# Програмування та тестування

## Розробка програми

Розроблена програма призначена лише для операційної системи Windows, тому програма несумісна з іншими операційними системами (наприклад: Linux, Debian, MacOS тощо).

Після запуску програми, користувач повинен буде ввести логін та пароль, після чого відкриється форма призначена для адміністратора або для менеджера з продажу в залежності від ролі користувача.

В середовищі MS-SQL Server було створено схему даних, яка зображена на рисунку 4.1.

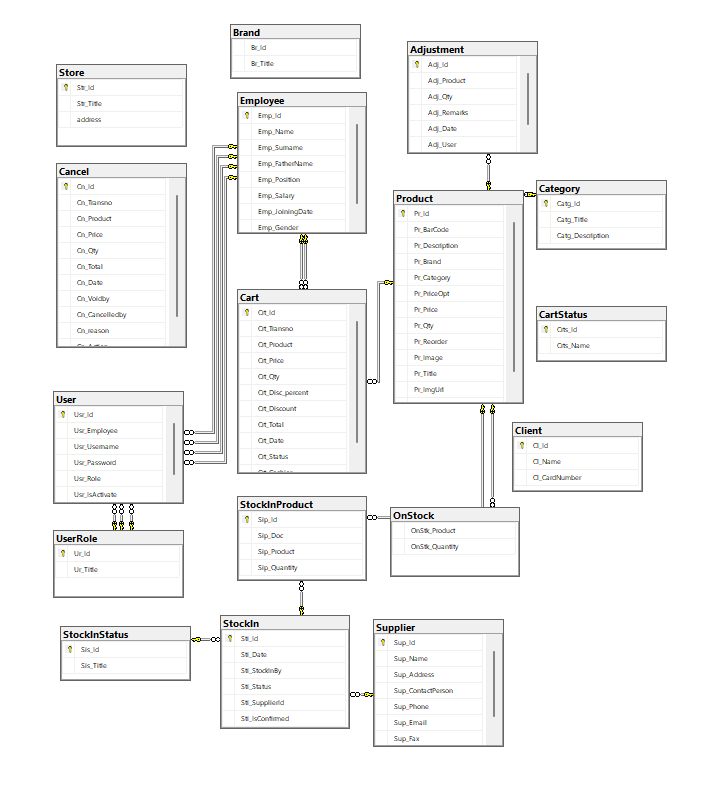


Рисунок 4.1 – Схема даних

Для переходу на інші форми, користувач повинен обрати потрібний пункт меню.

На головній формі адміністратора в меню доступні наступні пункти:

* Головна;
* Товар:
* Список товарів;
* Категорії;
* Виробник;
* Склад:
* Список замовлень;
* Списання
* Постачальники;
* Записи;
* Налаштування:
* Користувач;
* Оптова база;

Обравши пункт “Товар”, в меню з’являється 3 підпункти: Список товарів, Категорії, Виробник. Натиснувши на Список товарів, користувачеві відображається форма зі списком товарів та інформацією про нього при натискані певної кнопки. Натиснувши на Категорії, користувачеві відображається форма зі списком категорій. Натиснувши на Виробник, користувачеві відображається форма зі списком виробників товарів.

Обравши пункт “Склад”, в меню з’являється 2 підпункти: Список замовлень та Списання. Натиснувши на Список замовлень, користувачеві відображається форма зі списком замовлень. На цій формі також можна створити нове замовлення або редагувати існуючі. Натиснувши на Списання, користувачеві відображається форма для списання товарів.

Обравши пункт Постачальники, користувачеві відображається форма зі списком постачальників, на якій також можна зареєструвати нового постачальника.

Обравши пункт Записи, користувачеві відображається форма з усіма звітами та інструментами аналізу.

Обравши пункт “Налаштування”, в меню з’являється 2 підпункти: Користувач та Оптова база. Натиснувши на пункт Користувач, користувачеві відображається форма для налаштування всіх користувачів зареєестрованих в системі. Натиснувши на пункт Оптова база, користувачеві відображається форма для налаштування інформації про оптову базу.

На головній формі адміністратора в меню доступні наступні пункти:

* Нова транзакція;
* Знайти продукт;
* Карта клієнта;
* До сплати;
* Очисти кошик;
* Змінити пароль;

Натиснувши на пункт Нова транзакція на формі очищається список товарів, що були додані до кошика та створюється нова транзакція.

Обравши пункт Знайти продукт, користувачеві відображається форма зі списком продуктів, на якій він повинен обрати товар та підтвердити кількість, яка буде додана до кошика.

Обравши пункт Карта клієнта , користувачеві відображається форма для введення коду карти клієнта та реєстрації нової в випадку її відсутності. Карта клієнта надасть 5-ти відсоткову знижку на кожен продукт.

Обравши пункт До сплати, на екрані відобразиться форма, в якій потрібно ввести суму, яку надав клієнт, потім на формі підрахується решта. Після натиснення кнопки Підтвердити транзакція закриється та видрукується чек.

Обравши пункт Очистити кошик, на формі очиститься таблиця зі списком товарів в кошику, при цьому транзакція залишиться тією ж самою.

Обравши пункт Змінити пароль, касиру відобразиться форма на якій він може змінити пароль для свого профілю.

## Етапи налагодження

Налагодження - це процес виявлення причин виникнення помилки та її подальшого виправлення (на відміну від тестування, що є процесом виявлення самого факту існування помилки).

Процес Налагодження включає в себе:

* Вироблення системних вимог;
* Вироблення вимог до програмного забезпечення;
* Аналіз та проектування;
* Реалізацію - процес написання коду;
* Продуктизацію – процес встановлення відношення програмного забезпечення вимогам і доведення до них;
* Експлуатацію – процес використання програми.

Під час створення програмного продукту в середовищі Visual Studio використовувались інструменти для відлагодження програмного коду. Один з них - це відлагоджувальник, який дозволяє розробникам встановлювати точки зупинки в програмному коді та відстежувати значення змінних та виконання програмного коду по крокам. Відлагоджувальник також дозволяє розробникам виконувати розгалуження в програмному коді та виконувати відлагодження в певних умовах. Інші інструменти для відлагодження в Visual Studio включають в себе вікна «Вихідні теки» та «Вихідний вивід», які дозволяють розробникам переглядати вихідний код та відслідковувати виконання програмного коду. Вікна «Локальні» та «Глобальні» змінні дозволяють розробникам переглядати значення змінних під час виконання програмного коду.

## Типи помилок

Під час розробки програмного продукту можуть виникати різні типи помилок, які можуть бути класифіковані в кілька категорій. Нижче перераховано деякі з найпоширеніших типів помилок:

* Синтаксичні помилки: ці помилки виникають під час компіляції програмного коду та пов'язані з невірним використанням синтаксичних елементів мови програмування, таких як знаки пунктуації, ключові слова, типи даних тощо.
* Логічні помилки: ці помилки пов'язані з невірним виконанням програмного коду під час його виконання. Наприклад, програма може вивести неправильні результати через неправильні умови вибору, помилки у логіці обчислень або неправильний алгоритм.
* Помилки виконання: ці помилки виникають під час виконання програми та пов'язані з недостатнім обсягом ресурсів, таких як пам'ять або процесорний час, що призводить до аварійного завершення програми.
* Помилки взаємодії: ці помилки пов'язані з невірною взаємодією програмного продукту з іншими програмами, операційним середовищем або залежними компонентами. Наприклад, програма може взаємодіяти з базою даних, але через помилки підключення вона не може здійснювати запити.
* Помилки конфігурації: ці помилки пов'язані з неправильним налаштуванням програмного продукту або залежних компонентів, що можуть призводити до неправильної роботи програми або до аварійного завершення.
* Помилки безпеки: ці помилки пов'язані з недостатньою безпекою програмного продукту, що може призвести до несанкціонованого доступу до системи, крадіжки даних або вірусного атаки.

Кожен помилок має свої особливості та специфічні методи їх виявлення та виправлення. Для виявлення синтаксичних та логічних помилок використовують методи відлагодження та тестування. Для виявлення помилок виконання та взаємодії можуть використовуватись моніторинг, аналіз логів та діагностика проблем. Помилки конфігурації можуть бути виявлені за допомогою перевірки налаштувань системи та залежних компонентів. Для виявлення помилок безпеки можуть використовуватись методи тестування на проникнення та аудит безпеки програмного продукту.

Важливо виявляти та виправляти помилки якомога раніше в процесі розробки, оскільки чим пізніше помилка буде виявлена, тим важче її буде виправити. Тому важливо забезпечувати високу якість коду та правильну його організацію, а також використовувати ефективні методи виявлення та виправлення помилок.

Також під час розробки програмного продукту активно використовувався механізм try-catch. Це механізм обробки помилок, який дозволяє контролювати помилки в програмі та забезпечує їх відловлювання та обробку, щоб уникнути краху програми. Конструкція try включає блок коду, який потенційно може породити виключення або помилку, і блок catch, який оброблює ці виключення та надає інструкції щодо їх обробки. Наприклад, у випадку, коли ми виконуємо ділення на нуль, програма породить помилку DivisionByZeroException. Ми можемо використати конструкцію try-catch для перехоплення цієї помилки та виведення користувачеві повідомлення про помилку замість краху програми.

В ході розробки програмного забезпечення виникали деякі помилки різного типу. Приклад синтаксичної помилки в курсовому проекті (див. рис. 4.1)

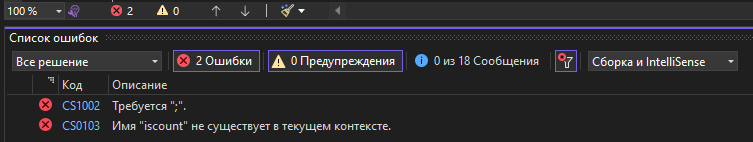


Рисунок 4.1 – Синтаксична помилка