DB Final Project Proposal
General Store Management System

Session: 2022 – 2026

Submitted by (G3):
Tabish Akhtar 2022-CS-78
Saqlain Mansab 2022-CS-80
Zulqarnain Akram 2022-CS-84

Supervised by:
Mr. Nazeef Ul Haq

Department of Computer Science
University of Engineering and Technology
Lahore Pakistan

Table of Contents

Project Description:

Our project aims to revolutionize the management of general shops, simplifying tasks such as inventory tracking, sales monitoring, and customer, supplier, and employee management. The main authority is given to the admin, who utilizes the management system to oversee and manage employees, products, and suppliers for their shop. The admin has the capability to define product categories and sub-categories tailored to the shop's offerings. Moreover, the admin can implement discounts through the issuance of coupon codes for specific products, enhancing promotional activities and more sale.

Employees also play a crucial role in the system, with the authority to add customers and suppliers to the database. When a customer visits the shop, employees promptly enter their information into the system and record the products purchased. Similarly, when a supplier comes to the shop, employees add their data and any products purchased from them. The system accommodates both cash payments and product borrowing options for customers, providing flexibility in transaction methods.

Additionally, the admin has the capability to manage employee attendance efficiently. Through the system, the admin can mark employee attendance, ensuring accurate records of staff presence. Furthermore, the system automatically logs the time of employee logins, maintaining a comprehensive record of system usage for record-keeping purposes.

Furthermore, our project incorporates advanced reporting and analytics features to provide valuable insights into shop operations. Through customizable reports and dashboards, shop owners can gain a deeper understanding of sales trends, customer behavior, and product performance. This data-driven approach empowers shop owners to make informed decisions and optimize business strategies for growth and success.

Moreover, our system prioritizes user-friendly design and intuitive navigation, ensuring ease of use for both admin and employees. With a simple and intuitive interface, users can quickly access key features and functionalities, enhancing efficiency and productivity in day-to-day operations.

In conclusion, our project represents a comprehensive solution for general shop management, integrating essential features and functionalities to streamline operations, enhance productivity, and drive business growth.

Project Features:

Here are project features available to the user of application.

Admin:

Manage Employees:

Add, update, and remove employee details, including its name, profession, salary and contact information.

Manage Customers:

Add, update, and remove employee details, including its name, address and contact information.

Manage Categories:

Add, update, and remove products categories and sub categories of products.

Manage Products:

Add, update and remove products for shop by selecting their sub category and other information.

Manage Suppliers:

Add, update and remove supplier details including its name, address, description and contact information.

Discount Coupons:

Set up and manage discount coupons for specific products to attract customers.

Attendance Tracking:

Monitor employee attendance and manage work hours for payroll purposes.

Reporting and Analytics:

Access comprehensive reports and analytics to track sales, inventory, and employee performance.

Employees:

Manage Customers:

Add, update, and remove employee details, including its name, address and contact information.

Manage Suppliers:

Add, update and remove supplier details including its name, address, description and contact information.

Place Order:

Employees can place order by entering product id, customer id , quantity and coupon code(optional) for customers.

Purchase Products:

Employees can purchase products from suppliers by entering their details and product details for shop.

Payment Options:

Accept cash payments and manage product borrowing for customers.

Technology Stack:

Language (C#/Java)	C#
Platform (Web/Desktop)	Desktop
Frontend Technology	C# Window Forms
IDEs	Visual Studio Community 2022, SSMS 2022

Project Actors:

The project actors who will use this system are:

Admin:

The admin is main authority of the shop management system. Admin oversee all the customer, employees, products, products categories discount coupon. Admin can also see specific reports like stock available, most wanted products, less selling products, employee attendance, customers preferences. This comprehensive oversight empowers the admin to make informed decisions, optimize business strategies, and drive success within the shop environment.

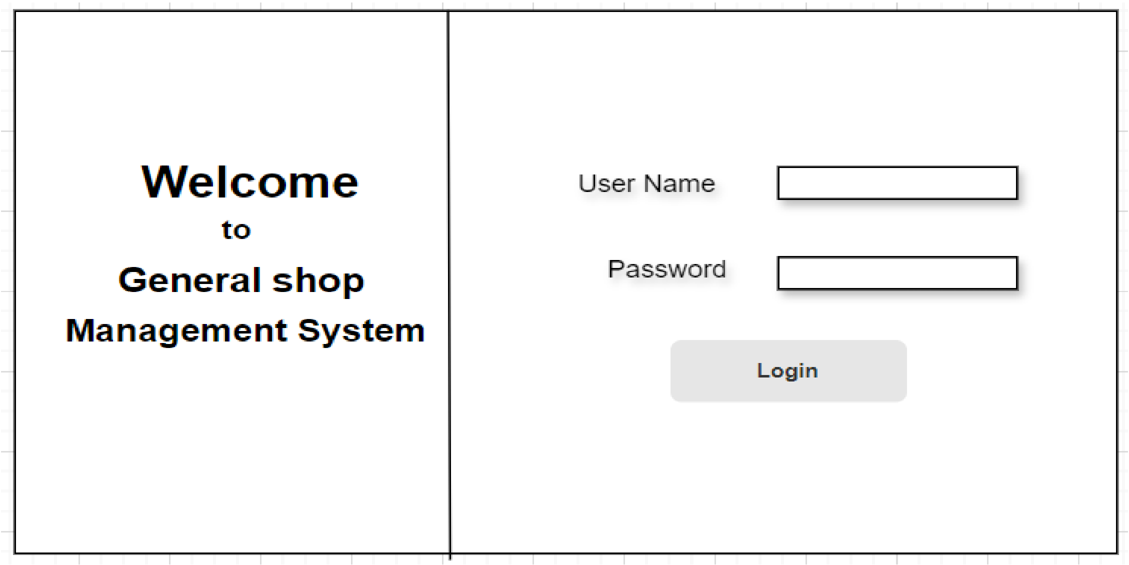
Employee:

The employee plays a vital role in the shop management system, being responsible for various tasks. They handle customer details, including adding, updating, or removing information such as name, address, and contact details. Similarly, employees manage supplier details, ensuring accuracy in name, address, description, and contact information. When processing customer orders, employees input essential details like product ID, customer ID, quantity, and, if applicable, a coupon code. Additionally, they handle the purchasing process by entering supplier and product details to replenish shop inventory. Lastly, employees facilitate transactions by accepting cash payments and managing product borrowing for customers.

Use Cases:

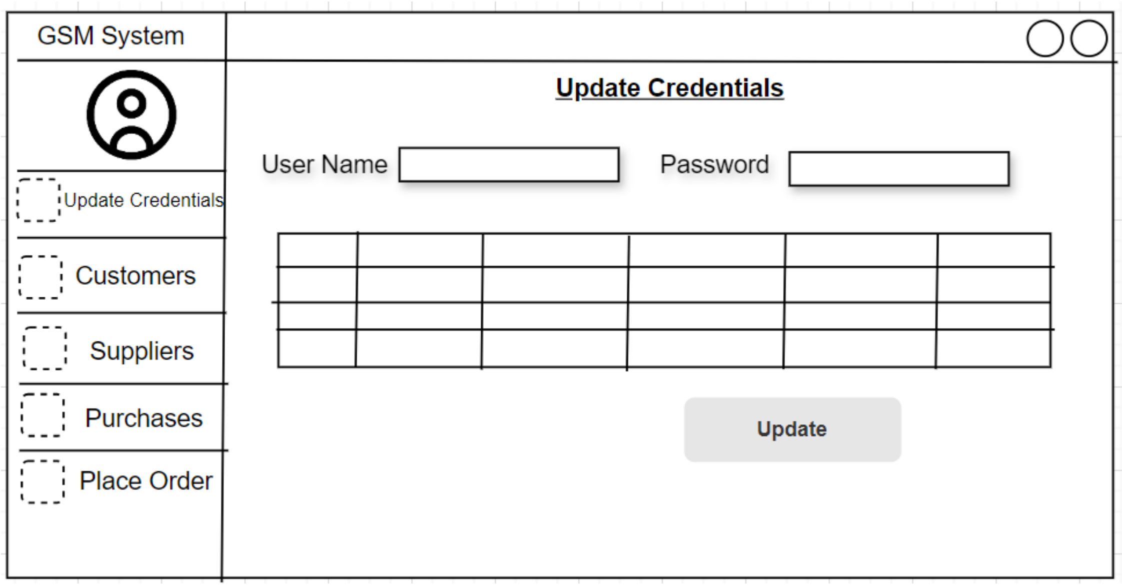
All the use cases should be written in the following format. The name of use should be start with a word e.g. Add Student. It should not be as Student/ Student Add. Add Student is separate use case and Edit Student is separate use case:

Use Case 1(Login):

Use Case ID	U01
Name	Login
Actor	Admin, Employee
Description	**Main Flow:** • Admin or Employee opens the application. • User enters valid User name and password • When clicking on “Login button”, then system ensures that information entered is correct. • If entered information is correct then system open dashboard of admin or employee accordingly. **Alternate Flow:** • If any from User Name or password box is left blanked then system prompts and error message. • After clicking on Login button, if entered information is wrong then system prompts an error message of invalid data entered. • If an employee forgot his user name or password then employee will contact to admin for assistance. • If an admin forgot his credential then admin will contact developer of application for assistance.
Layout in pencil tool	 Welcome to General shop Management System User Name Password Login
Validators	• User Name and Password Box accepts strings up to 30 characters and 20 characters respectively and not Null. • "The 'Login' button functionality is designed to activate only when all mandatory fields are completed with accurate information.

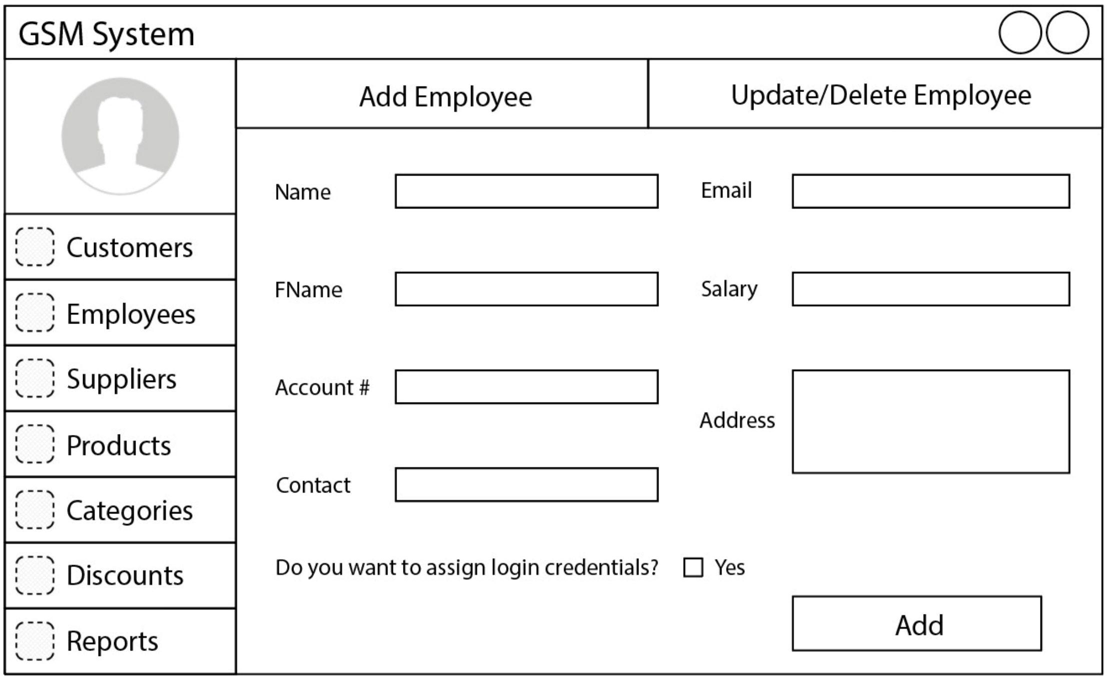
Use Case 2(Update Credentials):

Use Case ID	U02
Name	Update Credentials
Actor	Admin
Description	**Main Flow:** • Admin or Employee opens the application. • User enters valid User name and password • When clicking on “Login button”, then system ensures that information entered is correct. • If entered information is correct then system open dashboard of admin or employee accordingly.

	Alternate Flow: • If any from User Name or password box is left blanked then system prompts and error message. • After clicking on Login button, if entered information is wrong then system prompts an error message of invalid data entered. • If an employee forgot his user name or password then employee will contact to admin for assistance. • If an admin forgot his credentials then admin will contact developer of application for assistance.
Layout in pencil tool	 GSM System Update Credentials Customers Suppliers Purchases Place Order Update Credentials User Name Password Update
Validators	• User Name and Password Box accepts strings up to 30 characters and 20 characters respectively and not Null. • The first row in data grid is admin credentials and other next are employee credentials. • Admin clicks on data grid then that specific row comes in user name and password text boxes • "The 'Login' button functionality is designed to activate only when all mandatory fields are completed with accurate information.

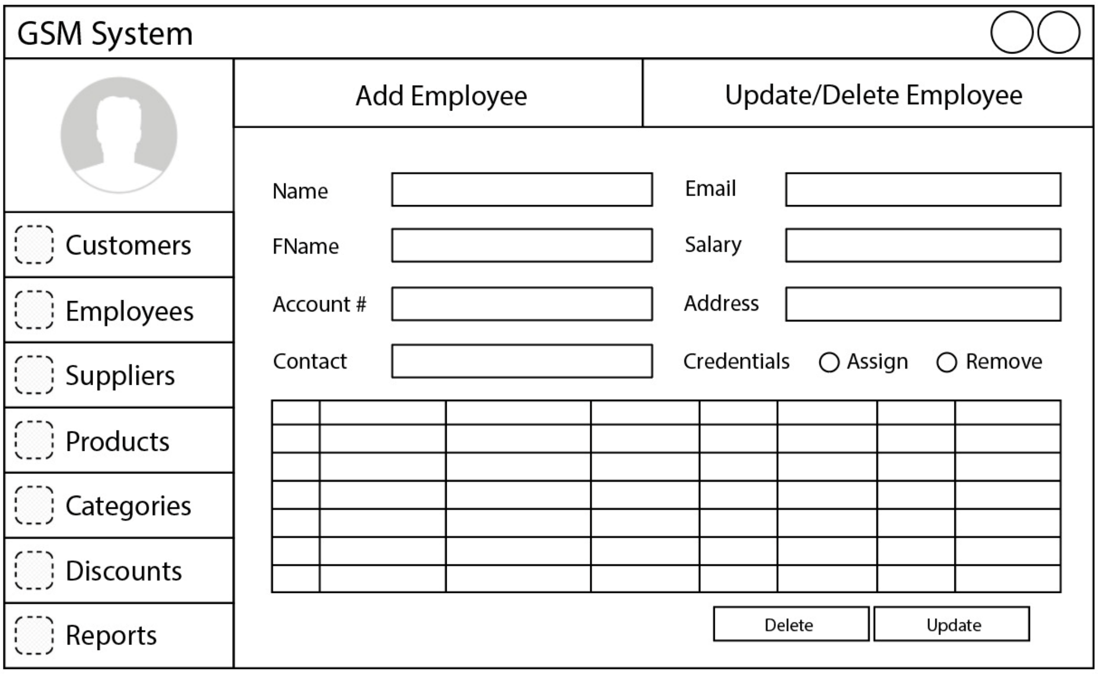
Use Case 3(Add Employees):

Use Case ID	U03
Name	Add Employees
Actor	Admin
Description	**Main Flow:** • The admin navigates to the "Add Employees" section of the shop system interface. • The admin fills in the necessary details for the new employee, including name, email, and salary. The system provides clear instructions and designated input fields for each piece of information. • Upon submission, the system validates the entered information to ensure all required fields are filled out correctly. It checks for data format (e.g., valid email format) and completeness. • If all information is entered correctly, the system generates a unique ID for the new employee. This ID serves as a reference for the employee within the system. • The system adds the new employee along with their details and generated ID to the database. • A confirmation message is displayed to the admin, indicating that the employee has been successfully added to the system. • The admin can then proceed with other actions within the shop system interface. **Alternate Flow:** • If the admin enters invalid information or leaves required fields blank, the system prompts them with error messages indicating the specific issues.

	• If the system detects that the employee being added already exists in the system (based on unique identifiers like email or employee ID), it prompts the admin with a message indicating the duplication. • The admin can choose to either cancel the addition or proceed with entering new employee's information.
Layout in pencil tool	 GSM System Add Employee Update/Delete Employee Customers Employees Suppliers Products Categories Discounts Reports Name Email FName Salary Account # Address Contact Do you want to assign login credentials? □ Yes Add
Validators	• Name and FName Box accepts strings up to 50 characters and not Null. • Email box only accept string of email address format (e.g., example@gmail.com) and string up to 60 characters. • Salary box accepts an integer of not more than 18 characters with 2 decimal places and not null. • Account # box accepts an integer of not more than 24 characters and not null. • Address box accepts a string not more than 255 characters and may be null. • Contact box accepts an integer not more than 11 characters and not null. • User can select or un select credentials check box. If admin checks this then the name of employee is assign as user Name and Account# of employee is assign as password to employee. • "The 'Add' button functionality is designed to activate only when all mandatory fields are completed with accurate information.

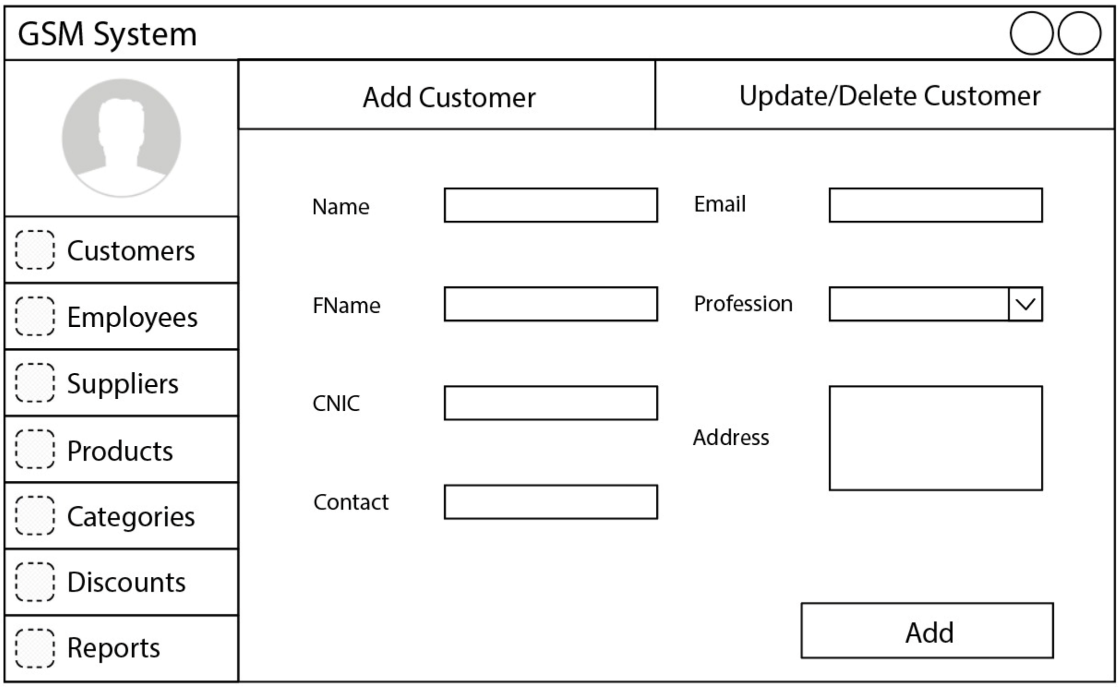
Use Case 4(Edit Employees):

Use Case ID	U04
Name	Edit Employees
Actor	Admin
Description	**Main Flow:** • The admin accesses the "Edit Employees" feature from the admin dashboard or menu. • The admin selects the employee whose details they want to modify from the list of employees displayed. • The admin makes the necessary changes to the employee's details, such as name, email, or salary, in the provided input fields. • The system validates the modified information to ensure it meets the required criteria and is error-free.

	• If the modified information passes validation, the admin saves the changes by clicking the "Save" or "Update" button. • The system confirms that the changes have been successfully saved and updates the employee's record in the database. **Alternate Flow:** • If the modified information fails validation (e.g., invalid email format, salary below minimum threshold), the system displays error messages indicating the specific issues. • The admin corrects the errors as indicated by the system and resubmits the employee details. • The system revalidates the modified information after corrections. • If the modified information passes validation, the admin saves the changes as described in the main flow. **Deletion Flow:** • The admin selects the employee they want to delete from the list of employees displayed. • The system prompts the admin to confirm the deletion by displaying a confirmation dialog box or message. • The admin confirms the deletion, indicating that it's intentional. • Upon confirmation, the system permanently removes the employee's record, including all associated information, from the database. • The system confirms that the employee has been successfully deleted.
Layout in pencil tool	 GSM System Add Employee Update/Delete Employee Customers Employees Suppliers Products Categories Discounts Reports Name Email FName Salary Account # Address Contact Credentials ○ Assign ○ Remove Delete Update
Validators	• In Data Grid Admin can click on any row, and the data from that row will appear in the designated boxes. They can then modify this information. • Name and FName Box accepts strings up to 50 characters and not Null. • Email box only accept string of email address format (e.g., example@gmail.com) and string up to 60 characters. • Salary box accepts an integer of not more than 18 characters with 2 decimal places and not null. • Account # box accepts an integer of not more than 24 characters and not null. • Address box accepts a string not more than 255 characters and may be null. • Contact box accepts an integer not more than 11 characters and not null. • user can select or un select credentials check box. Admin can remove or assign employees credentials. • The UPDATE button will update the information of an employee if their current ID exists in the database. • The Delete button will update the information of an employee if their current ID exists in the database. • Before deleting an employee, the system should check for any dependencies or associated data, such as assigned tasks or transactions. If dependencies exist, the system should prompt the user to resolve them before proceeding with the deletion.

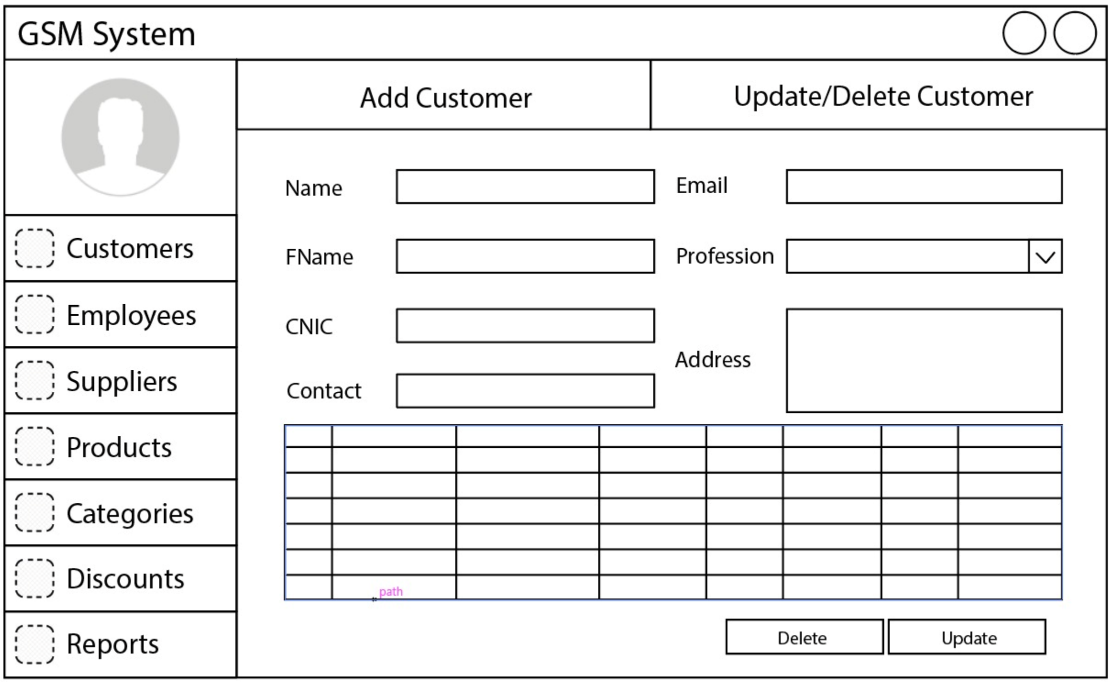
	• When Admin removes an employee then it’s all data like timestamp and information remains on database but will not display on list of employees. • When user selects a row from data grid view and click on delete button then a confirmation prompt appears on screen with OK button. This helps prevent accidental deletion.

Use Case 5(Add Customers):

Use Case ID	U05
Name	Add Customers
Actor	Admin, Employees
Description	**Main Flow:** • The admin navigates to the "Add Customers" section of the shop system interface. • The admin fills in the necessary details for the new customer, including name, email, contact information, profession, CNIC, and address. The system provides clear instructions and designated input fields for each piece of information. • If all information is entered correctly, the system generates a unique ID for the new customer. This ID serves as a reference for the customer within the system. • A confirmation message is displayed to the admin, indicating that the customer has been successfully added to the system. • The admin can then proceed with other actions within the shop system interface. **Alternate Flow:** • If the admin enters invalid information or leaves required fields blank, the system prompts them with error messages indicating the specific issues. • If the system detects that the customer being added already exists in the system (based on unique identifiers like email or customer ID), it prompts the admin with a message indicating the duplication. • The admin can choose to either cancel the addition or proceed with entering the new customer's information. • If errors are present, the admin corrects the provided information based on the system prompts. • If validation passes, the admin saves the details. • Upon successful addition, a confirmation message is displayed to the admin.
Layout in pencil tool	 GSM System Add Customer Update/Delete Customer Customers Employees Suppliers Products Categories Discounts Reports Name Email FName Profession CNIC Address Contact Add
Validators	• Name, Profession and FName boxes accept strings of up to 50 characters in length and cannot be null (empty). • Email box only accept string of email address format (e.g., example@gmail.com) and string up to 60 characters.

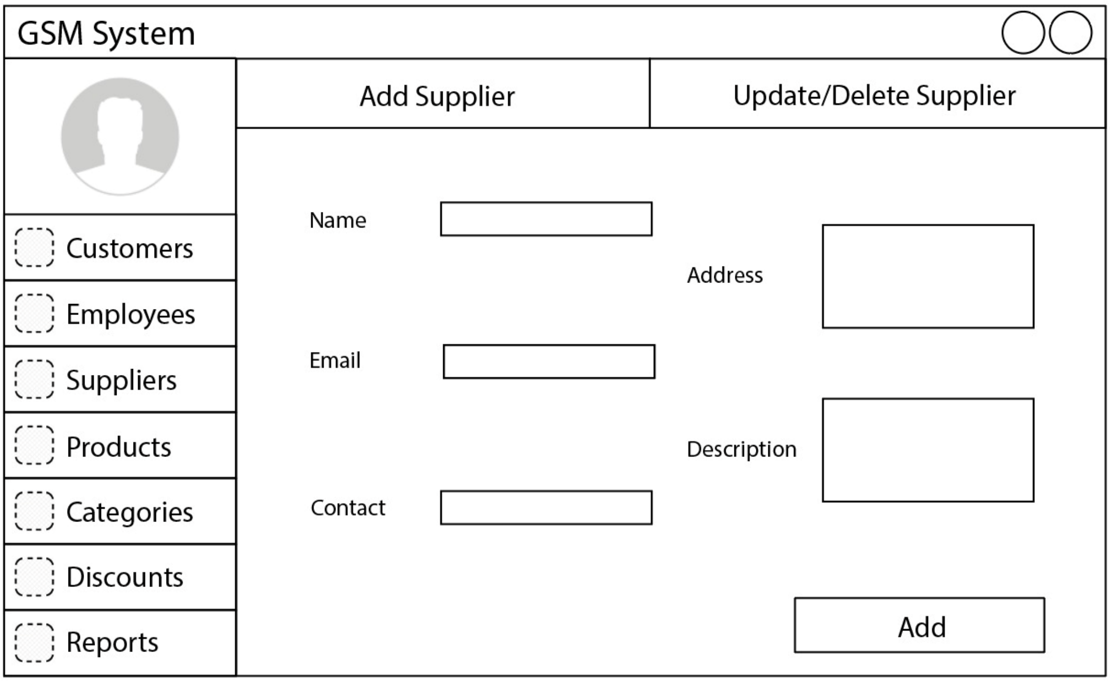
	• Address box accepts a string not more than 255 characters and may be null. • Contact box accepts an integer not more than 11 characters and not null. • CNIC box accepts an integer of length 13 digits. • "The 'Add' button functionality is designed to activate only when all mandatory fields are completed with accurate information.

Use Case 6(Edit Customers):

Use Case ID	U06
Name	Edit Customers
Actor	Admin, Employees
Description	**Main Flow:** • The user navigates to the "Edit Customers" section of the shop system interface. • The user selects a customer from the list whose details need to be modified. • The user changes the necessary information for the selected customer, including name, email, contact information, profession, CNIC, and address details. • A confirmation message is displayed to the user, indicating that the changes have been successfully saved. • Optionally, the user can choose to delete the selected customer by confirming the deletion. • The system double-checks with the user to ensure that the deletion is intentional. • If confirmed, the customer's record, including all associated details, is permanently removed from the database. • A confirmation message is displayed to the user, indicating that the customer has been successfully deleted. **Alternate Flow:** • If the user enters invalid information or leaves required fields blank while editing customer details, the system prompts them with error messages indicating the specific issues. • The user corrects the provided information based on the system prompts. • After corrections are made, the system re-validates the entered data. • If validation passes, the user saves the changes. • At any point during the editing process, the user can choose to cancel and exit the editing interface without saving changes.
Layout in pencil tool	 GSM System Add Customer Update/Delete Customer Customers Employees Suppliers Products Categories Discounts Reports Name Email FName Profession CNIC Address Contact path Delete Update
Validators	• Name, Profession and FName boxes accept strings of up to 50 characters in length and cannot be null (empty). • Email box only accept string of email address format (e.g., example@gmail.com) and string up to 60 characters. • Address box accepts a string not more than 255 characters and may be null. • Contact box accepts an integer not more than 11 characters and not null.

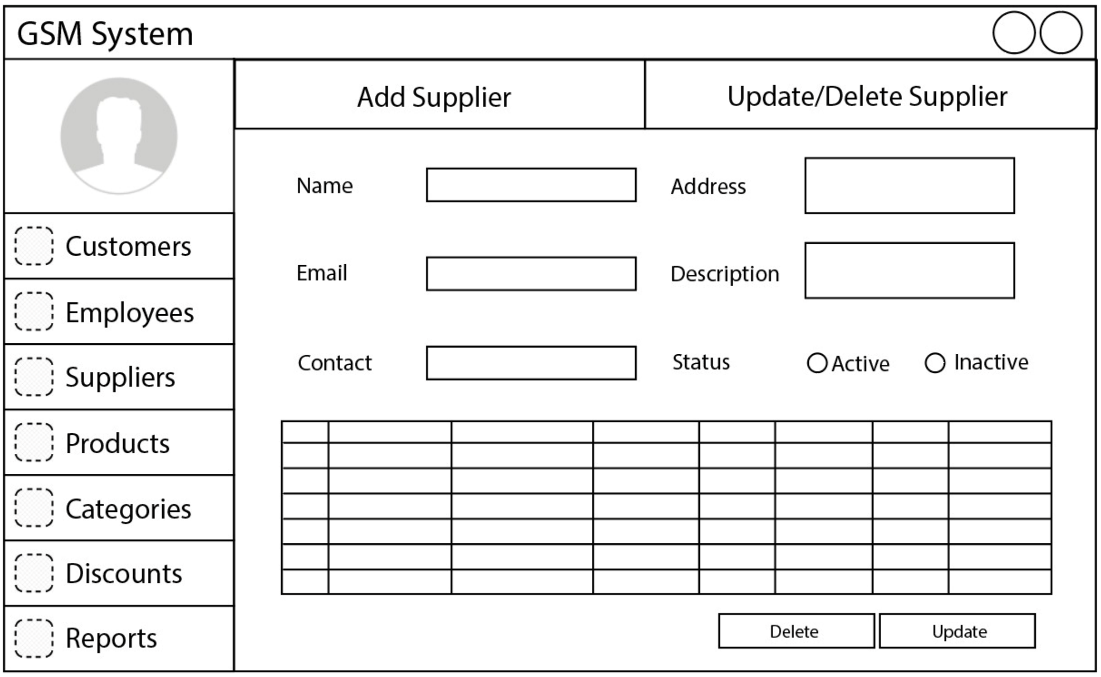
	• CNIC box accepts an integer of length 13 digits. • When delete button clicked then it checks the selected row of customer from data grid and only deletes if current ID exists in database. • The Update button will update the information of an employee if their current ID exists in the database.

Use Case 7(Add Supplier):

Use Case ID	U07
Name	Add Supplier
Actor	Admin, Employees
Description	**Main Flow:** • The user navigates to the "Add Supplier" section of the shop management system interface. • The user enters relevant details for the new supplier, including name, email, contact information, description, and address, into the designated input fields. • The system validates the entered information to ensure all required fields are filled out correctly and that the data format is accurate (e.g., valid email format). • Upon successful validation, the system generates a unique ID for the new supplier. This ID serves as a reference for the supplier within the system. • The system adds the new supplier along with their details and generated ID to the database. • A confirmation message is displayed to the user, indicating that the supplier has been successfully added to the system. • When the supplier visits the shop to sell products, their unique supplier ID is used for product purchases, allowing for easy tracking and management of supplier transactions. **Alternate Flow:** • If the user enters invalid information or leaves required fields blank while adding a new supplier, the system prompts them with error messages indicating the specific issues. • The user corrects the provided information based on the system prompts. • After corrections are made, the system re-validates the entered data. • If validation passes, the user saves the supplier details, and the process continues from step 5 in the main flow. • At any point during the addition process, the user can choose to cancel and exit the interface without saving changes.
Layout in pencil tool	 GSM System Add Supplier Update/Delete Supplier Customers Employees Suppliers Products Categories Discounts Reports Name Address Email Description Contact Add
Validators	• The Name box accept string of up to 50 characters in length and cannot be null (empty). • Email box only accept string of email address format (e.g., example@gmail.com) and string up to 60 characters. • Address box accepts a string not more than 255 characters and may be null. • Contact box accepts an integer not more than 11 characters and not null.

	- Description box accepts a string up to 255 characters and can be null. - The Add button will only add a new supplier if all required fields, such as Name, email, contact, etc., are filled out.

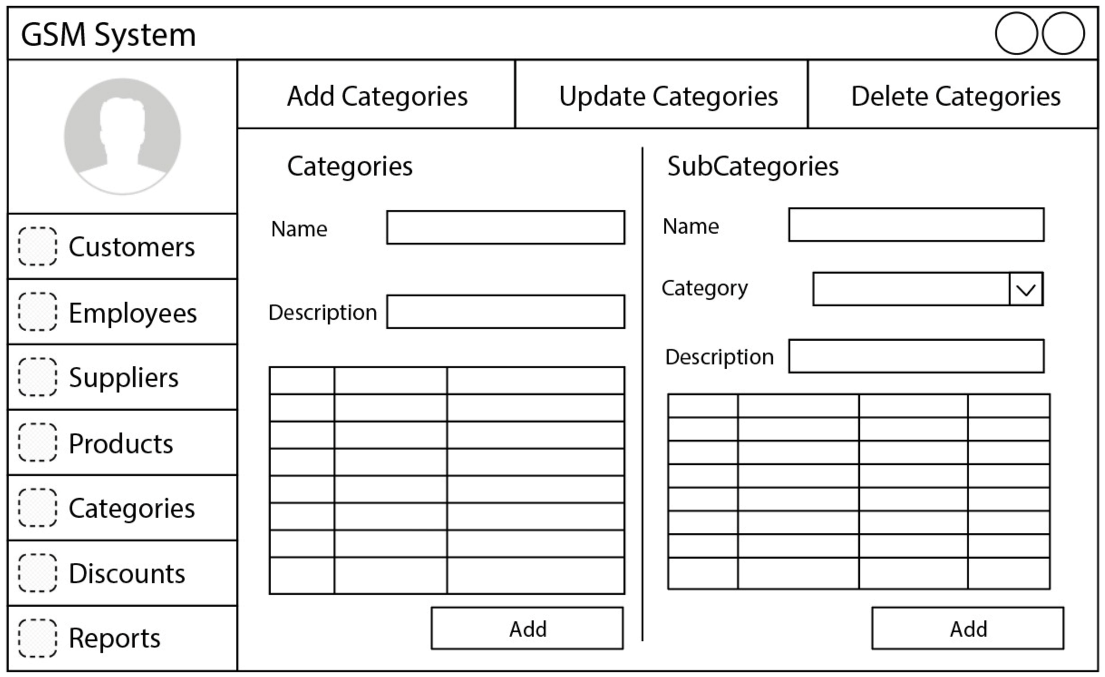
Use Case 8(Edit Supplier):

Use Case ID	U08
Name	Edit Supplier
Actor	Admin, Employees
Description	**Main Flow:** - The user navigates to the "Edit Supplier" section of the shop management system interface. - The user selects a supplier from the list of existing suppliers displayed on the interface. - The user makes changes to the selected supplier's information, such as name, email, contact information, description, and address details, in the designated input fields. - The system updates the supplier's record in the database with the modified information. - A confirmation message is displayed to the user, indicating that the changes have been successfully saved. - If the user chooses to delete the supplier, they select the supplier from the list and confirm the deletion. - The system prompts the user to confirm the deletion, ensuring that it is intentional. - Upon confirmation, the system permanently removes the supplier's record, including all associated details, from the database. - The interface updates to reflect the changes, and the user can proceed with other actions. **Alternate Flow:** - If the user enters invalid information or leaves required fields blank while editing a supplier's details, the system prompts them with error messages indicating the specific issues. - The user corrects the provided information based on the system prompts. - After corrections are made, the system re-validates the entered data. - If validation passes, the user saves the updated supplier details. - At any point during the editing process, the user can choose to cancel and exit the interface without saving changes.
Layout in pencil tool	 GSM System Add Supplier Update/Delete Supplier Customers Employees Suppliers Products Categories Discounts Reports Name Address Email Description Contact Status ○Active ○ Inactive Delete Update
Validators	- The Name box accept string of up to 50 characters in length and cannot be null (empty). - Email box only accept string of email address format (e.g., example@gmail.com) and string up to 60 characters. - Address box accepts a string not more than 255 characters and may be null.

	• Contact box accepts an integer not more than 11 characters and not null. • Description box accepts a string up to 255 characters and can be null. • User can select one of these two status radio buttons either active or in-active. • When delete button clicked then it checks the selected row of customer from data grid and only deletes if current ID exists in database. • The Update button will update the information of a supplier if their current ID exists in the database.

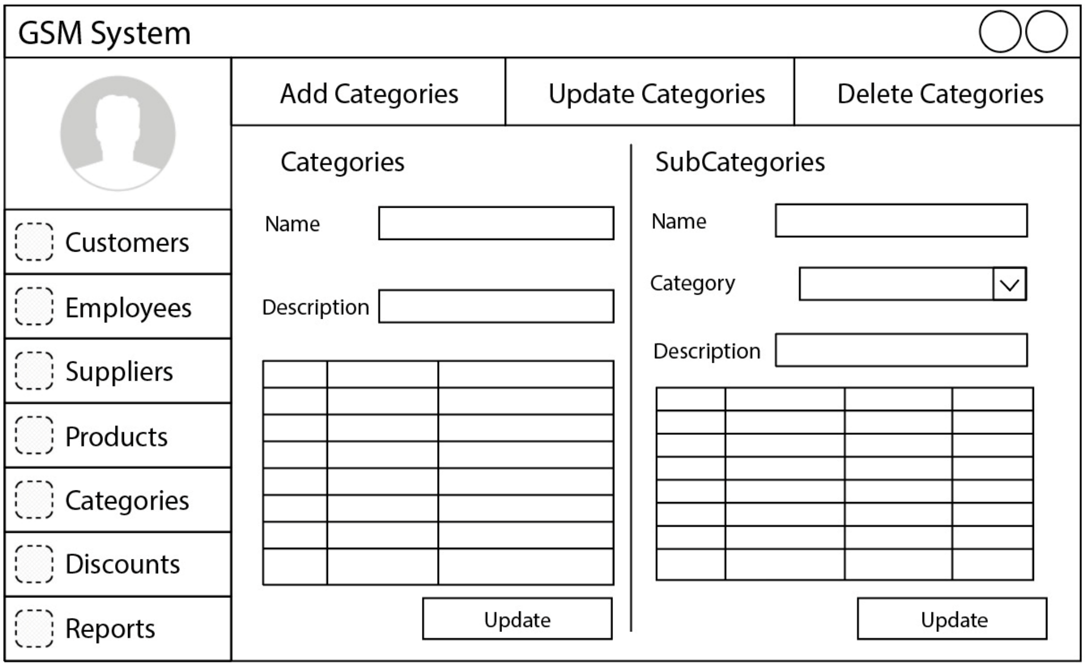
Use Case 9(Add Categories):

Use Case ID	U09
Name	Add Categories
Actor	Admin
Description	**Main Flow:** • The admin navigates to the "Add Categories" section within the shop management system interface. • The system displays a data grid showing the existing categories. • The admin selects a category from the list to which they want to add subcategories. • Upon selecting a category, the system provides options to add subcategories directly on the same screen. • The admin enters the details of the new subcategory, including its name and any additional information required. • After entering the details, the admin saves the new subcategory, and it is added to the selected category. • A confirmation message is displayed, indicating that the subcategory has been successfully added. • The admin can repeat the process to add more subcategories to the same or different categories as needed. • The newly added categories and subcategories serve as organizational tools for managing products effectively. **Alternate Flow:** • If there are no existing categories in the system, the admin is prompted with a message indicating that no categories are available to add subcategories to. • If the admin enters invalid information or leaves required fields blank while adding a subcategory, the system prompts them with error messages indicating the specific issues. • The admin corrects the provided information based on the system prompts. • After corrections are made, the system re-validates the entered data. • If validation passes, the system saves the new subcategory.

Layout in pencil tool	 GSM System Add Categories Update Categories Delete Categories Customers Employees Suppliers Products Categories Discounts Reports Categories Name Description Add SubCategories Name Category Description Add
Validators	• The Name box accept string of up to 50 characters in length and cannot be null (empty). • In Description box user can write any string up to 255 characters and can be null. • Only added categories will be displayed in category combo box. • This button when clicked checks all the required boxes and if these boxes fulfill requirement then the categories or subcategories will be added to database.

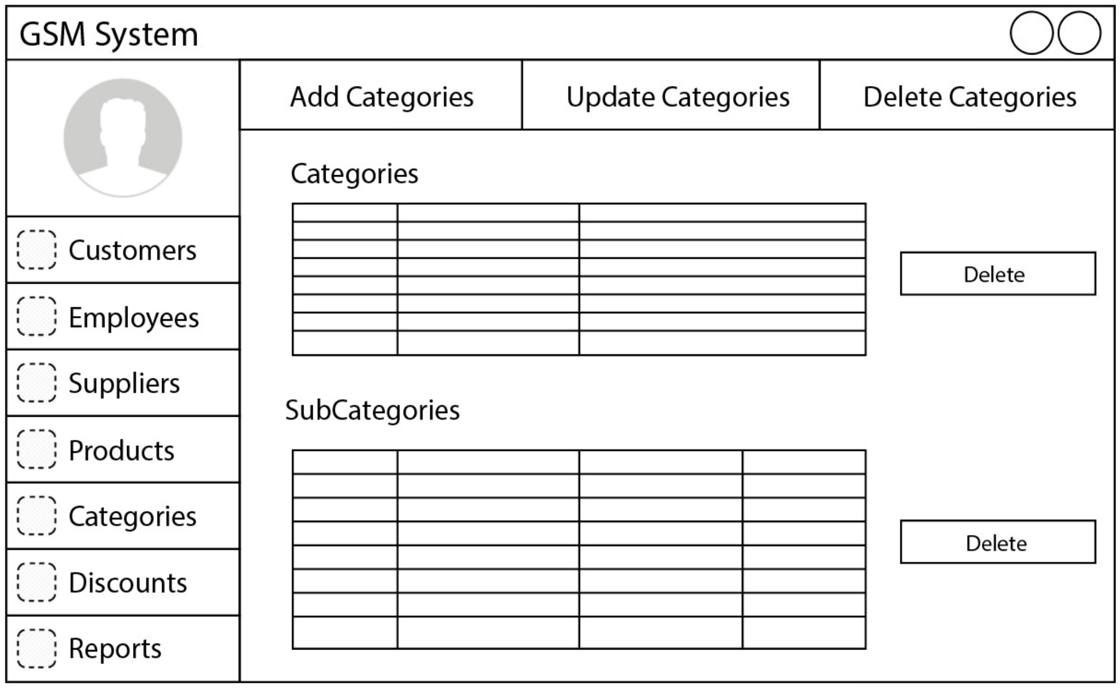
Use Case 10(Update Categories):

Use Case ID	U10
Name	Update Categories
Actor	Admin
Description	**Main Flow:** • The admin navigates to the "Update Categories" section within the shop management system interface. • The system displays a list of current categories available for modification. • The admin selects a category from the list that they want to update. • Upon selecting a category, the system allows the admin to make changes to its name, description, or any other relevant details. • If the updates pass verification, the admin applies the changes, and the category's information is updated in the system. • A confirmation message is displayed, indicating that the category has been successfully updated. • The admin can repeat the process to update more categories as needed. **Alternate Flow:** • If there are no existing categories in the system, the admin is prompted with a message indicating that there are no categories available to update. • At any point during the process, the admin can choose to cancel updating a category and return to the category management interface without making any changes. • If the admin enters invalid information or leaves required fields blank while updating a category, the system prompts them with error messages indicating the specific issues. • The admin corrects the provided information based on the system prompts. • After corrections are made, the system re-validates the entered data. • If validation passes, the system applies the changes.

Layout in pencil tool	 GSM System Add Categories Update Categories Delete Categories Customers Employees Suppliers Products Categories Discounts Reports Categories Name Description Update SubCategories Name Category Description Update
Validators	• The Name box accept string of up to 50 characters in length and cannot be null (empty). • In Description box user can write any string up to 255 characters and can be null. • Only added categories will be displayed in category combo box. • This button when clicked checks all the required boxes and if these boxes fulfill requirement then the categories or subcategories will be updated.

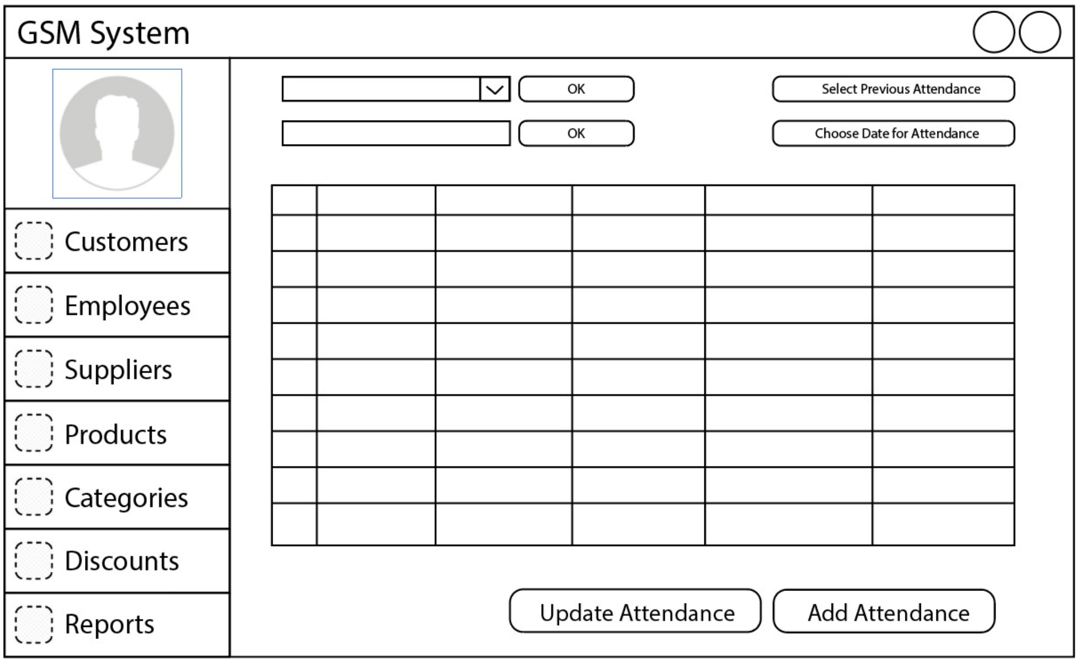
Use Case 11(Delete Categories):

Use Case ID	U11
Name	Delete Categories
Actor	Admin
Description	**Main Flow:** • The admin navigates to the "Delete Categories" section within the shop management system interface. • The system displays a list of current categories available for deletion. • The admin selects a category from the list that they want to delete. • After selecting a category, the admin clicks the "Delete" button to initiate the deletion process. • A prompt message appears, asking the admin to confirm the deletion of the selected category. • If the admin confirms the deletion, the system proceeds with deleting the selected category and all associated products. • The system removes the selected category and all associated products from the database. • A confirmation message is displayed, indicating that the category and associated products have been successfully deleted. • The admin can repeat the process to delete more categories as needed. **Alternate Flow:** • If there are no existing categories in the system, the admin is prompted with a message indicating that there are no categories available to delete. • At any point during the process, the admin can choose to cancel the deletion of a category and return to the category management interface without making any changes. • If the admin cancels the confirmation prompt for deletion, the deletion process is halted, and the selected category remains unchanged. • If an error occurs during the deletion process (e.g., database connection issues), the system prompts the admin with an error message and provides guidance on how to proceed. • The admin can choose to retry the deletion process

Layout in pencil tool	 GSM System Add Categories Update Categories Delete Categories Categories Delete SubCategories Delete Customers Employees Suppliers Products Categories Discounts Reports
Validators	• To delete a category, the user must click on the specified row in the data grid and then click on the delete button. If the user wants to delete a subcategory, they must select the subcategory from the subcategory data grid. • When clicked, this button checks if a row is selected. If a row is selected, it proceeds to delete the corresponding category or subcategory.

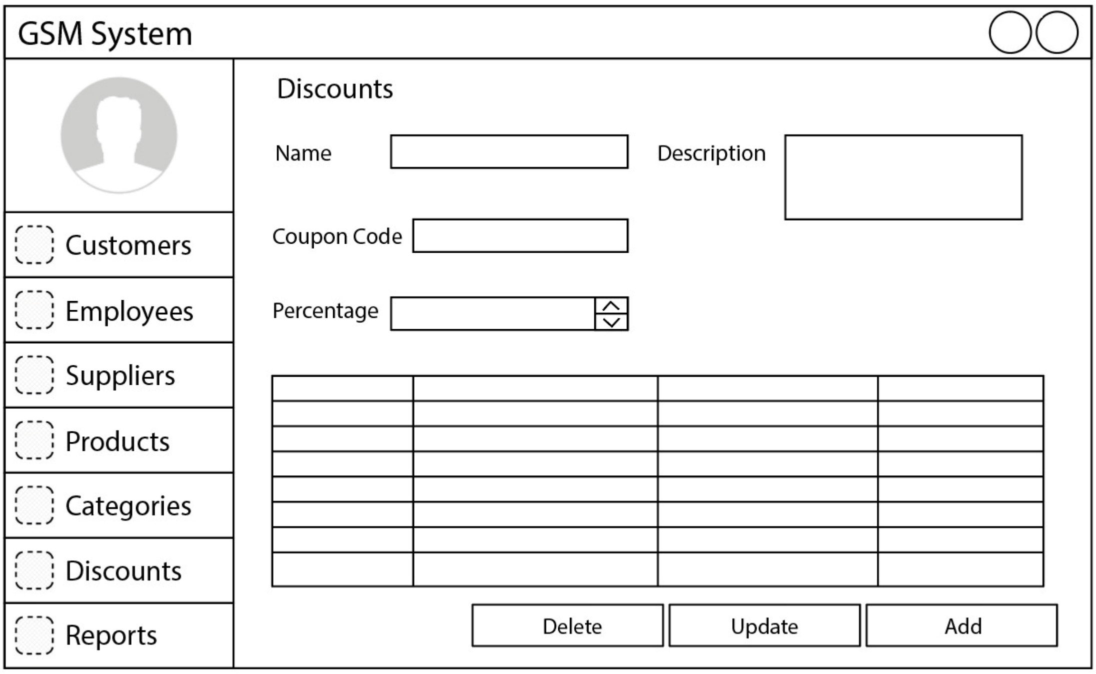
Use Case 12(Employees Attendance):

Use Case ID	U12
Name	Employee Attendance
Actor	Admin
Description	Main Flow: • The admin navigates to the "Employee Attendance" section within the shop management system interface. • The system displays a list of employees along with their attendance records for the selected date. • The admin can manually mark attendance for each employee by selecting their attendance status (present, absent, or on leave). • If needed, the admin can edit existing attendance records by modifying the attendance status for specific employees. • After marking or editing attendance, the admin saves the changes to update the attendance records in the database. • A confirmation message is displayed, indicating that the attendance records have been successfully updated. • The admin can repeat the process to manage attendance for other dates or make further edits as needed. **Alternate Flow:** • If there are no existing attendance records for the selected date, the system prompts the admin with a message indicating that there are no records to display. • If an error occurs during the attendance management process (e.g., database connection issues), the system prompts the admin with an error message and provides guidance on how to proceed.

Layout in pencil tool	 GSM System OK OK Select Previous Attendance Choose Date for Attendance Customers Employees Suppliers Products Categories Discounts Reports Update Attendance Add Attendance
Validators	• When previous attendance button is clicked, a drop-down box and an "Ok" button appear, allowing the admin to choose a date to display previous attendance records. • Admin can only select the date from within the drop-down box. • The "Ok" button, located near the drop-down, displays the attendance for the selected date when a date is chosen from the drop-down. • When ‘Choose date for attendance’ button is clicked, a date-time picker and an "Ok" button appear, enabling admins to mark attendance. • Admin can only select current and previous dates to mark attendance. • When the Ok button located near date time picker is clicked, employee data is displayed on the grid, and admins can mark the attendance of these employees. • To delete an entry, users must click on the specified row in the data grid and then click on the delete button. If users want to delete a subcategory, they must select the subcategory from the subcategory data grid. • When clicked, this button checks if a row is selected. If a row is selected, it proceeds to delete the corresponding category or subcategory.

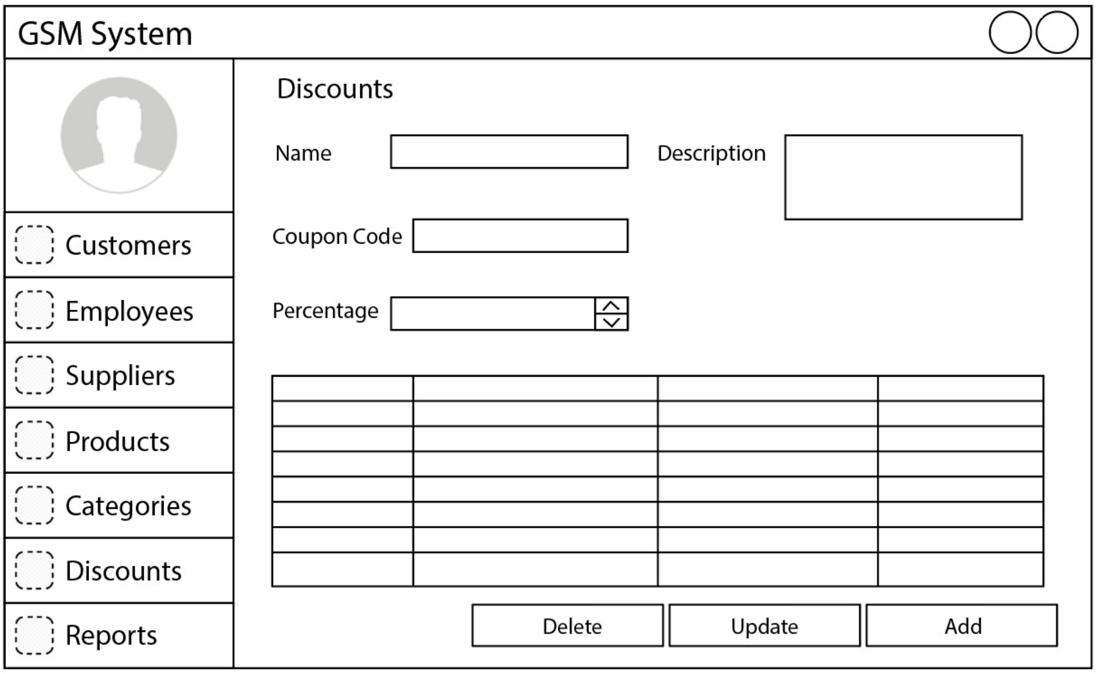
Use Case 13(Add Discounts):

Use Case ID	U13
Name	Add Discounts
Actor	Admin
Description	**Main Flow:** • The admin navigates to the "Discount Management" section within the shop management system interface. • The admin selects the option to add a new discount. • The admin fills out the necessary details for the new discount, including discount type, amount, validity period, and applicable products or categories. • After entering the discount details, the admin saves the new discount. • A confirmation message is displayed, indicating that the discount has been successfully added to the system. • **Alternate Flow:** • If an error occurs during the addition of the discount (e.g., incomplete information, database error), the system prompts the admin with an error message and provides guidance on how to correct the issue. • If entered coupon code already exists in database then system shows error message of duplication. • The admin can choose to retry adding the discount with corrected information or cancel the operation altogether.

Layout in pencil tool	 GSM System Discounts Name Description Coupon Code Percentage Customers Employees Suppliers Products Categories Discounts Reports Delete Update Add
Validators	• The Name box accept string of up to 50 characters in length and cannot be null (empty). • In Description box user can write any string up to 255 characters and can be null. • Coupon code box accept any type of string for special code purpose that not already present. • Percentage box accepts an integer for percentage discount up to 5 characters with 2 decimal places.

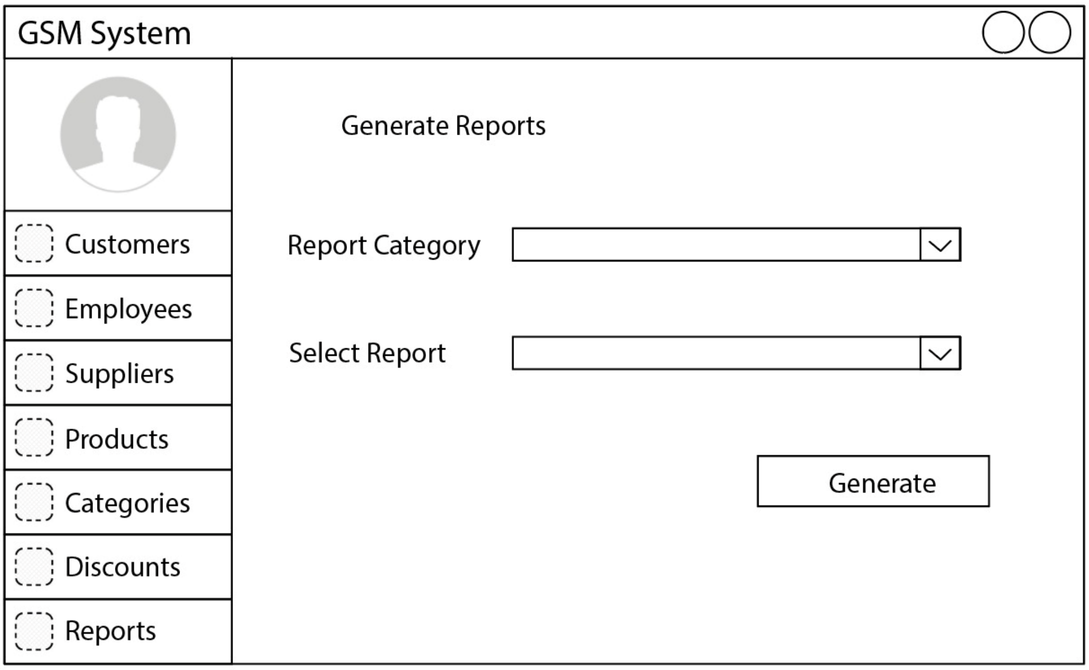
Use Case 14(Edit Discounts):

Use Case ID	U14
Name	Edit Discounts
Actor	Admin
Description	**Main Flow:** • The admin navigates to the "Discount Management" section within the shop management system interface. • The admin selects the option to view existing discounts. • The admin selects the discount they wish to update or delete from the list of existing discounts. • If updating, the admin modifies the necessary details for the selected discount, such as discount type, amount, validity period, or applicable products/categories. • After making updates, the admin saves the changes to the discount. • A confirmation message is displayed, indicating that the discount has been successfully updated/deleted. **Alternate Flow:** • If an error occurs during the update process (e.g., incomplete information, database error), the system prompts the admin with an error message and provides guidance on how to correct the issue. • The admin can choose to retry updating the discount with corrected information or cancel the operation altogether. • When the admin selects the option to delete a discount, a confirmation prompt appears to ensure the deletion is intentional. • The admin confirms the deletion by selecting "Yes". • The system removes the selected discount from the database. • A confirmation message is displayed, indicating that the discount has been successfully deleted.

Layout in pencil tool	 GSM System Customers Employees Suppliers Products Categories Discounts Reports Discounts Name Description Coupon Code Percentage Delete Update Add
Validators	• The Name box accept string of up to 50 characters in length and cannot be null (empty). • In Description box user can write any string up to 255 characters and can be null. • Coupon code box accept any type of string for special code purpose that not already present. • Percentage box accepts an integer for percentage discount. • To delete an entry, users must click on the specified row in the data grid and then click on the delete button. If users want to delete a subcategory, they must select the subcategory from the subcategory data grid. • Update button when clicked checks all the required boxes and if these boxes fulfill requirement then the it updates discounts if its ID matches with ID of discount in database. • When delete clicked, this button checks if a row is selected. If a row is selected, it proceeds to delete the corresponding discount.

Use Case 15(Generate Reports):

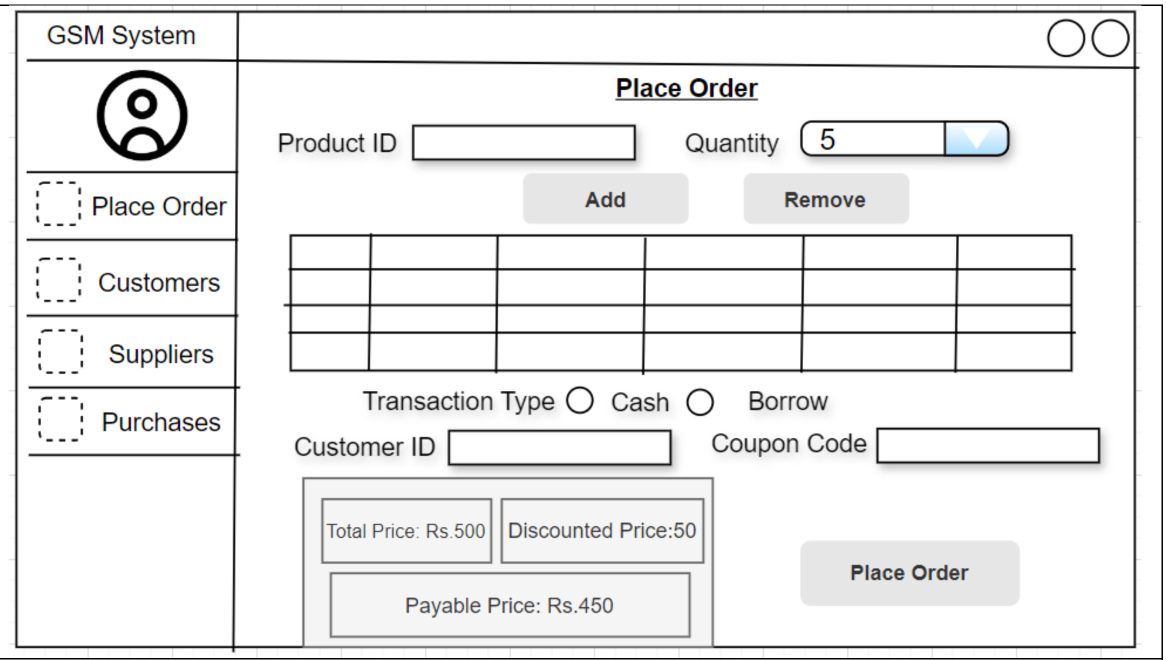
Use Case ID	U15
Name	Generate reports
Actor	Admin
Description	**Main Flow:** • The user opens the C# WinForms application and navigates to the section for generating reports with the Crystal Report extension. • Data is fetched from databases or other sources to be included in the report. • Using the Crystal Reports designer tool in Visual Studio, the system designs the layout of the report, including elements such as headers, footers and table attributes. • User selects a topic from combo box on which report should be generated. • The generated report is displayed within the WinForms application, where users can view it or export it in various formats such as PDF or Excel. **Alternate Flow:** • If an error occurs during the report generation process (e.g., connectivity issues, data source errors), the system prompts the user with an error message and provides guidance on resolving the issue. • The user can choose to retry generating the report after correcting the issue or cancel the operation altogether. • If no option is selected from combo box then system shows error. • Upon successful completion of the export process, the system provides a confirmation message indicating that the report has been successfully exported and saved to the specified location.

Layout in pencil tool	 GSM System Customers Employees Suppliers Products Categories Discounts Reports Generate Reports Report Category Select Report Generate
Validators	• The admin can only select predefined categories from the dropdown menu. • The "Select Report" dropdown menu contains various report topics, from which the admin can choose one. • This button is activated only when both dropdown menus have selections, and upon clicking, it generates a report in PDF format based on the chosen category and report topic.

Use Case 16(Place Order):

Use Case ID	U16
Name	Place Order
Actor	Employee
Description	**Main Flow:** • A customer visits the shop. • An employee adds the customer's details to the customer list, including name, contact information, and other details. • The customer browses the shop's offerings and selects items they wish to purchase. • An employee assists the customer in selecting items and enters their purchases into the system, recording details such as item name, quantity, price, and any applicable discounts. • If the shop is running a promotion, coupon codes are prominently displayed for customers to view. • The customer selects items for purchase and proceeds to the checkout process. • At the checkout, the customer provides their selected items to the employee, who scans or manually enters each item into the system. • If the customer has a coupon code, they provide it to the employee for validation. • The system validates the coupon code and applies any applicable discounts to the customer's purchase. • The customer completes the transaction by making payment through cash, card, or other accepted payment methods. • The employee finalizes the sale, processes the payment, and provides the customer with a receipt or confirmation of their purchase. **Alternate Flow:** • If product Id or Customer Id is entered wrong then system displays an error message. • If the coupon code is invalid or expired then it will not be applicable. • Once the issue is resolved to the customer's satisfaction, the transaction is completed, and the customer receives their purchase along with any applicable discounts or adjustments.

<table>
<tr><td>Layout in pencil tool</td><td>



Quantity 5

Total Price: Rs.500

Discounted Price:50

Payable Price: Rs.450

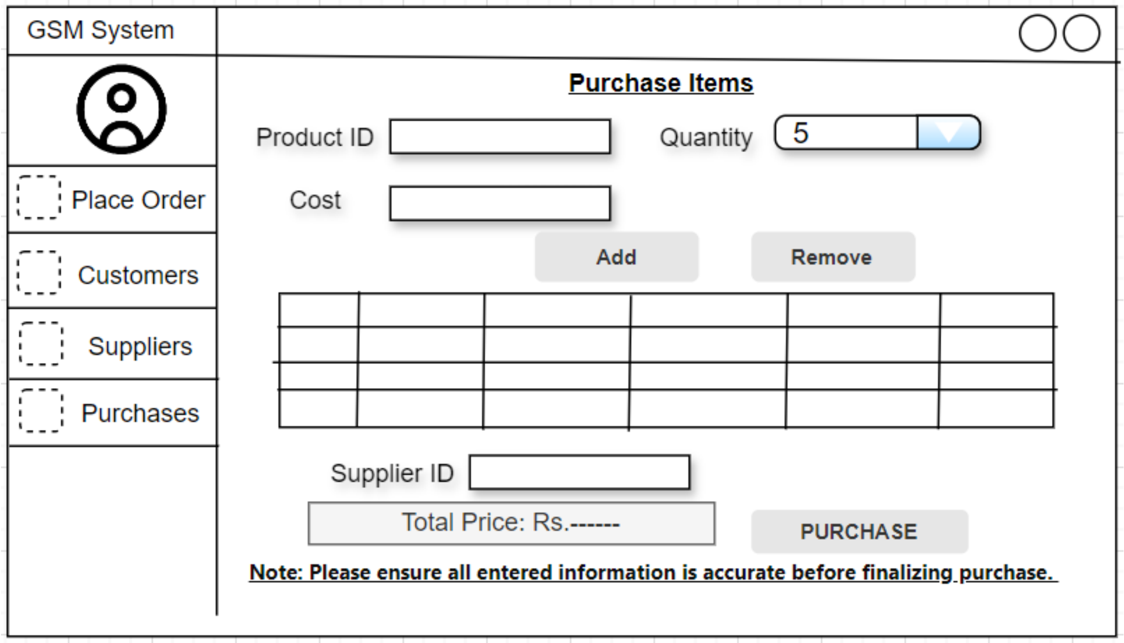
</td></tr>
<tr><td>Validators</td><td>

- Product ID, Coupon and Customer Id boxes contain authentic information of product Id and customer id.
- Quantity drop-down box accepts only integers.
- Coupon code box accept any type of string for special code purpose that not already present. Its optional part.
- User can only select one radio button from Transaction type either payment or borrowing.
- Price details label is not editable by user, it only updates when employee enter the details of products and it adjusts its discount and total price values accordingly.
- When all the required boxes are filled with accurate information, then order button adds the order to orders table.

</td></tr>
</table>

Use Case 17(Purchase Products):

Use Case ID	U17
Name	Purchase Products
Actor	Employee
Description	**Main Flow:** • A supplier arrives at the shop to provide goods. • An employee registers the supplier's information, including name, contact details, email and address. • Once registered, the employee proceeds to select products from the supplier to purchase for the shop. • The employee enters the product details, including the supplier's ID, product details (such as name, description, and price), and the quantity of each item being purchased. • The system automatically calculates the total price based on the quantity and price of each product. • The employee reviews the purchase details and confirms the transaction. • The shop's inventory is updated to reflect the newly purchased products. **Alternate Flow:** • If a new product is introduced to the shop and is not yet listed in the product inventory: An employee notifies the admin about the new product. • The admin reviews the product details provided by the employee. • If approved, the admin adds the new product to the products list, including details such as name, description, price, and any other relevant information. • Once the new product is added to the inventory, employees can proceed to purchase it from the supplier using the standard procedure outlined in the main flow.

	• If Product ID or supplier Id is wrong or cost of product is more than fixed by admin then purchase is not proceeded.
Layout in pencil tool	 GSM System Place Order Customers Suppliers Purchases Purchase Items Product ID Quantity 5 Cost Add Remove Supplier ID Total Price: Rs.------ PURCHASE Note: Please ensure all entered information is accurate before finalizing purchase.
Validators	• Product ID, Supplier ID boxes contain authentic information of product Id and supplier id. • Quantity drop-down box accepts only integers. • Total Price Label is not editable by user, it only updates when employee enter the details of products and it adjusts its total price values accordingly. • When all the required boxes are filled with accurate information, then products are purchased from supplier by clicking on purchase button.

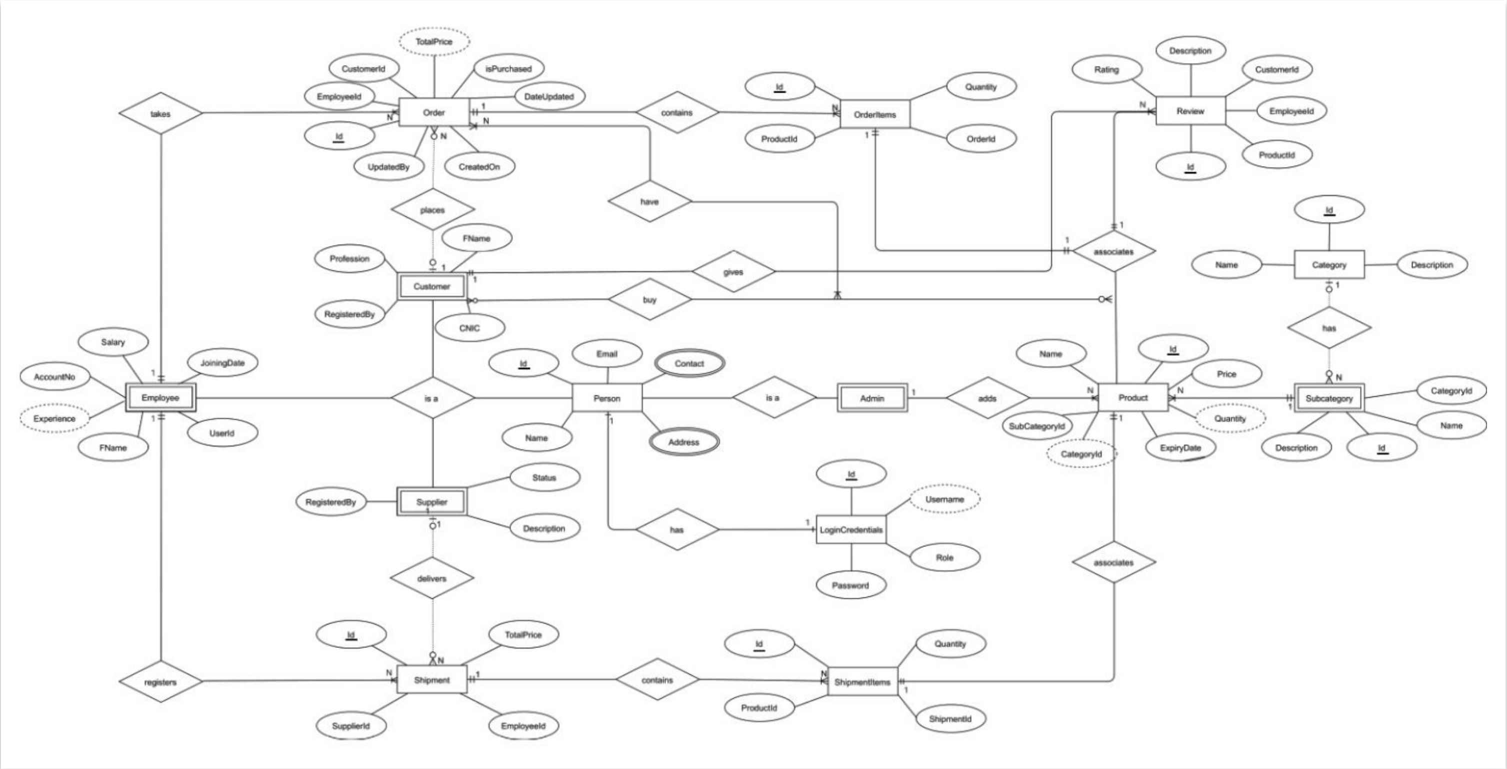
User Interface Details

Use Case Id	TextBox	DropDown	Password Box	Table	Date Field	Buttons	AutoComplete	Radio Button	CheckBox	Menu	Text Area	ProgressBar
U01	2	0	1	0	0	1	0	0	0	0	0	0
U02	2	0	1	1	0	1	0	0	0	1	0	0
U03	7	0	0	0	0	1	0	0	1	1	0	0
U04	7	0	0	1	0	2	0	2	0	1	0	0
U05	6	1	0	0	0	1	0	0	0	1	0	0
U06	6	1	0	1	0	2	0	0	0	1	0	0
U07	5	0	0	0	0	1	0	0	0	1	0	0
U08	5	0	0	1	0	2	0	2	0	1	0	0
U09	4	1	0	2	0	2	0	0	0	1	0	0
U10	4	1	0	2	0	2	0	0	0	1	0	0
U11	4	1	0	2	0	2	0	0	0	1	0	0
U12	0	1	0	1	1	6	0	0	0	1	0	0
U13	4	0	0	1	0	3	0	0	0	1	0	0
U14	4	0	0	1	0	3	0	0	0	1	0	0
U15	0	2	0	0	0	1	0	0	0	1	0	0
U16	3	1	0	0	0	1	0	2	0	1	0	0
U17	2	1	0	0	0	1	0	0	0	1	0	0

Classes:

Class Name	Software/ Domain	Is Abstract (Yes/No)	Is Singleton (Yes/No)	Is the class will has parametrized constructor(Yes/No)
Order	Domain	No	No	Yes
Customer				
Supplier				
Employee				
Shipment				
Category				
SubCategory				
Product				
Discount				
Transaction	Software	Yes		
Person				
Item				
Credentials		No		
Invoice				
PDFGenerator				
ExceptionGenerator				
Admin			Yes	No
Shop Details				Yes

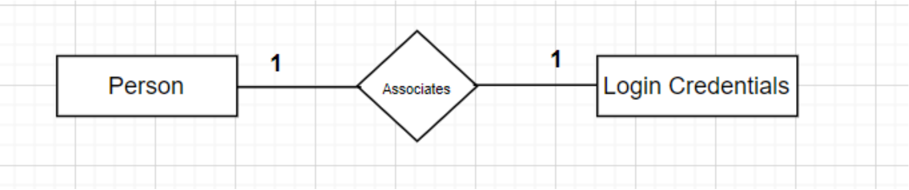
ER Diagram:



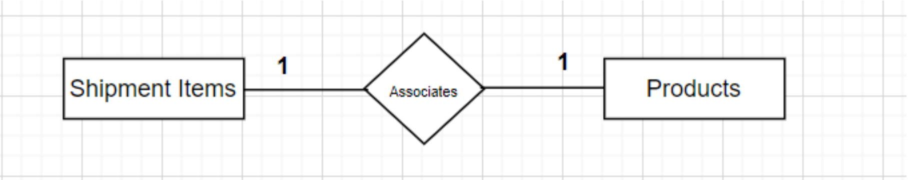
ER Design:

One to One Relationship

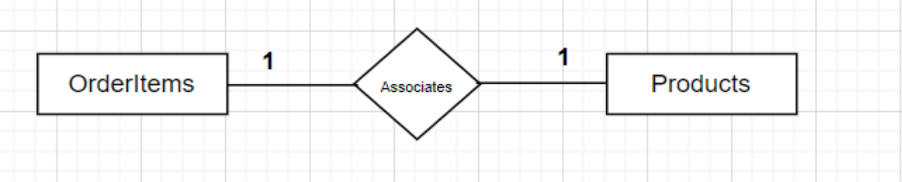
1)Person - Login Credentials:



2)Shipment Item – Product:

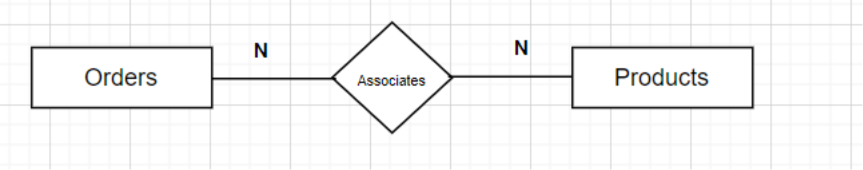


3) Order Items – Product

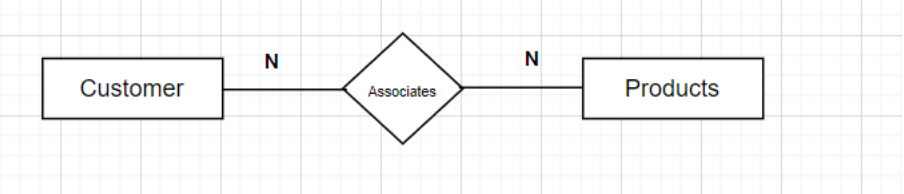


Many to Many Relationship:

1)Order – Products:

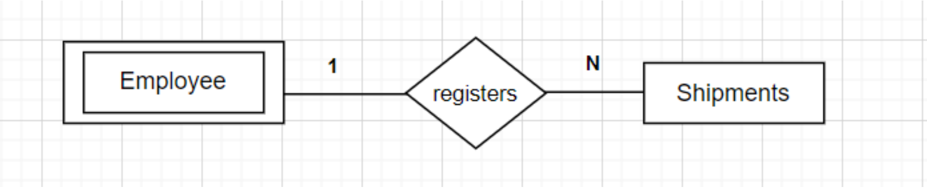


2) Customer – Products:

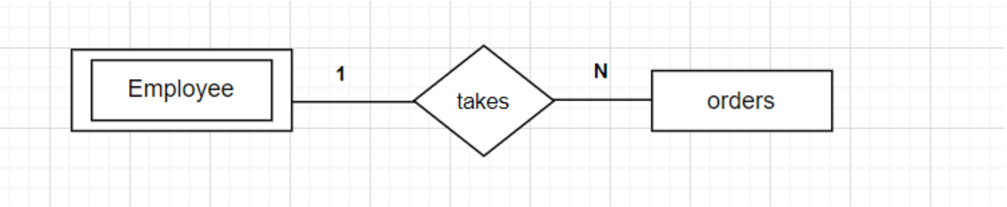


One to Many Relationship

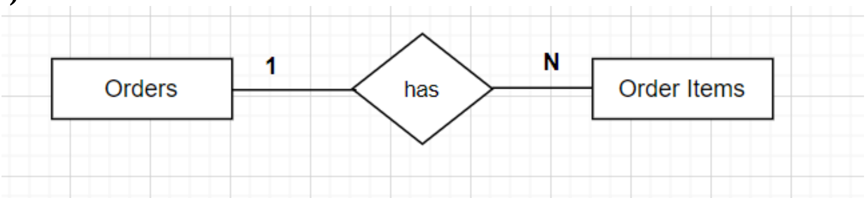
1)Employee – Shipments:



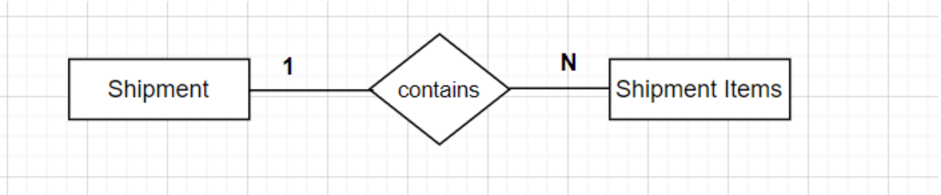
2)Employee – Orders:



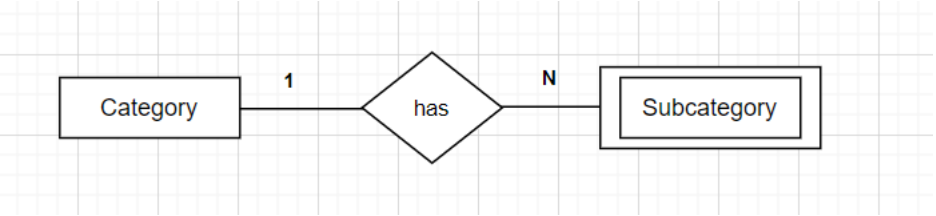
3)Order – Order Items:



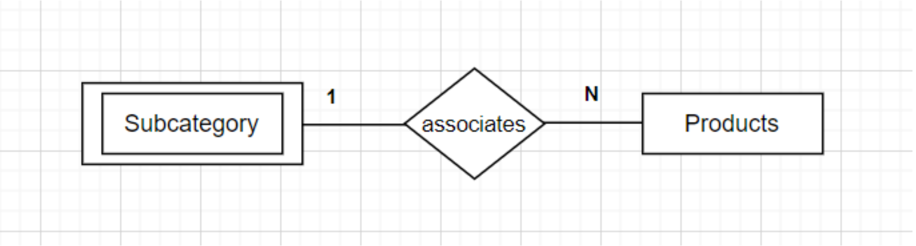
4)Shipment – Shipment Items:



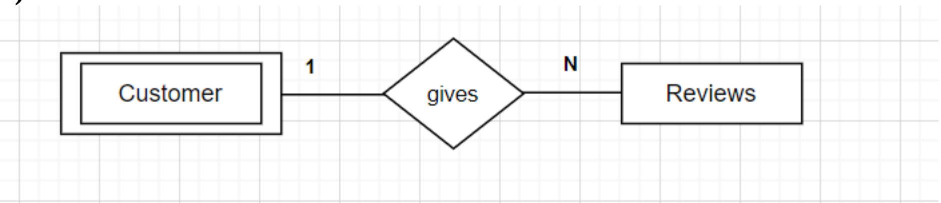
5)Category – Sub Category:



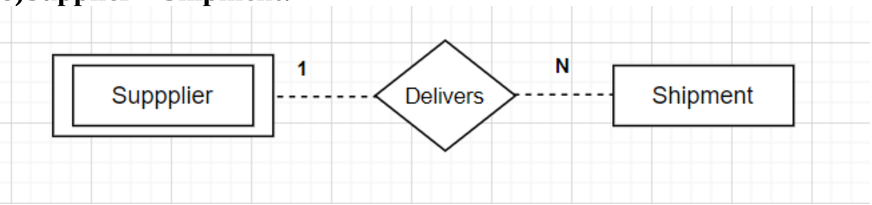
6)Sub Category – Products:



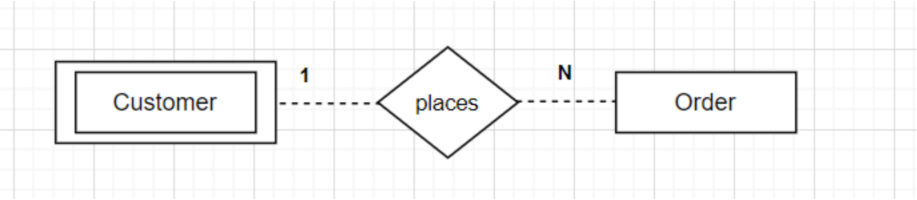
7)Customer – Reviews:



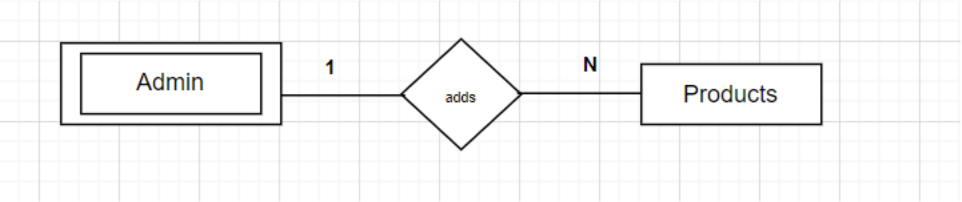
8)Supplier – Shipment:



9)Customer – Order:

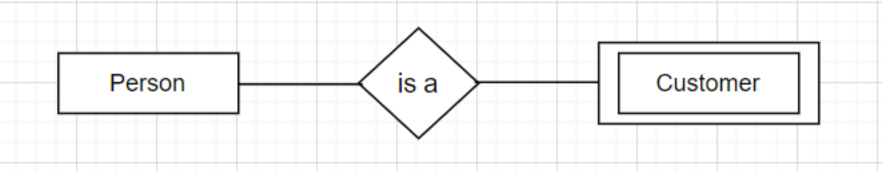


10)Admin – Products:

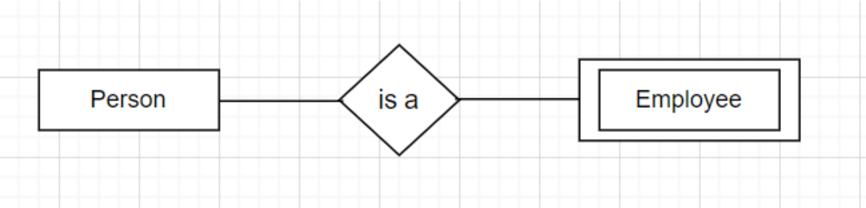


Multi-Level Inheritance:

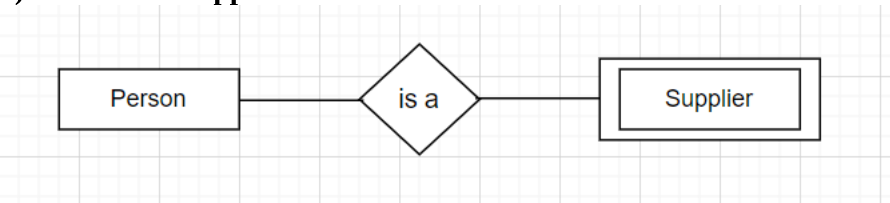
1) Person and customer:



2) Person and Employee:

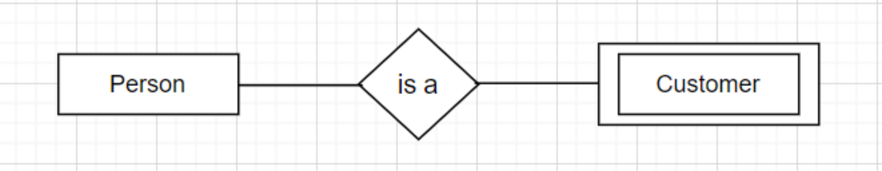


3) Person and Supplier:

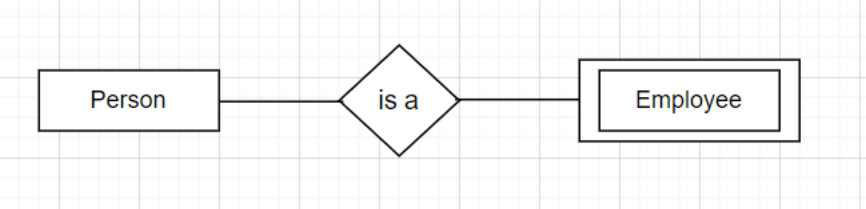


Weak Entities:

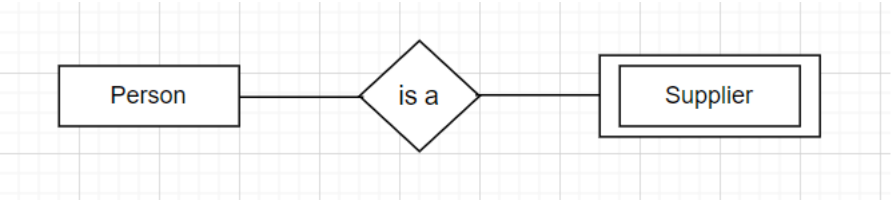
1)Customer:



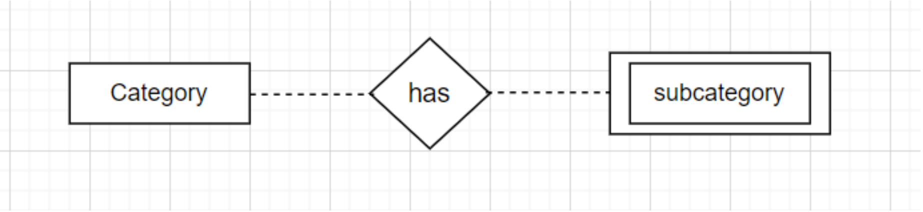
2)Employee:



3) Supplier

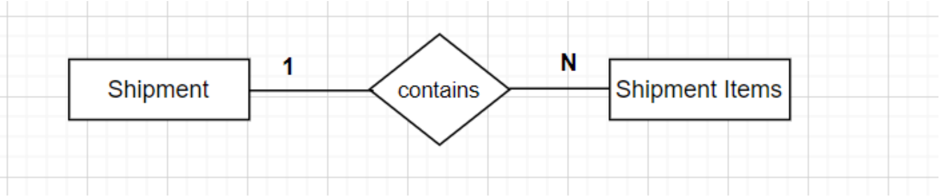


4) Subcategory

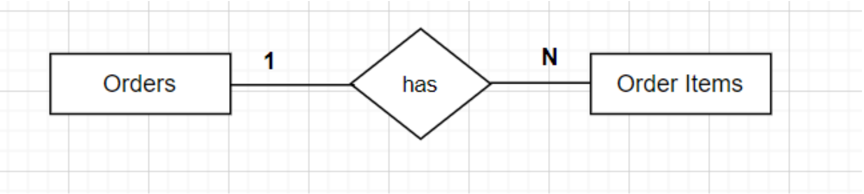


Strong Relationship:

1)Shipment – Shipment Items:



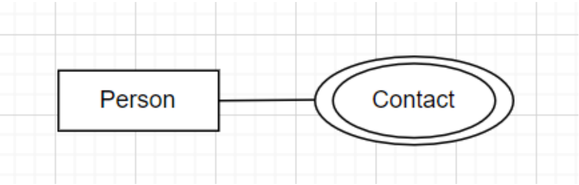
2)Order – Order Items:



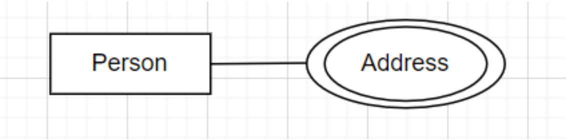
Multivalued Attributes:

The attributes which can have more than one values are:

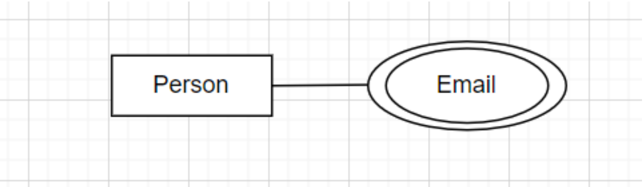
1)Contact:



2)Address:



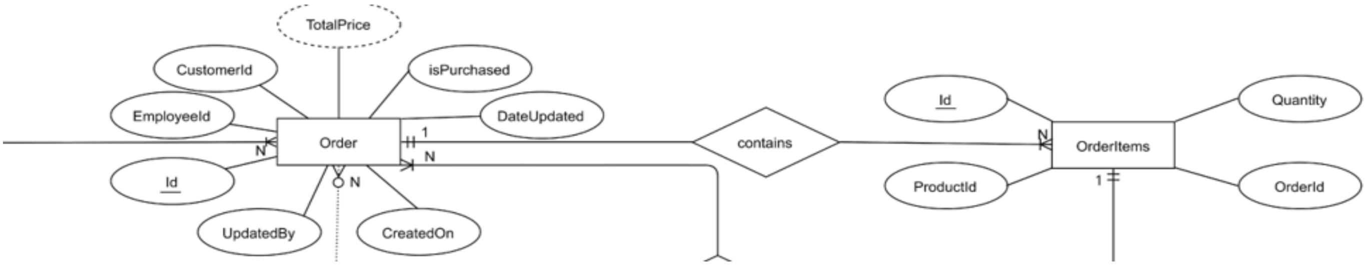
3)Email:



Computed Attributes:

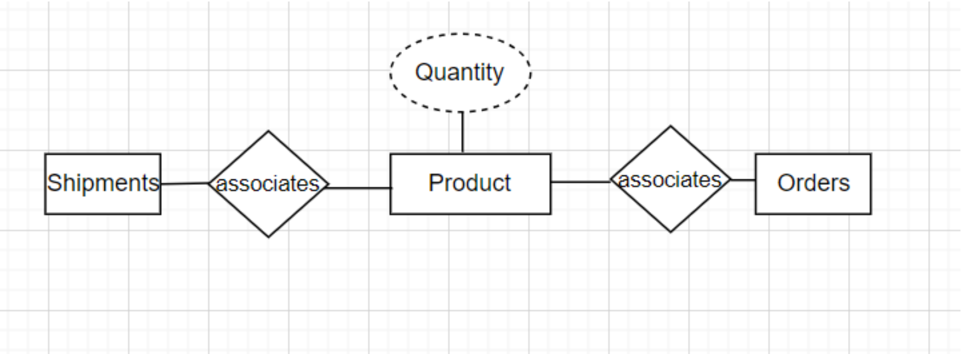
1)Total Price:

The total price an order is calculated by computing sum of orderId in OrderItems and then multiplying that sum with price of product.



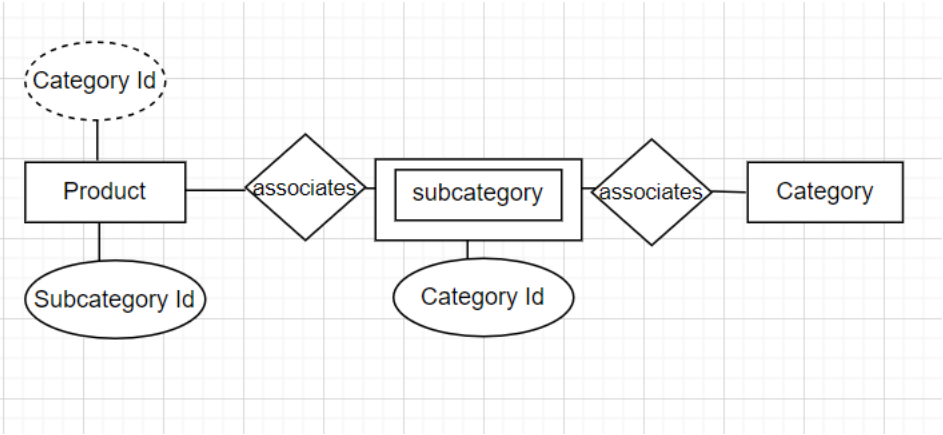
2) Product Quantity

The quantity of products is calculated by comparing sums of orders and shipments.



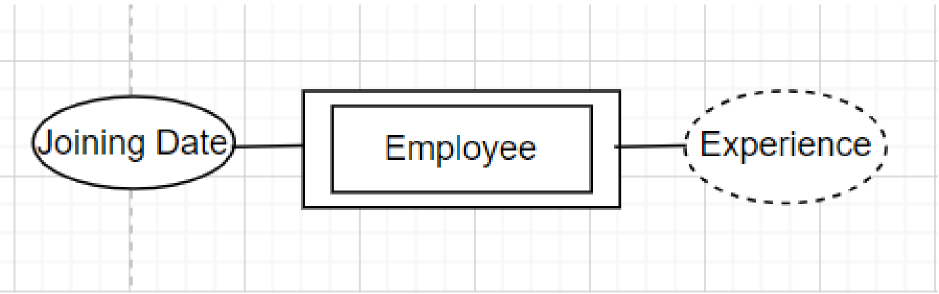
3) Category Id of Products:

Category Id attribute of products is derived from subcategoryID which further derived to category table.



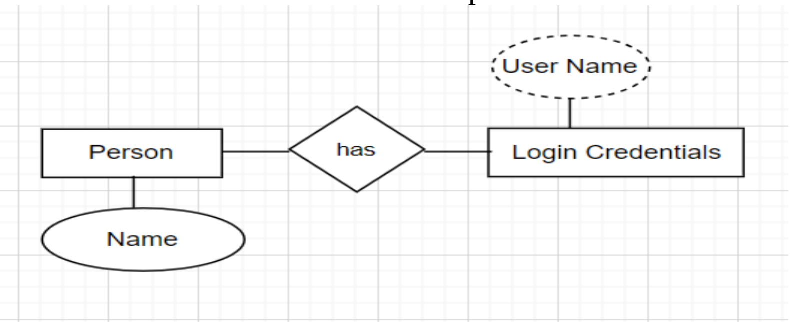
4)Experience of Employee:

Experience of employee is calculated by subtracting joining date and today’s date.



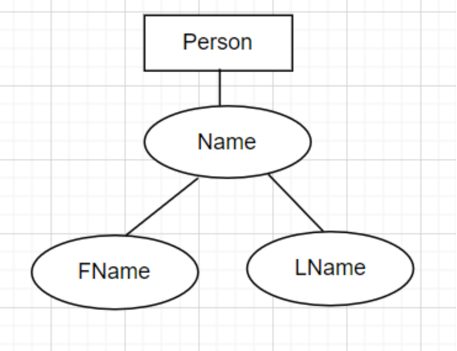
5) User Name:

User name is derived from Name of person.

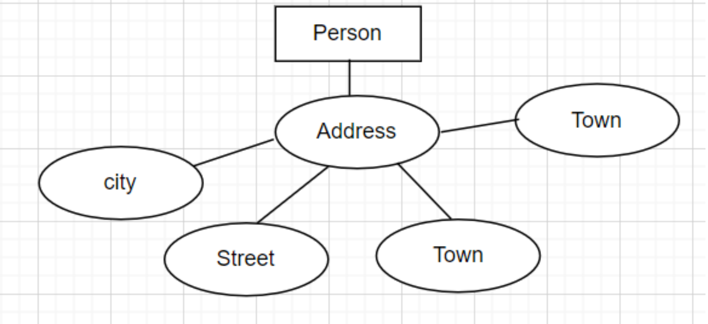


Complex Attributes:

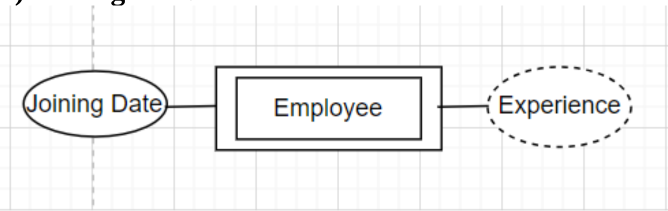
1) Person Name:



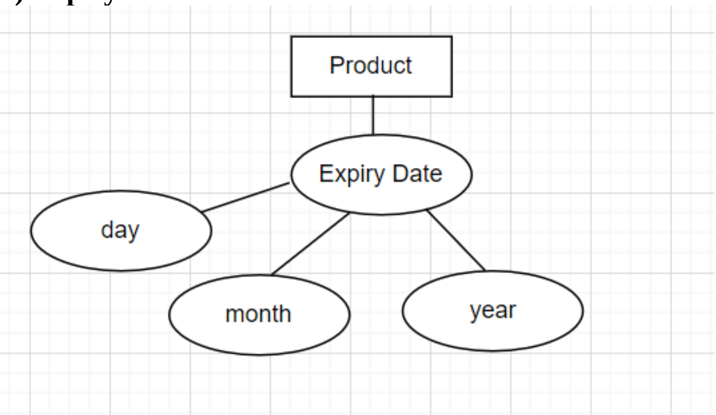
2) Person Address:



3) Joining date:



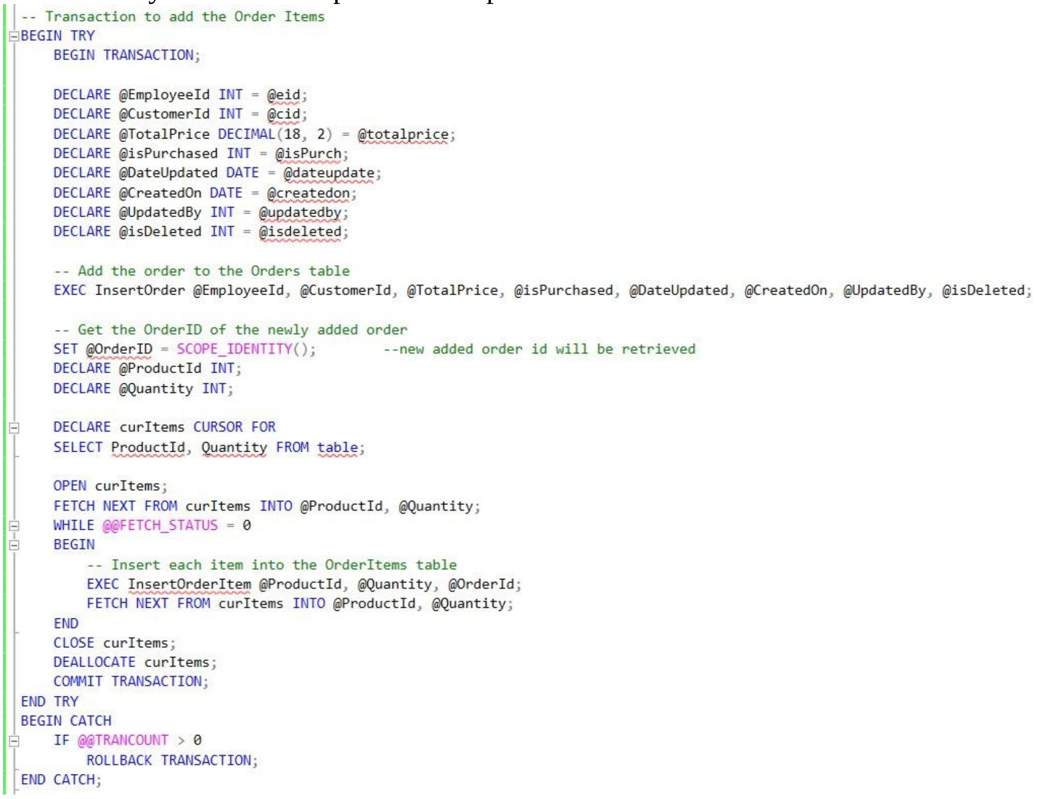
4) Expiry date



Transactions

Add Order Items:

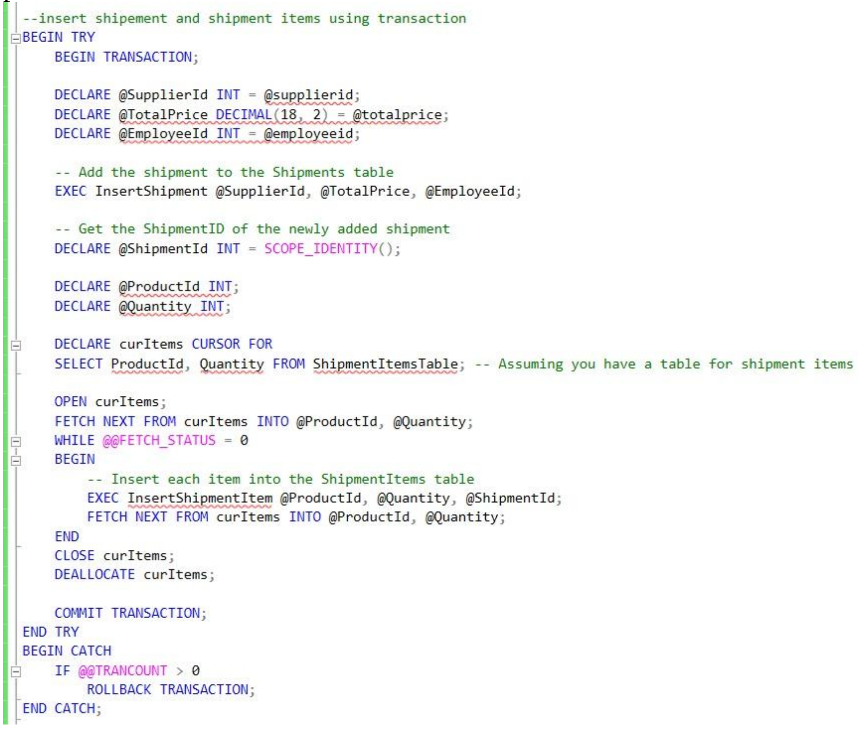
This transaction is used to add the Orders and OrderItems in the database, where the insertion operation is performed in once. The transaction uses the two predefined stored procedures for its execution. This transaction will ensure the data consistency and atomicity in the database. Moreover depending on the no of items the transaction will automatically handles the tuple insertion process.



```
-- Transaction to add the Order Items
BEGIN TRY
    BEGIN TRANSACTION;

    DECLARE @EmployeeId INT = @eid;
    DECLARE @CustomerId INT = @cid;
    DECLARE @TotalPrice DECIMAL(18, 2) = @totalprice;
    DECLARE @isPurchased INT = @isPurch;
    DECLARE @DateUpdated DATE = @dateupdate;
    DECLARE @CreatedOn DATE = @createdon;
    DECLARE @UpdatedBy INT = @updatedby;
    DECLARE @isDeleted INT = @isdeleted;

    -- Add the order to the Orders table
    EXEC InsertOrder @EmployeeId, @CustomerId, @TotalPrice, @isPurchased, @DateUpdated, @CreatedOn, @UpdatedBy, @isDeleted;

    -- Get the OrderID of the newly added order
    SET @OrderID = SCOPE_IDENTITY();        --new added order id will be retrieved
    DECLARE @ProductId INT;
    DECLARE @Quantity INT;

    DECLARE curItems CURSOR FOR
    SELECT ProductId, Quantity FROM table;

    OPEN curItems;
    FETCH NEXT FROM curItems INTO @ProductId, @Quantity;
    WHILE @@FETCH_STATUS = 0
    BEGIN
        -- Insert each item into the OrderItems table
        EXEC InsertOrderItem @ProductId, @Quantity, @OrderId;
        FETCH NEXT FROM curItems INTO @ProductId, @Quantity;
    END
    CLOSE curItems;
    DEALLOCATE curItems;
    COMMIT TRANSACTION;
END TRY
BEGIN CATCH
    IF @@TRANCOUNT > 0
        ROLLBACK TRANSACTION;
END CATCH;
```

Insert shipment and shipment item:

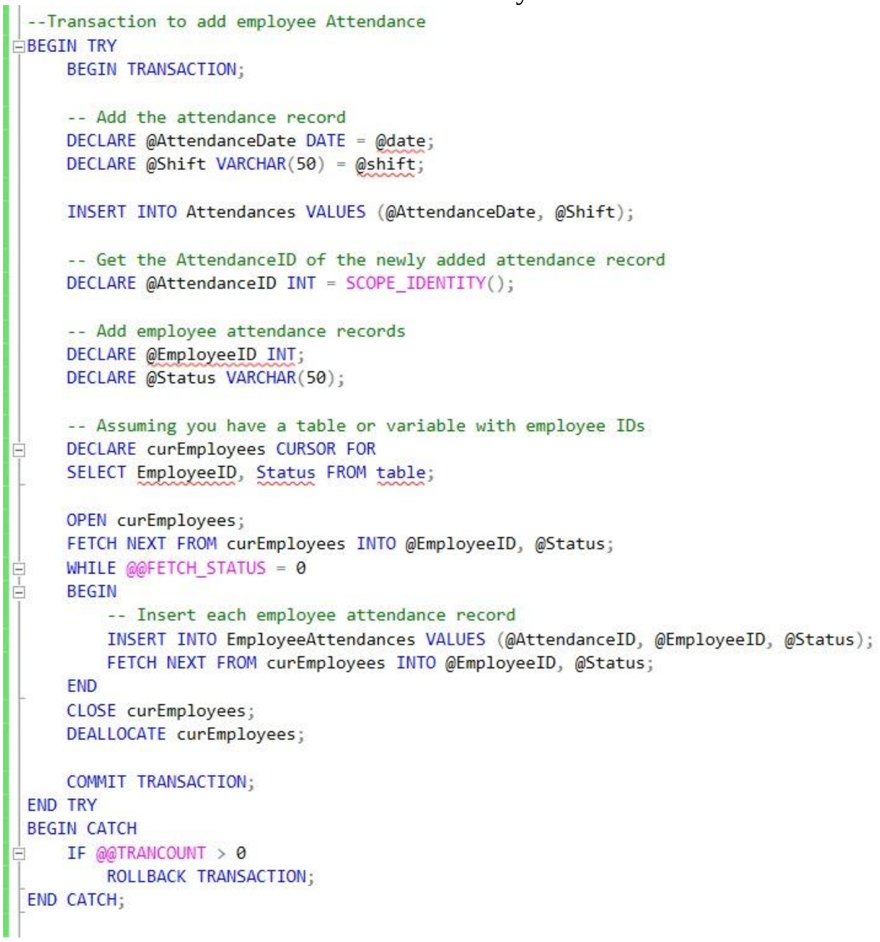
This transaction is like above used to add the Shipments and ShipmentItems in the database, where the insertion operation is performed. The transaction also uses the two predefined stored procedures for its execution. Depending on the no of items the transaction will automatically handles the tuple insertion process. A table of data is passed as a parameter



```
--insert shipement and shipment items using transaction
BEGIN TRY
    BEGIN TRANSACTION;

    DECLARE @SupplierId INT = @supplierid;
    DECLARE @TotalPrice DECIMAL(18, 2) = @totalprice;
    DECLARE @EmployeeId INT = @employeeid;

    -- Add the shipment to the Shipments table
    EXEC InsertShipment @SupplierId, @TotalPrice, @EmployeeId;

    -- Get the ShipmentID of the newly added shipment
    DECLARE @ShipmentId INT = SCOPE_IDENTITY();

    DECLARE @ProductId INT;
    DECLARE @Quantity INT;

    DECLARE curItems CURSOR FOR
    SELECT ProductId, Quantity FROM ShipmentItemsTable; -- Assuming you have a table for shipment items

    OPEN curItems;
    FETCH NEXT FROM curItems INTO @ProductId, @Quantity;
    WHILE @@FETCH_STATUS = 0
    BEGIN
        -- Insert each item into the ShipmentItems table
        EXEC InsertShipmentItem @ProductId, @Quantity, @ShipmentId;
        FETCH NEXT FROM curItems INTO @ProductId, @Quantity;
    END
    CLOSE curItems;
    DEALLOCATE curItems;

    COMMIT TRANSACTION;
END TRY
BEGIN CATCH
    IF @@TRANCOUNT > 0
        ROLLBACK TRANSACTION;
END CATCH;
```

Add Employee Attendance:

This transaction will be useful for marking employee attendance and is designed to make this process atomic. The Attendance data is provided in table to transaction and after adding the attendance the employee status is also added for attendance in the database. In case of any error or issue it will be Rollback the transaction.

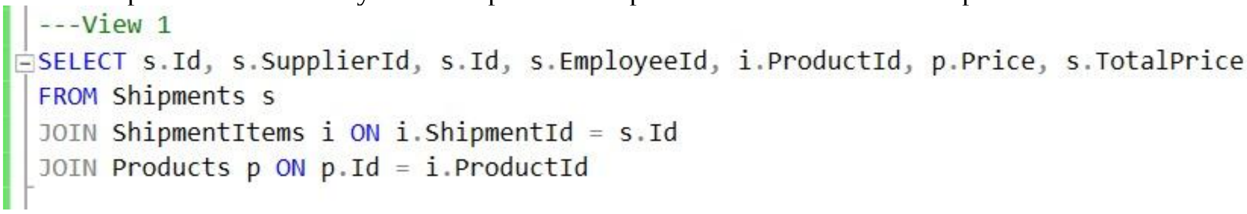


```sql
--Transaction to add employee Attendance
BEGIN TRY
    BEGIN TRANSACTION;

    -- Add the attendance record
    DECLARE @AttendanceDate DATE = @date;
    DECLARE @Shift VARCHAR(50) = @shift;

    INSERT INTO Attendances VALUES (@AttendanceDate, @Shift);

    -- Get the AttendanceID of the newly added attendance record
    DECLARE @AttendanceID INT = SCOPE_IDENTITY();

    -- Add employee attendance records
    DECLARE @EmployeeID INT;
    DECLARE @Status VARCHAR(50);

    -- Assuming you have a table or variable with employee IDs
    DECLARE curEmployees CURSOR FOR
    SELECT EmployeeID, Status FROM table;

    OPEN curEmployees;
    FETCH NEXT FROM curEmployees INTO @EmployeeID, @Status;
    WHILE @@FETCH_STATUS = 0
    BEGIN
        -- Insert each employee attendance record
        INSERT INTO EmployeeAttendances VALUES (@AttendanceID, @EmployeeID, @Status);
        FETCH NEXT FROM curEmployees INTO @EmployeeID, @Status;
    END
    CLOSE curEmployees;
    DEALLOCATE curEmployees;

    COMMIT TRANSACTION;
END TRY
BEGIN CATCH
    IF @@TRANCOUNT > 0
        ROLLBACK TRANSACTION;
END CATCH;
```

Views

View1:

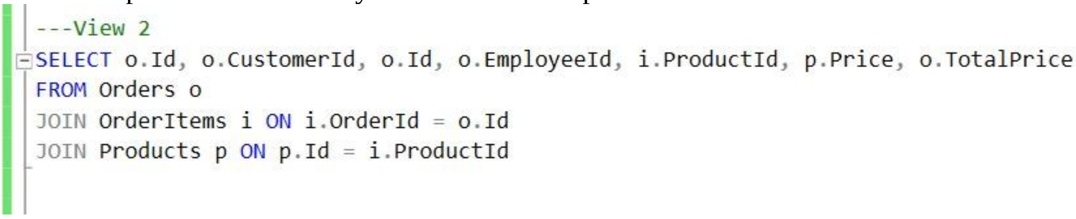
This view present the summary of the shipments and products delivered in the shipments.



```sql
---View 1
SELECT s.Id, s.SupplierId, s.Id, s.EmployeeId, i.ProductId, p.Price, s.TotalPrice
FROM Shipments s
JOIN ShipmentItems i ON i.ShipmentId = s.Id
JOIN Products p ON p.Id = i.ProductId
```

View2:

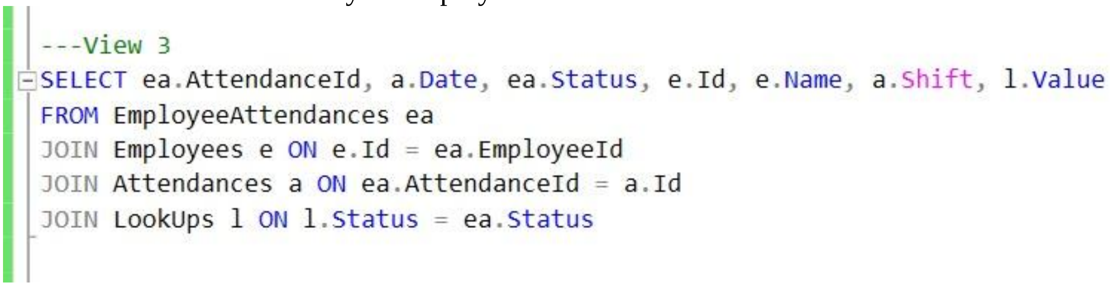
This view present the summary of the orders and products sold in the orders.



```sql
---View 2
SELECT o.Id, o.CustomerId, o.Id, o.EmployeeId, i.ProductId, p.Price, o.TotalPrice
FROM Orders o
JOIN OrderItems i ON i.OrderId = o.Id
JOIN Products p ON p.Id = i.ProductId
```

View3

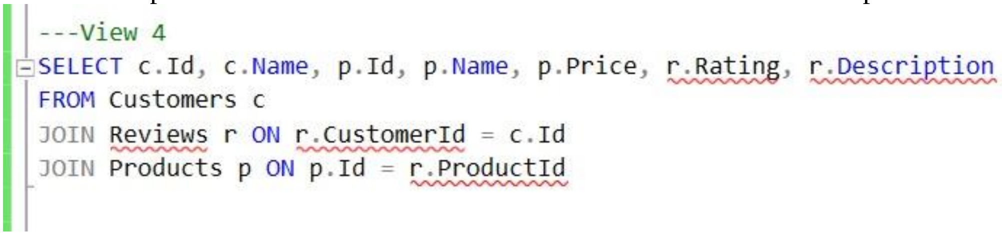
This view shows the summary of Employee Attendance and their status with all the marked attendances.



```
---View 3
SELECT ea.AttendanceId, a.Date, ea.Status, e.Id, e.Name, a.Shift, l.Value
FROM EmployeeAttendances ea
JOIN Employees e ON e.Id = ea.EmployeeId
JOIN Attendances a ON ea.AttendanceId = a.Id
JOIN LookUps l ON l.Status = ea.Status
```

View4:

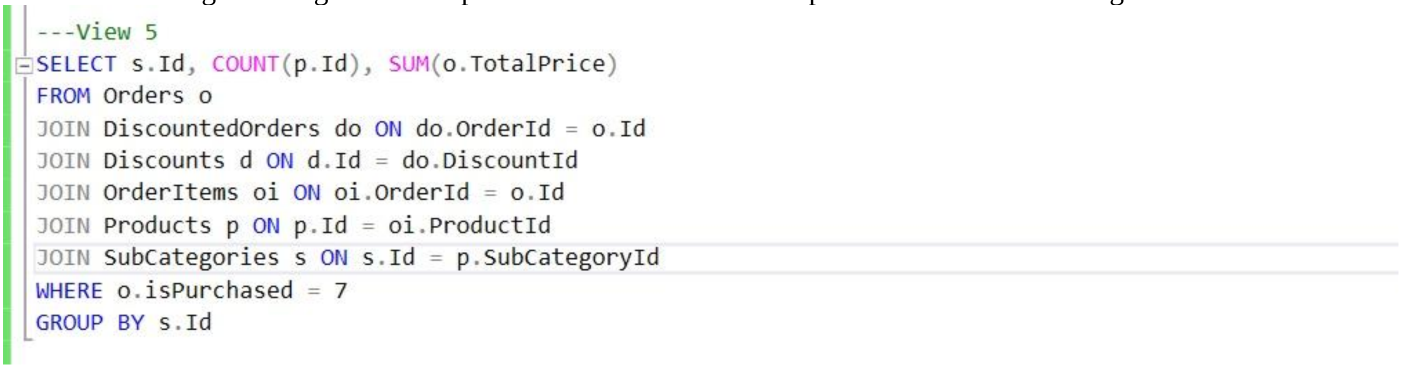
This view represents the overview of customers reviews for the different products.



```
---View 4
SELECT c.Id, c.Name, p.Id, p.Name, p.Price, r.Rating, r.Description
FROM Customers c
JOIN Reviews r ON r.CustomerId = c.Id
JOIN Products p ON p.Id = r.ProductId
```

View5

This view is designed to figure out the performance of discounts coupons for different subcategories

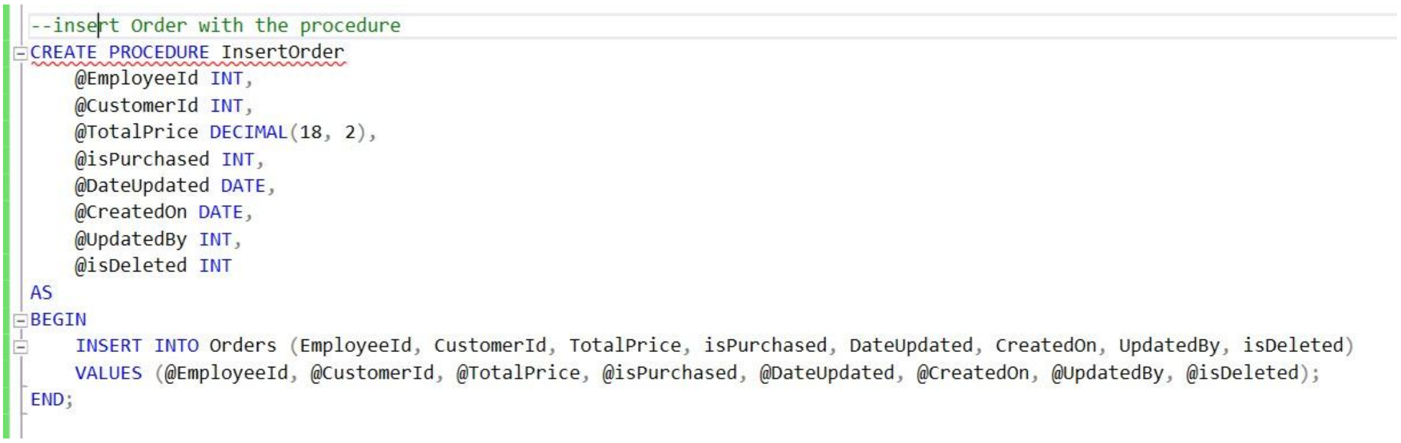


```
---View 5
SELECT s.Id, COUNT(p.Id), SUM(o.TotalPrice)
FROM Orders o
JOIN DiscountedOrders do ON do.OrderId = o.Id
JOIN Discounts d ON d.Id = do.DiscountId
JOIN OrderItems oi ON oi.OrderId = o.Id
JOIN Products p ON p.Id = oi.ProductId
JOIN SubCategories s ON s.Id = p.SubCategoryId
WHERE o.isPurchased = 7
GROUP BY s.Id
```

Stored Procedures

Insert Order:

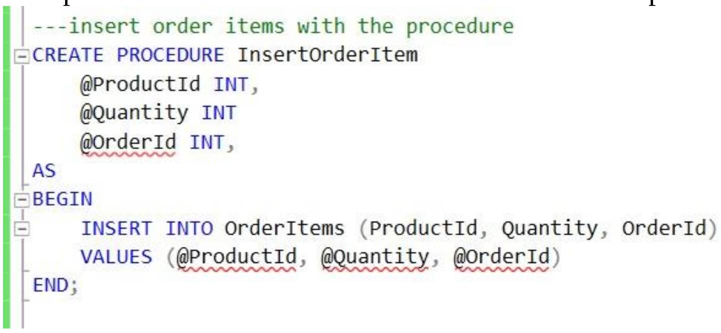
This is a parameterized stored procedure which takes the attributes of the order as the parameters and add them in the Orders table.



```
--insert Order with the procedure
CREATE PROCEDURE InsertOrder
    @EmployeeId INT,
    @CustomerId INT,
    @TotalPrice DECIMAL(18, 2),
    @isPurchased INT,
    @DateUpdated DATE,
    @CreatedOn DATE,
    @UpdatedBy INT,
    @isDeleted INT
AS
BEGIN
    INSERT INTO Orders (EmployeeId, CustomerId, TotalPrice, isPurchased, DateUpdated, CreatedOn, UpdatedBy, isDeleted)
    VALUES (@EmployeeId, @CustomerId, @TotalPrice, @isPurchased, @DateUpdated, @CreatedOn, @UpdatedBy, @isDeleted);
END;
```

Insert OrderItem

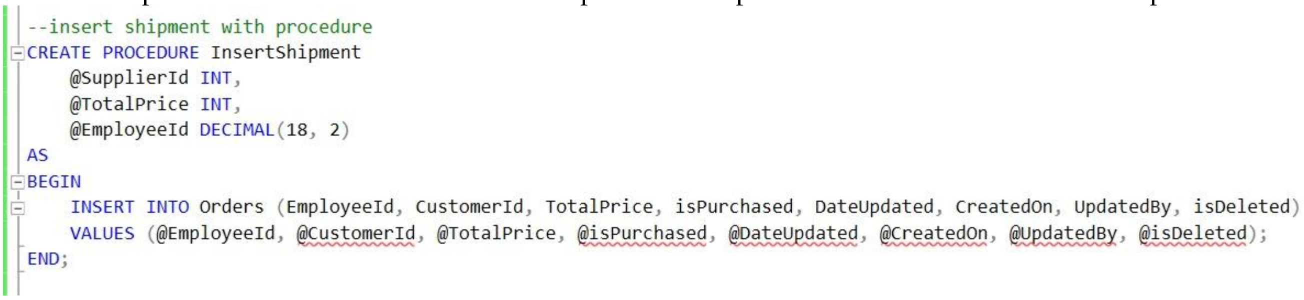
This procedure takes the attribute of an order item and perform insertion operation in the OrderItems table.



```
---insert order items with the procedure
CREATE PROCEDURE InsertOrderItem
    @ProductId INT,
    @Quantity INT
    @OrderId INT,
AS
BEGIN
    INSERT INTO OrderItems (ProductId, Quantity, OrderId)
    VALUES (@ProductId, @Quantity, @OrderId)
END;
```

Insert Shipment:

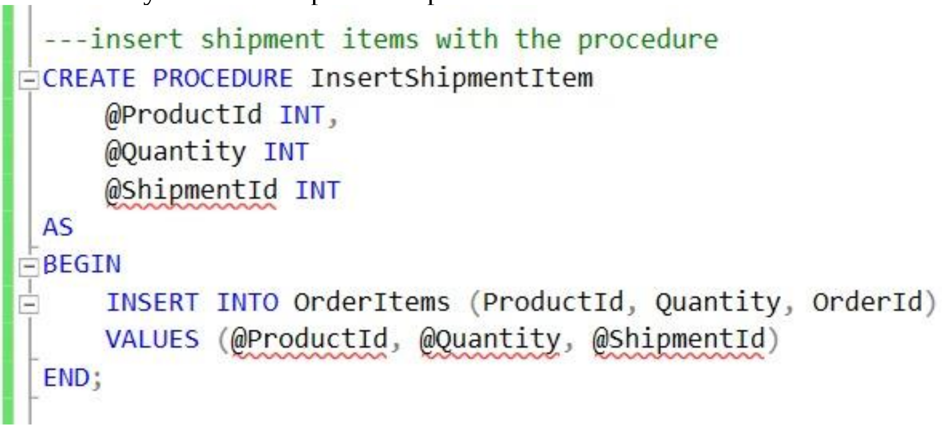
This stored procedure takes the attributes of a shipment as the parameters and add them in the Shipments table.



```
--insert shipment with procedure
CREATE PROCEDURE InsertShipment
    @SupplierId INT,
    @TotalPrice INT,
    @EmployeeId DECIMAL(18, 2)
AS
BEGIN
    INSERT INTO Orders (EmployeeId, CustomerId, TotalPrice, isPurchased, DateUpdated, CreatedOn, UpdatedBy, isDeleted)
    VALUES (@EmployeeId, @CustomerId, @TotalPrice, @isPurchased, @DateUpdated, @CreatedOn, @UpdatedBy, @isDeleted);
END;
```

Insert ShipmentItem:

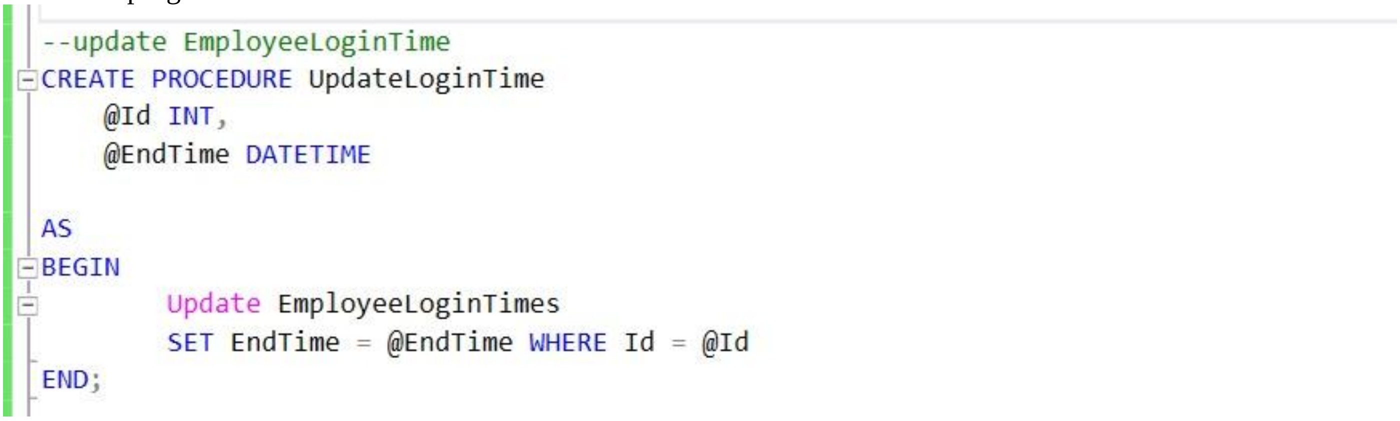
This stored procedure takes the values of an shipment item and adds them in the ShipmentItems table. This procedure makes it easy to add the tuples multiple times



```
---insert shipment items with the procedure
CREATE PROCEDURE InsertShipmentItem
    @ProductId INT,
    @Quantity INT
    @ShipmentId INT
AS
BEGIN
    INSERT INTO OrderItems (ProductId, Quantity, OrderId)
    VALUES (@ProductId, @Quantity, @ShipmentId)
END;
```

Update Logintime

This Procedure is used for the updation purpose in EmployeeLoginTime and Update the EndTime once the User leaves the program.



```
--update EmployeeLoginTime
CREATE PROCEDURE UpdateLoginTime
    @Id INT,
    @EndTime DATETIME

AS
BEGIN
        Update EmployeeLoginTimes
        SET EndTime = @EndTime WHERE Id = @Id
END;
```

Exceptions:

Type of Exception	**Why this exception will occur**	**How you will handle the exception**
AuthenticationError	Occurs when the entered username or password is incorrect.	Provide clear error messages to inform the user about the incorrect username or password. If employee has forgot login credentials then employee can contact to admin. If that case for admin then admin can contact with developer of application.
MissingFieldError	Occurs when attempting to perform an action (e.g., adding) without providing values for required fields.	Implement client-side validation to ensure all required fields are filled before allowing the action to proceed. Display error messages indicating which fields are missing or invalid, and prompt the user to complete the required information before proceeding with the action.
NullReferenceException	Occurs when application attempts to access members (methods, properties, fields) of a null object. Like if a person who is not registered in database tries to login then null object is returned from LoginCredentials and this exception s thrown.	Ensure that objects are properly initialized before accessing them. Use Try-catch to detect null.
ArgumentException	Thrown when one or more arguments provided to a method are invalid. Like when we expect positive value of price but a negative value is entered then that exception is thrown.	Validate input parameters before using them in methods. Provide clear error messages indicating which argument is invalid and why.
SqlConnectionError	Occurs when there are issues establishing a connection to the SQL database, such as incorrect credentials or network problems.	Double-check connection string details, including database name. Ensure that the SQL server is reachable and configured to accept connections. Also ensure device is connected to internet.
NotImplementedException	Thrown when an attempt is made to access an element of an array or collection with an index that is outside its bounds. It happens when a user search non-existence Id.	Ensure that Id index values used for accessing customers employees or products are within valid ranges. Catch the exception and handle it gracefully, possibly by entering correct unique Id.
DivideByZeroException	Occurs when attempting to divide an integer or decimal value by zero. This exception is thrown in calculations like if a customer has no reviews, and we want to calculate average reviews.	Ensure that denominators in division operations are not zero. Catch the exception and provide appropriate error handling, such as displaying a message to the user or using a default value as a fallback.

FormatException	Occurs when a string cannot be parsed into the desired format. Like if date and time is entered in wrong format then this exception is thrown.	Use appropriate parsing methods and ensure that input data conforms to expected formats. Provide user-friendly error messages and guidance for incorrect input formats.
PriceExceedsLimitError	Occurs when the price of a shipment product exceeds the price fixed by the admin.	Implement validation checks during product selection or checkout to ensure that the price of the shipment product does not exceed the price limit set by the admin. Display a clear error message informing the user that the selected product exceeds the allowed price limit. Provide options for the user to review and modify their selection or choose a different product.
InsufficientQuantityError	Occurs when the customer requests a quantity of a product that exceeds the available stock in the database.	Implement real-time inventory tracking and validation to ensure that the quantity requested by the customer does not exceed the available stock. Display an error message informing the customer that the requested quantity exceeds the available stock. Provide options for the customer to reduce the quantity or choose a different product.
DuplicateDataError	Occurs when attempting to add data (e.g., employee, customer, category, subcategory, discount) that already exists in the database.	Implement uniqueness constraints and validation checks to prevent duplicate entries at the database level. Additionally, utilize client-side checks to prevent users from submitting duplicate data. Notify the user about the attempted duplication and provide guidance on how to proceed. Offer options to review existing entries or correct the input before resubmitting.
ReportGenerationError	Occurs when there are issues with generating or exporting the Crystal Report to PDF format.	Ensure proper configuration and setup of Crystal Reports components and dependencies. Validate input parameters and data sources to prevent errors during report generation. Provide feedback to the user indicating that there was an issue with report generation.

Project Plan

Use Case Id	Use Case Name	Member Name	Estimated Completion Date
U01	Login	Saqlain Mansab	02-05-2024
U02	Update Credentials		
U03	Add Employees		
U04	Edit Employees		
U05	Add Customers		
U06	Edit Customers		
U07	Add Supplier		
U08	Edit Supplier	Zulqarnain Akram	03-05-2024
U09	Add Categories		
U10	Update Categories		
U11	Delete Categories		
U12	Employees Attendance		
U13	Add Discounts	Tabish Akhtar	04-05-2024
U14	Edit Discounts		
U15	Generate Reports		
U16	Place Order		
U17	Purchase Product		

Conclusion:

In conclusion, our project makes running a shop easier. It helps the admin manage everything like stock, sales, and the people who work there. The admin can set up different categories for products and give out discounts to attract more customers. Employees can also use the system to add new customers and suppliers, record what's sold, and keep track of who's at work. Our project also gives the boss helpful reports to understand how the shop is doing. With easy-to-use buttons and menus, anyone can use our system without any trouble.

Overall, our project is like a handy tool for shop owners, making it simpler to run their business and grow successfully.