

Київський національний торговельно-економічний університет

Кафедра комп'ютерних наук та інформаційних систем

ВИПУСКНА КВАЛІФІКАЦІЙНА РОБОТА

на тему:

«Розробка програмної системи управління запасами»

Студентки 4 курсу, 8 групи

Мовчан Єлизавети
Павлівни

спеціальності

122 «Комп'ютерні науки»

підпис студента

Науковий керівник
Кандидат технічних наук,
доцент,

Томашевська
Тетяна
Володимирівна

підпис керівника

Гарант освітньої програми
кандидат технічних наук, доцент

Демідов Павло
Георгійович

підпис керівника

Київ 2020

Київський національний торговельно-економічний університет

Факультет інформаційних технологій
Кафедра комп'ютерних наук та інформаційних систем
Спеціальність 122 «Комп'ютерні науки»

Затверджую

Зав. кафедри _____ Пурський О. І.

« » грудня 2021р.



Завдання
на випускню кваліфікаційну роботу (проект) студенту

Мовчан Єлизаветі Павлівні

(прізвище, ім'я, по батькові)

1. Тема випускної кваліфікаційної роботи (проекту)

«Розробка програмної системи управління запасами»

Затверджена наказом ректора від «29» листопада 2021 р. № 3929

2. Строк здачі студентом закінченої роботи 10 червня 2022 року

3. Цільова установка та вихідні дані до роботи

Мета роботи: розробка програмної системи управління запасами підприємства

Об'єкт дослідження: процес проектування інформаційної системи управління запасами.

Предмет дослідження: засоби створення інформаційної системи управління запасами

4. Перелік графічного матеріалу _____

5. Консультанти по роботі із зазначенням розділів, за якими здійснюється консультування:

Розділ	Консультант (прізвище, ініціали)	Підпис, дата	
		Завдання видав	Завдання прийняв
1	Томашевська Т. В.	24.12.2021 р.	24.12.2021 р.
2	Томашевська Т. В.	24.12.2021 р.	24.12.2021 р.
3	Томашевська Т. В.	24.12.2021 р.	24.12.2021 р.

6. Зміст випускної кваліфікаційної роботи (проекту) (перелік питань за кожним розділом)

ВСТУП

РОЗДІЛ 1. Аналітичне дослідження інформаційних систем управління запасами

- 1.1. Задачі та принципи обліку запасів на підприємстві
- 1.2. Огляд існуючих інформаційних систем управління запасами
- 1.3. Бізнес-вимоги до інформаційної системи управління запасами

Висновки до розділу

РОЗДІЛ 2. ПРОЕКТУВАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ УПРАВЛІННЯ ЗАПАСАМИ

- 2.1. Проектування системи управління запасами
- 2.2. Розробка архітектури інформаційної системи управління запасами
- 2.3. Обґрунтування вибору програмних та технічних засобів розробки інформаційної системи управління запасами

Висновки до розділу

РОЗДІЛ 3. РОЗРОБКА ІНФОРМАЦІЙНОЇ СИСТЕМИ УПРАВЛІННЯ ЗАПАСАМИ

- 3.1. Розробка алгоритму роботи системи
- 3.2. Проектування інтерфейсу системи
- 3.3. Створення та тестування системи управління запасами

Висновки до розділу

ВИСНОВКИ

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

7. Календарний план виконання роботи

№ пор.	Назва етапів випускної кваліфікаційної роботи	Строк виконання етапів роботи	
		За планом	фактично
1	2	3	4
1	Вибір теми випускної кваліфікаційної роботи	01.10.2021	01.10.2021
2	Розробка та затвердження завдання на випускну кваліфікаційну роботу	24.12.2021	24.12.2021
3	Вступ	25.02.2021	
4	Розділ 1. Загальна характеристика підходів до аналізу WEB-ресурсів	06.04.2021	
5	Розділ 2. Аналіз сучасних методів та технологій розробки аналізаторів WEB-ресурсів	12.05.2021	
6	Розділ 3. Розробка аналізатора WEB-ресурсів	16.05.2021	
7	Висновки	20.05.2021	
8	Здача випускної кваліфікаційної роботи на кафедрі науковому керівнику	25.02.2021	
9	Попередній захист випускної кваліфікаційної роботи	26.05.2021	
10	Виправлення зауважень, зовнішнє рецензування випускної кваліфікаційної роботи	27.05.2021	
12	Представлення готової зшитої випускної кваліфікаційної роботи на кафедрі	10.06.2021	
13	Публічний захист випускної кваліфікаційної роботи	За розкладом роботи ЕК	

8. Дата видачі завдання «24» грудня 2021 р.

9. Керівник випускної кваліфікаційної роботи (проекту)

Томашевська Т. В.*(прізвище, ініціали, підпис)*

10. Гарант освітньої програми

Демідов П.Г.*(прізвище, ініціали, підпис)*

11. Завдання прийняв до виконання студент-дипломник

Мовчан Є. П.*(прізвище, ініціали, підпис)*

Вступ

Управління запасами, такий же важливий процес роботи підприємства, як і саме виробництво, бухгалтерія, управління кадрами та інше. У епоху інформатизації, абсолютно всі процеси модернізуються та переносяться в інформаційний простір, процес управління запасами не виключення.

Розробка системи управління запасами – це комплексне завдання, яке потребує детального аналізу теми та вже наявних систем. Звичайно, вони налічують достатню кількість, щоб між ними обирати слушну, але крім переваг, кожна з таких систем має і недоліки. Відсутність локалізації потрібною мовою, складний інтерфейс, безліч не потрібних функцій, високу ціну за використання. Оскільки поле систем управління запасами не таке широке, воно може забезпечити конкуренцію, тому нова система буде мати непогані шанси виділитись на цьому полі, адже завжди можна додати, придумати чи реалізувати щось нове, те, чого нема в інших, це своєю чергою призведе до розвитку світу інформаційних систем управління запасами.

Мета – розробка програмної системи управління запасами підприємства.

Завдання – дослідити структуру чинних систем управління запасами, за для досягнення мети.

Об'єкт дослідження – процес проектування інформаційної системи управління запасами.

Предмет дослідження – засоби створення інформаційної системи управління запасами.

РОЗДІЛ 1. Аналітичне дослідження інформаційних систем управління запасами

1.1 Задачі та принципи обліку на підприємстві

Для забезпечення безперебійного процесу виробництва, підприємство повинно мати у наявності оборотні засоби у матеріальній, нематеріальній і грошовій формі.

Наприклад, для виготовлення огорожі з решітки, необхідно мати в наявності певну кількість прутка і кутника у запасі. Такі запаси, в процесі виробництва переходять у форму готової продукції. Під час виробництва вони приймають матеріальну форму двічі. Так, придбання предметів праці є результатом процесу матеріального забезпечення виробництва. Перехід грошей в товар. Цей товар у результаті виробництва стає новим товаром.

Запаси являють собою предмети праці та складають основну частину оборотних засобів. Вони містять у собі значну кількість товарно-матеріальних цінностей, які використовують у виробництві, готову продукцію та товари, незавершене виробництво. Ці цінності являють собою об'єкти обліку запасів.

Основні завдання обліку матеріальних запасів:

1. Вчасне і коректне оформлення документів надходження і використання матеріальних запасів;
2. Систематичний контроль збереження матеріальних запасів, завдяки обліку за матеріально-відповідальними особами та місцями збереження;
3. Забезпечення постійного контролю використання згідно з нормативами та за призначенням;

4. Контроль наявності запасів згідно з нормативами;

5. Періодичне звіряння даних обліку з фактичною наявністю матеріальних запасів у місцях їх збереження та за матеріально-відповідальними особами.

Підприємство, для забезпечення виконання цих завдань, має розробити внутрішні інструкції з обліку матеріальних запасів, правильно організувати раціональний документообіг та складське господарство, забезпечити підбір кадрів та підготовку матеріально-відповідальних осіб та кадрів пов'язаних з обліком, раціонально організувати синтетичний та аналітичний облік.

Огляд існуючих інформаційних систем управління запасами

Проведемо огляд та порівняємо найбільш популярні з існуючих систем управління запасами.

1. Quickbooks — це пакет бухгалтерського програмного забезпечення, розроблений компанією Intuit. Вперше представлені в 1983 році продукти QuickBooks були орієнтовані в основному на малий та середній бізнес і пропонували локальні бухгалтерські програми, а також хмарні версії, які приймали ділові платежі, керували та оплачували рахунки, а також функції нарахування заробітної плати. Також добре підходить для управління запасами та товарами.

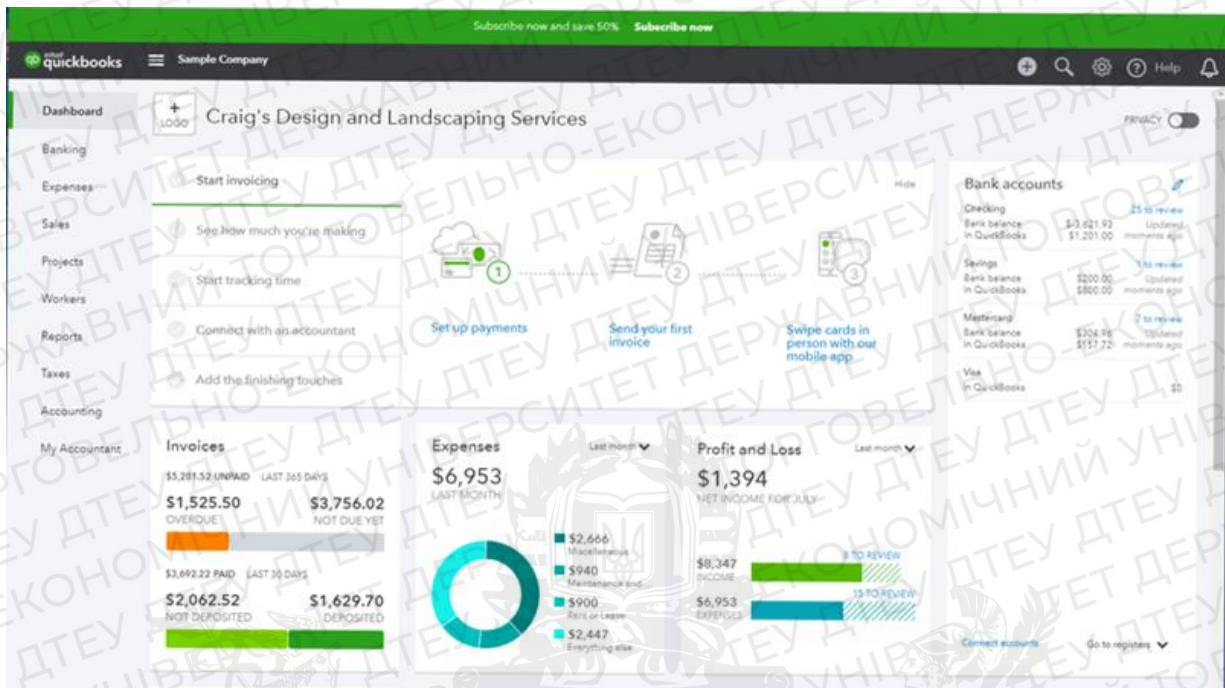


Рис. 1.1(Інтерфейс програми)

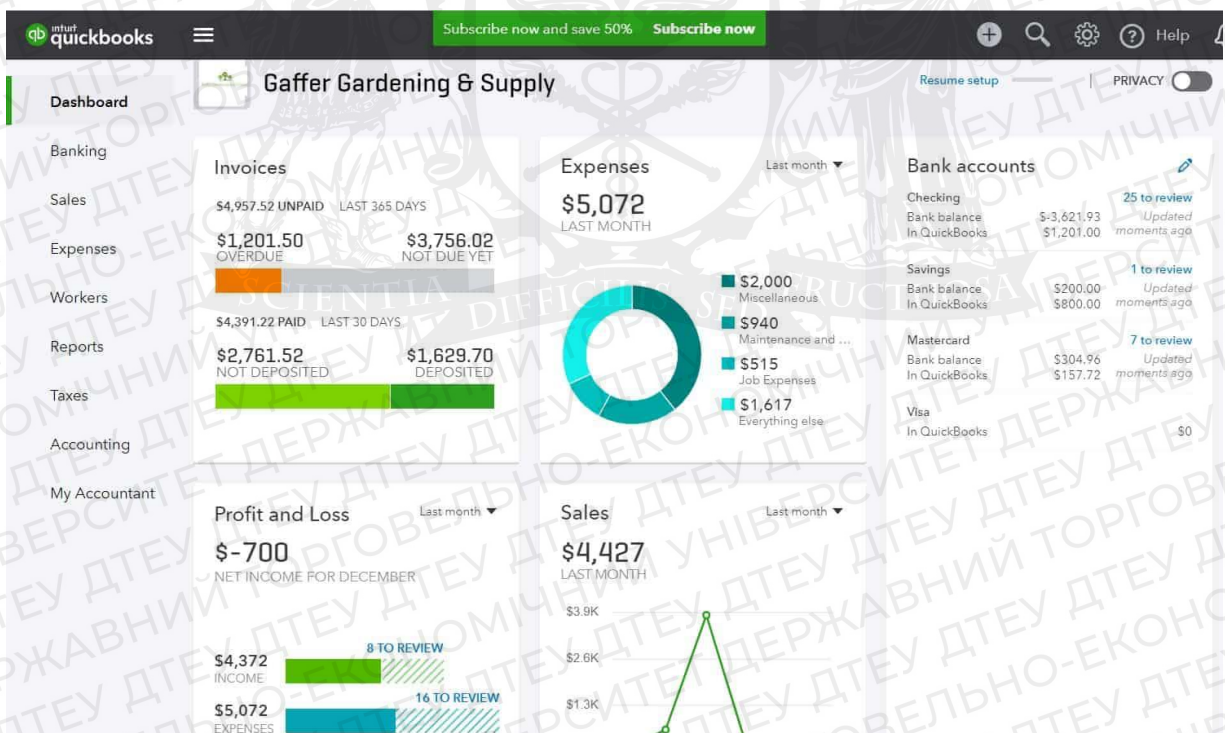


Рис. 1.2(Інтерфейс програми)

Програма доступна після придбання пакету. На сайті доступно 3 рівні підписки з щомісячною оплатою та оплатою за рік. Компанія пропонує

безкоштовну пробну версію кожного пакету на 30 днів, а також 70% знижку на помісячну оплату.

Інформаційна система має веб-версію з приємним та простим інтерфейсом, але підтримує тільки англійську мову, що ускладнює освоєння системи.

Quickbooks добре підходить для користувачів, які добре володіють англійською мовою, інтерфейс приємний та зрозумілий, інформаційна система має весь необхідний функціонал для ведення бізнесу.

2. Zoho CRM — це система управління відносинами з покупцями. Відтак, тут є базові функції CRM(англ. customer relationship management), а саме поставки, інвентаризація, і певні можливості виставлення рахунків, що більше лежить в області ERP (англ. Enterprise Resource Planning). Zoho CRM є вільною, якщо ви вкладаєтеся в обмеження трьох користувачів, і коштує \$12 на місяць за кожного додаткового користувача. Також існує Enterprise Version (версія масштабу підприємства) Zoho CRM з місячною ціною \$25 за кожного користувача.

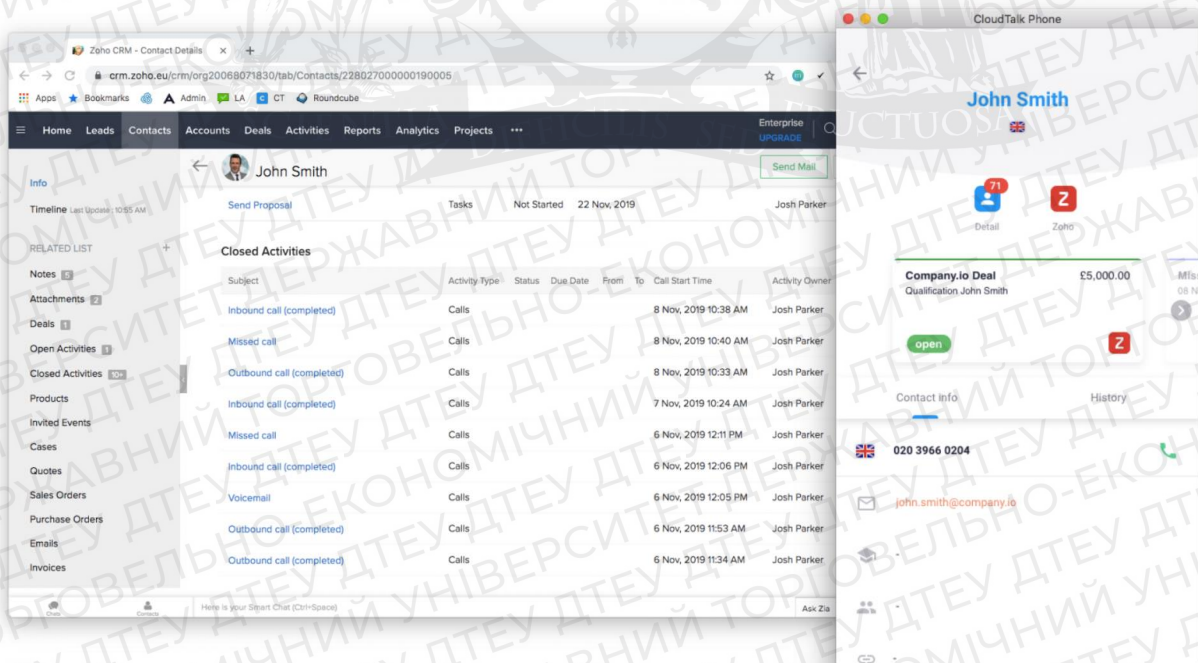


Рис. 1.3(Інтерфейс програми)

The screenshot displays a CRM dashboard for a user named Sol Davidson. The interface includes a top navigation bar with menu items like SalesInbox, Home, Feeds, Leads, Accounts, Contacts, Potentials, Activities, and Reports. The main content area is divided into several sections:

- Welcome Sol Davidson**: A header section.
- Open Tasks**: A table listing tasks with columns for Subject, Due Date, Status, Priority, Related To, Activity Type, and Activity Owner.

SUBJECT	DUE DATE	STATUS	PRIORITY	RELATED TO	ACTIVITY TYPE	ACTIVITY OWNER
Email Nicole - touching base again	Oct 18, 2017	Not Started	Normal	Linchpin Apps	Tasks	Sol Davidson
Call up Jose Gonzales to understand his requirements	Oct 29, 2017	Not Started	Low	Jose Gonzales	Tasks	Sol Davidson
Meet Natalie for a discussion on the proposal	Oct 11, 2017	Not Started	High	Faulkner Enterprises	Tasks	Sol Davidson
- Pipeline by Stage**: A funnel chart showing a distribution of leads across different stages, represented by colored segments (pink, purple, blue, cyan, green).
- Today's Leads**: A table listing leads with columns for Lead Name, Company, Email, Phone, and Lead Source.

LEAD NAME	COMPANY	EMAIL	PHONE	LEAD SOURCE
Jose Gonzales	Ideate Inc	gonzales.ideate@gmail.com		Advertiser
Raymond Richards	Bank of America	rayrichards@mailinator.com		
Joseph Marler	Marler Tech	josephmarler@zoho.com		Partner
- SalesSignals Feeds**: A vertical list of activity notifications on the right side, including items like 'Ashlin Paul Rating #793', 'Nicole Hawkins Linchpin Apps', 'Kelly Smith Escalated #701', and 'Edward Burnette Cloverprises'.

Рис. 1.4(Інтерфейс програми)

Поділ програми на різні версії допомагає користувачу обрати те програмне забезпечення, яке найбільше підходить його бізнесу.

- Stitch Labs** – ще один інструмент для ведення бізнесу, реалізований як веб сайт, який має декілька інтерфейсів з функціями для різного виду бізнесу. Інтерфейс сайту підтримує тільки англійську мову, тож підходить він користувачам з високим рівнем володіння мовою. Навігаційне меню досить просте та зрозуміле.

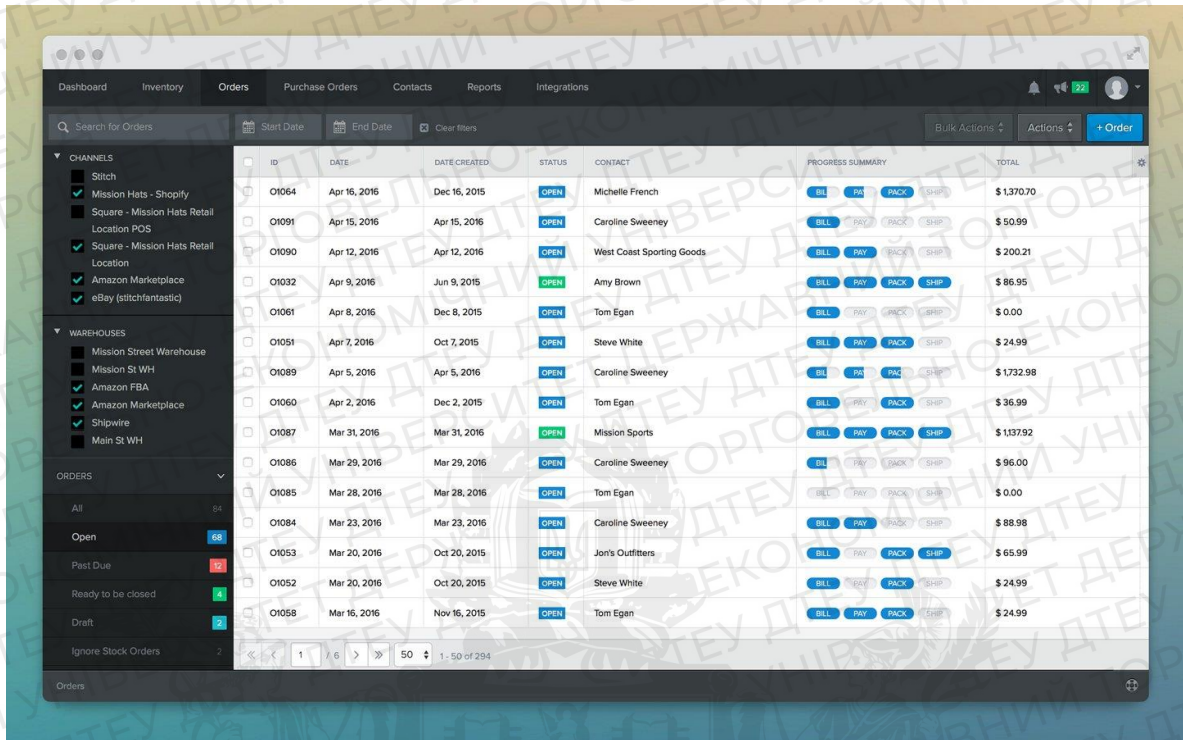


Рис. 1.5(Интерфейс програми)

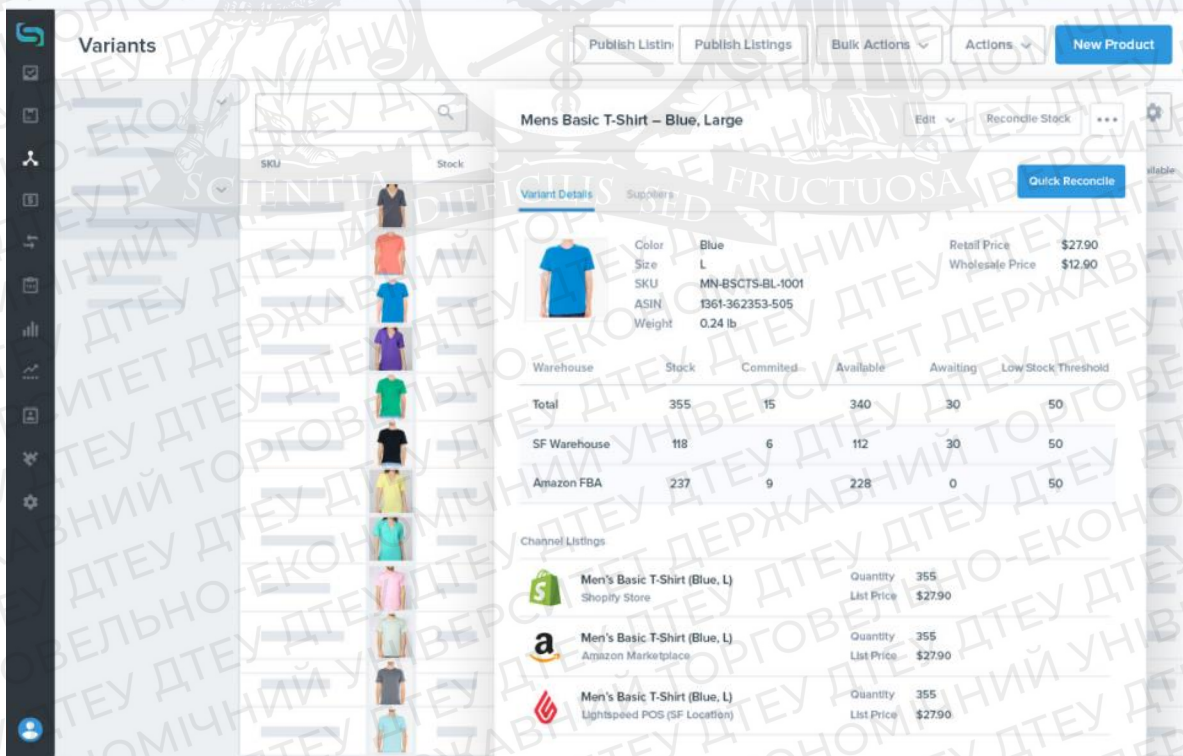


Рис. 1.6(Интерфейс програми)

Плата за користування системою становить від 13500 грн/міс до індивідуально визначеної ціни.

- 4. 1С:** Підприємство – програмне забезпечення, призначене для ведення бухгалтерського, податкового та управлінського обліку на підприємстві а також для формування стандартних бухгалтерських, статистичних та податкових звітностей. З 2017 року заборонений на державних підприємствах України для використання.

«1С: Бухгалтерія для України 2.0» була першою версією програми. Ця версія була розроблена для операційної системи DOS.

Наступною версією, яка з'явилась у 1995 році, була версія «6.0», яка вже мала графічний інтерфейс та працювала на ОС Windows 95.

Першою версією, яка закріпилась на українському ринку стала версія «7.7» «1С: Підприємство 7.7 Бухгалтерський облік для України», пізніше її змінила версія «7.5», проіснувала вона не довго. Українська локалізація була здійснена компанією «Абі Україна». Вона була випущена у 1999 році і підтримувалась до 2016 року. Далі система тільки поновлювалась згідно змін законодавства, а останні поновлення цієї версії випустили у 2020 році.

У 2002 році була випущена версія «8.0», яку з часом замінила нова версія «8.1». Цю версію випустили у 2003 році і підтримувалась вона розробниками до 2011 року.

Починаючи з 2008 року українські конфігурації почали носити назви незалежні від номерів платформ. Так з'явилась «1С: Бухгалтерії для України — редакція 1.0», потім «1С: Бухгалтерії для України — редакція 1.1» і нарешті версія «1С: Бухгалтерії для України — редакція 1.2», яку випустили у 2011 році і підтримували до 2018 року. Українська локалізація була реалізована компаніями «1С» і «Абі Україна».

У 2016 році було випущено «1С: Бухгалтерії для України — редакція 2.0». Вона написана на керованих формах і використовувала інтерфейс «Таксі».

Для підтримки користувачів передбачено перелік сервісів підтримки, який надається за договором супроводу «1С: ІТС Україна».

Можливості «1С: Бухгалтерія для України»:

- Ведення обліку декількох організацій в єдиній інформаційній базі
- Облік «від документу», застосування типових операцій
- Партійний облік
- Складський облік
- Облік торговельних операцій
- Облік комісійної торгівлі
- Облік операцій з тарою
- Облік банківських і касових операцій
- Облік розрахунків з контрагентами
- Облік основних засобів, нематеріальних і малоцінних активів
- Облік основного й допоміжного виробництва
- Облік непрямих витрат
- Облік напівфабрикатів
- Облік ПДВ
- Облік заробітної плати, кадровий облік
- Підтримка різних схем оподаткування
- Податковий облік з податку на прибуток
- Спрощена система оподаткування
- Завершальні операції періоду
- Стандартні бухгалтерські звіти

Програма коштує від 2240 грн до 140000 грн, залежно від версії продукту та його призначення.

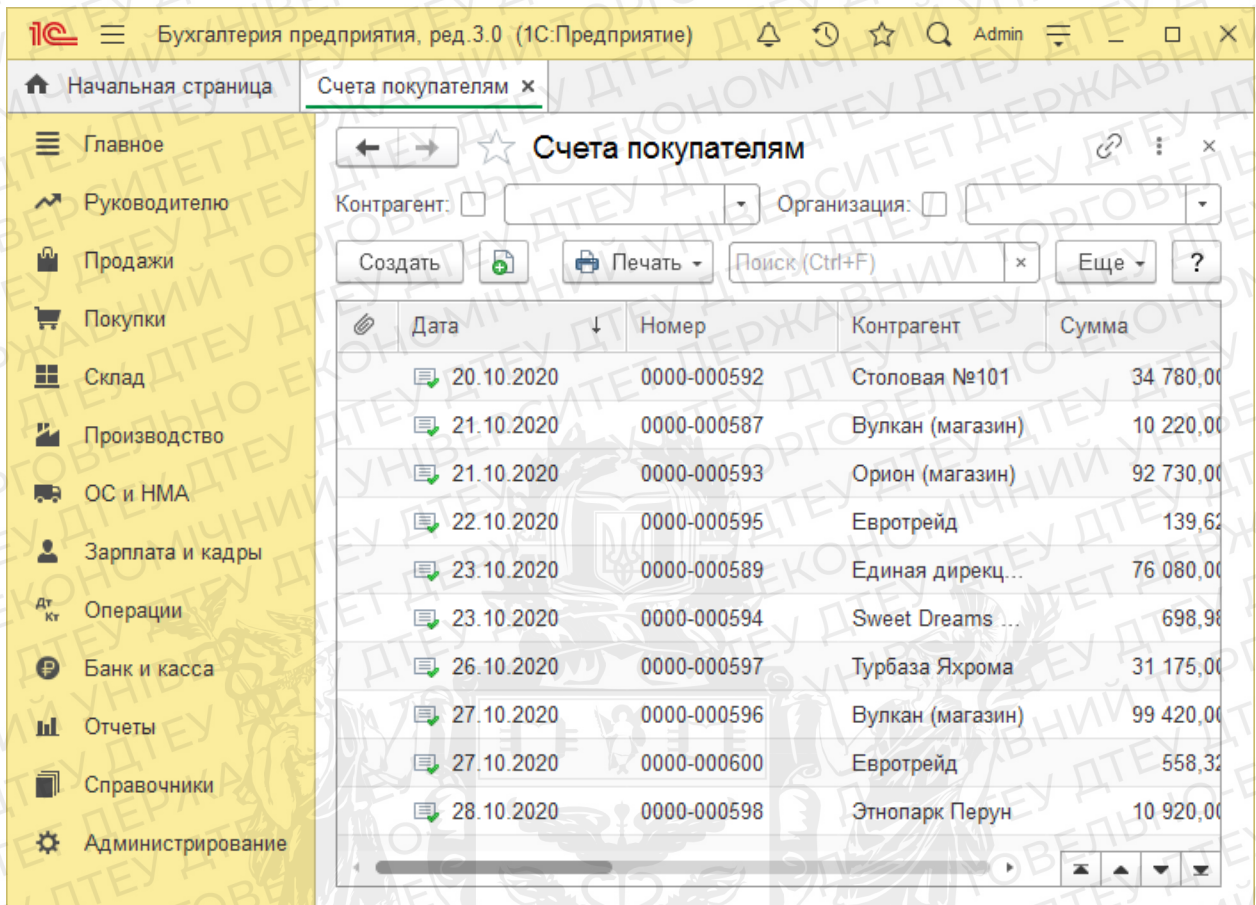


Рис. 1.7(Интерфейс программы)

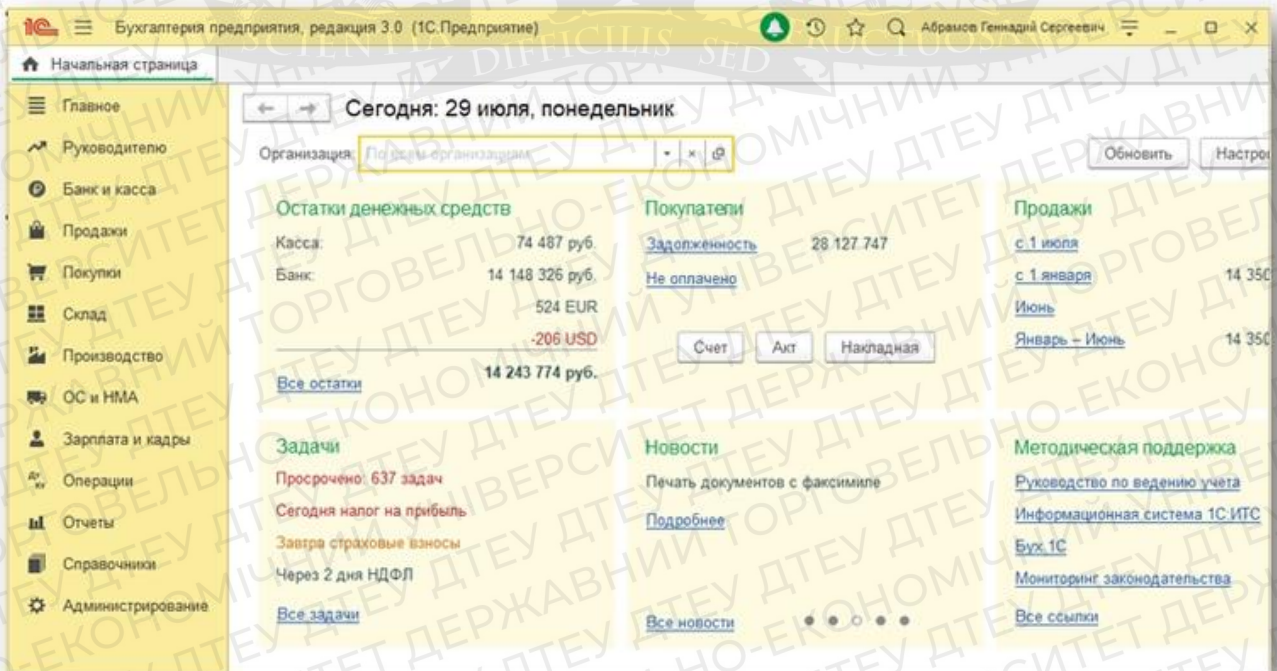


Рис. 1.8(Интерфейс программы)

Враховуючи можливості даного програмного забезпечення, воно підходить більшості підприємств в Україні, адже підтримує українську мову інтерфейсу, та необхідний функціонал для керування персоналом, товаром, запасами та підприємством в цілому.

1.3. Бізнес-вимоги до інформаційної системи управління запасами

Запаси підприємства-це оборотні активи, які призначені для реалізації або споживання протягом операційного циклу або протягом 12 місяців з дати балансу. Запаси бувають виробничі та споживчі, для покращення роботи підприємства проводиться автоматизація процесів виробництва продукції, процес управління персоналом та бухгалтерський облік. Використання сучасного програмного забезпечення для управління підприємством значно підвищить якість виробництва. Необхідність використання інформаційної системи управління запасами на підприємстві полягає у взаємозв'язку процесів виробництва, інформатизація управління запасами пришвидшить ведення обліку та процес планування виробництва на наступний квартал. Це у свою чергу поліпшить та прискорить процес виробництва та управління на підприємстві. додаток призначений для забезпечення автоматизації процесів складського обліку товарів, а також контролю актуальних залишків товарів в локаціях.

Програма повинна мати такі функціональні можливості:

- здійснювати прийомку товару (зчитування штрихкодів, завантаження накладних, відправлення накладних в облікову систему підприємства);
- призначення міток товарів на складі;
- переміщення товарів між складами (відправка, прийом товарів за документами);
- інвентаризація товарів;

- підбір/пошук товару.

Мобільний додаток повинен обмінюватися інформацією з розгорнутим веб-сервером, доступ до якого встановлюється для кожного мережного пристрою індивідуально.

Вимоги до надійності.

Додаток повинен передбачати базовий захист від несанкціонованого доступу. Система повинна забезпечувати валідність даних, що вводяться користувачем.

Вимоги до usability.

При розробці додатку для підвищення ефективності роботи з програмою потрібно дотримуватись принципу "Чим простіше, тим краще".

Швидкість роботи програми.

Необхідно забезпечити стабільно високу швидкість обробки запитів.

Зручність навігації.

Одним з критеріїв розробки додатку повинен стати принцип простоти навігації. Користувач повинен інтуїтивно і безперешкодно переміщатися між вікнами програми та отримувати необхідну інформацію або послугу.

Висновки до розділу

Отже, система управління запасами – це інструмент, необхідний для покращення роботи підприємства, програма повинна бути простою у використанні, мати зрозумілий інтерфейс та функціонал, який забезпечить стабільну і продуктивну роботу системи управління запасами.

РОЗДІЛ 2. ПРОЕКТУВАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ УПРАВЛІННЯ ЗАПАСАМИ

2.1. Проектування системи управління запасами

Мета проекту: розробка сучасної системи управління запасами підприємства, яка матиме простий та зрозумілий інтерфейс та функціонал, необхідний для забезпечення стабільної роботи у системі.

Ціль: запровадити використання нової системи управління запасами на підприємствах України, для покращення їх роботи.

Проект являє собою систему управління запасами, яка буде складатись з програми на комп'ютер та бази даних для зберігання інформації. Кожне підприємство може створювати свою базу даних для зберігання їх та використання за допомогою ПК, яким надано доступ до інформації що зберігається. Програма матиме простий інтерфейс, для легкого освоєння функціоналу. Система дасть змогу обмінюватись документами, зв'язаними із запасами на складі, дозволить легко вести облік запасів та здійснювати їх реалізацію у разі необхідності.

Переваги:

1. Простий інтерфейс;
2. Вигідна ціна;
3. Програма доступна на ПК.

Недоліки:

1. Одна мова інтерфейсу (українська);

Планування роботи над системою

- Потрібно розробити блок схему роботи системи, викласти чітку послідовність можливих взаємодій програми і користувача.
- Далі необхідно побудувати прототип програми, розмістити елементи меню та інший інтерфейс системи у середовищі програмування
- Після виконаної роботи – розробити дизайн програми, обрати кольори,

шрифти, вид кнопок і тому подібне.

- До готової програми приєднуємо базу даних, у яку буде вноситися та зберігатися інформація про запаси підприємства.

За допомогою програмного забезпечення для моделювання бізнес процесів ERWin створимо модель системи управління запасами.

Це комп'ютерна програма призначена для проектування баз даних. Моделі даних візуалізують структуру даних, забезпечуючи ефективний процес організації, управління та адміністрування таких аспектів діяльності підприємства, як рівень складності даних, технологій баз даних та середовища розгортання.

Користувачі можуть використовувати erwin Data Modeler як спосіб створення концептуальної моделі даних та створення логічної моделі даних, яка не залежить від конкретної технології бази даних. Ця схематична модель може бути використана для створення фізичної моделі даних. Потім користувачі можуть направити інженерну мову визначення даних, необхідний створення схеми для низки систем управління базами даних. Програмне забезпечення включає функції для графічної модифікації моделі, у тому числі діалогові вікна для вказівки кількості взаємозв'язків сутностей, обмежень бази даних, індексів і унікальності даних. Ервін підтримує три мови моделювання даних: IDEF1X, варіант розробки інформаційних технологій, розроблений Джеймсом Мартіном, та форму позначення розмірного моделювання.

Програмне забезпечення дозволяє користувачам створювати моделі даних шляхом зворотного проектування вже існуючих баз даних, заснованих на декількох різних форматах. Ще одна включена функція - це здатність Ервіна створювати багаторазово використовувані стандарти проектування: "включаючи стандарти іменування, стандарти типів даних, шаблони моделей і т. д.". Програмне забезпечення включає кілька функцій для зміни способу відображення моделі даних, включаючи параметри для декількох кольорів, шрифти, схеми, тематичні області та макети.

Функція повного порівняння erwin дозволяє користувачеві порівнювати дві версії моделі, відображає відмінності та дозволяє об'єднувати та оновлювати дані у будь-якому напрямку. Станом на березень 2016 року до комплекту програмного забезпечення також входить власний конструктор звітів. Оновлення erwin DM 2018 включало Netezza, MySQL 8.x, PostgreSQL 10.4 та Hive; модель підраховує звіти; та підтримка РІІ. Оновлення 2019 року включало DB2 z/OS v12, SQL Server 2017, Teradata v16.20 та PostgreSQL 11.2 на додаток до покращень звітності, таким як властиві та фільтри, що визначаються користувачем.



Спочатку потрібно створити загальну блок схему системи, яка міститиме всі необхідні елементи для глибшої побудови моделі.

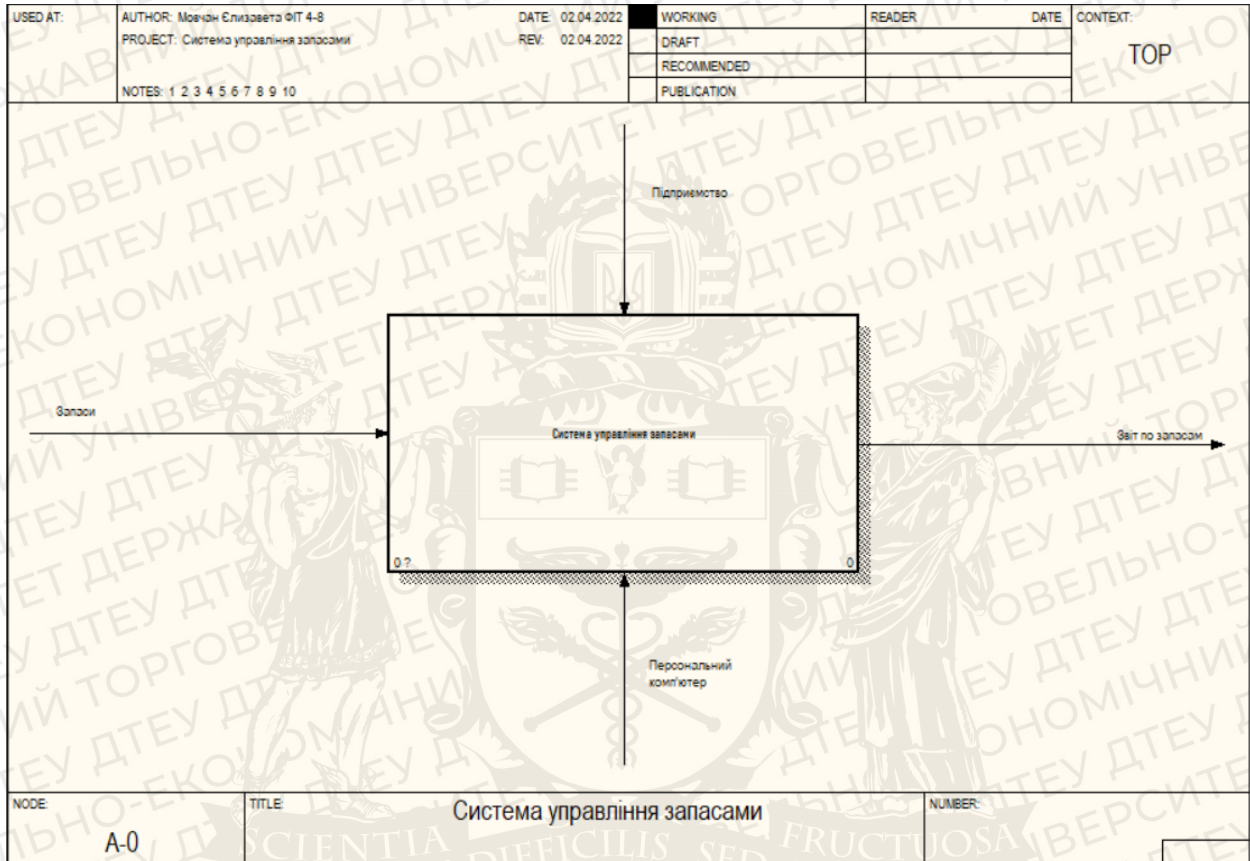


Рис. 2.1(Головна блок схема)

По центру схеми розміщено систему управління запасами, стрілка зліва (запаси) означає введення даних, потрібних для роботи системи. Верхня стрілка (Підприємство) означає управління, нижня стрілка (Персональний комп'ютер) означає метод роботи системи, тобто що потрібно для її використання. Остання стрілка справа (Звіт по запасам) означає вихідні дані, тобто те, що отримаю користувач в результаті роботи в системі.

Наступний крок – створення розширеної моделі інформаційної системи управління запасами.

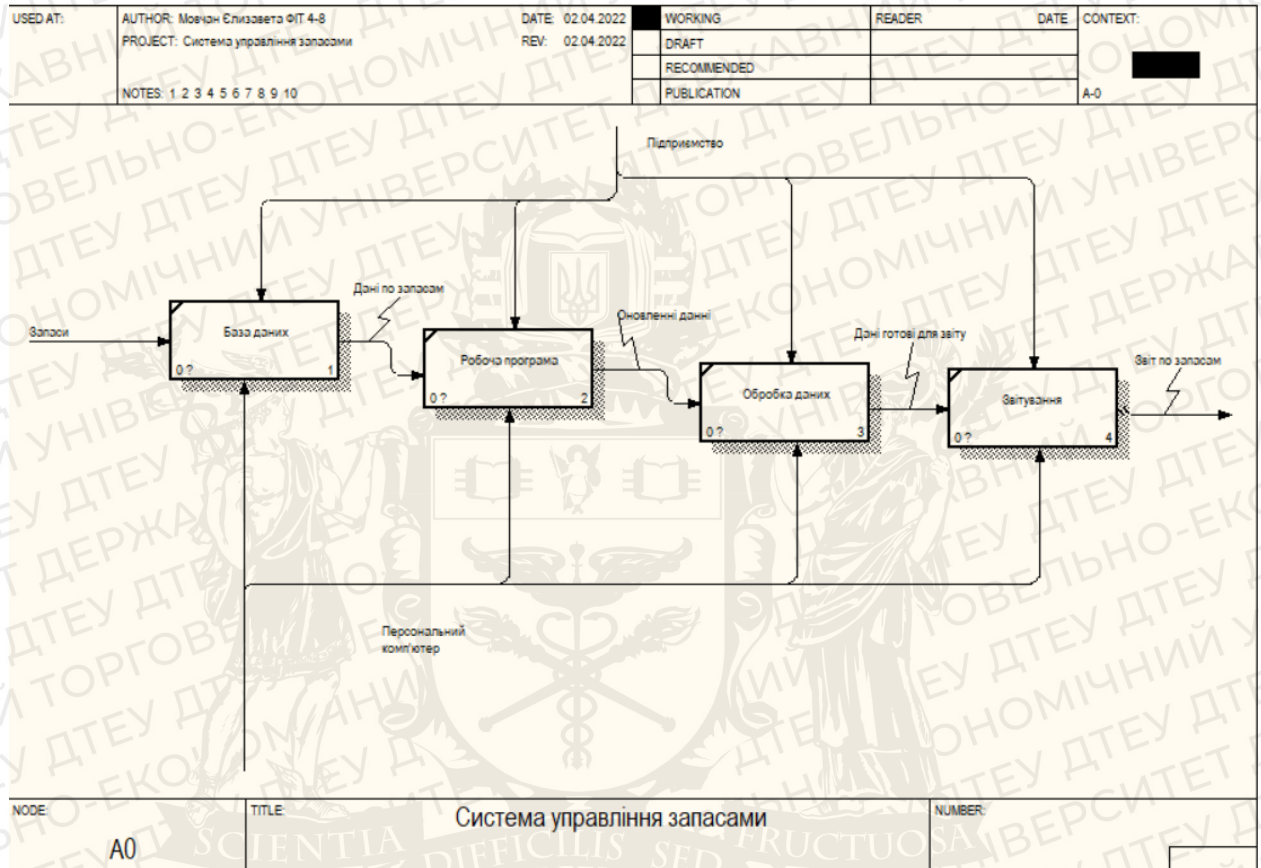


Рис. 2.2(Розширена модель)

На рисунку зображена схема, зі збереженням початкових стрілок. Нові стрілки показують рух даних по системі управління запасами. Модель показує процес роботи системи.

2.2. Розробка архітектури інформаційної системи управління запасами

Інформаційна система управління запасами складається з двох елементів: програми для персонального комп'ютера та бази даних, що об'єднує інформацію по підприємству і забезпечую доступ до неї, через ПК

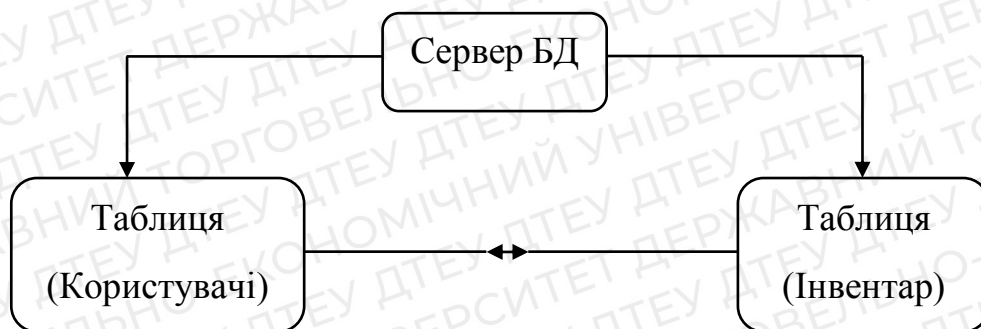
Програма буде розроблена мовою програмування C# (Сі-шарп), для розробки системи використовуємо сучасне середовище програмування "Microsoft Visual Studio", це середовище має простий та зручний інтерфейс та має всі необхідні інструменти для комфортної роботи над програмою. Найголовніше те, що дане середовище має шаблон програми із графічним інтерфейсом, який має набір інструментів для зручної розробки інтерфейсу .

Щодо інтерфейсу програми, він буде виконаний у мінімалістичному стилі, із використанням класичного шрифту та спокійних кольорів, яких повинно бути не більше двох. Для кнопок використаємо напівжирний шрифт та встановимо його розмір більшим на 2 пункти від основного тексту. Програма буде складатись із певної кількості вікон, кожне з яких буде виконувати поставлене завдання, наприклад, одне призначене для реєстрації користувача, інше для редагування даних у базі.

Базу даних розробимо використовуючи SQL – це декларативна мова програмування для взаємодії користувача з базами даних, що застосовується для формування запитів, оновлення і керування реляційними БД, створення схеми бази даних та її модифікації, системи контролю за доступом до бази даних. Сама по собі SQL не є ані системою керування базами даних, ані окремим програмним продуктом. На відміну від дійсних мов програмування (C або Pascal), SQL може формувати інтерактивні запити або, будучи вбудованою в прикладні програми, виступати як інструкції для керування даними. Окрім цього, стандарт SQL містить функції для визначення зміни, перевірки та захисту даних.

SQL — це діалогова мова програмування для здійснення запиту і внесення змін до бази даних, а також керування базами даних. Багато баз даних підтримує SQL з розширеннями до стандартної мови. Ядро SQL формує командна мова, яка дозволяє здійснювати пошук, вставку, оновлення і вилучення даних за допомогою використання системи керування і адміністративних функцій. SQL також включає CLI (Call Level Interface) для доступу і керування базами даних дистанційно.

Схема бази даних



База даних має 2 таблиці, вони зв'язані між собою, доступ до даних таблиці Інвентар надається тільки зареєстрованим користувачам, їх дані містить таблиця Користувачі.

2.3. Обґрунтування вибору системи управління запасами програмних та технічних засобів розробки інформаційної

Microsoft Visual Studio — серія продуктів фірми Майкрософт, які містять інтегроване середовище розробки програмного забезпечення та низку інших інструментальних засобів. Ці продукти дають змогу розробляти як консольні програми, так і програми з графічним інтерфейсом, включно з підтримкою технології Windows Forms, а також вебсайти, вебзастосунки, вебслужби як у рідному, так і в керованому кодах для всіх платформ, що підтримуються Microsoft Windows, Windows Mobile, Windows Phone, Windows CE, .NET Framework, .NET Compact Framework та Microsoft Silverlight.

Переваги середовища:

1. Продуктивність

Масштабування для роботи над проектами будь-якого розміру та складності у 64-розрядному інтегрованому середовищі розробки. Код із новим редактором Razor, який може виконувати рефакторинг між файлами. Діагностика проблем із візуалізацією асинхронних операцій та застосуванням автоматичних аналізаторів.

2. Сучасність

Розробка кросплатформових мобільних та класичних додатків за допомогою .NET MAUI. Створюйте швидкі веб-інтерфейси на C# за допомогою Blazor. Складання, налагодження та тестування додатків .NET та C++ у середовищах Linux. Використовуйте можливості гарячого перезавантаження у програмах .NET та C++. Зміна сторінок ASP.NET у поданні веб-конструктора.

3. Інноваційність

Завершення коду з урахуванням штучного інтелекту. Працюйте разом у режимі реального часу за допомогою загальних сеансів написання коду.

Клонуйте репозиторії, переміщайтеся по робочих елементах та готуйте окремі рядки для фіксації. Автоматичне налаштування робочих процесів CI/CD, які можуть розгортатися в Azure.

4. Менше введення, більше коду

IntelliCode – це потужний набір засобів автоматичного завершення коду, які розпізнають контекст вашого коду: імена змінних, функції та тип створюваного коду. Це дозволяє IntelliCode відразу завершувати цілий рядок, допомагаючи вам упевненіше і точніше створювати код.

5. Докладна аналітика про код

CodeLens допомагає легко знаходити важливі аналітичні відомості, наприклад, які зміни зроблено, до чого вони привели, а також було проведено модульне тестування вашого методу. Основні відомості, такі як посилання, автори, тести та історія фіксацій, завжди доступні та допоможуть вам прийняти виважені робочі рішення.

6. Запобігання проблемам

Інтегрована налагодження - найважливіша складова всіх продуктів Visual Studio. Можна проводити розбір коду, вивчати значення, що зберігаються в змінних, налаштовувати контрольні значення змінних, щоб відстежувати зміну значень, вивчати шлях виконання вашого коду та інші особливості роботи програмного забезпечення.

7. Миттєвий вплив

Аналізуйте обсяг протестованого коду та переглядайте миттєві результати у наборі тестів, оптимізованих для підвищення ефективності. Просунуті функції, що тестують код прямо під час введення, дозволяють дізнатися про наслідки кожної зміни, що вноситься. Завдяки інтеграції WSL можна проводити тестування у Windows та Linux, щоб переконатися, що ваш додаток працює на обох платформах.

8. Вбудоване керування версіями

Visual Studio 2022 має вбудовану підтримку керування версіями Git для клонування, створення та відкриття власних репозиторіїв. Вікно інструментів Git містить все необхідне для фіксації та надсилання змін до коду, управління гілками та вирішення конфліктів злиття. Якщо у вас є обліковий запис GitHub, можна керувати цими репозиторіями безпосередньо у Visual Studio.

Враховуючи вище описані переваги, можемо зробити висновок, що дане середовище програмування добре підходить для виконання поставленої задачі.

Що до системи керування базою даних, буде доцільно використовувати MySQL – вільна система керування реляційними базами даних, яка була розроблена компанією «ТсХ» для підвищення швидкодії обробки великих баз даних. Ця система керування базами даних (СКБД) з відкритим кодом була створена як альтернатива комерційним системам. MySQL з самого початку була дуже схожою на mSQL, проте з часом вона все розширювалася і зараз MySQL — одна з найпоширеніших систем керування базами даних. Вона використовується, в першу чергу, для створення динамічних вебсторінок, оскільки має чудову підтримку з боку різноманітних мов програмування.

MySQL — компактний багатопотоковий сервер баз даних. Характеризується високою швидкістю, стійкістю і простотою використання.

MySQL вважається гарним рішенням для малих і середніх застосувань. Сирцеві коди сервера компілюються на багатьох платформах. Найповніше можливості сервера виявляються в UNIX-системах, де є підтримка багатопоточності, що підвищує продуктивність системи в цілому.

Можливості сервера MySQL:

- простота у встановленні та використанні;
- підтримується необмежена кількість користувачів, що одночасно працюють із БД;

- кількість рядків у таблицях може досягати 50 млн;
- висока швидкість виконання команд;
- наявність простої і ефективної системи безпеки.

Невеликі переваги використання MySQL такі:

- Масштабованість:

MySQL забезпечує відмінну масштабованість для управління та координації роботи з глибоко вбудованими додатками, використовуючи менший слід навіть у масивних складах, що містять величезну кількість даних. Гнучкість на вимогу - чудова особливість MySQL. Підприємства електронної комерції можуть повністю налаштувати унікальні вимоги до сервера баз даних, оскільки MySQL - це рішення з відкритим кодом.

- Безпека:

MySQL - одна з найзахищеніших систем управління базами даних у всьому світі, яка використовується найпопулярнішими веб-додатками, такими як YouTube, Facebook, Twitter, WordPress тощо. Остання версія MySQL забезпечує підтримку та безпеку транзакційної обробки, що забезпечує велику користь бізнесу, особливо у випадку будь-якої електронної комерції, яка вимагає великої кількості грошових переказів.

- Продуктивність:

MySQL призначений для обслуговування найвибагливіших програм та забезпечення одночасно належної швидкості. Він пропонує підвищену продуктивність, надаючи унікальні кеші пам'яті та повнотекстові покажчики. Таким чином, будь то веб-сайт електронної комерції, що виконує мільйони запитів щодня будь-якого типу операційної системи обробки операцій, MySQL пропонує унікальну систему зберігання даних, що дозволяє адміністраторам

налаштувати сервер mysql без будь-якого недоліку, що забезпечує високу продуктивність.

- Економічно ефективним:

Mysql пропонує підприємствам значну економію коштів, забезпечуючи належне управління та надійність, що може заощадити час на усунення несправностей, які в іншому випадку будуть витрачені на проблеми з роботою та вирішення проблем простоїв.

- Немає простоїв:

Mysql пропонує широкий спектр високодоступних рішень, що забезпечує цілодобовий час роботи за допомогою конфігурацій master / slave реплікації та спеціалізованих серверів кластерів.

- Повний контроль робочого процесу:

Mysql пропонує комплексне рішення, що забезпечує функції самокерування, що автоматизують розширення простору та адміністрування баз даних. Він пропонує мінімальний час налаштування і може бути скоріше використаний, який можна встановити в UNIX, Windows, Linux, Macintosh тощо.

- Рішення з відкритим кодом:

Mysql з відкритим кодом вирішує декілька проблем, таких як налагодження, технічне обслуговування, швидкі оновлення та покращений досвід користувача. Захищена обробка забезпечує ефективну транзакцію для обробки великих наборів даних.

- Транзакційна підтримка:

Mysql є найбільш переважним механізмом транзакційних баз даних на ринку, оскільки він забезпечує повну цілісність даних. Він пропонує необмежене

блокування на рівні рядків, ізольовану, послідовну та довговічну - цілком підтримку атомних транзакцій та підтримку транзакцій у кількох версіях.

Висновки до розділу

Отже, система складається з програми для персонального комп'ютера та база даних, яку буде створено на сервері, призначеному для управління базами даних, програма буде підключатись до бази даних та зможе використовувати інформацію записану в таблиці. Додаток буде розроблений у середовищі програмування Microsoft Visual Studio, базу даних під'єднаю також за допомогою цього середовища, адже його функціонал забезпечує легку та продуктивну роботу над проектом, а система аналізу коду допоможе знаходити та виправляти помилки, що в разі прискорює роботу та забезпечує комфортну розробку програми.

РОЗДІЛ 3. РОЗРОБКА ІНФОРМАЦІЙНОЇ СИСТЕМИ УПРАВЛІННЯ ЗАПАСАМИ

3.1. Розробка алгоритму роботи системи

Система має 2 головних елемента – це сервер бази даних та програма для взаємодії з цим сервером. На сервері створюємо базу даних з таблицями, перша таблиця містить інформацію про зареєстрованих користувачів, а саме ім'я користувача (логін) та пароль. Друга таблиця містить інформацію саме про запаси підприємства.

Сервер бази даних локальний, тому для використання даних потрібно запуснути сервер, для забезпечення стабільної роботи системи, знадобиться придбання хостингу, на якому буде запуснено сервер, це дасть змогу програмі у будь який час отримати доступ до даних в БД. Вигляд локального серверу, який містить базу даних приведено на рисунку 3.1.

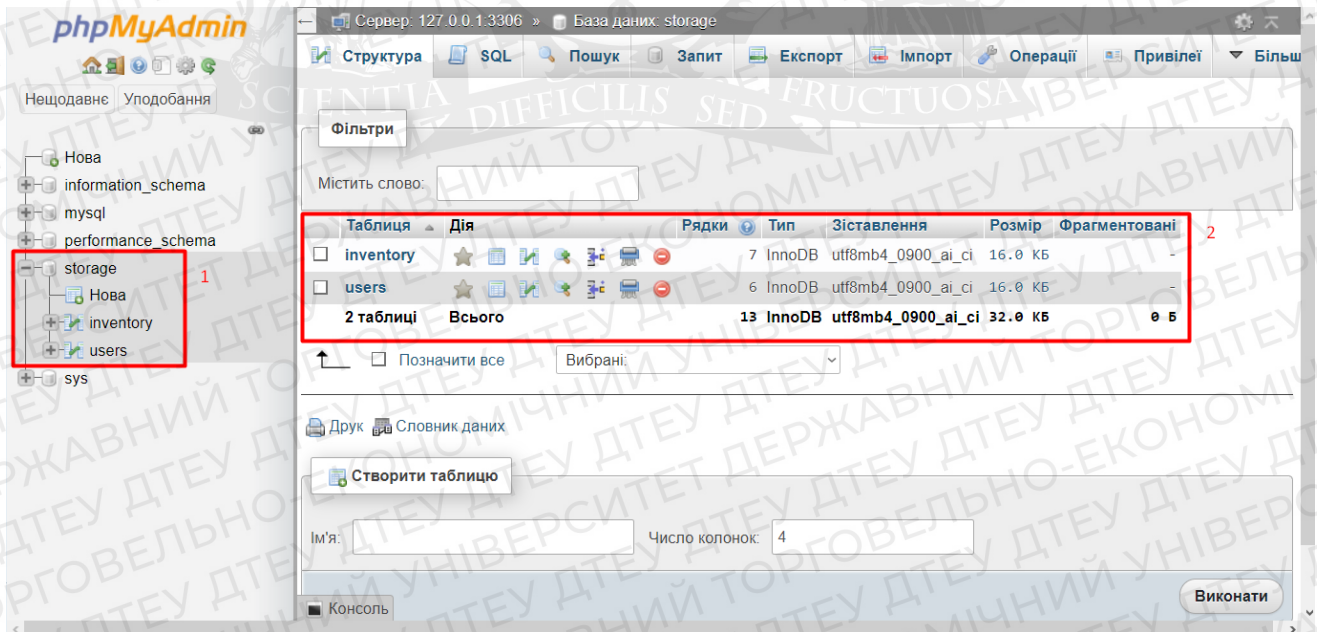


Рис. 3.1(сервер бази даних)

Прямокутником №1 виділено структуру бази даних, її назву та таблиці, які їй належать. Прямокутником №2 виділено інформацію про таблиці.

Розглянемо вміст таблиць на рисунку 3.2.

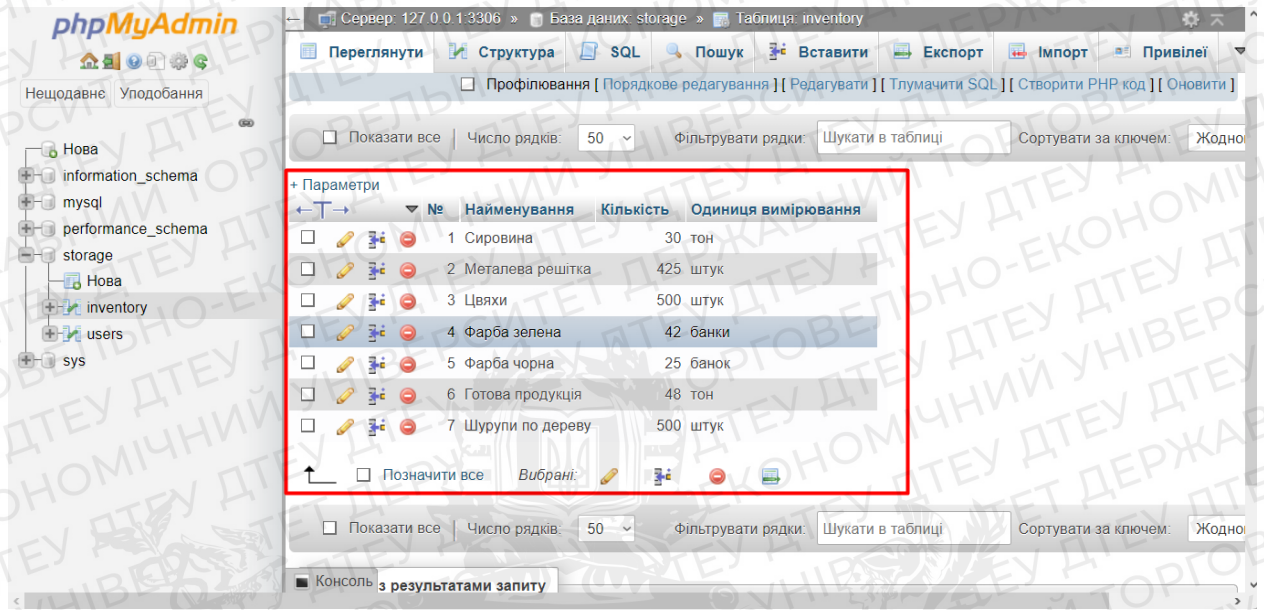
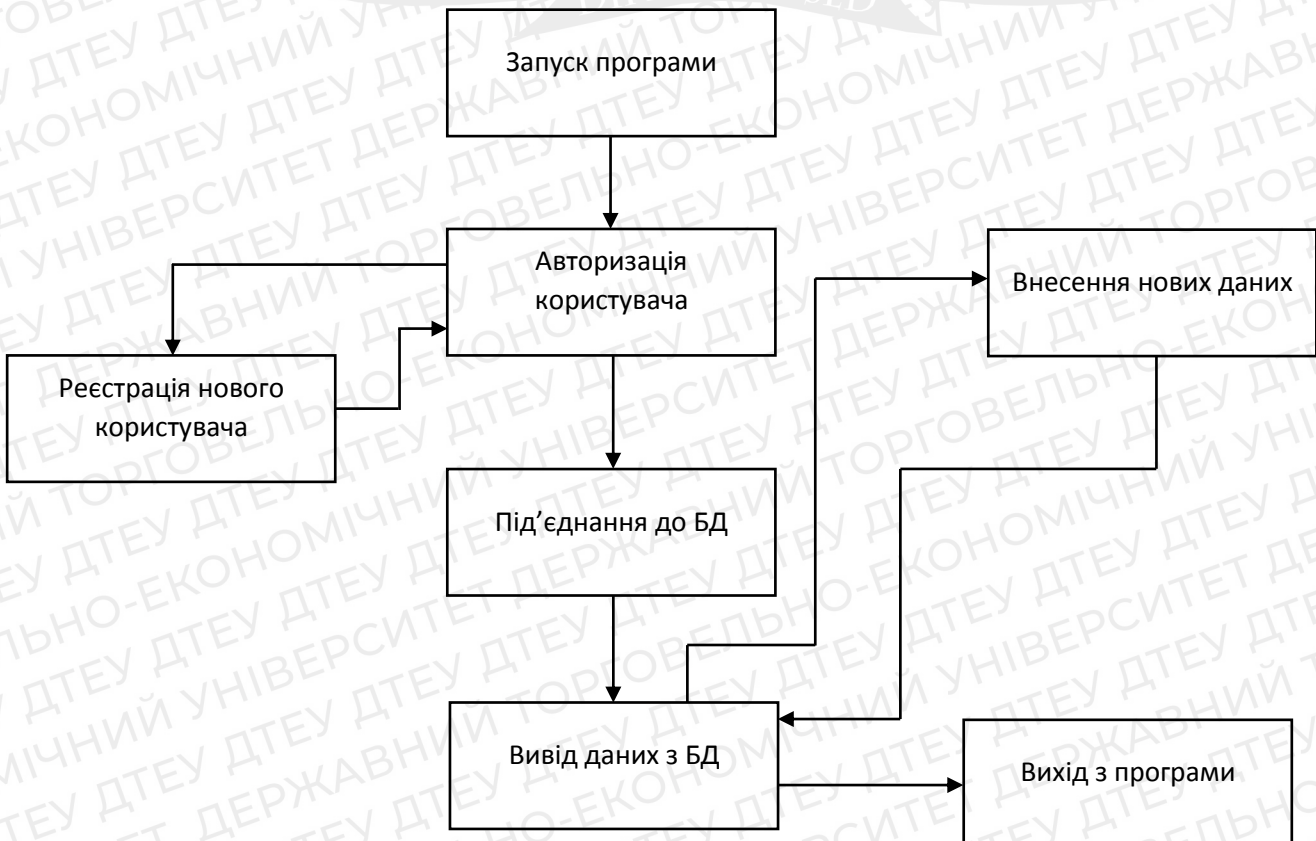


Рис. 3.2(Таблиця запасів)

Прямокутником виділено панель управління даними таблиці.

Алгоритм роботи системи виглядає наступним чином:

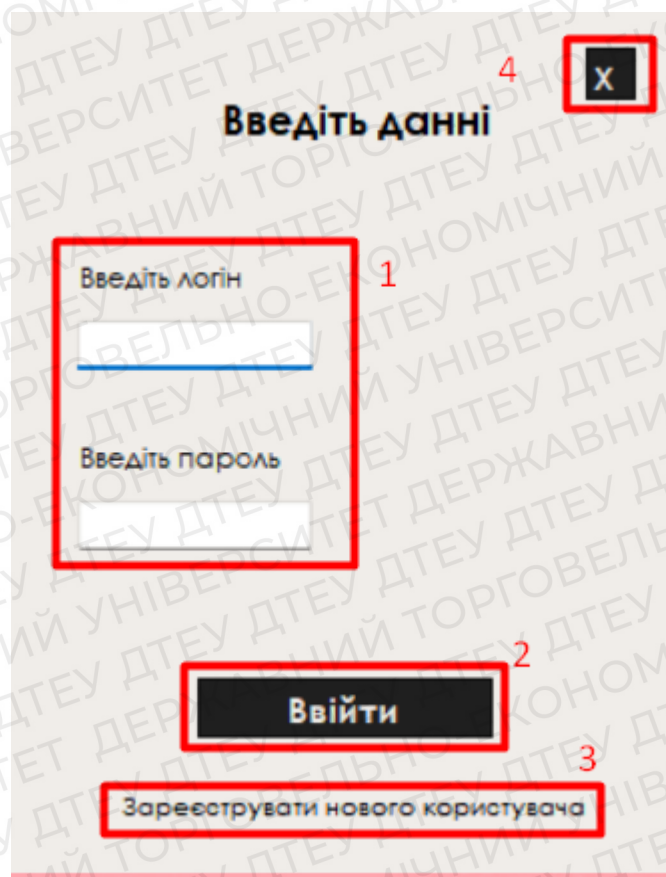


Після запуску програми, користувач має увійти в систему, якщо є необхідність, користувач може зареєструватися в системі, після відкриття вікна реєстрації користувач вводить свої дані, після чого програма їх зберігає у базі даних і переміщує користувача на вікно авторизації. Після успішної авторизації користувач потрапляє на головне вікно програми, де виконує підключення до бази даних та виводить дані таблиці. За необхідності користувач може додати елемент таблиці, для цього він переходить у відповідне вікно, вводить необхідні дані та зберігає їх, після чого програма знову відкриває головне вікно і користувач може перевірити оновлення даних, шляхом повторного їх виведення на екран. Після завершення роботи з системою, користувач виходить з програми і зв'язок між нею та сервером переривається.

3.2. Проектування інтерфейсу системи

Програма складається з чотирьох вікон – вікно авторизації, реєстрації, головного вікна та вікна вводу нових даних.

Інтерфейс системи дуже простий і складається з базових елементів, таких як напис, текстове поле, кнопка та поле виводу даних.



Розглянемо інтерфейс системи на рисунках 3.3, 3.4, 3.5 та 3.6.

На рисунку зображено всі елементи вікна.

№1 – поля для вводу даних.

№2 – кнопка, яка перевіряє дані та відкриває головне вікно.

№3 – напис, який слугує для переходу до вікна реєстрації, при натисканні на нього мишкою.

№4

—

кнопка

виходу

з

програми.

Рис. 3.3(вікно авторизації)

На рисунку зображено всі елементи вікна.

Заповніть поля

Введіть логін 1

Введіть пароль

Повторіть пароль

Зареєструватись 2

Повернутись до авторизації 3

№1 – поля для вводу даних.

№2 – кнопка, яка вносить нові дані в таблицю БД та повертає користувача до вікна авторизації.

№3 – напис, який повертає користувача до вікна авторизації, після натискання на нього мишкою.

Рис. 3.4(вікно реєстрації)

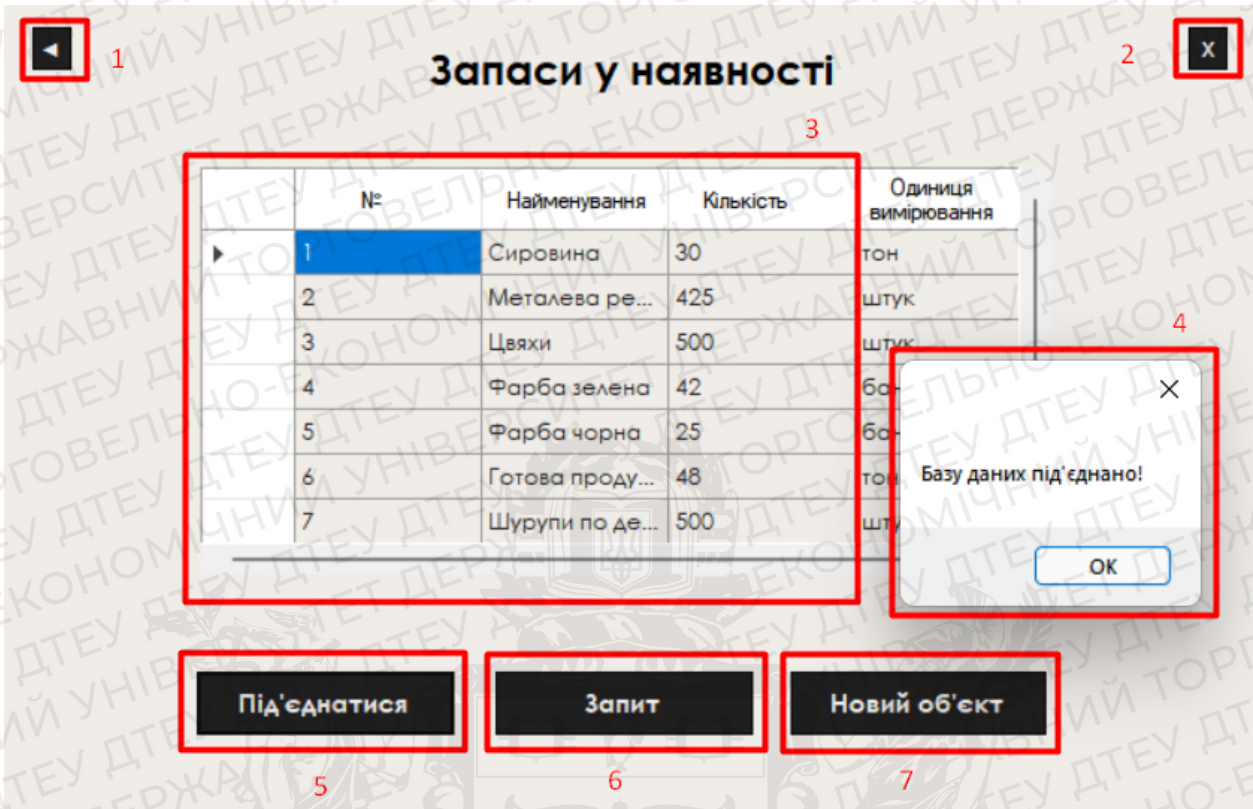


Рис. 3.5 (головне вікно)

На рисунку зображено всі елементи вікна.

№1 – кнопка повернення до вікна авторизації.

№2 – кнопка виходу з програми.

№3 – поле виводу даних із БД.

№4 – повідомлення, яке з'являється після натискання кнопки “Під'єднатися”.

№5 – кнопка, що підключає програму до БД.

№6 – кнопка, яка виводить дані в поле виводу.

№7 – кнопка, що відкриває вікно внесення нових даних до БД.

Введіть данні

Найменування 1

Кількість

Одиниця вимірювання

Зберегти

Завершити

2 3

На рисунку зображено всі елементи вікна.

№1 – поля для вводу даних.

№2 – кнопка, що зберігає дані в базу даних.

№3 – кнопка, що повертає користувача до головного вікна.

Рис. 3.6(вікно введення нових даних)

3.2. Створення та тестування системи управління запасами

Програму написано мовою програмування С#, базу даних створено за допомогою програмного забезпечення, що використовує мову SQL.

Результати роботи системи приведено нижче.

+ Параметри

	id	username	Password
<input type="checkbox"/>	1	movch.lis	1q2w3e4r5
<input type="checkbox"/>	2	admin	admin
<input type="checkbox"/>	4	Dmytro	123

Рис. 3.7(данні таблиці users)

Запустимо програму та зареєструємо нового користувача.

У вікні реєстрації введемо наступні дані: логін – Anna, пароль – 123321.

Заповніть поля

Введіть логін

Anna

Введіть пароль

123321

Зареєструватись

Повернутись до авторизації

Данні оновлено

OK

Рис. 3.8(результат роботи вікна)

Після вводу даних та натискання на кнопку “Зареєструватися”, програма вивела повідомлення про те що дані оновлено.

Перевіряємо таблицю в базі даних.

+ Параметри













	id	username	Password
<input type="checkbox"/>   	1	movch.lis	1q2w3e4r5
<input type="checkbox"/>   	2	admin	admin
<input type="checkbox"/>   	4	Dmytro	123
<input type="checkbox"/>   	11	Anna	123321

Рис. 3.9(оновленні дані таблиці)

Програма вдало внесла зміни в таблицю та додала нового користувача, Виконаємо авторизацію по новому користувачу.

Введіть данні

Введіть логін
Anna

Введіть пароль
123123

Перевірте дані
OK

Ввійти

[Зареєструвати нового користувача](#)

Рис. 3.10(Виведене повідомлення про помилку)

При введенні неправильних даних, програма вивела повідомлення з проханням перевірити дані.

Після введення коректних даних, програма відкриває головне вікно, розглянемо повний функціонал програми.

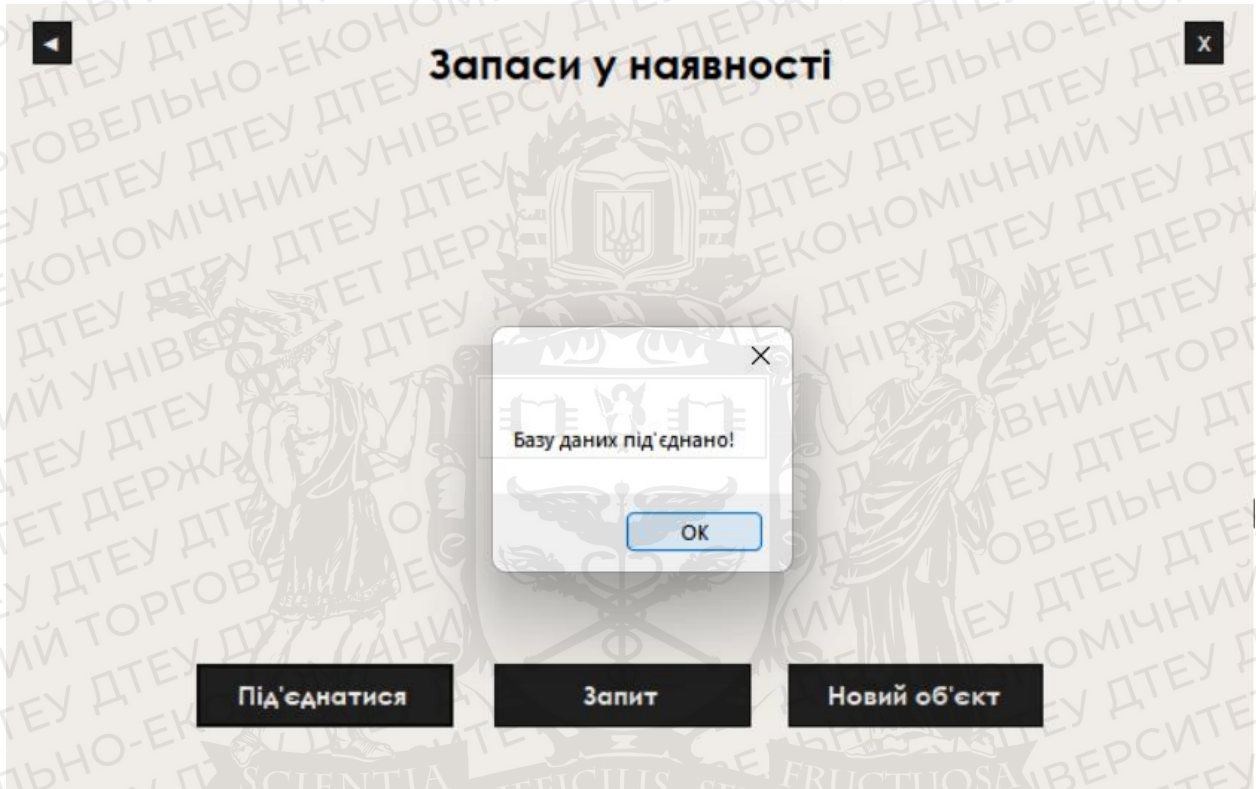


Рис. 3.11(Повідомлення про під'єднання БД)

Запаси у наявності

	№	Найменування	Кількість	Одиниця вимірювання
▶	1	Сировина	30	тон
	2	Металева ре...	425	штук
	3	Цвяхи	500	штук
	4	Фарба зелена	42	банки
	5	Фарба чорна	25	банк
	6	Готова проду...	48	тон
	7	Шурупи по де...	500	штук

Під'єднатися Запит Новий об'єкт

Рис. 3.12(Виведена таблиця)

Введіть данні

Найменування

Кількість

Одиниця вимірювання

Зберегти **Завершити**

✕

Данні оновлено

Рис. 3.13(Повідомлення про оновлення даних)

Запаси у наявності

№	Найменування	Кількість	Одиниця вимірювання
4	Фарба зелена	42	банки
5	Фарба чорна	25	банок
6	Готова проду...	48	тон
7	Шурупи по де...	500	штук
14	Стержень ме...	152	штуки

Під'єднатися Запит Новий об'єкт

Рис. 3.14(Оновлена таблиця)

Після авторизації, програма відкрила головне вікно, де при натисканні кнопки Під'єднатися, програма виконала під'єднання до БД. Після натискання кнопки Запит, програма вивела дані з таблиці на екран.

При натисканні кнопки Новий об'єкт програма відкрила вікно для вводу нових даних. Після заповнення полів та натискання кнопки Зберегти програма внесла зміни до БД та вивела повідомлення про успішне оновлення даних. При натисканні кнопки Завершити, програма повернула користувача до головного вікна. Чи внесла програма нові дані до БД, відправимо запит на виведення даних ще раз та перевіримо дані таблиці.

На рис.№17 бачимо що з'явився новий рядок з даними, які ми внесли через програму.

Висновки до розділу

Після створення бази даних, був розроблений алгоритм роботи системи управління запасами, який являє собою алгоритм взаємодії програми та бази даних, також на даному етапі було розроблено алгоритм роботи програми.

Наступний етап складався з проектування інтерфейсу системи та розробки прототипу програми, на цьому етапі була розроблена структура програми, визначено функціонал, необхідний для роботи системи.

Створення системи – перед останній етап роботи, було розроблено дизайн вікон програми, окрім цього, було розроблено тіло програми, тобто код, який саме і об'єднував програму та базу даних в одну систему.

На останньому етапі розробки системи було проведено її тестування, шляхом перевірки працездатності кожної реалізованої функції системи, як результат, програма вдало впоралася з усіма поставленими задачами.

Висновки

Завданням випускної кваліфікаційної роботи було створення інформаційної системи управління запасами. Процес розробки складався з конкретних етапів:

1. Аналіз існуючих систем;
2. Вибір програмних засобів для розробки;
3. Проектування системи;
4. Розробка інтерфейсу;
5. Створення та тестування системи.

Така послідовність етапів дала змогу продуктивно працювати над проектом і, як наслідок, привела до успішної реалізації системи управління запасами.

На першому етапі було проведено аналіз існуючих систем, це потрібно було для розуміння принципів їх роботи, виявлення недоліків та переваг.

Вибір програмних засобів – це важлива частина роботи, тому було обрано сучасне, зручне та доступне середовище програмування Microsoft Visual Studio, у якості СУБД було обрано сервер MySQL, адже це популярна система, яка забезпечує хороший захист даних та доступна для взаємодії із програмами, розробленими власноруч.

Наступний етап передбачав проектування системи, що означає повний її опис як проекту, була визначена мета, завдання, переваги та недоліки, також було змодельовано процес роботи системи, взаємодії програми та бази даних.

Останнім етапом стала розробка програми, саме її кодова частина, також було створено базу даних, з якою працюватиме програма.

У результаті маємо систему управління запасами, управління якою відбувається через програму для ПК, основними її функціями є перевірка даних користувачів, які зареєстровані у системі, реєстрація нових користувачів, перегляд та поповнення інформації про запаси.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. [В. Б. Неруш, В. В. Курдеча “ІМІТАЦІЙНЕ МОДЕЛЮВАННЯ СИСТЕМ ТА ПРОЦЕСІВ”](#)
2. [Павлиш В. А. Основи інформаційних технологій і систем: Навчальний посібник. / Павлиш В. А., Гліненко Л. К. - Львів: Видавництво Львівської політехніки, 2013. – 500 с.](#)
3. [Анісімов А.В. Інформаційні системи та бази даних: Навчальний посібник для студентів факультету комп’ютерних наук та кібернетики. / Анісімов А.В., Кулябко П.П. – Київ. – 2017. – 110 с](#)
4. [Березін О. В., Дуда С. Т., Міценко Н. Г. Управління потенціалом підприємства: Навчальний посібник. Львів: Магнолія 2006, 2011. 6 с.](#)
5. [Баранець Г. В. Управління матеріальними та фінансовими потоками підприємства на основі логістичного підходу: дис. канд. екон. наук: 08.00.04; НАН України, Інститут економіки промисловості. Донецьк, 2007. 195 с.](#)
6. <https://visualstudio.microsoft.com>
7. <https://www.mysql.com>
8. <https://docs.microsoft.com/ru-ru/dotnet/csharp/>
9. <https://metanit.com/sharp/tutorial/1.1.php>
10. <https://ospanel.io/download/>
11. <https://www.phpmyadmin.net>