

CS5002NI Software Engineering

MAIN-SIT 35% Coursework

AY 2024-2025

Credit: 30

Student Name: Niran Bhatta

London met ID: 23047617

College ID: np01cp4a230046

Assignment Due Date: Monday, 12 May 2025

Assignment Submission Date: Monday, 12 May 2025

Words Count: 3955

I confirm that I understand my coursework needs to be submitted online via MST Classroom under the relevant module page before the deadline in order for my assignment to be accepted and marked. I am fully aware that late submissions will be treated as non-submission and a marks of zero will be awarded.

Contents

| | |
|---|-----------|
| Use Case Description..... | 3 |
| Work Breakdown Structure | 6 |
| Gantt Chart..... | 9 |
| INTRODUCTION | 10 |
| Aims and Objectives | 11 |
| Sequence Diagram..... | 12 |
| Activity Diagram | 13 |
| CONCLUSION | 14 |
| Class Diagram..... | 15 |
| Further Development | 16 |
| Development Section..... | 18 |
| References | 26 |
| | |
| Figure 1 Work Breakdown Structure | 6 |
| Figure 2 Use Case Diagram | 8 |
| Figure 3 Gantt Chart | 9 |
| Figure 4 Sequence diagram for Order Management..... | 12 |
| Figure 5 Activity Diagram of User management | 13 |
| Figure 6 Class Diagram..... | 16 |
| Figure 7 Login Page UI | 19 |
| Figure 8 Register Page UI | 19 |
| Figure 9 Main Dashboard UI | 20 |
| Figure 10 Product Page U..... | 20 |
| Figure 11 Add Product UI..... | 21 |
| Figure 12 Sales Repot Overview UI..... | 21 |
| Figure 13 Sales Report Details UI | 22 |
| Figure 14 Payment Page UI | 22 |
| Figure 15 User Profile UI | 23 |
| Figure 16 Low Stock Products UI | 23 |
| Figure 17 Recent transactions UI..... | 24 |
| Figure 18 Payment Confirmation UI | 24 |
| Figure 19 Sidebar UI | 25 |
| Figure 20 Credentials change UI..... | 25 |
| Figure 21 Header bar for user information UI..... | 26 |

Use Case Description

1) High-Level Use Case Description of Use Management

Use case: System Access

Actors: Admin, Customer

Description:

Users of the system can register and log into the platform through available functions. Users need to register as new members to utilize the system and existing members can perform a login process. Users gain access to system functionalities after authenticating successfully but only receive the features their roles allow them to view.

2) High-Level Use Case Description of Order Management

Use case: Create Purchase Order

Actors: Admin

Description:

System administrators perform stock restocking duties through the process of purchasing orders creation. The system requires administrators to check stock requirements and choose suppliers and create official purchasing documents that enable automated purchasing processes.

3) High-Level Use Case Description of Dispatch Order

Use case: Dispatch Order

Actors: Admin

Description:

The process of order dispatch begins when payment is confirmed through the system by the admin. The administrator follows an order validation process before choosing delivery routes and label generation and performs shipment status monitoring.

4) High-Level Use Case Description of Generate Report

Use case: Generate Report

Actors: Admin, Customer

Description:

Recordkeeping and analysis functions are handled through report generation by both administrative staff and customers. Sales as well as refund reports are accessible to admin personnel while customer users have access to transaction logs and orders history reports.

5) High-Level Use Case Description of Real-Time Stock Update

Use case: Real-Time Stock Update

Actors: Admin

Description:

The system tracks inventory modifications through its automatic monitoring feature. The inventory system sets alert that gives warning when stock quantities drop below defined limits to prevent product shortage.

6) High-Level Case Description of View Product

Use case: View Product

Actors: Customer

Description:

The product listings present detailed information about descriptions as well as prices with availability status and image visualizations. Customer access is available to evaluate products before buying.

7) High-Level Use Case Description of Make Payment

Use case: Make Payment

Actors: Customer

Description:

Customers perform their payments after using the order placement interface. Customers have two payment options either through online payment gateway systems or they can opt for cash-on-delivery. The system accepts transactions while updating order status information.

Expanded Use Case of Order Management

| Actor Actions | System Response |
|--|--|
| Admin logs into the system | System checks login details and grant access |
| Admin selects Create Purchase Order | System displays the list of available items |
| Admin selects supplier and products needed | System adds chosen item to a new purchase order cart |
| Admin reviews and then confirm the purchase order | System saves purchase order and updates the status |
| Admin sends purchase order to distributor/supplier | System proceeds transaction |

Expanded use case for user management

| Actor Actions | System Response |
|--|--|
| User opens login page | System displays login page |
| User enters credentials | System checks credentials and grant access |
| New User selects Register option | System displays registration page |
| User fills up required credentials for registering | System checks and save registration information |
| User is redirected to dashboard according to their roles | System displays the page according to the role(Whether the user is Admin or customer) |

Work Breakdown Structure

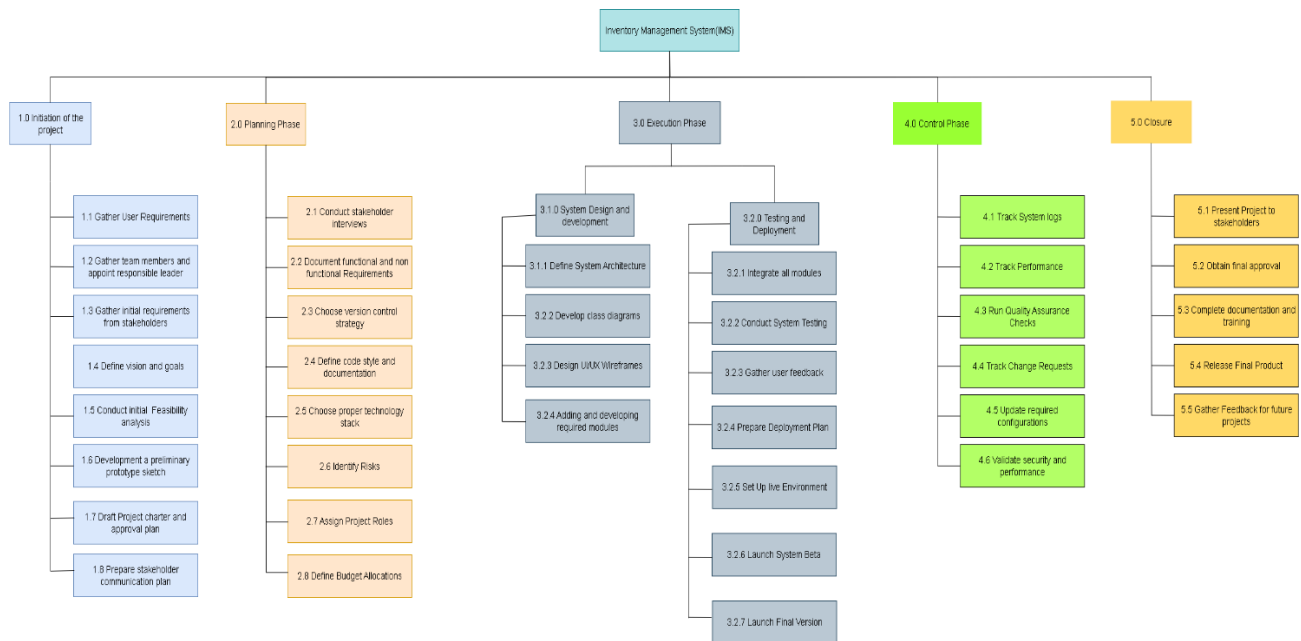


Figure 1 Work Breakdown Structure

The WBS chart records the five chief phases of the project lifecycle involved in implementing an Inventory Management system (IMS). Initiation; Planning, Execution, Control, and Closure. In the Initiation Phase, groundwork for the project is done as project collects user needs, assembles project team, reviews the initial stakeholder feedback, setting the vision and goal, assess the feasibility, construct a rudimentary prototype, complete the project charter and the approval process, and project strategy for keeping stakeholders informed. During the Entries into Planning Phase, emphasis is on the activities of thorough planning, including interviewing stakeholders, defining requirements, defining version control approach, specifying coding norms and documentation criteria, choosing the right technology framework, documenting risks, structuring roles in the project, and assigning financial resources. In the Execution Phase, the major development occur, like system design, architecture definition, UI/UX wireframe development, module creation, all components integration, system testing, collecting user feedback, compose its deployment roadmap, provisioning the live environment, launching a beta release, and eventually, rolling. While execution occurs, the Control Phase, runs in parallel, charting the path of the project through ongoing log reviews, performance monitoring, quality assurance, handling of change requests, configuration management, and security and performance validation. When the Closure Phase ends, stakeholders take a look at the project, final approval is given, all project documents and training materials have been completed, the project has been officially launched, and positive feedback has been collected for continuous improvement. With the help of this hierarchical breakdown, we can understand precisely what tasks and terms are needed in order for the Inventory Management System to be effectively implemented.

Use Case Diagram

Use case diagram identifies the major attributes of an online sales platform, through which users interact with the system. Initially, one can visit products and examine exhaustive information about each available stock. Through the choice of items they want, users trigger the process of creating a purchase order, and thus, officially state their willingness to conclude a transaction. As a result, the action sets off a process to capture corresponding product information and the order request with respect to the user's virtual shopping cart. After the user enters the payment details, he confirms the purchase, which enables the system to process payment and produce a payment report for reference.

Customers can perform a series of functions including item browsing, book searches, book detail views, addition to wish list, adding to the shopping cart, payment etc. The Guest User's capabilities are more constrained with the major focus on browsing items, the finding of books, and the viewing of book details. Each actor invokes such functions as browsing items, searching for books and book details. Customers are able to use the

functionality of the use-cases "Add to Wish List," "Add to Shopping Cart," "Checkout," "Make Payment" and "View Order Details." Through interacting with the "Login/Register" use case, a Guest User enters a registered Customer. In the diagram, use case interactions are graphically shown with arrows given the names "includes" and "extends" in which it goes to show conditional add on of some functionalities to other use cases. To give an example, the "Checkout" use case has "Confirm Order," and "Shipping Details" and for "Make Payment" there are functionalities of "Process Credit Card" and "Process PayPal".

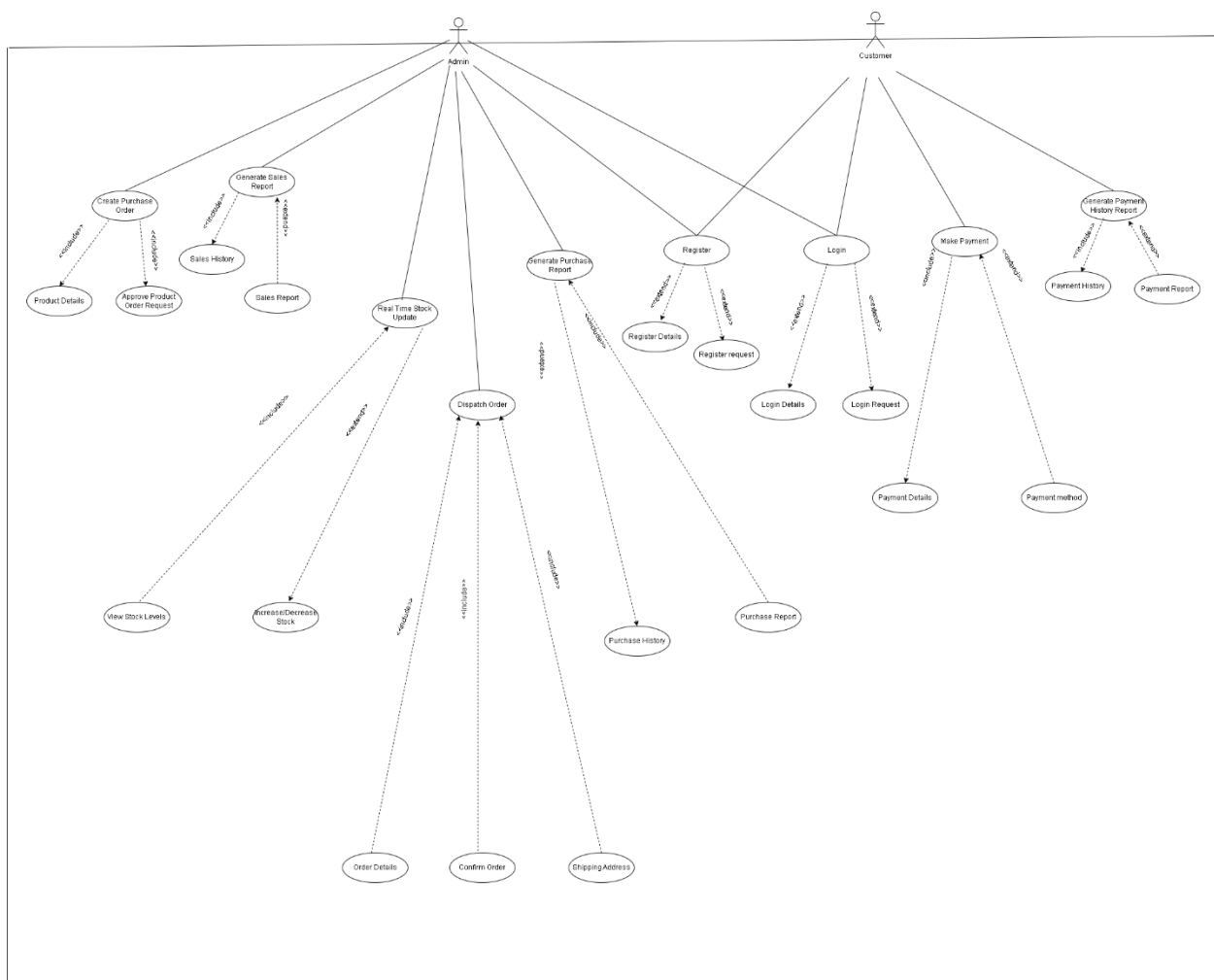


Figure 2 Use Case Diagram

Gantt Chart

A Gantt chart is a graphical structure (a list of tasks separate by dates) used to visually illustrate a task's schedule via horizontal bars. It provides an easy to understand graphical overview of all tasks, their duration, when they start and when they finish, their relationships and key points of progress in the project schedule. Using this, stakeholders and project managers can measure how their progress is going, see to calculations of deadlines, and understand the general schedule of the project with minimal effort.

This Gantt chart is an illustration of the project timeline on creating an Inventory Management System (IMS). The chart groups the project timeline into the key phases Initiating, Planning, Executing, Controlling, and Closing and their respective individual tasks. Tasks are presented as rectangular bars in the chart, their sizes indicating the duration of each task until completion, and the start and end dates are indicated graphically on the chart as well. Critical project milestones are highlighted in the chart too.

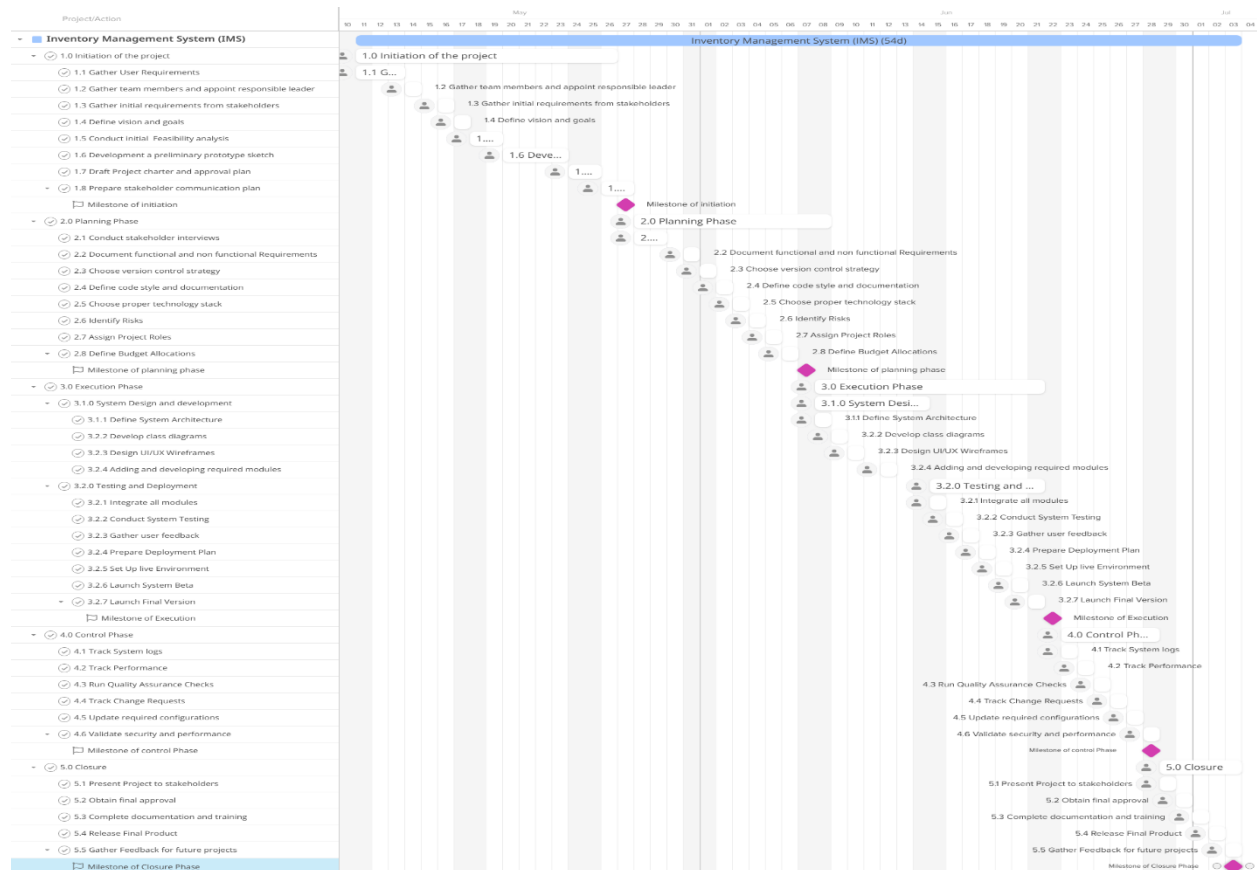


Figure 3 Gantt Chart

INTRODUCTION

Global Tech Corporation initiated a large project for the software development of the warehouse activities of Nepal with a new automated Inventory Management System (IMS). Unfortunately, the system was defective with the initial release because of the inefficiency of system analysis and design, i.e., non-OOAD. Therefore, the system was not effective, the customers were dissatisfied, and eventually the company lost some customers. Now the company has gained experience from mistakes and is focusing on effective system analysis and design for the future work so the future projects will be effective, low-maintenance, and scalable.

The project idea is the reengineering of the IMS on world-class software practices like OOAD. This will make the system modular, scalable, and user-friendly. The system will possess core functionalities like user authentication, purchase and sale handling, product and payment processing, and generation of reports for making intelligent, data-driven business decisions. Global Tech Corporation initiated a large project of building warehouses in Nepal with a revolutionary automated Inventory Management System (IMS). The initial release was not up to the mark because of the defective analysis and design of the system, mainly because the system was not developed on the lines of Object-Oriented Analysis and Design (OOAD). Because of this, inefficiency occurred, the customers were unsatisfied, and the business was lost. Now, the company learned the lesson from the experience and is focusing on the correct analysis and design of the system for future business such that the business is scalable, manageable, and a success. The project aim is re-designing IMS on widely accepted software development principles like OOAD. This will allow for the development of a modular, extendable, and user-friendly system. The functionalities of the new system will be user access control, management of purchase and sale, management of product and payment, and generation of reports for the purpose of making intelligent, data-driven decision. Global Tech Corporation planned a big project of warehouse management development of Nepal by using a new automated IMS. Unfortunately, the initial release was unsatisfactory because the system was analyzed and designed poorly, mainly because Object-Oriented Analysis and Design (OOAD) was not utilized. Therefore, the system was not productive, customers were unhappy, and thus the business opportunity was missed. The organization had already gained a lesson from the experience and is stressing correct analysis and design of the system for future processes so that future processes can be scalable, smooth, and profitable.

The aim of the project is the re-design of the IMS with the application of tested software development principles like OOAD. This will allow for the design of a modular, scalable, and user-friendly system. The system will contain critical functionalities like user permission handling, purchase and selling, product and payment handling, and report generation for making data-driven, intelligent conclusions.

Aims and Objectives

Aims

- To create a strong Inventory Management System (IMS) that rectifies the shortcomings of the older version.
- To will apply Object-Oriented Analysis and Design (OOAD) principles to design the system in a modular, scalable, and maintainable way.
- To enhance warehouse operations through automation of significant processes such as inventory tracking, sales management, and reporting.

Objectives

System Functionality:

- Enforce role-based and user authentication access for Admin and Customer (Buyer) roles.
- Support processing of purchase orders to enable users to view and input purchases and compare prices.
- Offer sales management functionalities like tracking of orders, delivery information, and dispatch

Secure payment processing for buyers.

- Support detailed reporting functionality to study purchases and sales and calculate profit and loss.
- Develop a Work Breakdown Structure (WBS) and Gantt Chart to define the project scope and schedule.
- Develop Use Case Models with descriptions and diagrams to identify the system's requirements.
- Develop Sequence/Collaboration and Activity Diagrams to show the system's interaction and workflows.
- Develop an Analysis Class Diagram to identify domain classes and their relations.

Methodology and Planning

- Select a suitable software development process (e.g., Agile) and explain how it can be applied to the project.
- Define the system design, design patterns, and tools to be employed.
- Create a test plan to determine the system's reliability and a maintenance plan for future upkeep.
- Prototyping: Develop a minimum of 15 UI prototypes to establish major system features and validate design concepts. Through these objectives, the project will deliver an operational, effective, and scalable IMS to meet the needs of Global Tech Corporation and its stakeholders.
-

Sequence Diagram

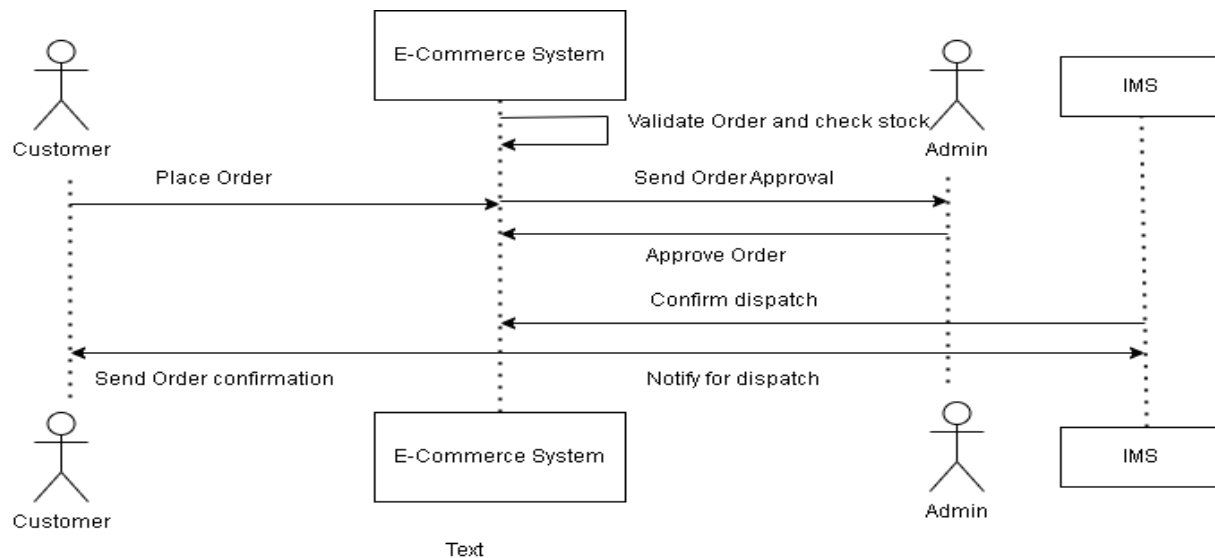


Figure 4 Sequence diagram for Order Management

The sequence diagram shows the communication between an E-Commerce System and an Inventory Management System (IMS) within two concurrent environments. On the two occasions, the E – Commerce System sends a request to the IMS. A response to the E-Commerce System is responded by the IMS once the request is processed. It means that the E-Commerce System needs to input the stock information from IMS and IMS feeds back with the needed data or confirmation. The parallel presentation of these interactions assumes that they could occur independent or simultaneous for different events or products throughout the full e-commerce workflow.

Activity Diagram

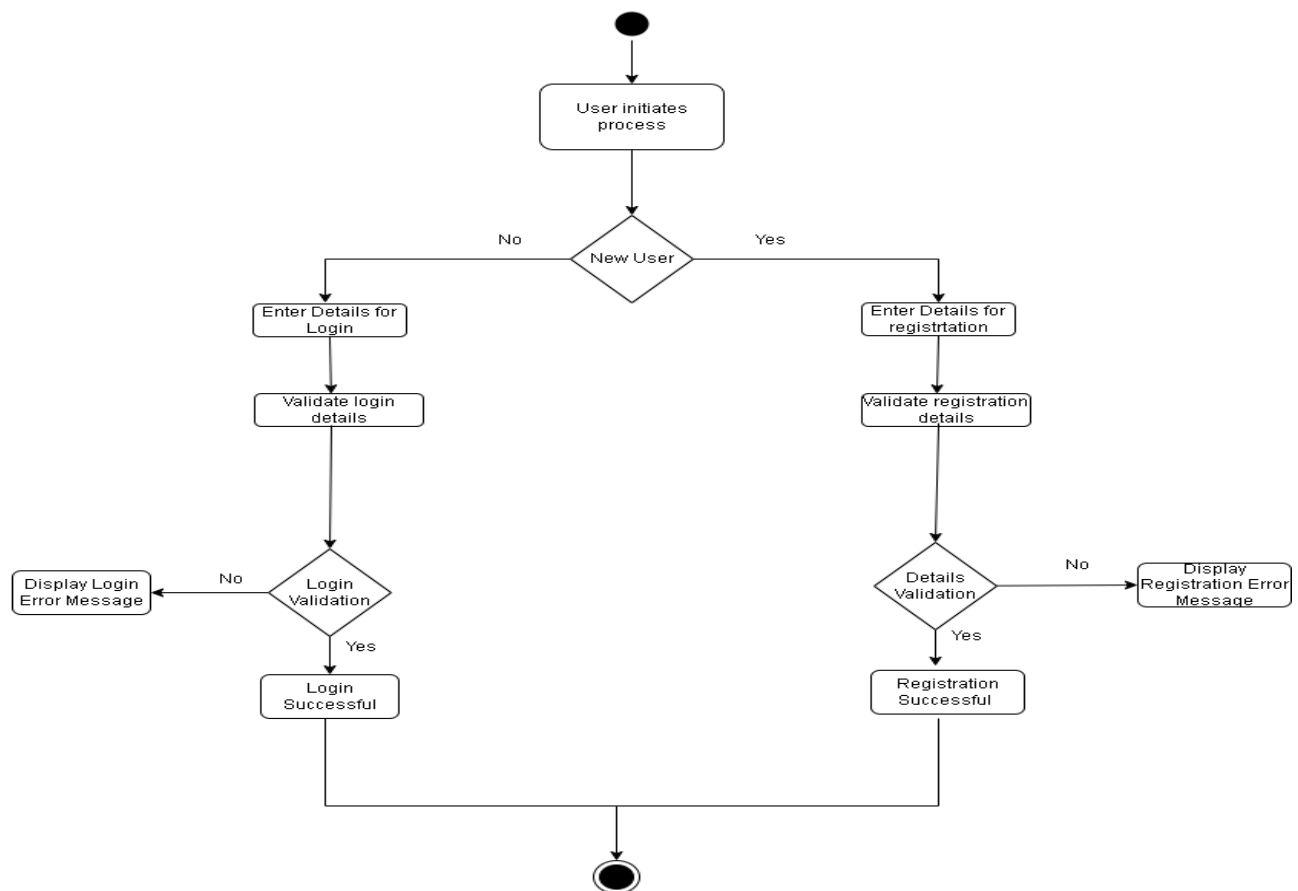


Figure 5 Activity Diagram of User management

The diagram shows here shows how user login and registration are managed in the application. The process starts as soon as a user starts to interact, and the system identifies whether the user is new or an old one to the platform. When the returning users enter the login information, the system checks it. In the event of a failed login validation, the system sends the text for a login error to the user. otherwise, the login is successful. New users are asked to fill forms to submit registration data and these are then verified. Participants are asked for a registration error message if their details fail validation. otherwise, the registration is successful. This diagram provides the obvious differences between the login and registration process including the validation steps and errors that can take place.

CONCLUSION

The main goal of this project was to strengthen the Inventory Management System (IMS) of Global Tech Corporation by implementing better software design methodologies like Object-Oriented Analysis and Design (OOAD). The company had to face many problems in the past because of improper planning and designing of the previous system. It created dissatisfaction among customers, financial loss, and business performance degradation. Learning from such mistakes, focus in this project was on ensuring the new system would be efficient, easier to maintain, and ready for future growth. A few key documents and diagrams were created and reviewed thoroughly in this project. We created a Work Breakdown Structure (WBS) and a Gantt Chart to schedule the project time and activities. We developed use case diagrams and elaborated the use cases at the high level and detailed levels. To understand system processes better, sequence diagrams and activity diagrams were prepared. An analysis class diagram was also prepared in order to illustrate how different parts of the system will communicate.

The project also determined the future design of the system by selecting an appropriate software development method, system architecture, design patterns, test approaches, and maintenance approaches. For a better representation of the system, we developed 15 different user interface (UI) prototypes displaying major features like user login, product management, order input, payment, and reporting. By accomplishing this phase, we now have a firm ground on which to build the new IMS. The new system will simplify it for Global Tech Corporation to control warehouse operations. The new system allows admins and customers to interact with the system efficiently, making buying and selling simpler, track stock in real-time, and generate beneficial reports.

Class Diagram

The class diagram visualizes an Inventory Management System structure that arranges users in addition to products as well as inventory control mechanisms with orders procedures and payments methods and production reporting features. User account functions belong to the User class which saves name-based information with email addresses and passwords while assigning roles to users. The Product class contains all required store items ranging from names to descriptions and pricing information and stock levels in addition to supplier specifications for viewing product details. The Inventory class both performs tracking of stock items and sends warning notifications when inventory quantities adjust. Supply relations use Purchase Order to handle orders and the Sales Order class manages delivery details in addition to customer orders and shipment information. The Payment class functions as a storage platform for payment transaction records which includes financial amounts and payment types joined with date timestamps. The Report class enables the system to generate diverse reports that compile data about sales activities with stock quantity information. The classes demonstrate equal characteristics and operations because of their systematic object-oriented design structure. Below is the table given of the use cases with its corresponding domain class.

| Use Cases | Domain Class |
|--------------------|--|
| Browse Items | Product |
| View Item Details | Product |
| Add to Cart | User, Cart, Product, Cart item |
| Checkout | User, Cart, Order, Payment, Delivery Address |
| Make Payment | User, Order, Payment |
| View Order Details | User, Order, Product |
| Login/Register | User |

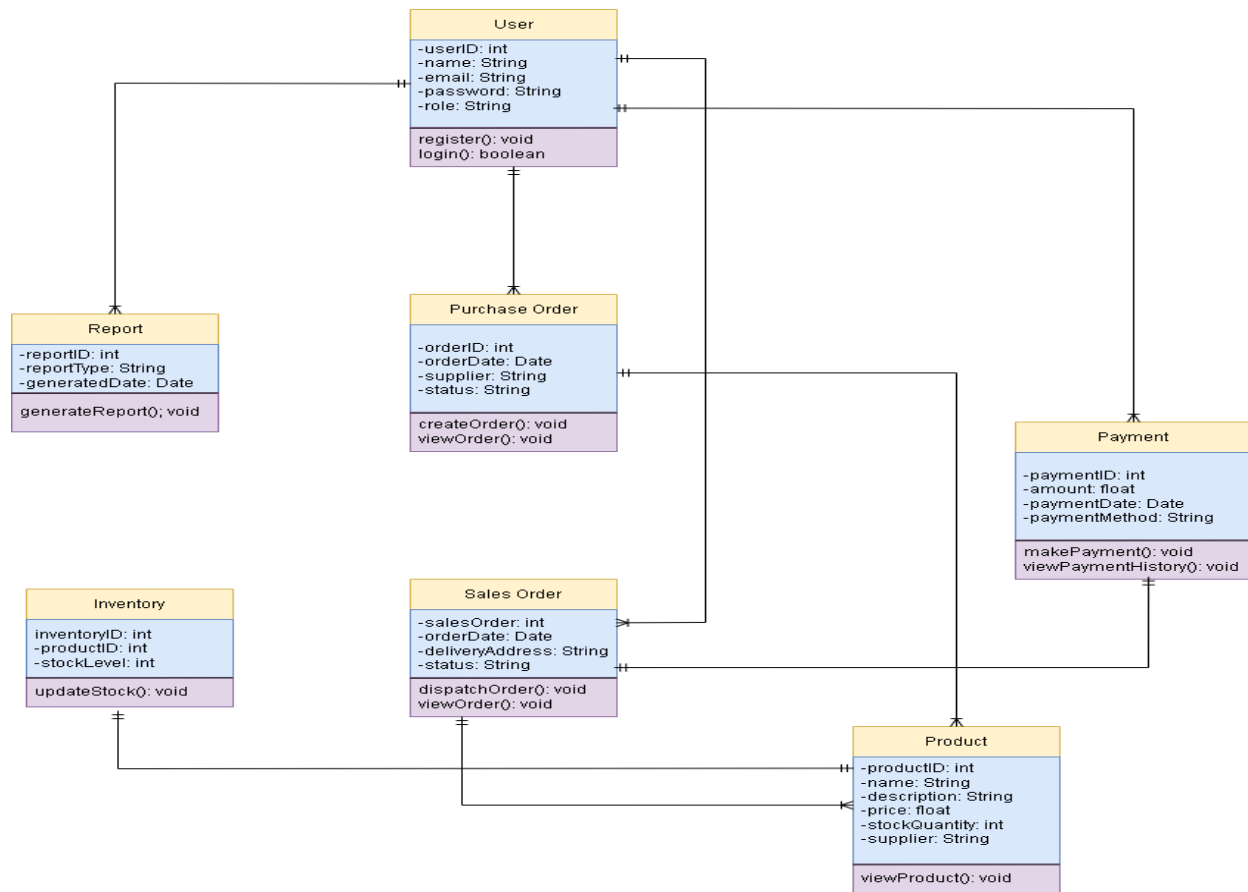


Figure 6 Class Diagram

Further Development

1) Methodology

The choice of the Agile methodology is considered to be the most suitable for the construction of the Inventory Management System (IMS). The Agile approach is structured in small sprints, typically one to two weeks long, this would sustain development in cycles. After completing each sprint, deliverables are transferred to stakeholders to view, forming a continuous feedback arena and an iterative improvement of the system with respect to urgent. Adopting the Agile way, we are able to reduce the development uncertainties, deliver value faster and promote teamwork through daily check ins. The ability to phase development of important features (user authentication, product management, and reporting) of IMS through the Agile approach ensures stakeholders can provide input on a constant basis.

2) Architectural Choice

The IMS is made using Model-View-Controller (MVC) architectural pattern. This methodology divides an application into three inter-relatively interconnected parts: the Model, which managed business processes and data, the View, accountable for presenting information and capturing user interaction and controller, which acted as an interface connecting user inputs with the Model. The MVC architecture allows developers to develop flexible, reusable components that are easy to maintain. Due to its complexity, the IMS can support the MVC architecture in order to allow cooperation on the development of various parts simultaneously. It conducts testing and debugging effectively, and adapt to a modification of the application without a complication.

3) Design Pattern

For the ease of scalability, maintenance and orderly programming style, several object-oriented design patterns will be used for the development of this system. For handling shared resources like database connection, the Singleton pattern is used to ensure that a single active instance is held on. In use of the Factory pattern, the system will also create objects corresponding to varying requirements like roles such as Admin and Customer. Besides that, Observer pattern allows for real-time communication of inventory changes to appropriate system components. In addition, the Strategy pattern will make it easier to integrate a wide range of payment methods, so that the system will be able to process online payments along with cash deliveries. Using the patterns, we enhance the overall reliability and flexibility of the IMS.

4) Development Plan

The IMS development process is divided into phases: Strategic plan that will direct selection of technology and tools and the other one is sequence that has to be followed while introducing features. In the guide's effort to focus on the frontend, our approach will be using HTML5 and CSS3 along with a modern JavaScript framework that is React.JS in order to encourage the creation of flexible user interface. To build the backend framework, Node.js in combination with Express or Python and Django will be used, thus compatible with the skills possessed by the team and the growing needs of the project. When it comes to high performing databases, the decision will be between MySQL or MongoDB. It will then be prioritized initially to integrate basic modules like user registration and login followed by the development of the product and the order management, reporting

and payment processing module will be the last step. Git will be used for version control, while both Figma and Adobe XD will facilitate a collaborative UI prototyping to increase stability and progress.

5) Testing Plan

In order to maintain the stability, reliability and functionality of the system, we will put a testing plan. Unit testing will start by independently validating critical components such as login function, payment processing. Later on, integration testing will show that separate elements work together such as order posting is as smooth as stock updates are made. Given system testing, the entire application will be tested while using functional and nonfunctional specifications and the UAT will involve actual users in making sure that the application fits the expectations of users. In order to ensure stability and avoid possible side effects of new updates always the same testing procedures will be performed.

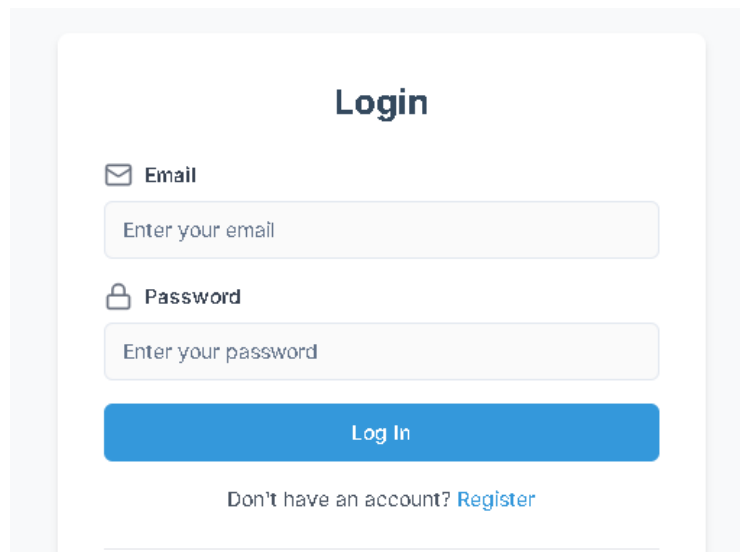
6) Maintenance Plan

In order to maintain its long-term efficiency and stability, IMS has to be maintained constantly after deployment. It includes bug fix updating, securing patch implementation, and system performance review in a bid of catching any possible problems before they build up. A sophisticated system of backup will be put in place to ensure that the data is safe, and including organized backups with a recovery process. To facilitate effective management of issues, users will be able to create tickets for known problems or feature suggestions and have an improved method in identifying, managing issues. Scheduled updates in documentation will demonstrate system changes and new features, providing communication among a development team, administrators and end users. continuous evaluation in basis of regular reviews will ensures that the IMS stays in line with changing business needs.

Development Section

All the wireframes are made using Figma.

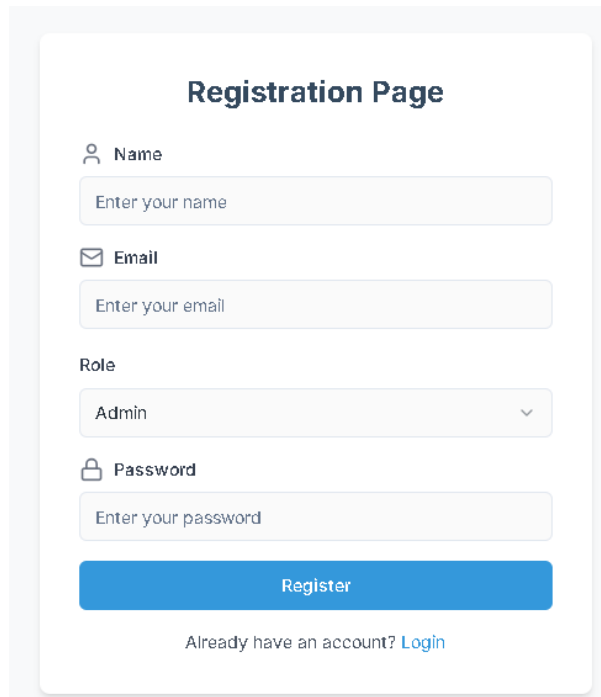
Login Page:



The Login Page UI features a central white card with a light gray border. At the top, the word "Login" is displayed in a bold, dark gray font. Below the title, there are two input fields: the first is labeled "Email" with an envelope icon and contains the placeholder text "Enter your email"; the second is labeled "Password" with a lock icon and contains the placeholder text "Enter your password". A prominent blue button with the text "Log In" is positioned below the password field. At the bottom of the card, a link reads "Don't have an account? [Register](#)".

Figure 7 Login Page UI

Register Page:



The Registration Page UI is presented as a white card with a light gray border. The title "Registration Page" is centered at the top in a bold, dark gray font. The form includes four input fields: "Name" (with a person icon and placeholder "Enter your name"), "Email" (with an envelope icon and placeholder "Enter your email"), "Role" (a dropdown menu currently showing "Admin"), and "Password" (with a lock icon and placeholder "Enter your password"). A blue "Register" button is located below the password field. At the bottom, a link states "Already have an account? [Login](#)".

Figure 8 Register Page UI

Main Dashboard:

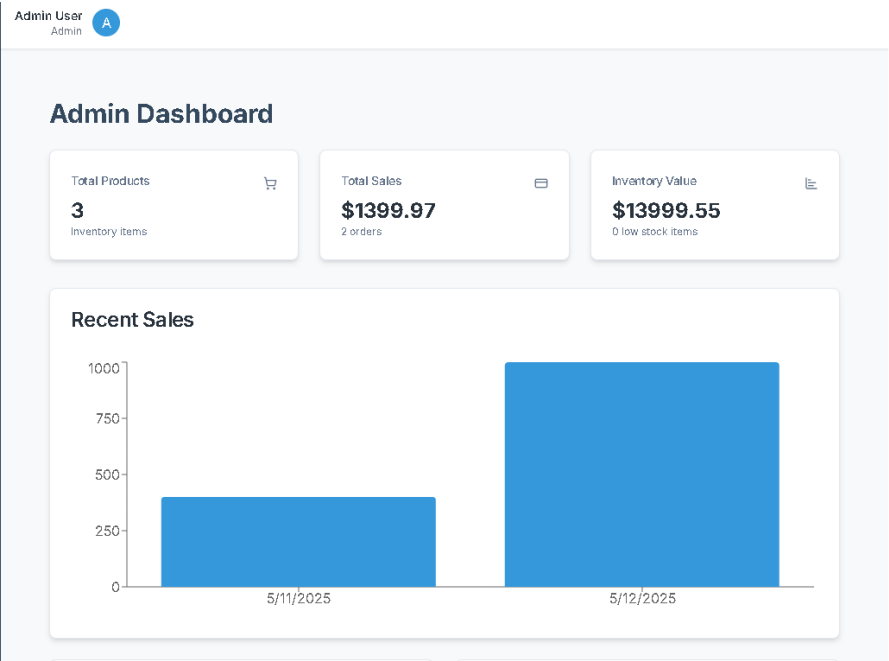


Figure 9 Main Dashboard UI

Products List:

Manage Products

+ Add Product

Search products...

| Product Name | Category | Price | Quantity | Actions |
|--------------|-------------|----------|----------|---------|
| Laptop | Electronics | \$999.99 | 10 ↓ ↑ | Actions |
| Desk Chair | Furniture | \$199.99 | 15 ↓ ↑ | Actions |
| Coffee Maker | Appliances | \$49.99 | 20 ↓ ↑ | Actions |

Figure 10 Product Page U

Add Product:

Add Product

| | |
|---|---|
| Product Name | Category |
| <input type="text" value="Enter product name"/> | <input type="text" value="Enter product category"/> |
| Price | Quantity |
| <input type="text" value="Enter product price"/> | <input type="text" value="Enter product quantity"/> |
| Supplier (Optional) | |
| <input type="text" value="Enter supplier name"/> | |
| <div><button>Cancel</button><button>Submit</button></div> | |

Figure 11 Add Product UI

Sales Report Overview:

Sales Reports

Generate Sales Report

Start Date

End Date

Generate

Total Sales
\$1399.97
2 orders from Apr 12 to May 12

Profit
\$419.99
Estimated at 30% margin

Overview

Details

Figure 12 Sales Repot Overview UI

Sales Report Details:

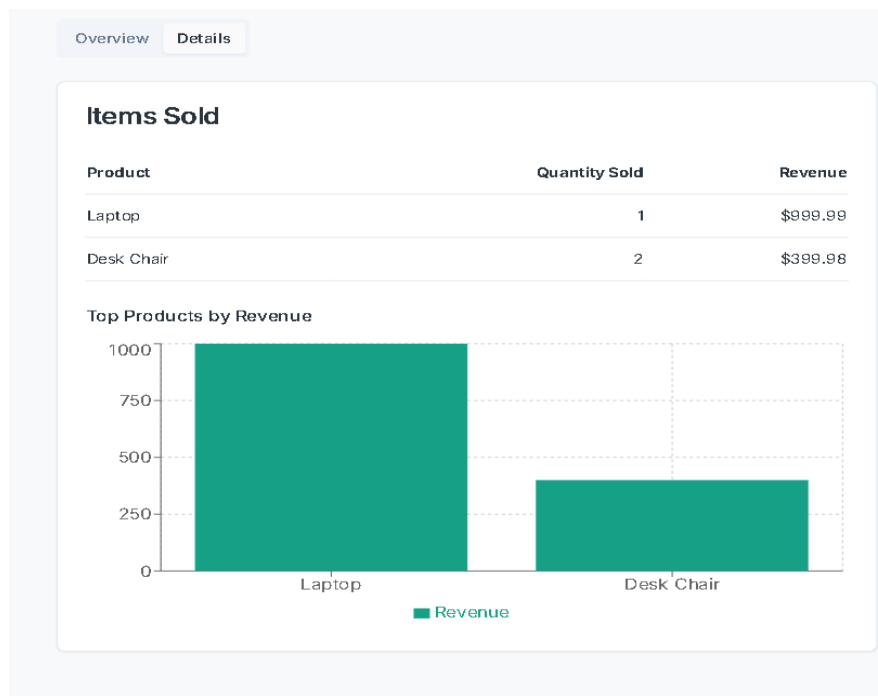


Figure 13 Sales Report Details UI

Payment:

The figure shows a 'Payment Page' interface. At the top, there is a heading 'Payment Page'. Below this, there is a section titled 'Add Products'. This section contains two input fields: 'Select Product' with a dropdown menu showing 'Select a product' and a downward arrow, and 'Quantity' with a text input field containing the number '1'. To the right of these fields is a blue 'Add' button. Below the 'Add Products' section is a section titled 'List of Products'. This section contains a table with four columns: 'PRODUCT', 'PRICE', 'QUANTITY', and 'TOTAL'. The table is currently empty, and a message 'Your cart is empty' is displayed in the center of the table area.

Payment Page

Add Products

Select Product: Select a product

Quantity: 1

Add

List of Products

| PRODUCT | PRICE | QUANTITY | TOTAL |
|--------------------|-------|----------|-------|
| Your cart is empty | | | |

Figure 14 Payment Page UI

User Profile:

Admin User
Admin

A

Customer Profile Page

A

Admin User
Admin
admin@example.com

Account Settings

Profile Password

Name
Admin User

Email
admin@example.com

Role
Admin

Save Changes

Figure 15 User Profile UI

Low Stock Products:

| Low Stock Products | | |
|--------------------|-------|-------|
| Name | Stock | Price |
| No low stock items | | |

Figure 16 Low Stock Products UI

Recent Transactions:

| Recent Transactions | | |
|---------------------|-----------|----------|
| Customer | Date | Amount |
| Niran | 5/12/2025 | \$199.99 |
| Niran | 5/12/2025 | \$49.99 |

Figure 17 Recent transactions UI

Payment Confirmation:

Payment Details

Complete your purchase

Customer Name (Optional)

| | |
|---------------------|----------------|
| Subtotal | \$49.99 |
| Tax | \$5.00 |
| Total Amount | \$54.99 |


 **Pay Now**

Figure 18 Payment Confirmation UI

Navigation Sidebar:

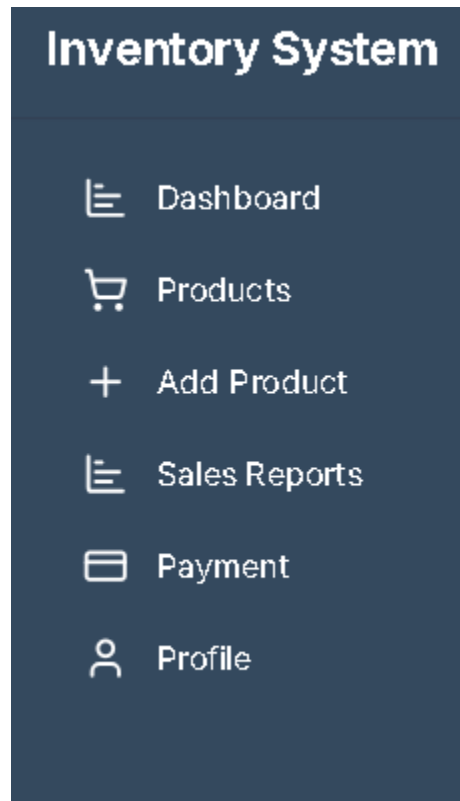


Figure 19 Sidebar UI

Change Password:

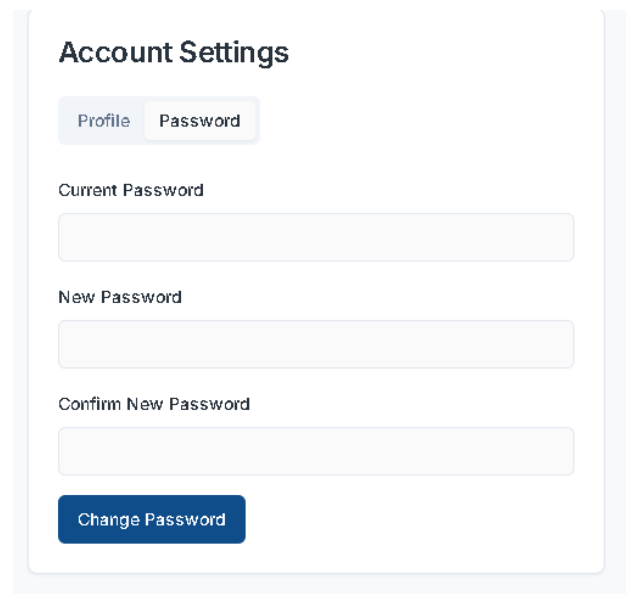
A light gray form titled 'Account Settings'. It has two tabs: 'Profile' and 'Password', with 'Password' being the active tab. The form contains three input fields: 'Current Password', 'New Password', and 'Confirm New Password'. At the bottom of the form is a dark blue button labeled 'Change Password'.

Figure 20 Credentials change UI

Header Bar:



Figure 21 Header bar for user information UI

References

- Aramyan, P. (2024, November 21). Retrieved from Birdview: <https://birdviewpsa.com/blog/the-role-and-value-of-strategic-planning-in-project-management/>
- Gurnov, A. (2024, April 4). Retrieved from Wrike: <https://www.wrike.com/project-management-guide/faq/what-is-agile-methodology-in-project-management/>
- Ramotion. (2023, May 2). Retrieved from Ramotion: <https://www.ramotion.com/blog/mvc-architecture-in-web-application/>
- Team, A. E. (2023, December 4). Retrieved from Adobe for Business: <https://business.adobe.com/blog/basics/how-to-create-work-breakdown-structure-wbs>
- Tsentserensky, S. (2025, March 12). Retrieved from Slickplan: <https://slickplan.com/blog/how-to-make-a-uml-diagram>

(23047617) Niran Bhatta.docx

 Islington College, Nepal

Document Details

Submission ID

trn:oid:::3618:95452106

Submission Date

May 12, 2025, 12:27 PM GMT+5:45

Download Date

May 12, 2025, 12:28 PM GMT+5:45

File Name

(23047617) Niran Bhatta.docx

File Size

24.0 KB

23 Pages





3,620 Words

20,796 Characters




5% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

Match Groups

-  **20 Not Cited or Quoted 5%**
Matches with neither in-text citation nor quotation marks
-  **0 Missing Quotations 0%**
Matches that are still very similar to source material
-  **0 Missing Citation 0%**
Matches that have quotation marks, but no in-text citation
-  **0 Cited and Quoted 0%**
Matches with in-text citation present, but no quotation marks

Top Sources

- 2%  Internet sources
- 0%  Publications
- 5%  Submitted works (Student Papers)

Integrity Flags

0 Integrity Flags for Review

Our system's algorithms look deeply at a document for any inconsistencies that would set it apart from a normal submission. If we notice something strange, we flag it for you to review.

A Flag is not necessarily an indicator of a problem. However, we'd recommend you focus your attention there for further review.

Match Groups

- 20 Not Cited or Quoted 5%**
Matches with neither in-text citation nor quotation marks
- 0 Missing Quotations 0%**
Matches that are still very similar to source material
- 0 Missing Citation 0%**
Matches that have quotation marks, but no in-text citation
- 0 Cited and Quoted 0%**
Matches with in-text citation present, but no quotation marks

Top Sources

- 2% Internet sources
- 0% Publications
- 5% Submitted works (Student Papers)

Top Sources

The sources with the highest number of matches within the submission. Overlapping sources will not be displayed.

| | | |
|---|-----------------|-----|
| 1 | Internet | |
| www.coursehero.com | | 2% |
| 2 | Submitted works | |
| islingtoncollege on 2025-03-21 | | <1% |
| 3 | Submitted works | |
| Universiti Putra Malaysia on 2025-02-27 | | <1% |
| 4 | Submitted works | |
| University of Westminster on 2024-04-01 | | <1% |
| 5 | Submitted works | |
| Laureate Higher Education Group on 2013-07-14 | | <1% |
| 6 | Internet | |
| ebin.pub | | <1% |
| 7 | Submitted works | |
| Eastern Mediterranean University on 2023-06-21 | | <1% |
| 8 | Submitted works | |
| National Institute of Business Management Sri Lanka on 2024-11-05 | | <1% |
| 9 | Submitted works | |
| University of Huddersfield on 2024-05-14 | | <1% |
| 10 | Submitted works | |
| islingtoncollege on 2025-04-06 | | <1% |

11

Submitted works

islingtoncollege on 2025-03-24

<1%

Use Case Description

High-Level Use Case Description of Use Management

Use case: System Access

Actors: Admin, Customer

Description:

Users of the system can register and log into the platform through available functions.

Users need to register as new members to utilize the system and existing members can perform a login process. Users gain access to system functionalities after authenticating successfully but only receive the features their roles allow them to view.

High-Level Use Case Description of Order Management

Use case: Create Purchase Order

Actors: Admin

Description:

System administrators perform stock restocking duties through the process of purchasing orders creation. The system requires administrators to check stock requirements and choose suppliers and create official purchasing documents that

enable automated purchasing processes.

High-Level Use Case Description of Dispatch Order

Use case: Dispatch Order

Actors: Admin

Description:

The process of order dispatch begins when payment is confirmed through the system by the admin. The administrator follows an order validation process before choosing delivery routes and label generation and performs shipment status monitoring.

High-Level Use Case Description of Generate Report

Use case: Generate Report

Actors: Admin, Customer

Description:

Recordkeeping and analysis functions are handled through report generation by both administrative staff and customers. Sales as well as refund reports are accessible to admin personnel while customer users have access to transaction logs and orders history reports.

High-Level Use Case Description of Real-Time Stock Update

Use case: Real-Time Stock Update

2 Actors: Admin

Description:

The system tracks inventory modifications through its automatic monitoring feature.

The inventory system sets alert that gives warning when stock quantities drop below defined limits to prevent product shortage.

1 High-Level Case Description of View Product

Use case: View Product

Actors: Customer

Description:

The product listings present detailed information about descriptions as well as prices with availability status and image visualizations. Customer access is available to evaluate products before buying.

1 High-Level Use Case Description of Make Payment

Use case: Make Payment

Actors: Customer

Description:

Customers perform their payments after using the order placement interface.

Customers have two payment options either through online payment gateway systems or they can opt for cash-on-delivery. The system accepts transactions while updating

order status information.

Expanded Use Case of Order Management

Actor Actions

System Response

Admin logs into the system

System checks login details and grant access

Admin selects Create Purchase Order

System displays the list of available items

Admin selects supplier and products needed

System adds chosen item to a new purchase order cart

Admin reviews and then confirm the purchase order

System saves purchase order and updates the status

Admin sends purchase order to distributor/supplier

System proceeds transaction

Expanded use case for user management

Actor Actions

System Response

User opens login page

System displays login page

User enters credentials

System checks credentials and grant access

New User selects Register option

System displays registration page

User fills up required credentials for registering



System checks and save registration information

User is redirected to dashboard according to their roles

System displays the page according to the role(Whether the user is Admin or customer)

Work Breakdown Structure

Figure 1 Work Breakdown Structure

  The WBS chart records the five chief phases of the project lifecycle involved in implementing an Inventory Management system (IMS). Initiation; Planning, Execution, Control, and Closure. In the Initiation Phase, groundwork for the project is done as project collects user needs, assembles project team, reviews the initial stakeholder feedback, setting the vision and goal, assess the feasibility, construct a rudimentary prototype, complete the project charter and the approval process, and project strategy for keeping stakeholders informed. During the Entries into Planning Phase, emphasis is on the activities of thorough planning, including interviewing stakeholders, defining requirements, defining version control approach, specifying coding norms and documentation criteria, choosing the right technology framework, documenting risks, structuring roles in the project, and assigning financial resources. In the Execution Phase, the major development occur, like system design, architecture definition, UI/UX wireframe development, module creation, all components integration, system testing, collecting user feedback, compose its deployment roadmap, provisioning the live environment, launching a beta release, and eventually, rolling. While execution occurs, the Control Phase, runs in parallel, charting the path of the project through ongoing log reviews, performance monitoring, quality assurance, handling of change requests, configuration management, and security and performance validation. When the Closure Phase ends, stakeholders take a look at the project, final approval is given, all project documents and training materials have been completed, the project has been officially launched, and positive feedback has been collected for continuous improvement. With the help of this hierarchical breakdown, we can understand precisely what tasks and terms are needed in order for the Inventory Management

System to be effectively implemented.

Use Case Diagram

Use case diagram identifies the major attributes of an online sales platform, through which users interact with the system. Initially, one can visit products and examine exhaustive information about each available stock. Through the choice of items they want, users trigger the process of creating a purchase order, and thus, officially state their willingness to conclude a transaction. As a result, the action sets off a process to capture corresponding product information and the order request with respect to the user's virtual shopping cart. After the user enters the payment details, he confirms the purchase, which enables the system to process payment and produce a payment report for reference.

Customers can perform a series of functions including item browsing, book searches, book detail views, addition to wish list, adding to the shopping cart, payment etc. The Guest User's capabilities are more constrained with the major focus on browsing items, the finding of books, and the viewing of book details. Each actor invokes such functions as browsing items, searching for books and book details. Customers are able to use the functionality of the use-cases "Add to Wish List," "Add to Shopping Cart," "Checkout," "Make Payment" and "View Order Details." Through interacting with the "Login/Register" use case, a Guest User enters a registered Customer. In the diagram, use case interactions are graphically shown with arrows given the names "includes"

and "extends" in which it goes to show conditional add on of some functionalities to other use cases. To give an example, the "Checkout" use case has "Confirm Order," and "Shipping Details" and for "Make Payment" there are functionalities of "Process Credit Card" and "Process PayPal".

Figure 2 Use Case Diagram

Gantt Chart

A Gantt chart is a graphical structure (a list of tasks separate by dates) used to visually illustrate a task's schedule via horizontal bars. It provides an easy to understand graphical overview of all tasks, their duration, when they start and when they finish, their relationships and key points of progress in the project schedule. Using this, stakeholders and project managers can measure how their progress is going, see to calculations of deadlines, and understand the general schedule of the project with minimal effort.

This Gantt chart is an illustration of the project timeline on creating an Inventory Management System (IMS). The chart groups the project timeline into the key phases Initiating, Planning, Executing, Controlling, and Closing and their respective individual tasks. Tasks are presented as rectangular bars in the chart, their sizes indicating the duration of each task until completion, and the start and end dates are indicated graphically on the chart as well. Critical project milestones are highlighted in the chart too.

Figure 3 Gantt Chart

INTRODUCTION

Global Tech Corporation initiated a large project for the software development of the warehouse activities of Nepal with a new automated Inventory Management System (IMS). Unfortunately, the system was defective with the initial release because of the inefficiency of system analysis and design, i.e., non-OOAD. Therefore, the system was not effective, the customers were dissatisfied, and eventually the company lost some customers. Now the company has gained experience from mistakes and is focusing on effective system analysis and design for the future work so the future projects will be effective, low-maintenance, and scalable.

The project idea is the reengineering of the IMS on world-class software practices like OOAD. This will make the system modular, scalable, and user-friendly. The system will possess core functionalities like user authentication, purchase and sale handling, product and payment processing, and generation of reports for making intelligent, data-driven business decisions. Global Tech Corporation initiated a large project of building warehouses in Nepal with a revolutionary automated Inventory Management System (IMS). The initial release was not up to the mark because of the defective analysis and design of the system, mainly because the system was not developed on the lines of Object-Oriented Analysis and Design (OOAD). Because of this, inefficiency

occurred, the customers were unsatisfied, and the business was lost. Now, the company learned the lesson from the experience and is focusing on the correct analysis and design of the system for future business such that the business is scalable, manageable, and a success. The project aim is re-designing IMS on widely accepted software development principles like OOAD. This will allow for the development of a modular, extendable, and user-friendly system. The functionalities of the new system will be user access control, management of purchase and sale, management of product and payment, and generation of reports for the purpose of making intelligent, data-driven decision. Global Tech Corporation planned a big project of warehouse management development of Nepal by using a new automated IMS. Unfortunately, the initial release was unsatisfactory because the system was analyzed and designed poorly, mainly because Object-Oriented Analysis and Design (OOAD) was not utilized. Therefore, the system was not productive, customers were unhappy, and thus the business opportunity was missed. The organization had already gained a lesson from the experience and is stressing correct analysis and design of the system for future processes so that future processes can be scalable, smooth, and profitable.

The aim of the project is the re-design of the IMS with the application of tested software development principles like OOAD. This will allow for the design of a modular, scalable, and user-friendly system. The system will contain critical functionalities like user permission handling, purchase and selling, product and payment handling, and report generation for making data-driven, intelligent conclusions.

Aims and Objectives

Aims

To create a strong Inventory Management System (IMS) that rectifies the shortcomings of the older version.

To will apply Object-Oriented Analysis and Design (OOAD) principles to design the system in a modular, scalable, and maintainable way.

To enhance warehouse operations through automation of significant processes such as inventory tracking, sales management, and reporting.

Objectives

System Functionality:


Enforce role-based and user authentication access for Admin and Customer (Buyer) roles.

Support processing of purchase orders to enable users to view and input purchases and compare prices.

Offer sales management functionalities like tracking of orders, delivery information, and dispatch

Secure payment processing for buyers.

Support detailed reporting functionality to study purchases and sales and calculate profit and loss.

 Develop a Work Breakdown Structure (WBS) and Gantt Chart to define the project scope and schedule.

Develop Use Case Models with descriptions and diagrams to identify the system's requirements.

Develop Sequence/Collaboration and Activity Diagrams to show the system's interaction and workflows.

Develop an Analysis Class Diagram to identify domain classes and their relations.

Methodology and Planning

Select a suitable software development process (e.g., Agile) and explain how it can be applied to the project.

Define the system design, design patterns, and tools to be employed.

Create a test plan to determine the system's reliability and a maintenance plan for future upkeep.

Prototyping: Develop a minimum of 15 UI prototypes to establish major system features and validate design concepts. Through these objectives, the project will deliver an operational, effective, and scalable IMS to meet the needs of Global Tech Corporation and its stakeholders.

Sequence Diagram

Figure 4 Sequence diagram for Order Management

The sequence diagram shows the communication between an E-Commerce System and an Inventory Management System (IMS) within two concurrent environments. On the two occasions, the E – Commerce System sends a request to the IMS. A response to the E-Commerce System is responded by the IMS once the request is processed. It means that the E-Commerce System needs to input the stock information from IMS and IMS feeds back with the needed data or confirmation. The parallel presentation of these interactions assumes that they could occur independent or simultaneous for different events or products throughout the full e-commerce workflow.

Activity Diagram

Figure 5 Activity Diagram of User management

The diagram shows here shows how user login and registration are managed in the application. The process starts as soon as a user starts to interact, and the system identifies whether the user is new or an old one to the platform. When the returning users enter the login information, the system checks it. In the event of a failed login validation, the system sends the text for a login error to the user. otherwise, the login is successful. New users are asked to fill forms to submit registration data and these are then verified. Participants are asked for a registration error message if their details fail validation. otherwise, the registration is successful. This diagram provides the obvious differences between the login and registration process including the validation steps and errors that can take place.

CONCLUSION

11 The main goal of this project was to strengthen the Inventory Management System (IMS) of Global Tech Corporation by implementing better software design methodologies like Object-Oriented Analysis and Design (OOAD). The company had to face many problems in the past because of improper planning and designing of the previous system. It created dissatisfaction among customers, financial loss, and business performance degradation. Learning from such mistakes, focus in this project was on ensuring the new system would be efficient, easier to maintain, and ready for future growth. A few key documents and diagrams were created and reviewed thoroughly in this project. We created a Work Breakdown Structure (WBS) and a Gantt Chart to schedule the project time and activities. We developed use case diagrams

and elaborated the use cases at the high level and detailed levels. To understand system processes better, sequence diagrams and activity diagrams were prepared. An analysis class diagram was also prepared in order to illustrate how different parts of the system will communicate.

The project also determined the future design of the system by selecting an appropriate software development method, system architecture, design patterns, test approaches, and maintenance approaches. For a better representation of the system, we developed 15 different user interface (UI) prototypes displaying major features like user login, product management, order input, payment, and reporting. By accomplishing this phase, we now have a firm ground on which to build the new IMS. The new system will simplify it for Global Tech Corporation to control warehouse operations. The new system allows admins and customers to interact with the system efficiently, making buying and selling simpler, track stock in real-time, and generate beneficial reports.

Class Diagram

The class diagram visualizes an Inventory Management System structure that arranges users in addition to products as well as inventory control mechanisms with orders procedures and payments methods and production reporting features. User account functions belong to the User class which saves name-based information with email addresses and passwords while assigning roles to users. The Product class contains all required store items ranging from names to descriptions and pricing information and stock levels in addition to supplier specifications for viewing product

details. The Inventory class both performs tracking of stock items and sends warning notifications when inventory quantities adjust. Supply relations use Purchase Order to handle orders and the Sales Order class manages delivery details in addition to customer orders and shipment information. The Payment class functions as a storage platform for payment transaction records which includes financial amounts and payment types joined with date timestamps. The Report class enables the system to generate diverse reports that compile data about sales activities with stock quantity information. The classes demonstrate equal characteristics and operations because of their systematic object-oriented design structure. Below is the table given of the use cases with its corresponding domain class.

Use Cases

Domain Class

Browse Items

Product

View Item Details

Product

Add to Cart

User, Cart, Product, Cart item

Checkout

User, Cart, Order, Payment, Delivery Address

Make Payment

User, Order, Payment

View Order Details

User, Order, Product


Login/Register

User

Figure 6 Class Diagram

Further Development

Methodology

 The choice of the Agile methodology is considered to be the most suitable for the construction of the Inventory Management System (IMS). The Agile approach is structured in small sprints, typically one to two weeks long, this would sustain development in cycles. After completing each sprint, deliverables are transferred to stakeholders to view, forming a continuous feedback arena and an iterative improvement of the system with respect to urgent. Adopting the Agile way, we are able to reduce the development uncertainties, deliver value faster and promote teamwork through daily check ins . The ability to phase development of important features (user authentication, product management, and reporting) of IMS through the Agile approach ensures stakeholders can provide input on a constant basis.

Architectural Choice

The IMS is made using Model-View-Controller (MVC) architectural pattern. This methodology divides an application into three inter-relatively interconnected parts: the Model, which managed business processes and data, the View, accountable for presenting information and capturing user interaction and controller, which acted as an interface connecting user inputs with the Model. The MVC architecture allows developers to develop flexible, reusable components that are easy to maintain. Due to its complexity, the IMS can support the MVC architecture in order to allow cooperation on the development of various parts simultaneously. It conducts testing and debugging effectively, and adapt to a modification of the application without a complication.

Design Pattern

For the ease of scalability, maintenance and orderly programming style, several object-oriented design patterns will be used for the development of this system. For handling shared resources like database connection, the Singleton pattern is used to ensure that a single active instance is held on. In use of the Factory pattern, the system will also create objects corresponding to varying requirements like roles such as Admin and Customer. Besides that, Observer pattern allows for real-time communication of inventory changes to appropriate system components. In addition, the Strategy pattern will make it easier to integrate a wide range of payment methods, so that the system will be able to process online payments along with cash deliveries. Using the patterns, we enhance the overall reliability and flexibility of the IMS.

Development Plan

The IMS development process is divided into phases: Strategic plan that will direct selection of technology and tools and the other one is sequence that has to be followed while introducing features. In the guide's effort to focus on the frontend, our approach will be using HTML5 and CSS3 along with a modern JavaScript framework that is React.JS in order to encourage the creation of flexible user interface. To build the backend framework, Node.js in combination with Express or Python and Django will be used, thus compatible with the skills possessed by the team and the growing needs of the project. When it comes to high performing databases, the decision will be between MySQL or MongoDB. It will then be prioritized initially to integrate basic modules like user registration and login followed by the development of the product and the order management, reporting and payment processing module will be the last step. Git will be used for version control, while both Figma and Adobe XD will facilitate a collaborative UI prototyping to increase stability and progress.

Testing Plan

In order to maintain the stability, reliability and functionality of the system, we will put a testing plan. Unit testing will start by independently validating critical components such as login function, payment processing. Later on, integration testing will show that separate elements work together such as order posting is as smooth as stock updates

are made. Given system testing, the entire application will be tested while using functional and nonfunctional specifications and the UAT will involve actual users in making sure that the application fits the expectations of users. In order to ensure stability and avoid possible side effects of new updates always the same testing procedures will be performed.

Maintenance Plan

In order to maintain its long-term efficiency and stability, IMS has to be maintained constantly after deployment. It includes bug fix updating, securing patch implementation, and system performance review in a bid of catching any possible problems before they build up. A sophisticated system of backup will be put in place to ensure that the data is safe, and including organized backups with a recovery process. To facilitate effective management of issues, users will be able to create tickets for known problems or feature suggestions and have an improved method in identifying, managing issues. Scheduled updates in documentation will demonstrate system changes and new features, providing communication among a development team, administrators and end users. continuous evaluation in basis of regular reviews will ensures that the IMS stays in line with changing business needs.

Development Section

All the wireframes are made using Figma.

Login Page:



Figure 7 Login Page UI

Register Page:

Figure 8 Register Page UI

Main Dashboard:

Figure 9 Main Dashboard UI

Products List:

Figure 10 Product Page UI

Add Product:

Figure 11 Add Product UI

Sales Report Overview:

Figure 12 Sales Report Overview UI

Sales Report Details:

Figure 13 Sales Report Details UI

Payment:

Figure 14 Payment Page UI

User Profile:

Figure 15 User Profile UI

Low Stock Products:

Figure 16 Low Stock Products UI

Recent Transactions:

Payment Confirmation:

Figure 18 Payment Confirmation UI

Navigation Sidebar:

Figure 19 Sidebar UI

Change Password:

Figure 20 Credentials change UI

Header Bar:

Figure 21 Header bar for user information UI