|  |  |
| --- | --- |
| **Executive Summary (0.25) & Conclusion (0.25)** | **0.5 points** |
| **I. Business Requirements** | **1.0 points** |
| **II. System Requirements Analysis (each sub item is 1pts)** | **3.0 points** |
| **III. System Requirements Analysis (each sub item is 1pts)** | **3.0 points** |
| **IV .System Requirements Implementation** | **2.0 points** |
| **V. System Testing, Deployment and Demo** | **1.5 points** |
| **(*include* *Visio/PowerDesign/Drawing source* + *source code*)** |  |
| **Total** | **11 points** |

**You are encouraged to use Power Design software to conduct modelling including but not limited to UML class diagram to see how to generate both SQL and Java/C# code to earn points for implementation.**

**Remove all picture / unused part from this template, you should keep the titles, when you conduct your report.**

**Should maintain the navigation on the left hand-side of this report when opening with MS Word.**

**B2B Information Systems for Agents’ Retail Stores**

**Requirements Analysis & Design (RAD)**

**By Students:**

1. **521H0439 : Mai Thanh Binh**
2. **521H0156 : Nguyen Mai Tan Thanh**

|  |  |  |  |
| --- | --- | --- | --- |
| **Reference:** | **Team\_07\_RAD\_Requirements\_Modelling\_v0.1** | | |
| **Audience:** | **Mr. Pham Thai Ky Trung** | **Document Version:** | **December, 2023** |
| **Outcome:** | **Your Project Title** | | |
| **Abstract:** | This document provides an in-depth analysis of a proposed video rental system with the requirements modelled utilizing the UML framework. The document is a collaboration between the members of Team XX. | | |
|  | | | |

# Intellectual Property

***Copyright 2022 Team XX.***

The following documentation, the content therein and/or the presentation of its information is proprietary to and embodies the confidential processes, designs, technologies and otherwise of Team XX. All copyright, trademarks, trade names, patents, industrial designs, and other intellectual property rights contained herein are, unless otherwise specified, the exclusive property of Team XX.

The ideas, concepts and/or their application, embodied within this documentation remain and constitute items of intellectual property which nevertheless belong to Team XX.

The information (including, but by no means limited to, data, drawings, specification, documentation, software listings, source and/or object code) shall not be disclosed, manipulated, disseminated or otherwise in any manner inconsistent with the nature and/or conditions under which this documentation has been issued.

The information contained herein is believed to be accurate and reliable. Team XX accepts no responsibility for its use in any way whatsoever. Team Five shall not be liable for any expenses, damages and/or related costs which may result from the use of the information contained herein.

The information contained herein is subject to change without notice.

All Rights Reserved. Copyright herein is expressly protected at common law, statute and under various International and Multi-National Treatises (including, but by no means limited to, the Berne Convention for the Protection of Literary and Artistic Works).

# Table of Contents

[Intellectual Property 3](#_3cqmetx)

[Table of Contents 4](#_30j0zll)

[Executive Summary (0.25 point) 6](#_1rvwp1q)

[I.](#_3znysh7) Business Requirements (1 point) 7

[1.1](#_3znysh7) Organization Chart / Project Chart/Gantt Chart 7

[1.2](#_tyjcwt) Business Modelling / Requirements 7

[1.3](#_3dy6vkm) Business Processes / Flowchart of Requirements 8

[1.4](#_1t3h5sf) Activity Diagram 8

[1.5](#_4d34og8) List of Requirements 9

[II.](#_2s8eyo1) System Requirements Analysis (3.0 points) 10

[2.1](#_2s8eyo1) Translate from Business Use Case 10

[2.1.1.](#_17dp8vu) System Narrative 10

[2.1.2.](#_3rdcrjn) Users and their goals 10

[2.1.3.](#_26in1rg) List of Events 11

[2.1.4.](#_lnxbz9) List of Actors 11

[2.1.5.](#_35nkun2) List of Use Cases 11

[2.1.6.](#_1ksv4uv) Use Case Diagram 12

[2.1.7.](#_44sinio) Domain Class Model Diagram 13

[2.2](#_2jxsxqh) Use Case Descriptions 14

[2.2.1.](#_2jxsxqh) Use Case: Name 1 14

[2.2.2.](#_z337ya) Use Case: Name 2 16

[2.2.3.](#_3j2qqm3) Use Case: Name 3 16

[2.2.4.](#_1y810tw) Use Case: Name 4 16

[2.3](#_4i7ojhp) Verifying use cases for Actor 17

[2.3.1.](#_4i7ojhp) Verifying uses cases: Name 1 for ~~Actor 1~~ Customer 17

[2.3.2.](#_2xcytpi) Verifying uses cases for Actor 2 17

[2.3.3.](#_1ci93xb) Verifying uses cases for Actor 3 17

[2.3.4. 17](#_3whwml4)

[III.](#_2bn6wsx) System Requirements Design (3.0 points) 18

[3.1](#_qsh70q) Design Class for Use Case 1 18

[3.2.1.](#_3as4poj) Design Classes in Detailed Design 18

[3.2.2.](#_1pxezwc) Design Class Diagram 19

[i.](#_49x2ik5) Domain Design Class 19

[ii.](#_2p2csry) Controller 19

[iii.](#_147n2zr) UI 20

[iv.](#_3o7alnk) Data Access 20

[v.](#_23ckvvd) Design Class 20

[3.2.3.](#_ihv636) OOD with Communication 21

[3.2.4.](#_32hioqz) OOD with Sequence Diagram 21

[3.2.5.](#_1hmsyys) Final Design Class Diagram 22

[3.2](#_41mghml) Design Class for Use Case 2 23

[3.2.1.](#_41mghml) Design Classes in Detailed Design 23

[3.2.2.](#_2grqrue) Design Class Diagram 23

[i.](#_vx1227) Domain Design Class 23

[ii.](#_3fwokq0) Controller 23

[iii.](#_1v1yuxt) UI 23

[iv.](#_4f1mdlm) Data Access 23

[v.](#_2u6wntf) Design Class 23

[3.2.3.](#_19c6y18) OOD with Communication 23

[3.2.4.](#_3tbugp1) OOD with Sequence Diagram 23

[3.2.5.](#_28h4qwu) Final Design Class Diagram 23

[IV.](#_nmf14n) System Requirements Implementation (2pts) 24

[4.1](#_nmf14n) Design Class for Sub System 24

[4.2](#_37m2jsg) Package Diagram for Sub System 25

[4.3](#_1mrcu09) Implementation 26

[4.3.1.](#_1mrcu09) Map persistent objects to the tables in a database 26

[4.3.2.](#_46r0co2) Modifying sequence diagrams 27

[4.3.3.](#_2lwamvv) UI design 27

[4.3.4.](#_111kx3o) SQL Code 28

[4.3.5.](#_3l18frh) Software Classes Method Code 28

[V.](#_206ipza) SYSTEM TESTING, DEPLOYMENT AND DEMONSTRATION (1.5 pts) 29

[5.1](#_206ipza) Testing: Test plan & Test case 29

[5.2](#_4k668n3) Deployment 29

[5.3](#_2zbgiuw) Demonstration 29

[Conclusions/ Recommendations (0.25 point) 30](#_1egqt2p)

[References 30](#_3ygebqi)

[Appendix 30](#_sqyw64)

# Table of Figures

[Figure 1: Org Chart 7](#_2et92p0)

# Executive Summary (0.25 point)

The purpose of this document is to provide an overview of the proposed system….

The proposed retail management system offers a comprehensive and well-organized solution to address different aspects of retail operations. It encompasses features for managing accounts, product catalog, sales tracking, and inventory control. The system's modular design ensures easy navigation and efficient management across multiple retail stores.

One of the key advantages of the system is its centralized control and management capabilities. It provides a unified platform to oversee and manage accounts, products, customers, sales transactions, inventory, and reports. This centralized approach promotes consistency and streamlines operations across all retail locations.

The system also prioritizes user-friendliness with an intuitive interface. It offers clear navigation menus and role-based access controls, allowing staff members to perform their tasks efficiently. This user-friendly design enhances productivity and minimizes the learning curve for users.

Real-time inventory tracking is another crucial feature of the system. It provides up-to-date information on stock levels, enabling managers to make informed decisions regarding inventory management, purchasing, and product distribution. This real-time visibility helps optimize inventory levels and avoid stockouts or excess inventory.

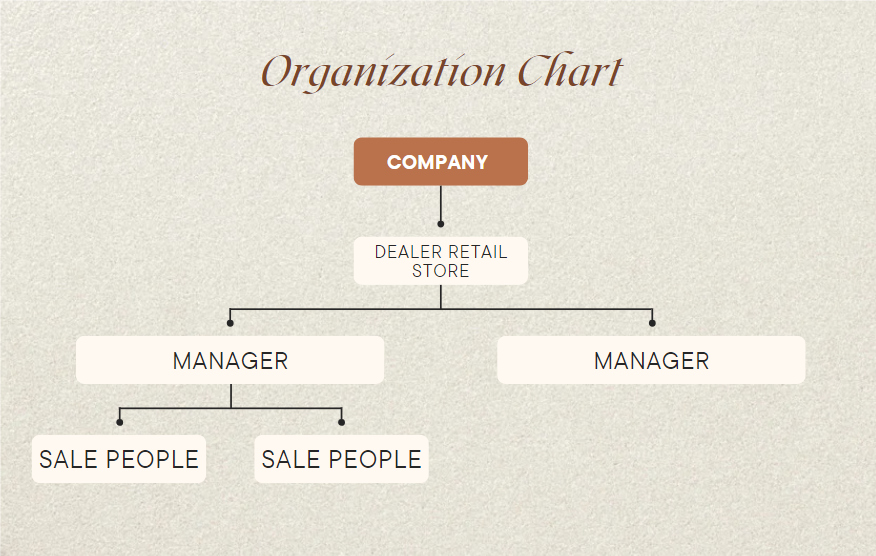
The system's reporting capabilities are comprehensive and valuable for decision-making. It generates detailed reports that provide insights into sales trends, customer behavior, and inventory performance. These reports support data-driven decision-making, allowing retailers to identify opportunities, track performance metrics, and make informed business decisions.

Overall, the proposed retail management system offers a well-structured and comprehensive solution for managing various aspects of retail operations. Its centralized control, user-friendly interface, real-time inventory tracking, and comprehensive reporting contribute to streamlining operations, enhancing efficiency, and facilitating informed decision-making in the retail industry.

# Business Requirements (1 point)

# Organization Chart / Project Chart/Gantt Chart

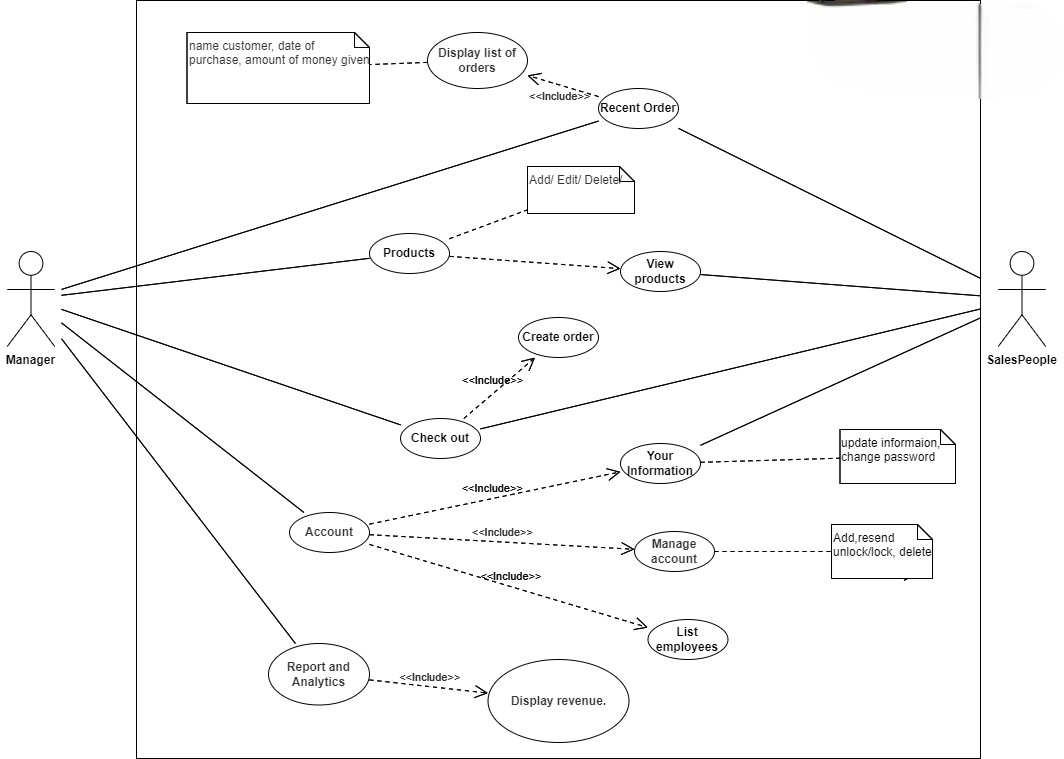
Can use chapter 11 from Guide to show here



**Figure 1: Org Chart**

# Business Modelling / Requirements

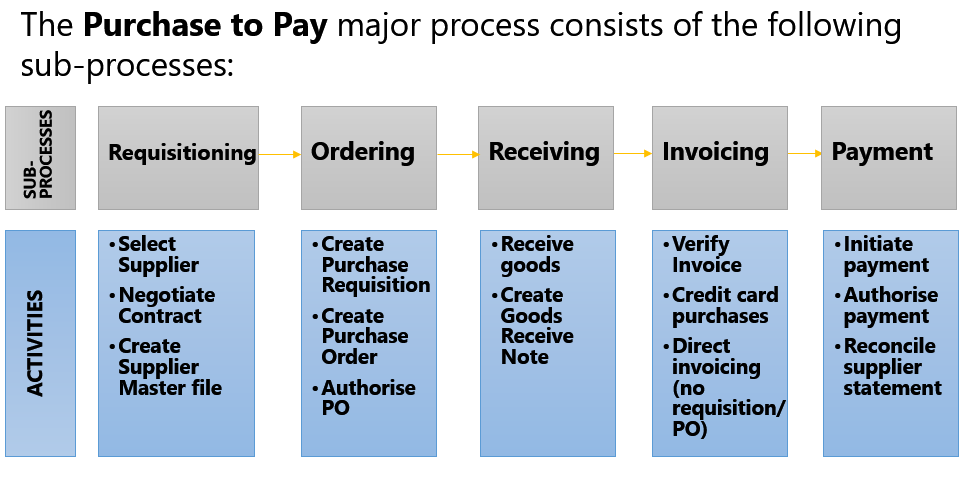
**MODEL BUSINESS USE CASE**



**MODEL BUSINESS ACTIVITY DIAGRAM**

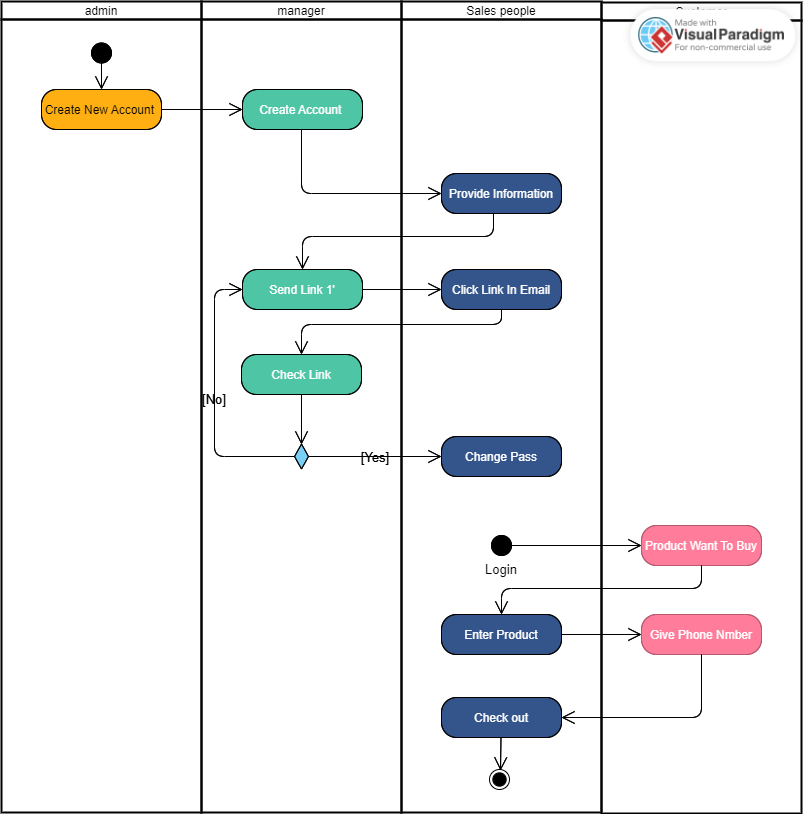
# Business Processes / Flowchart of Requirements

If you think about the process then you can refer to the following :



Activtites on the picture can also mean “use case” or “ system function”

# Activity Diagram



# List of Requirements

|  |  |  |
| --- | --- | --- |
| List of requirements | Function | Non-function |
| Administrator account creation with customizable password | X |  |
| Salesperson account creation by the administrator, including name and Gmail address | X |  |
| Email notification to salesperson with a login link | X |  |
| Time-limited login link for new employees | X |  |
| Salesperson login using their username | X |  |
| Salesperson forced to create a new password on the first login | X |  |
| Profile management for all employees (view, update profile picture, change password) | X |  |
| Administrator functions for managing employee accounts (view, resend login email, lock/unlock account, view sales information) | X |  |
| Administrator-only access to product catalog management | X |  |
| View product listings | X |  |
| Add new products with barcode/QR code, name, import price, retail price, category, and creation date | X |  |
| Update product information | X |  |
| Delete products not associated with any orders | X |  |
| Capture customer's phone number at checkout | X |  |
| Display customer information (name, address) if the customer has made a previous purchase | X |  |
| Create a new customer account for first-time buyers | X |  |
| View customer information and purchase history | X |  |
| View order details (products, selling prices) | X |  |
| Add products to the purchase list (search, barcode/QR code entry) | X |  |
| Display the purchase list with product information (quantity, unit price, total amount) | X |  |
| Enter customer information during the payment process (phone number) | X |  |
| Calculate and display the total order amount | X |  |
| Accept payment from the customer and calculate the change | X |  |
| Generate and print an invoice | X |  |
| View sales results by timeline (date range, month, quarter, year) | X |  |
| Display total amount received, number of orders, and number of products | X |  |
| View a list of orders arranged chronologically | X |  |
| Detailed view of individual orders | X |  |
| Secure password storage and transmission |  | X |
| Limited access to administrative functions and sensitive information |  | X |
| Clear and organized display of customer and order information |  | X |
| User-friendly interface for sales staff and administrators |  | X |
| Ability to handle a growing number of products, customers, and orders |  | X |
| System stability and availability |  | X |
| Proper error handling and notification for exceptional scenarios |  | X |

# 

# System Requirements Analysis (3.0 points)

# Translate from Business Use Case

## System Narrative

The sales point application is designed to provide a comprehensive solution for managing sales transactions, customer information, and product catalog in a retail environment. The system aims to streamline the sales process, improve customer service, and enable effective management of sales data.

The system starts with an administrator account, which is created with a customizable password. The administrator has the authority to create accounts for salespeople, who cannot create their own accounts. The salesperson's account is created by the administrator, who provides the salesperson's full name and Gmail address. An email notification is sent to the salesperson, containing a login link.

The salesperson receives the email and clicks on the login link within a one-minute window. This time-limited link ensures the security of the login process. The salesperson is required to use their username, which is the front part of their email, to log into the system. If the salesperson tries to log in without using the link, they receive an error message instructing them to log in using the link in their email.

On the first login, the salesperson is prompted to create a new password. This step ensures the security of their account. After logging in, all employees, including administrators, can view and update their profile information, including their profile picture and password.

Administrators have additional functions related to employee management. They can view a list of employees, including their avatar, full name, inactive status (if any), and locked status (if any). Administrators can view the details of an employee, resend the one-minute login email, lock/unlock an account, and view the sales information of an employee.

The system provides a product catalog management feature, accessible only to administrators. Administrators can view the product listings, add new products with barcode/QR code, name, import price, retail price, category, and creation date. They can also update product information and delete products that are not associated with any orders. Sales staff can view the product list but cannot make changes to the content. The sales staff do not have access to the original price of the product.

Customer management is an essential part of the system. At the checkout counter, the salesperson asks for the customer's phone number. If the customer has made a previous purchase, their name and address are automatically displayed. If it is the customer's first time buying, the salesperson enters the customer's full name and address, and the system automatically creates a new account for the customer. Customer accounts cannot be created manually.

In the sales interface, employees can easily view customer information, including name, phone number, address, and their purchase history. The purchase history includes information such as the total amount, amount given by the customer, excess amount paid back, date of purchase, and product quantity. Employees can also view the details of an order to see a list of products and their selling prices.

The transaction processing function is the core feature of the system. Sales staff can add products to the purchase list by searching or entering barcodes/QR codes. The added products are displayed in a list view, showing the number of items, unit price, and total amount for each product. The interface provides summary information such as the total amount of the entire order. During the payment process, the salesperson can enter the customer's phone number or enter the customer's full name and address for new customers. The system calculates the change and generates an invoice, which can be printed as a PDF file.

The system includes reporting and analytics capabilities. Sales results can be viewed by timelines such as a specific date range, month, quarter, or year. The system displays information such as the total amount received, the number of orders, the number of products, and a list of orders arranged chronologically. Users can select a specific order to view its details.

Both administrators and sales staff have access to the reporting and analytics features. Administrators can see additional information such as the total profit. The system ensures security by implementing secure password storage and transmission. It also provides a user-friendly interface for sales staff and administrators, with intuitive product search and entry methods. The system aims for efficient performance, scalability to handle a growing number of products, customers, and orders, reliability in terms of stability and availability, and proper error handling and notification for exceptional scenarios.

## Users and their goals

|  |  |
| --- | --- |
| User/actor | Goal |
| Administrators | The administrators are responsible for managing the system and overseeing the sales operations. Their goals include:Creating and managing employee accounts.Managing the product catalog, including adding, updating, and deleting products.Monitoring sales performance and generating reports.Ensuring the security and integrity of the system.Providing support and assistance to sales staff. |
| Customers | The salespeople are the primary users of the application, involved in day-to-day sales activities. Their goals include:Logging into the system to access their account and sales-related information.Updating their profile information and avatar.Adding products to the purchase list for customers.Processing transactions, including entering customer information and accepting payments.Providing excellent customer service by accessing customer information and purchase history.Generating invoices for customers.Viewing their own sales performance and metrics. |
| Salespeople | Customers interact indirectly with the sales point application through the salespeople. Their goals include:Making purchases and providing their phone number or personal information at the checkout counter.Receiving accurate and efficient service during the transaction process.Receiving proper and clear invoices for their purchases.Potentially benefiting from personalized service based on their purchase history. |

## 

## List of Events

List of events and its use case

|  |  |
| --- | --- |
| Event | Use case |
| Administrator account creation | Allows the creation of an administrator account with a customizable password, granting access to system management functionalities |
| Administrator creates a new salesperson account | Permits the administrator to create accounts for salespeople, enabling them to access the application and perform sales-related functions |
| Administrator updates employee account information | Allows the administrator to modify employee account details, such as name or contact information |
| Administrator send link 1 minute | permit send again a link to salespeople login |
| Administrator views employee sales information | Provides the administrator with insights into the sales performance of individual employees, supporting evaluation and decision-making |
| Administrators block an employee account | Enables the administrator to remove employee accounts that are no longer needed or have become inactive |
| Administrator manages the product catalog | Allowing the addition, modification, or removal of products |
| Administrator generates sales reports | Provides the administrator with detailed sales reports, including information such as total sales, revenue, and trends, aiding in analysis and decision-making |
| Salesperson logs into the system in the first time | When login in the first time, salesperson need to change password |
| Salesperson adds products to the purchase list | Enables the salesperson to select and add products to the list for a specific customer's purchase |
| Salesperson searches for products by barcode or name | Provides the salesperson with a convenient way to search for products using barcode scanning or by entering the product name |
| Salesperson enters customer information | Allows the salesperson to collect and input customer information, such as phone number or address, for completing the transaction |
| Salesperson calculates and accepts payment | Supports the salesperson in calculating the total order amount, accepting payment from the customer, and processing the transaction |
| Generates an invoice | Enables the salesperson to create an invoice summarizing the purchase details, which can be provided to the customer |
| Salesperson views their own sales performance metrics | Allows the salesperson to access their individual sales performance data, including total sales, commissions, and targets |
| Salesperson views customer information and purchase history | Provides the salesperson with access to customesr details, such as name, contact information, and previous purchase history, to personalize the sales experience |
| Customer provides their phone number or personal information at the checkout counter | Allows the customer to provide their contact information for identification and future reference |
| Customer makes a purchase | Represents the customer's action of selecting and buying products from the store. |
| Customer receives an invoice for their purchase | Provides the customer with a clear and accurate invoice summarizing the purchased items and the total amount paid |
| New employee account creation email sent | Triggers the sending of an email to a newly created salesperson account, providing them with login instructions |
| Salesperson receives and clicks on the login link in the email | Allows the salesperson to access the application by clicking the login link received in the email |
| Salesperson creates a new password on the first login | Case: Prompts the salesperson to set a password for their account during their initial login |
| System validates and authenticates user login credentials | Verifies the provided username (email) and password to authenticate the user's identity and grant access to the system |
| System displays the product catalog to the salesperson | Presents the salesperson with the available products, allowing them to select and add items to the purchase list |
| System updates the purchase list with added products and quantities | Tracks and updates the purchase list with the selected products and their respective quantities |
| System calculates the total order amount | Performs the calculation of the total amount to be paid by the customer based on the selected products and their prices |
| System generates a change amount during payment calculation | Determines the change to be returned to the customer based on the amount tendered |
| System generates and saves invoices: | Automatically generates and saves invoices summarizing the purchase details for record-keeping and customer reference |

## List of Actors

1. Administrator:

- Description: Represents the administrator who has system management privileges and performs administrative tasks.

- Responsibilities:

- Creates and manages administrator accounts.

- Logs into the system to access administrative functionalities.

- Creates and manages salesperson accounts.

- Updates employee account information.

- Views employee sales information.

- Deletes employee accounts.

- Manages the product catalog (add, update, delete products).

- Generates sales reports.

2. Salesperson:

- Description: Represents the salesperson who interacts with the system to perform sales-related activities.

- Responsibilities:

- Logs into the system.

- Updates profile information.

- Adds products to the purchase list.

- Searches for products by barcode or name.

- Enters customer information during the payment process.

- Calculates and accepts payment.

- Generates an invoice.

- Views own sales performance metrics.

- Views customer information and purchase history.

3. Customer:

- Description: Represents the customer who makes purchases at the store.

- Responsibilities:

- Provides phone number or personal information at the checkout counter.

- Make a purchase.

- Receives an invoice for the purchase.

4. System:

- Description: Represents the video rental system.

- Responsibilities:

- Sends new employee account creation emails.

- Validates and authenticates user login credentials.

- Displays the product catalog to the salesperson.

- Updates the purchase list with added products and quantities.

- Calculates the total order amount.

- Generates a change amount during payment calculation.

- Generates and saves invoices.

## List of Use Cases

## Administrator Account Creation: Allows the creation of an administrator account with a customizable password, granting access to system management functionalities.

## Administrator Creates a New Salesperson Account: Permits the administrator to create accounts for salespeople, enabling them to access the application and perform sales-related functions.

## Administrator send link: allows the system to create a link for a limited time.

## Administrator Updates Employee Account Information: Allows the administrator to modify employee account details, such as name or contact information.

## Administrator Views Employee Sales Information: Provides the administrator with insights into the sales performance of individual employees, supporting evaluation and decision-making.

## Administrator Blocks an Employee Account: Enables the administrator to remove employee accounts that are no longer needed or have become inactive.

## Administrator Manages the Product Catalog: Allows the administrator to add, modify, or remove products in the product catalog.

## Administrator Generates Sales Reports: Provides the administrator with detailed sales reports, including information such as total sales, revenue, and trends, aiding in analysis and decision-making.

## Salesperson Logs into the System for the First Time: When logging in for the first time, the salesperson is prompted to change their password for security purposes.

## Salesperson Adds Products to the Purchase List: Enables the salesperson to select and add products to the list for a specific customer's purchase.

## Salesperson Searches for Products by Barcode or Name: Provides the salesperson with a convenient way to search for products using barcode scanning or by entering the product name.

## Salesperson Enters Customer Information: Allows the salesperson to collect and input customer information, such as phone number or address, for completing the transaction.

## Salesperson Calculates and Accepts Payment: Supports the salesperson in calculating the total order amount, accepting payment from the customer, and processing the transaction.

## Generates an Invoice: Enables the salesperson to create an invoice summarizing the purchase details, which can be provided to the customer.

## Salesperson Views Their Own Sales Performance Metrics: Allows the salesperson to access their individual sales performance data, including total sales, commissions, and targets.

## Salesperson Views Customer Information and Purchase History: Provides the salesperson with access to customer details, such as name, contact information, and previous purchase history, to personalize the sales experience.

## Customer Provides Their Phone Number or Personal Information at the Checkout Counter: Allows the customer to provide their contact information for identification and future reference.

## Customer Makes a Purchase: Represents the customer's action of selecting and buying products from the store.

## Customer Receives an Invoice for Their Purchase: Provides the customer with a clear and accurate invoice summarizing the purchased items and the total amount paid.

## New Employee Account Creation Email Sent: Triggers the sending of an email to a newly created salesperson account, providing them with login instructions.

## Salesperson Receives and Clicks on the Login Link in the Email: Allows the salesperson to access the application by clicking the login link received in the email.

## Salesperson Creates a New Password on the First Login: Prompts the salesperson to set a password for their account during their initial login.

## System Validates and Authenticates User Login Credentials: Verifies the provided username (email) and password to authenticate the user's identity and grant access to the system.

## System Displays the Product Catalog to the Salesperson: Presents the salesperson with the available products, allowing them to select and add items to the purchase list.

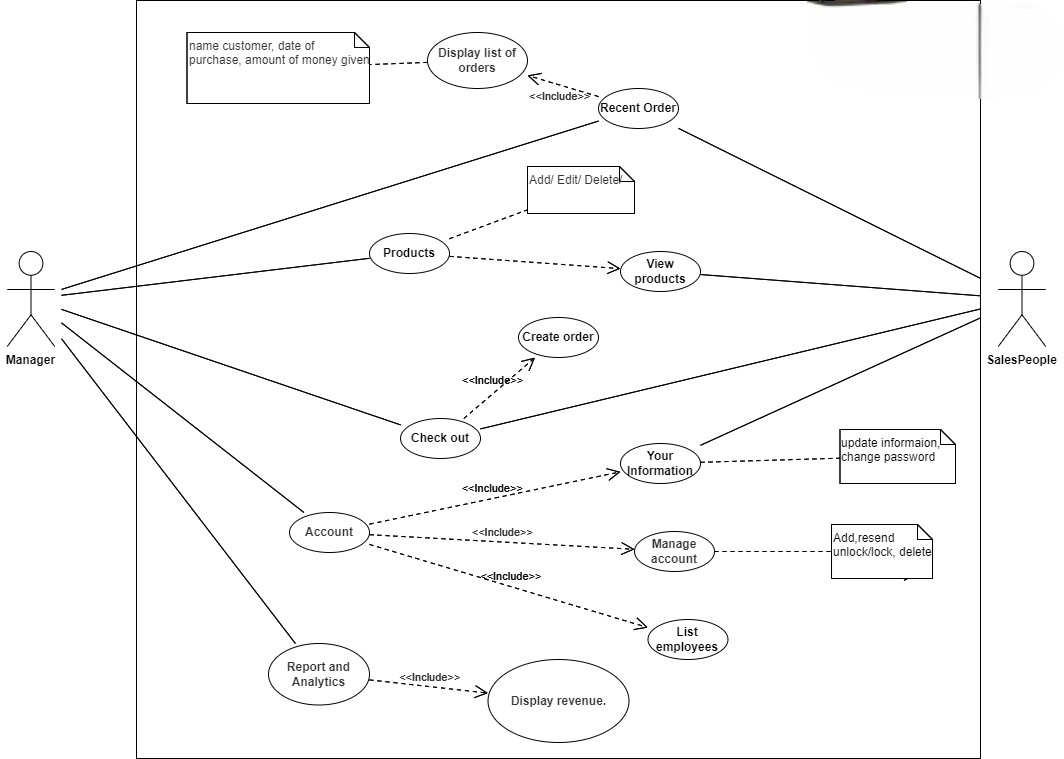
## System Updates the Purchase List with Added Products and Quantities: Tracks and updates the purchase list with the selected products and their respective quantities.

## System Calculates the Total Order Amount: Performs the calculation of the total amount to be paid by the customer based on the selected products and their prices.

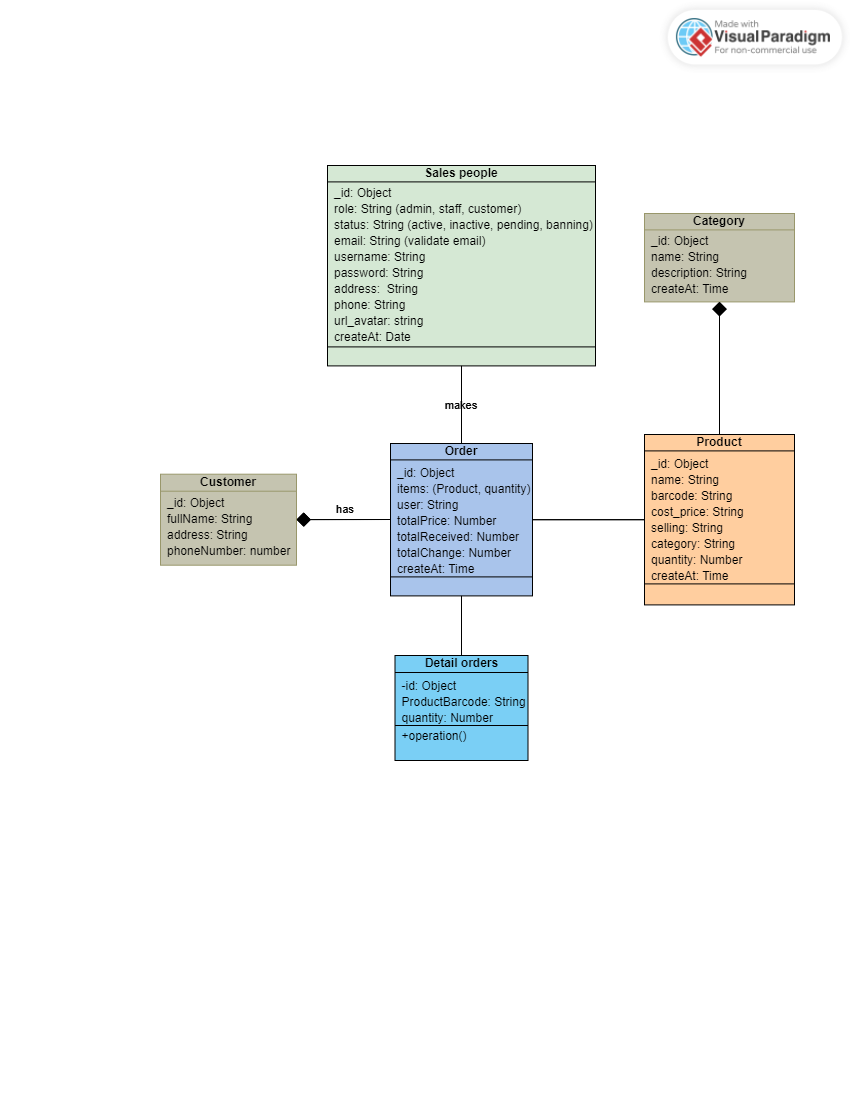
## System Generates a Change Amount During Payment Calculation: Determines the change to be returned to the customer based on the amount tendered.

## System Generates and Saves Invoices: Automatically generates and saves invoices summarizing the purchase details for record-keeping and customer reference.

## Use Case Diagram



## Domain Class Model Diagram



***Subclass model to system for send link 1 minute to login***

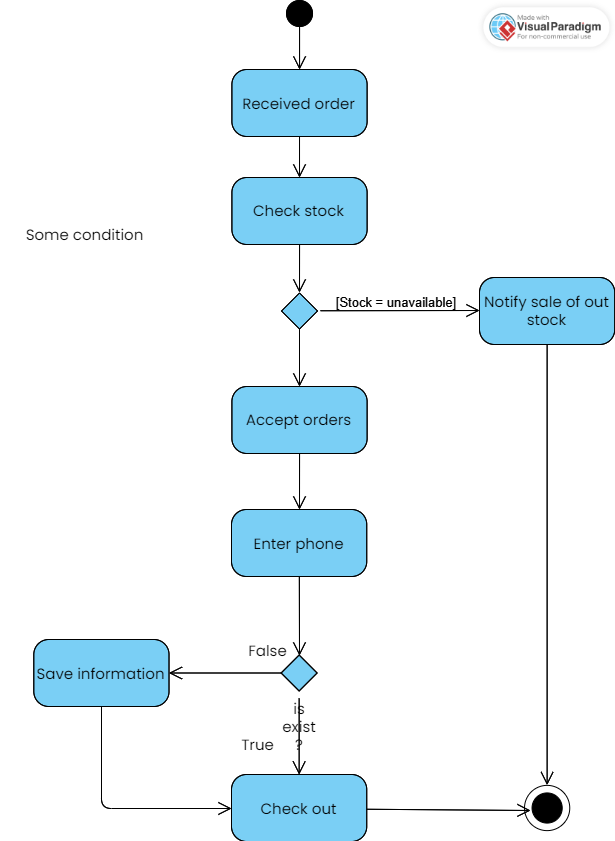
# Use Case Descriptions

## Use Case: Recent Orders

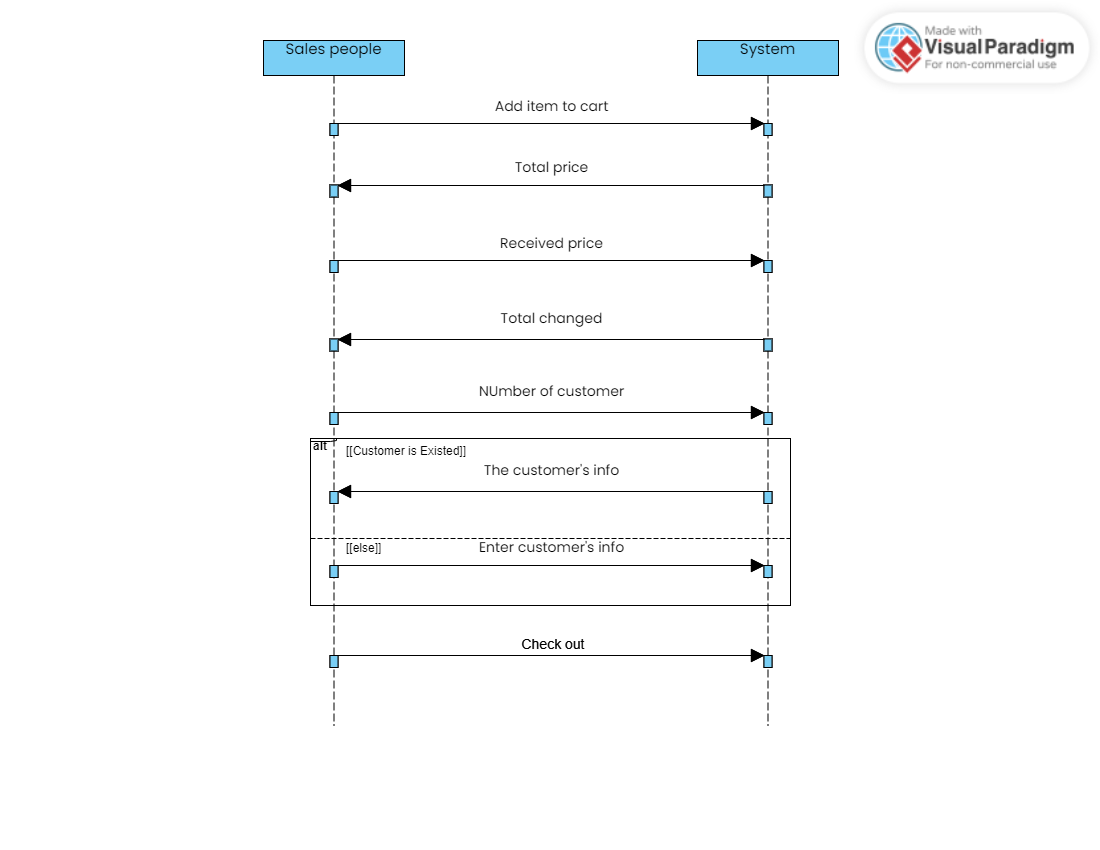
1. Use case : name 1 fully description

|  |  |  |
| --- | --- | --- |
| **USE CASE NAME** | Recent Orders | |
| **SCENARIO** | When user want to see list orders | |
| **TRIGGERING EVENT** | User want to see list orders | |
| **BRIEF DESCRIPTION** | This use case describes the process by which a user can view their recent orders through the Order Management System. | |
| **ACTORS** | Manager / Salespeople | |
| **RELATED USE CASE** | Login | |
| **STAKEHOLDER** | Manager / Salespeople | |
| **PRECONDITION** | Login successfully | |
| **POSTCONDITION** | User can manage orders easy. | |
| **FLOW OF EVENTS** | Manager / Salespeople | System |
| 1. Open “Apple store” app. 2. Click “Recent Order” on menubar. | Access database and perform query. |
| **EXCEPTION CONDITION** | None | |

1. Activity diagram for Use Case: Use Case Name 1



1. System sequence diagram for Use Case: Use Case Name 1



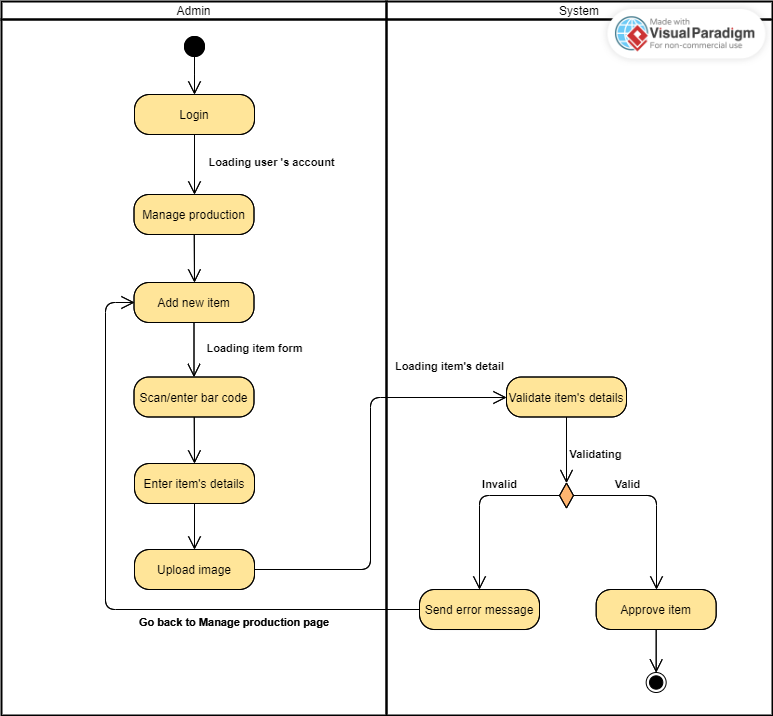
## Use Case: Products

1. Use case : products fully description

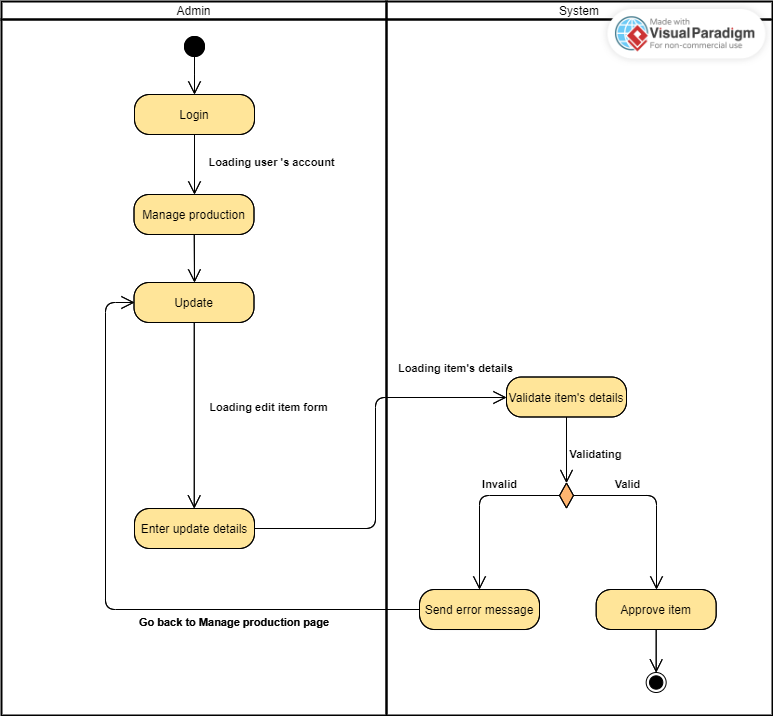
|  |  |  |
| --- | --- | --- |
| **USE CASE NAME** | Products | |
| **SCENARIO** | When user want to manage products | |
| **TRIGGERING EVENT** | User want to manage products | |
| **BRIEF DESCRIPTION** | User use manage products when they need do something. | |
| **ACTORS** | Manager. | |
| **RELATED USE CASE** | Login | |
| **STAKEHOLDER** | Manager | |
| **PRECONDITION** | Login successfully | |
| **POSTCONDITION** | Users can effectively manage products within the system.  Product information is accurate, up-to-date, and easily accessible to customers.  Product catalog is organized and easy to navigate.  Product performance is tracked and analyzed for informed decision-making. | |
| **FLOW OF EVENTS** | Actor | System |
| 1. Open “Apple store” app. 2. Click “Products” on menu bar. 3. Add/ Edit/ Delete product. (role: admin ) - Role employee: View products | Access database and add/ Edit/ Delete product. |
| **EXCEPTION CONDITION** | None | |

1. Activity diagram for Use Case: Products

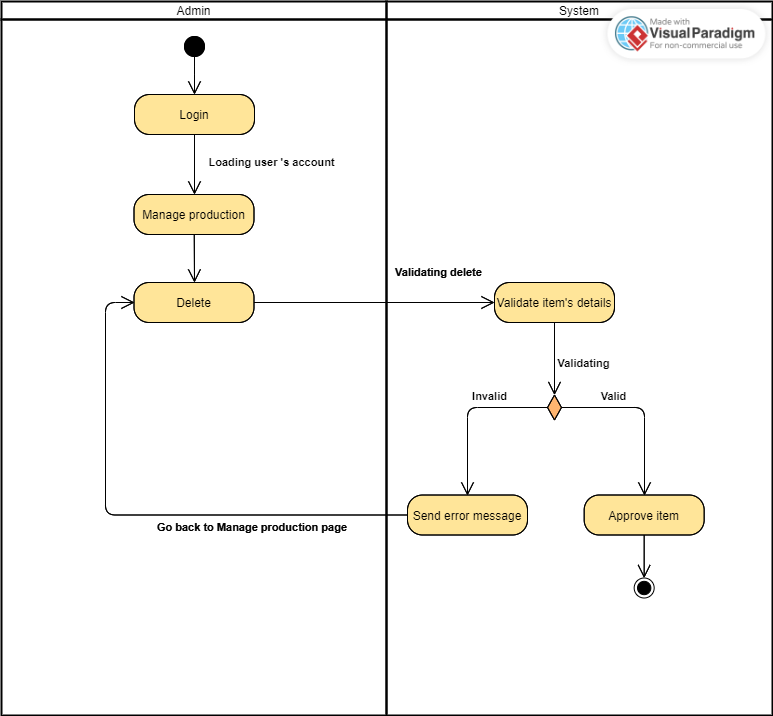
* Add product



* Edit product

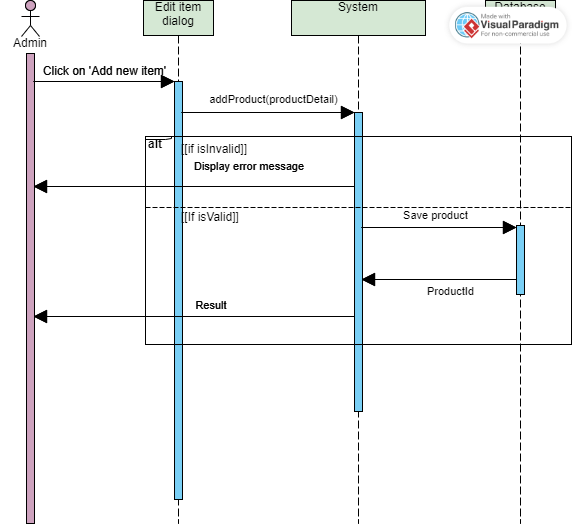


* Delete product

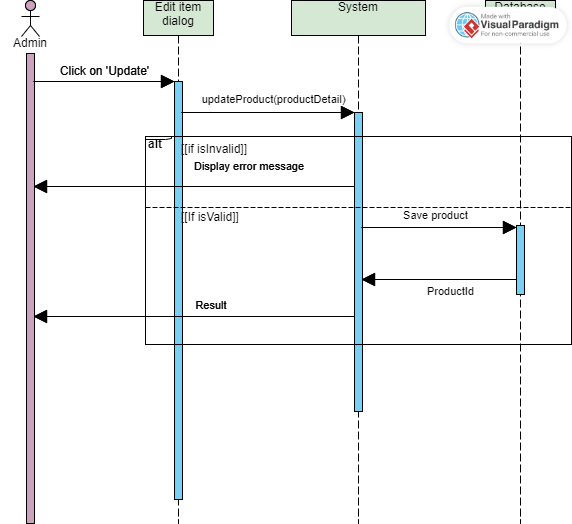


1. System sequence diagram for Use Case: Products

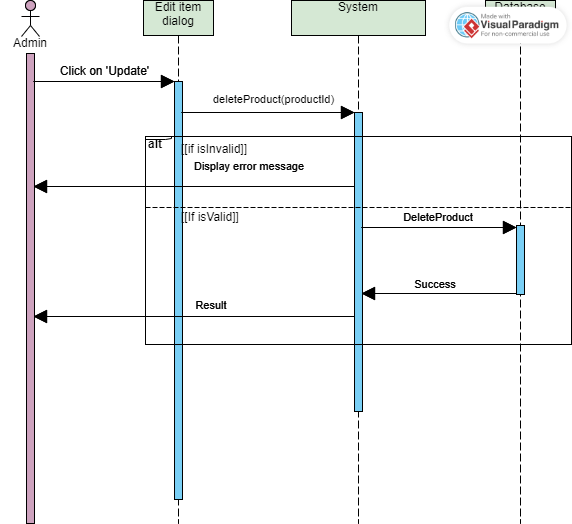
* Add product



* Edit product



* Delete product

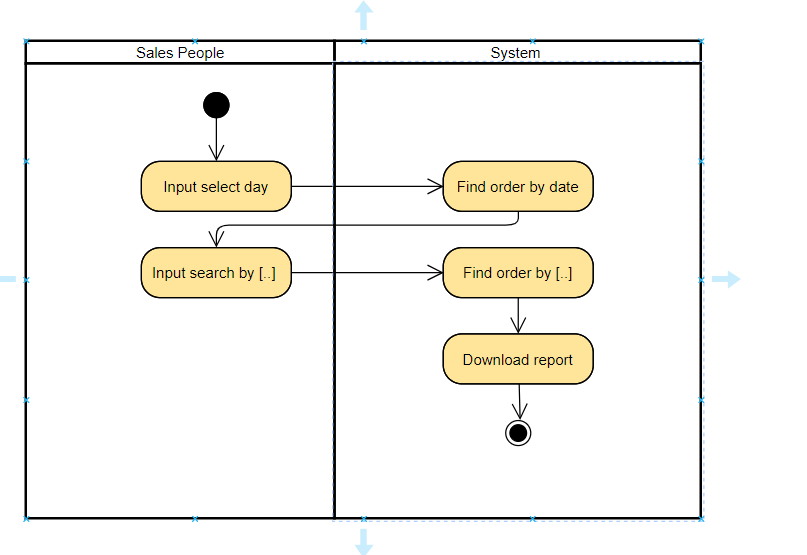


## Use Case: Report and Analytics

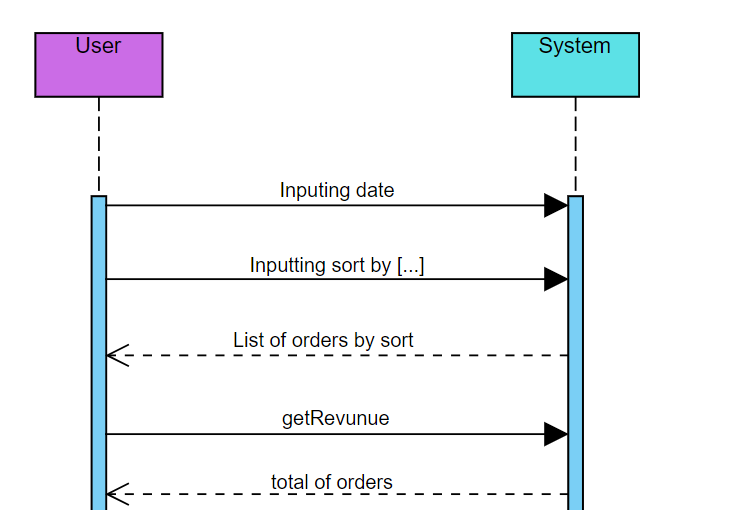
1. Use case : Customers fully description

|  |  |  |
| --- | --- | --- |
| **USE CASE NAME** | Report And Analytics | |
| **SCENARIO** | When user want to see sales report. | |
| **TRIGGERING EVENT** | User want to manage sales report | |
| **BRIEF DESCRIPTION** | User use view report when they need to know business situation. | |
| **ACTORS** | Manager/ SalesPeople | |
| **RELATED USE CASE** | Login | |
| **STAKEHOLDER** | Manager | |
| **PRECONDITION** | Login successfully | |
| **POSTCONDITION** | The user gains insights from the report's data. | |
| **FLOW OF EVENTS** | Actor | System |
| 1. Open “Apple store” app. 2. Click “Report and Analytics” on menu bar. | Access database and perform query. |
| **EXCEPTION CONDITION** | None | |

1. Activity diagram for Use Case: Name 3



1. System sequence diagram for Use Case: Name 3

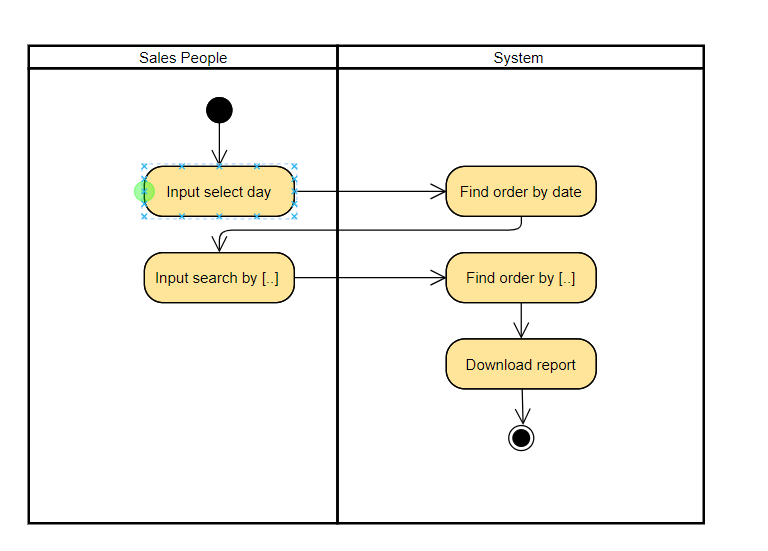


## Use Case: Manage Account

1. Use case : name 4 fully description

|  |  |  |
| --- | --- | --- |
| **USE CASE NAME** | Manage Account. | |
| **SCENARIO** | When user manage employees. | |
| **TRIGGERING EVENT** | User want to manage employees. | |
| **BRIEF DESCRIPTION** | User use manage employees when they need do something. | |
| **ACTORS** | Manager. | |
| **RELATED USE CASE** | Login | |
| **STAKEHOLDER** | Manager | |
| **PRECONDITION** | Login successfully | |
| **POSTCONDITION** | The user gains a comprehensive understanding of the employee's personal and professional information.  The user can utilize the employee data for various HR-related purposes, such as performance management, benefits administration, or communication.  The user can identify potential areas for employee development or support. | |
| **FLOW OF EVENTS** | Actor | System |
| 1. Open “Apple store” app. 2. Click “Manage Account” on menu bar. 3. Add/ resend-email/ lock – unlock/ delete/ change password employees. | Access database and control database follow function. |
| **EXCEPTION CONDITION** | None | |

1. Activity diagram for Use Case: Name 4



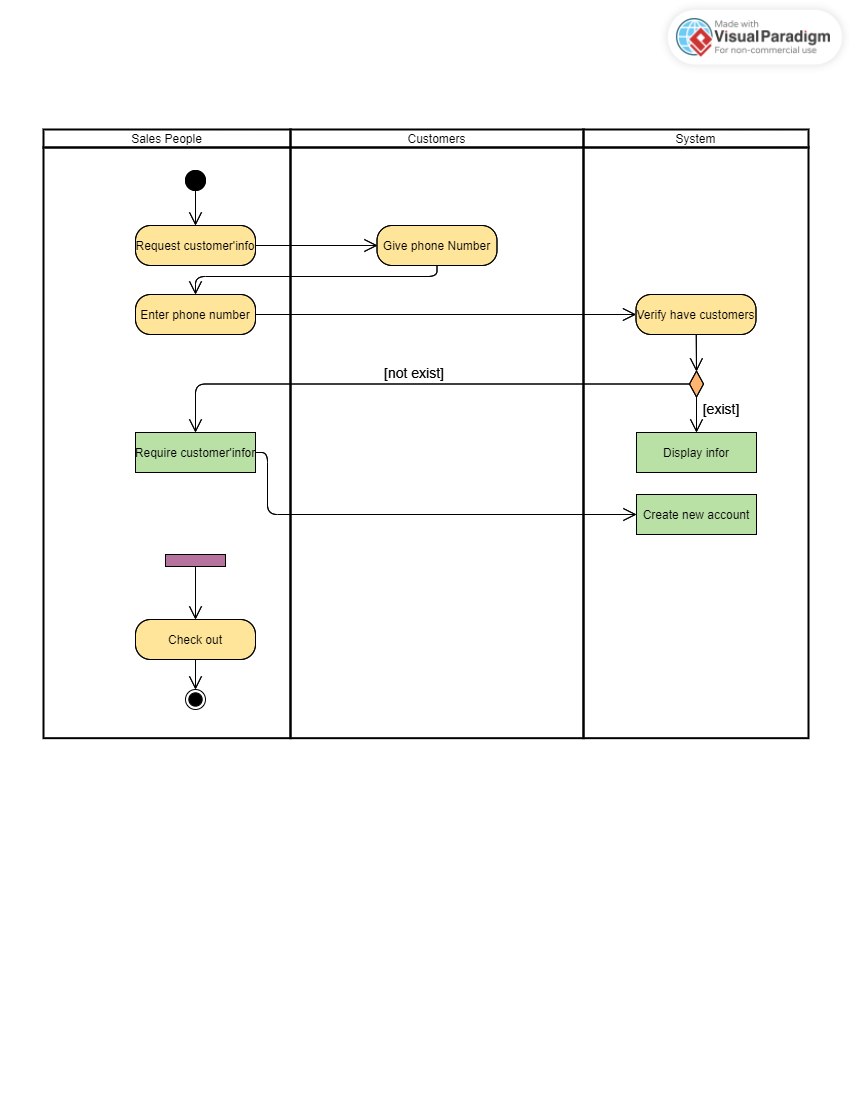
1. System sequence diagram for Use Case: Name 4

## Use Case: Check out

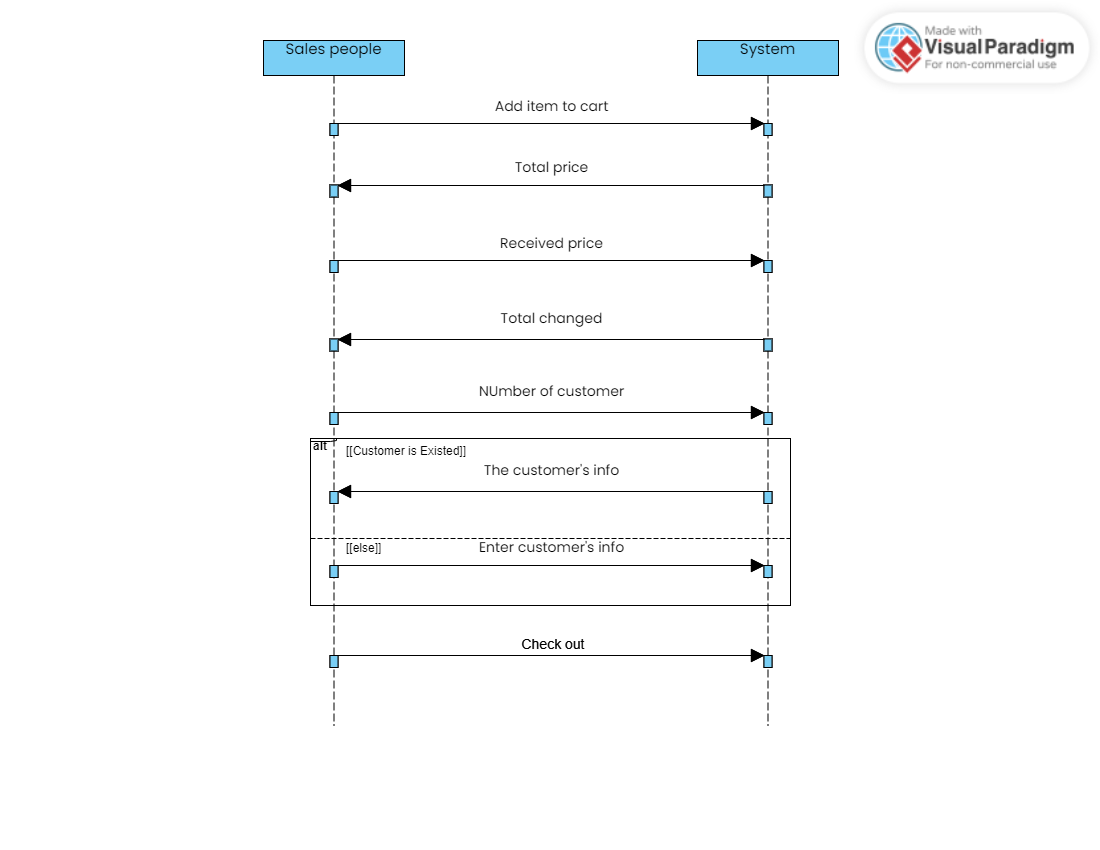
1. Use case : name 4 fully description

|  |  |  |
| --- | --- | --- |
| **USE CASE NAME** | Check out | |
| **SCENARIO** | When user want to create orders. | |
| **TRIGGERING EVENT** | User want to create orders. | |
| **BRIEF DESCRIPTION** | User use check out when they need create orders. | |
| **ACTORS** | Manager/ employee. | |
| **RELATED USE CASE** | Login | |
| **STAKEHOLDER** | Manager/ employee | |
| **PRECONDITION** | Login successfully | |
| **POSTCONDITION** | Users can effectively create orders for customers. | |
| **FLOW OF EVENTS** | Actor | System |
| 1. Open “Apple store” app. 2. Click “Check out” on menu bar. 3. Input valid information and make payment for orders. | 1. Access database and add orders. |
| **EXCEPTION CONDITION** | None | |

1. Activity diagram for Use Case: Name 4



1. System sequence diagram for Use Case: Name 4



1. **Verifying use cases for Actor**

## Verifying uses cases for Administrator

|  |  |  |
| --- | --- | --- |
| **Data entity/domain class** | **C R U D** | **Verified use case** |
| Salesperson | Create | Create a new salesperson account |
|  | Read/report | View salesperson account details  Generate salesperson performance report |
|  | Update | Update salesperson account information  Assign roles and permissions to salesperson account |
|  | Delete | Delete salesperson account |

## Verifying uses cases for Salesperson

|  |  |  |
| --- | --- | --- |
| **Data entity/domain class** | **C R U D** | **Verified use case** |
| Product | Create |  |
|  | Read/report |  |
|  | Update |  |
|  | Delete |  |

## Verifying uses cases for Actor 3

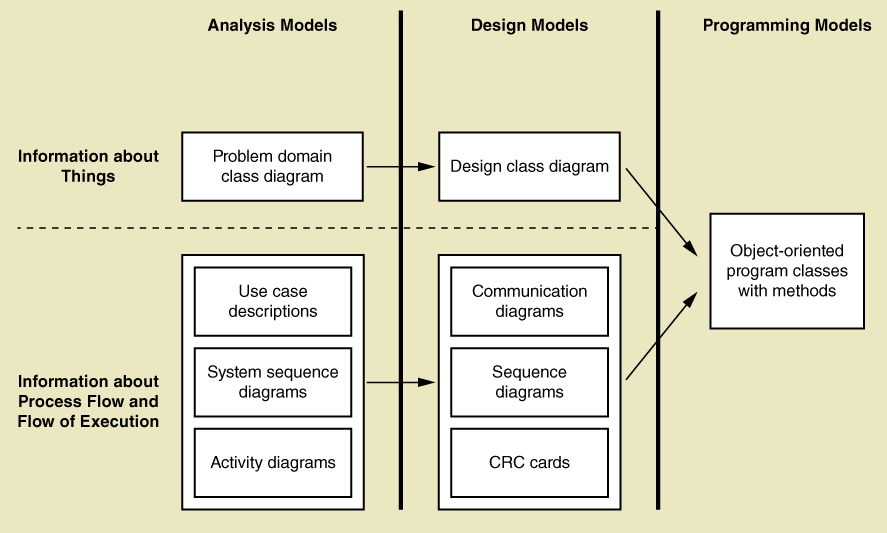
|  |  |  |
| --- | --- | --- |
| **Data entity/domain class** | **C R U D** | **Verified use case** |
|  | Create |  |
|  | Read/report |  |
|  | Update |  |
|  | Delete |  |

## Verifying uses cases for Actor 4

|  |  |  |
| --- | --- | --- |
| **Data entity/domain class** | **C R U D** | **Verified use case** |
|  | Create |  |
|  | Read/report |  |
|  | Update |  |
|  | Delete |  |

# System Requirements Design (3.0 points)

This part is to conduct from Analysis to Design:

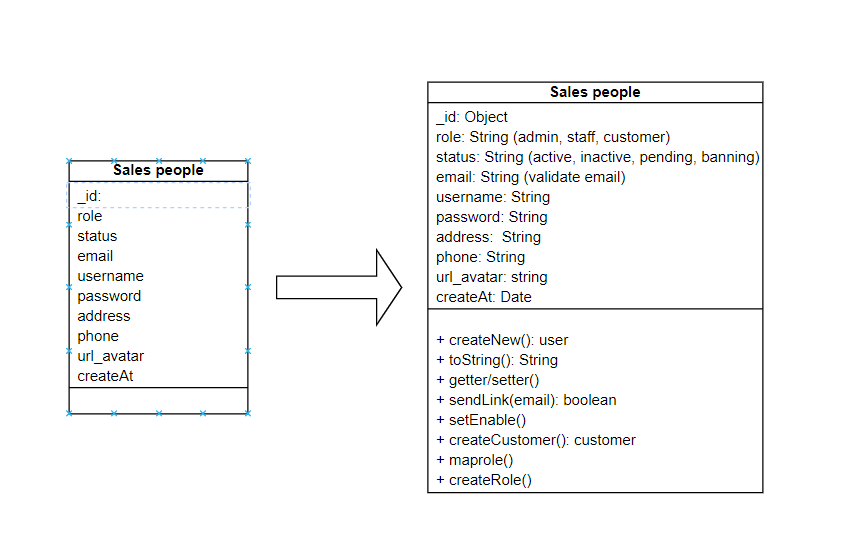


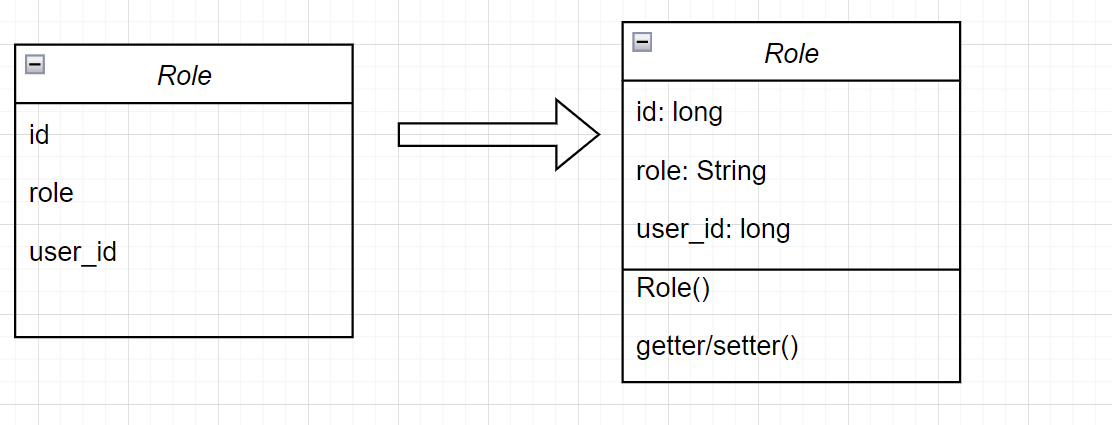
For this part, you should complete Design Models Diagram as above.

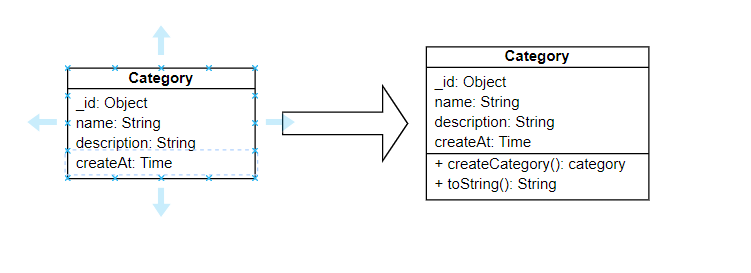
# Design Class for Use Case 1

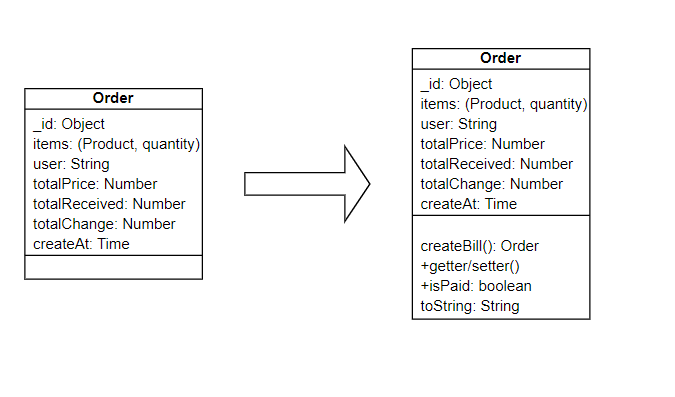
## Design Classes in Detailed Design

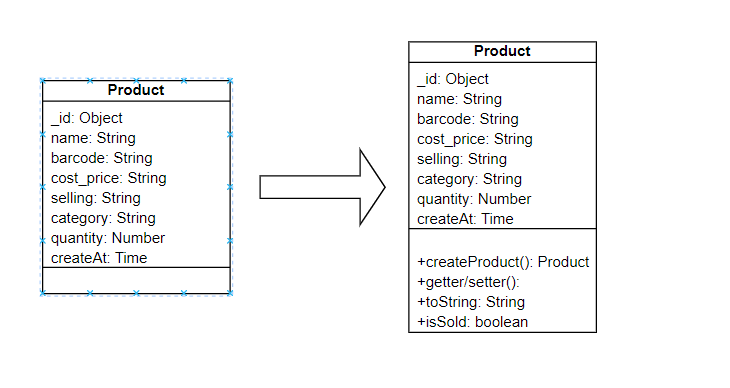
Convert Domain Classes to Design Classes

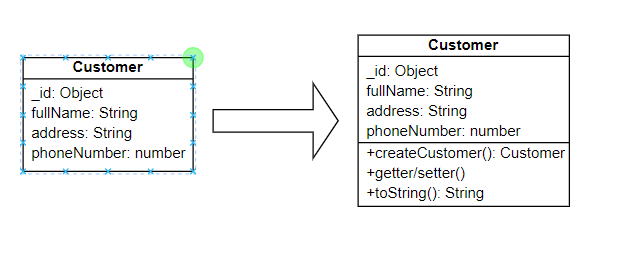






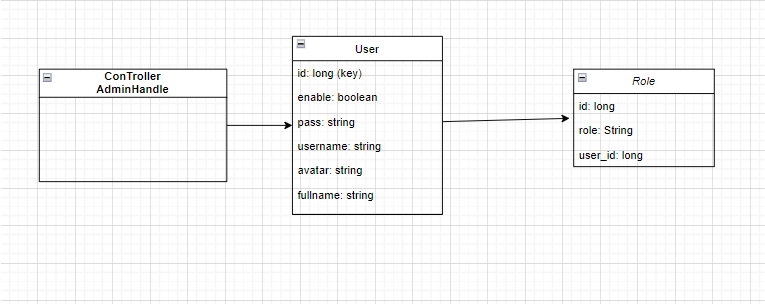




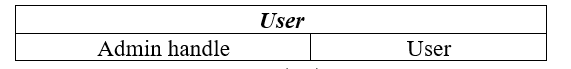


## Design Class Diagram

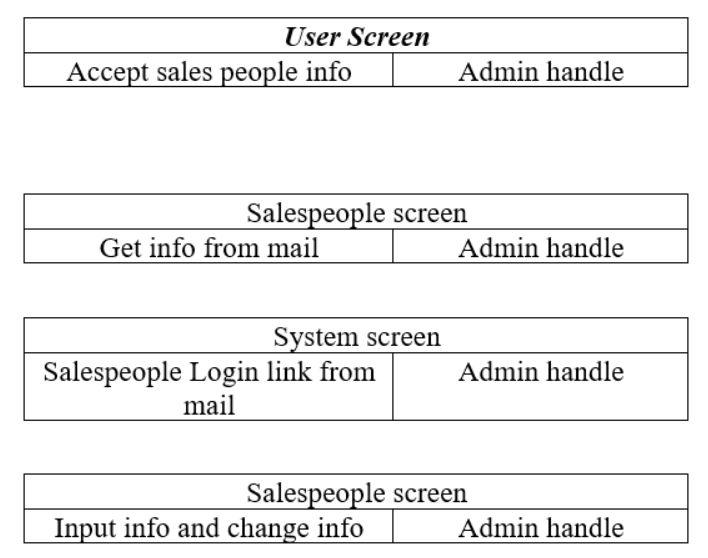
## Domain Design Class

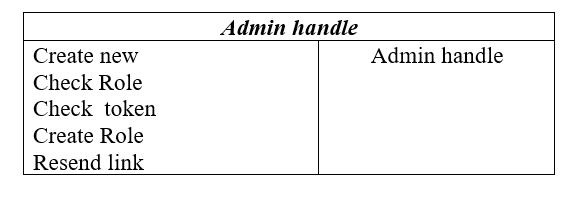


## Controller



## UI





## Data Access

## Design Class

## A diagram of a computer Description automatically generatedOOD with Communication

A diagram with text and arrows

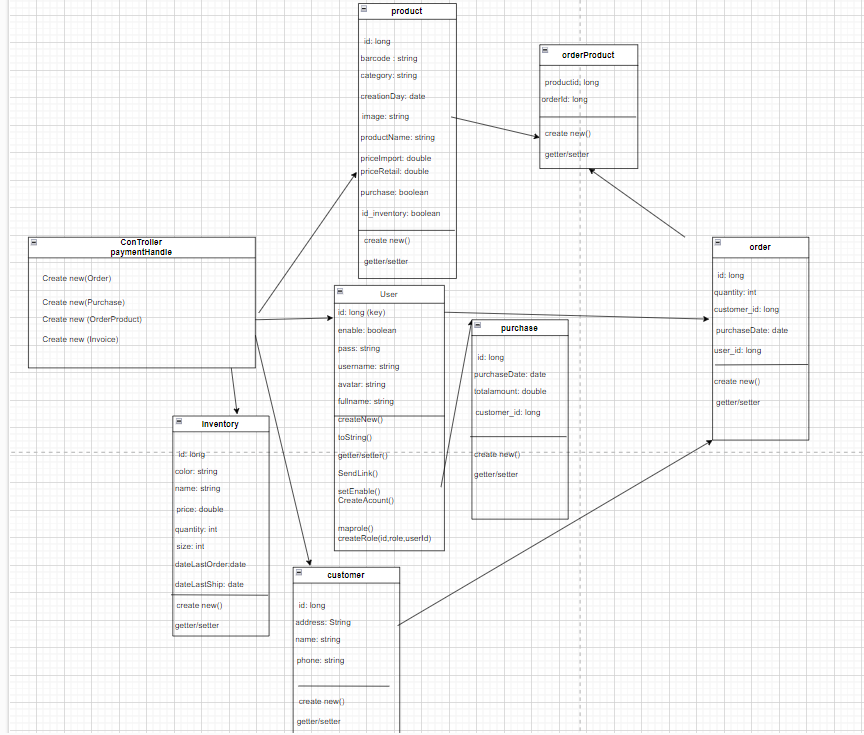
Description automatically generated

## OOD with Sequence Diagram

A diagram of a company

Description automatically generated

## Final Design Class Diagram

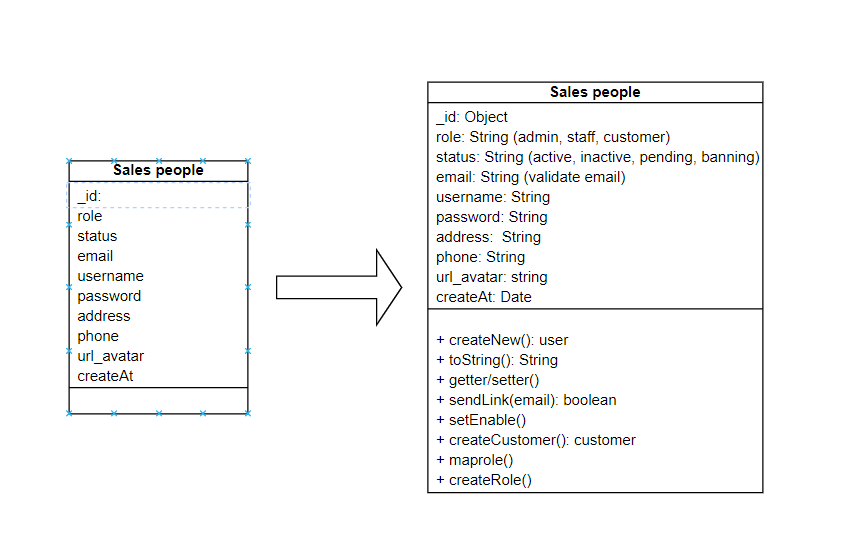


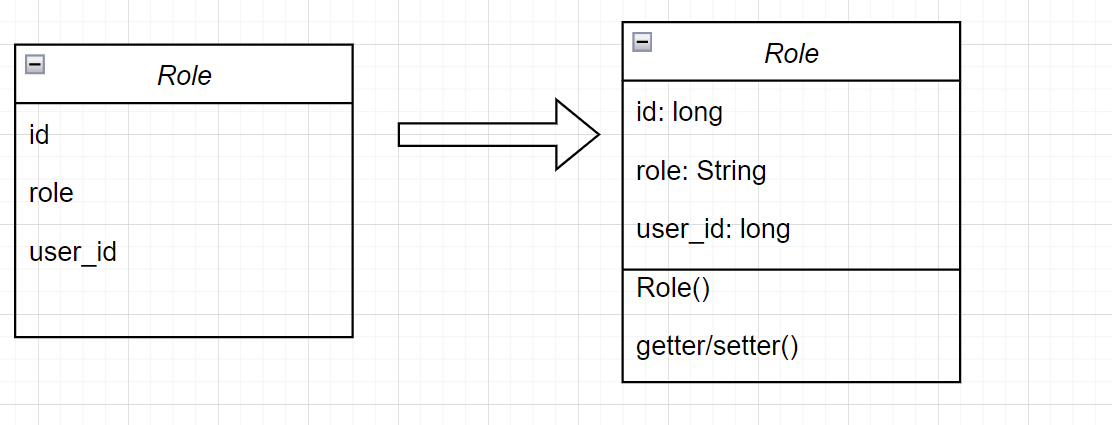
# Design Class for Use Case 2

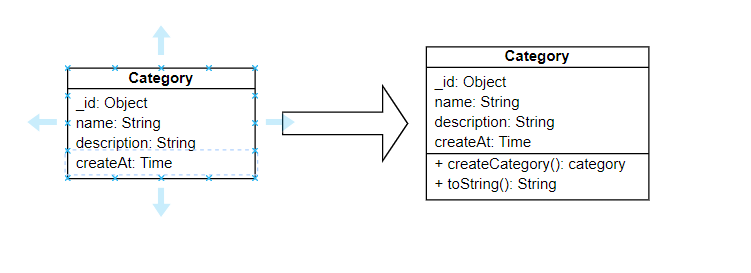
## Design Classes in Detailed Design

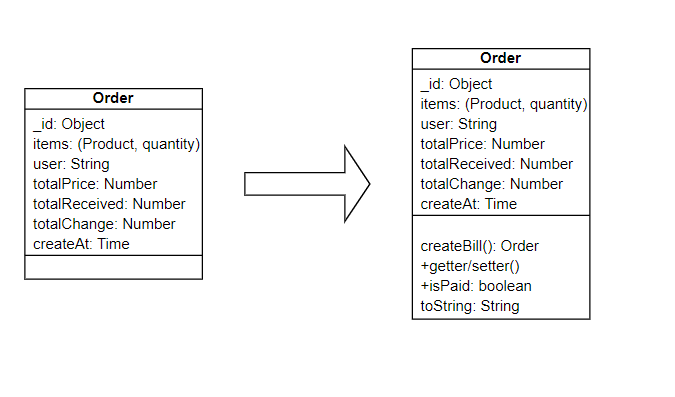
## Design Class Diagram

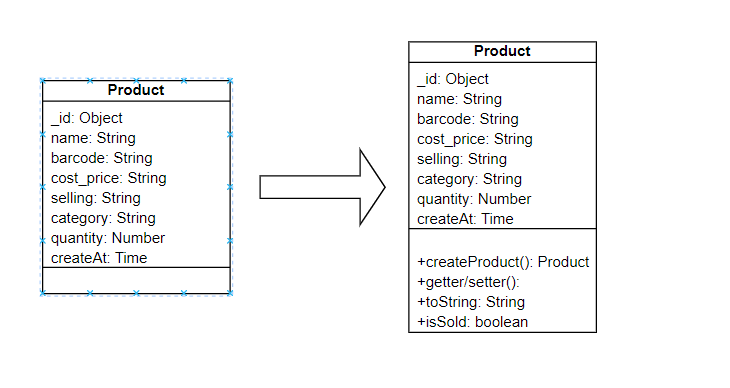
## Domain Design Class

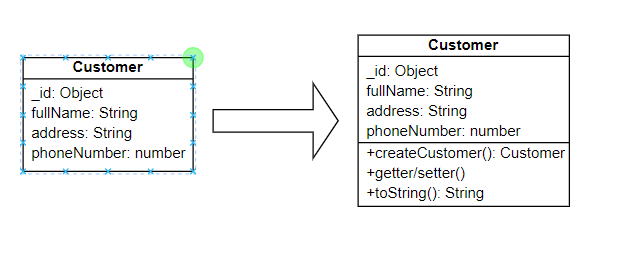












## Controller



## UI

A screenshot of a computer screen

Description automatically generated

## Data Access

## Design Class

A diagram of a computer

Description automatically generated

## OOD with Communication

A diagram with text and arrows

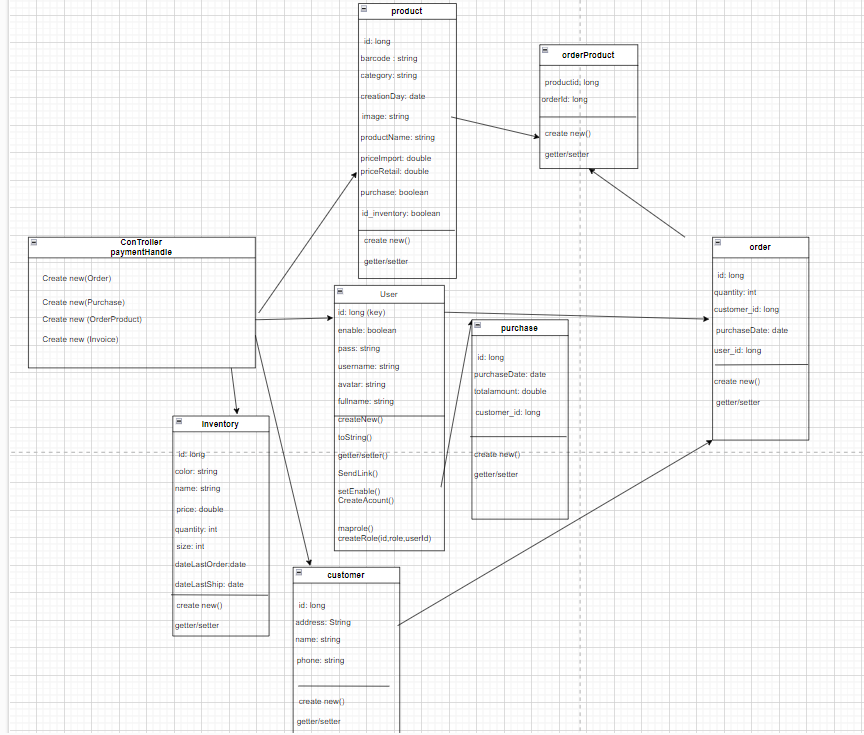
Description automatically generated

## OOD with Sequence Diagram

A diagram of a company

Description automatically generated

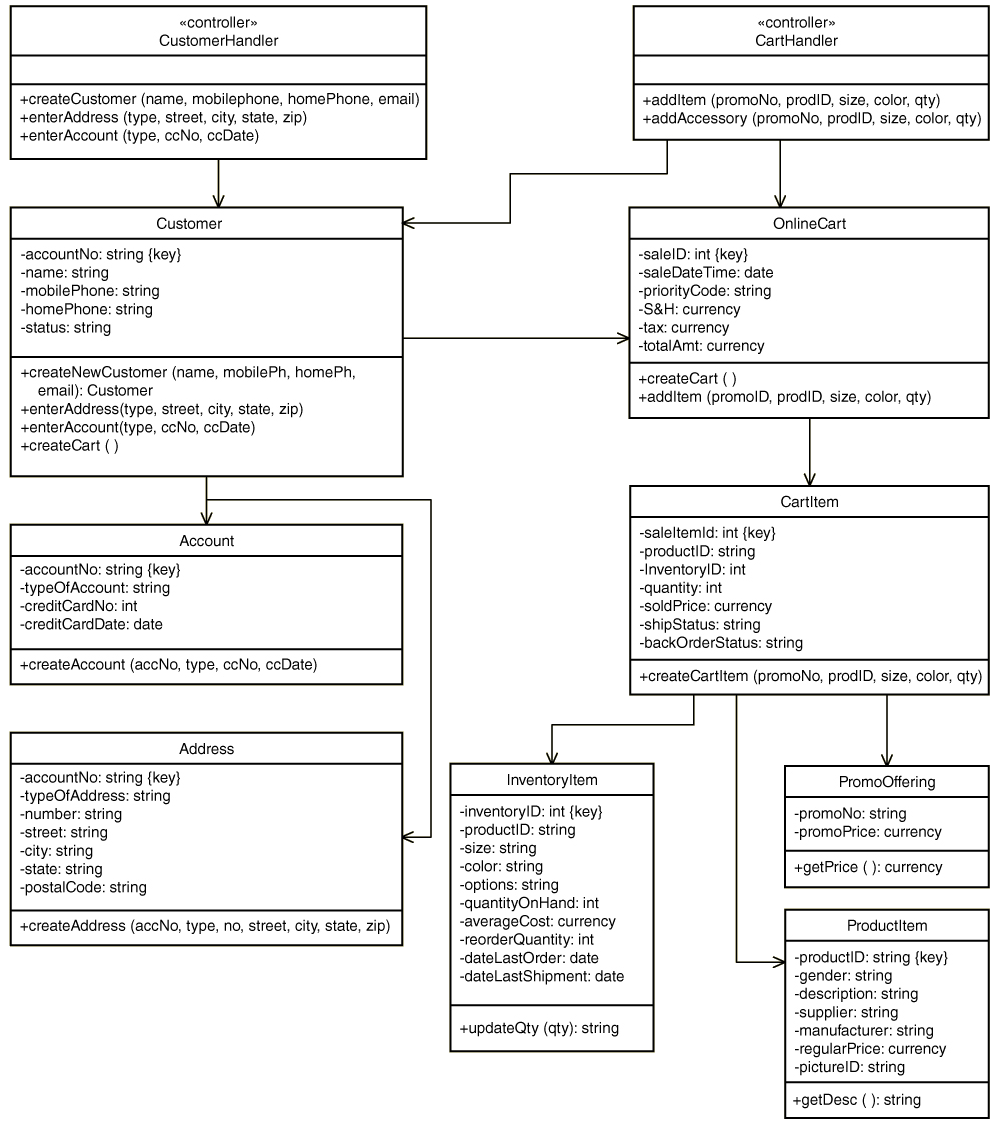
## Final Design Class Diagram



# System Requirements Implementation (2pts)

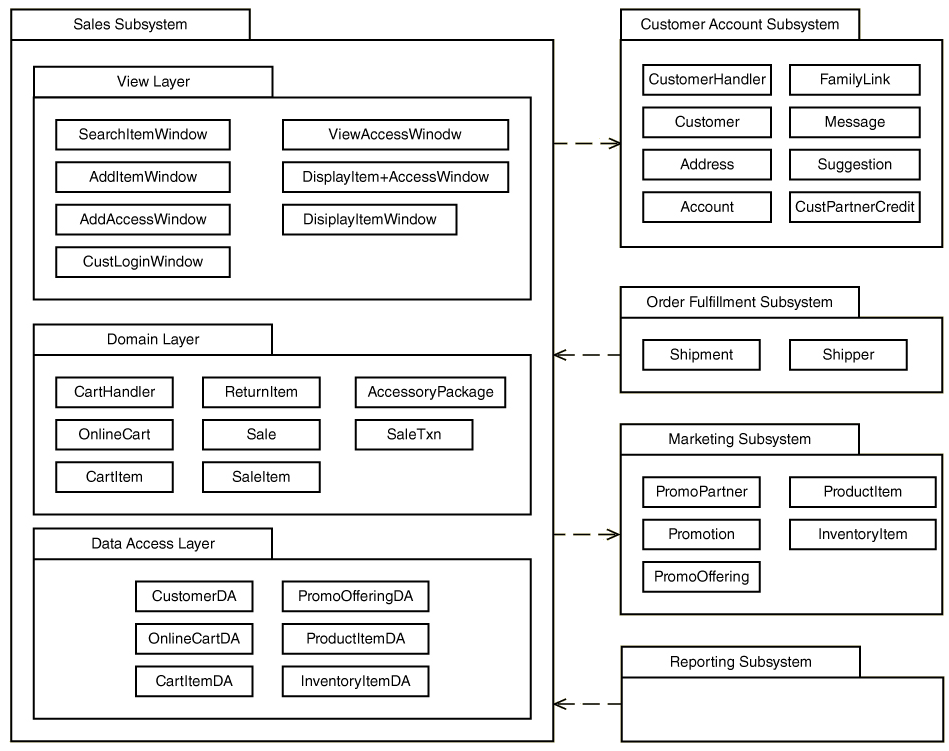
# Design Class for Sub System

Design Class for your Sub-System include many “many use cases”



# Package Diagram for Sub System

Develop architecture of your sub-system by package diagram



# Implementation

## Map persistent objects to the tables in a database

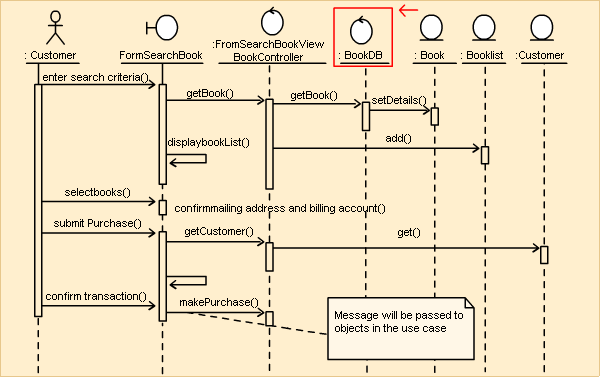
Refer to chapter 9 – Database Design

This is a sample of how to map persistent objects to the tables in a database.

|  |
| --- |
|  |
|  |

## Modifying sequence diagrams

* The database framework design by the architect indicates that we have to code the SQL statements to access the database tables. All classes access the database tables through the corresponding DB object.
* This is a sample of modifying sequence diagram after adding a database object (Book DB)



## UI design

Design prototype of your sub-system

## SQL Code

1. Create Database



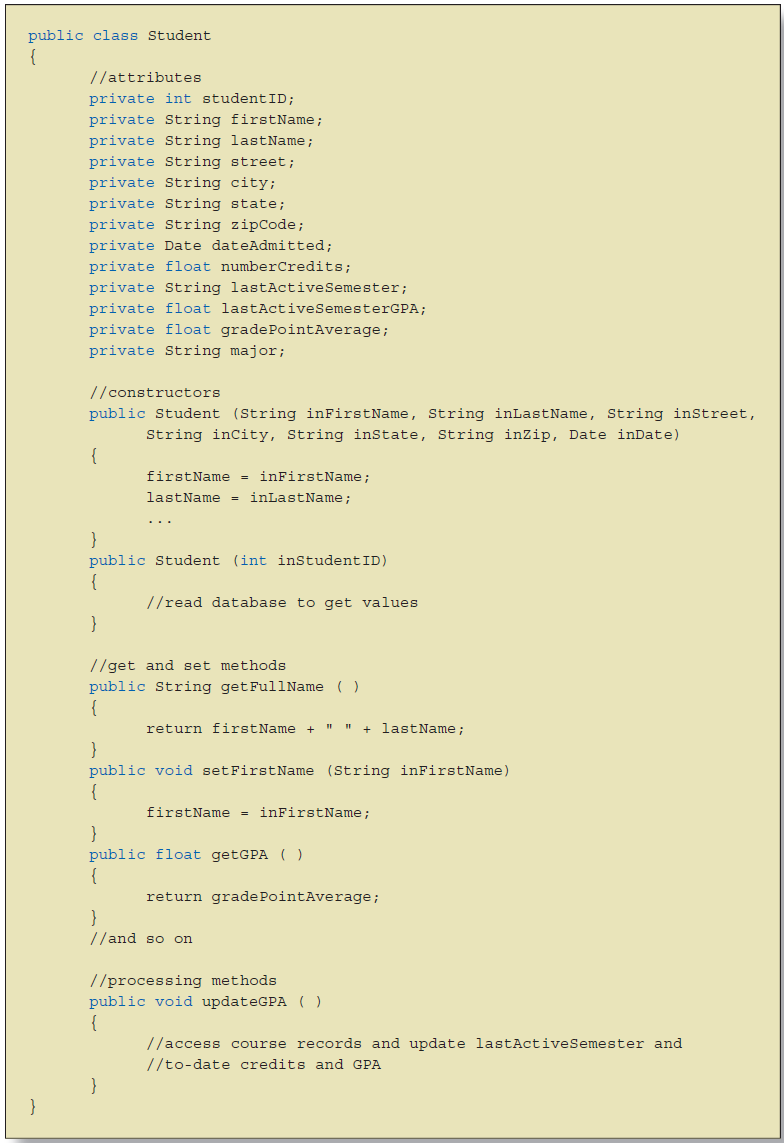
1. Create Table

* Account:
* const mongoose = require('mongoose')
* const Schema = mongoose.Schema
* const AccountSchema = new Schema({
* fullname: { type: String, unique: true },
* email: { type: String, unique: true },
* username: { type: String, unique: true },
* password: String,
* phone: String,
* role: String,
* isActive: Boolean,
* status: String,
* url\_avatar: String,
* isFirst: Boolean
* })
* module.exports = mongoose.model('Account', AccountSchema)
* Cart:
* const mongoose = require('mongoose')
* const Schema = mongoose.Schema
* const CartSchema = new Schema({
* employeeName: String,
* productBarcode: String,
* productName: String,
* price: Number,
* quantity: Number,
* totalPrice: Number,
* url\_image: String
* })
* module.exports = mongoose.model('Cart', CartSchema)
* Customer:
* const mongoose = require('mongoose')
* const Schema = mongoose.Schema
* const CustomerSchema = new Schema({
* name: String,
* phone: { type: Number, unique: true },
* address: String
* })
* module.exports = mongoose.model('Customer', CustomerSchema)
* Orders:
* const mongoose = require('mongoose')
* const Schema = mongoose.Schema
* const OrderSchema = new Schema({
* employeeName: String,
* customerPhone: String,
* customerName: String,
* customerAddress: String,
* totalQuantity: Number,
* totalPrice: Number,
* received: Number,
* refunds: Number,
* creation\_date: String
* })
* module.exports = mongoose.model('Order', OrderSchema)
* const mongoose = require('mongoose')
* Product
* const Schema = mongoose.Schema
* const ProductSchema = new Schema({
* barcode: { type: String, unique: true },
* name: { type: String, unique: true },
* import\_price: Number,
* retail\_price: Number,
* category: String,
* creation\_date: String,
* url\_image: String
* })
* module.exports = mongoose.model('Product', ProductSchema)

## Software Classes Method Code

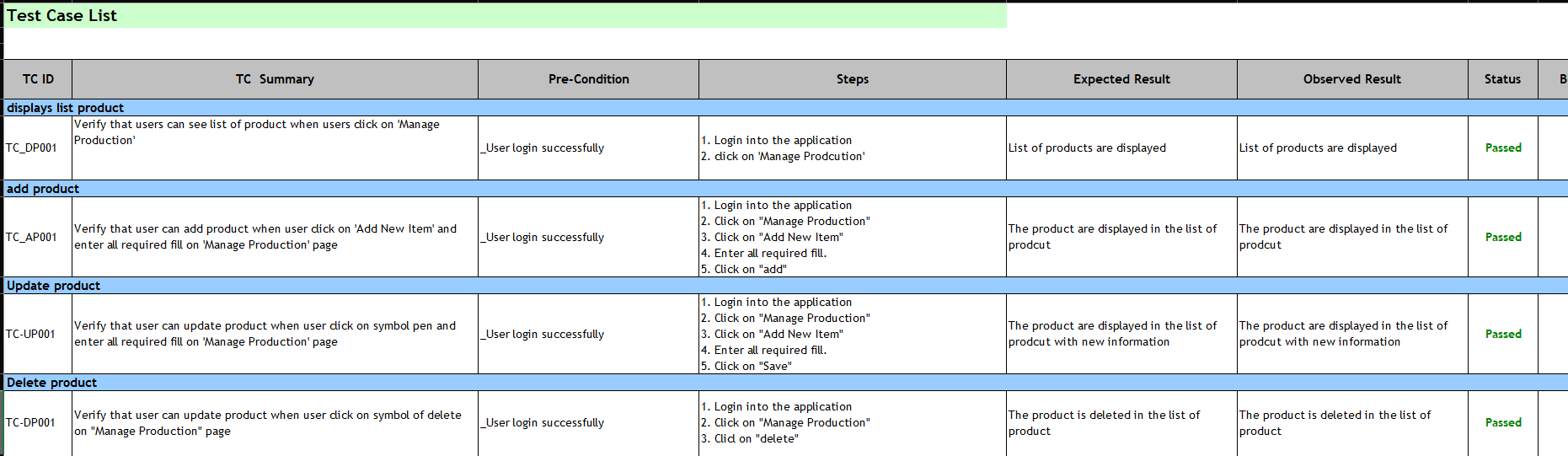
1. C#, VB.NET, JAVA

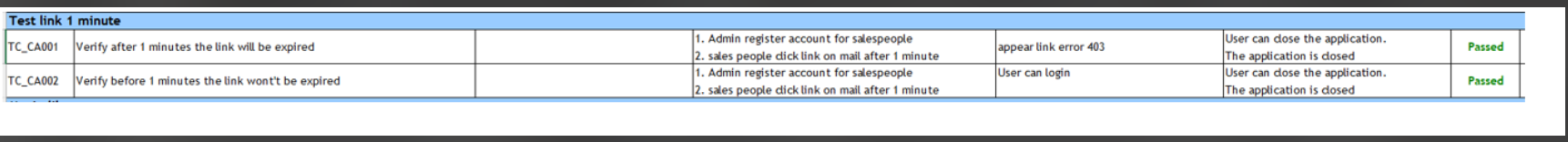
**Write the Code for the Design Class to Implement**



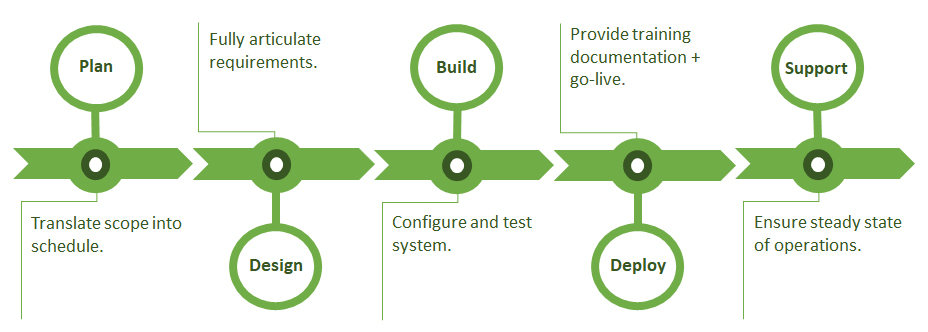
# SYSTEM TESTING, DEPLOYMENT AND DEMONSTRATION (1.5 pts)

# Testing: Test plan & Test case





# Deployment



# Demonstration

Demo your sub-system: can be a prototype with steps how to use your sub-system or your developed software for sub-system

# Conclusions/ Recommendations (0.25 point)

This system is just a functional process without complicated process so far. The identified actors play an important role to the full functioning of the business,….

# References

1. *Slide* ***Requirements Modelling***
2. *Book-MultiAuthor, Gra*

***Internet Link:***

1. *<http://www.uml-dox.net/Addison.Wesley-UML.for.Mere.Mo/0321246241/ch02lev1sec6.html>*

# Appendix

Please see next page

***Appendix 1:*** ***the Full size of use case Diagram***

***Appendix 2: the full size of Collaboration Diagram***

***Appendix 3: the full size of Class Diagram***