Міністерство освіти і науки України

Харківський національний університет радіоелектроніки

Кафедра програмної інженерії

КУРСОВА РОБОТА

ПОЯСНЮВАЛЬНА ЗАПИСКА

з дисципліні “ Об’єктно-орієнтоване програмування ”

Довідник начальника в'язниці.

Керівник , проф. Бондарєв В. М.

Студент гр. ПЗПІ-22-6 Гармаєв М.С

Комісія:

Проф. Бондарєв В.М.,

Ст. викл. Черепанова Ю.Ю.,

Ст. викл. Ляпота В.М.

Харків 2023

ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ РАДІОЕЛЕКТРОНІКИ

Кафедра *програмної інженерії*

Рівень вищої освіти *перший (бакалаврський)*

Дисципліна *Об’єктно-орієнтоване програмування*

Спеціальність *121 Інженерія програмного забезпечення*

Освітня програма: *Програмна інженерія*

Курс 1 . Група *ПЗПІ-22 -6* . Семестр *2* .

***ЗАВДАННЯ***

***на курсовий проект студента***

***Гармаєв Михайло Саянович***

1. Тема проекту: *Склад*
2. Термін здачі студентом закінченого проекту: ***“27” - червня - 2023 р.***
3. Вихідні дані до проекту:

*Специфікація програми, методичні вказівки до виконання курсової*

1. Зміст розрахунково-пояснювальної записки:

*Вступ, опис вимог, проектування програми, інструкція користувача, висновки*

КАЛЕНДАРНИЙ ПЛАН

|  |  |  |
| --- | --- | --- |
| *№* | *Назва етапу* | *Термін виконання* |
| 1 | Видача теми, узгодження і затвердження  теми | 13.02.2023 – 14.03.2023 р. |
| 2 | Формулювання вимог до програми | 31.03.2023 – 03.04.2023 р. |
| 3 | Розробка підсистеми зберігання та пошуку даних. | 03.04.2023 – 05.05.2023 р. |
| 4 | Розробка функцій ….. | 05.05.2023 – 23.06.2023 р. |
| 5 | Розробка функцій зберігання та  завантаження даних | 14.06.2023 – 21.06.2023 р. |
| 6 | Тестування і доопрацювання розробленої  програмної системи. | 23.06.2023 – 25.06.2023 р. |
| 7 | Оформлення пояснювальної записки,  додатків, графічного матеріалу | 31.03.2023 – 27.06.2023 |
| 8 | Захист | 05.06.2023 – 16.06.2023 р. |

Студент гр. ПЗПІ-22-6 Гармаєв Михайло Саянович

Керівник Бондарєв Володимир Михайлович \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(Прізвище, Ім'я, По батькові)

« 13 » лютого 2023 р.

# РЕФЕРАТ

Пояснювальна записка до курсової роботи: 44 с., 10 рис., 3 табл., 2 додатки,

7 джерел.

ПОКУПЕЦЬ, СКЛАД, ЗВІТ, ООП, .NET, МОВА C#

Метою роботи є розробка програми «Склад», яка буде надавати користувачу можливість управління базою товарів, що зберігаються на складі, з метою ефективного контролю над обліком, надходженням і видачею товарів, з подальшим формування прибуткової та видаткової накладних.

В результаті отримана програма, що дозволяє ефективно управляти списком товарів на складі, зберігати характеристики такі як: найменування, одиниця виміру, ціна одиниці, кількість, дата останнього завезення. Завдяки цій програмі можна здійснювати наступні дії: додавання товару, реєстрація надходження і вигрузки товару, виводити інвентарну відомість.

В процесі розробки використано середовища Microsoft Visual Studio 2022, фреймворк Windows Forms, платформи .NET 7.0, мова програмування C#.

**ЗМІСТ**

ЗМІСТ

Вступ.........................................................................................................................6

1. ОПИС ВИМОГ ДО ПРОГРАМИ......................................................................7
   1. Сценарії.........................................................................................................7
   2. Функції........................................................................................................10
2. ПРОЄКТУВАННЯ ПРОГРАМИ.....................................................................16
3. ІНСТРУКЦІЯ КОРИСТУВАЧА.....................................................................20
   1. Встановлення..............................................................................................20
   2. Аутентифікація...........................................................................................21
   3. Пошук..........................................................................................................22
   4. Видалення в’язня........................................................................................23
   5. Додавання в’язня........................................................................................24

Висновок................................................................................................................27

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ.....................................................................28

Додаток А...............................................................................................................29

**ВСТУП**

Завдання:

**Склад.** База товарів, що зберігаються на складі: найменування, одиниця виміру, ціна одиниці, кількість, дата останнього завезення. Реєстрація надходження товару (формування прибуткової накладної) і відгрузки (видаткова накладна). Вивід інвентарної відомості.

Метою роботи є створення програми для власників складів з товарами для зручного та ефективного управління складським інвентарем. Основні завдання включають:

1. Система управління товаром: Розробити функціонал, який дозволяє додавати нові товари до бази даних, зберігати їх характеристики (найменування, одиниця виміру, ціна одиниці тощо) та відстежувати кількість товару на складі.
2. Облік надходження товару: Розробити функціонал, що дозволяє реєструвати надходження товару на склад. Такий облік допоможе власникам складів відстежувати та контролювати постачання товару.
3. Облік видачі товару: Розробити функціонал, що дозволяє реєструвати видачу товару зі складу. Такий облік допоможе власникам складів контролювати витрати товару та зберігати відповідний облік.
4. Генерація звітів та інформаційних відомостей: Розробити функціонал, який дозволяє генерувати звіти та інформаційні відомості про стан складського інвентарю.

В результаті, отримана програма, користувачем якої буде власник складу з товарами, дозволить ефективно управляти інвентарем, точно відстежувати наявність товарів, контролювати постачання та видачу товару, а також отримувати необхідні звіти (прибуткова та видаткова накладні).

**1 ОПИС ВИМОГ ДО ПРОГРАМИ**

**Сценарій 1 «Реєстрація надходження товару»**

*Передумови:*

Користувач пройшов аутентифікацію, відкрив головне вікно програми і вибрав форму "Реєстрація надходження товару". База товарів містить інформацію про доступні товари.

*Основний сценарій:*

1. Користувач вибирає товар зі списку або вводить його назву у відповідне поле.
2. Користувач вказує кількість товару, що надійшла на склад.
3. Користувач натискає на кнопку «Додати» для запису товару у список для формування прибуткової накладної.
4. Програма перевіряє наявність вибраного товару в базі даних.
5. Програма оновлює інформацію про кількість товару в базі даних.
6. Програма відображає повідомлення про успішну реєстрацію надходження товару.

*Додатковий сценарій:*

1. Користувач натискає кнопку «Додати новий товар» для реєстрації надходження нового товару на склад.
2. Користувач послідовно вводить назву, ціну та дату завезення.
3. Користувач натискає на кнопку «Додати» для запису товару у список для формування прибуткової накладної.

*Альтернативні сценарій:*

1. Вибраний товар не знайдено в базі даних.
2. Система відображає повідомлення про помилку і просить користувача перевірити правильність вибраного товару.
3. Користувач може повторити кроки 1-3, вибравши інший товар.

**Сценарій 2 «Вигрузка товару»**

*Передумови:*

Користувач пройшов аутентифікацію, відкрив головне вікно програми і вибрав форму " Вигрузка товару ". База товарів містить інформацію про доступні товари.

*Основний сценарій:*

1. Користувач вибирає товар зі списку або вводить його назву у відповідне поле.
2. Користувач вказує кількість товару для відвантаження.
3. Користувач натискає кнопку "Додати" для реєстрації відвантаження товару.
4. Система перевіряє наявність вибраного товару в базі даних та достатність його кількості на складі.
5. Система оновлює інформацію про кількість товару в базі даних.
6. Система відображає повідомлення про успішне відвантаження товару.

*Додатковий сценарій:*

1. Користувач натискає кнопку «Видалити товар» для видалення товару зі складу.
2. Користувач вводить або вибирає зі списку назву товару.
3. Користувач натискає на кнопку «Видалити» для видалення запису товара.

*Альтернативний сценарій:*

1. Вибраний товар не знайдено в базі даних або його кількості на складі недостатньо.
2. Система відображає повідомлення про помилку і просить користувача перевірити правильність вибраного товару або кількість товару на складі.
3. Користувач може повторити кроки 1-3, вибравши інший товар або зменшивши кількість для відвантаження.

**Сценарій 3 «Вивід інвентарної відомості»**

*Передумови:*

Користувач пройшов аутентифікацію, відкрив головне вікно програми і вибрав форму "Вивід інвентарної відомості". База товарів містить інформацію про доступні товари.

*Основний сценарій:*

1. Користувач вибирає дату для отримання інформації про останнє надходження товарів у цей день.
2. Користувач натискає кнопку "Пошук".
3. Система перевіряє наявність інформації про завезення товару для обраної дати.
4. Система відображає інформацію про надходження товарів на обрану дату.

*Альтернативні сценарій:*

1. Система відображає повідомлення про відсутність даних і просить користувача обрати іншу дату.
2. Користувач може повторити крок 1, вибравши іншу дату для отримання інформації про інвентарну відомість.

**Функція 1 «Автентифікація»**

Користувач запускає програму і відкривається форма аутентифікації (Рис. 1.1 – Автентифікація)

Форма аутентифікації містить два поля - для введення логіна і пароля.

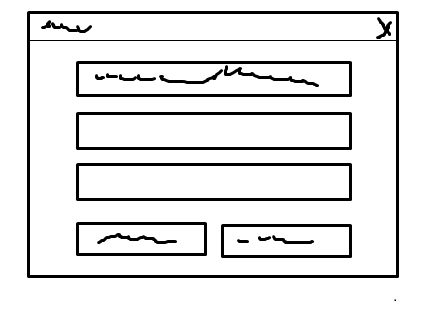


Рис. 1.1 – Автентифікація

Користувач вводить свій логін (максимальна довжина 11 символів) і пароль (максимальна довжина 6 символів) у відповідні поля.

Форма має дві кнопки: «Увійти» та «Скасувати».

Користувач натискає кнопку «Увійти» для підтвердження введеного логіна і пароля.

Якщо введений логін і пароль є правильними, програма відкриває основну форму програми.

Якщо введений логін або пароль неправильний, виводиться повідомлення про помилку і користувач повинен ввести коректні дані або натиснути кнопку «Скасувати».

Якщо користувач натискає кнопку «Скасувати», відкривається повідомлення для підтвердження дії, якщо користувач натискає «Да» форма аутентифікації закривається і програма припиняє свою роботу, але якщо користувач натискає «Нет» программа продовжує свою роботу.

Користувач має лише 5 спроб для введення логіна та пароля.

При кожній невірній спробі программа буде виводити помилку «Невірний логін або пароль. Кількість спроб n», де n кількість спроб, що залишились. Після користувач натискає на «Ок» і може продовжити вводити дані для входу.

Якщо залишилась одна спроба програма видасть помилку «У вас залишилась 1 спроба. Після неї у вас не буде можливості увійти». Після цього користувач натискає на «Ок» і має останню спробу ввести правильну комбінацію логіна та паролю.

Якщо усі спроби вичерпано, програма видасть повідомлення «Ви вичерпали ліміт спроб. Виконання програми буде припинено» і припинить свою роботу.

**Функція 2 «Головна»**

Після успішної автентифікації відкривається головна панель – форма з трьома кнопками: «Реєстрація надходження товару», «Вигрузка товару», «Вивід інвентарної відомості» та картинкою складу (Рис. 1.2 – Головна)

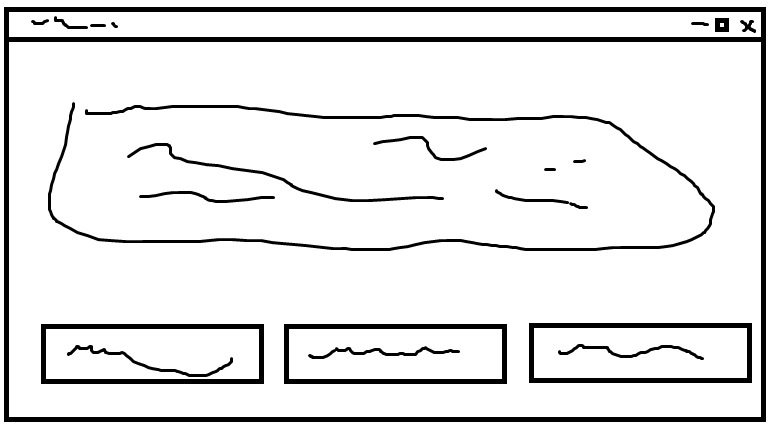


Рис. 1.2 – Головна

Кнопка «Реєстрація надходження товару», за допомогою якої користувач може зареєструвати прибуття нового товару до бази даних.

Кнопка «Вигрузка товару», за допомогою якої користувач може відвантажити товар із бази даних.

Кнопка «Вивід інвентарної відомості», за допомогою якої користувач може отримати вивід інвентарної накладної для відображення докладної інформації про товари та їх рух у системі.

**Функція 3 «Реєстрація надходження товару»**

Панель «Додавання товару» – форма для реєстрації надходження товару(Рис. 1.3 – Додавання товару). Користувач натискає на кнопку «Реєстрація надходження товару» з головного меню програми.

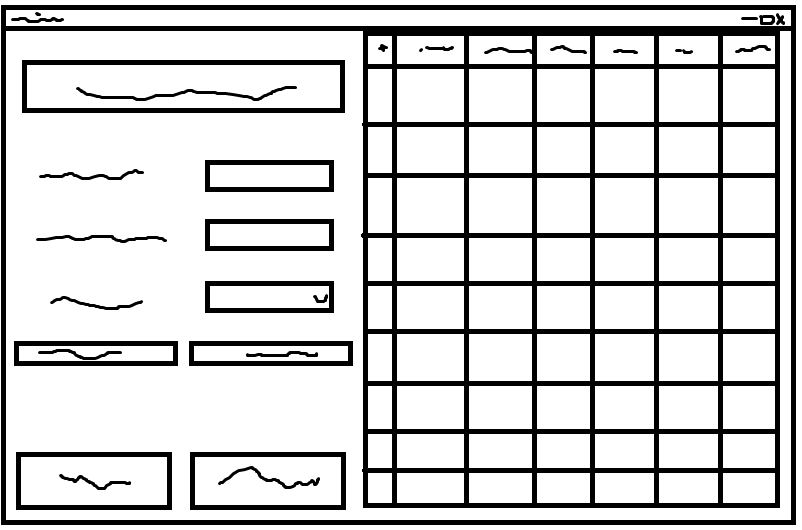


Рис. 1.3 – Додавання товару

Користувач вводить назву товару у полі введення «Введіть назва товару». Пише або вибирає назву зі списку.

Якщо товару з такою назвою не було знайдено виводиться помилка: «Оберіть товар зі списку або введіть правильну назву»

Потім вказує кількість одиниць товару в полі введення «Введіть кількість товару».

Якщо користувач ввів дробове, від’ємне число або текст/символи програма видасть помилку: «Введене значення не є додатнім або значення невірне. Будь ласка, спробуйте ще раз».

Вибирає дату в полі Введіть дату реєстрації»

Якщо обрана дата є меншою за сьогоднішню програма видасть помилку: «Обрана дата не може бути меншою за сьогоднішню»

Ці повідомлення виводяться при спробі користувача додати товар, натискаючи на кнопку «Додати», яка обробляє занесення товару в список для формування прибуткової накладної та для збереження змін у базу даних.

Якщо користувач ввів усі поля правильно і натискає на кнопку «Додати», з’явиться вікно попередження «Ви впевнені у коректності введених даних? Ви не зможете змінити інформацію», задля того щоб користувач перевірив правильність введених даних. При цьому натискаючи на «Отмена» усі введені дані залишаються, але при натисканні на «Ок» відкриється повідомлення «Товар успішно додано». Після натискання на «Ок» у вікні останнього повідомлення дані про товар будуть занесені у список та збережуться в базу.

На формі є додаткова кнопка «Скасувати», яка очистить поля.

Якщо користувач хоче додати новий товар, якого ще не було на складі, він натискає на кнопку «Додати новий товар».

Відкриється панель – форма для занесення інформації про новій товар(Рис. 1.4 – Додавання нового товару).

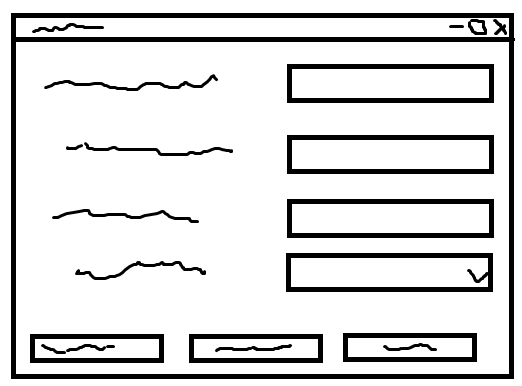


Рис. 1.4 – Додавання нового товару

Користувач вводить назву товара, якщо назва співпадає з назвою товару, що вже зберігається на складі виведеться помилка «Товар з таким ім’ям уже існує. Введіть іншу назву» та при натисканні на «Ок» усі поля залишаться заповненими.

Після введення назви користувач вводить ціну та кількість товару, а також дату реєстрації. Усі перевірки і відповідні помилки ідентичні з додаванням вже існуючого товару.

Якщо якесь з полів не заповнено, а користувач натискає на кнопку «Додати» з’явиться помилка «Будь ласка, заповніть усі поля».

Також є кнопка «Скасувати» для швидкого очищення полів.

Якщо користувач ввів усі поля правильно і натискає на кнопку «Додати», з’явиться вікно попередження «Ви впевнені у коректності введених даних? Ви не зможете змінити інформацію», задля того щоб користувач перевірив правильність введених даних. При цьому натискаючи на «Отмена» усі введені дані залишаються, але при натисканні на «Ок» відкриється повідомлення «Товар успішно додано». Після натискання на «Ок» у вікні останнього повідомлення дані про товар будуть занесені у список та збережуться в базу. А поля очистяться автоматично.

Якщо користувач додав усі нові товари або хоче повернутися на попередню форму – він натискає на кнопку «Повернутися». При чому, якщо користувач вірно увів усі поля, але не додав їх (не натиснув на кнопку «Додати») програма видасть кікно підтвердження «Дані не збережено. Ви впевнені?» якщо користувач натискає «Да» то він повертається назад, в іншому випадку поля залишаються заповненими.

Після додавання усіх товарів користувачу потрібно натиснути на кнопку «Сформувати накладку». Якщо до цього не було додано жодного товару, при натисканні на цю кнопку відкриється помилка «Додайте хоча б один товар перед підтвердженням». В іншому випадку відкриється форма «Прибуткова накладка» (Рис. 1.5 – Прибуткова накладна), де в таблиці буде виведена уся інформація про доданий товар, порахована загальні ціні закупки кожного з товарів та ціна усієї поставки, а також дата.

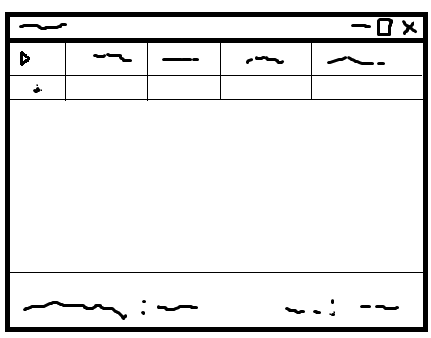


Рис. 1.5 – Прибуткова накладна

Перевірка підтвердження після натискання на кнопку «Сформувати накладку» не має сенсу, оскільки користувач перед додаванням вже декілька разів підтверджував додавання. Але за потреби окремих користувачів таку перевірку можна зробити.

Після натискання на хрестик у формі «Прибуткова накладна» відкривається головне меню.

Також у формі «Додавання товару» є кнопка «Повернутися в головне меню» при натисканні на неї користувач переходить в головне меню програми.

**Функція 4 «Вигрузка товару»**

Панель «Вигрузка товару» – форма для відвантаження товару(Рис. 1.6 – Вигрузка товару).

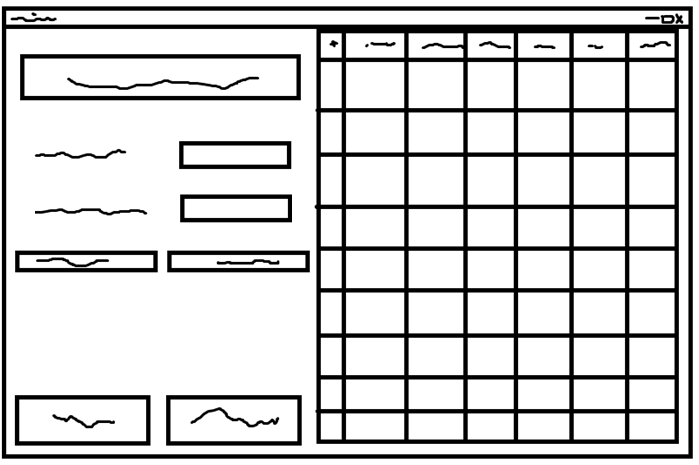


Рис. 1.6 – Вигрузка товару

Користувач вводить назву товару у полі введення «Введіть назва товару». Пише або вибирає назву зі списку.

Якщо товару з такою назвою не було знайдено виводиться помилка: «Оберіть товар зі списку або введіть правильну назву»

Потім вказує кількість одиниць товару в полі введення «Введіть кількість товару».

Якщо користувач ввів дробове, від’ємне число або текст/символи програма видасть помилку: «Введене значення не є додатнім або значення невірне. Будь ласка, спробуйте ще раз». Також программа перевіряє чи достатньо товару для вигризки, якщо введена кількість більша ніж є на складі з’явиться відповідна помилка.

Якщо користувач ввів усі поля правильно і натискає на кнопку «Додати», з’явиться вікно попередження «Ви впевнені у коректності введених даних? Ви не зможете змінити інформацію», задля того щоб користувач перевірив правильність введених даних. При цьому натискаючи на «Отмена» усі введені дані залишаються, але при натисканні на «Ок» відкриється повідомлення «Товар успішно додано». Після натискання на «Ок» у вікні останнього повідомлення дані про товар будуть занесені у список та збережуться в базу. А поля очистяться автоматично.

На формі є додаткова кнопка «Скасувати», яка очистить поля та кнопка «Повернутися в головне меню» при натисканні на неї користувач переходить в головне меню програми.

Якщо користувач хоче видалити товар з бази даних він натискає на кнопку «Видалити товар». Відкривається форма «Видалення товару» (Рис 1.7 – Видалення товару) користувач вводить назву і натискає кнопку «Видалити» відкриється попередження «Ви впевнені у коректності введених даних. Ви не зможете змінити інформацію» при натисканні на «Ок» товар буде виделено з бази даних, якщо натиснуто «Отмена» введена назва збережется.

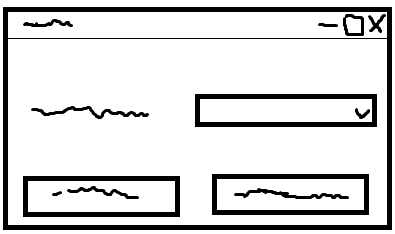


Рис 1.7 – Видалення товару

Якщо користувач ввів неправильну назву чи не вибрав товар і натиснув «Видалити» відкриється вікно помилки «Будь ласка, виберіть назву для видалення або змініть назву введеного товару»

Після видалення відкриється інформаційне повідомлення «Запис успішно видалено». Щоб повернутися назад користувач натискає кнопку «Повернутися».

Після додавання усіх товарів користувачу потрібно натиснути на кнопку «Сформувати накладку». Якщо до цього не було додано жодного товару, при натисканні на цю кнопку відкриється помилка «Додайте хоча б один товар перед підтвердженням». В іншому випадку відкриється форма «Видаткова накладка» (Рис. 1.8 – Видаткова накладна), де в таблиці буде виведена уся інформація про доданий товар, порахована загальні ціні вигрузки кожного з товарів та ціна усієї вигрузки, а також дата вивезення товару.

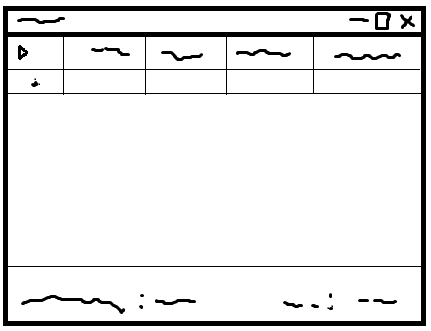


Рис. 1.8 – Видаткова накладна

**Функція 5 «Вивід інвентарної відомості»**

Користувач натискає на кнопку «Вивід інвентарної відомості» у головній формі.

Після цього відкриється панель – форма «Інвентарна відомість» (Рис. 1.9 – Інвентарна відомість)

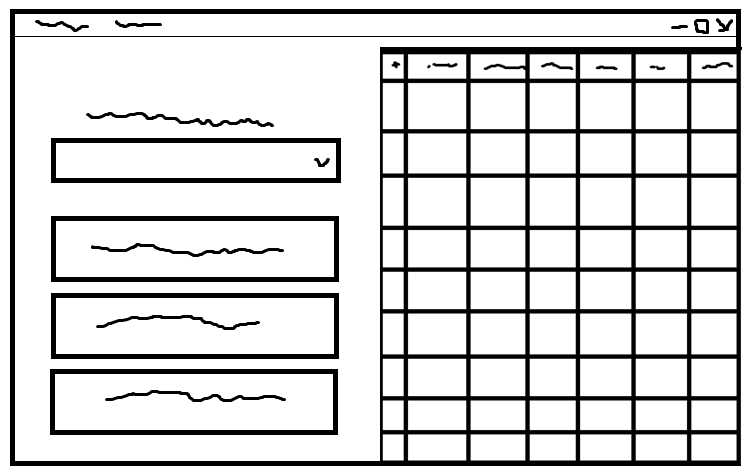


Рис. 1.9 – Інвентарна відомість

При виклику функції "Вивід інвентарної відомості" програма взаємодіє з базою даних, отримує дані про товари та їх кількість. Далі формується звіт, який може бути відображений на екрані.

Звіт інвентарної відомості містить такі поля для кожного товару:

* Унікальний, автоінкрементний Id для можливості в подальшому розширювати функціонал програми
* Назва товару: інформація про назву конкретного товару на складі.
* Одиниця виміру: одиниці одні – кілограми, але за потреби можна розширити функціонал.
* Кількість: кількість одиниць товару, що наявна на складі.
* Дата останнього надходження: дата, коли останній раз товар був доданий на склад.

Цей звіт надає користувачу повну інформацію про стан інвентаря, що допомагає управляти запасами, контролювати наявність товарів і приймати відповідні рішення щодо замовлення та розподілу товарів.

Є кнопка пошуку, користувач вибирає дату і программа шукає останнє завезення товару. Якщо записів на цю дату не знайдено виведеться повідомлення «Записів не знайдено. Виберіть іншу дату», в іншому випадку програма напише «Знайдено n записів», де n – кількість записів. Користувач побачить лише ті товари, що мали вказану користвучем дату.

Також на формі є кнопка «Скинути» для виведення даних за увесь час.

Якщо користувач хоче повернутися в головне меню є відповідна кнопка.

**2 ПРОЄКТУВАННЯ ПРОГРАМИ**

В програмі «Склад» буде використовуватися настільна архітектура з GUI на основі фреймворку Windows Forms на мові .Net 7.0. Структура програми буде виглядати наступним чином: головна форма, форма автетифікації, форма реєстрації товару, форма вигризки товару та форма виводу інвентарної відомості.

Усі дані будуть зберігатися у базі даних SQL, що складається з однієї таблиці над якою будуть проводитися дії. Для роботи з базою даних буде використовуватися MySQL Workbench 6.3 CE.

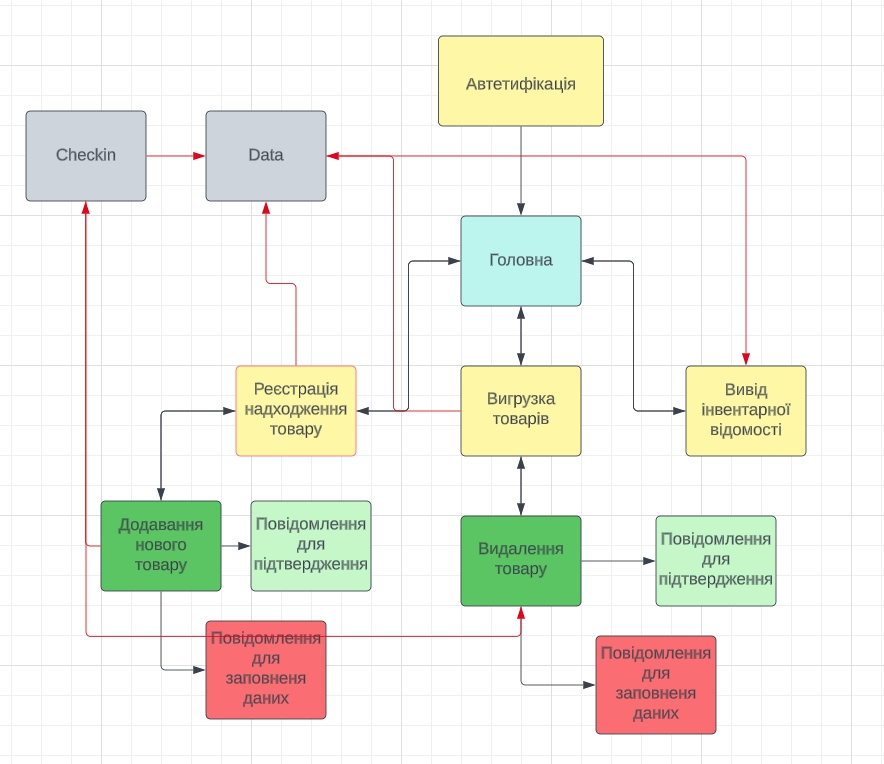


Рисунок 2.1 – UML-діаграма для покажу зв’язків програми

Клас Checkin:

Має наступні властивості:

Id: int

Name: string

Units: string

Price: int

Quantity: int

LastTimeDelivery: DateTime

Клас Context:

Наслідує клас DbContext.

Має наступні властивості:

Checkin: DbSet<Checkin>

Методи та функції:

OnConfiguring(DbContextOptionsBuilder optionsBuilder): перевизначений метод для налаштування підключення до бази даних.

Клас ProductInfo:

Має наступні статичні властивості:

AddedProducts: List<ProductInfo> - список доданих продуктів.

Має наступні властивості:

Name: string

Quantity: int

Price: int

DeliveryDate: DateTime

Методи та функції:

ClearAddedProducts(): очищує список доданих продуктів.

Структура зв'язків між класами:

Клас Context має властивість Checkin типу DbSet<Checkin>, що представляє зв'язок агрегації з класом Checkin.

Клас ProductInfo має статичну властивість AddedProducts типу List<ProductInfo>, що представляє список доданих продуктів.

У основних формах GoodsUpdating, GoodsUploading та InventoryListForm кокористовується екземпляри класу Context для роботи з базою даних. А також екземпляр класу ProductInfo для збереження товарів у список для виводу накладних.

**3 ОПИС ІНСТРУКЦІЇ КОРИСТУВАЧА**

1. **Встановлення програми на комп’ютер**

Щоб програма коректно виконувалася необхідно встановити на ПК середу розробки Visual Studio 2022 з офіційного веб-сайту та інструмент для візуального проектування баз даних MySQL Workbench версії 5.7.21. Для нормального працювання програми слід встановити ці інструменти саме вказаних версій.

Після встановлення Visual Studio та MySQL Workbench треба скачати проект собі на ПК з гугл диску (посилання на диск знаходиться у Додатку А)

Тепер локально треба зробити імпорт бази даних. Файл для імпорту знаходиться за шляхом: “Stock” –> “DataBase” –> “ stock\_checkin.sql ”

Відкривши та увійшовши у MySQL Workbench виконати наступні кроки:

1. Зверху зліва в панелі натиснути “Server”

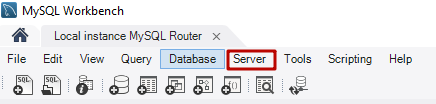


Рисунок 3.1.1 – Шлях до імпорту файлу

1. Натиснути “Data Import”

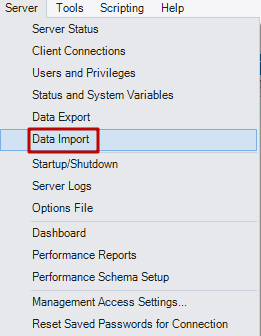
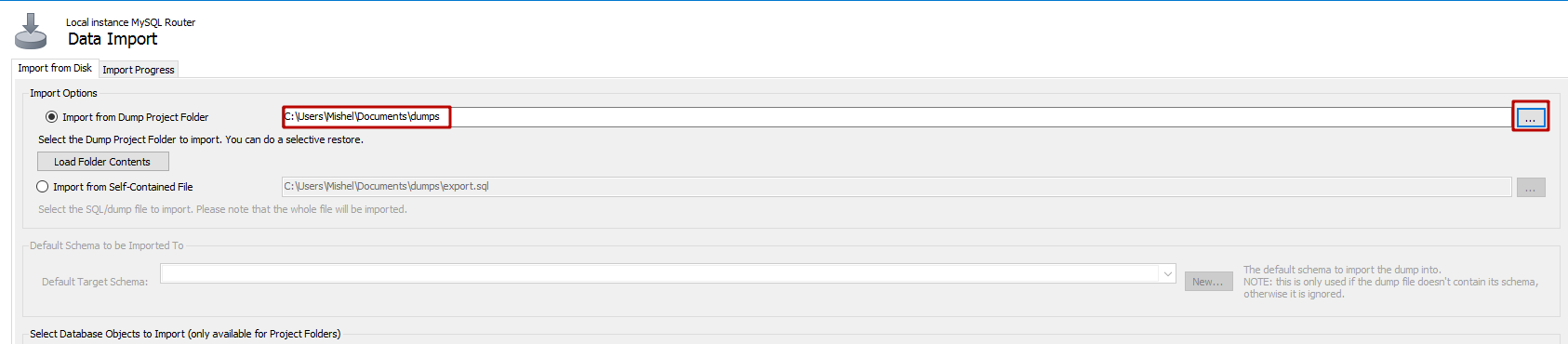


Рисунок 3.1.2 – Вибір імпорту даних

1. Вказати шлях до файлу з базою даних або вибрати файл через «Обзор папок»

Рисунок 3.1.3 – Вибір шляху до файла з базою даних

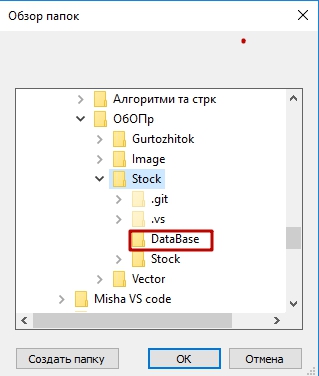


Рисунок 3.1.4 – Вибір папки з файлом через «Обзор папок»

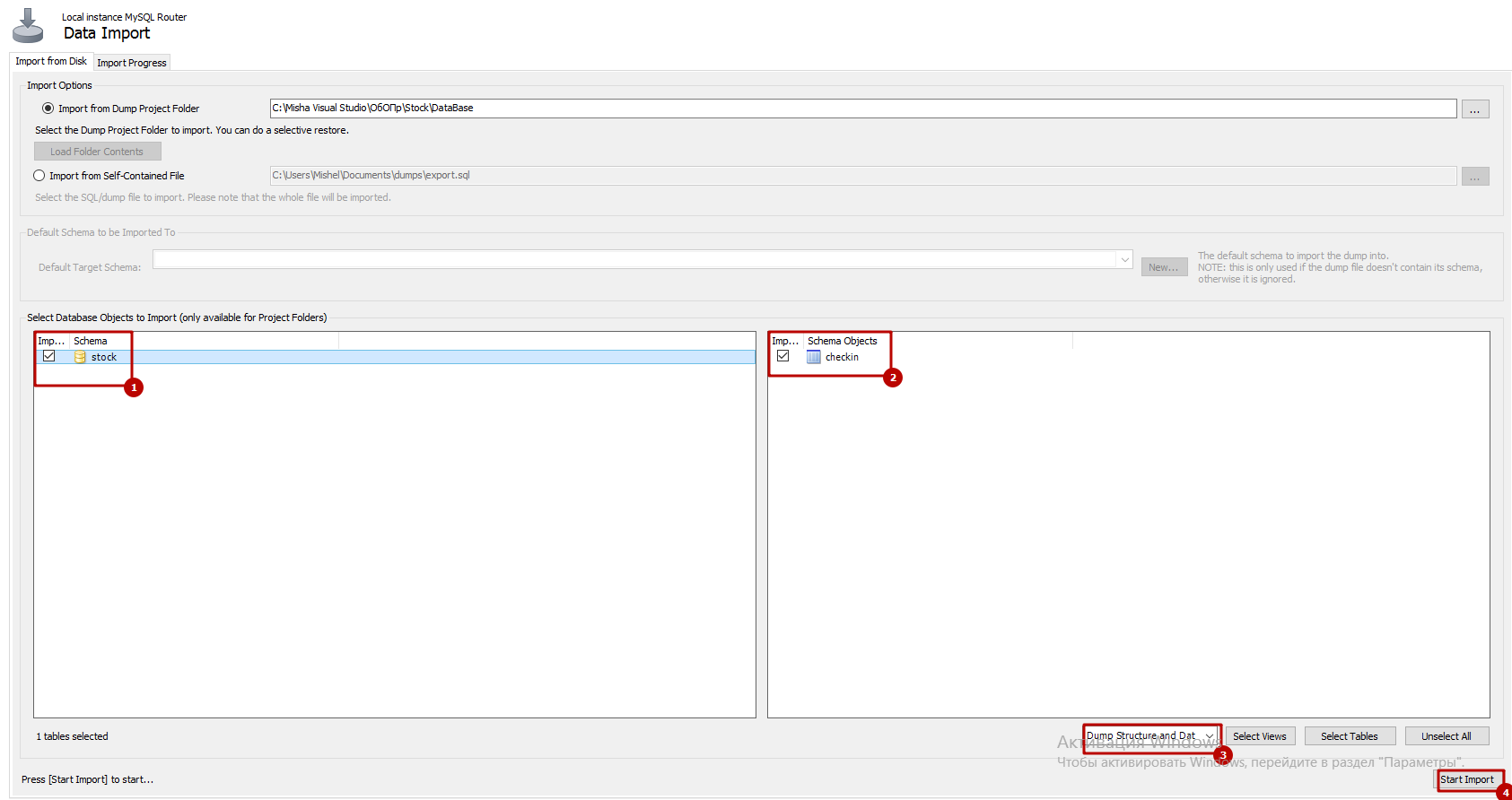
1. Після встановлення шляху двічі натискаємо на назву “stock” (1) та перевіряємо вибір таблиці “checkin” (2). Перевіряємо, що імпортується структура та дані (3) та натискаємо “Start import” для початку імпортування (4).

Рисунок 3.1.5 – Завершення імпорту

1. В результаті імпорт повинен пройти успішно і на екрані з’явиться відповідне повідомлення:

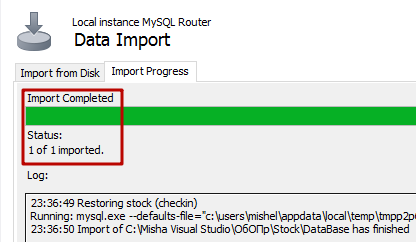


Рисунок 3.1.6 – Успішний імпорт бази даних

Зауваження 1 – ні в якому разі не можна змінювати файл “ stock\_checkin.sql ” це деже важливо для коректного імпорту бази даних

Зауваження 2 – Щоб подивитися зміст рисунків 3.1.3 та 3.1.5 збільшіть масштаб документа записки.

Тепер на вашому ПК є база даних для коректної роботи програми.

Щоб запустити програму слід повернутися до папки “Stock” та піти за шляхом “Stock” – “Stock” – “bin” – “Debug” – “net7.0-windows” – “Stock.exe” та двічі натиснути на вибраний файл програми.

Після цих дій відкриється програма і можна працювати.

1. **Автентифікація**

Для автентифікації треба ввести логін та пароль. Цей логін та пароль знає лише власник складу. На рисунку 3.2.1 зображено зовнішній вид вікна.

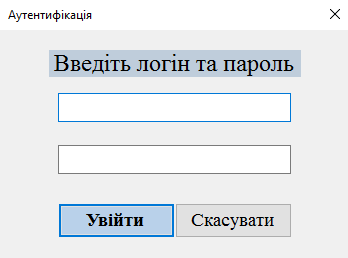


Рисунок 3.2.1 – Зовнішній вигляд вікна аутетифікація

Припустимо, я власник складу тому я знаю логін та пароль, а тому у відповідні поля (верхнє поле логін, нижнє пароль) вношу логін «Stock owner» та пароль «123456», натиснувши на «Ок» відкриється головне меню (Рис. 3.2.2 – Головне меню).

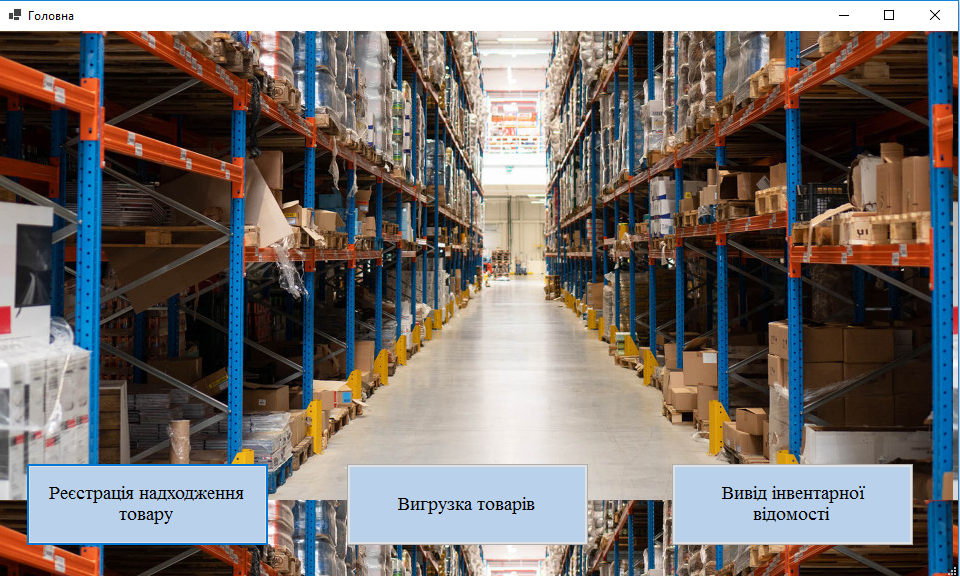


Рисунок 3.2.2 – Головне меню

1. **Реєстрація надходження товару**

Щоб зареєструвати товар необхідно натиснути на кнопку «Реєстрація надходження товару» та заповнити усі поля. Приклад наведено на рисунку 3.3.1. Натиснути на кнопку «Додати» (Рис. 3.3.2 – Додавання товару) та підтверджуємо (Рис. 3.3.3 –Попередження) відкриється вікно успішно доданого товару (Рис. 3.3.4 – Підтвердження додавання). Якщо поля заповнені невірно будуть випливати помилки (рисунки 3.3.5, 3.3.6, 3.3.7)

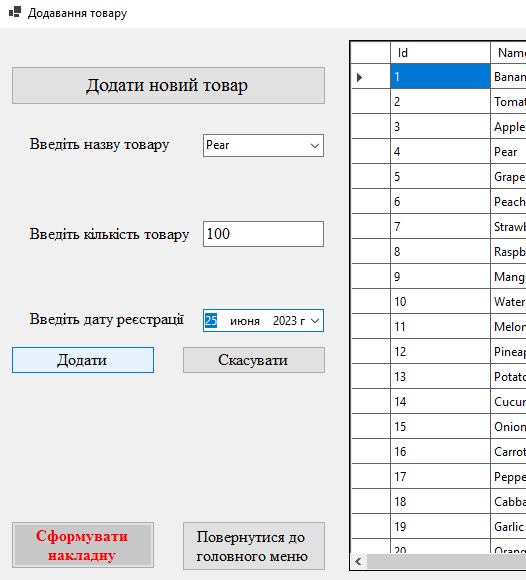


Рисунок 3.3.2 – Додавання товару

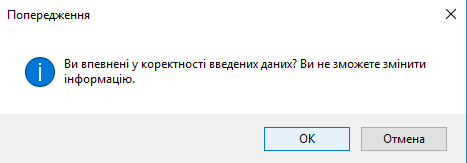


Рисунок 3.3.3 – Попередження

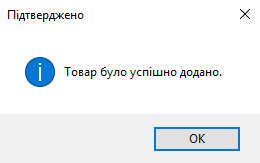


Рисунок 3.3.4 – Підтвердження додавання

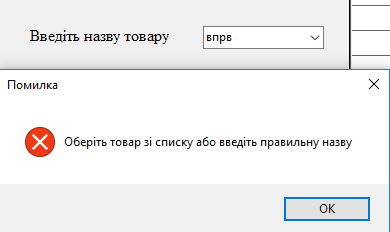


Рисунок 3.3.5 – Помилка при неправильній назві

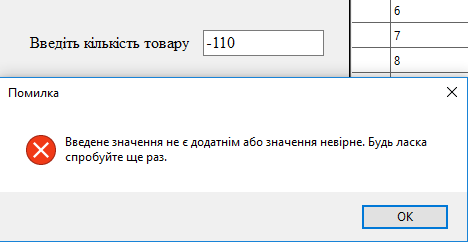


Рисунок 3.3.6 – Помилка при неправильній кількості

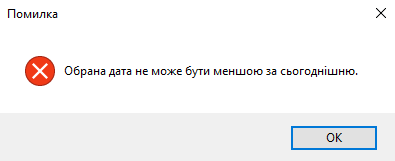


Рисунок 3.3.7 – Помилка при невірній даті

Розглянемо додавання нового товару, для цього слід натиснути на відповідну кнопку та заповнити усі поля (Рис 3.3.8 – Додавання нового товару).

* Назва товару (регістр не важливий)
* Ціна товару
* Кількість товару
* Дата завезення

Якщо усі поля було заповнено вірно і програма не знайшла товару з такою назвою відкриється вікно «Попередження» та «Підтвердження додавання» (див. рис. 3.3.3 та 3.3.4)

А також на рисунку 3.3.9 зображена помилка, коли поля не заповнені. Усі інші помилки ідентичні з функцією Додавання вже існуючого товару.

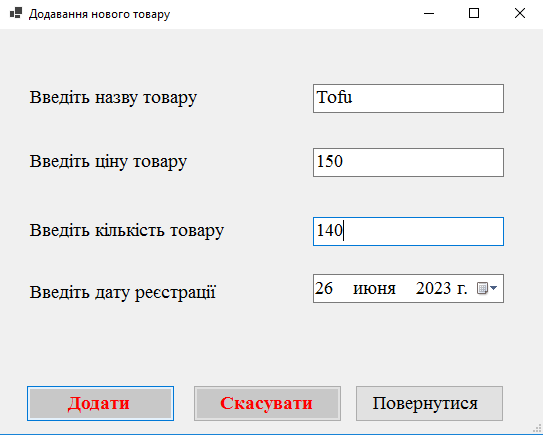


Рисунок 3.3.8 – Додавання нового товару

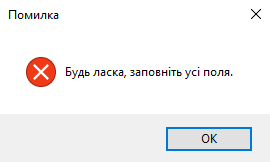


Рисунок 3.3.9 – Помилка для заповненя усіх полів

Після додавання усіх товарів сформуємо прибуткову накладку, натиснувши на відповідну кнопку (Рис. 3.3.10 – Формування прибуткової накладної).

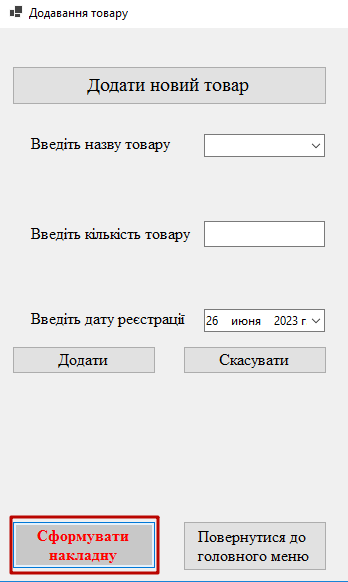


Рисунок 3.3.10 – Формування прибуткової накладної

Зовнішній вид отриманої накладної зображено на рисунку 3.3.11:

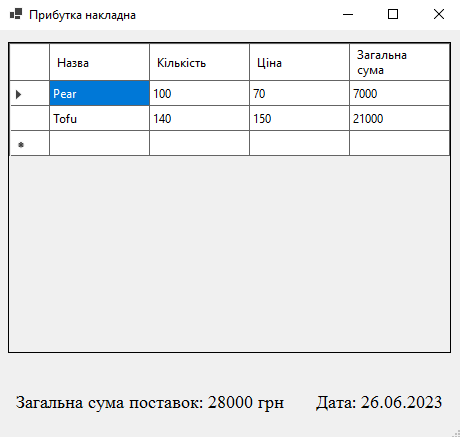


Рисунок 3.3.11 – Прибуткова накладна

1. **Вигрузка товару**

Перейшовши до відповідного меню можна приступити до заповненя полів для вигризки товарів зі складу. Натиснувши на кнопку «Додати» товар потрапляє до списку, який виведеться при натисканні на кнопку «Сформувати накладну». Помилки та підтвердження для вигризку товару ідентичні з минулим пунктом.

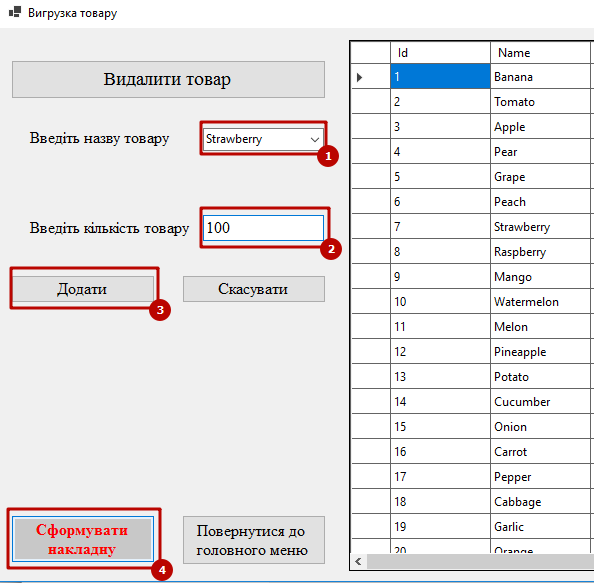


Рисунок 3.4.1 – Вигрузка товарів. Формування видаткової накладної

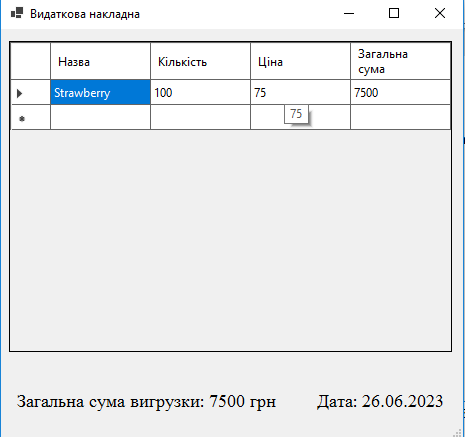


Рисунок 3.4.2 – Видаткова накладна

Зауваження: якщо товар не було додано до списку, програма видасть помилку:

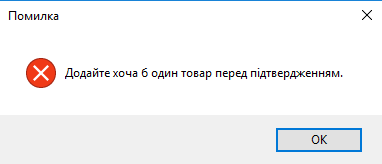


Рисунок 3.4.3 – Помилка при формуванні накладної

Розглянемо видалення товару з бази даних, для цього натиснемо відповідну кнопку у вікні «Вигрузка товару».

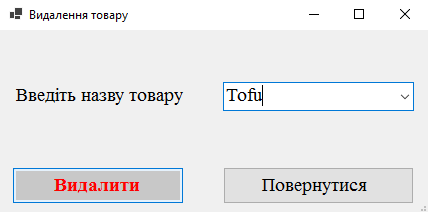


Рисунок 3.4.4 – Видалення товару

Вибравши товар зі списку або при введені правильної назви, натискаємо на видалити та підтверджуємо дію натискаємо «Ок»:

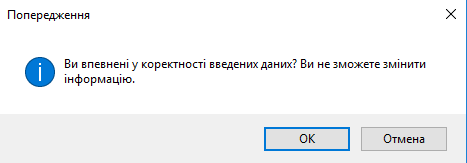


Рисунок 3.4.5 – Підтвердження видалення

В результаті видалення отримаємо повідомлення:

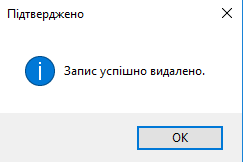


Рисунок 3.4.6 – Успішне видалення

Таким чином можна видаляти товар з бази або робити вигризку товару зі складу.

1. **ВИСНОВОК**

В результаті виконаної роботи отримано програму, що дозволяє керувати складом. Було реалізовано завдання та виконано усі умови, зроблені усі перевірки, написано функції додавання, видалення, оновлення та читання. Написано оптимальний код та використано майже найкраще рішення щодо зберігання даних. В ході роботи було описано всі можливі сценарії та їх альтернативи. Побудовано структуру проекту. Всі зміни вносилися на репозиторій для відстеження ходу роботи.

1. **ПОСИЛАННЯ НА ДЖЕРЕЛА**
2. .NET documentation. /Microsoft Learn. URL: <https://learn.microsoft.com/en-us/dotnet/> (дата звернення: 13.02.2023).
3. MySQL Workbench /URL: <https://www.mysql.com/products/workbench/> (дата звернення 03.04.2023)
4. Database First Approach In Entity Framework URL: <https://www.c-sharpcorner.com/article/database-first-approach-in-entity-framework/> (дата звернення: 15.04.2023)
5. C# programming. Lesson 16. Entity Framework. Part 1 URL: <https://www.youtube.com/watch?v=X4XRI8ZGsDo> (дата звернення: 20.04.2023)
6. Подключение к существующей базе данных URL: <https://metanit.com/sharp/entityframeworkcore/1.3.php> (дата звернення: 28.04.2023)
7. Entity Framework - Insert Update and Delete in C# URL: <https://www.youtube.com/watch?v=hg3H_pAzoPI> (дата звернення: 13.05.2023, 20.05.2023)
8. MySQL URL: <https://metanit.com/sharp/entityframeworkcore/7.2.php> (дата звернення: 01.06.2023)
9. C# - Entity Framework CRUD Operation - Part 01 URL: <https://www.youtube.com/watch?v=8xPzjYBYu-M> (дата звернення: 15.06.2023)
10. What is CRUD? URL: <https://www.crowdstrike.com/cybersecurity-101/observability/crud/#:~:text=CRUD%20is%20the%20acronym%20for,in%20relational%20and%20NoSQL%20databases>. (дата звернення 18.06.23)