PortMelb Clothing Management System

Abdussalam Ali, Kent institute, 10 barrack street, sydney 2000

Akash yadav (k221798)

Anjal Tharu (k230512)

sarad adhikari (k220103)

sushil rayamajhi (k220203)

2024

**Table of Contents**

**1. Project Description:**

* 1. **Business case and the problem to be addressed/resolved by the project**
  2. **Purpose and objectives**
  3. **Detailed Scope**
  4. **Proposed system design specifications**
  5. **Methodology (to implement the project)**

1. **System analysis and requirements:**
   1. **software and hardware requirements**

**Hardware Requirements:**

**Software Requirements:**

* 1. **Functional and non-functional requirements**
  2. **User requirements (use cases, use case diagram)**

1. **Database design:**
2. **User Interface Design**
3. **Test plan**
4. **Feedback from the lecturer and the responses**
5. **References**
6. **Project Description**
   1. **Business Case and Problems To Be Resolved By the Project:**

Designed to streamline running a retail clothing company, the advanced and automated PortMelb Clothes Shop Management System is Among the jobs our system does are creating corporate reports, ordering processing, inventory control, and client data management. It solves consumer inefficiencies and complaints as well as shop management problems. Through means of optimisation of these factors, this method assists the company to improve general sales performance, operational efficiency, and customer delight.  
  
**Problems to be addressed and Resolved:**  
In the extremely competitive retail market of today, conventional methods of running a clothing company—manual record-keeping, physical inventory control, and paper-based order processing—are inadequate. Usually, the retailer of clothes runs with:  
  
Inefficient Inventory Management:  
Manual inventory control can lead to either too much or too low inventory, therefore influencing either income loss or higher inventory costs.  
Lack of quick stock level access might lead to customer dissatisfaction and delay in order fulfilment.

Order Processing Delays:  
Manual processing consumer orders takes time and is prone to mistakes, therefore delaying order fulfilment and raising the likelihood of customer complaints.  
Monitoring and managing orders can be challenging in the absence of connectivity across sales channels including physical stores and online platforms.

Poor Customer Data Management:  
Good customer data management requires a consistent method to monitor consumer preferences, buying trends, and comments.  
Lack of data-driven insights reduces client retention and undermines efforts at customised marketing.

Inadequate Reporting Analysis  
Fast and accurate sales, inventory, and consumer behaviour data calls for an automated system. Lack of current and accurate company data lowers the ability of strategic planning and the capacity of making well-informed judgements.

Customer satisfaction:  
Customers demand a perfect shopping experience distinguished by fast delivery, simple access to product information, and fast processing of orders.  
The limitations of present technology might lead to a negative customer experience, thereby affecting sales and producing negative reviews.

**Possible Solutions:**  
Comprising significant administrative chores, the advanced and automated Clothing Shop Management System is aimed to tackle these problems. The system will give the following rights:  
  
Inventory Management: Automated inventory level monitoring together with real-time updates and refill reminders facilitates supply management far more simply.  
Order Processing: Good delivery depends on a well-organized system of order processing coordinated with the integration of several sales channels.  
Customer Data Management: Consolidating client data management aids to offer targeted marketing campaigns and better customer relationship management (CRM).  
Reporting and Analytics: Reporting and analytics provide data-driven decision-making when using automated report producing and real-time analytics.  
Improved Customer Satisfaction: An interface that enables consumers review things, make purchases, and check the progress of their orders and is simple for use helps to increase degrees of satisfaction and loyalty.

**Expected Benefits:**  
Applying the Clothing Shop Management System will enable the business to enjoy many advantages. By means of less effort, automation of repetitious processes enabled by enhanced operational efficiency would help to reduce human mistakes, save time, and boost general output. Improved inventory control and order processing technology will speed up order fulfilment and increase product availability, therefore enabling a boost in sales. Customized buying experiences grounded in consumer data let one raise customer happiness and loyalty. The company's capacity to access real-time corporate data helps it to make well-informed strategic decisions, thereby raising its awareness and therefore its competitiveness.

* 1. **Purpose and Objectives:**

**Purpose:**

By means of a complete, integrated platform for inventory management, order processing, customer relationship maintenance, and production of analytical business data, the PortMelb Clothing Shop Management System strives to simplify and improve operation of a clothing retail firm. The system seeks to enhance the shopping experience for the customers and raise the efficiency and effectiveness of retail activities.

**Objectives:**

Get first year's online sales up to a thirty percent: By way of a user-friendly online shopping experience, integrating with well-known payment processors, and streamlining of the product catalogue, the system seeks to greatly increase online sales.

Aim for a 95% customer satisfaction: Features meant to guarantee a higher degree of client happiness include tailored client experiences, quick order processing, and effective customer service.

Launch the system within 6 months of its initiation: Clearly define our project plan with specific benchmarks to guarantee the solution is built, tested, and applied during the allocated period.

Maintain 99.9% System Uptime: Strong and adaptable infrastructure lets one guarantee the excellent availability of the system, so reducing the frequency of downtime and disruption of company activities.

Increase operational efficiency by 20%: Automate boring chores, enhance inventory control, and simplify order processing to lower reliance on manual labour and raise the effectiveness of retail operations.

Improve order fulfillment by 10%: Maximum order processing made possible by good inventory control and automated technologies would boost accuracy and speed and finally aid to raise fulfilment rates.

Expland customer base by 25% by the first year: Properly attract and keep more business using focused campaigns, CRM systems, and data-driven marketing approaches.

Reduce inventory holding cost by 10%: Real-time stock monitoring and just-in-time inventory control systems will help to eliminate extra supply and hence reduce related inventory maintenance expenses.

Enhanced data security and compliance: Make sure the system closely relates to industry-leading standards for data security and completely conforms with pertinent criteria, therefore safeguarding private client information.

Improve customer retention by 20%: Use first-rate customer service, loyalty programmes, and customised marketing campaigns to raise returning business frequency and establish deep client connections.

Automate 80% of routine administrative tasks: Create reports, update inventory, and regulate orders using automation to help to reduce the daily human staff labour required.

* 1. **Scope of the Project:**The Clothing Shop Management System project spans the restrictions, outputs, and capabilities the system will cover.This document defines the aspects that will be incorporated into the system as well as those that will not, therefore guaranteeing clarity for all the persons involved in the project.The project's scope mostly consists of the following major elements.

Customer Management: The system will assist with generally data management, order tracking, login, registration, and profile modifications. By use of their profiles, customers will be able to watch the development of their purchases, check their purchase history, and amend their personal data.

Administrative Management: Administrative users will oversee among other elements of the system product, inventory, customer, supplier, and sales management. The system will have tools for generating, amending, and deleting information on items, suppliers, and consumers.

Product Management: The system will oversee tasks such item addition, modification, removal, and arrangement of product information depending on their kind and firm. Including inventory monitoring will assist to keep accurate records, create alerts for limited availability, and simplify stock level control.

Order and sales management: The system will produce invoices, maintain order statuses, and handle client orders for sales management purposes. The system will handle return processing and maintain sales data including monitoring sales performance over time.

Supplier management: The system will have a module especially in charge of tracking supplier performance, controlling supplier relationships, and managing supplier data. Administrators will be able to add, update, and remove supplier records in addition to create linkages between suppliers and the items they offer.

Reporting and Analytics: Within the range of data the system creates, sales, inventory, customer activity, and supplier performance will all be present. Analytics technologies will provide value on customer preferences, inventory rotation, and sales patterns.

User authentication and authorisation: Authorial permission and user verification: Strong login mechanisms of the system will ensure the security of consumer and administrative users.Control based on roles guarantees users of appropriate privileges depending on their responsibilities.

**Exclusions:**

Physical Store Operations: Actual retail activities outside of system control such staff scheduling, in-store POS system management, and physical security.

Marketing and SEO Tools: The system will lack advanced marketing automation or search engine optimisation (SEO). These tasks are supposed to be done using different systems.

Third Party Integrations: Though these are not part of the project scope initially, integration with outside CRM, ERP, or payment gateway systems may be introduced into further phases.

Customer Support Chatbot: The first scope of the system excludes a chatbot driven by artificial intelligence for customer service. Customer service will consequently be handled using traditional channels like email and phone.

**Key Deliverables:**

Software Application: The main product is the entire and effective Clothing Shop Management System with all necessary modules and capabilities.

User Mannuals: There will be extensive user manuals and training accessible for both administrative users and customers.

Technical Documentations: Technical stuff will be created specifically for developers and IT personnel. This content will address API rules, database schemas, and system design.

Training Material: Among other training resources, tutorials and FAQs will be created to enable customers utilise the system.

Deployment Plan: The deployment strategy will contain a comprehensive review encompassing the staging and production environments, deployment scripts, and reversal policies.

* 1. **Proposed System Design Specifications:**

The proposed design specifications will be explained via different diagrams designed for this project. These diagrams will include, system architecture and network architecture for the PortMelb clothing shop management system, Sequence diagram, use case diagram and activity diagram for the system and each use cases. Below are the required diagrams:

**System Architecture:**

Client Layer: Contains all the user interface for cutomers and admins.

Application Layer: Manages Front end, backend and user authentication.

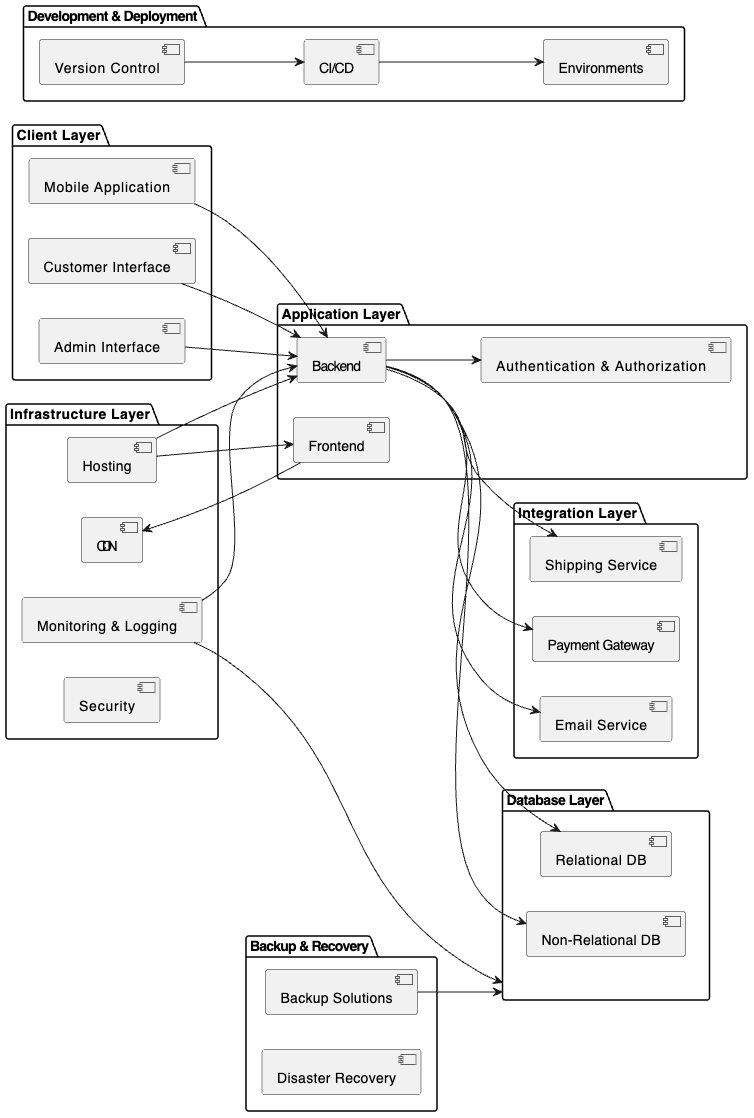
Database layer: Contains all relational and non relational databases.

Integration layer: Handles payment, email and Shipping services.

Infrastructure layer: Covers hosting, CDN, monitoring and security.

Development and deployment: Manages Version Control, CI/CD and environments.

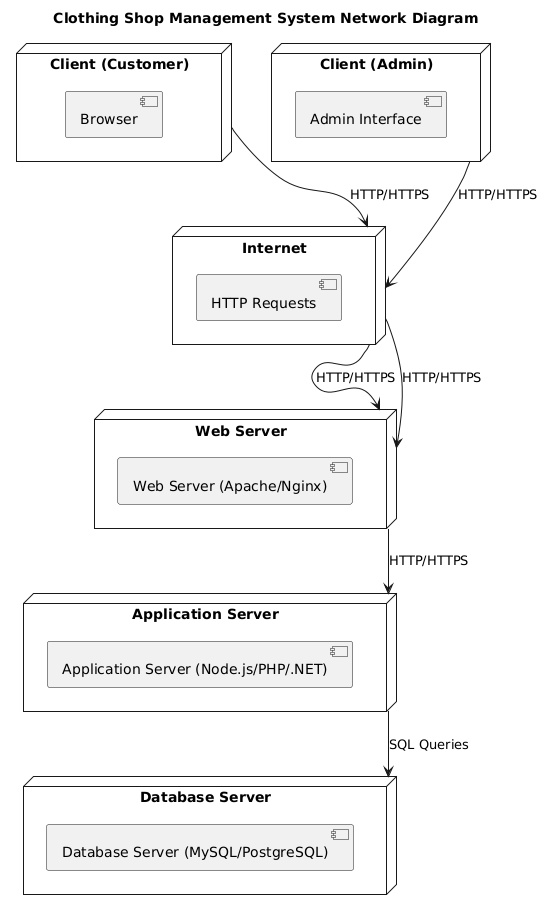
Backup and Recovery: Ensures data backup and recovery strategy.

****

**Fig1: System Architecture for the PortMelb CSMS**

**Network Diagram:**

Currently, the Portmelb Clothing Store is facing challenges due to manual procedures and disconnected systems, which are causing inefficiencies in inventory control, sales monitoring, customer relations building, and general operation coordination. These outcomes include poor departmental communication, issues with overstocking or stockouts, and difficulties in evaluating consumer preferences and sales data. Our Clothing Shop Management System provides a comprehensive platform that streamlines and automates multiple processes, effectively addressing these challenges. By integrating various modules for inventory control, sales monitoring, customer administration, product processing, and reporting, the system aims to enhance the customer experience, boost efficiency, and provide valuable research for making informed decisions. By adopting this strategy, the retailer aims to reduce costs, increase sales, enhance inventory management, and ultimately improve overall business performance and profitability.

  
**Fig2: Network Architecture for PortMelb CSMS**

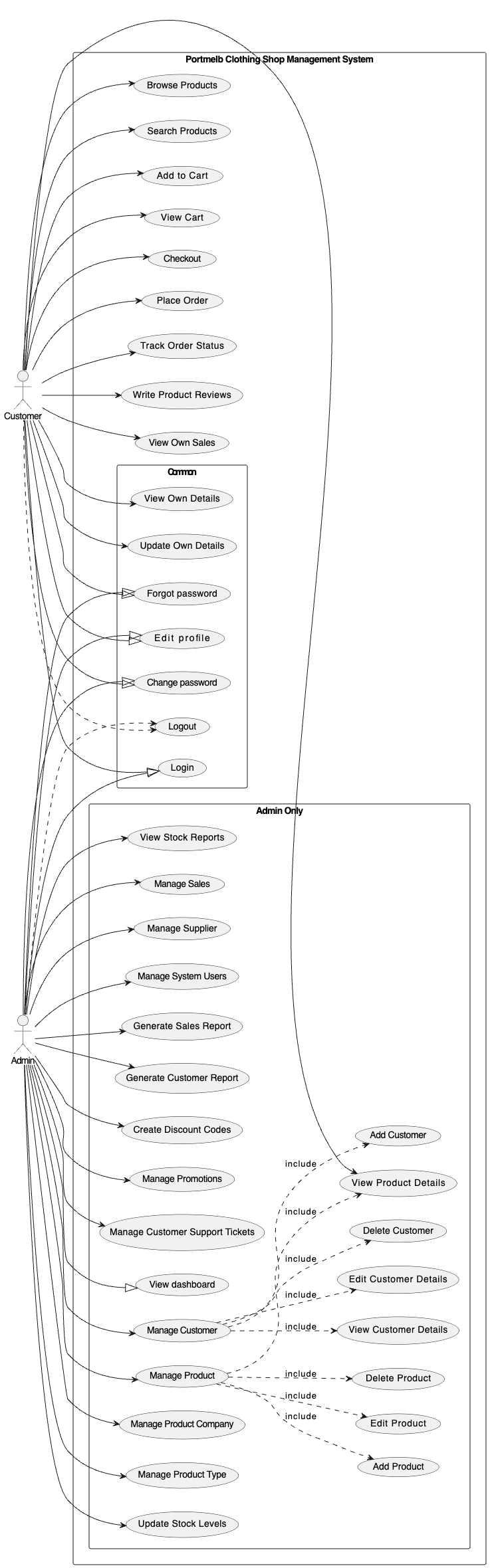
**Sequence Diagram:**

**A screenshot of a diagram

Description automatically generated**

**Fig3: Sequence Diagram for the PortMelb CSMS  
  
  
Use Case Diagram:**

Seeing how two actors—Customer and Admin—within the Portmelb Clothing Shop Management System interact, the use case diagram shows Customers may browse for things, handle their basket and orders, see their personal data and sales record. They also shared with the Admin login, logoff, password control, and profile editing. Apart from these shared obligations, the Admin receives privileged access to a spectrum of administrative tasks including customer administration, product inventory, supplier control, sales, and preparing many reports. The picture also underlines the hierarchical relationship between several use cases, so some are included inside or improve the capabilities of others (e.g., "Add Customer," "View Customer Details," etc., are part of "Manage Customer"). From the perspective of its two primary users, this graph presents a concise overview of the general system capacity. Below is the diagram :



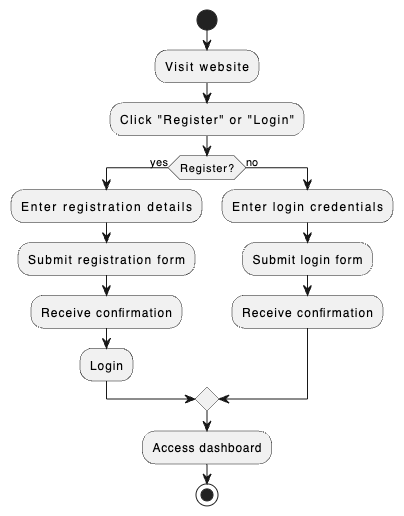
**Fig4: Use Case Diagram for PortMelb CSMS**

**Activity Diagram:**

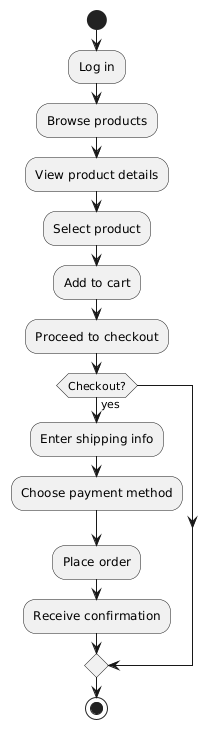
**A diagram of a product

Description automatically generated  
Fig5: Activity diagram for the whole CSMS**

This was the activity diagram for the PortMelb System as a whole now below are the breakdown of the activity diagrams according to each modules or use cases.



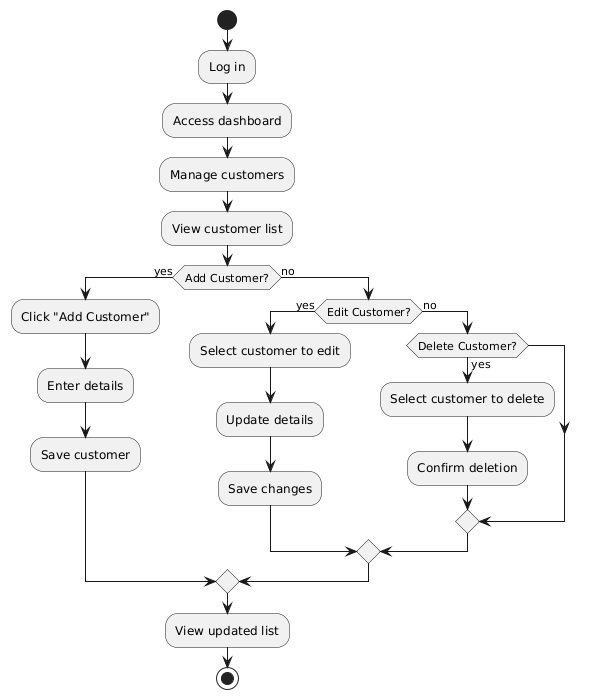
**Fig6:Registration/login activity diagram**

****

**Fig7:Activity diagram for product browsing and purchase**

**A diagram of a customer

Description automatically generatedFig8: activity diagram for managing products**

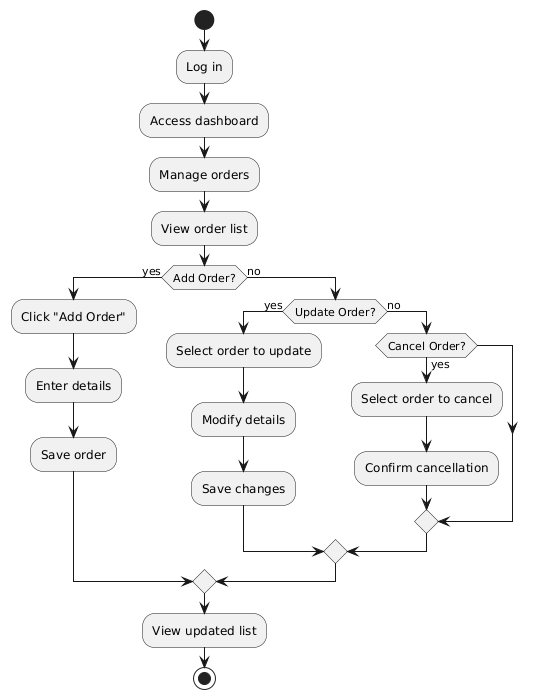
****

**Fig9: Activity diagram for managing customers**

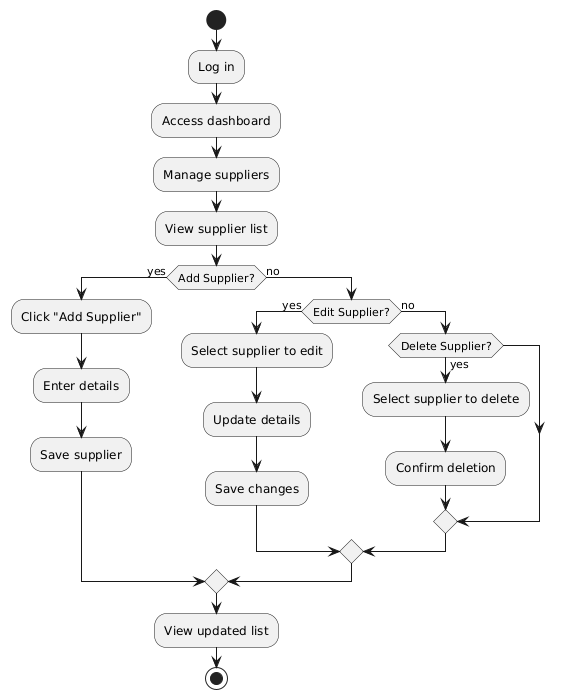
**A diagram of a product report

Description automatically generated**

**Fig10: Activity diagram to generate reports**

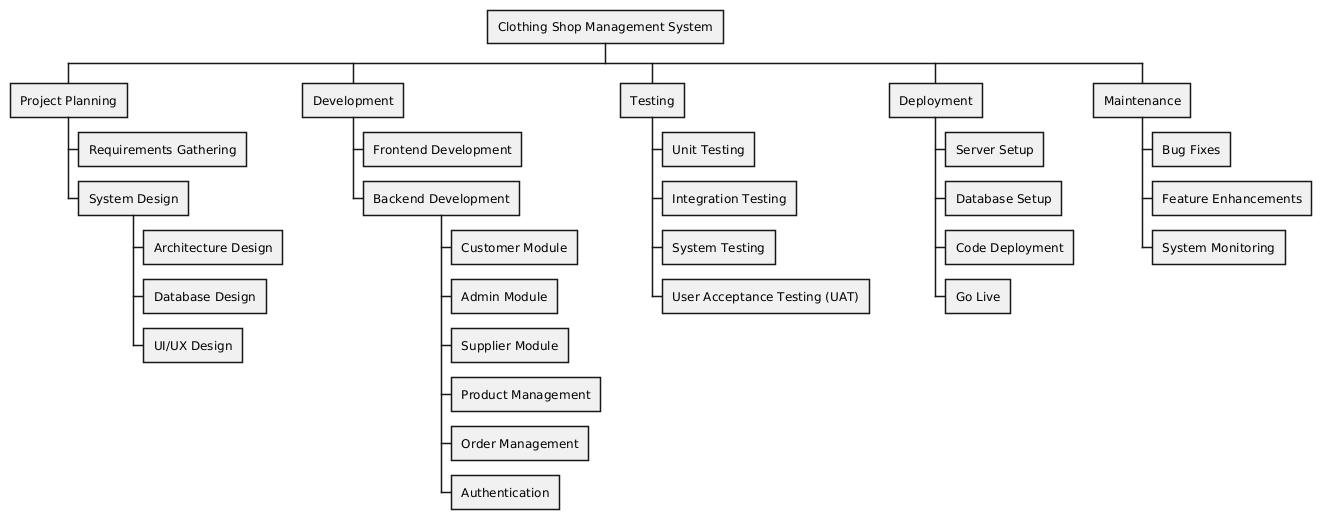
****

**Fig11: Activity diagram to manage order**

****

**Fig12: Activity diagram to manage suppliers**

* 1. **Methodology(to implement the project):**



**Fig13: WBS for the project**

The methodology for the implementation of the project consists of following steps

1. Planning: Planning phase consists of requirement gathering and system design where we design the architecture, database and UX/UI for the system
2. Development: Development phase consists of frontend development and backend development which can be further classified into the development of different modules or use cases such as customer module, admin module etc.
3. Testing: Testing phase consists of 4 kinds of testing which includes unit testing, integration testing, system testing, and user acceptance testing.
4. Deployment: this phase consists of server setup, database setup, code deployment and going live
5. Maintenance: this phase is the last phase for the implementation of the project where after going live, we fix the bugs, enhance features and monitor the system.

Below is the Gantt chart which shows the project timeline and milestones set for completion of different phases.

A screenshot of a project

Description automatically generated

**Fig14: gantt chart for the project**

1. **System analysis and Requirements;**
   1. **Hardware and Software Requirements**

**Hardware requirements**

Hardware demands will depend on aspects such system complexity, data volume, and expected user numbers. The hardware components needed are listed generally here:

Server:  
  
Modern multi-core CPU capable of managing database loadings and application server management.  
Enough RAM will let the operating system, database server, application server, and other running room run as it should. The RAM capacity will rely on the system's expected use and size. Starting with least 8GB, raise as needed.  
Enough space to contain the operating system, application files, database, backups, Solid State Drive (SSD), or Hard Disc Drive (HDD), Faster performance demands SSDs.  
Consider the RAID layout. Data redundancy and improved performance can be achieved with a RAID arrangement, say RAID 1 for mirroring and RAID 5 for striping with parity.

Client:  
  
Desktop or laptop computers: Effective system access and execution of web browsers depends on sufficient memory and processing capabilities.  
Enough screen size and resolution will allow good viewing and system interface working with.  
Input instruments for system interaction are keyboard and mouse.  
Should the shop handle inventory using barcodes, a barcode scanner would be needed.  
Optional for printing invoices and client receipts is receipt printer.

Network's infrastructure:  
  
Reliable router connecting server to the internet and client workstations.  
Switches link many local network devices.  
Device connectivity to the network is made possible via Ethernet cables.  
Option: optional Create an access point should staff or client devices call for wireless connection.

Additional tools:  
  
Scalability: Should the organisation forecast significant expansion, the hardware should be scalable to manage growing traffic and data volumes.  
Consider adding a backup strategy to ensure business continuity should hardware fail and prevent data loss.  
An uninterruptible power supply (UPS) can help halt harm caused by a power outage and data loss from systems failing.

**Software requirements :**

Operating System:

Operating a database server, application server, and web server component of the system calls for either Windows, Linux, or macOS.

Database Management System(DBMS):

Dependable database systems including MySQL, PostSQL, or Microsoft SQL Server are needed for storing and managing data related to consumers, commodities, sales, inventories, and suppliers under a Database Management System (DBMS).

Web Server:

Serving web pages and client requests call for web server software such as Apache or Nyx.

Application Server/Framework:

Development of the backend logic for management of user requests, processing data, and database interaction calls for a robust application server or framework such Node.js, Python (with Django or Flask), PHP (with Laravel or Symfony), or.NET.

Frontend Technology:

The user interface for managers and consumers both will be created using HTML, CSS, and JavaScript frameworks such React, Angular, or Vue.js.

Security Software:

Antivirus software, firewalls, and intrusion detection systems are absolutely vital for safeguarding sensitive data from security risks.

Reporting Tools:

Making sales reports, inventory reports, and other analytical reports could call for libraries or tools for generating reports—such Crystal Reports, Jasper Reports.

Email Server/service:

Using an email server or outside email provider (like SendGrid, Mailchimp), send orders confirmations, notifications, and marketing emails to customers.

Payment Gateway Integrations:

Client online payment processing will call for interaction with a secure payment gateway (like Stripe, PayPal).

Additional tools:

Version control systems like Git enable appropriate working among developers and help to manage code changes.

Development depends on integrated development environments (IDEs) fit for the chosen technology or code editors.

Structures for end-to- end testing and unit testing will help to assure system reliability and quality.

Deployment solutions like Docker, Kubernetes—may be required depending on the complexity and scalability criteria to implement the programme to the production environment.

* 1. **Functional and Non-Functional Requirements**

**Functional Requirements**

Customer Management:

Register new customers

Login and manage customer profile

View and update customer details

Delete customer accounts (admin only)

Product Management:

Add, edit, and delete products (admin only)

View product details

Manage product inventory

Product Type and Company Management:

Add, edit, and delete product types and companies (admin only)

View product types and companies

Sales Management:

Track and manage sales (admin only)

View sales history and details

Generate sales reports

Supplier Management:

Add, edit, and delete supplier information (admin only)

View supplier details

Reporting:

Generate and view reports for customers, products, sales, suppliers, and system users

User Management:

Admin user login

Change password and edit profile

Manage system users (add, edit, delete)

**Non-Functional Requirements**

Performance:

System should handle up to 1000 concurrent users

Response time for user actions should be under 2 seconds

Scalability:

System should be scalable to handle increasing data volume and user load

Security:

Secure authentication and authorization mechanisms

Data encryption (both in transit and at rest)

Regular security updates and patches

Reliability:

System uptime of 99.9%

Regular backups and disaster recovery procedures

Usability:

User-friendly interface with intuitive navigation

Accessible design compliant with accessibility standards

Maintainability:

Well-documented code and system components

Easy-to-understand system architecture and design

* 1. **User Requirements**

End-Users:

Customers: Ability to register, log in, browse products, view and purchase items, and manage their profiles.

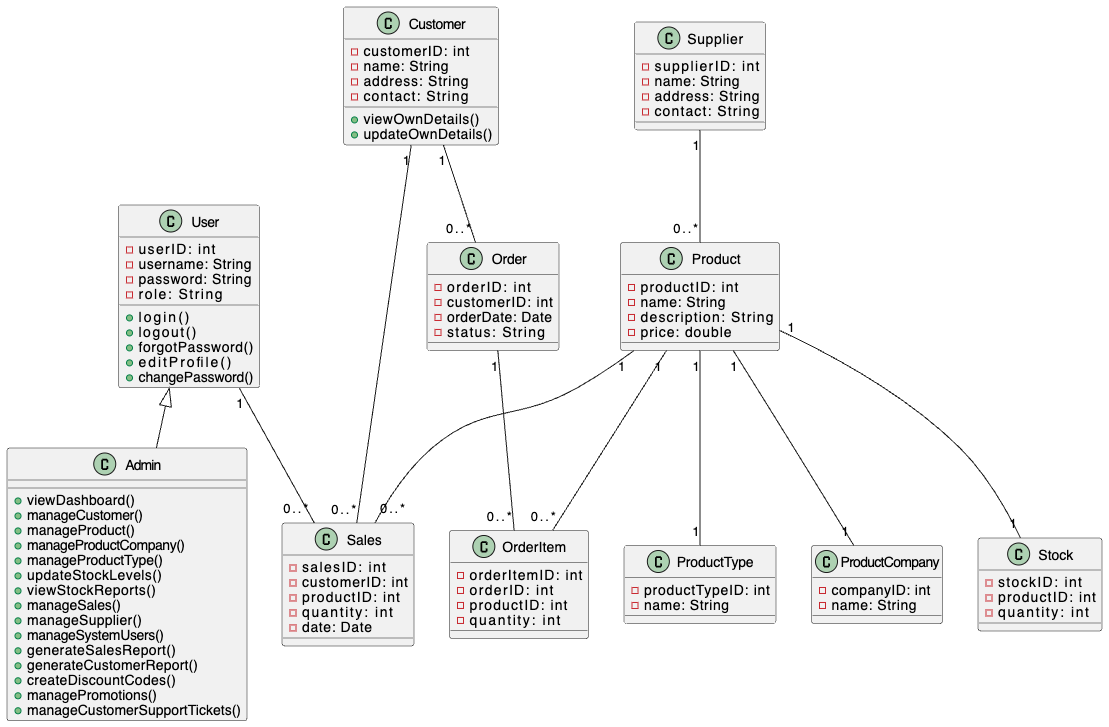
Admins: Ability to manage all aspects of the system, including products, customers, sales, suppliers, and generate reports.

Admin Users:

Roles: System administrators with full access to all system functionalities.

Capabilities: Manage user accounts, oversee system operations, and handle troubleshooting.

1. **Database Design:**



**Fig15:Class diagram for the PortMelb CSMSA diagram of a product

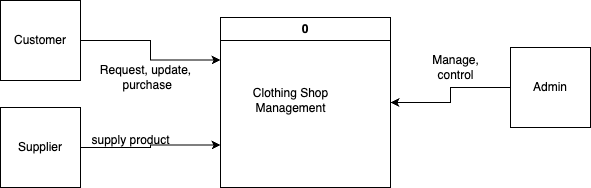
Description automatically generated**

**Fig16: Entity relationship diagram for the PortMelb CSMS**

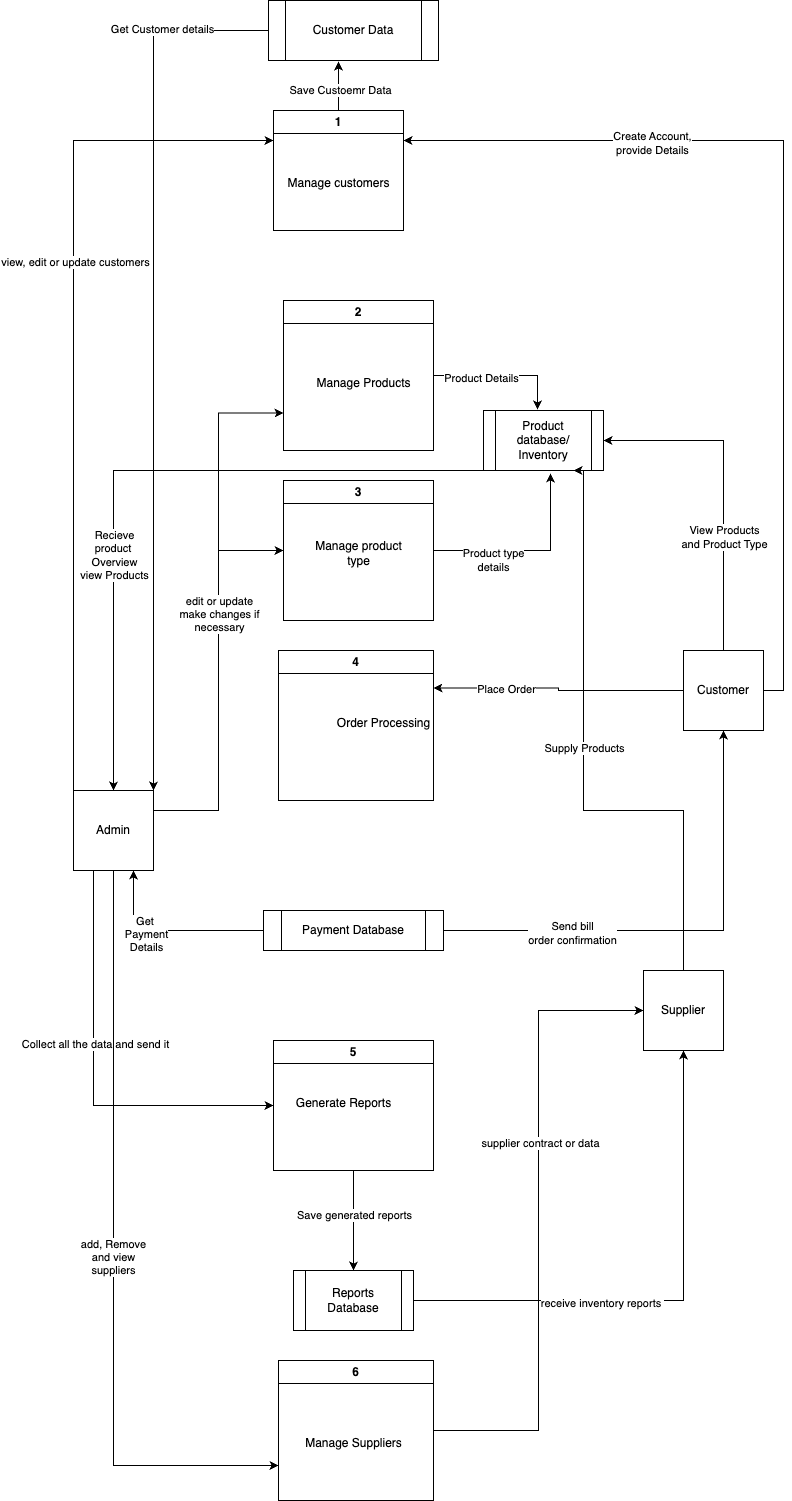
**Logical Model and Normalisation to be Considered:**

The Portmelb Clothing Shop Management System seeks to boost operational efficiency and customer experience of the company by means of automation of important procedures. System logic is driven by consumer and item data collection and structuring; most analysis comes from inventory control and report creation. Handling sales The shop aims to improve running efficiency by means of tailored marketing and simplified order processing, so optimising inventory levels, therefore guiding data-driven decisions, so improving operational efficiency. Perfect and quick platform for corporate management will support increase of sales and profitability.  
  
Precise standardising of the system's database architecture provides data integrity and helps to avoid redundancies. This technique guarantees atomic values, finds main keys in the First Normal Form (1NF), and removes repetitive groups. Furthermore, it ensures in the Second Normal Form (2NF) that every non-key feature is entirely reliant on the primary key, hence addressing partial dependencies. Finally, the Third Normal Form (3NF) removes transitive links whereby non-key characteristics depend on other non-key characteristics. This methodical approach to data management will help the Portmelb Clothing Shop to appropriately handle data and thereby increase general system efficiency by use of a well-designed and effective database.

**Data flow Diagram (DFD):**

****

**Fig17: context level data flow diagram**

**  
fig17: Level 1 Data flow diagram for the system**

**Security Implementations:**

Access Control: Strict access control rules will help to protect the database. Put strict password rules demanding regular changes and complexity. Use least privilege and give users exactly the rights they need. Distribute rights according on user roles using Role-Based Access Control (RBAC), therefore guaranteeing that every user may only access and control data relevant to their particular duties inside the system.

Data Encryption: Encoding plain text or data into a coded form—often known as ciphertext—allows one to stop unwanted access or eavesdropping.Make sure sensitive data is kept safe both now and on route. Strong encryption techniques include AES-256 to protect data saved in the database therefore ensuring its safety even in the situation of physical storage vulnerability. TLS/SSL should be used to avoid illegal interception and hence ensure the security of data transit between application and database servers.

Data Masking: Often known as anonymizing, data masking is the method of concealing or changing private data to preserve data confidence and privacy.For private data security enhancement using anonymizing and masking techniques In context of presenting or processing sensitive data, masking is the practice of substituting non-sensitive characters or values for that data. Anonymizing personally identifying information (PII) from data sets used for analytics or testing permanently alters or eliminates them. This ensures anonymity even in non-production environments.

Input Validation and Sanitization: Verify and sanitise all user input to drastically lower the likelihood of SQL injection attacks and other similar flaws. Combining input validation helps to guarantee that user-entered data follows specified forms and ranges. Eliminate or encrypt any characters that may be used for ill purpose to ensure the security of input data and hence manage the system. To segregate SQL code from user input, employ prepared statements or parametric searches, therefore offering still another defence against SQL injection.

Audit Trails and Logging: Maintaining current comprehensive records and logs will help you to monitor and identify any unlawful access or database security breaches. Every activity inside a database should be completely documented in audit trails encompassing user actions, data modifications, access attempts. Recording database failures, warnings, and other events over the logging process allows the system's general health to be monitored and any future security problems to be detected.

Regular backups: Make a rigors backup plan to make sure data might be rebuilt should unplanned events, inadvertent loss, or data corruption hit. Store backups at a far-off, well-protected region to reduce the likelihood of physical damage or theft and therefore ensure their preservation.

Vulnerability Assessment: Penetration testing and vulnerability assessments are two different ways one examines the security of a system or network. While a vulnerability assessment is the detection and study of such weaknesses in the system, penetration testing actively exploits probable system vulnerabilities to assess system capabilities to withstand assaults. Both methods are quite important in identifying and correcting security weaknesses. Do frequent vulnerability tests and penetration testing to identify and correct any flaws in the database security. These proactive approaches enable one to identify such flaws before hostile individuals utilise them.

Security Awareness Training: Emphasise the need of safeguarding personal information and provide employees comprehensive security policy guidance. Train on social engineering initiatives, phishing attack detection and avoidance of other typical security hazards. The competent and educated workforce of a database environment determines its security.

1. **User Interface Design:**

**User Interface Storyboarding:**

**A screenshot of a login form

Description automatically generated**

**Fig: login Registration page**

**A screenshot of a computer

Description automatically generated**

**Fig: Admin Dashboard**

**A white background with black lines

Description automatically generated**

**Fig: manage customers module**

**A white background with black lines

Description automatically generated**

**Fig: manage products module**

**A screenshot of a website

Description automatically generated**

**Fig: customer dashboard with featured products**

**A white background with black lines

Description automatically generated**

**Fig: placed order section**

**A screenshot of a computer

Description automatically generated**

**Fig: shopping cart**

**A screenshot of a checkout

Description automatically generated**

**Fig: Checkout Module**

**Input Data Forms:**

Customer Registration/Update Form

Fields:

Customer ID (auto-generated or entered by admin)

First Name

Last Name

Email

Phone Number

Address

City

State

Zip Code

Password (for new customers or password reset)

Confirm Password (for new customers or password reset)

Product Entry/Update Form

Fields:

Product ID (auto-generated or entered by admin)

Product Name

Description

Price

Category

Type

Company

Quantity in Stock

Supplier ID

Image Upload

Sales Entry Form

Fields:

Sales ID (auto-generated)

Customer ID (dropdown or search)

Product ID (dropdown or search)

Quantity Sold

Date of Sale

Total Price

Supplier Entry/Update Form

Fields:

Supplier ID (auto-generated or entered by admin)

Supplier Name

Contact Person

Email

Phone Number

Address

City

State

Zip Code

Stock Update Form

Fields:

Product ID (dropdown or search)

Quantity Added/Removed

Reason for Update (e.g., new stock, sale, damage, return)

Date of Update

Output Report Forms for Portmelb Clothing Shop Management System:

**Output Report Forms:**

Sales Report

Filters:

Date Range

Product Category

Product Type

Customer

Data Displayed:

Sales ID

Customer Name

Product Name

Quantity Sold

Date of Sale

Total Price

Profit Margin (calculated)

Summary Statistics:

Total Sales

Total Profit

Average Sale Value

Best Selling Products

Inventory Report

Filters:

Product Category

Product Type

Supplier

Stock Level (e.g., low stock, out of stock)

Data Displayed:

Product ID

Product Name

Category

Type

Company

Quantity in Stock

Reorder Level

Supplier

Customer Report

Filters:

Date Range

Customer Name

City

State

Data Displayed:

Customer ID

Customer Name

Contact Information

Total Purchases

Average Purchase Value

Last Purchase Date

1. **Test Plan**

**Unit Testing**Objective: The main objective of unit testing is to confirm that, evaluated separately, particular system components or modules operate as intended.

Scope: Unit testing will address all fundamental modules—including Customer Management, Product Management, Sales Management, Inventory Management, and Supplier Management. Most importantly, it will also address profile management, login, logoff, password changing.

Approach: Unit testing employing white-box testing will be carried out whereby test cases will be built depending on the fundamental logic and code structure of every module. Appropriate unit testing tools and frameworks will be used; if at all feasible unit tests will be automated to facilitate quick execution and effective regression testing.

**Integration Testing**

Objective: Among the numerous system components and modules, integration testing guarantees flawless data flow and communication.

Scope: Integration testing will largely cover significant interactions like those between Product Management and Inventory Management during stock adjustments, between Customer Management and Sales Management during order putting, and the frontend to backend component communication generally.

Approach: Black-box testing methods will be applied, in which case the creation of test cases will be guided by expected results and system external behaviour. Evaluating interfaces and data flow across modules during the simulation of real-world events and user interactions will take front stage.

**Security Testing**

Objective: Finding and fixing any weaknesses that can affect system security or data integrity is the goal of security testing.

Scope: Among security tests are assessments of the system's resistance to typical risks include SQL injection, cross-site scripting (XSS), and unauthorised access attempts, SQL injection, and Analysis can help one to determine the efficiency of access control methods, input validation, and data encryption. Penetration testing will also help to replicate real attack situations.

Approach: we will run both automated and human security testing approaches. We will use professional security testing tools and frameworks; we may include security specialists to do comprehensive vulnerability assessments and rigours penetration testing.  
User acceptability testing (UAT) is the process by which end users of a software system evaluate and check its functionality, usability, and compatibility to thus ensure that it fits their needs and expectations.

**User Acceptance Testing**

Objective: User acceptability testing (UAT) is the process by which end users of a software system evaluate and check its functionality, usability, and compatibility to thus ensure that it fits their needs and expectations. The system fulfils end users' expectations and needs according to user acceptance testing (UAT).

Scope: UAT will involve those from several user groups—store managers, salespeople, consumers—who will be engaged in the testing process. Real-world situations will let one assess the performance, usability, and usefulness of the system. We shall compile user comments and manage them to guarantee the system's efficiency.

Approach: User requirements will drive created test scenarios and use cases. Users will have access to a test environment so they may go about their usual duties and leave comments. The comments will be compiled and analyzed to identify any areas needing improvement before the system is finally used.

1. **Feedback from the lecturer:** as the feedback, our lectured guided us through the different methodologies of the diagrams and the databases we had created, corrected our mistakes and reviewed our work and we adjusted the changes according to his suggestions immediately.
2. **References:**

Leveson, N.G., Heimdahl, M.P.E., Hildreth, H. and Reese, J.D., 1994. Requirements specification for process-control systems. *IEEE transactions on software engineering*, *20*(9), pp.684-707.

Ceri, S., Pernici, B. and Wiederhold, G., 1987. Distributed database design methodologies. *Proceedings of the IEEE*, *75*(5), pp.533-546.

Kompaniets, V., Lyz, A. and Kazanskaya, A., 2020, October. An empirical study of goal setting in UX/UI-design. In *2020 IEEE 14th International Conference on Application of Information and Communication Technologies (AICT)* (pp. 1-5). IEEE.

Niese, O., 2003. *An integrated approach to testing complex systems* (Doctoral dissertation, Technical University of Dortmund, Germany).

Koç, H., Erdoğan, A.M., Barjakly, Y. and Peker, S., 2021, March. UML diagrams in software engineering research: a systematic literature review. In *Proceedings* (Vol. 74, No. 1, p. 13). MDPI.

Ahmed, S., 2020. How to Write Software Design Documents. [Online] Available at: https://blog.tara.ai/software-design-documents/ [Accessed 16 February 2021].

Dysart, M., 2019. 11 Project Objectives Examples & How to Write Them. [Online] Available at: https://thedigitalprojectmanager.com/project-objectives/ [Accessed 16 February 2021].