali	2
Product Management Page	Ali	3
Order Management Page	Ali	3

Recommendation Engine Integration		16
Algorithm Development	Ali	8
Integration with Product Catalog	Ali	8
Testing		14
Unit Testing	Ali	4
Integration Testing	Adil	4
System Testing	Danyal	4
User Acceptance Testing (UAT)	Danyal	2
Optimization and Refinement		21
Performance Optimization	Adil	7
Security Audits and Enhancements	Danyal	7
Iterative Improvements based on Feedback	Adil	7
Deployment		14
Initial Deployment for Internal Testing	Ali	7
Full System Deployment	Ali	7
Launch And Marketing		14
Marketing Campaign Launch	Adil	7
User Communication Plan	Danyal	7
Documentation		14
Project Documentation Compilation	Adil	7
Final Project Report	Ali	7

1.10. Tools and Technology with reasoning

- **Framework**
 - Django framework will be used to make the project.
- **Front End Development:**
 - For front end development use HTML, CSS, JavaScript, Bootstrap because they are the mostly used website components.
- **For Back End Development:**
 - For backend development we will use python because of its simplicity, versatility, and active community support.
- **For DBMS:**
 - To handle databases, we will use MySQL for efficient data storage and retrieval.
- **For AI Recommendation system:**
 - To implement AI powered recommendation system, we will use python language and deploy it on dataset taken from Kaggle then integrate with our website.

1.11. Vision Document

1. Introduction:

The vision for our project is to develop an advanced e-commerce platform with an AI-powered personalized recommendation system that revolutionizes the online shopping experience. By leveraging cutting-edge technology, we aim to address the evolving needs of modern consumers and enhance user satisfaction while driving business growth for e-commerce retailers.

2. Key Objectives:

- **Enhance User Experience:**
The primary goal is to create a seamless and personalized shopping journey for each user, improving engagement and satisfaction.
- **Boost Sales:**
By providing tailored product recommendations based on user preferences, we aim to increase conversion rates and drive sales for e-commerce businesses.
- **Stay Competitive:**
We aspire to position our platform as an innovative solution that sets our clients apart from competitors, driving market differentiation and growth.

3. Core Features:

- **User Registration and Authentication:**
Seamless onboarding process for users with secure authentication mechanisms.
- **Product Catalog Display:**
Comprehensive display of products with intuitive navigation and filtering options.
- **AI-Powered Recommendation Engine:**
Integration of machine learning algorithms to analyse user behaviour and preferences for personalized product recommendations.
- **User Feedback and Ratings:**
Mechanism for users to provide feedback and ratings, enhancing trust and credibility.
- **Order Management:**
Efficient order processing and management system to streamline transactions.
- **Admin Panel:**
Backend management interface for administrators to manage products, orders, and user data.

4. Stakeholders:

- **Online Shoppers:**
Primary users who benefit from personalized recommendations and enhanced shopping experience.
- **E-commerce Retailers:**
Business owners who seek to improve sales and customer satisfaction through innovative technology solutions.
- **Administrators:**
Individuals responsible for managing and maintaining the e-commerce platform.

5. System Boundaries:

The system encompasses the e-commerce website, including the user interface, backend servers, and database management system. It interfaces with external systems for payment processing and data analytics.

6. Constraints:

- Technical Constraints:
Compatibility with existing hardware and software infrastructure.
- Regulatory Constraints:
Compliance with data privacy regulations and security standards.
- Economic Constraints:
Cost-effectiveness in terms of development and maintenance.
- Time Constraints:
Timely delivery of project milestones and deployment.

7. Key Features and Use Cases:

- User Registration:
Allow users to create accounts and log in securely.
- Product Search and Recommendation:
Provide personalized product recommendations based on user preferences.
- Shopping Cart Management:
Enable users to add and remove items from their shopping carts seamlessly.
- Checkout Process:
Facilitate smooth and secure payment transactions for completed orders.
- Admin Dashboard:
Provide administrators with tools to manage products, orders, and user accounts effectively.

8. Product Requirement Verification:

To ensure the fulfilment of the vision document, checkpoints include:

- Comprehensive exploration of underlying problems and stakeholder needs.
- Accurate formulation of problem statements and system boundaries.
- Complete identification and description of stakeholders and actors.
- Thorough exploration of constraints, including technical, regulatory, and economic aspects.
- Identification and definition of key features consistent with identified problems and constraints.

1.12. Risk List

Data Privacy and Security Breach

- Risk: Unauthorized access to user data or security breaches could lead to loss of customer trust, legal ramifications, and damage to reputation.
- Mitigation: Implement robust security measures, encryption protocols, regular security audits, and compliance with data protection regulations like GDPR.

Algorithmic Bias

- Risk: AI recommendation algorithms may exhibit bias, leading to inaccurate or unfair recommendations, affecting user satisfaction and trust.
- Mitigation: Regularly audit and refine recommendation algorithms, incorporate diverse training datasets, and implement bias detection mechanisms.

Technical Integration Challenges

- Risk: Difficulty integrating AI recommendation system with existing e-commerce platform architecture may lead to delays or technical issues.
- Mitigation: Conduct thorough testing and validation of integration components, involve experienced developers, and allocate sufficient time for integration tasks.

Scalability Issues

- Risk: Inability of the system to handle increased user traffic and data processing demands may lead to performance degradation or system crashes.
- Mitigation: Design the system with scalability in mind, utilize cloud-based infrastructure, implement caching mechanisms, and conduct load testing.

User Adoption and Engagement

- Risk: Users may not adopt or engage with the personalized recommendation features as expected, leading to underutilization and diminished ROI.
- Mitigation: Conduct user research and feedback sessions, optimize user interface and recommendation presentation, and incentivize user engagement through rewards or discounts.

Data Quality and Availability

- Risk: Insufficient or poor-quality data for training AI algorithms may result in inaccurate or irrelevant recommendations.
- Mitigation: Implement data validation and cleansing processes, utilize multiple data sources, and regularly update and refine training datasets.

1.13. Product Features/ Product Decomposition

Product features categorized according to functional requirements:

User Management:

User Registration and Authentication:

- Allow users to create accounts securely.
- Implement authentication mechanisms for secure login.

User Profile Management:

- Enable users to update their profiles with personal information, preferences, and contact details.
- Provide options to manage account settings and preferences.

Product Catalog:

3. Product Catalog Display:

- Display a comprehensive catalog of products categorized by type, brand, price range, and other relevant attributes.
- Include high-quality images, detailed descriptions, and pricing information for each product.

Search and Filters Functionality:

- Implement search functionality allowing users to search for products by keyword, category, or attribute.
- Provide filtering options to refine search results based on user preferences.

Shopping Cart and Checkout:

5. Shopping Cart Management:

- Allow users to add items to their shopping carts, view cart contents, and update quantities.
- Provide options for users to remove items or save them for later.

Checkout Process:

- Facilitate a smooth and intuitive checkout process with multiple payment options.
- Include features for order summary, shipping address input, and order confirmation.

AI-Powered Recommendation Engine:

7. Personalized Product Recommendations:

- Utilize machine learning algorithms to analyse user behaviour and preferences.
- Generate personalized product recommendations based on user history, browsing patterns, and purchase history.

Order Management:

8. Order Tracking and Management:

- Allow users to track the status of their orders in real-time.
- Provide order history and tracking details for past purchases.

Admin Panel:

9. Products Management:

- Provide administrators with tools to add, edit, or remove products from the catalog.
- Enable batch updates and bulk import/export of product data.

Orders Management:

- Allow administrators to view, process, and manage customer orders.
- Provide options for order fulfilment, shipping status updates, and order cancellation/refund processing.

Reporting and Analytics:

Analytics Tools:

- Implement reporting and analytics tools to track user behaviour, sales trends, and website performance metrics. - Generate insights and reports to aid in decision-making and business optimization.

Security and Compliance:

Security Measures Implementation:

- Implement robust security measures to protect user data, including encryption, secure authentication, and access controls. - Ensure compliance with data privacy regulations such as GDPR.

Notification System:

Notification and Communication:

- Implement a notification system to alert users about order status updates, promotions, and special offers. - Provide options for users to opt-in or opt-out of email or SMS notifications.

Responsive Design:

Responsive Design Implementation:

- Ensure the website is optimized for a seamless user experience across different devices and screen sizes. - Implement responsive design principles for improved accessibility and usability.

Chapter 2:

Software Requirement Specification (For Object Oriented Approach)

2.1 Introduction:

2.1.1 Systems Specifications

The following are the clauses that must be included when describing the system specifications.

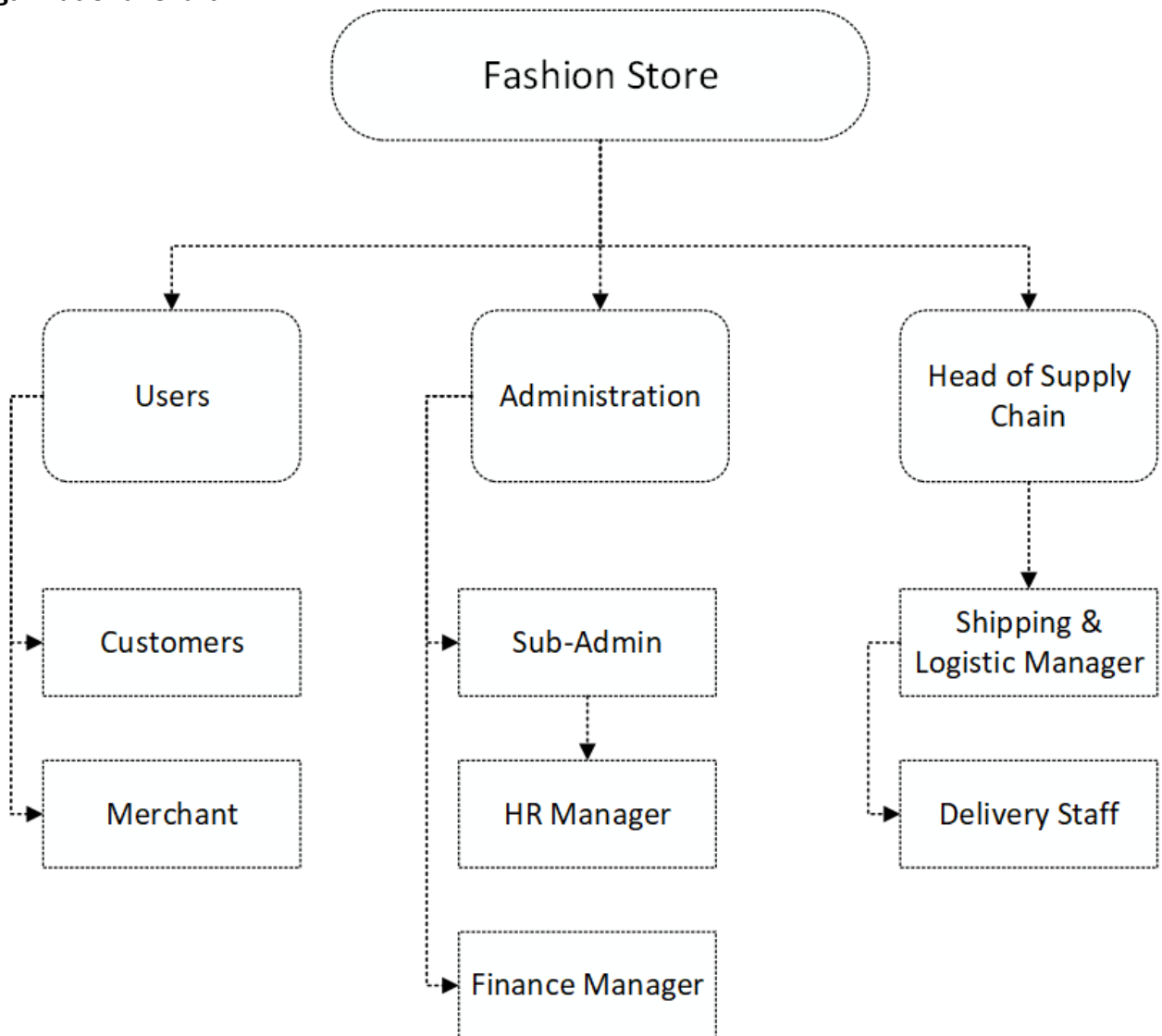
Introduction

The project, led by students at the University of Gujrat, focuses on developing an E-Commerce Website with an AI-Powered Recommendation System. It aims to enhance the online shopping experience through personalized product recommendations, increase user engagement, and boost sales. With a user-friendly interface and innovative technology, the project aims to stay competitive in the E-Commerce market. The use of a 3-tier architecture, advanced AI algorithms, and strategic milestones illustrates the commitment to achieving goals such as improved customer satisfaction and higher conversion rates. The project aligns with modern E-Commerce trends, integrating AI for a unique and tailored shopping journey.

Existing System

The existing system for the proposed e-commerce website lacks an AI-powered personalized recommendation system. It includes standard features such as user registration, product catalog display, shopping cart management, and basic search functionality. However, without AI integration, the system relies on manual user interactions and lacks the ability to offer personalized product recommendations. The absence of a sophisticated recommendation engine limits the platform's potential to enhance the user experience and boost sales through tailored suggestions. The proposed Object-Oriented Approach aims to bridge this gap by introducing a comprehensive system architecture that includes user profiles, a sophisticated recommendation engine, and other essential features to elevate the overall functionality and user satisfaction of the existing system.

Organizational Chart



Scope of the System

The project's scope encompasses the development of an innovative E-Commerce platform tailored for the fashion industry. Focused on enhancing user experience, the system will feature a user-friendly interface, a diverse product catalog, and an advanced AI-powered recommendation engine. The boundaries extend to user registration, profile management, product display, search functionality, shopping cart and checkout processes, AI-driven recommendations, user feedback, and order management. The system aims to personalize the shopping journey, ensuring engagement, increased conversion rates, and improved customer satisfaction.

Summary of Requirements: (Initial Requirements)

The proposed system, an E-Commerce platform with an AI-powered recommendation system, must address key high-level requirements for seamless functionality. These include user registration and authentication, efficient user profile management, a visually appealing product catalog with search and filtering capabilities, streamlined shopping cart and checkout processes, integration of an AI-driven recommendation engine, mechanisms for user feedback and ratings, robust order management, and responsive design implementation. The system is designed to enhance user engagement, increase conversion rates, and achieve a high level of personalization in product recommendations. Security measures, performance optimization, and scalability are integral aspects that the system must adhere to.

2.1.2. Identifying External Entities

External entities identified include Customer, Order, Order Product, Shipment, Invoice, Product, Payment, Account, Credit Card, Cheque, and Request.

a. Over Specify Entities from Abstract:

- User: Individuals engaging with the E-Commerce platform, registering, and participating in online shopping.
- Admin: Authorized personnel responsible for system management, product updates, and order oversight.
- Product: Fashion items featured in the E-Commerce catalog available for users to browse and purchase.
- AI Recommendation System: The intelligent system analyzing user behavior and preferences to provide personalized product suggestions.
- Order: The transactional entity representing user purchases and associated details.
- Feedback and Ratings: Information gathered from users regarding their experiences and opinions on products.
- Security Measures: Tools and protocols implemented to ensure data privacy and protect user information.
- Payment Gateway: External entity facilitating secure online transactions during the checkout process.

b. Perform Refinement:

- Customer: Combining User and Feedback entities, recognizing users as customers providing feedback.
- Administrator: Further specifying Admin, focusing on managerial roles overseeing product and order management.
- Product Catalog: Refining the Product entity, emphasizing the comprehensive catalog available for users.

- AI Engine: A refined version of the AI Recommendation System, emphasizing its role in analyzing and generating recommendations.
- Transaction: A more specific entity representing the purchase transactions made by users.
- Security System: A refined Security Measures entity, focusing on the overall system's security protocols.
- Payment Service Provider: A more specific representation of the Payment Gateway, emphasizing its role in processing online transactions.

2.1.3. Context Level Data Flow Diagram:

Describes a data flow diagram with only one process representing the entire system, including external entities and major data flows. E-Commerce System: The central process representing the entire system.

1. External Entities:

- Customer (User): Engages in activities like registration, browsing products, and making purchases.
- Administrator: Manages the system, oversees product updates, and handles orders.
- AI Engine: Analyzes user behavior and preferences to generate personalized recommendations.
- Product Catalog: Represents the extensive collection of fashion items available for users.
- Transaction: Involves the process of making purchases by the users.
- Feedback and Ratings: Users provide feedback and ratings on products.
- Security System: Ensures the security and privacy of user information.
- Payment Service Provider: Facilitates secure online transactions during the checkout process.

2. Major Data Flows:

- Product Information Flow: Data related to the product catalog is exchanged between the system and the Product Catalog entity.
- User Interaction Flow: Information flows back and forth between the Customer, Administrator, and the E-Commerce System.
- AI Recommendation Flow: Involves the exchange of data between the AI Engine and the E-Commerce System for personalized recommendations.
- Transaction Data Flow: Represents the flow of transaction-related information during the purchase process.
- Feedback and Ratings Flow: Users provide feedback and ratings, influencing the system's understanding of preferences.
- Security Protocols Flow: Data related to security measures flows between the Security System and the E-Commerce System.

- **Payment Transaction Flow:** Involves the secure exchange of data between the Payment Service Provider and the E-Commerce System.

2.1.4. "Shall" Statements:

Requirement	Description
Project Scope	Develop an E-commerce website for a Fashion store.
Integration	Integrate an AI-powered recommendation system.
Objective	Enhance the online shopping experience for users by making it more engaging, efficient, and personalized.
Functionality	Analyse user behaviour and preferences to suggest relevant products.
Goal	Guide users towards products aligning with their preferences to increase sales and improve ROI.
Customer Satisfaction	Provide a seamless and personalized shopping journey to enhance satisfaction.
Business Competitiveness	Help E-commerce businesses stay competitive and relevant by positioning the platform as an innovative solution.
Metrics Improvement	Increase user engagement metrics, such as click-through rates and time spent on the website.
Target Metrics Increase	Aim for a minimum 20% increase in click-through rates and time spent on the website within the first year.
Conversion Rate Improvement	Achieve a 15% increase in conversion rates within the first six months of implementation.
User Profiling	Implement a user profiling system to tailor product recommendations accurately.

2.1.5. Allocate Requirements:

Area	Use Case	Requirement	Member
User Registration and Authentication	User Registration	Gather Functional Requirements	Danyal
	User Authentication	Gather Functional Requirements	Danyal
User Profile Management	View/Edit User Profile	Define Non-functional Requirements	Adil
Product Catalog Display	Browse Product Catalog	User Stories and Use Cases	Ali
	Search for Products	User Stories and Use Cases	Ali
	Filter Products	User Stories and Use Cases	Ali
Shopping Cart Management	Add/Remove Items from Shopping Cart	Gather Functional Requirements	Danyal
Checkout Process	Proceed to Checkout	Gather Functional Requirements	Danyal
AI-Powered Recommendation Engine Integration	Receive AI Recommendations	Gather Functional Requirements	Danyal

User Feedback and Ratings	Provide Product Feedback and Ratings	Define Non-functional Requirements	Adil
Order Management	Place Order	Gather Functional Requirements	Danyal
Responsive Design Implementation	Ensure Website Compatibility Across Devices	User Interface Design	Adil
Admin Panel for System Management	Admin Product Management	User Interface Design	Adil
	Admin Order Management	User Interface Design	Adil
	Admin User Management	User Interface Design	Adil
Reporting and Analytics Tools	Generate Reports	Define Non-functional Requirements	Adil
Security Measures Implementation	Implement Data Security Measures	Define Non-functional Requirements	Adil
Notification System	Send Order Notifications	Gather Functional Requirements	Danyal

2.1.6. Prioritize Requirements:

Priority	Requirement
Highest Priority	User Registration and Authentication
	AI-Powered Recommendation Engine Integration
	Security Measures Implementation
	Checkout Process
	Order Management
Medium Priority	Product Catalog Display
	Search and Filters Functionality
	Shopping Cart Management
	User Profile Management
	Admin Panel for System Management
	User Feedback and Ratings
	Reporting and Analytics Tools
Lowest Priority	Responsive Design Implementation

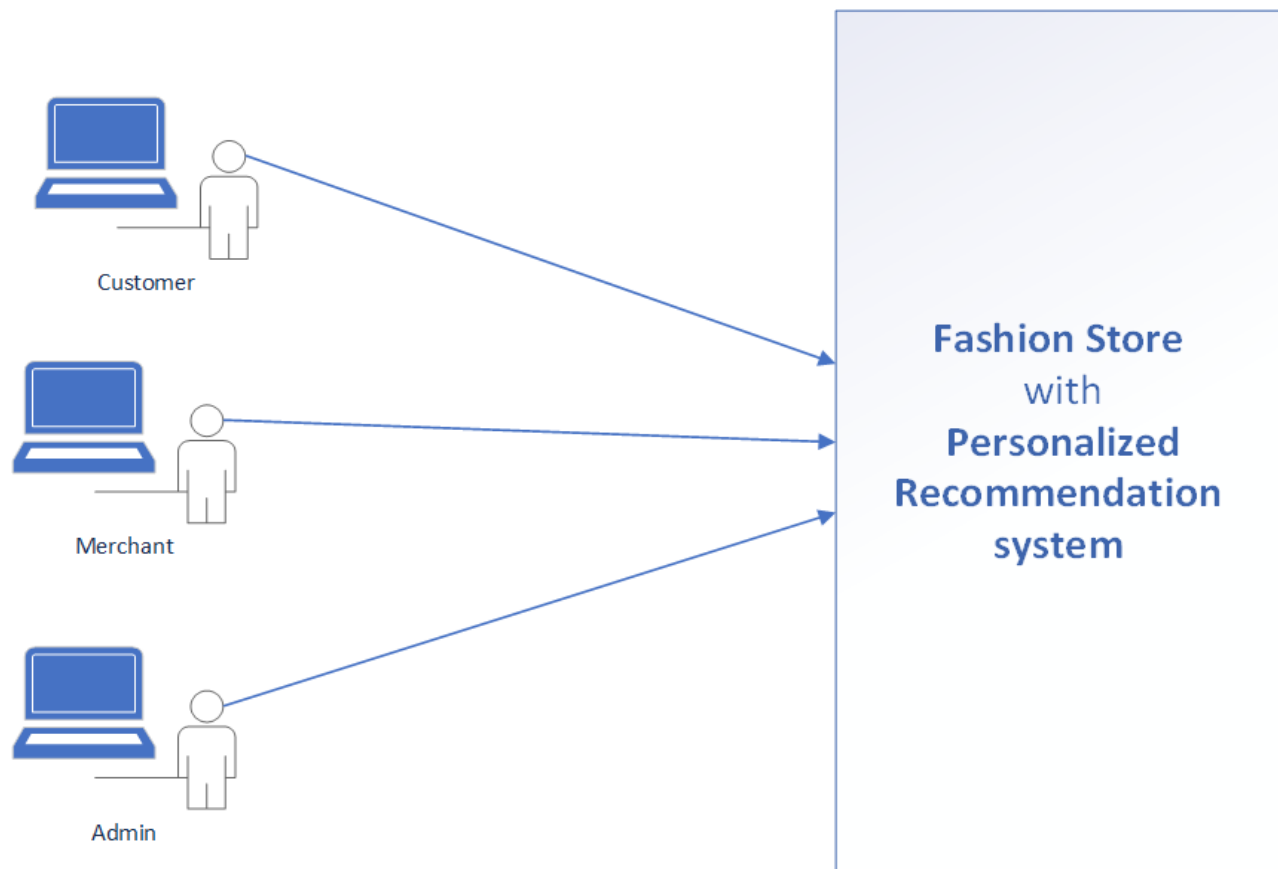
2.1.7. Requirements Trace-ability Matrix:

Requirement ID	Description	Functional Component/Module
REQ-001	User Registration and Authentication	Authentication Module
REQ-002	User Profile Management	User Profile Module

REQ-003	Product Catalog Display	Product Management Module
REQ-004	Search and Filters Functionality	Search Module
REQ-005	Shopping Cart Management	Shopping Cart Module
REQ-006	Checkout Process	Checkout Module
REQ-007	AI-Powered Recommendation Engine Integration	Recommendation Engine Module
REQ-008	User Feedback and Ratings	Feedback and Rating Module
REQ-009	Order Management	Order Management Module
REQ-010	Responsive Design Implementation	Frontend/UI Module
REQ-011	Admin Panel for System Management	Admin Panel Module
REQ-012	Reporting and Analytics Tools	Analytics Module
REQ-013	Security Measures Implementation	Security Module
REQ-014	Notification System	Notification Module

2.1.8. Use Case Diagram

Level 0 Use Case Diagram



Level 1 Use Case Diagram



2.1.9. Use case Description!

Use Case # 1

Use Case Name	Login	
Scope	The System under Design	
Level	User-goal	
Primary Actor	Customers, Admin	
Stakeholders and Interests	They want to interact with system, to perform the actions related to buying and selling cars	
Preconditions	Application must be running.	
Main Success Scenario	ACTOR	SYSTEM
		1. System provides” actor ID” and “password” field to an actor.
	2. After writing down the actor ID and Password, actor clicks on Login button.	3. System validates the actor ID and Password from the Database.
		4. After successfully Login system shows main menu of the application.
Post conditions	Actor successfully logged in.	
Alternatives	If any error occurs, the system asks for check Actor ID and password and again login.	

Use Case # 2

Use Case Name	Log Out
Scope	The System under Design
Level	User-goal
Primary Actor	Customers, Admin
Stakeholders and Interests	They want to interact with system, to perform the actions related to buying and selling cars

Preconditions	Application must be running, and user must be Logged In.	
Main Success Scenario	ACTOR	SYSTEM
	1. Actor Press the logout button.	2. On response System will close all the windows page of the system and securely close the user connection from the database and return to Login page of the Application.
Post conditions	Actor successfully logged Out.	
Alternative Scenario	If entered current password is invalid, then system will generate “Invalid current password” message to an actor.	

Use Case # 3

Use Case Name	User's Profile	
Scope	The System under Design	
Level	User-goal	
Primary Actor	Customers, Admin	
Stakeholders and Interests	They want to interact with system, to perform the actions related to buying and selling cars	
Preconditions	Application must be running, and user must be Logged In.	
Main Success Scenario		
	ACTOR	SYSTEM
	1. Actor Press the Profile button.	2. On response System will show the form which contains all the information related to the user from the database.
	3. Actor will press home button	4. On Response to the home button, system will return to the home page of the application.

Post conditions	The user can see their profile information to make sure about their information is valid or invalid.
------------------------	--

Use Case # 4

Use Case Name	Change Password							
Scope	The System under Design							
Level	User-goal							
Primary Actor	Customers, Admin							
Stakeholders and Interests	They want to interact with system, to perform the actions related to buying and selling cars							
Preconditions	Application must be running, and user must be Logged In.							
Main Success Scenario	<table><tr><td>ACTOR</td><td>SYSTEM</td></tr><tr><td>1. Actor Press the Change Password button.</td><td>2. On response System will show the form which contains Current password, New Password and Confirm New Password fields.</td></tr><tr><td>3. Actor will give them appropriate data to fulfill operation and click on Save Button.</td><td>4. On Response to the Save Button, the system will Check whether current password is true or false from the database. If true, then system will edit the current password to the new password and “successfully password changed” message will display.</td></tr></table>		ACTOR	SYSTEM	1. Actor Press the Change Password button.	2. On response System will show the form which contains Current password, New Password and Confirm New Password fields.	3. Actor will give them appropriate data to fulfill operation and click on Save Button.	4. On Response to the Save Button, the system will Check whether current password is true or false from the database. If true, then system will edit the current password to the new password and “ successfully password changed ” message will display.
ACTOR	SYSTEM							
1. Actor Press the Change Password button.	2. On response System will show the form which contains Current password, New Password and Confirm New Password fields.							
3. Actor will give them appropriate data to fulfill operation and click on Save Button.	4. On Response to the Save Button, the system will Check whether current password is true or false from the database. If true, then system will edit the current password to the new password and “ successfully password changed ” message will display.							
Post conditions	The password is changed in the database.							

Use Case # 5

Use Case Name	Upload Product
Scope	The System under Design
Level	User-goal
Primary Actor	Admin
Stakeholders and Interests	They want to interact with system, to upload the product on the website to put their product for sale.

Preconditions	Application must be running. Actor should be authenticated and validated to use the software.	
Main Success Scenario	ACTOR	SYSTEM
	1. When actor press New Product .	2. System will show the form containing different controls to get required data related to product to add into database.
	3. When actor press the Upload button.	4. System will add the car data in the “Products List” of databases.
	5. Now all actors can view the Product data on main page	
Post conditions	New products were added in the database and can be viewed on the website.	

Use Case # 6

Use Case Name	View Products					
Scope	The System under Design					
Level	User-goal					
Primary Actor	Customer, Admin					
Stakeholders and Interests	They want to interact with system, to perform the actions related to buying and selling cars					
Main Success Scenario	<table><tr><td>ACTOR</td><td>SYSTEM</td></tr><tr><td>1. After pressing the button “Home” button.</td><td>2. System will show all the Products and their details that are saved in the Product List of databases. It will show product’s information like product name, category, size, price etc. for each car.</td></tr></table>		ACTOR	SYSTEM	1. After pressing the button “Home” button.	2. System will show all the Products and their details that are saved in the Product List of databases. It will show product’s information like product name, category, size, price etc. for each car.
ACTOR	SYSTEM					
1. After pressing the button “Home” button.	2. System will show all the Products and their details that are saved in the Product List of databases. It will show product’s information like product name, category, size, price etc. for each car.					
Alternative	If there is no product available, then “No Product Available” message displays in the “Product List” of the actor.					

Preconditions	Application must be running. Network must be connected. Actor should be authenticated and validated to use the software.
Post conditions	The system will display all the registered tasks that user has generated.

Use Case # 7

Use Case Name	Add to cart									
Scope	The System under Design									
Level	User-goal									
Primary Actor	Customers									
Stakeholders and Interests	They want to interact with system, to perform the actions related to buying a product									
Main Scenario	Success	<table><tr><th>ACTOR</th><th>SYSTEM</th></tr><tr><td>1. Select “Product” from car list by click clicking the Buy button provided along all Products.</td><td>2. System will display a confirmation form containing extra details regarding transaction like address, credit id</td></tr><tr><td>3. Actor will input details and click on Submit button</td><td>4. System adds in the Orders list of the actor, which will be shown to user in the cart icon</td></tr><tr><td>5. When Actor click on Cart icon</td><td>6. System will show the list of products bought by the specific actor</td></tr></table>	ACTOR	SYSTEM	1. Select “Product” from car list by click clicking the Buy button provided along all Products.	2. System will display a confirmation form containing extra details regarding transaction like address, credit id	3. Actor will input details and click on Submit button	4. System adds in the Orders list of the actor, which will be shown to user in the cart icon	5. When Actor click on Cart icon	6. System will show the list of products bought by the specific actor
ACTOR	SYSTEM									
1. Select “Product” from car list by click clicking the Buy button provided along all Products.	2. System will display a confirmation form containing extra details regarding transaction like address, credit id									
3. Actor will input details and click on Submit button	4. System adds in the Orders list of the actor, which will be shown to user in the cart icon									
5. When Actor click on Cart icon	6. System will show the list of products bought by the specific actor									
Alternative	If there is no product in the Orders list, display this message, “Nothing ordered”									
Preconditions	Application must be running. Network must be connected. Actor should be authenticated and validated to use the software.									
Post conditions	The system will display all the products bought by specific customer.									

Use case # 8

Use Case Name	Add a customer
----------------------	-----------------------

Scope	The System under Design							
Level	User-goal							
Primary Actor	Customer, Admin							
Stakeholders and Interests	They want to interact with system, to perform the actions related to buying and selling of products							
Preconditions	Application must be running. Actor should be authenticated and validated to use the software.							
Main Success Scenario	<table><tr><th>ACTOR</th><th>SYSTEM</th></tr><tr><td>1. After Selecting “Add User” for admin and “Register” for new customer, Actor provides form to fill.</td><td>2. System prompts form that contain user fields like its Name, Email, Address, userid and password.</td></tr><tr><td>3. After filling in the form and after clicking button “Save”. System act against it.</td><td>4. Response to Actor action a USER creates. New users will be added to the database. User ID will be generated</td></tr></table>		ACTOR	SYSTEM	1. After Selecting “Add User” for admin and “Register” for new customer, Actor provides form to fill.	2. System prompts form that contain user fields like its Name, Email, Address, userid and password.	3. After filling in the form and after clicking button “Save”. System act against it.	4. Response to Actor action a USER creates. New users will be added to the database. User ID will be generated
ACTOR	SYSTEM							
1. After Selecting “Add User” for admin and “Register” for new customer, Actor provides form to fill.	2. System prompts form that contain user fields like its Name, Email, Address, userid and password.							
3. After filling in the form and after clicking button “Save”. System act against it.	4. Response to Actor action a USER creates. New users will be added to the database. User ID will be generated							
Alternative	This user already exists.							
Post conditions	The newly created User added in the database of the system and next time it can be used to login.							

Use Case # 9

Use Case Name	Update User	
Scope	The System under Design	
Level	User-goal	
Primary Actor	Administrators	
Stakeholders and Interests	They want to interact with system, to perform the actions related to buying and selling of products.	
Preconditions	Application must be running. Actor should be authenticated and validated to use the software.	
Main Success Scenario		
	ACTOR	SYSTEM
	1. Admin click on the Users Button	2. System shows a list of all users
	3. After click Update user button	4. System will prompt a form containing fields for getting data related to user.

	5. After filling the field user must press Update button.	6. Response to Actor action an information added in the form will be updated in the database
Alternative	If such username is not available, then system generate error message.	
Post conditions	The information user updates and stores in the database.	

Use Case # 10

Use Case Name	Delete User	
Scope	The System under Design	
Level	User-goal	
Primary Actor	Admin	
Stakeholders and Interests	They want to interact with system, to perform the actions related to buying and selling products	
Preconditions	Application must be running. User should be authenticated and validated to use the software.	
Main Success Scenario		
	ACTOR	SYSTEM
	1. First user must select the user from the list he wants to delete, then press the delete button.	2. System will act against delete button and delete the user from the database.
Alternative	If such username is not available, then system generate error message.	
Post conditions	The Specified User has deleted from the databases.	

Use Case # 11

Use Case Name	Update Product			
Scope	The System under Design			
Level	User-goal			
Primary Actor	Administrators			
Stakeholders and Interests	They want to interact with system, to perform the actions related to buying and selling products			
Preconditions	Application must be running. Actor should be authenticated and validated to use the software.			
Main Success Scenario	<table><tr><td>ACTOR</td><td>SYSTEM</td></tr></table>		ACTOR	SYSTEM
ACTOR	SYSTEM			

	1. After Selecting Update Product from products list.	2. System will take Product_id of which product wants to update and show a form for getting new data related to product
	3. After filling the fields actor must press Update button to update it.	4. Response to Actor action a product detail fields are changed to desired ones and is also edited in the database.
Alternative	If such product_id is not available, then system generate error message.	
Post conditions	The details of product have changed to desired ones and stored in database.	

Use Case # 12

Use Case Name	Delete Product	
Scope	The System under Design	
Level	User-goal	
Primary Actor	Admin	
Stakeholders and Interests	They want to interact with system, to perform the actions related to buying and selling product	
Preconditions	Application must be running. User should be authenticated and validated to use the software.	
Main Success Scenario	ACTOR	SYSTEM
	1. First user must select the product from the list he wants to delete, then press the delete button.	2. System will act against delete button and delete the product details from the database after confirming.
Alternative	If such username is not available, then system generate error message.	
Post conditions	The Specified product has deleted from the databases.	

Use Case # 13

Use Case Name	Update Order
----------------------	---------------------

Scope	The System under Design	
Level	User-goal	
Primary Actor	Admin	
Stakeholders and Interests	They want to interact with system, to perform the actions related to buying and selling products	
Preconditions	Application must be running. Actor should be authenticated and validated to use the software.	
Main Success Scenario	ACTOR	SYSTEM
	1. After Selecting Update Orders	2. System will take order_id of which details are to be updated.
	3. After filling the fields actor must press Update button to change details	4. Response to Actor action a detail fields are changed to desired ones and is also edited in the database
Alternative	If such order_id is not available, then system generate error message.	
Post conditions	The details of order have changed to desired ones and stored in database.	

Use Case # 14

Use Case Name	Delete Order	
Scope	The System under Design	
Level	User-goal	
Primary Actor	Customer, Admin	
Stakeholders and Interests	They want to interact with system, to perform the actions related to buying and selling products	
Preconditions	Application must be running. User should be authenticated and validated to use the software.	
Main Success Scenario	ACTOR	SYSTEM
	1. First Actor must select the order from the order list he want to delete, then press the delete button.	2. System will act against delete button and delete the order details from the database after confirming.
Alternative	If such Order_id is not available, then system generate error message.	
Post conditions	The Specified Order has deleted from the databases.	

Use case # 15

Use Case Name	Generate Reports	
Scope	The System under Design	
Level	User-goal	
Primary Actor	Admin	
Stakeholders and Interests	They want the automated system that efficiently uses to minimize academic problems.	
Preconditions	Application must be running. Actor should be authenticated and validated to use the software.	
Main Success Scenario	ACTOR	SYSTEM
	1. When actor press Generate reports button	2. System takes data from database like Products sold, customers and profit.
	3. After clicking button “Save Report” . System act against it.	4. Response to Actor action, report will be download to your computer
Alternative	When no such data is present then no report can be created	
Post conditions	Generated report will be saved to your PC.	

Use Case # 16

Use Case Name	See Recommendations					
Scope	The System under Design					
Level	User-goal					
Primary Actor	Customers					
Stakeholders and Interests	They want to interact with system, to perform the actions related to buying products					
Preconditions	Application must be running. Actor should be authenticated and validated to use the software.					
Main Success Scenario	<table><tr><th>ACTOR</th><th>SYSTEM</th></tr><tr><td>1. Actor clicks on the “Recommendation” option in the menu.</td><td>2. System will take the customers data and gives to recommendation system; recommendation system give the personalized recommended products</td></tr></table>		ACTOR	SYSTEM	1. Actor clicks on the “Recommendation” option in the menu.	2. System will take the customers data and gives to recommendation system; recommendation system give the personalized recommended products
ACTOR	SYSTEM					
1. Actor clicks on the “Recommendation” option in the menu.	2. System will take the customers data and gives to recommendation system; recommendation system give the personalized recommended products					

	3. After getting the recommended products, actor can buy the products.	
Alternative	In case of any error, system show the error message.	
Post conditions	The actor gets the personalized recommendations from recommendation system.	

Use Case # 17

Use Case Name	Recommendations Patterns	
Scope	The System under Design	
Level	User-goal	
Primary Actor	Admin	
Stakeholders and Interests	They want to interact with system, to perform the actions related to recommendations	
Preconditions	Application must be running. Actor should be authenticated and validated as admin to use the software.	
Main Success Scenario		
	ACTOR	SYSTEM
	1. Actor clicks on the “Recommendation” option in the menu.	2. System will show options related to recommendation system.
	3. Actor selects the” Recommendation patterns”	4. System shows the major recommendation pattern
Alternative	In case of any error, system show the error message.	
Post conditions	The actor gets the visual graph of patterns of recommended products	

Use Case # 18

Use Case Name	Change Recommendations Model	
Scope	The System under Design	
Level	User-goal	
Primary Actor	Admin	
Stakeholders and Interests	They want to interact with system, to perform the actions related to recommendations	
Preconditions	Application must be running. Actor should be authenticated and validated as admin to use the software.	
Main Success Scenario		
	ACTOR	SYSTEM
	1. Actor clicks on the “Recommendation” option in the menu.	2. System will show options related to recommendation system.

	3. Actor clicks on the “Change Model” to select other model from list.	4. System shows list of models created.
	5. Actors select the model and click on “submit” button.	6. System gets the model and activate to give the updated recommendations.
Alternative	In case of any error, system show the error message.	
Post conditions	The recommendation model changed in the recommendation system of the website.	

Use Case # 19

Use Case Name	Sold Recommendations products	
Scope	The System under Design	
Level	User-goal	
Primary Actor	Admin	
Stakeholders and Interests	They want to interact with system, to perform the actions related to recommendations	
Preconditions	Application must be running. Actor should be authenticated and validated as admin to use the software.	
Main Success Scenario	ACTOR	SYSTEM
	1. Actor clicks on the “Recommendation” option in the menu.	2. System will show options related to recommendation system.
	3. Actor clicks on the “view sold recommended product’	4. System shows list of products that are purchased through recommendations
Alternative	In case of any error, system show the error message.	
Post conditions	Sold recommendations products are shown	

Chapter 3:

Design Document

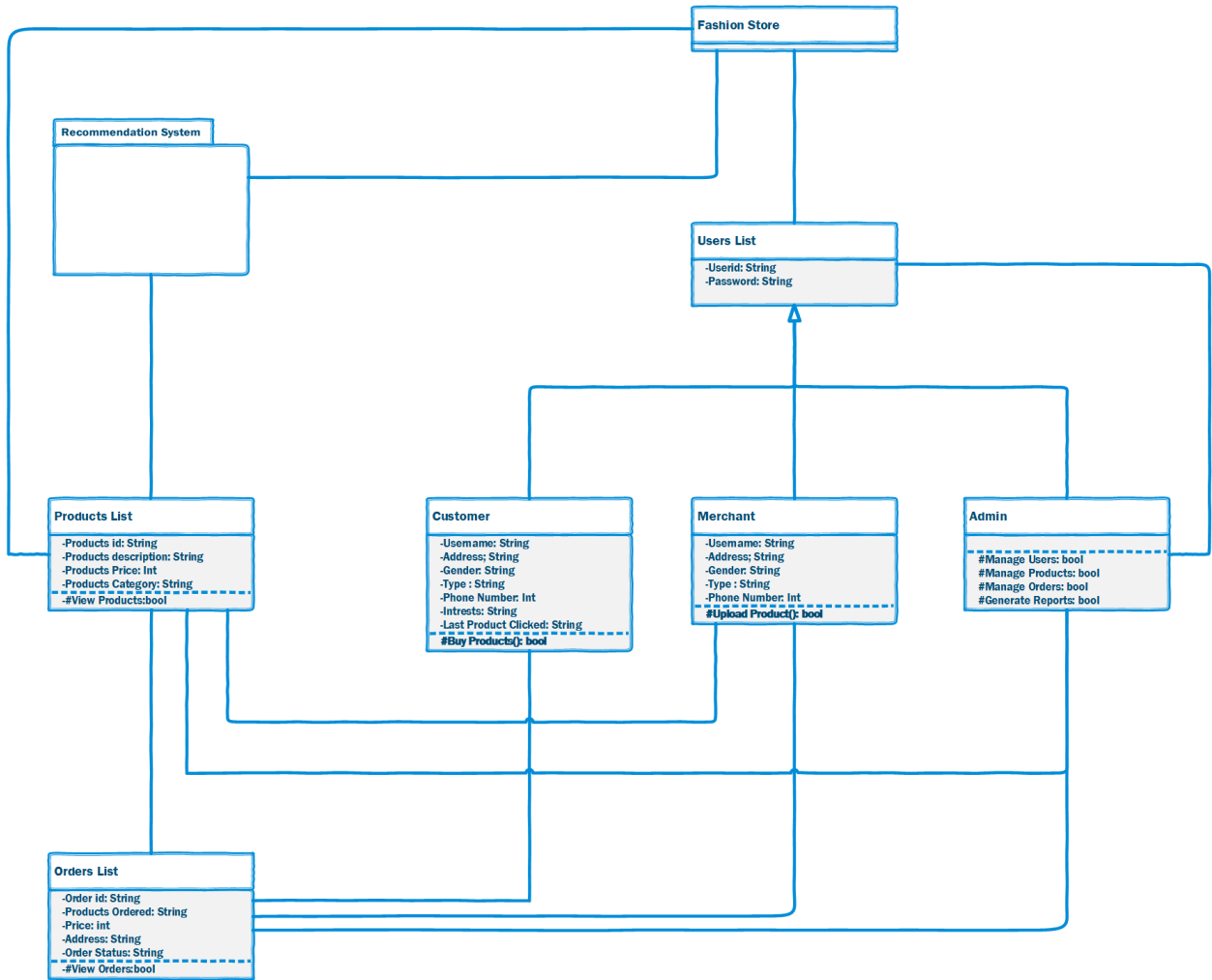
3.1. Introduction:

Third deliverable is all about the software design. In the previous deliverable, analysis of the system is completed. So, we understand the current situation of the problem domain. Now we are ready to strive for a solution for the problem domain by using object-oriented approach. Following artifacts must be included in the 3rd deliverable.

1. Domain Model
2. System Sequence Diagram
3. Sequence Diagram
4. Collaboration Diagram
5. Operation Contracts
6. Design Class Diagram
7. State Transition Diagram
8. Data Model

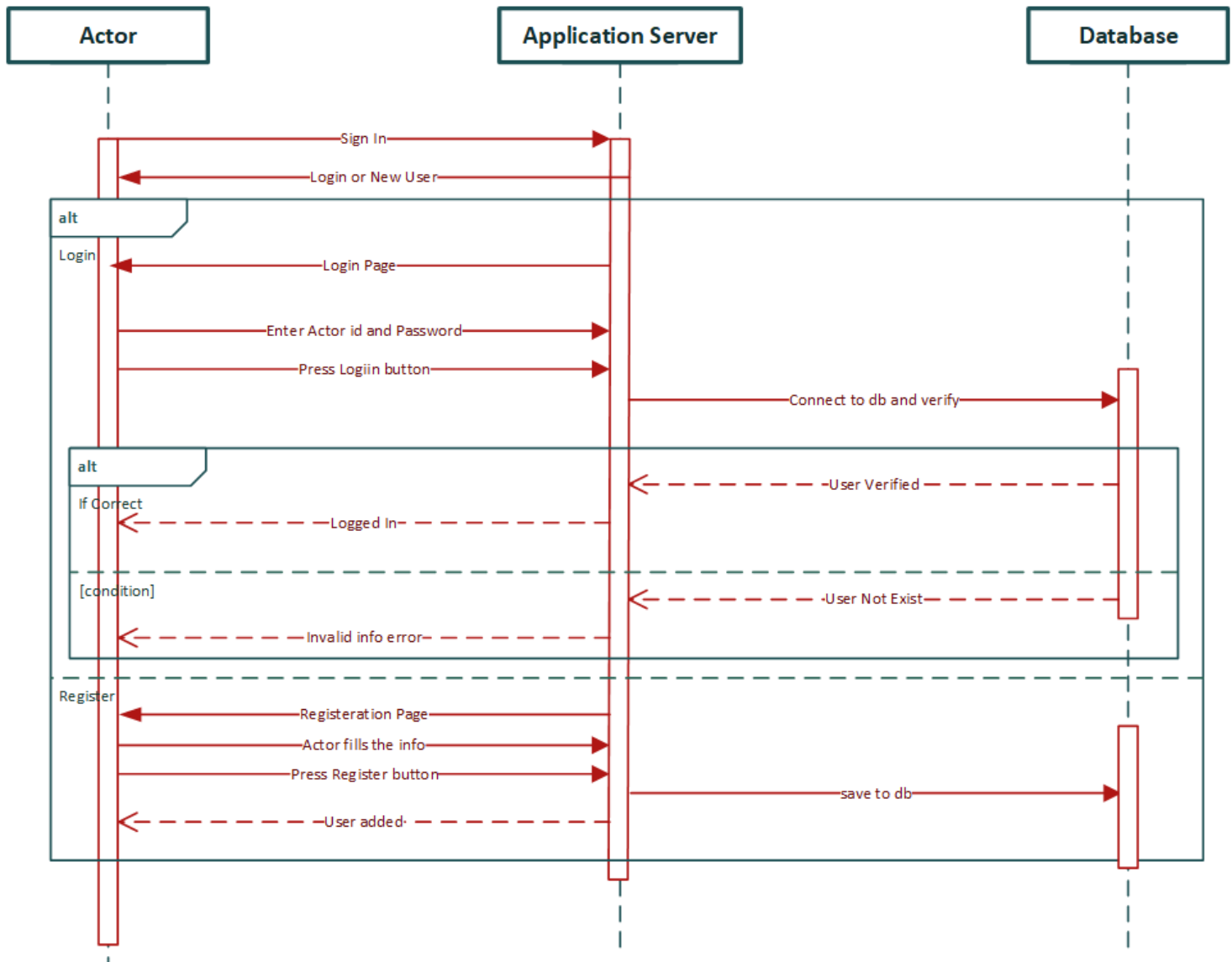
Now we discuss these artifacts one by one as follows:

3.2. Domain Model

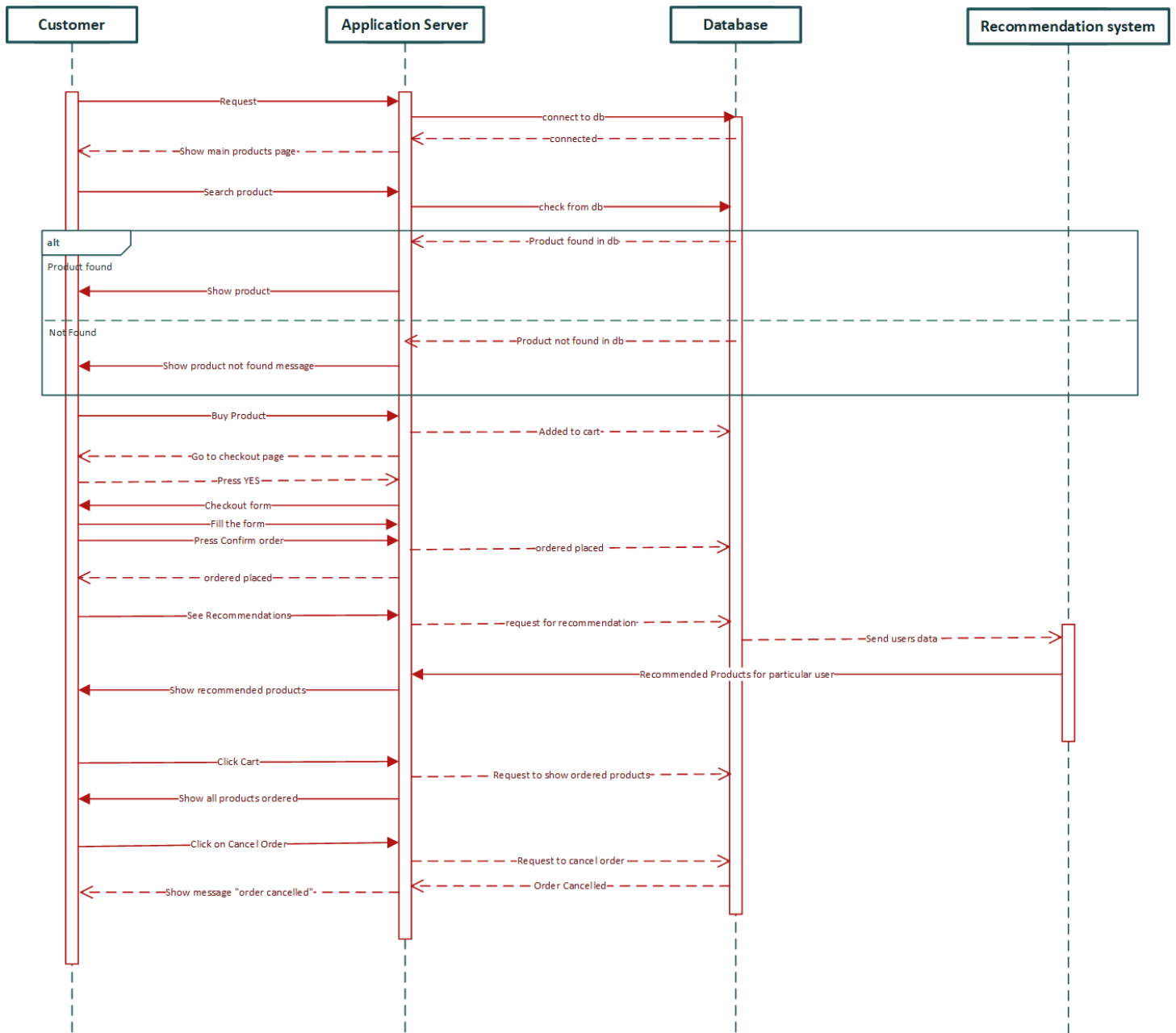


3.3. Sequence Diagram

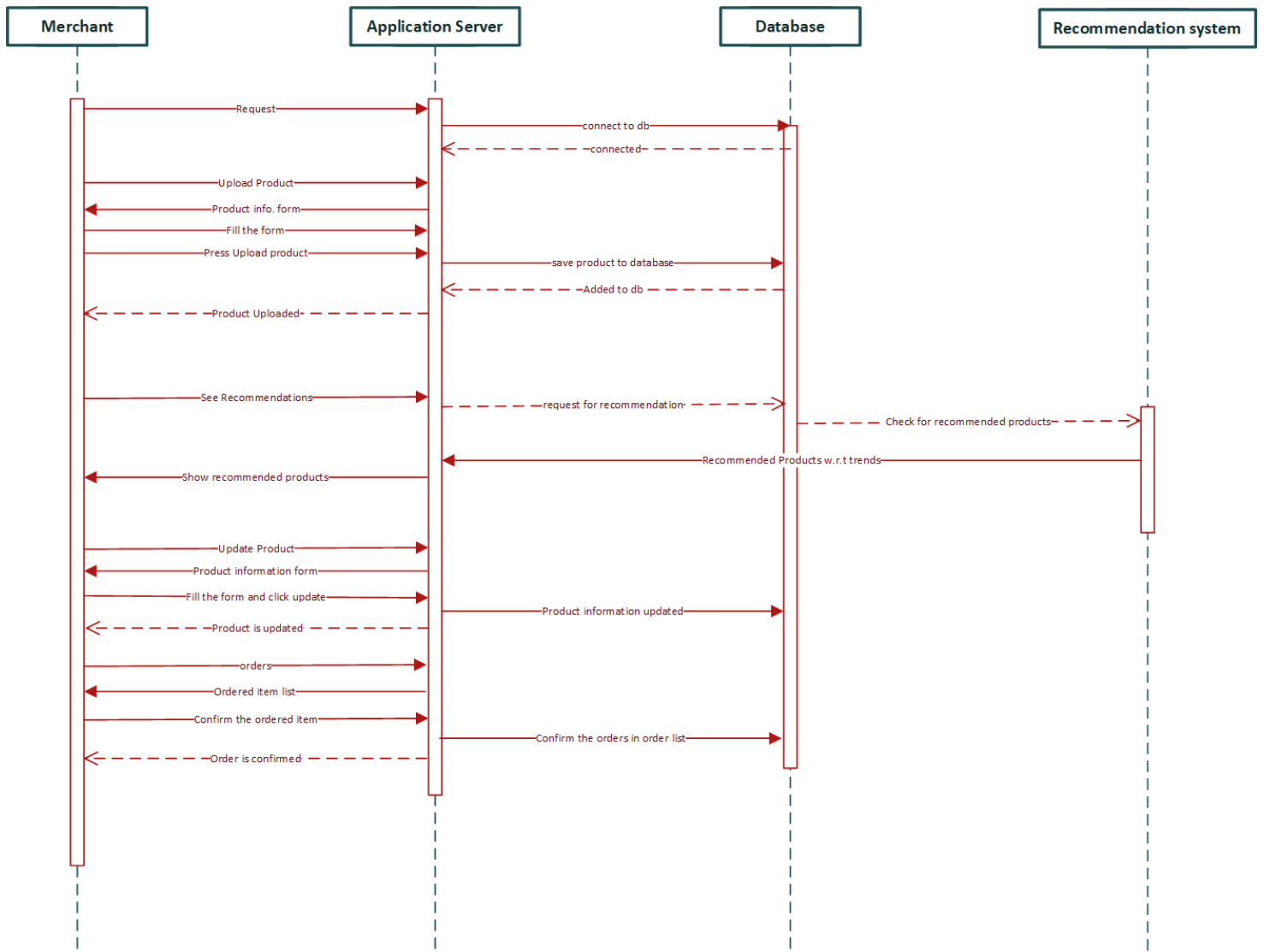
Login & Registration Module



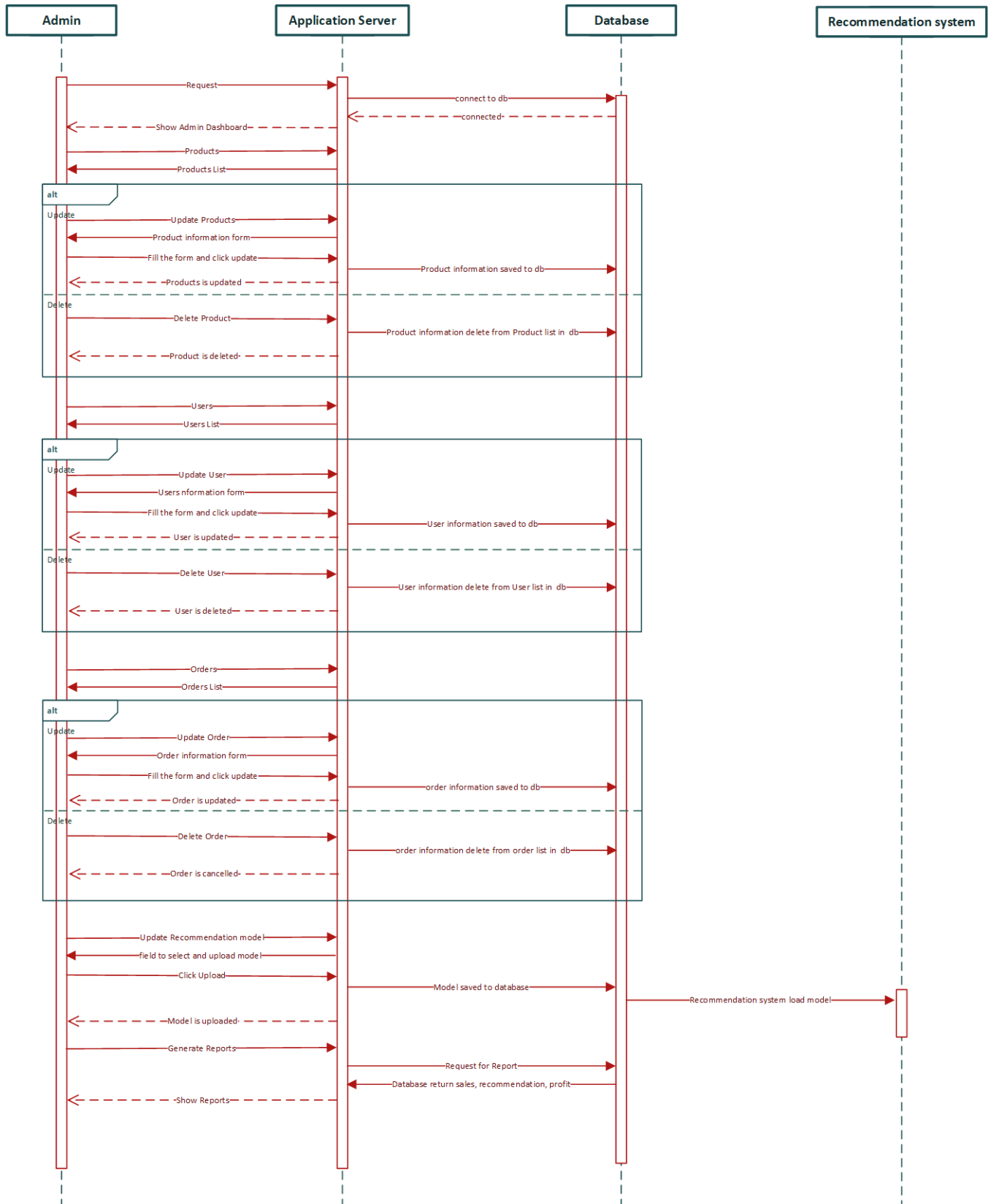
Customer Module



Merchant Module



Admin



3.4. Operation Contracts

Login

- **Preconditions:** The application must be running.
- **Inputs:** Actor ID, Password
- **Outputs:** Main menu of the application
- **Postconditions:** Actor successfully logged in.
- **Exceptions:** If the actor ID or password is invalid, the system asks for re-entry.

Log Out

- **Preconditions:** Application must be running, and user must be logged in.
- **Inputs:** None
- **Outputs:** None
- **Postconditions:** Actor successfully logged out.

User's Profile

- **Preconditions:** Application must be running, and user must be logged in.
- **Inputs:** Profile button press
- **Outputs:** User profile information form
- **Postconditions:** Actor views their profile information.

Change Password

- **Preconditions:** Application must be running, and user must be logged in.
- **Inputs:** Current password, New password, Confirm new password
- **Outputs:** Success message
- **Postconditions:** Password changed successfully.
- **Exceptions:** If the current password is incorrect, the system prompts for re-entry.

Upload Product

- **Preconditions:** Application must be running.
- **Inputs:** Product details form
- **Outputs:** Product added confirmation
- **Postconditions:** New product added to the database.

View Products

- **Preconditions:** Application must be running.

- **Inputs:** Home button press
- **Outputs:** List of available products
- **Postconditions:** Actor views all available products.

Add to Cart

- **Preconditions:** Application must be running, user must be logged in.
- **Inputs:** Product selection, Transaction details
- **Outputs:** Order confirmation
- **Postconditions:** Product added to the user's cart.

Add a Customer

- **Preconditions:** Application must be running.
- **Inputs:** Customer information form
- **Outputs:** Success message
- **Postconditions:** New customer added to the database.

Update User

- **Preconditions:** Application must be running.
- **Inputs:** User selection, Updated user information
- **Outputs:** Success message
- **Postconditions:** User information updated in the database.

Delete User

- **Preconditions:** Application must be running.
- **Inputs:** User selection
- **Outputs:** Success message
- **Postconditions:** User deleted from the database.

Update Product

- **Preconditions:** Application must be running.
- **Inputs:** Product selection, Updated product information
- **Outputs:** Success message
- **Postconditions:** Product information updated in the database.

Delete Product

- **Preconditions:** Application must be running.
- **Inputs:** Product selection
- **Outputs:** Success message
- **Postconditions:** Product deleted from the database.

Update Order

- **Preconditions:** Application must be running.
- **Inputs:** Order selection, Updated order details
- **Outputs:** Success message
- **Postconditions:** Order details updated in the database.

Delete Order

- **Preconditions:** Application must be running.
- **Inputs:** Order selection
- **Outputs:** Success message
- **Postconditions:** Order deleted from the database.

Generate Reports

- **Preconditions:** Application must be running.
- **Inputs:** Generate report button press
- **Outputs:** Report download
- **Postconditions:** Report generated and saved to the user's device.

See Recommendations

- **Preconditions:** Application must be running.
- **Inputs:** Recommendation button press
- **Outputs:** Personalized product recommendations
- **Postconditions:** Actor views personalized recommendations.

Recommendations Patterns

- **Preconditions:** Application must be running.
- **Inputs:** Recommendation patterns button press
- **Outputs:** Recommendation patterns graph

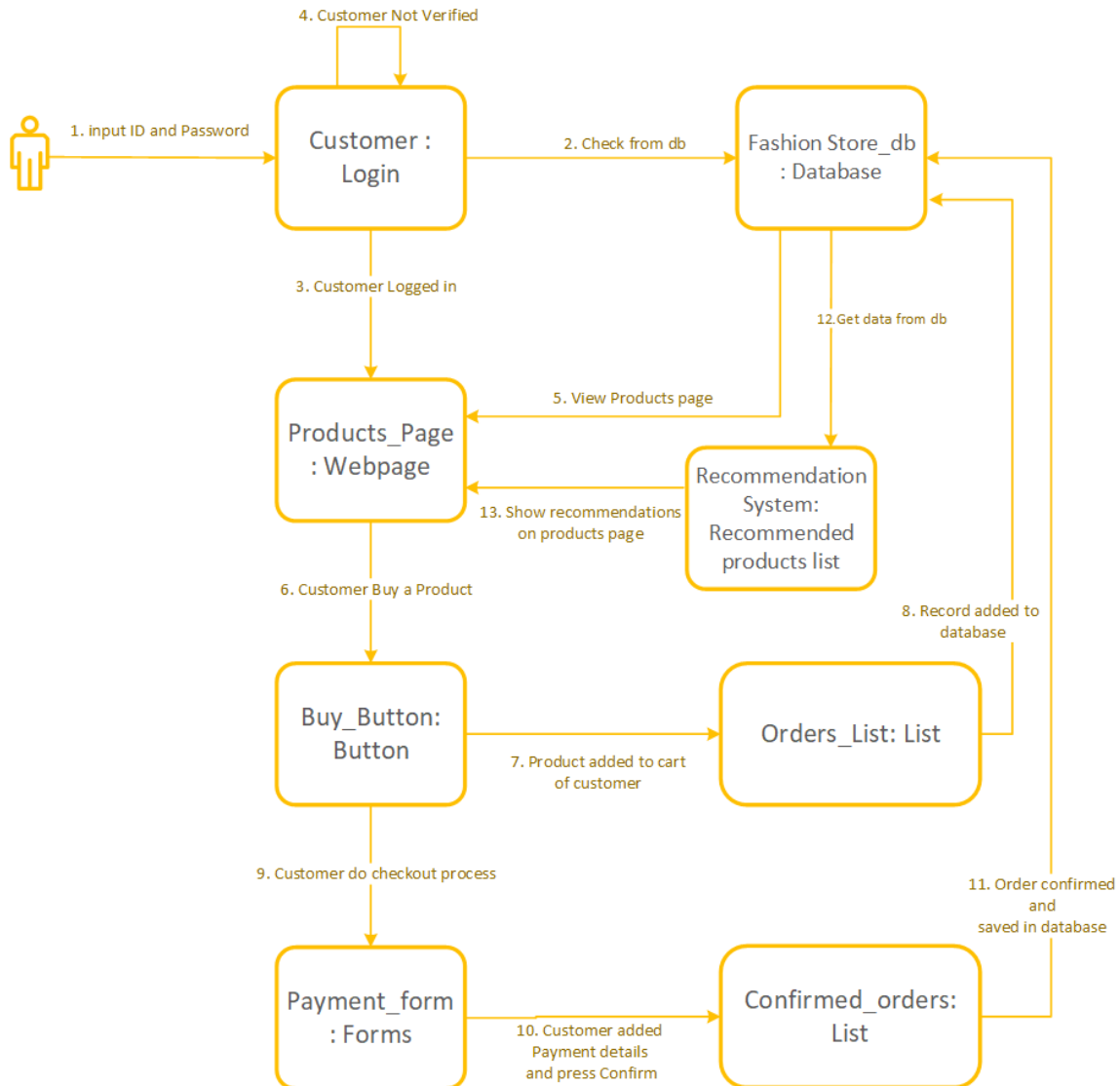
- **Postconditions:** Actor views recommendation patterns.

Recommendations Model

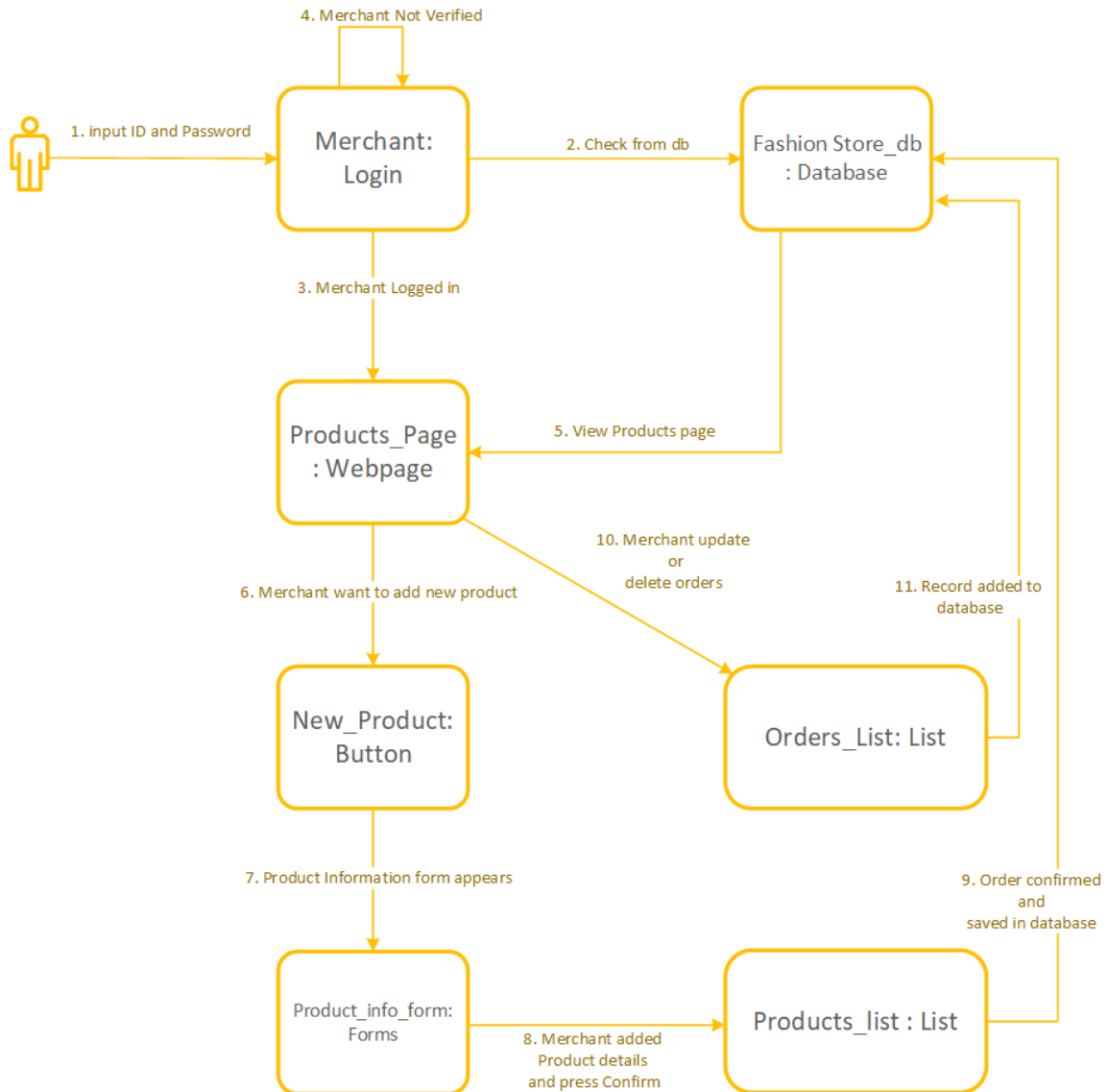
- **Preconditions:** Application must be running.
- **Inputs:** Change model button press, Model file upload
- **Outputs:** Success message
- **Postconditions:** Recommendation model updated in the system.

3.5. Collaboration Diagram

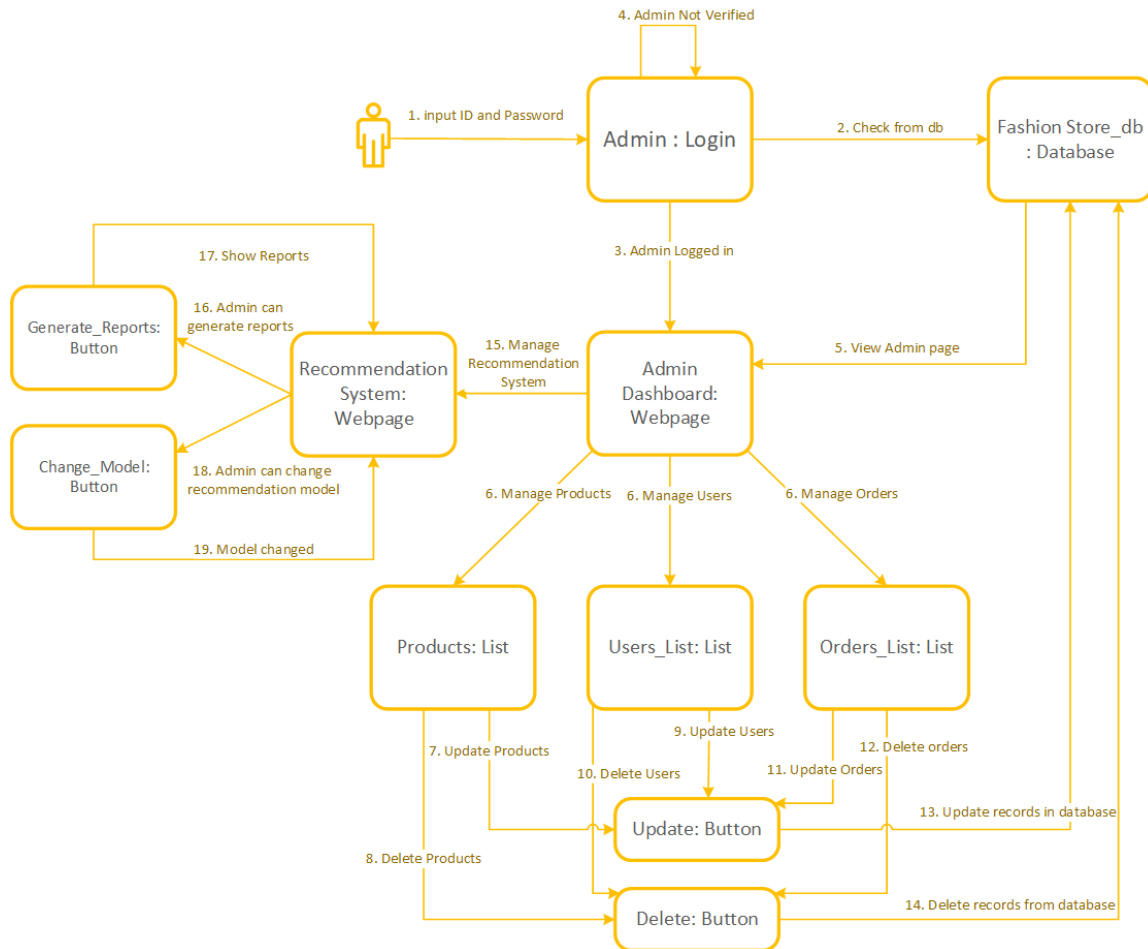
Customer Collaboration Diagram



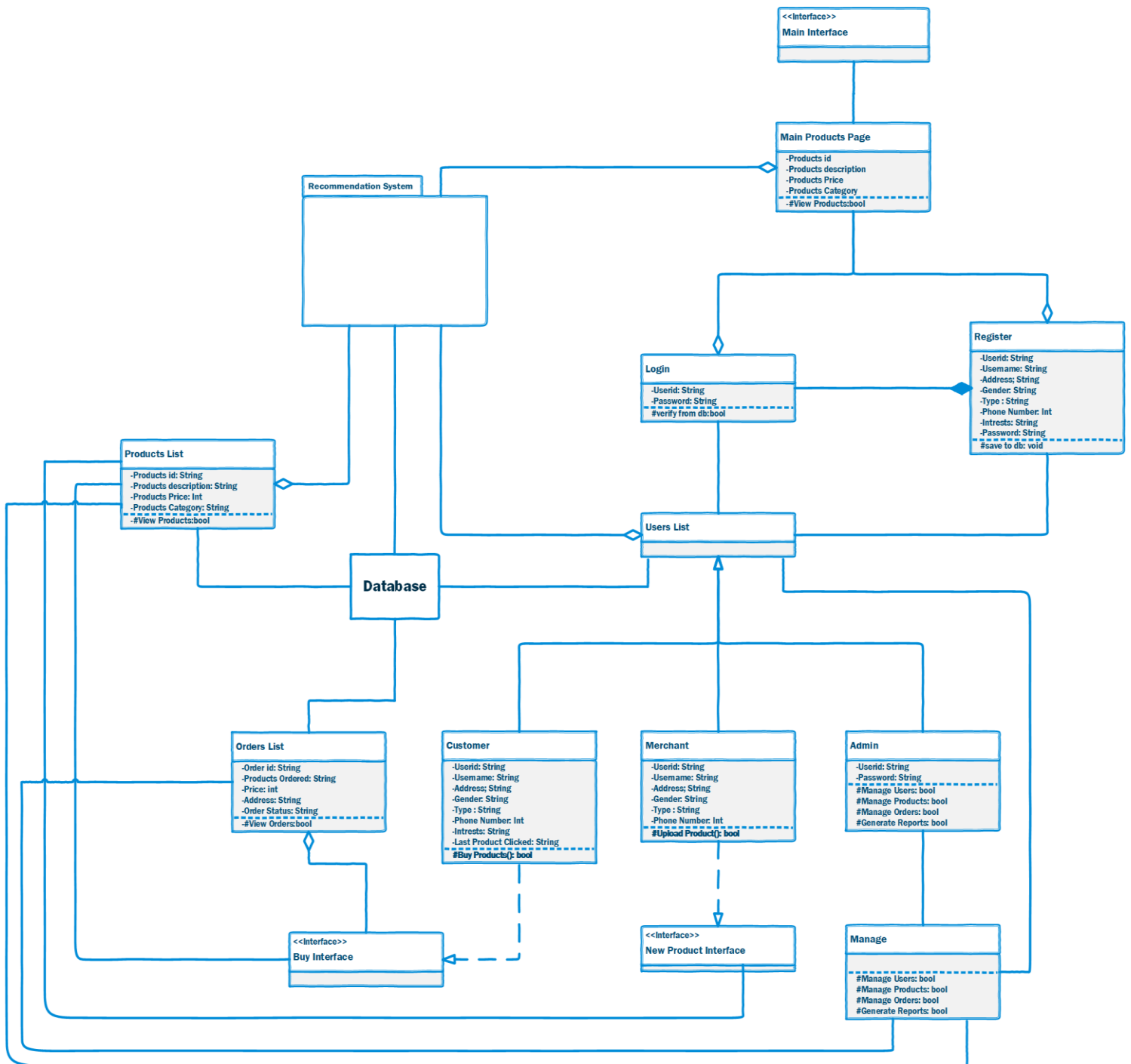
Merchant Collaboration Diagram



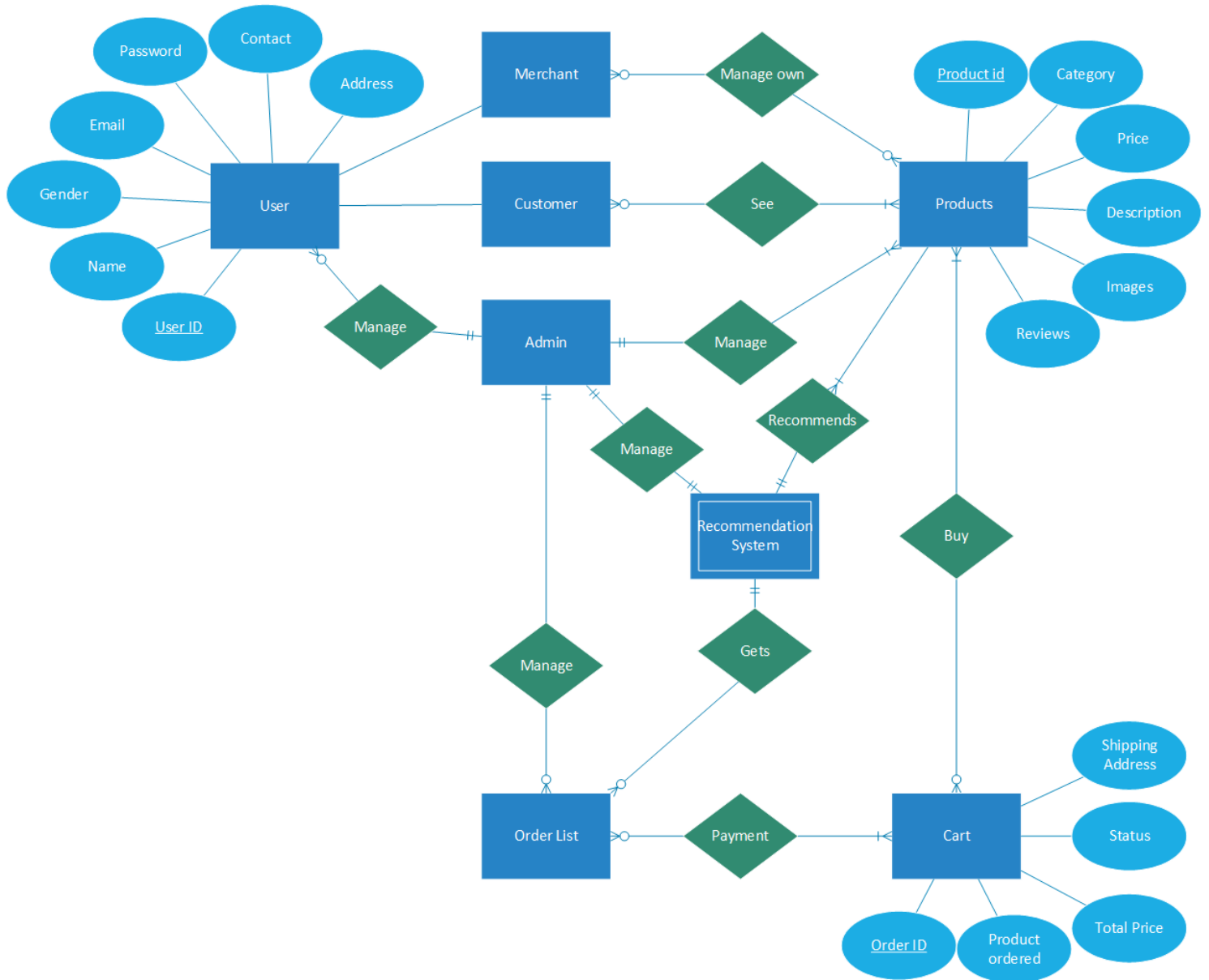
Admin Collaboration Diagram



3.7. Class Diagram

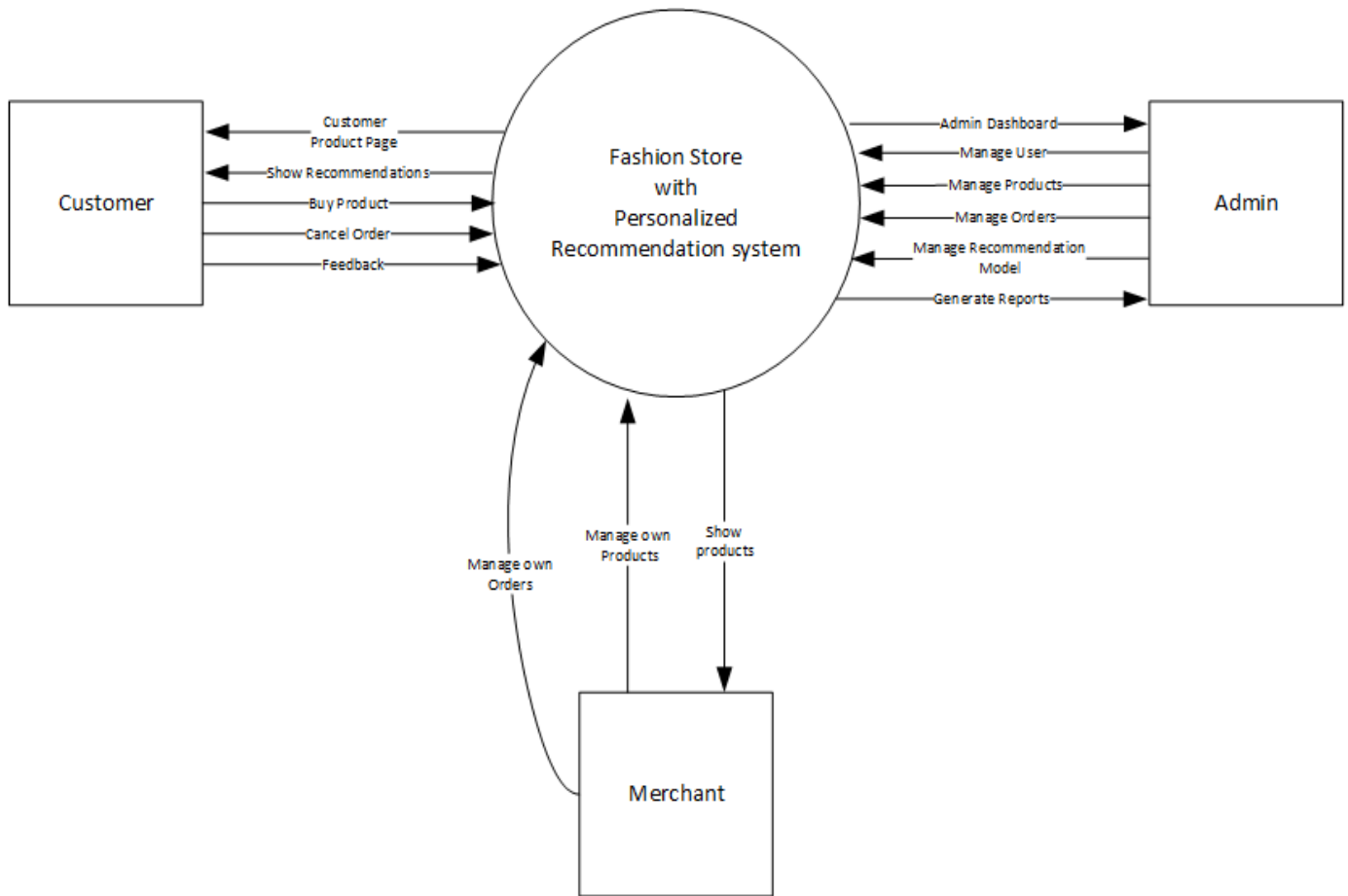


3.2. Entity Relationship Diagram:

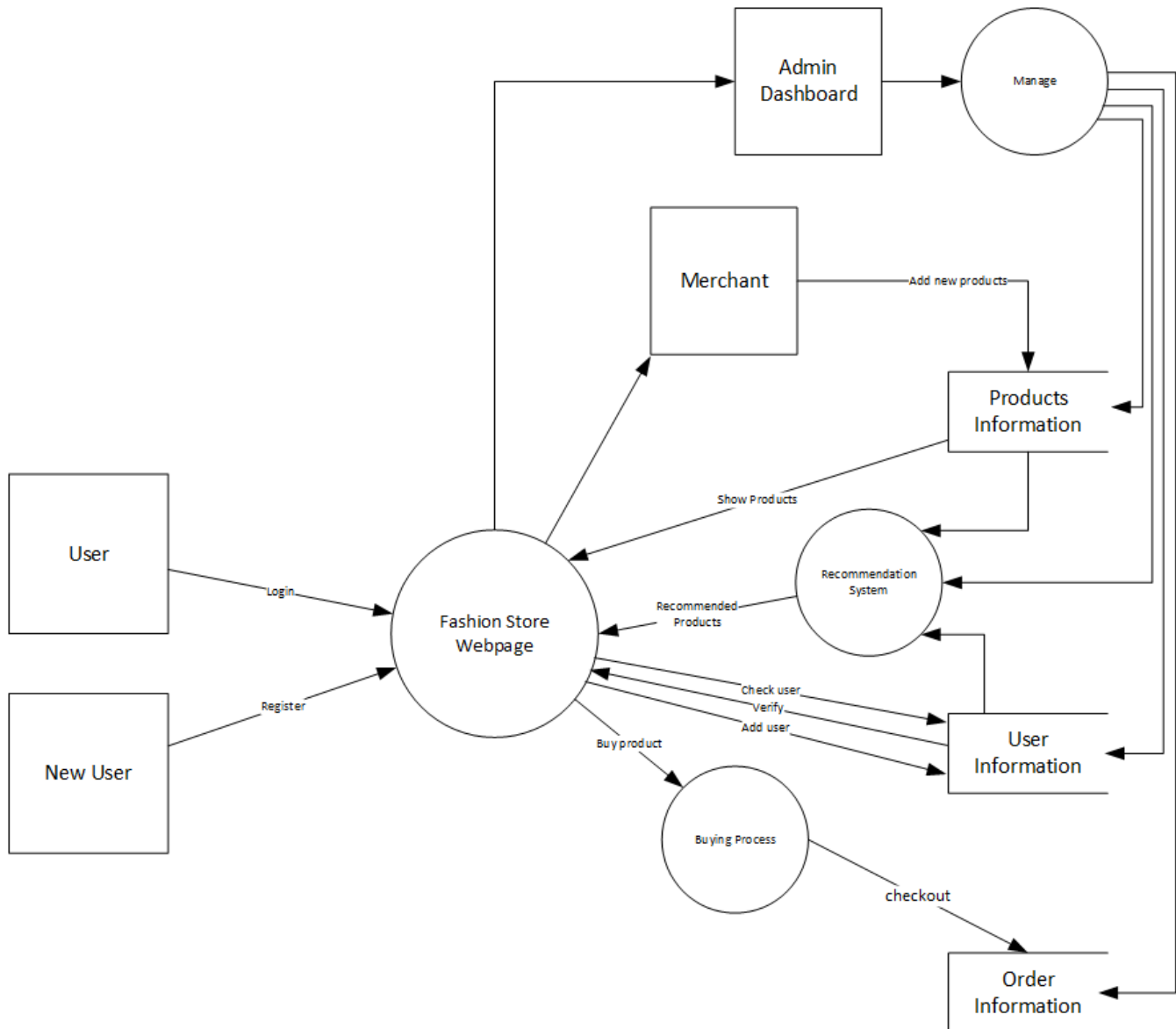


3.3. Data flow diagram (Functional Model)

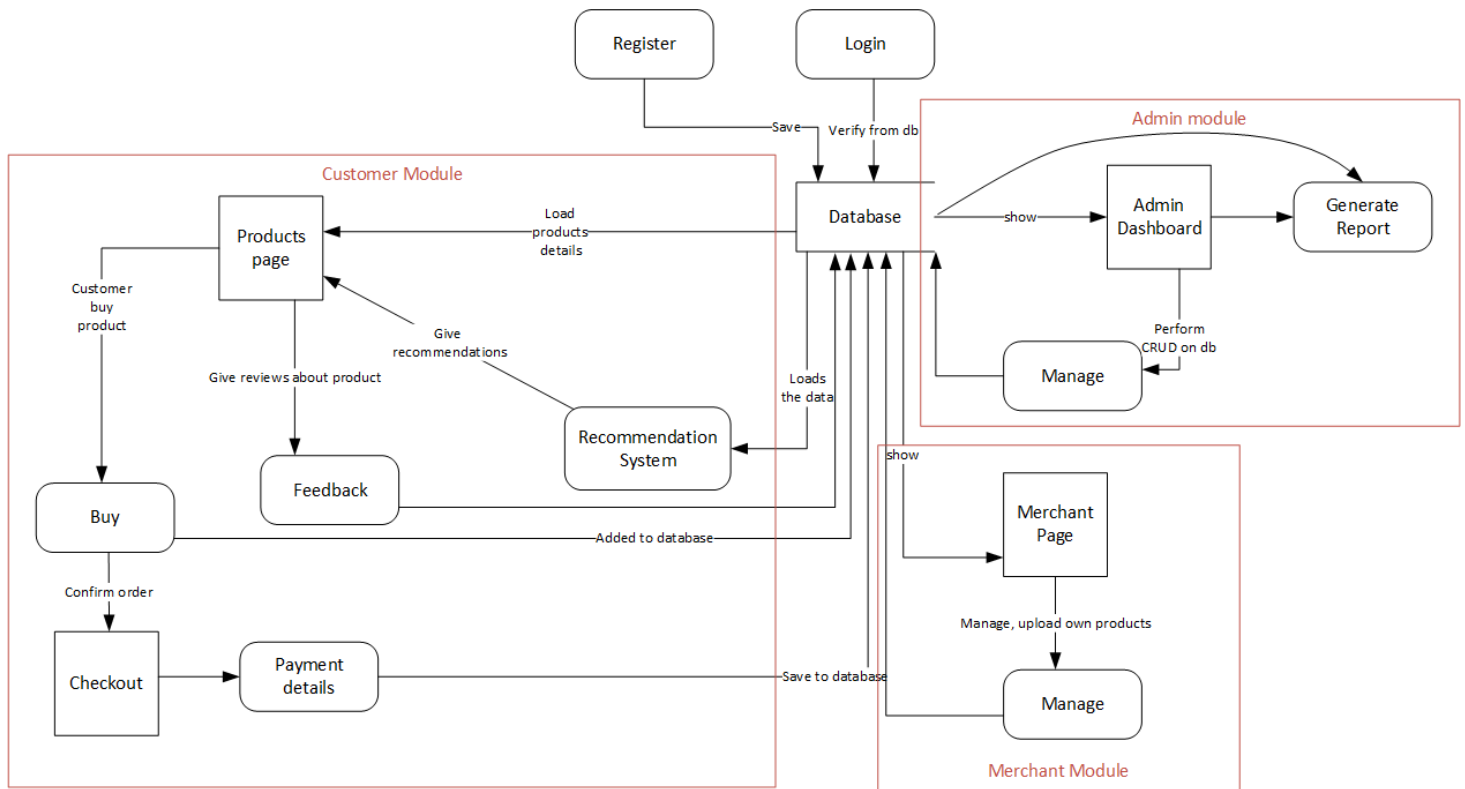
Level 0



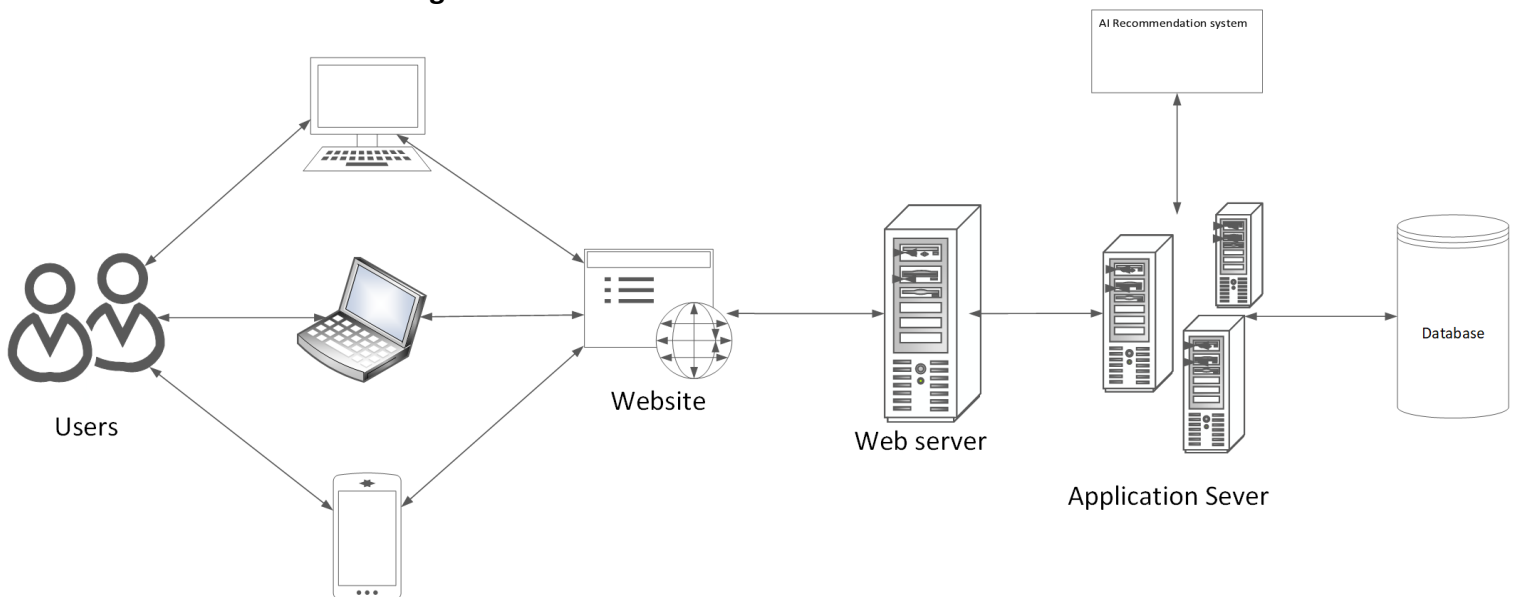
Level 1



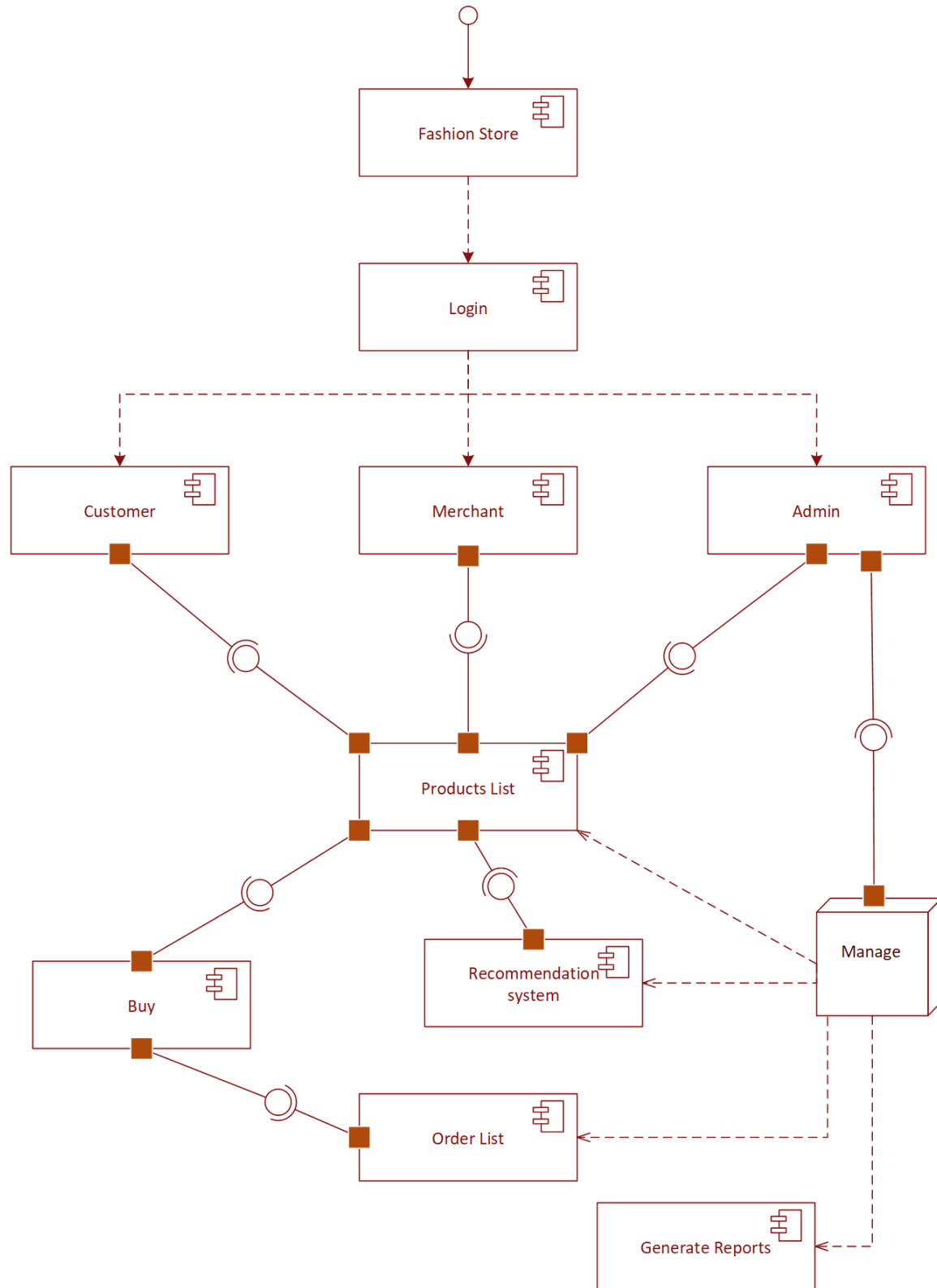
Level 2



3.5. Architectural design

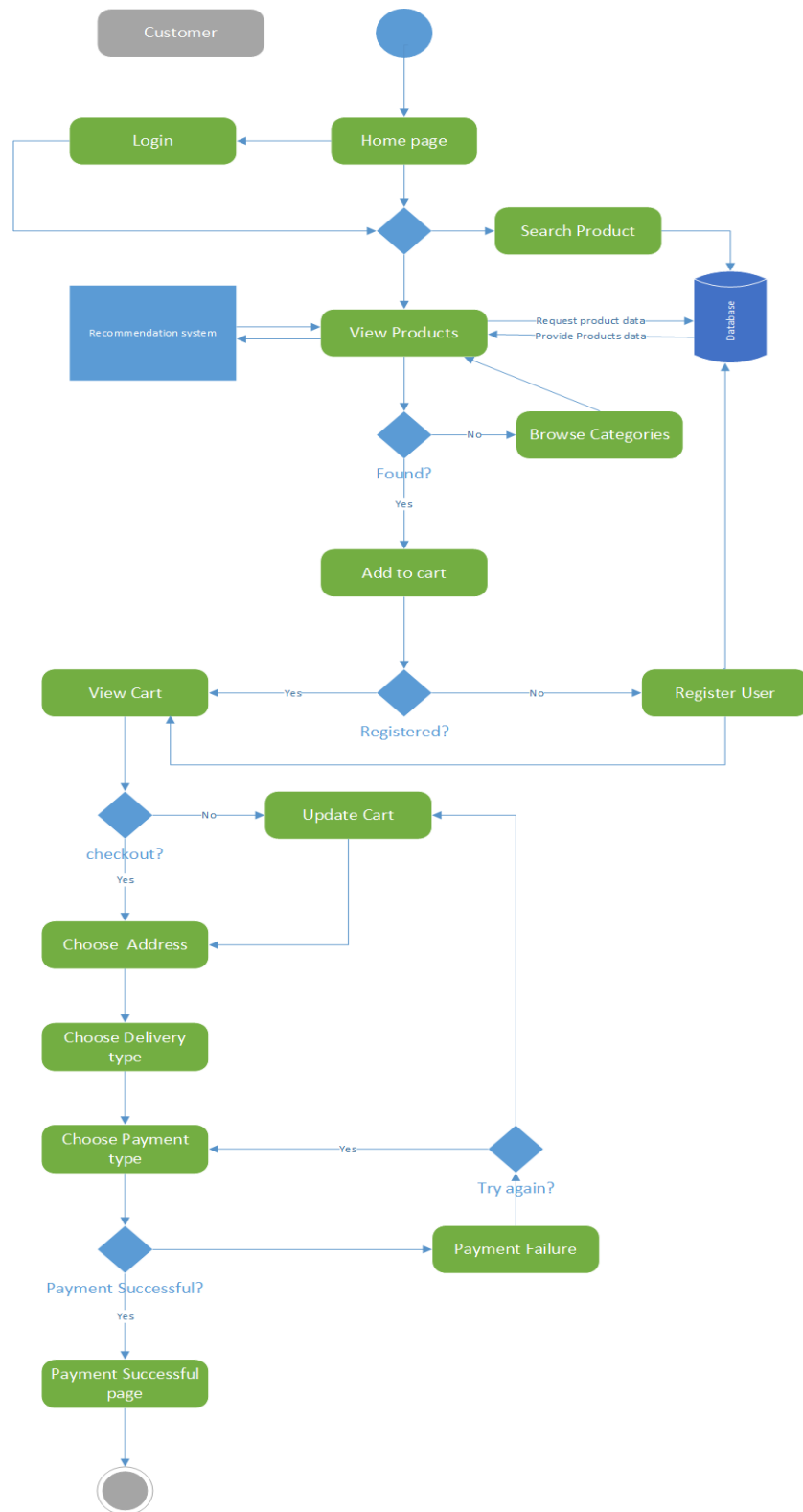


3.6. Component Level Design

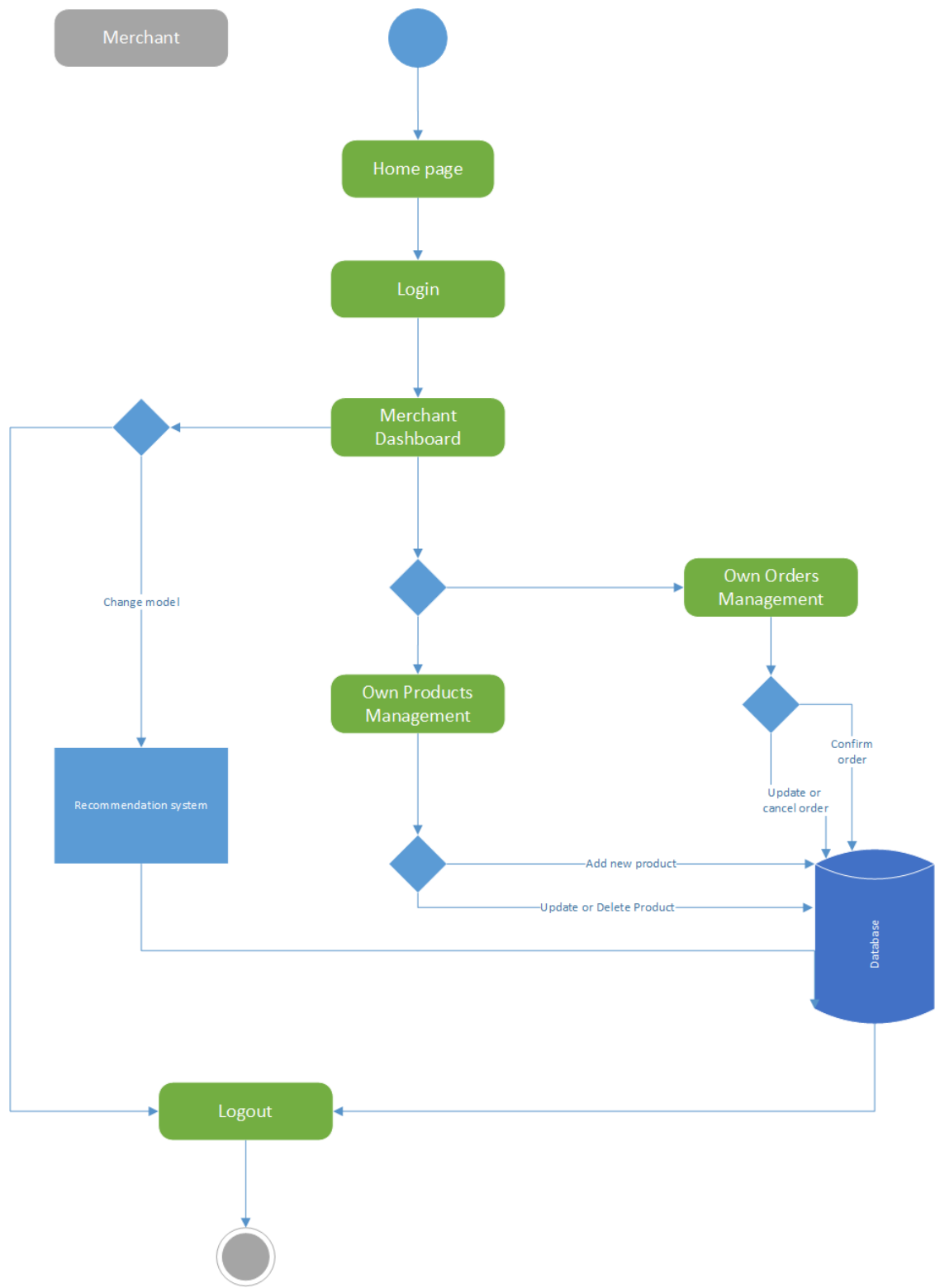


Activity Diagrams

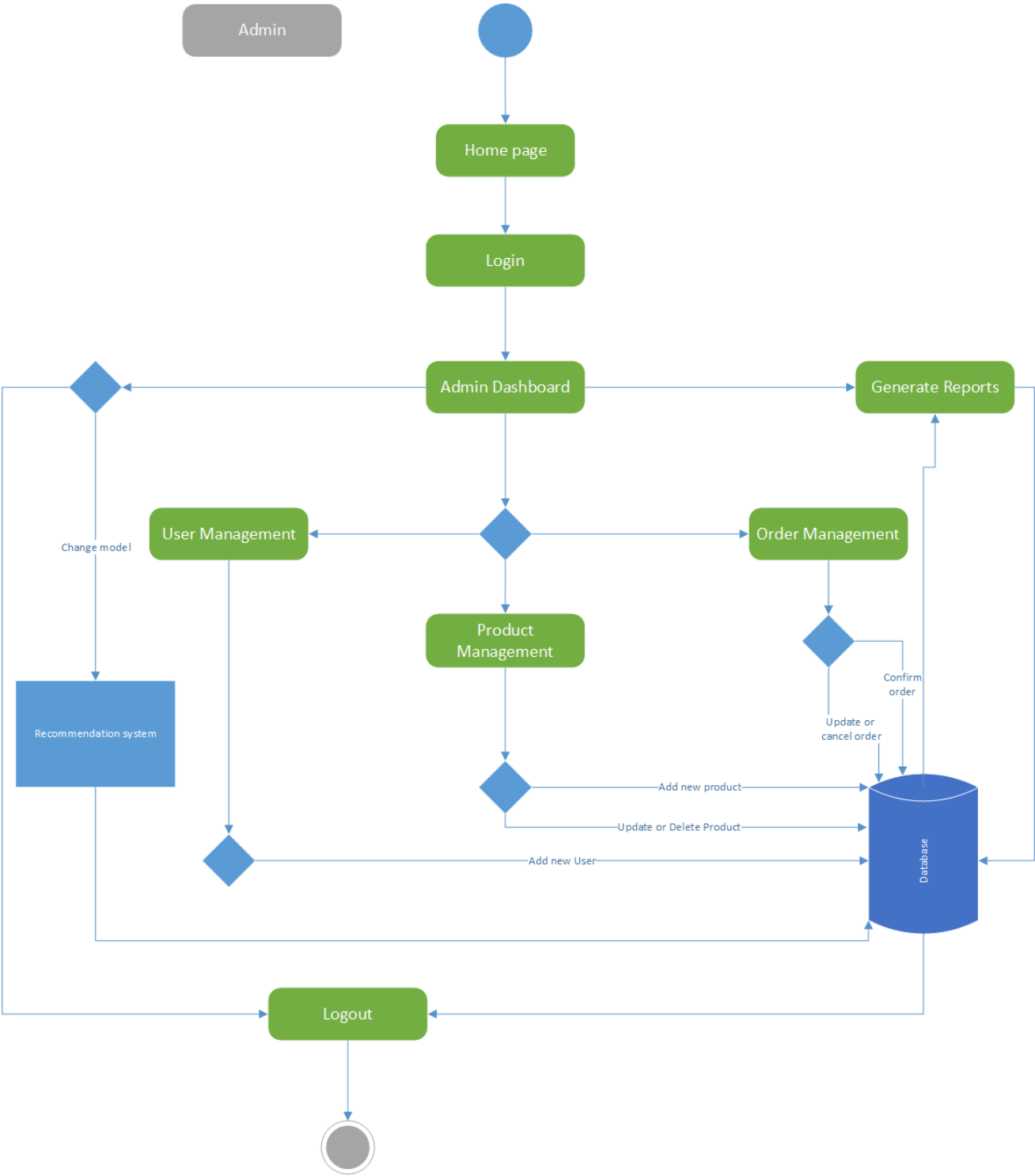
Customer Module



Merchant Module



Admin Module



Chapter 4:

User interface Design

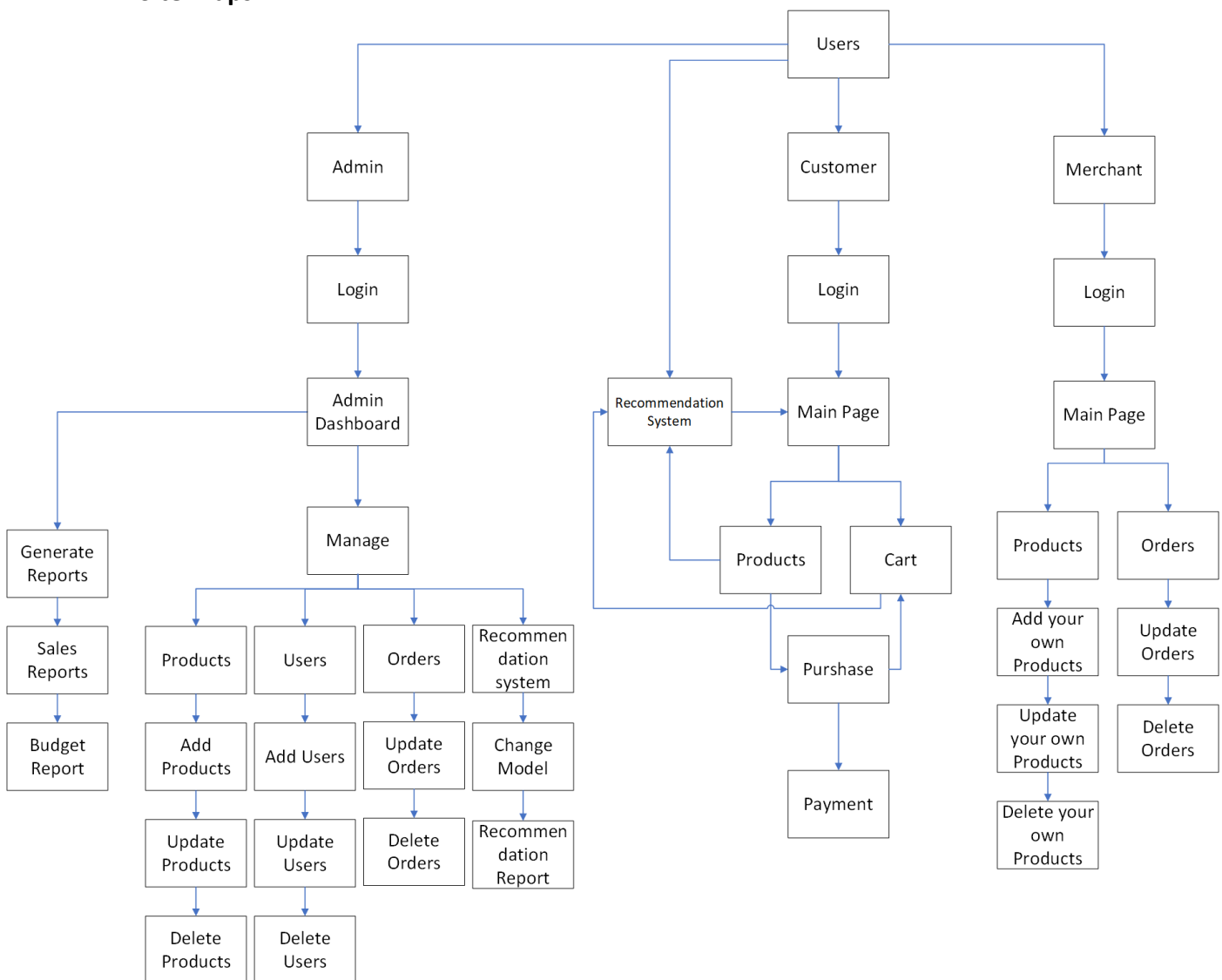
4.1. Introduction

A user interface design consists of three main parts:

Page elements should be visualized on paper before building them in the computer. Just as you draw a site map to plan the site, use cartoons and storyboards to begin blocking out the site's appearance and navigational scheme.

1. Site maps
2. Storyboards
3. Navigational maps
4. Traceability Matrix

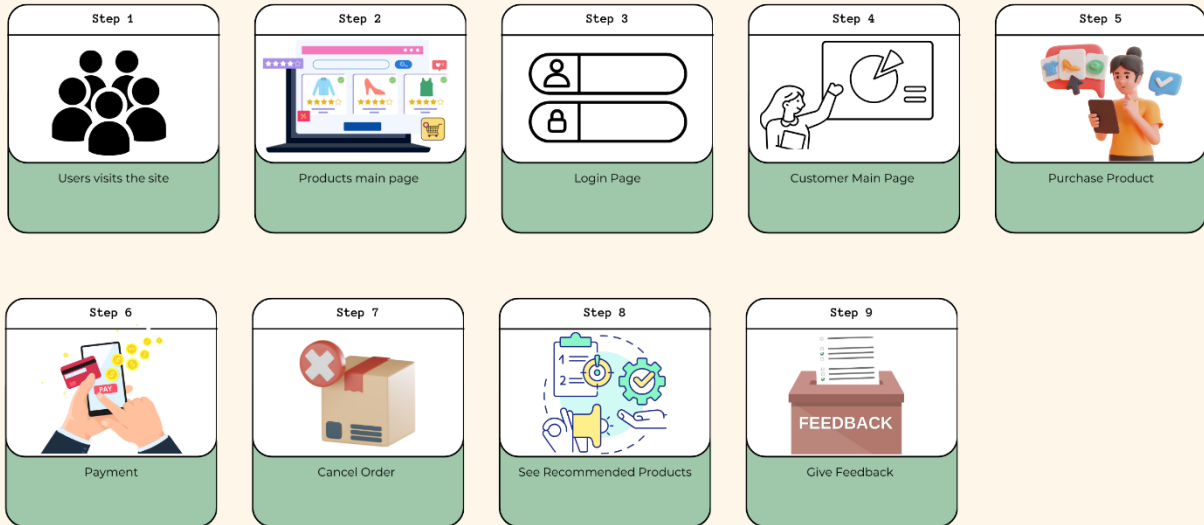
4.2. Site Maps



4.3. Story boards

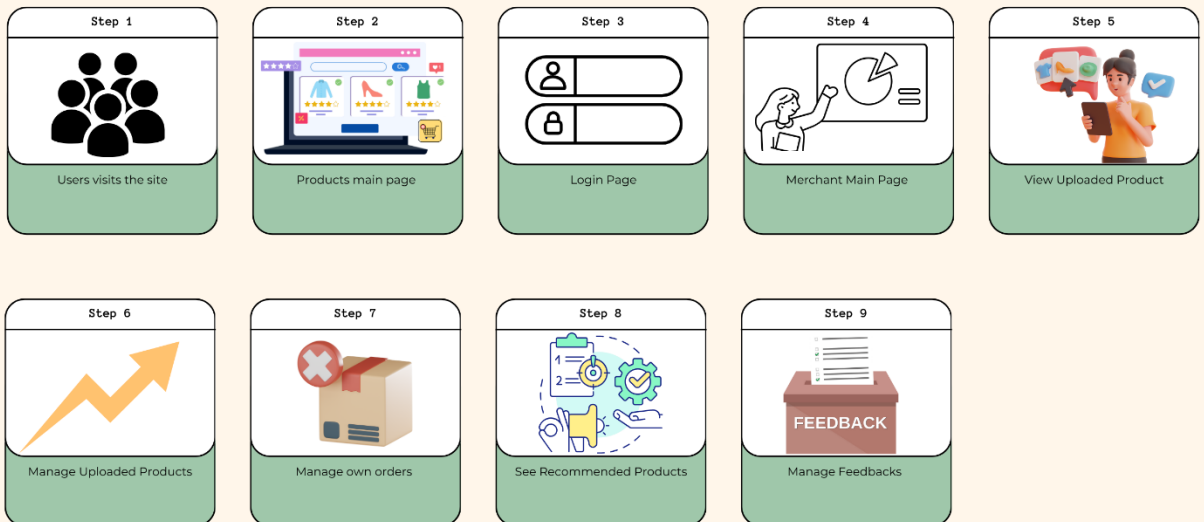
Customer Module

FASHION STORE CUSTOMER STORY BOARD



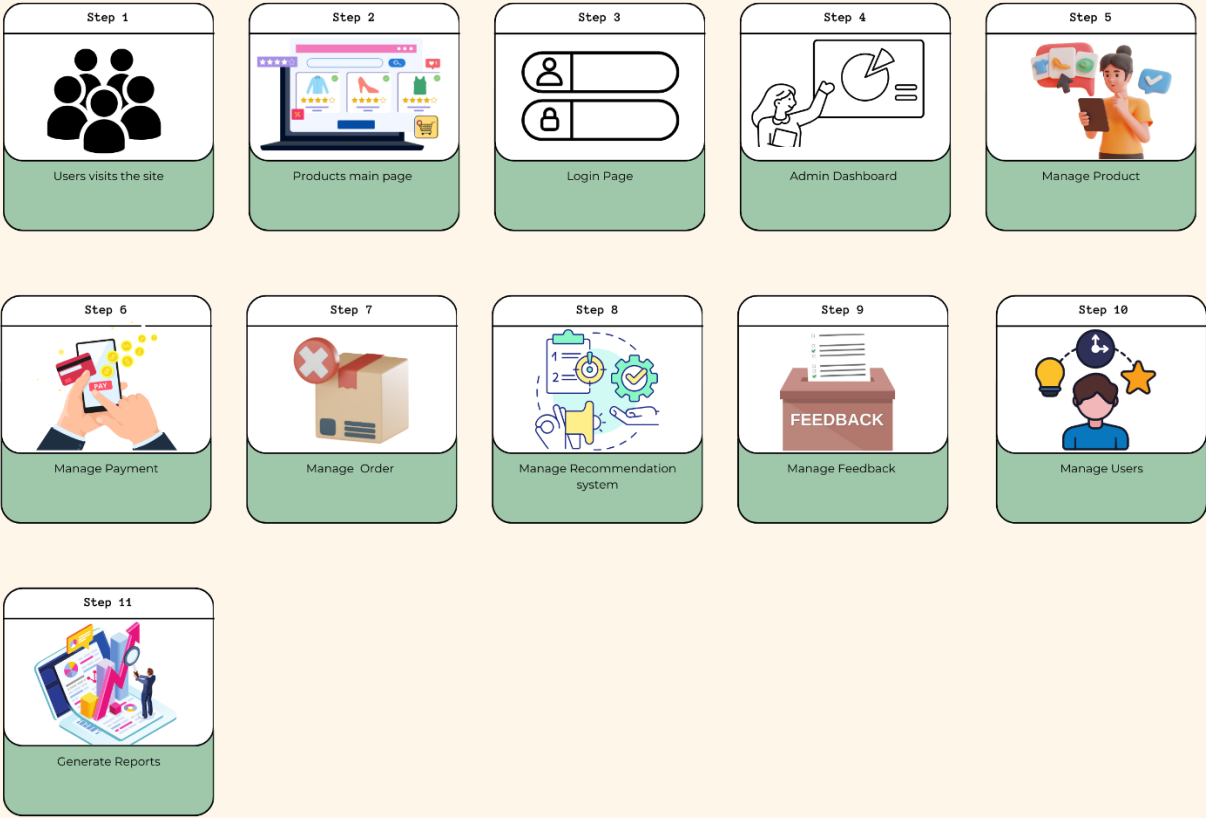
Merchant Module

FASHION STORE MERCHANT STORY BOARD



Admin Module

FASHION STORE
ADMIN STORY BOARD



4.5 Trace-ability Matrix

Requirement ID	Description	Functional Component/Module
REQ-001	User Registration and Authentication	Authentication Module
REQ-002	User Profile Management	User Profile Module
REQ-003	Product Catalog Display	Product Management Module
REQ-004	Search and Filters Functionality	Search Module
REQ-005	Shopping Cart Management	Shopping Cart Module
REQ-006	Checkout Process	Checkout Module
REQ-007	AI-Powered Recommendation Engine Integration	Recommendation Engine Module
REQ-008	User Feedback and Ratings	Feedback and Rating Module
REQ-009	Order Management	Order Management Module
REQ-010	Responsive Design Implementation	Frontend/UI Module
REQ-011	Admin Panel for System Management	Admin Panel Module
REQ-012	Reporting and Analytics Tools	Analytics Module
REQ-013	Security Measures Implementation	Security Module
REQ-014	Notification System	Notification Module

Chapter 5:

Software Testing

5.1 Introduction:

The "E-Commerce website with AI-powered personalized recommendation system" aims to revolutionize online shopping by creating a user-centric platform that not only offers an extensive array of products but also employs cutting-edge AI technology to enhance the overall shopping experience. In a world where online shoppers seek personalized and efficient solutions, this project becomes imperative. The significance lies in creating a seamless and secure environment for user registration, profile management, and efficient shopping processes. The AI-powered recommendation engine, fuelled by TensorFlow or Scikit-learn, adds a layer of intelligence, providing users with personalized product suggestions based on their preferences. This not only elevates user satisfaction but also drives sales by presenting relevant and enticing options. The project's robust backend, utilizing Python and MySQL, ensures data integrity and efficient order management. The deployment on a Heroku Django server guarantees accessibility, while the responsive design caters to users across devices. The inclusion of an admin panel, reporting tools, security measures, and a notification system ensures that the system operates flawlessly, offering a comprehensive and secure e-commerce solution to our target audience of online shoppers.

Following are standard artifacts, which must be included in this deliverable:

1. Test Plan
2. Test Design Specification
3. Test Case Specification
4. Test Procedure Specification
5. Test Item Transmittal Report
6. Test Log
7. Test Incident Report
8. Test Summary Report

5.2. Test plan

This test plan outlines the testing strategy and activities for the E-Commerce website with an AI-powered recommendation system. It aims to ensure the functionality, performance, and reliability of the platform.

5.2.1. Purpose

The purpose of this test plan is to define the testing approach, tasks, responsibilities, and deliverables for the development and implementation of the E-Commerce website with an AI-powered personalized recommendation system.

5.2.2. Outline

A test plan shall have the following structure:

1. Test plan identifier
2. Introduction
3. Test items
4. Features to be tested
5. Features not to be tested
6. Approach
7. Item pass/fail criteria
8. Suspension criteria and resumption requirements

9. Test deliverables
10. Testing tasks
11. Environmental needs
12. Responsibilities
13. Staffing and training needs
14. Schedule
15. Risks and contingencies
16. Approvals

The sections shall be ordered in the specified sequence. Additional sections may be included immediately prior to Approvals. If some or all the content of a section is in another document, then a reference to that material may be listed in place of the corresponding content. The referenced material must be attached to the test plan or available to users of the plan.

Details on the content of each section are contained in the following sub-clauses.

5.2.2.1. Test plan identifier

Test Plan Identifier: TP-001

5.2.2.2. Introduction

Summarize the software items and software features to be tested. The need for each item and its history may be included. References to the following documents, when they exist, are required in the highest-level test plan:

- a. Project authorization.
- b. Project plan.
- c. Quality assurance plan.
- d. Configuration management plan.
- e. Relevant policies.
- f. Relevant standards.

In multilevel test plans, each lower-level plan must reference the next higher-level plan.

5.2.2.3. Test items

All components and functionalities of the E-Commerce website and the AI recommendation system.

1. User Registration and Authentication
2. User Profile Management
3. Product Catalog Display
4. Search and Filter Functionality

5. Shopping Cart Management
6. Checkout Process
7. AI-powered Recommendation Engine
8. User Feedback and Ratings
9. Order Management
10. Responsive Design
11. Admin Panel for System Management
12. Reporting and Analytics Tools
13. Security Measures
14. Notification System

5.2.2.4. Features to be tested

- User Registration and Authentication
- User Profile Management
- Product Catalog Display
- Shopping Cart Management
- Checkout Process
- AI-powered Recommendation Engine
- User Feedback and Ratings
- Order Management
- Admin Panel for System Management
- Security Measures
- Notification System

5.2.2.5. Features not to be tested

Deprecated Features:

Features: Any functionalities marked as deprecated.

Reason: Deprecated features are no longer actively developed or supported, and testing resources are better utilized on current and relevant features.

Future Enhancements:

Features: Functionalities planned for future releases that are not part of the current scope.

Reason: Testing resources should focus on the current project scope, and future enhancements will undergo testing when they become part of a subsequent release.

Third-party Services:

Features: Integrations with external services or APIs.

Reason: External services are out of the project's control, and their functionality is the responsibility of the service providers. Testing should confirm correct integration rather than the external service's behaviour.

Performance Testing:

Features: Performance-related aspects such as load testing or stress testing.

Reason: Performance testing is often a separate category, and it's excluded here. Specific performance-related testing would be conducted separately to ensure the system can handle expected loads.

Non-functional Requirements:

Features: Non-functional aspects like scalability, reliability, and maintainability.

Reason: While these are crucial for overall system performance, they often require specialized testing approaches and are better addressed separately from functional testing.

Out-of-Scope Functionalities:

Features: Any functionalities explicitly defined as out-of-scope in the project requirements.

Reason: To align with the project's defined scope and avoid unnecessary testing efforts on functionalities not intended for the current release.

Browser/Device Compatibility:

Features: Compatibility with specific browsers or devices not explicitly specified.

Reason: Focusing on specified browsers/devices ensures targeted testing efforts and avoids unnecessary compatibility checks.

Accessibility Features:

Features: Specific accessibility features that are not mandated by project requirements or relevant regulations.

Reason: To align with project requirements and regulations while optimizing testing efforts.

5.2.2.6. Approach**Overall Approach to Testing:**

The overall approach to testing our E-Commerce website and AI recommendation system involves a meticulous strategy to ensure comprehensive validation of each major feature group. This encompasses a combination of manual and automated testing methods, leveraging specialized tools, to guarantee the reliability, security, and optimal functionality of the system.

Major Feature Groups:

User Management and Authentication:

Approach:

Conduct thorough end-to-end testing for user registration and authentication processes.

Utilize automated testing tools, such as Selenium, to simulate user interactions and validate secure authentication.

Techniques:

Perform boundary value analysis for user inputs during registration.

Employ penetration testing using tools like OWASP Zap to identify and address potential security vulnerabilities.

Completion Criteria:

Achieve 100% success rate in user registration and authentication.

Zero security vulnerabilities identified through penetration testing.

Product Catalog and Search Functionality:

Approach:

Implement systematic testing of product catalog display and search/filter functionalities.

Utilize automated regression testing, leveraging tools like JUnit, for continuous validation of catalog updates.

Techniques:

Conduct equivalence partitioning to test various product categories.

Implement load testing using tools like Apache JMeter to ensure search responsiveness under varying loads.

Completion Criteria:

Attain 95% or higher accuracy in product catalog display.

Maintain search response time within acceptable limits under peak loads.

Shopping Cart and Checkout Process:

Approach:

Conduct end-to-end testing of shopping cart management and the checkout process.

Utilize automated testing with tools like TestNG for transactional scenarios and order placement.

Techniques:

Perform boundary testing for maximum and minimum cart quantities.

Use stress testing with tools like Gatling to validate the system's resilience during peak checkout periods.

Completion Criteria:

Achieve 100% accuracy in shopping cart management.

Ensure successful completion of 95% or more simulated checkout transactions.

AI-powered Recommendation Engine:

Approach:

Implement scenario-based testing to evaluate the accuracy and relevance of AI-generated recommendations.

Utilize A/B testing frameworks, like Google Optimize, for comparing the performance of different recommendation algorithms.

Techniques:

Analyse the coverage and diversity of recommendation suggestions based on user scenarios.

Utilize user feedback data to iteratively enhance the recommendation engine.

Completion Criteria:

Attain 90% or higher user satisfaction with AI-generated recommendations.

Demonstrate continuous improvement in recommendation accuracy based on user feedback.

Comprehensiveness and Traceability:**Comprehensiveness Techniques:**

Leverage code coverage tools (e.g., JaCoCo) to ensure all critical paths are exercised.

Implement a requirement traceability matrix to link each test case to the corresponding project requirement.

Additional Completion Criteria:

Maintain the error frequency below 2% for critical functionalities.

Ensure resolution of all high and medium priority issues before release.

Constraints:**Test Item Availability:**

Ensure timely access to the latest build of the E-Commerce website and AI recommendation system.

Establish close collaboration with development teams for the prompt delivery of testable features.

Testing Resource Availability:

Ensure availability of skilled testers for both manual and automated testing.

Optimize testing efforts by leveraging automation for repetitive tasks.

Deadlines:

Adhere to project timelines and milestones aligning with the overall development schedule.

Prioritize testing activities based on project deliverables and deadlines.

Testing Phase	Description	Responsibilities	Criteria for Transition
Unit Testing	Test individual components of the website, such as user registration, product search, and recommendation engine functionalities. Developer conducts unit testing and gets approval from the team lead. Proof of unit testing (test cases, sample output) provided before passing it to the tester.	Developer, Team Lead	All individual components function as designed
Integration Testing	Combine and test functionality between different modules, e.g., user login flow with shopping cart and checkout process. Tester and developer collaborate to perform testing. No specific test tools are required. Proceed to System Testing after fixing critical defects (up to 2 majors allowed).	Tester, Developer	All modules work together seamlessly. Critical defects are fixed.
System Testing	Conduct a comprehensive evaluation of the entire website against project requirements. Tester performs testing to ensure the website functions as a whole and meets user needs.	Tester	All critical and major defects are fixed. The entire website functions as designed.
Acceptance Testing	Real online shoppers, with tester and developer assistance, perform acceptance testing. Users compare the website's functionality and user experience with project goals. Parallel testing with manual processes is conducted for a week after System Testing. Proceed to Acceptance Testing after fixing critical and major defects.	End Users, Tester, Developer	Critical and major defects are corrected and verified by end users. The website meets user expectations and fulfils project goals.
Regression Testing	Regularly perform tests to ensure website functionalities remain unchanged after code modifications. Tester conducts tests to identify any regressions introduced by new code.	Tester	No unexpected changes are observed in website functionality across all platforms (web and mobile if applicable).

5.2.2.7. Item pass/fail criteria

User Registration and Authentication:

Pass Criteria:

User can successfully register with valid information.

Authentication process is secure and allows access to registered users.

User credentials are securely stored in the database.

Fail Criteria:

User registration fails or encounters errors.

Security vulnerabilities in the authentication process.

Product Catalog and Search Functionality:

Pass Criteria:

Product catalog displays accurately with all relevant details.

Search and filter functions provide accurate results.

Load testing shows acceptable response times.

Fail Criteria:

Inaccurate product information in the catalog.

Search and filter functions do not yield correct results.

Unacceptable response times under load.

Shopping Cart and Checkout Process:

Pass Criteria:

Shopping cart management is accurate and responsive.

Checkout process completes successfully.

Transactions are processed securely.

Fail Criteria:

Errors in adding, removing, or updating items in the shopping cart.

Checkout process fails or encounters errors.

Security vulnerabilities in transaction processing.

AI-powered Recommendation Engine:

Pass Criteria:

AI-generated recommendations are accurate and relevant.

Recommendations adapt based on user feedback.

User satisfaction with AI-generated suggestions is 90% or higher.

Fail Criteria:

Inaccurate or irrelevant AI-generated recommendations.

Lack of adaptation to user preferences.

User satisfaction falls below 90%.

System Integration:

Pass Criteria:

Successful integration of individual modules.

Critical defects from integration testing are fixed.

Programs proceed to System/Integration test.

Fail Criteria:

Integration failures between modules.

Critical defects remain unresolved.

Overall System Testing:

Pass Criteria:

Complete, integrated system operates according to specified requirements.

All critical and major defects are corrected.

No unexpected changes in functionality.

Fail Criteria:

System does not meet specified requirements.

Unresolved critical and major defects.

Acceptance Testing:

Pass Criteria:

End users successfully perform acceptance testing.

Parallel testing with existing manual processes is successful.

Correction and verification of critical and major defects.

Fail Criteria:

End users encounter issues during acceptance testing.

Unresolved critical and major defects.

Regression Testing:

Pass Criteria:

Website and application functionalities remain unchanged after code changes.

No unexpected changes in behaviour.

Fail Criteria:

Unexpected changes or disruptions in website or application functionalities.

5.2.2.8. Suspension criteria and resumption requirements

Suspension Criteria and Resumption Requirements:

Suspension Criteria:

Critical Security Vulnerabilities:

Suspension Criteria: If any critical security vulnerabilities are identified during testing, especially in user authentication or transaction processing.

Resumption Requirements: Testing will be suspended until the security vulnerabilities are addressed, and a thorough security audit is conducted. All affected test cases must be re-executed.

System Downtime or Unavailability:

Suspension Criteria: If the system experiences downtime or becomes unavailable during testing, impacting the testing environment.

Resumption Requirements: Testing will be halted until the system is restored to full functionality. Test cases affected by the downtime need to be re-executed.

Major Functional Defects:

Suspension Criteria: Identification of major functional defects that impede the overall testing progress.

Resumption Requirements: Testing will be paused until major defects are addressed and verified. Affected test cases and related scenarios must be re-executed.

Unacceptable Performance under Load:

Suspension Criteria: If the website or AI recommendation system exhibits unacceptable performance under load during load testing.

Resumption Requirements: Testing will be suspended until performance issues are resolved. Load tests will be repeated and affected test cases re-executed.

Resumption Requirements:

Resolution of Suspended Issues:

After suspension due to critical security vulnerabilities, system downtime, major functional defects, or unacceptable performance issues, testing will only resume when these issues are effectively addressed and verified.

Re-execution of Affected Test Cases:

Once the suspended issues are resolved, all test cases that were affected by the suspension must be re-executed to ensure the integrity of the testing process.

Validation of Fixes:

After fixing critical security vulnerabilities, major functional defects, or performance issues, a thorough validation process will be conducted to ensure that the implemented fixes are effective and do not introduce new issues.

Confirmation of System Stability:

Before resuming testing after any suspension, the stability and availability of the entire system, including the website and AI recommendation system, must be confirmed to ensure the testing environment's reliability.

Documentation Update:

All testing documentation, including test plans, test cases, and defect logs, will be updated to reflect the resolution of suspended issues and the re-execution of affected test cases.

5.2.2.9.

a. Test Plan:

- Comprehensive document outlining the testing strategy, objectives, scope, schedule, resources, and responsibilities for ensuring the reliability and functionality of the E-commerce website and AI-powered recommendation system. This plan will focus on functionalities like user registration and authentication, product catalog display, search and filter functionalities, shopping cart management, checkout process, AI recommendation engine, user feedback and ratings, order management, and responsive design.

b. Test Design Specifications:

- Detailed specifications defining the design of test cases for key features, including:
 - User registration and authentication (focusing on different user types and scenarios)
 - Product catalog display (including search, filter, and sorting functionalities)
 - Shopping cart management (adding, removing, and modifying items)
 - Checkout process (payment gateway integration, different payment methods)
 - AI recommendation engine (accuracy, relevance, and personalization based on user behaviour)
 - User feedback and ratings (submission, moderation, and visibility)
 - Order management (order tracking, cancellation, and returns)
 - Responsive design (compatibility across different devices and browsers)

c. Test Case Specifications:

- Individual documents detailing specific steps for each test case, covering various scenarios for each feature, including:
 - Valid and invalid user credentials during registration and login
 - Search for products using keywords, categories, and filters
 - Adding and removing products from the shopping cart
 - Completing the checkout process with different payment methods
 - Verification of AI-generated recommendations based on user preferences and purchase history
 - Submitting and reviewing user feedback and ratings
 - Placing, tracking, cancelling, and returning orders
 - Testing website functionality and layout across various screen sizes and devices

d. Test Procedure Specifications:

- Documentation providing step-by-step procedures for executing tests, including:
 - Setting up the testing environment (database, server configuration)
 - Executing test cases (manual and automated testing)
 - Capturing and documenting results
 - Post-execution activities (defect logging, reporting)
 - Specific instructions for deploying the website and testing on the Heroku Django server, including security measures

e. Test Item Transmittal Reports:

- Reports providing status updates on test items (features and modules), including:
 - Current testing phase (e.g., unit testing, integration testing)
 - Completion status of test cases for each feature
 - Successful deployments to the Heroku environment
 - Identified bugs and defects with severity levels

f. Test Logs:

- Detailed records of testing activities, including:
 - Execution logs for user registration, product catalog, AI recommendation engine, and responsiveness testing
 - Captured results and screenshots for each test case
 - Detailed logs for any errors or unexpected behaviour encountered during testing

g. Test Incident Reports:

- Reports documenting unexpected events, issues, or defects discovered during testing, including:
 - Description of the incident (error message, steps to reproduce)
 - Impact of the incident on website functionality
 - Severity level and priority for resolution
 - Steps taken or recommended for resolving the incident
 - Specific incident reports for security vulnerabilities discovered during testing

h. Test Summary Reports:

- Summarized reports providing an overview of the entire testing process, including:
 - Key metrics (number of test cases executed, number of defects identified, defect resolution rate)
 - Achievements and key findings during testing
 - Summary of identified defects categorized by severity and feature
 - Recommendations for improvement in functionalities, user experience, and AI algorithm
 - User feedback summary regarding the effectiveness and satisfaction with AI-generated recommendations

i. Test Input Data and Test Output Data:

- Documentation specifying test data used for various scenarios, including:
 - Sample user registration details with different profiles and preferences
 - Product catalog entries with variations in categories, descriptions, and prices
 - Expected AI-generated recommendations based on predefined user profiles and purchase history
 - Test data variations for responsive design testing across different devices and browsers

j. Test Tools (Module Drivers and Stubs):

- Documentation on any specific test tools used during testing, including:

- Any custom-developed module drivers or stubs specifically designed for the E-commerce website and AI recommendation system
- Details on any tools used for specific purposes, such as:
 - Load testing tools to simulate high user traffic
 - Security testing tools to identify vulnerabilities
 - AI algorithm validation tools to measure the accuracy and fairness of recommendations

5.2.2.10. Testing tasks

Identify the set of tasks necessary to prepare for and perform testing. Identify all inter task dependencies and any special skills required.

5.2.2.11. Environmental needs

Hardware:

Necessary: Computers with at least 4GB RAM for front-end testing and 6GB RAM for back-end testing, with a quad-core 2GHz+ CPU.

Desired: Dedicated servers with sufficient processing power and storage for simulating high user traffic.

Communications and System Software:

Necessary: Stable internet connection for online testing. Operating system compatibility with Windows 10 for development and testing.

Desired: Virtual private network (VPN) for secure communication during testing.

Mode of Usage:

Necessary: Stand-alone mode for initial testing.

Desired: Integration with an existing network environment to simulate real-world scenarios.

Security:

Necessary: Implementation of firewalls, SSL certificates, and intrusion detection systems for securing the website and recommendation system.

Desired: Regular security audits and compliance with industry standards for data protection.

Special Test Tools:

Necessary: Automated testing tools for regression testing, load testing, and security testing.

Desired: AI algorithm validation tools and monitoring tools for real-time analytics.

Other Software or Supplies:

Necessary: Web servers (e.g., Apache) and database management system (MySQL) for website and data management.

Desired: Analytics tools like Google Analytics for tracking user behaviour and website performance.

Testing Needs:

Necessary: Dedicated office space for the testing team with access to the latest build of the E-Commerce website and AI recommendation system.

Desired: Availability of a staging environment that replicates the production environment for more accurate testing.

Security Level:

Necessary: Ensure a high level of security for proprietary components, including software, data, and hardware. Strict access controls for testing environments.

Desired: Encryption protocols and secure data storage practices.

Source for Unavailable Needs:

Identify external sources for any unavailable needs such as cloud platforms for testing scalability, security consultancy services for vulnerability assessments, and industry-standard compliance documentation for security measures

5.2.2.12. Responsibilities

Identify the groups responsible for managing, designing, preparing, executing, witnessing, checking, and resolving. In addition, identify the groups responsible for providing the test items identified in 7.2.2.3 and the environmental needs identified in 5.2.2.11.

These groups may include the developers, testers, operations staff, user representatives, technical support staff, data administration staff, and quality support staff

Responsibilities

Task	Ali	Adil	Danyal
Write Test Case	✓	×	×
Prepare Test Case	×	✓	×
Execute Test Case	×	×	✓

5.2.2.13 Staffing and training needs

Danyal (Test Manager):

Skills Required:

- Project Management
- Communication and Coordination
- Documentation
- Training Options:
- Project Management Certification Courses
- Communication and Leadership Workshops

Ali (Backend Developer and AI Specialist):

Skills Required:

- Python Development
- AI/ML Frameworks (TensorFlow)
- Integration Testing
- Training Options:
- Python Certification Programs
- AI/ML Training and Workshops

Adil (Frontend Developer):

Skills Required:

- HTML, CSS, JavaScript, Bootstrap
- Responsive Design
- User Interface Integration
- Training Options:
- Frontend Development Bootcamps
- Responsive Design Courses

Staffing and Training Needs Common to All Three Members:

Collaboration and Communication:

Skills Required:

- Effective Collaboration
- Clear Communication
- Teamwork
- Training Options:
- Team Building Workshops
- Communication and Collaboration Training

Problem-Solving:

Skills Required:

- Analytical Thinking
- Problem-Solving
- Decision Making

Training Options:

- Problem-Solving Techniques Courses
- Decision-Making Workshops

Continuous Learning Program:

5.2.2.14. Schedule

Activities	Member	Time Required
Test Planning and Specification	Ali	12 days
Test Report	Adil	6 days
Test Delivery	Danyal	6 days
Perform Testing (All)	All	6 days

5.2.2.15. Risks and contingencies

High-Risk Assumptions and Contingency Plans for the Test Plan:

Assumption: Timely Delivery of AI Models

Contingency Plan: If there are delays in the delivery of AI models for the recommendation system, a pre-trained or placeholder model can be used for testing purposes. Additionally, the team will collaborate closely with the AI development team to expedite the delivery.

Assumption: Availability of Responsive Design Test Devices

Contingency Plan: In case there is a shortage of devices for testing responsive design across various platforms, emulators and virtual machines will be utilized to simulate different devices and screen sizes.

Assumption: User Availability for UAT

Contingency Plan: If there are challenges in getting real users for User Acceptance Testing (UAT), the team will conduct additional internal testing and usability studies. Feedback from team members and stakeholders will be considered to simulate user perspectives.

Assumption: Adequate Test Environment

Contingency Plan: In the event of any issues with the test environment setup, the team will work closely with the IT department to troubleshoot and resolve the problems promptly. Alternative test environments will be considered if needed.

Assumption: Bug Fixes and Patch Implementation

Contingency Plan: In case there are delays in fixing identified bugs or implementing patches, the team will prioritize critical issues and develop a phased release plan. This ensures that essential functionalities are not hindered by non-critical issues.

Assumption: Adherence to Security Standards

Contingency Plan: If there are challenges in meeting security standards, the team will work closely with cybersecurity experts to identify and implement interim security measures. A thorough security review will be conducted before the final release.

Assumption: External Service Availability

Contingency Plan: In the case of external service downtimes affecting testing activities, the team will implement mock services or stubs to simulate the behaviour of external dependencies. This ensures continuous testing even when external services are unavailable.

Assumption: User Engagement Metrics

Contingency Plan: If achieving the targeted user engagement metrics becomes challenging, the team will analyse alternative metrics and adjust the success criteria. Continuous monitoring and analysis will guide adjustments to the project goals.

5.2.2.16 Approvals

Name: Mr. Muhammad Abrar

Title: Fashion Store with AI Recommendation system

Signature: _____

Date: _____

5.4. Test Case Specification

5.4.1. Purpose

This document outlines test cases designed to evaluate the functionality, performance, and user experience of the E-commerce website with an AI-powered recommendation system. The test cases aim to ensure the system operates as intended and meets user expectations.

5.4.2. Outline

A test case specification shall have the following structure:

- a. Test case specification identifier
- b. Test items
- c. Input specifications
- d. Output specifications
- e. Environmental needs
- f. Special procedural requirements
- g. Inter case dependencies

The sections shall be ordered in the specified sequence. Additional sections may be included at the end. If some or all the content of a section is in another document, then a reference to that material may be listed in place of the corresponding content. The referenced material must be attached to the test case specification or available to users of the case specification. Since a test case may be referenced by several test design specifications used by different groups over a long time, enough specific information must be included in the test case specification to permit reuse.

Details on the content of each section are contained in the following sub-clauses.

5.4.2.1. Test case specification identifier

Specify the unique identifier assigned to this test case specification.

5.4.2.2 Test items

Identify and briefly describe the items and features to be exercised by this test case.

For each item, consider supplying references to the following test item documentation:

- a. Requirements specification
- b. Design specification
- c. Users guide
- d. Operations guide
- e. Installation

Module	Test Case ID	Requirement	Pre-Conditions	Test Steps	Expected Result	Environmental Needs
User Registration	TC-UR-01	User shall register for an account.	User is on the registration page.	1. User opens the registration page. 2. User enters valid information in all required fields. 3. User clicks on the "Register" button.	A success message is displayed confirming account registration. User is redirected to the account dashboard.	Hardware: Standard web browsing device. Software: Compatible web browser.
User Registration	TC-UR-02	User shall encounter an error message for missing information.	User is on the registration page.	1. User opens the registration page. 2. User leaves required fields blank and clicks "Register".	An error message is displayed for each missing mandatory field.	Hardware: Standard web browsing device. Software: Compatible web browser.
User Login	TC-UL-01	User shall log in to the system using	User is on the login page.	1. User opens the login page. 2. User enters their	User is successfully logged in and	Hardware: Standard web browsing

		valid credentials.		registered email address and password. 3. User clicks on the "Login" button.	redirected to the account dashboard.	device. Software: Compatible web browser.
User Login	TC-UL-02	User shall encounter an error message for invalid credentials.	User is on the login page.	1. User opens the login page. 2. User enters an incorrect email address or password. 3. User clicks on the "Login" button.	An error message is displayed indicating invalid credentials.	Hardware: Standard web browsing device. Software: Compatible web browser.
Search Functionality	TC-SF-01	Search results should display relevant products based on the search query.	User is on the website.	1. User enters a search query in the search bar. 2. User clicks on the "Search" button.	Search results page displays products matching the search query and relevant variations.	Hardware: Standard web browsing device. Software: Compatible web browser.
Product Details Page	TC-PDP-01	Product details page should display accurate and complete information.	Product page is accessible.	User has access to a product page.	Product page displays product name, description, images, price, specifications, and ratings/reviews.	Hardware: Standard web browsing device. Software: Compatible web browser.
Shopping Cart & Checkout	TC-SCC-01	User shall be able to add and remove items from the shopping cart.	User has logged in and browsed products.	1. User adds a product to the shopping cart. 2. User views the shopping cart. 3. User removes an item from the cart.	The shopping cart reflects the added and removed items, updating the total price.	Hardware: Standard web browsing device. Software: Compatible web browser.
Shopping Cart & Checkout	TC-SCC-02	User shall be able to complete the checkout process successfully.	User has added items to the shopping cart and has a valid payment method.	1. User proceeds to checkout from the cart. 2. User enters shipping and billing information. 3. User selects a payment method and completes the transaction.	User receives confirmation of successful order placement and receives order details.	Hardware: Standard web browsing device. Software: Compatible web browser.
User Profile Management	TC-UPM-01	User shall be able to	User is logged in	1. User edits their profile information	Updated profile information is	Hardware: Standard web

		update their profile information.	and has access to their profile page.	(e.g., name, email address). 2. User saves the changes.	reflected in the user's profile page.	browsing device. Software: Compatible web browser.
--	--	-----------------------------------	---------------------------------------	---	---------------------------------------	---

5.4.2.5.3. Other

NONE.

5.4.2.6. Special procedural requirements

We have no constraint when to perform testing related to our project.

5.4.2.7. Inter case dependencies

Following is the list of identifiers related to test cases that must be executed prior to this test case and summary of the nature of the dependencies:

1. Registration before login.
2. Logout after login.
Registered before Post project.

5.5. Test procedure specification

Section	Description
TPS Identifier	TPS-ECommerce-001
Purpose	Execute test cases for an E-commerce website with an AI recommendation system, evaluating functionalities based on the associated test design specification.
Special Requirements	<ul style="list-style-type: none"> * Prerequisite Procedures: E-commerce website and AI recommendation system setup must be complete. * Special Skills: Testers need a basic understanding of E-commerce and AI. * Environmental Requirements: Production-like testing environment.
Procedure Steps	
Log	Use the standard logging template (Clauses 9 & 10) to record test results, incidents, and other relevant events.
Set Up	<ol style="list-style-type: none"> 1. Ensure E-commerce website deployment and AI recommendation system integration. 2. Verify presence of necessary test data.
Start	<ol style="list-style-type: none"> 1. Open a web browser and navigate to the E-commerce website. 2. Log in with designated test user credentials.
Proceed	<ol style="list-style-type: none"> 1. Execute test cases in the specified sequence. 2. Record observations and results.
Measure	Use designated tools to measure and record relevant metrics (e.g., response time, recommendation accuracy).
Shut Down	Suspend testing, document incidents, and report issues in case of unscheduled events.
Restart	If applicable, identify procedural restart points and describe actions to resume testing from each point.
Stop	After completing all test cases, bring execution to an orderly halt and document test results and observations.

Wrap Up	Restore the testing environment to its initial state and document any changes made during testing.
Contingencies	Describe actions to handle unexpected events or deviations during execution. Provide guidelines for addressing unforeseen issues.

5.6. Test item transmittal report

Test Log Identifier	Description	Activity	Date	Authors
TL1	Conducted on a laptop with Chrome Version (90.0.4430.93) and 4GB RAM.	Website testing	[Date]	Ali Iftikhar, Muhammad Adil, Danyal Yousaf
TL2	Conducted on a desktop with Firefox Version (95.0) and 8GB RAM.	AI-powered recommendation system testing	[Date]	Ali Iftikhar, Muhammad Adil, Danyal Yousaf
TL3	Conducted on a server with MySQL Version (8.0) and 16GB RAM.	Database management system testing	[Date]	Ali Iftikhar, Muhammad Adil, Danyal Yousaf
TL4	Conducted on multiple devices including laptops, tablets, and smartphones.	User interface testing	[Date]	Ali Iftikhar, Muhammad Adil, Danyal Yousaf
TL5	Conducted on a server with simulated high user traffic.	Performance testing	[Date]	Ali Iftikhar, Muhammad Adil, Danyal Yousaf
TL6	Conducted penetration testing to identify vulnerabilities.	Security testing	[Date]	Ali Iftikhar, Muhammad Adil, Danyal Yousaf
TL7	Conducted to ensure seamless integration of different modules.	Integration testing	[Date]	Ali Iftikhar, Muhammad Adil, Danyal Yousaf

5.7. Test log

Incident Details

Date: [Date of the incident]
Time: [Time of the incident]
Location: Test environment

Incident Description	
Description:	During testing of the E-commerce website, it was discovered that users were able to add items to their shopping cart without being logged in. The system did not prompt users to log in or create an account before adding items.
Incident Investigation	
Investigation:	The testing team replicated the scenario and reviewed system logs to identify the cause. It was found that the system did not have proper authentication checks in place for adding items to the shopping cart.
Root Cause	
Root Cause:	The root cause of the issue was identified as a missing authentication check in the shopping cart functionality.
Resolution	
Resolution:	The development team was notified and provided with details of the incident. They implemented the necessary authentication checks in the shopping cart functionality. The fix was tested and verified by the testing team.

Preventative Measures	
Preventative Measures:	To prevent similar issues in the future, the development team will conduct thorough testing to ensure all authentication checks are in place before deploying updates to the production environment.
Incident Closure	
Incident Closure:	The incident was considered resolved and closed after the fix was implemented, tested, and verified. It was documented in the final test report and reported to the project manager.

5.8. Test incident report

Test Report Summary	
Purpose:	The purpose of this test summary report is to provide an overview of the testing phase, focusing on identifying and addressing errors and bugs in the E-commerce website with AI-Powered Personalized Recommendation System.
Outline	
Test Report Identifier:	E-commerce Website Testing Summary
Summary:	The testing phase included thorough testing of various modules and components of the E-commerce website with AI-Powered Personalized Recommendation System. All tested items were found to be functioning well without significant errors or bugs.
Variances:	No significant variances were encountered during the testing phase.
Comprehensive Assessment:	The comprehensive assessment of the testing phase indicates a satisfactory level of system validation, with minimal errors or issues identified.
Summary of Results:	The testing revealed that the Information Module, Admission Module, Faculty Module, Student Panel, Teacher Panel, Admin Panel, Principal Panel, and Reports functionalities are all working as expected.
Evaluation:	The evaluation of the testing phase indicates that the E-commerce website with AI-Powered Personalized Recommendation System is in a stable state, with minimal errors or bugs affecting its functionality.
Summary of Activities:	Various testing activities were conducted, including functional testing, user interface testing, performance testing, and security testing, to ensure the robustness and reliability of the system.
Approvals:	This test summary report is approved by [Name or Role].

5.9. Test summary report

Test Item Transmittal Report	
Project Title:	E-Commerce Website with AI-Powered Personalized Recommendation System
Date:	[Date]
To:	[Recipient Name/Team]
From:	[Your Name/Testing Team]
Subject:	Test Item Transmittal for [Specific Test Item]
Tested Items	
1. Information Module	Thoroughly tested for accurate display and retrieval of product information.
2. Admission Module	Tested functionalities such as user registration, login, and account management.
3. Faculty Module	Evaluated for management of faculty profiles, courses, and related information.

4. Student Panel	Extensive testing conducted for course enrolment, assignment submission, and grade tracking.
5. Teacher Panel	Validated functionalities including lesson planning, assignment creation, and student interaction.
6. Admin Panel	Tested administrative tasks such as user management, content moderation, and system configuration.
7. Principal Panel	Evaluated access to key administrative functions and data analytics tools.
8. Reports	Tested various reports generated by the system for accuracy, completeness, and relevance.