

Вінницький національний технічний університет
Факультет інтелектуальних інформаційних технологій та автоматизації
Кафедра комп'ютерних систем та управління

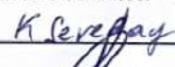
КОМПЛЕКСНА МАГІСТЕРСЬКА КВАЛІФІКАЦІЙНА РОБОТА

на тему:

Розробка автоматизованої системи обліку складських товарів і підтримки торговельної діяльності. Частина 2. Розробка мобільного застосунку.

Виконав: студент 2 курсу, групи 1АКІТР-24м спеціальності 174 – Автоматизація та комп'ютерно-інтегровані технології та роботехніка

(шифр і назва спеціальності)

Катерина СЕРЕДА 
(ПІБ студента)

Керівник: д.т.н., професор кафедри КСУ

Марія ЮХИМЧУК
(науковий ступінь, вчене звання / посада, ПІБ керівника)

« 12 » 12 2025 р.

Опонент: к.т.н., доцент кафедри АІТ

Володимир ГАРМАШ 
(науковий ступінь, вчене звання / посада, ПІБ опонента)

« 12 » 12 2025 р.

Допущено до захисту

Завідувач Кафедри КСУ

д.т.н., проф.

Вячеслав КОВТУН 

(науковий ступінь, вчене звання)

« 13 » 12 2025р.

Вінницький Національний Технічний Університет
Факультет інтелектуальних інформаційних технологій та автоматизації
Кафедра комп'ютерних систем управління
Галузь знань – 17 – Електроніка, автоматизація та електронні комунікації
Спеціальність – 174 - Автоматизація, комп'ютерно-інтегровані технології та
робототехніка
Освітньо-професійна програма – Інтелектуальні комп'ютерні системи

ЗАТВЕРДЖУЮ

Зав. кафедри КСУ ВНТУ,

д.т.н., професор

В'ячеслав КОВТУН

“ 26 ” 09 2025 р.

ЗАВДАННЯ

НА КОМПЛЕКСНУ МАГІСТЕРСЬКУ КВАЛІФІКАЦІЙНУ РОБОТУ
студенту групи ІАКІТ-24м Середі К. Я.

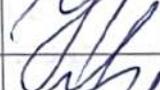
1. Тема магістерської кваліфікаційної роботи: Розробка автоматизованої системи обліку складських товарів і підтримки торговельної діяльності. Частина 2. Розробка мобільного застосунку.
керівник комплексної магістерської кваліфікаційної роботи д.т.н., Юхимчук М.С.
затверджені наказом ВНТУ від «24» вересня 2025 року №313
2. Термін подання студентом роботи «12» грудня 2025 року.
3. Вихідні дані до роботи: Стеценко, В. М. Автоматизація обліку на підприємствах торгівлі / В. М. Стеценко. – Київ: Ліра-К, 2019. – 234 с.
Шевченко, І. В. Розробка мобільного додатку для обліку товарів на складі / І. В. Шевченко, Н. М. Коваль // Вісник ХНУРЕ. – 2022. – № 3. – С. 85–90.
4. Зміст текстової частини: вступ, аналіз предметної області та існуючих систем автоматизації навчального процесу, огляд та вибір інструментів розробки системи, проектування, розробка та тестування розробленої системи, економічна частина, висновки
5. Перелік ілюстративного матеріалу (з точним зазначенням обов'язкових креслень). Архітектура системи, структура бази даних, UM/ L Use-Case діаграми, вигляд скранів розробленого додатку.

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис	Дата
		Завдання видав	Виконання прийняв
1-3	Юхимчук М. С., професор кафедри комп'ютерних систем управління		
4	Ратушняк О.Г., доцент кафедри економіки підприємства і виробничого менеджменту		

2. Дата видачі завдання "25" вересня 2025 року

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва та зміст етапу	Терміни виконання		Примітка
		початок	закінчення	
1	Аналіз предметної області та об'єкта автоматизації	26.09.2025	04.10.2025	
2	Аналіз існуючих програмних продуктів і систем для автоматизації складського обліку та торгівлі	05.10.2025	14.10.2025	
3	Формування вимог та проектування архітектури систем	15.10.2025	24.10.2025	
4	Проектування та створення бази даних	25.10.2025	04.11.2025	
5	Розробка UML-діаграм варіантів використання та діяльності	05.11.2025	13.11.2025	
6	Розробка програмного забезпечення (мобільний застосунок)	14.11.2025	27.11.2025	
7	Тестування роботи системи, налагодження функцій, усунення помилок	28.11.2025	02.12.2025	
8	Оформлення текстової частини та додатків МКР	03.12.2025	06.12.2025	
9	Розрахунок економічної частини	07.12.2025	09.12.2025	
10	Оформлення МКР та збір підписів супроводжувальних документів	10.12.2025	13.12.2025	
11	Захист МКР	15.12.2025	18.12.2025	

Студент



Катерина СЕРЕДА

Керівник роботи

Марія ЮХИМЧУК

АНОТАЦІЯ

УДК 004.4:668.787

Середа К. Я. Розробка автоматизованої системи обліку складських товарів і підтримки торговельної діяльності. Частина 2. Розробка мобільного застосунку. Магістерська кваліфікаційна робота зі спеціальності 174 – Автоматизація, комп'ютерно-інтегровані технології та робототехніка, освітня програма Інтелектуальні комп'ютерні системи. Вінниця: ВНТУ, 2025. 95 с.

На укр. мові. Бібліогр.: 50 назв; рис.: 55; табл. 4.

В ході виконання магістерської кваліфікаційної роботи було розроблено автоматизовану систему обліку складських товарів та підтримки торгівлі. В роботі проведено аналіз об'єкту дослідження, спроектовано та розроблено структуру програмного забезпечення. Розроблено архітектуру автоматизованої системи, UML-діаграми функціонування системи та програмне забезпечення. Розроблений комплекс програм пройшов тестування. Результати тестування підтверджують правильність роботи програми, що дозволяє автоматизувати рутинні завдання та мінімізувати вплив помилок людського фактору на роботу підприємства. Програмні засоби мають зручний та зрозумілий інтерфейс для користувача.

Ключові слова: автоматизована система, складський облік, торговельна діяльність, мобільний застосунок, база даних, тестування.

ANNOTATION

UDC 004.4:668.787

Sereda K. Y. Development of an Automated System for Warehouse Inventory Management and Support of Commercial Activities. Part 2. Development of a Mobile Application. Master's Qualification Thesis in the specialty 174 – Automation, Computer-Integrated Technologies, and Robotics, study program “Intelligent Computer Systems.” Vinnytsia: VNTU, 2025. 95 p.

In Ukrainian. Bibliography: 50 sources; figures: 55; tables: 4.

During the execution of the master's qualification thesis, an automated system for warehouse inventory management and commercial activity support was developed. The work includes an analysis of the research object, as well as the design and development of the software structure. The architecture of the automated system, UML diagrams of system operation, and the software itself were created. The developed software suite was tested. The test results confirm the correctness of the program's operation, enabling the automation of routine tasks and minimizing the impact of human error on the enterprise's activities. The software tools feature a convenient and user-friendly interface.

Keywords: automated system, warehouse accounting, commercial activity, mobile application, database, testing.

ЗМІСТ

ВСТУП	4
1 ЗАГАЛЬНИЙ ОГЛЯД ТА АНАЛІЗ СУЧАСНИХ ТЕХНОЛОГІЙ ВЕДЕННЯ ЕЛЕКТРОННОЇ ФОРМИ ДОКУМЕНТАЦІЇ ТА УПРАВЛІННЯ СКЛАДОМ	6
1.1 Дослідження перспектив переходу компаній на технологію ведення електронної форми документації та управління складом .	6
1.2 Аналіз способів використання мобільних технологій для досягнення конкурентоспроможності компаній у веденні бізнесу .	11
1.3 Дослідження роботи портативних пристроїв, що забезпечують роботу складу з використанням мобільних технологій	12
1.4 Уточнена постановка задачі	16
2 ДОСЛІДЖЕННЯ РОБОТИ СУЧАСНИХ ПРОГРАМ-АНАЛОГІВ ТА РОЗРОБКА СТРУКТУРИ ВЛАСНОГО МОБІЛЬНОГО ДОДАТКУ	17
2.1 Аналіз та дослідження функціональних можливостей мобільних додатків програм-аналогів систем управління, що забезпечують облік складських товарів	17
2.2 Визначення функціональних можливостей автоматизованої системи, що розробляється	33
2.3 Проектування бази даних системи	34
2.4 Розробка UML-діаграм	39
2.4.1 Розробка UML-діаграми варіантів використання	41
2.4.2 Розробка UML-діаграми компонентів	44
3 РОЗРОБКА МОБІЛЬНОГО ДОДАТКУ АВТОМАТИЗОВАНОЇ СИСТЕМИ ОБЛІКУ СКЛАДСЬКИХ ТОВАРІВ ТА ПІДТРИМКИ ТОРГІВЛІ	45
3.1 Розробка мобільного додатку для ОС Android	45

3.2	Перевірка функціонування мобільного додатку автоматизованої системи обліку складських товарів та підтримки торгівлі	54
4	ЕКОНОМІЧНА ЧАСТИНА	70
4.1	Комерційний та технологічний аудит науково-технічної розробки	70
4.2	Прогнозування витрат на виконання науково-дослідної (дослідно-конструкторської) роботи	74
4.3	Розрахунок економічної ефективності науково-технічної розробки за її можливої комерціалізації потенційним інвестором	79
4.4	Розрахунок ефективності вкладених інвестицій та періоду їх окупності.....	82
4.5	Висновки до розділу	84
	ВИСНОВКИ	85
	СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	86
	Додатки	90
	Додаток А (обов'язковий) протокол перевірки навчальної (кваліфікаційної) роботи	91
	Додаток Б (обов'язковий) Технічне завдання	92
	Додаток В (довідковий) Лістинг програми	95
	Додаток Г (обов'язковий) Ілюстративна частина	104

ВСТУП

Актуальність теми. Автоматизація бізнес-процесів є ключовим чинником успішного розвитку підприємств. Із розширенням компанії зростає потреба у контролі великої кількості працівників і швидкому аналізі даних, що й обумовлює необхідність впровадження систем обліку та іншого програмного забезпечення [1]. Наразі автоматизація охоплює не лише великі компанії, але й середній та малий бізнес, де конкурентні переваги здобувають ті, хто впроваджує ефективні інструменти.

Дослідження показують, що використання систем обліку дозволяє збільшити дохід у середньому на 17%, незалежно від масштабів продажу, а продуктивність персоналу — на 20% завдяки автоматизації рутинних процесів. Окупність таких рішень зазвичай становить від 1 до 3 місяців.

У невеликих торгових точках облік ще можливий вручну, але зі зростанням бізнесу виникає ряд проблем, зокрема:

- накопичення нереалізованих товарів;
- труднощі з оперативним оновленням цін;
- втрати через прострочення товарів;
- неточності під час інвентаризації;
- затримки в закупівлях;
- необхідність постійної присутності власника;
- відсутність оперативної аналітики;
- складність обліку міжфілійного переміщення товарів;
- витрати часу на прийом товару;
- ручне введення інформації про поставки тощо [2].

Все більше підприємців автоматизують торговий облік, однак ризик невдалого вибору програмного забезпечення залишається. Проблеми виникають, коли система є складною в освоєнні, не підтримує наявне обладнання, не має необхідного функціоналу чи технічної підтримки.

Автоматизація дозволяє підприємцям контролювати всі процеси в мережі, покращувати обслуговування клієнтів, вести облік товарів, формувати звітність і проводити інвентаризацію [3].

Зв'язок з науковими програмами. Робота виконана на кафедрі програмного забезпечення ВНТУ в межах державної програми «Образний комп'ютер» (постанова КМУ №58 від 27.01.2010).

Мета і завдання дослідження. Метою роботи є підвищення ефективності торгової мережі шляхом розробки мобільного додатку для автоматизованого обліку складських товарів і торгівлі. Це дозволить мінімізувати людський фактор та ефективно використовувати ресурси.

Завдання:

Аналіз існуючих рішень у сфері обліку і торгівлі;

Дослідження технологій реалізації автоматизованих систем;

Вивчення практичних аспектів функціонування систем обліку;

Розробка мобільного додатку в режимі реального часу;

Проведення досліджень і тестування результатів.

Об'єкт дослідження: процес складського обліку додатку.

Предмет дослідження: методи і засоби програмування для реалізації складського обліку.

Методи дослідження: алгоритмічна теорія, Agile-підхід, Scrum, мови HTML, JavaScript і CSS.

Новизна:

Визначено ключовий функціонал для ефективної роботи додатку.

Розроблено структуру бази даних, що відповідає поставленим задачам.

Створено універсальний мобільний додаток для автоматизованої системи, що мінімізує помилки та підвищує ефективність використання ресурсів.

Практичне значення: розроблено функціональний мобільний додаток на основі отриманих теоретичних результатів.

Достовірність: підтверджена коректним формулюванням завдань, використанням відповідних методів та успішною реалізацією розробки згідно з принципами Agile.

Особистий внесок: усі результати отримані автором самостійно.

1 ЗАГАЛЬНИЙ ОГЛЯД ТА АНАЛІЗ СУЧАСНИХ ТЕХНОЛОГІЙ ВЕДЕННЯ ЕЛЕКТРОННОЇ ФОРМИ ДОКУМЕНТАЦІЇ ТА УПРАВЛІННЯ СКЛАДОМ

1.1 Дослідження перспектив переходу компаній на технологію ведення електронної форми документації та управління складом

Перехід на електронну форму документації, співпраці з клієнтом та його обслуговування - це завжди глобальна модернізація. Автоматизація підприємств роздрібної торгівлі починається з наступних процесів:

1. Перехід на нове обладнання, необхідне для фактичного втілення ідеї в життя.
2. Програмне забезпечення. Сама по собі техніка працювати, зрозуміло, не буде. Для організації всієї методики знадобиться якісне програмне забезпечення, яке необхідно купувати окремо. В даному питанні теж необхідний надійний постачальник. Компанія, що реалізує високоякісне обладнання і різні програмні забезпечення для ефективного функціонування і ведення бізнесу.
3. Навчання штату співробітників нововведень [4].

Етапи переходу на технологію умовно можна розбити на кілька ключових кроків, що зображені на рисунку 1.1.

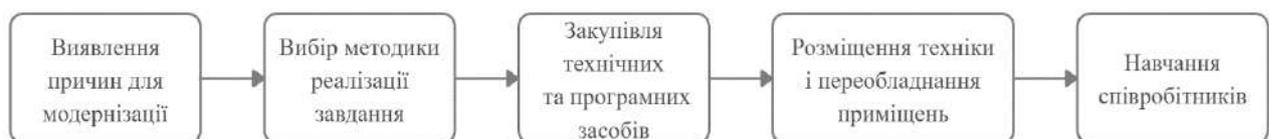


Рисунок 1.1 – Етапи переходу на технологію електронної форми документації

Детальніше про кроки можна сказати наступним чином:

- Виявлення причин для модернізації. Немає сенсу робити даний перехід виходячи із трендів. Існують конкретні причини, і їх потрібно розуміти. Це конкурентна боротьба, збільшення обороту, скорочення витрат, розширення клієнтської бази та підвищення рівня комфорту ведення бізнесу.

- Вибір методики реалізації завдання. Існує безліч видів обладнання і виробників, а також постачальників програмного забезпечення.
- Закупівля технічних та програмних засобів, якщо це необхідно.
- Навчання співробітників.

Існує всього дві основних методики переходу на нову систему [5]. Це повна або часткова варіація. У першому випадку, всі сфери роботи підприємства зазнають якісні зміни, переходять на новий регламент і технічне та програмне оснащення. А в другому - зачіпається лише одна конкретна сфера. Наприклад, складська. Або перехід до окремої категорії товарів.

Вибір часто залежить від можливостей власника, кількісного фактору точок реалізації продукції, складських приміщень та логістики. Автоматизація роздрібно-торговельної мережі - найчастіше це повний перехід. А ось невеликий магазинчик може частково перейти на нову методику поступово, в кілька етапів, рухаючись до повної зміни регламенту.

Всі функціональні особливості даного кроку складно поверхнево, але можна перерахувати головні аспекти:

- Здійснення оперативний пошук товарів.
- Звірка артикулів.
- Використання електронних форм розрахунку.
- Спрощена інвентаризація.
- Контроль якості продукції без втручання людини.
- Повне використання автоматизованих систем ведення бізнесу для роботи.

Останній аспект дуже важливий. Автоматизація магазину роздрібно-торгівлі, особливо товари якої мають невеликий термін придатності, завжди спирається на оперативність. Швидка звірка, управління, отримання інформації та зміна продуктів на прилавках торгових точок - це завдання складне. Тому інтеграція з мобільними гаджетами спрощує і прискорює процес [6].

Конкретика витрат на впровадження автоматизованих систем обліку товарів цілком і повністю залежить від обсягу закупівель, кількості продукції, клієнтської бази, площ, де відбувається реалізація цих товарів. Звичайно, чим значніше об'єкт, тим більше фінансових коштів знадобиться вкласти. Але слід виходити з трьох основних гілок витрат:

- Придбання нового автоматизованого обладнання.
- Витрати на введення в експлуатацію, підключення, перебудову приміщень.
- Купівля програмних засобів автоматизації [7].

Переваги систем автоматизації управління бізнес-процесами:

- Зростання кількості клієнтів, підвищення рівня зручності покупців.
- Оптимізація роботи, вчинення більшого обсягу дій за проміжок часу.
- Позбавлення від тяганини з рутинними завданнями.
- Використання інструментів аналітики в роботі.
- Оперативне оновлення ціни.
- Прискорення роботи персоналу з урахуванням маленького штату.
- Облік безготівкового розрахунку.
- Облік ресурсів, який не вимагає втручання людини.
- Аналітика ефективності і попиту.
- Бухгалтерський облік та аудит.
- Створення супровідної документації та звітності.

Автоматизація супермаркетів, великих торгових центрів - це відмінне рішення для спрощення бізнесу. На таких підприємствах неможливо працювати з клієнтами без технічних нововведень.

Єдина причина, мабуть, чому в процесі автоматизації бізнес-процесу може щось піти не так, це якщо постачальник обладнання та / або програмного забезпечення надав неякісний товар. Якщо техніка не може виконувати свої функції, а програмні засоби не інтегрований належним чином, то вся система не

зможе нормально функціонувати. Тому варто особливу увагу приділити вибору постачальника [8].

Автоматизована роздрібна торгівля - новий підхід до продажу

Слід розуміти, що рано чи пізно, але всі способи реалізації продукції перейдуть на подібну або навіть більш прогресивну форму. Легко зрозуміти величезну кількість плюсів цієї методики. Тому краще перейти раніше, ніж відставати і надолужувати.

Основне обладнання та програмне забезпечення, які беруть участь у стартовому наборі з автоматизації бізнес-процесів:

- Онлайн-каса.
- Сканер.
- Блокіратор каси.
- Спеціальні ваги для ваговій продукції.
- Термінал для безготівкової оплати.
- Автоматизована система управління складськими товарами та / або система управління фінансовою звітністю [9].

Готові рішення для автоматизації магазину роздрібною торгівлю дозволяють вирішити проблему зі складським обліком в одну мить. Усі товарні позиції потрапляють в базу ще на етапі поставки, а далі кожна зміна ретельно фіксується.

Головне, що варто зрозуміти - це прямий вплив забезпечення програмного на ефективність кінцевої роботи всього обладнання та компанії в цілому.

Автоматизоване оформлення цінників

Ще один аспект, здатний в значній мірі прискорити роботу точки відпустки продукції. З огляду на той факт, що в сучасному магазині може бути величезна кількість товарів, найменувань, кожне з яких містить тисячі позицій, оформлення цінників ручним методом - це великі витрати фізичних ресурсів. Їх можна направити в більш логічне і комерційно вигідне русло.

Товарні запаси

У сфері зберігання та наявності облік відіграє не менш важливу роль, ніж якісні умови для утримання. А також оперативна фіксація продуктів, що виходять за межі терміну придатності або експлуатаційного періоду. Щоб продавець завжди мав актуальною інформацією за наявністю позицій на складі або в торговому залі, варто ввести сучасні форми обліку та обміну відомостями [10, 11].

Автоматизований друк чеків

Для досягнення цієї мети знадобиться принтер. Це один з базових пристроїв, потрібних на будь-якому сучасному підприємстві. Таким чином, значно прискорюється сам відпуск продукції, а також сортування, розміщення і управління товарними запасами. Крім того, в деяких випадках чек містить масу додаткової інформації, регламенти акцій, маркетингову і рекламні відомості, фактично виступаючи способом крос-продажів [12].

Поліпшення ефективності роботи касира

Автоматизувати магазин - це в першу чергу означає, переобладнати робочу зону касового персоналу. Саме в цій точці і відбувається основна робота, зчитування знаків товару, зміна його стану в архівах сервера. А крім того, зважування продукції, якщо існує така потреба. Тому принтери, фіскальні реєстратори, каси, сканери штрих-коди, ваги та програмні засоби з автоматизації розміщуються саме тут в залежності від потреб торгової точки.

POS-система

Це комплексне рішення, яке містить в собі цілий набір первинних функцій. Такий технічний засіб зможе замінити безліч різноманітних пристроїв, значно скорочуючи не тільки час, але і кінцеву вартість кожного конкретного приладу.

Система розробляється на базі фіскального реєстратора, але в значній мірі розширює його можливості. Таким чином, якщо порівнювати з ціною кожного обладнання окремо, частина функцій автоматизації роздрібної торгівлі

надається продавцеві безкоштовно. Адже цінова політика комплексу все ж менше, ніж вартість окремого набору елементів [13].

Проте, не лише системи роздрібно́ї торгівлі відкриті до автоматизації їх роботи. Ці ж схеми можна застосувати й до оптової торгівлі. Тут також існує клієнтська база, конкуренція. Також оперативність має прямий вплив на ефективність. А масу завдань можна вирішити людськими силами, знадобиться система, яка самостійно веде облік, реєструє угоди. Відповідно, введення нового обладнання та програмного забезпечення значно підвищить продуктивність діяльності оптовика.

1.2 Аналіз застосування мобільних технологій як засобу підвищення конкурентоспроможності бізнесу

Сьогодні впровадження електронного документообігу та складського менеджменту із застосуванням мобільних пристроїв набуло ключового значення для успішного ведення бізнесу. Якщо ще кілька років тому компанії шукали способи обмеження використання мобільних телефонів працівниками, то нині ситуація кардинально змінилася [14].

Мобільні технології стали невід'ємним елементом робочого середовища, а керівництво активно заохочує їх використання як в офісі, так і поза його межами. Сучасні складські менеджери інтегрують мобільні додатки зі своїми ERP- та WMS-системами для моніторингу запасів і формування звітів, необхідних для оцінки ефективності логістичних процесів. Зміни, спричинені мобільними технологіями, є не просто еволюційними, а трансформаційними.

Основні переваги мобільних рішень:

1. Географічна адаптивність

Сучасні мобільні інструменти забезпечують гнучке управління даними за межами центральних офісів, що дає змогу ефективно координувати роботу віддалених підрозділів [12, 15]. Наприклад, компанії можуть розміщувати склади поблизу клієнтів або постачальників, оптимізуючи логістику. Мобільне сканування та збір даних дозволяють формувати команди у стратегічно вигідних точках, що сприяє гнучкості та інноваційності бізнесу [16].

2. Оптимізація витрат для віддалених локацій

Підприємства, що функціонують у віддалених регіонах, зменшують витрати завдяки функціям дистанційного моніторингу. Менеджери можуть спостерігати за процесами та вносити зміни до систем управління без фізичної присутності на об'єкті. Це знижує операційне навантаження та дозволяє оптимально розміщувати інфраструктуру [17].

3. Удосконалені підходи до інвентаризації

Мобільні пристрої забезпечують оперативне оновлення інформації про запаси у режимі реального часу. Завдяки скануванню штрих-кодів безпосередньо на виробничих ділянках або в малих складських зонах працівники можуть ефективно реєструвати зміни, що сприяє прозорості та точності даних [18].

4. Полегшене навчання персоналу

Завдяки знайомим інтерфейсам смартфонів і планшетів мобільні рішення легше інтегрувати у виробничі процеси. Це зменшує навчальне навантаження та сприяє швидкій адаптації працівників до нових технологій.

5. Підвищення ефективності виїзного обслуговування

Завдяки мобільним пристроям водії можуть електронно оформлювати документи, оновлювати дані про запаси та оперативно передавати інформацію до складу. Це дозволяє забезпечити високу точність і швидкість обслуговування, водночас знижуючи обсяг паперової роботи [19].

У відповідь на зростаючу складність логістики та зростаючі очікування клієнтів, компанії активно інтегрують мобільні додатки до ключових бізнес-систем. Це дозволяє прискорити процеси, розширити контроль і підвищити загальну ефективність. Зараз, коли цифрова трансформація охоплює більшість галузей, саме підприємства, які орієнтуються на мобільні технології, здатні успішно конкурувати в сучасних ринкових умовах [20].

1.3 Аналіз функціонування портативних пристроїв у складських операціях із використанням мобільних технологій

Сучасні тенденції у сфері складської логістики демонструють зміну підходу до ролі складу у бізнес-процесах. Якщо раніше склади переважно вважалися центрами витрат, де ключовим завданням було мінімізувати неефективність та знижувати похибки під час комплектування замовлень, то сьогодні вони перетворюються на стратегічні активи, що сприяють зростанню прибутковості підприємства. Основна увага зосереджується на оптимізації кожного етапу логістичного циклу: приймання, зберігання та відвантаження товарів.

Одним із головних інструментів трансформації складських процесів є портативні пристрої, які значно покращують управління та облік товарів. З огляду на високий попит, виробники активно розробляють широкий спектр таких пристроїв, що охоплюють різні аспекти роботи складу [21].

До найбільш розповсюджених категорій належать:

Міцні планшети, які функціонують як мобільні платформи для сканування штрих-кодів. Завдяки великому екрану вони зручні для роботи з декількома додатками одночасно, що є важливою перевагою під час виконання завдань на складі. Такі пристрої можуть бути змонтовані на транспортних засобах, слугуючи ефективною альтернативою стаціонарним терміналам. Недоліком є їхня масивність та більша вага, зумовлена конструктивною міцністю [22].

Кишенькові персональні комп'ютери та мобільні сканери, які поєднують компактність із функціональністю та забезпечують зручність у польових умовах, особливо у швидкому обліку й пересуванні територією складу.

Прикладом такого пристрою є Zebra ET50 (див. рис. 1.2) — міцний планшет, що працює на базі операційної системи Android. Його вироблено компанією, що спеціалізується на розробці інтегрованих апаратних і програмних рішень. Пристрій може функціонувати автономно або ж у поєднанні з відповідним ПЗ, що дає змогу адаптувати його під потреби конкретного бізнесу.



Рисунок 1.2 – Міцний планшет Zebra ET50, що призначений для корпоративного використання

Мобільні портативні пристрої (КПК) - це все-в-одному програми-сканери. Вони ідеально підходять для сканування на великі відстані і можуть виконувати безліч функцій, такі як отримання, збір, обробка та доставка даних. Важливо зауважити, що мова йде про корпоративні пристрої, спеціально створені для бруду, крапель та унікального середовища складу, а не про пристрої споживчого класу [23].

На рисунку 1.3 зображено КПК Honeywell СТ60, що є представником цієї категорії портативних пристроїв для управління складом. В ньому реалізований інтегрований і масштабований підхід, заснований на загальній апаратно-програмній платформі Mobility Edge™, яка значно полегшує перехід на нові версії операційної системи без шкоди для корпоративної безпеки, надійності, продуктивності і зручності управління. Мобільний комп'ютер Honeywell СТ60 отримав схвалення Android Enterprise Recommended. Це означає, що він відмінно захищений, зручний для розгортання в різних умовах праці і збереже свою актуальність протягом тривалого часу.



Рисунок 1.3 - КПК Honeywell ST60, що є представником портативних пристроїв для управління складом

У КПК передбачені безліч функцій [24], завдяки яким працівники можуть залишатися на зв'язку і зберігати високу продуктивність протягом усього робочого дня. Розширені можливості підключення до мереж полегшують передачу великих файлів, потокове відтворення відео і віддалений доступ до бізнес-додатків, а ємний акумулятор забезпечує роботу пристроїв протягом всієї робочої зміни. Міцна конструкція КПК витримує випадкові падіння і удари, при цьому самі апарати відрізняються ергономічністю і малою вагою для зручного повсякденного використання. Яскраві дисплеї з сенсорним екраном легко зчитуються як в приміщенні, так і під відкритим небом і сприймають торкання пальцем (в тому числі в рукавичці) або стилусом [25].

Переносні мобільні пристрої, що вже містять у собі сканери можуть допомогти працівникам виконувати завдання ефективніше та ергономічніше. Вони особливо корисні в роботі працівників, яким потрібно мати вільні руки, до того ж подібні сканери часто мають невелику вагу, щоб не затруднювати рухи людини. Такі пристрої безпомилково сканують штрих-коди, етикетки, документи. Прикладом є Zebra TC52 зображення якого, представлено на рисунку 1.4.



Рисунок 1.4 - Переносний мобільний пристрій із вбудованим сканером Zebra TC52

Рішення Mobility DNA від Zebra дозволяють більш ефективно використовувати мобільні комп'ютери, розширюючи їх функціональні можливості і спрощуючи розгортання і управління мобільними пристроями.

За допомогою портативних пристроїв, спеціального кріплення та додаткових пристроїв зчитування інформації, що працюють базі мобільних технологій, управління складом перетворюється на єдиний автоматизований процес, що дає змогу ефективно відслідковувати роботу всієї компанії та уникати значної кількості помилок людського фактору, отримавши можливість продуманого перерозподілу робочої та отримання збільшених прибутків [26].

1.4 Уточнена постановка задачі

Дослідивши тенденції розвитку сучасного світу програмних продуктів для автоматизації бізнес-процесів, зокрема управління складськими товарами, поставлено за мету розробити власний мобільний додаток для обліку складських товарів та торгівлі, що буде забезпечувати безперебійний доступ до бази даних товарів компанії та управління ними, а також інтеграцію сучасних функцій для комунікації та відстеження документообігу, що прискорить процес виконання рутинних завдань співробітниками компанії, а також сприятиме зменшенню втрат та ефективного розподілу вивільнених робочих ресурсів на інші області роботи компанії.

2 ДОСЛІДЖЕННЯ РОБОТИ СУЧАСНИХ ПРОГРАМ-АНАЛОГІВ ТА РОЗРОБКА СТРУКТУРИ ВЛАСНОГО МОБІЛЬНОГО ДОДАТКУ

2.1 Аналіз функціональності мобільних додатків-аналогів систем обліку складських товарів

Усі мобільні додатки, які застосовуються в бізнес-середовищі, умовно поділяються на дві основні категорії:

Програми, що задовольняють внутрішні потреби компанії;

Додатки, спрямовані на зовнішню взаємодію з клієнтами: підвищення продажів, розвиток бренду, лояльність споживачів тощо [27].

До внутрішніх рішень належать додатки для:

спільного доступу до документів;

внутрішньої комунікації (месенджери, трекери);

корпоративних соціальних мереж;

управління завданнями та проектами.

Зовнішні мобільні рішення є надзвичайно ефективними в галузях, де важлива взаємодія з клієнтом, зокрема:

туризм;

медицина;

автоіндустрія;

громадське харчування;

індустрія краси;

доставка й роздрібна торгівля;

фітнес і дозвілля [28–30].

Автоматизація роздрібного магазину — актуальне завдання для компаній, що мають декілька торгових точок або планують масштабування. Навіть невеликій мережі складно обійтися без автоматизованого обліку.

Такі системи здебільшого включають:

облік товарів у магазинах і на складах;

контроль готівки в касах;

функціонування касових терміналів.

Серед популярних рішень:

для великих мереж — МойСклад, Бизнес.Ру;

для малого бізнесу — Моє Діло.Торгівля, LiteBox.

Основні критерії вибору програмного забезпечення для обліку товарів:

Імпорт прайс-листів з різних форматів;

Вивантаження звітів про переміщення товарів;

Ведення детальної інформації про наявні, продані й зарезервовані товари;

Інтеграція з онлайн-касами;

Миттєве оновлення бази при змінах;

Автоматизоване формування замовлень;

Аналіз популярності товарів і формування сезонних звітів;

Облік повернень і контроль товарів, що «зависли»;

Автоматична корекція роздрібною ціни відповідно до закупівельної;

Сегментований облік фінансів;

Формування супровідної документації;

Підтримка інтеграції з CRM, бухгалтерією та модулями [31–32].

Додатковими перевагами є можливість:

інтеграції з хмарними технологіями;

доступу до бази з будь-якого пристрою;

роботи через веб-інтерфейс;

стабільності роботи без зависань і збоїв;

відкритості API для кастомізації.

Важливо також враховувати:

простоту інтерфейсу;

технічну підтримку 24/7;

гнучкість у реалізації різних функцій — від базового обліку до глибокої аналітики;

оптимальне співвідношення ціни і якості [33–34].

Огляд мобільного додатку «ЕКАМ.Бізнес»

Серед доступних рішень для малого бізнесу слід відзначити «ЕКАМ.Бізнес» — багатофункціональний мобільний додаток, який забезпечує:

облік доходів і витрат;

- перегляд фінансової статистики;
- формування собівартості;
- управління базою контрагентів;
- створення та супровід замовлень;
- ведення записів оплат на рахунок або картку.

Реєстрація займає лічені хвилини — необхідно лише вказати електронну адресу, реквізити компанії та номер телефону. Доступна також веб-версія програми, яка відкривається через QR-код. Додаток має версії для Android та iOS [35].

На рисунках 2.1–2.4 подано інтерфейс програми «ЕКАМ.Бізнес».

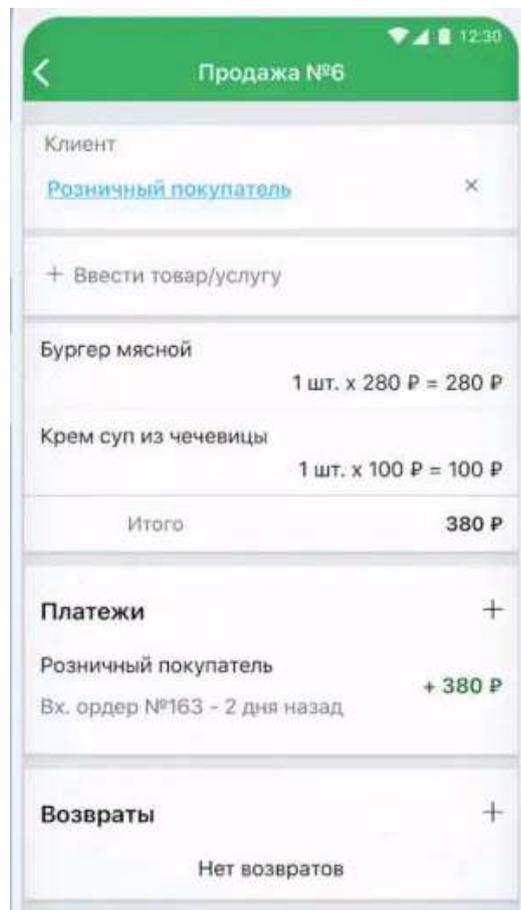


Рисунок 2.1 - Ведення продаж і замовлень в мобільному додатку ЕКАМ.Бізнес

The screenshot shows a mobile application interface with a green header 'Финансы'. Below the header is a navigation bar with a left arrow, the text 'июль 2018', and a right arrow. The main content is a table with two columns: 'Статья' (Article) and 'Прибыль / убыток' (Profit / Loss). The table lists various financial items and their corresponding values in Russian Rubles (Р).

Статья	Прибыль / убыток
Выручка	58 290 Р
Себестоимость	16 083 Р
Валовая прибыль	42 207 Р
Операционные расходы	21 000 Р
Списания	0 Р
Аренда	20 000 Р
Зарботная плата	1 000 Р
Операционная прибыль	21 207 Р
Налоги и сборы	0 Р
Чистая прибыль	21 207 Р

Рисунок 2.2 – Звіт про прибутковість бізнесу в мобільному додатку ЕКАМ.Бізнес

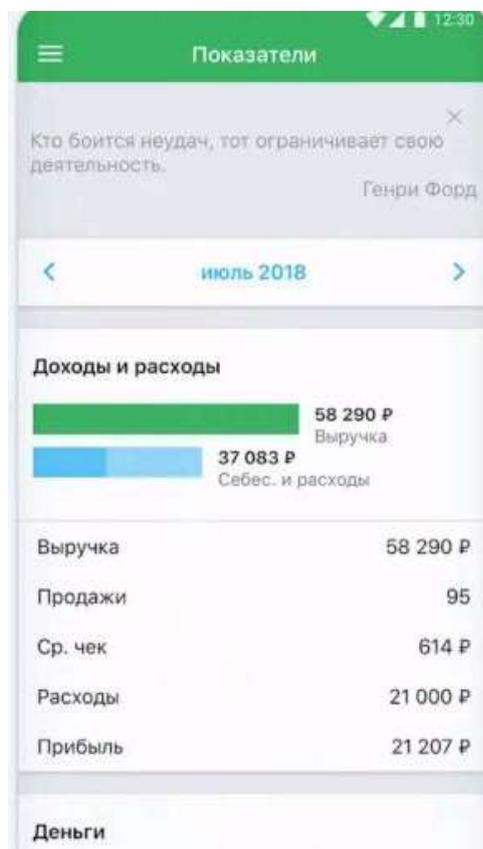


Рисунок 2.3 – Щомісячні звіти в мобільному додатку ЕКАМ.Бізнес



Рисунок 2.4 – Відслідковування виручки та грошей у касі в мобільному додатку
ЕКAM.Бізнес

Мобільний додаток «Trini POS» перетворює смартфон на повноцінну POS-систему для ведення обліку продажів. Процес реєстрації займає менше хвилини, і вже після підтвердження електронної пошти можна починати додавати товари до бази.

Основні можливості додатку:

- створення бази товарів із поділом на категорії;
- ручне введення продажів та формування чеків;
- доступ до статистики реалізації;
- аналіз популярності позицій;
- контроль залишків.

Функціонал для управління складом тут майже відсутній, тому додаток найбільше підходить для кафе, ресторанів, салонів краси чи інших закладів з невеликим потоком клієнтів. Програма доступна як для Android, так і для iOS.

На рисунках 2.1 – 2.8 представлені скріншоти робочих вікон мобільного додатку Trini POS.

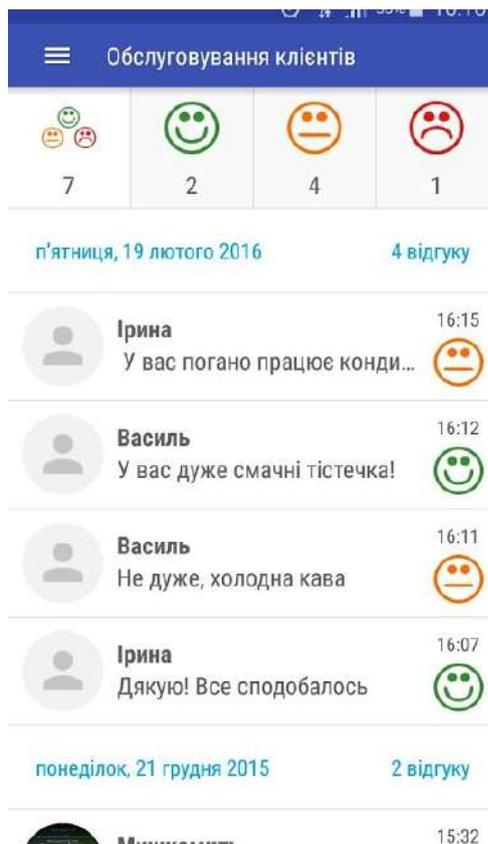


Рисунок 2.5 – Відображення відгуків клієнтів в мобільному додатку Trini POS

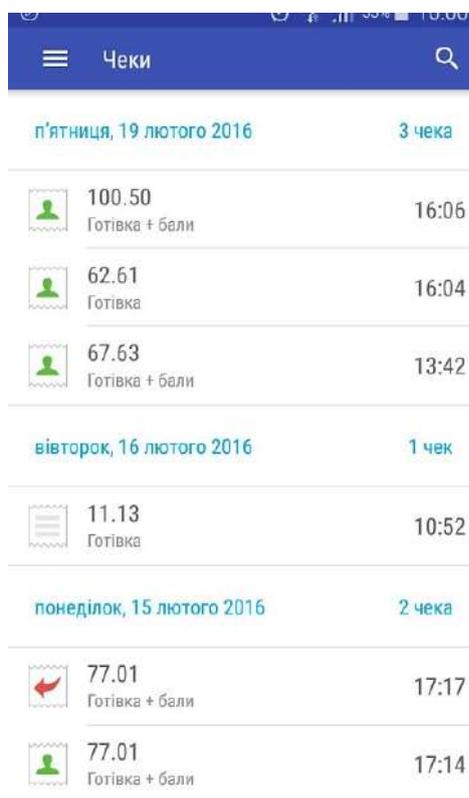


Рисунок 2.6 – Відображення закритих чеків в мобільному додатку Trini POS

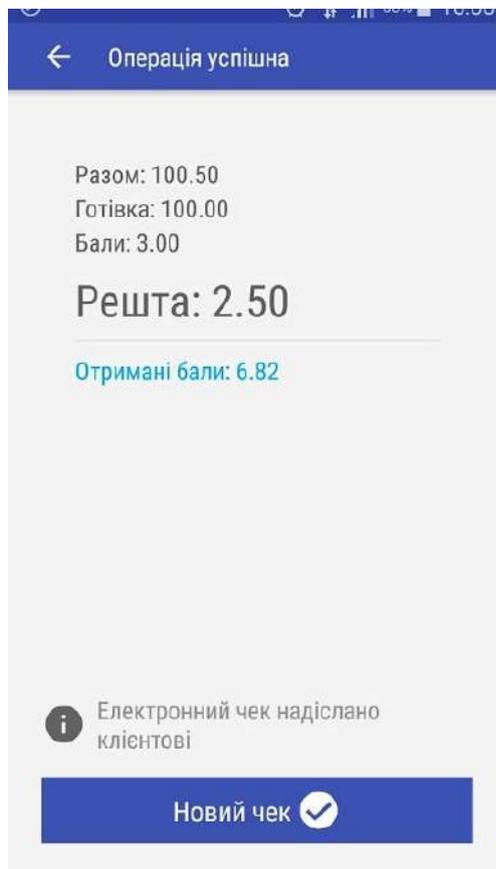


Рисунок 2.7 – Відображення повідомлення про успішне закриття чеку в мобільному додатку Trini POS

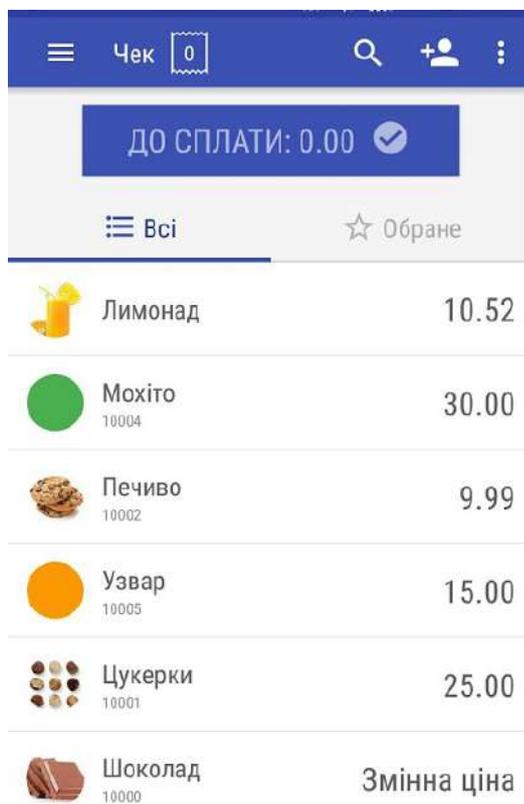


Рисунок 2.8 – Відображення чеку в мобільному додатку Trini POS

На рисунках 2.9 - 2.10 представлено роботу мобільного додатку на POS-терміналах.

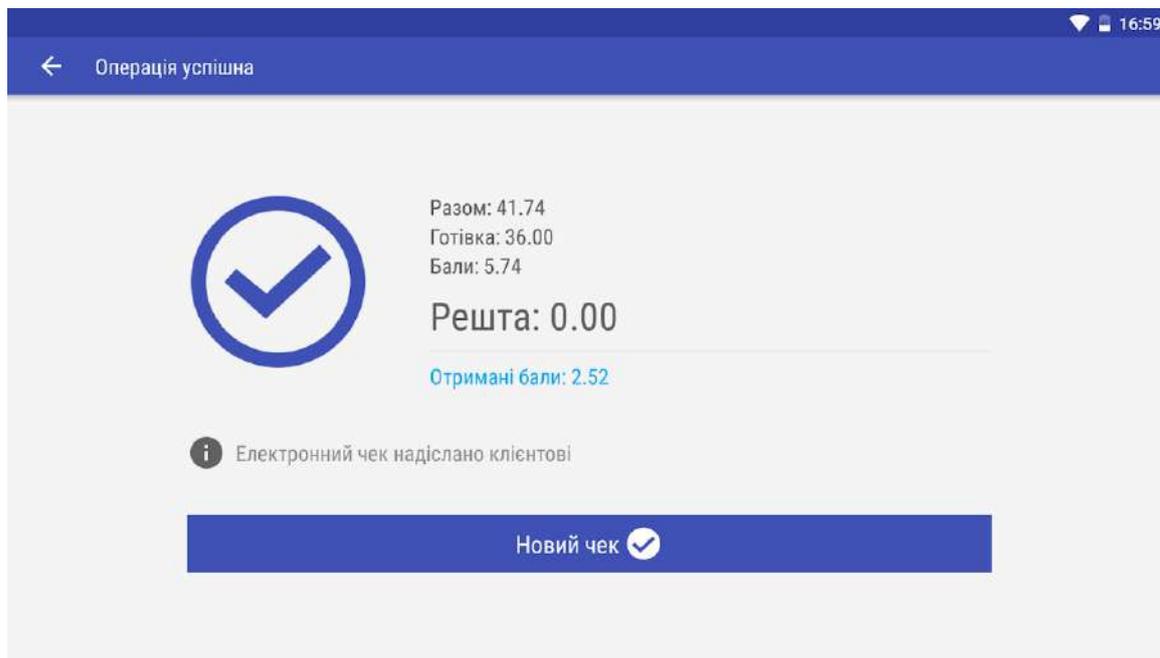


Рисунок 2.9 – Закриття чеку в мобільному додатку Trini POS

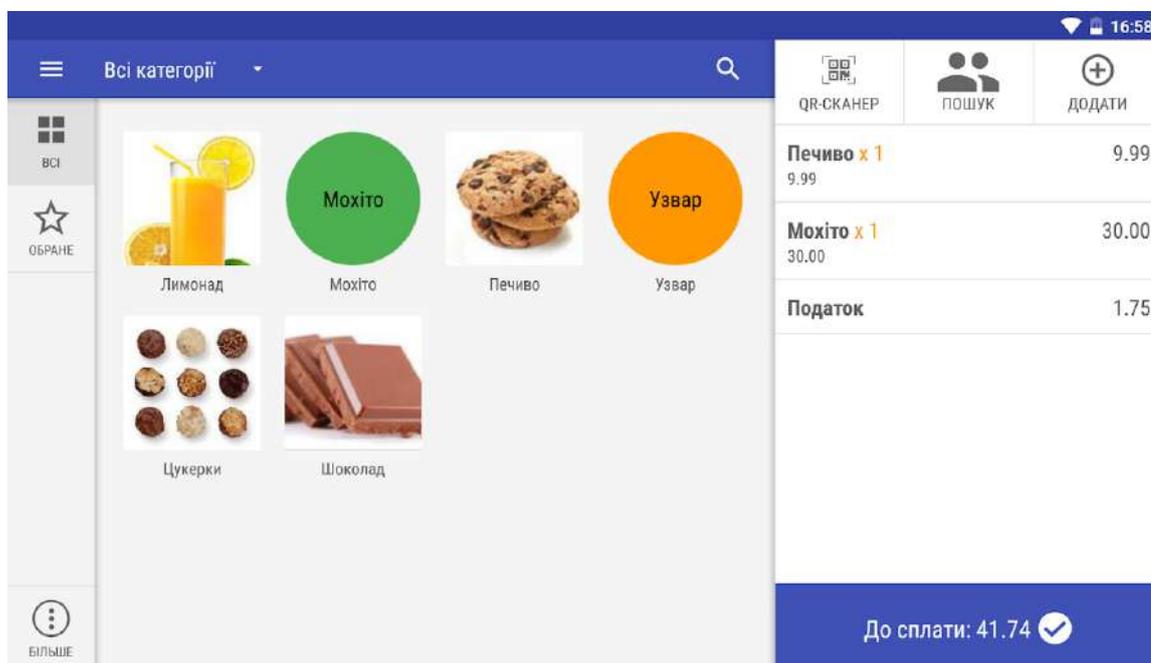


Рисунок 2.10 – Відкритий чек в мобільному додатку Trini POS

Мобільний додаток «Мобільна каса і склад» — це зручний та ефективний інструмент для ведення обліку роздрібних продажів і контролю залишків товарів. Він вирізняється простим у використанні інтерфейсом і надає такі можливості:

- створення довідників категорій, товарів, контрагентів та користувачів;
- оформлення документів закупівлі, продажу та списання товарів;
- ведення обліку з функціями друку документів або відправки їх на e-mail;
- друк чеків із підтримкою ККТ АТОЛ.
- Підтримка штрих кодів в обліку і пошуку товару.

Звітність включає в себе:

1. Список закупівель.
2. Список продажів.
3. Залишки товару.
4. Товарні обертів.
5. Грошові обороти.

У звітності реалізована можливість відправки по e-mail і експорту в *.CSV форматі.

Додаток підтримує зручний облік продажів і закупівель за користувачами та змінами. Передбачено експорт і імпорт усіх даних, що дає змогу переносити програму між різними пристроями та створювати резервні копії. Рішення підходить для пунктів видачі замовлень, роздрібних магазинів, служб кур'єрської доставки або виїзної торгівлі. Завантажити додаток можна на пристрої з Android.

На рисунках 2.11 – 2.13 представлені скріншоти робочих вікон мобільного додатку Мобільна каса і склад.

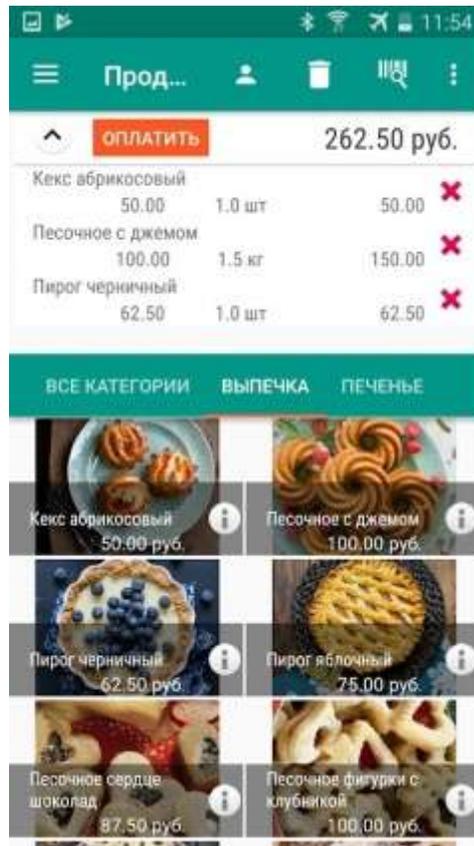


Рисунок 2.11 – Закриття чеку в мобільному додатку Мобільна каса і склад

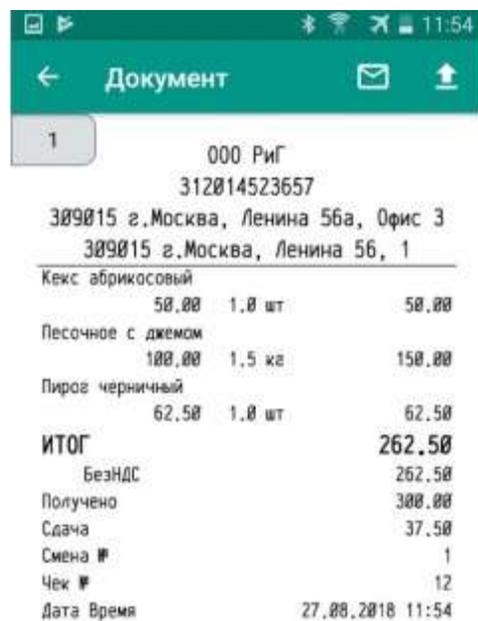


Рисунок 2.12 – Створення та відображення електронного чеку в мобільному додатку Мобільна каса і склад

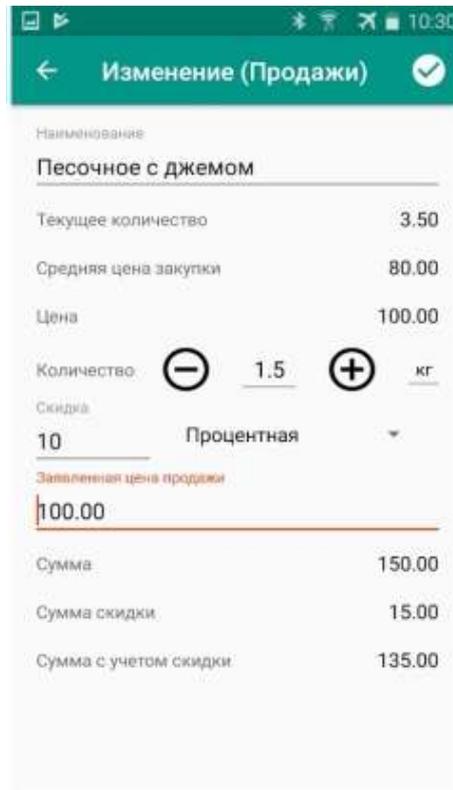


Рисунок 2.13 – Процес утворення чеку в мобільному додатку Мобільна каса і склад

Мобільний додаток «1С: Торговля і Склад. Облік продажів. 1С:Підприємство», створений за участі компанії 1С, являє собою повнофункціональне рішення для ведення торгівлі та базової бухгалтерії.

Серед основних можливостей:

- складський облік;
- оформлення продажів;
- ведення клієнтської бази;
- взаєморозрахунки з контрагентами;
- підтримка сканера штрих-кодів;
- фінансовий облік.

Окрім цього, за додаткову оплату доступний розширений функціонал: синхронізація з іншими пристроями, управління обліком одразу у кількох магазинах тощо. Також передбачене підключення баз даних із інших продуктів 1С. Додаток працює на Android та iOS.

На рисунках 2.14 – 2.17 представлені скріншоти робочих вікон мобільного додатку 1С: Торговля і Склад. Облік продажів. 1с Підприємство.

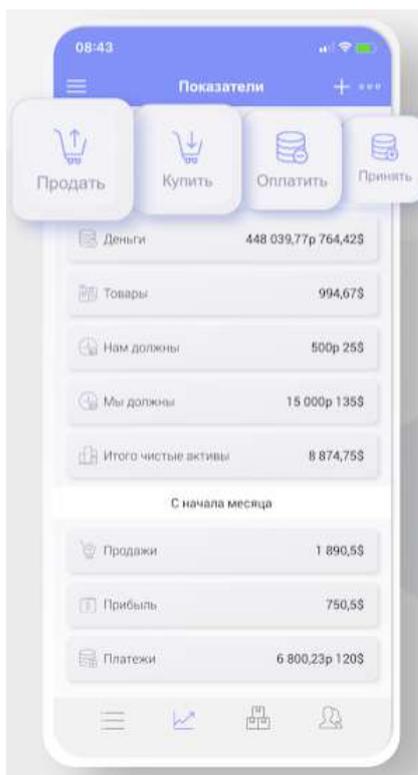


Рисунок 2.14 – Основные возможности в мобильном додатку 1С: Торговля і Склад.

Облік продажів. 1с Підприємство

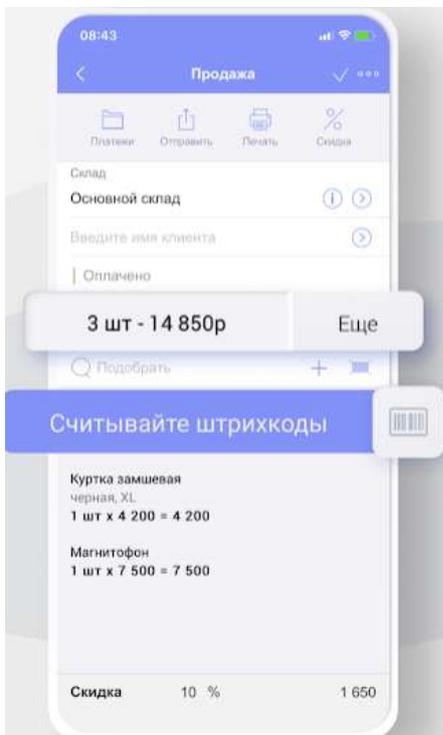


Рисунок 2.15 – Закрытый чек в мобильном додатку 1С: Торговля і Склад. Облік продажів. 1с Підприємство

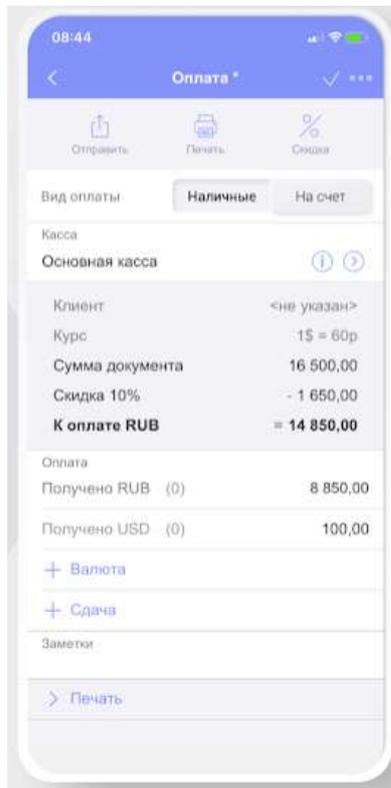


Рисунок 2.16 – Возможность работы с разными валютами в мобильном додатку 1С: Торговля і Склад. Облік продажів. 1с Підприємство

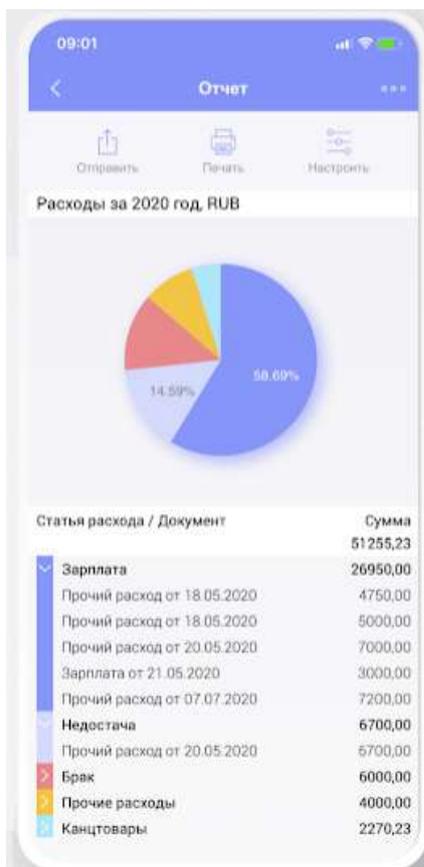


Рисунок 2.17 – Анализ витрат в мобильном додатку 1С: Торговля і Склад. Облік продажів. 1с Підприємство

Мобільний додаток «CloudShop облік товарів в магазині. Каса і склад»

- програма обліку товарів і рішення для автоматизації торгівлі, містить все для управління фірмою: від обліку надходження товарів і ведення бази клієнтів до збору статистики продажів і бухобліку в торгівлі [41, 42].

Можливості обліку товарів:

- Робота на складі: від інвентаризації товарів до синхронізації складу і торгівлі. Магазин-склад буде оснащений мобільною торгівлею: касир продавець зможе проводити продаж через додатковий додаток CloudShop: POS.
- Звіти відстежують статистику продажів, статтями витрат, роблять доступним бухгалтерський облік в торгівлі і облік магазину.
- Можна відслідковувати не тільки склад і торгівлю, а й базу клієнтів.
- Складська логістика. Можна здійснити списання товару, зробити переоцінку, зробити повернення, інвентаризацію товарів.

Додаток підходить для будь-якого торгового бізнесу, а особливо тим, чий профіль - продаж оптом. В оптовій торгівлі автоматизація магазину повинна відображати потрібні операції (прихід витрата, переоцінка, дії касира продавця, складську логістику) - це і є облік в магазині [43].

На рисунках 2.18 – 2.21 представлені скріншоти робочих вікон мобільного додатку 1С: Торговля і Склад. Облік продажів. 1с Підприємство.

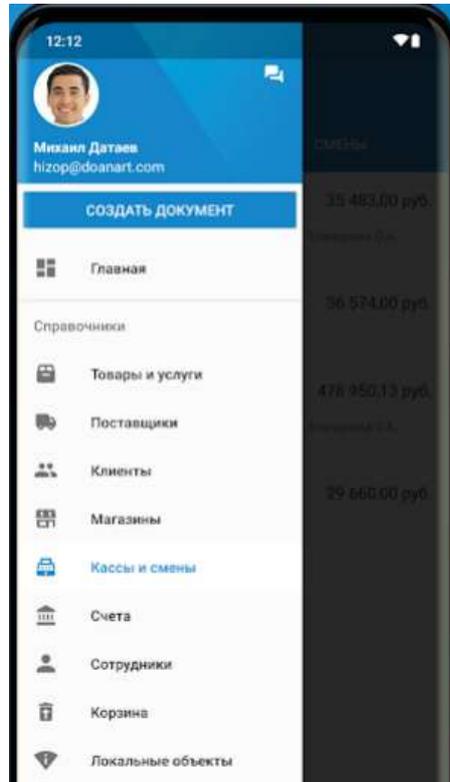


Рисунок 2.18 – Відображення облікового запису в мобільному додатку CloudShop облік товарів в магазині. Каса і склад

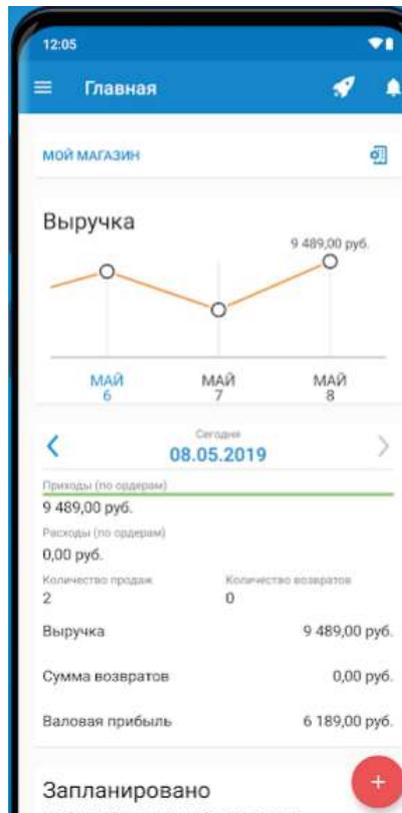


Рисунок 2.19 – Відображення контролю за витратами в мобільному додатку CloudShop облік товарів в магазині. Каса і склад

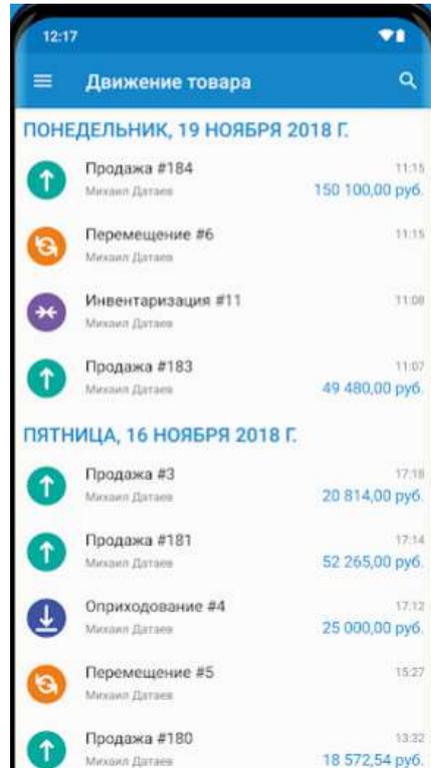


Рисунок 2.20 – Відображення історії переміщень в мобільному додатку CloudShop облік товарів в магазині. Каса і склад



Рисунок 2.21 – Використання сканеру штрих-коду в мобільному додатку CloudShop облік товарів в магазині. Каса і склад

Серед запропонованих програм-аналогів автоматизованих систем управління складськими товарами і торгівлі є мобільні додатки, що дійсно варті уваги, проте якщо у власника бізнесу виділений бюджет на проектування і розробку власної АСУ складськими товарами, то варто скористатись можливістю створення мобільного додатку під потреби власного бізнесу.

2.2 Визначення функціональних можливостей автоматизованої системи, що розробляється

Дослідивши функціональні можливості мобільних програм аналогів автоматизованих систем управління складськими товарами та торгівлі, складемо список функцій, які буде реалізовувати програмний комплекс, що розробляється. Система буде призначена для автоматизації діяльності невеликого магазину чи торгової точки та буде здатна:

- реєструвати торгові точки та склади, використовуючи користувацьку топологію мережі;
- вести облік покупців та постачальників;
- реєструвати прихід та відбування товару зі складу;
- відслідковувати залишки товарів;
- реєструвати фінансові операції;
- вести історію фінансових операцій;
- надавати звіти на основі даних таблиць бази даних;
- виконувати імпорт та експорт даних;
- автоматично виконувати резервне копіювання;
- система повинна бути захищеною та надійною.

Відповідно до цього в рамках комплексної магістерської кваліфікаційної роботи «Розробка автоматизованої системи обліку складських товарів та підтримки торгівлі. Частина 2. Розробка мобільного додатку» буде спроектовано та розроблено структуру бази даних системи, UML-діаграми функціонування та безпосередньо саме програмне рішення автоматизованої системи.

2.3 Проектування бази даних системи

За основу моделі бази даних для мобільного додатку, що розробляється, взято спроектовану модель із магістерської кваліфікаційної роботи «Розробка автоматизованої системи обліку складських товарів та підтримки торгівлі. Частина 1. Розробка бази даних і десктопного додатку». Модель бази даних спроектована спеціально під десктопне та мобільне рішення автоматизованої системи обліку складських товарів та підтримки торгівлі.

Продублюємо зображення ER-моделі бази даних на етапі визначення сутностей та зв'язків між ними на рисунку 2.22.

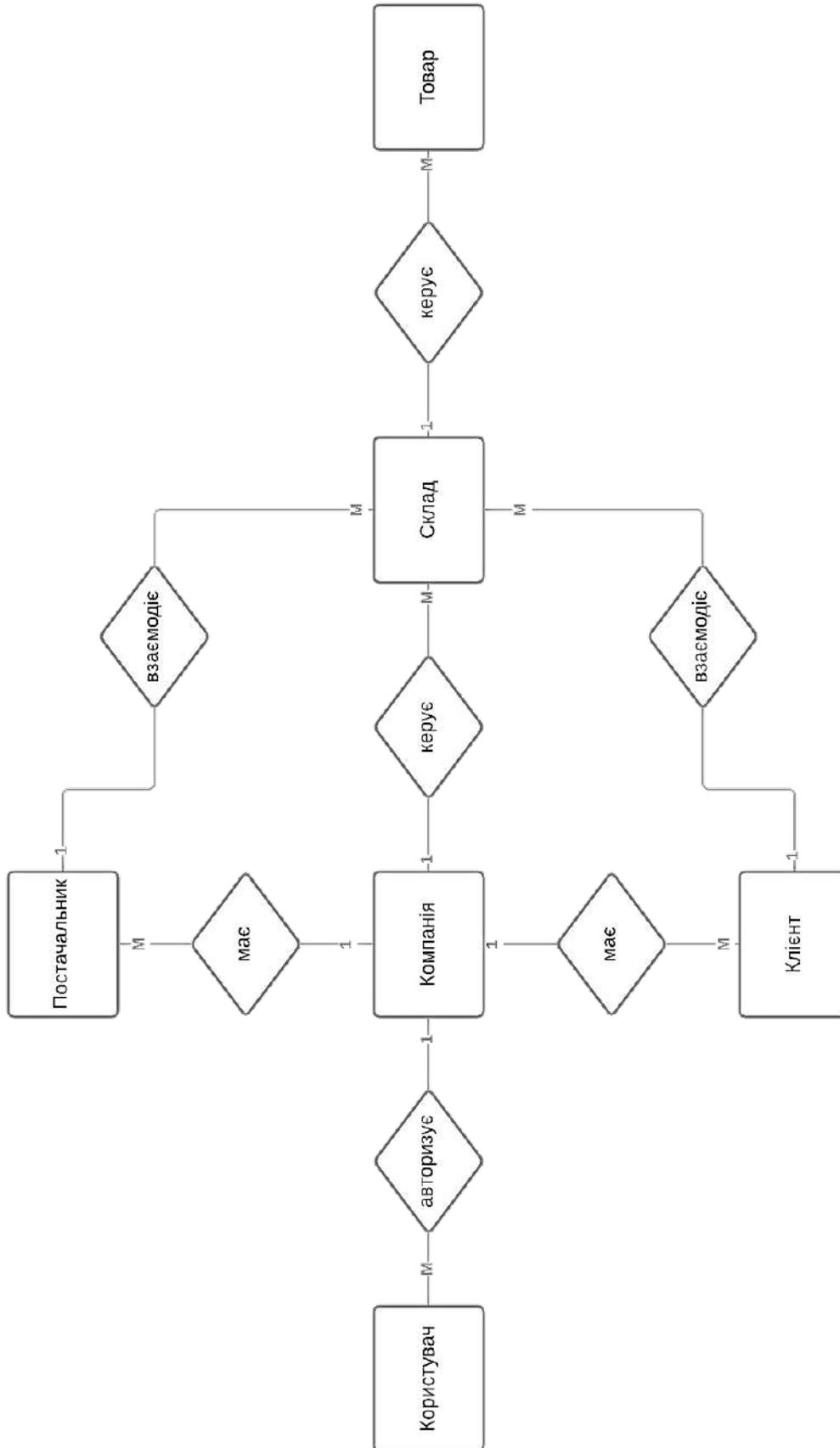


Рисунок 2.22 – ER-модель БД

Для кожної із сутностей визначимо необхідний набір атрибутів, що покриває усю необхідну про них інформацію. Атрибути сутностей представлено у вигляді зведеної таблиці 2.1.

Таблиця 2.1 – Атрибути сутностей БД

Сутність	Первинний ключ	Атрибути
Торгова точка	Код торгової точки	- Назва торгової точки - Адреса - Телефон
Користувач	Id користувача	- Ім'я користувача - Логін / e-mail - Пароль - Телефон
Клієнт	Id клієнта	- Ім'я клієнта - Телефон - Номер дисконтної карти
Постачальник	Id постачальника	- Ім'я постачальника - Телефон - e-mail - Адреса
Склад	Id складу	- Назва складу - Адреса - Телефон
Товар	Код товару	- Назва товару - Кількість - Характеристики

Також продублюємо структурну схему бази даних, що роз'яснює зв'язки між сутностями та містить перелік атрибутів, необхідних для утворення цих зв'язків, на рисунку 2.23.

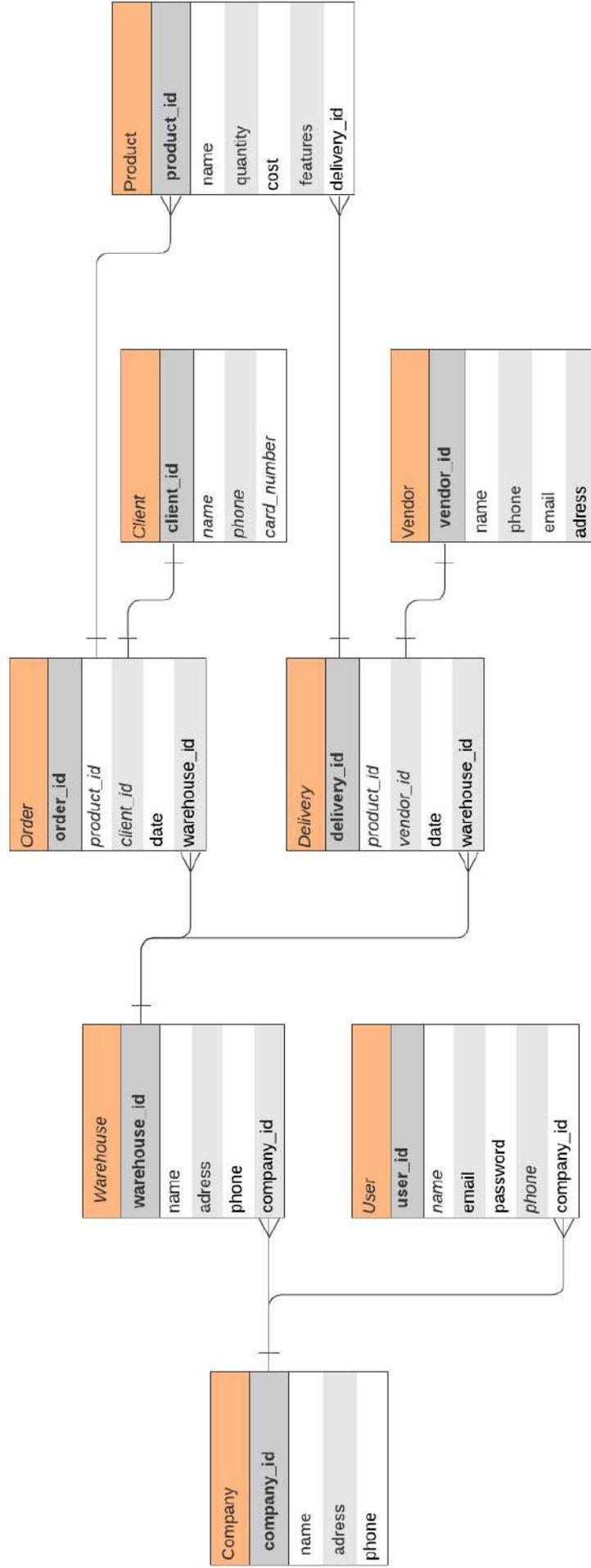


Рисунок 2.23 - Структурна схема бази даних системи

Сутності представляються у вигляді таблиць бази даних, де атрибути являються полями [45-47]. Так, для функціонування мобільного додатку автоматизованої системи обліку складських товарів та підтримки торгівлі, що розробляється в межах магістерської кваліфікаційної роботи, необхідно створити наступні таблиці в базі даних:

- Company – інформація про компанію, під іменем якої працюють усі магазини та склади мережі.
- User – інформація про адміністраторів чи продавців магазину, в яких є доступ до системи.
- Warehouse – інформація про торгові точки чи склади компанії.
- Order – інформація про замовлення всередині конкретної компанії.
- Delivery – інформація про поставки товарів всередині конкретної компанії.
- Client – інформація про клієнтів компанії.
- Vendor – інформація про постачальників товарів.
- Product – власне інформація про самі товари, що потрапляють на склади чи торгові точки конкретної компанії.

В процесі проектування бази даних в межах магістерської кваліфікаційної роботи «Розробка автоматизованої системи обліку складських товарів та підтримки торгівлі. Частина 1. Розробка бази даних і десктопного додатку» було проведено нормалізацію таблиць бази даних, тобто розділення даних по окремих зв'язаних таблицях, що допоможе уникнути надлишковості даних, дублювання та порушення їх цілісності при редагуванні [48].

2.4 Розробка UML-діаграм

UML-діаграми використовуються зазвичай для спрощення великої кількості письмової документації, забезпечуючи візуальну модель того, як працює система та як вона спроектована.

Мова UML призначена для задоволення широкого кола потреб за допомогою ряду різних типів діаграм, які показують систему з різних точок зору. Ці типи діаграм можна згрупувати за двома основними категоріями [49]:

- структурні діаграми, що показують статичну структуру системи та її частин, а також те, як вони пов'язані між собою";
- діаграми поведінки, що показують, відповідно, динамічну поведінку об'єктів у системі.

UML-діаграми тісно пов'язані із великим попереднім дизайном, що ще називається каскадною моделлю процесу розробки програмного рішення, в якому вимоги до процесу, а також сам дизайн, який відповідає цим вимогам, визначається на певному рівні детально та заздалегідь до початку проекту [50]. Роль та місце UML-діаграм в цій моделі зображено на рисунку 2.24.



Рисунок 2.24 – Роль UML-діаграм в каскадній моделі процесу розробки програмного рішення

Із переходом до гнучкої технології розробки програмного забезпечення, постає питання щодо доцільності написання UML-діаграм, адже:

- вимоги до проекту та дизайну програмного забезпечення, як правило, не визначаються детально до початку проекту;
- як вимоги, так і дизайн еволюціонують і доповнюються в процесі реалізації проекту;
- сама технологія передбачає набагато менше уваги до офіційної проектної документації [51].

Тому складність використання UML-діаграм у випадку – очевидна.

На початку команді проекту ще недостатньо інформації для детального визначення вимог та розробки попереднього проекту системи. До того ж, для постійного оновлення діаграм може знадобитися багато часу.

Однак є кілька причин, чому UML-діаграми все ще можуть бути корисними:

- Підтримка - використання UML-діаграм може бути корисним для підтримки системи після завершення розробки;
- Архітектура - використання UML-діаграм також може мати певне значення для визначення та стандартизації архітектури рішення.

Обидві ці функції залежать в значній мірі від складності системи, критичності ділової функціональності, яку вона надає, та загальної стратегії підтримки системи.

В межах виконання магістерської кваліфікаційної роботи, а саме розробки мобільного додатку автоматизованої системи обліку складських товарів та підтримки торгівлі, було вирішено все ж використати UML-діаграми для створення візуальної моделі системи, що будуть відображати структуру системи, а також деякі аспекти її роботи [52].

2.4.1 Розробка UML-діаграми варіантів використання

UML-діаграми варіантів використання застосовуються для збору вимог до системи, враховуючи як внутрішні, так і зовнішні впливи. Здебільшого це вимоги до її проектування. Тому під час аналізу системи з метою визначення її функціональних можливостей створюються варіанти використання та визначаються актори (дійові особи).

Після завершення початкового етапу завдання діаграми варіантів використання будуються для відображення зовнішнього вигляду системи. Основні цілі таких діаграм:

- збір вимог до системи;
- наочне представлення її зовнішнього вигляду;
- визначення зовнішніх і внутрішніх чинників, що впливають на роботу системи;
- відображення взаємодії між вимогами акторів.

Такі діаграми розглядають для високорівневого аналізу вимог. У процесі моделювання першочергово враховуються функціональні можливості системи,

що й формують основу діаграм. Іншими словами, UML-діаграми варіантів використання відображають системні функції у впорядкованому вигляді [53].

Актори визначаються як будь-що, що взаємодіє із системою. Це можуть бути як користувачі-люди, так і внутрішні чи зовнішні програмні модулі.

У випадку діаграми варіантів використання, розробленої для мобільного додатку автоматизованої системи обліку складських товарів і підтримки торгівлі, акторами виступають Користувач системи та База даних.

Для користувача передбачено наступні прецеденти по управлінню:

- Товарами;
- Затратами.
- Списками складів
- Списками контрагентів.
- Звітністю.

Кожен із зазначених вище варіантів використання передбачає звернення до Бази даних, тому вона на діаграмі винесена окремим актором, що обробляє запити по перегляду, створенню та редагуванню інформації в системі.

UML-діаграма варіантів використання системи, що розробляється, представлена на рисунку 2.25.

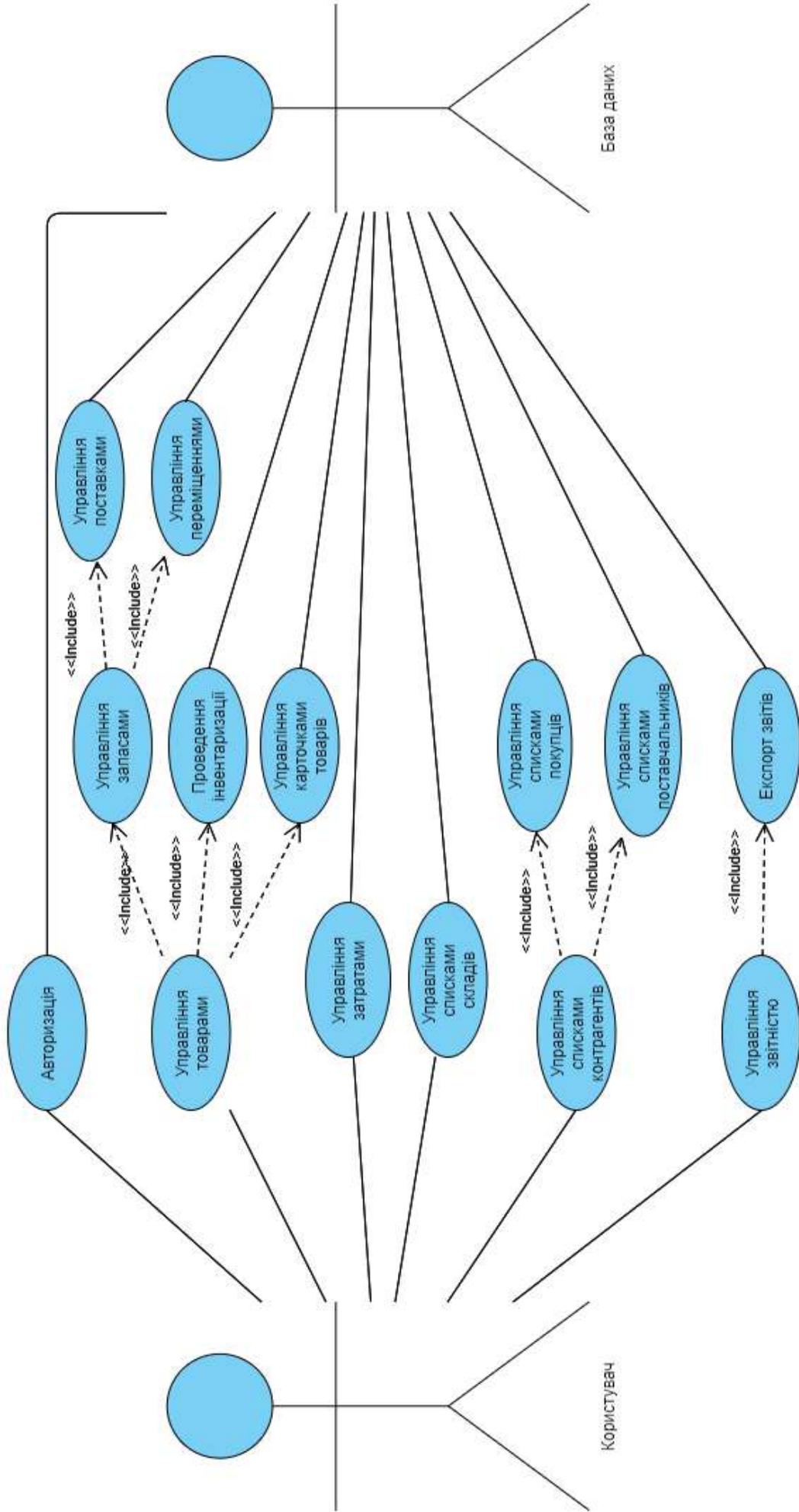


Рисунок 2.25 - UML-діаграма варіантів використання

2.4.2 Розробка UML-діаграми компонентів

Діаграма компонентів відображає структурні взаємозв'язки компонентів програмного рішення, та, в основному, використовується при роботі зі складними системами з багатьма компонентами. Компоненти спілкуються між собою за допомогою інтерфейсів, а інтерфейси пов'язані за допомогою роз'ємів [54]. На рисунку 2.26 UML-діаграма компонентів, автоматизованої системи що розробляється.

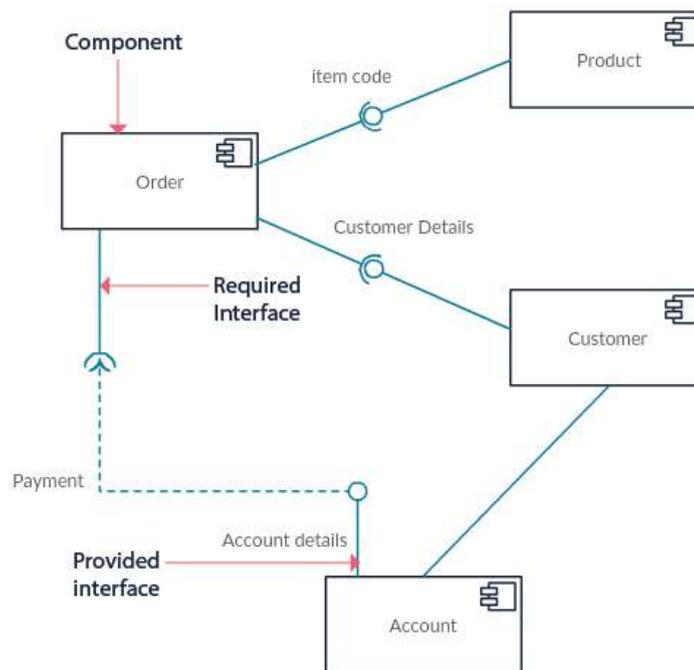


Рисунок 2.26 - UML-діаграма компонентів

Діаграми компонентів використовуються для візуалізації фізичних компонентів у системі, таких як бібліотеки, пакети, файли тощо. На діаграмі представлено чотири класи системи, що містять в собі інформацію про товар та повне замовлення, що проводиться через склад, про покупця та користувацький клас продавця, що керує складом. Те як взаємодіють ці класи показано на рисунку.

Діаграми компонентів також можна описати як статичний вигляд реалізації системи. Статична реалізація представляє організацію компонентів у конкретний момент. Однокомпонентна діаграма не може представляти всю систему, але колекція діаграм використовується для представлення цілого [55].

3 РОЗРОБКА МОБІЛЬНОГО ДОДАТКУ АВТОМАТИЗОВАНОЇ СИСТЕМИ ОБЛІКУ СКЛАДСЬКИХ ТОВАРІВ ТА ПІДТРИМКИ ТОРГІВЛІ

3.1 Розробка мобільного додатку для ОС Android

З моменту випущення операційної системи Android компанією Google у 2008 році, стартував напрям розробки мобільних додатків під дану ОС і до теперішнього часу стрімко розвивається, оскільки ОС Android встановлена на більш ніж 83% девайсів [56, 57].

Проте в цього напрямку є незручна специфіка, яка залежить від:

- Різноманітність пристроїв, орієнтованих на потреби користувачів, створює певні труднощі для розробників мобільних додатків як у сфері апаратного, так і програмного забезпечення. До цього відносяться різні розміри екранів, розташування датчиків, наявність додаткових камер тощо.
- Ще однією проблемою є широкий спектр версій операційних систем, встановлених на різних пристроях. Це вимагає постійного врахування особливостей відображення додатку, оскільки елементи управління можуть відрізнитися зовнішнім виглядом або логікою роботи залежно від версії ОС. У результаті програма може некоректно працювати як у новіших, так і в старіших версіях системи [58].
- Програмні бібліотеки та методи також постійно змінюються: одні стають застарілими і потребують заміни, інші — з'являються нові. Тому розробники змушені обирати між підтримкою найновіших можливостей ОС та забезпеченням сумісності з більшою кількістю пристроїв.
- Особливістю Android є архітектура додатків, що складається з незалежних компонентів — активіті та фрагментів. Такий підхід дозволяє адаптувати програми для пристроїв із різним рівнем продуктивності: за потреби окремі частини програми можуть вивантажуватись із пам'яті, що допомагає економити ресурси.

- Важливу роль відіграє Material Design — концепція створення інтерфейсів користувача. Її офіційна документація містить сотні матеріалів, які пояснюють принципи підходу та демонструють приклади правильного і неправильного використання елементів інтерфейсу.
- Наразі Google рекомендує використовувати мову програмування Kotlin замість Java. Хоча різниця між ними не така суттєва, як між Objective-C і Swift для iOS, підходи дещо відрізняються. Kotlin працює на віртуальній машині Java та повністю сумісний із Java-кодом, що дозволяє використовувати обидві мови одночасно — як у межах середовища розробки, так і в одному додатку [59].
- Тестування додатків зазвичай проводиться на великій кількості фізичних пристроїв (не лише на емуляторах), що зменшує ймовірність помилок, але не гарантує стабільної роботи на всіх моделях.
- Для ефективної розробки мобільних програм потрібні спеціалізовані інструменти та середовища, серед яких:
 - Android Studio — офіційне IDE для створення Android-додатків, що підтримує Java, C++ та Kotlin і дозволяє інтегрувати сторонні сервіси;
 - Android SDK — набір інструментів для проєктування та тестування мобільних програм, знання якого є суттєвою перевагою для розробника;
 - Git — система контролю версій, інтегрована з Android Studio, яка забезпечує збереження етапів розробки та спільну роботу команди над проєктом.

Для продуктивної розробки мобільних додатків потрібні спеціально програмні інструменти та середовища розробки, такі як:

- Android Studio – офіційне середовище програмування для Android. Створення продукту для даної платформи починають з установки Android Studio - інтегрованого середовища розробки (IDE). Цей

комплекс програмних засобів підтримує різні мови програмування - Java, C ++, Kotlin - і пропонує можливості для взаємодії з іншими додатками.

- Android SDK – універсальний набір інструментів для створення і тестування мобільних додатків з набором функціональних можливостей. Знання цього продукту - великий плюс для фахівця.
- Git – система контролю версій проекту, з яким взаємодіє Android Studio. Вона дозволяє фіксувати етапи розробки і допомагає розробникам взаємодіяти всередині команди.
- Ефективні бібліотеки, які можна підключити через систему автоматичної збірки Gradle [60-62].

Для створення мобільного додатку автоматизованої системи спочатку необхідно було скачати останню версію Android Studio та обрати Створення нового проекту, що зображено на рисунку 3.1.

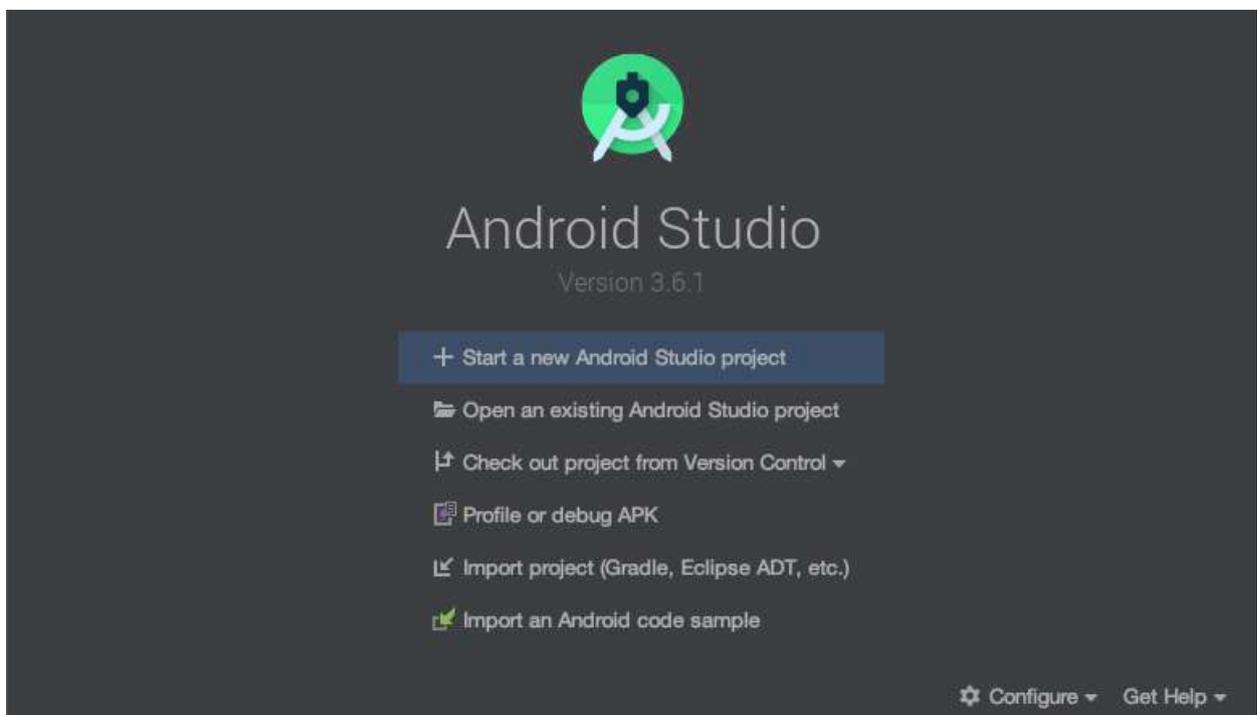


Рисунок 3.1 – Стартове вікно середовища програмування Android Studio

За замовчуванням Android SDK не включає в себе все необхідне для початку розробки. SDK розділяє інструменти, платформи та інші компоненти на пакети, які можна завантажити в міру необхідності за допомогою Диспетчера Android SDK.

Android Studio дозволяє вибирати форм-фактори або категорії цільових пристроїв для кожної програми. При розробці автоматизованої системи управління в межах магістерської кваліфікаційної роботи було використано мінімальні параметри SDK API 14: Android 4.0 (IceCreamSandwich) [63].

Після заповнення ідентифікатора додатку відкривається вікно проекту, що являє собою дерево та представлене на рисунку 3.2, основними гілками якого є «Додатки» і «Скрипти Gradle».

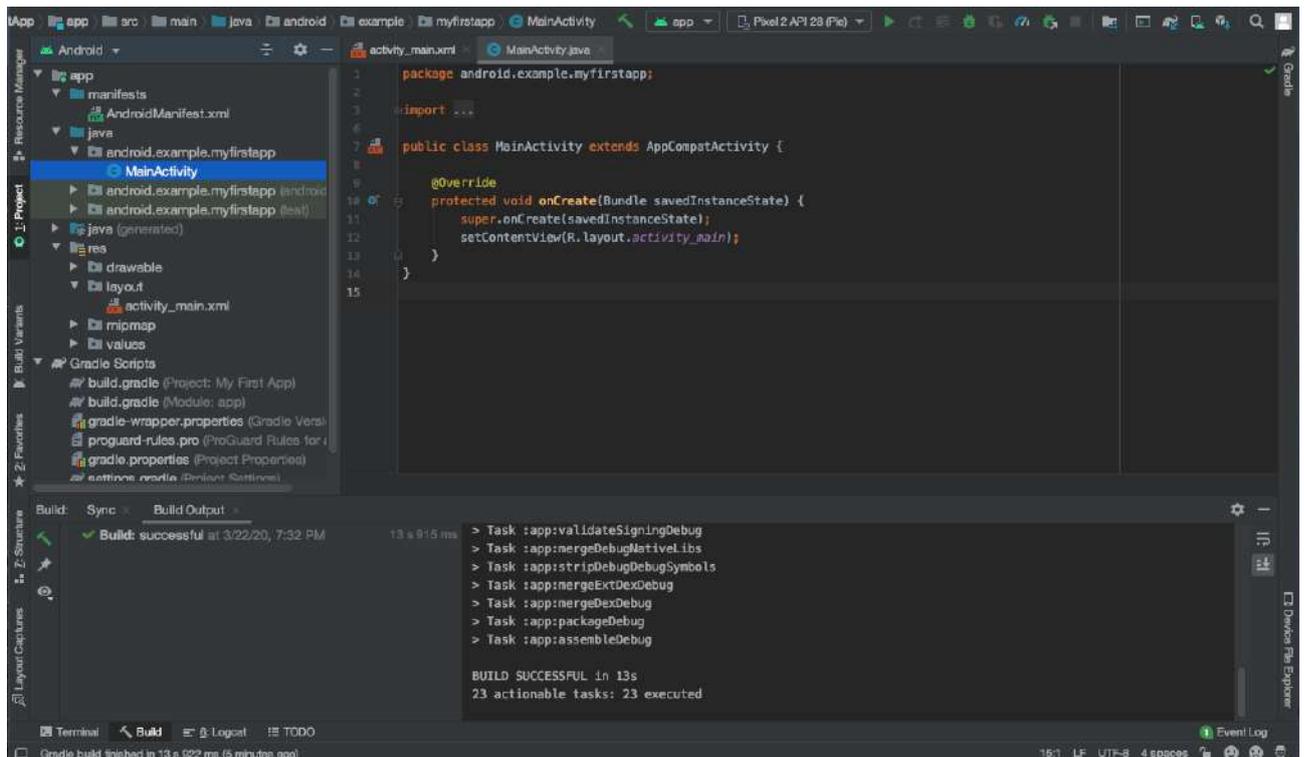


Рисунок 3.2 – Головне вікно зі створеним новим додатком

Усі додатки розбиті на розділи manifests, java і res:

- У manifests зберігається файл AndroidManifest.xml, який являє собою XML-файл, що описує структуру Android-додаток. У цей файл також записуються настройки дозволів (коли це може бути застосовано) та інші відомості про програму;

- У java зберігаються вихідні Java-файли програми у відповідності з ієрархією пакетів;
- У розділі res зберігаються файли ресурсів програми, які розбиті на піддиректорії drawable, layout, mipmap і values:
 - drawable: спочатку порожній підрозділ, в якому можна зберігати графічні ресурси програми;
 - layout: містить файли макета додатки; спочатку тут зберігається файл макета основного напрямку main.xml;
 - mipmap: містить файли ic_launcher.png, в яких зберігаються іконки для запуску програми з екрану в різних дозволах;
 - values: містить файли colors.xml, dimens.xml, strings.xml і styles.xml.

У розділі налаштувань Android Studio «Gradle Scripts» визначено різні файли .gradle і .properties, які використовуються системою збирання на основі Gradle [64]. Кожне відгалуження / розділ відповідає імені каталогу або імені файлу. Наприклад, res відповідає каталогу res, а strings.xml відповідає файлу strings.xml.

В Android передбачена вбудована підтримка однієї з найпоширеніших систем управління базами даних — SQLite. Для роботи з нею у пакеті android.database.sqlite реалізовано набір класів, які дозволяють створювати та використовувати власні БД у межах кожного додатку.

Робота з базою даних

Щоб інтегрувати SQLite в Android-додаток, спочатку створюється база даних за допомогою SQL-запитів. Збереження здійснюється у каталозі програми за шляхом:

DATA/data/[newwarehouse]/databases/[warehouseDB]

Операційна система Android вже містить ряд системних БД SQLite, які використовуються стандартними додатками, наприклад для контактів, галереї чи музики [65, 66].

Основний функціонал для роботи з БД забезпечує пакет `android.database` [68], тоді як конкретні можливості для SQLite реалізовані в `android.database.sqlite`.

Ключові класи

`SQLiteDatabase` — головний клас для виконання запитів та маніпуляцій із даними.

`SQLiteCursor` — дозволяє обробляти результати запитів і повертати набори рядків.

`SQLiteQueryBuilder` — забезпечує зручне формування SQL-запитів.

`SQLiteStatement` — використовується для створення SQL-виразів із плейсхолдерами для підстановки динамічних даних.

`SQLiteOpenHelper` — допоміжний клас для створення нової бази або оновлення існуючої.

Типи даних у SQLite

`INTEGER` — цілі числа (аналог `int` у Java);

`REAL` — числа з плаваючою точкою (`float`, `double`);

`TEXT` — рядки символів (`String`, `char`);

`BLOB` — бінарні дані (наприклад, зображення).

Дані у SQLite мають відповідати доступним типам у Java [69].

Приклад використання

Для роботи з БД у проєкті застосовуються два класи:

`DBHelper` (успадковує `SQLiteOpenHelper`) — у конструкторі викликається конструктор батьківського класу із зазначенням імені та версії БД. Метод `getWritableDatabase()` відкриває підключення та повертає об'єкт `SQLiteDatabase`, а метод `close()` закриває його. У разі відсутності чи застарілості БД потрібно реалізувати методи `onCreate` та `onUpgrade`.

`SQLiteDatabase` — надає методи для виконання операцій над даними: вставки, оновлення, видалення, читання.

Створення бази даних `warehouseDB`:

```

SQLiteDatabase db =
getBaseContext().openOrCreateDatabase("warehouseDB", MODE_PRIVATE, null);

```

Створення таблиці users:

```

SQLiteDatabase db =
getBaseContext().openOrCreateDatabase("warehouseDB", MODE_PRIVATE, null);
db.execSQL("CREATE TABLE IF NOT EXISTS users (name TEXT, tel
INTEGER, card INTEGER)");

```

Виконання запиту з поверненням даних:

Для вибірки даних застосовується метод `rawQuery()`, який приймає SQL-вираз і набір параметрів. Наприклад, для отримання всіх записів із таблиці:

```
Cursor query = db.rawQuery("SELECT * FROM users", null);
```

Якщо протрібно не просто виконати вираз, але і отримати з БД будь-які дані, то використовується метод `rawQuery ()`. Цей метод в якості параметра приймає SQL-вираз, а також набір значень для вираження SQL. Наприклад, отримання всіх об'єктів з бази даних:

```

SQLiteDatabase db = getBaseContext().openOrCreateDatabase("warehouseDB",
1 MODE_PRIVATE, null);
2 db.execSQL("CREATE TABLE IF NOT EXISTS users (name TEXT, tel INTEGER, card
3 INTEGER)");
4 Cursor query = db.rawQuery("SELECT * FROM users;", null);
5 if(query.moveToFirst()){
6
7     String name = query.getString(0);
8     int age = query.getInt(1);
}

```

Метод `db.rawQuery()` повертає об'єкт `Cursor`, за допомогою якого можна витягти отримані дані.

Можлива ситуація, коли в базі даних не буде об'єктів, і для цього методом `query.moveToFirst()` намагаємося переміститися до першого об'єкту, отриманого з БД. Якщо цей метод поверне значення `false`, значить запит не отримав ніяких даних з БД.

Перевіримо правильність з'єднання з базою даних та запису даних. Для цього у файлі `activity_main.xml` створимо примітивний інтерфейс.

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
3     android:id="@+id/activity_main"
4     android:layout_width="match_parent"
5     android:layout_height="match_parent"
6     android:padding="16dp"
7     android:orientation="vertical">
8     <Button
9         android:layout_width="wrap_content"
10        android:layout_height="wrap_content"
11        android:text="Click"
12        android:onClick="onClick"/>
13     <TextView
14        android:id="@+id/textView"
15        android:layout_width="wrap_content"
16        android:layout_height="wrap_content"
17        android:textSize="18sp" />
18 </LinearLayout>
```

А в класі MainActivity визначимо взаємодія з базою даних:

```
1 package com.example.eugene.sqliteapp;
2
3 import android.database.Cursor;
4 import android.database.sqlite.SQLiteDatabase;
5 import android.support.v7.app.AppCompatActivity;
6 import android.os.Bundle;
7 import android.view.View;
8 import android.widget.TextView;
9
10 public class MainActivity extends AppCompatActivity {
11
12     @Override
```

```

13     protected void onCreate(Bundle savedInstanceState) {
14         super.onCreate(savedInstanceState);
15         setContentView(R.layout.activity_main);
16     }
17
18     public void onClick(View view){
19         SQLiteDatabase db =
20     getBaseContext().openOrCreateDatabase("warehouseDB", MODE_PRIVATE, null);
21         db.execSQL("CREATE TABLE IF NOT EXISTS users (name TEXT, tel INTEGER,
22     card INTEGER)");
23         db.execSQL("INSERT INTO users VALUES ('Tom Smith', 0632387406,
24     8585478)");
25         db.execSQL("INSERT INTO users VALUES ('John Dow', 0675498031,
26     0868654)");
27
28         Cursor query = db.rawQuery("SELECT * FROM users;", null);
29         TextView textView = (TextView) findViewById(R.id.textView);
30         if(query.moveToFirst()){
31             do{
32                 String name = query.getString(0);
33                 int age = query.getInt(1);
34                 textView.append("Name: " + name + " Telephone: " + tel + "
35     Card Id: " + card + "\n");
36             }
37             while(query.moveToNext());
38         }
39         query.close();
40         db.close();
41     }
42 }

```

При натисканні на кнопку спочатку створюється в базі даних warehouseDB нова таблиця users, а потім в неї додаються два об'єкти в базу даних за допомогою SQL-вирази INSERT.

Далі за допомогою виразу SELECT отримуємо всіх доданих користувачів з бази даних у вигляді курсору Cursor [70].

Викликом `query.moveToFirst()` переміщаємося в курсорі до першого об'єкту, і так як у нас може бути більше одного об'єкта, то проходимо по всьому курсору в циклі `do ... while`.

Для отримання даних з курсору застосовуються методи `query.getString(0)` і `query.getInt(1)`. У дужках в методи передається номер стовпчика, з якого ми отримуємо дані.

Наприклад, вище ми додали спочатку ім'я користувача у вигляді рядка, а потім телефон та номер картки у вигляді числа. Значить, нульовим стовпцем буде йти рядку значення, яке отримуємо за допомогою методу `getString()`, а наступним - першим стовпцем йде числове значення, для якого застосовується метод `getInt()`.

Після завершення роботи з курсором і базою даних закриваємо всі пов'язані об'єкти:

```
1 query.close();  
2 db.close();
```

Якщо не закриємо курсор, то можемо зіткнутися з проблемою витоку пам'яті.

3.2 Перевірка функціонування мобільного додатку автоматизованої системи обліку складських товарів та підтримки торгівлі

Розглянемо можливості створеної автоматизованої системи, що спрощує процес управління складом з використанням мобільного пристрою чи планшета. Головний екран розробленого мобільного додатку зображено на рисунку 3.3.

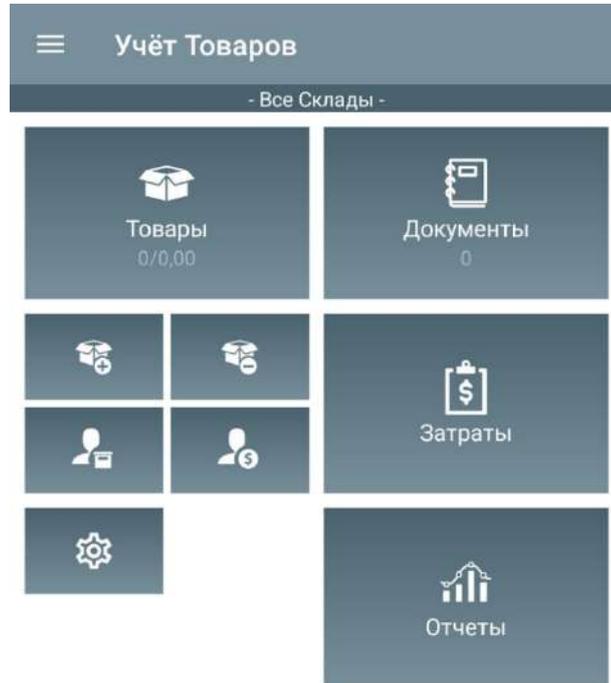


Рисунок 3.3 – Головний екран мобільного додатку автоматизованої системи обліку складських товарів та підтримки торгівлі

У розділі "Товари" міститься список всіх товарів компанії із зазначенням кількості на складі.

Цей розділ використовується для вибору товарів при додаванні в документи. Якщо для товару вказано штрих-код, то стає можливим швидко додавати товар в документ шляхом зчитування штрих-коду за допомогою камери. Кількість на складі не редагується, а формується автоматично на підставі документів.

Товари можна додавати вручну або імпортувати з файлу Excel.

Для імпорту з файлу в ньому повинні міститися 2 стовбці - з найменуванням та штрих-кодом. За замовчуванням - це стовпці А і В.

При додаванні товару вручну можна зчитати штрих-код товару з упаковки за допомогою камери, натиснувши на кнопку поруч з полем штрих-коду.

Для видалення товару зі списку потрібно використати довге натискання на потрібному рядку або жест "свайп".

Для пошуку і сортування товарів використовуються кнопки у верхній частині екрана.

Можна почати роботу не заповнюючи довідник товарів, а просто сформуванати прибутковий документ і відсутні товари будуть додані в довідник автоматично.

Почнемо заповнення бази даних мобільного додатку із створення груп товарів. Результат дії зображено на рисунку 3.4.

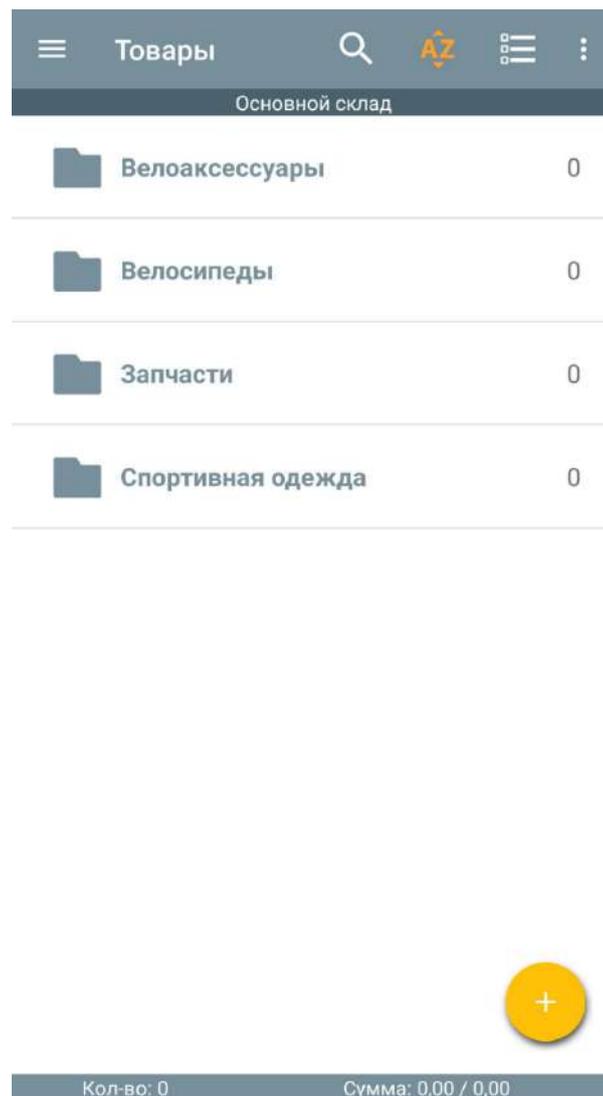


Рисунок 3.4 – Створені групи товарів

Створимо прибутковий документ, а разом із ним заповнимо деякі картки товарів, дія зображена на рисунку 3.5.

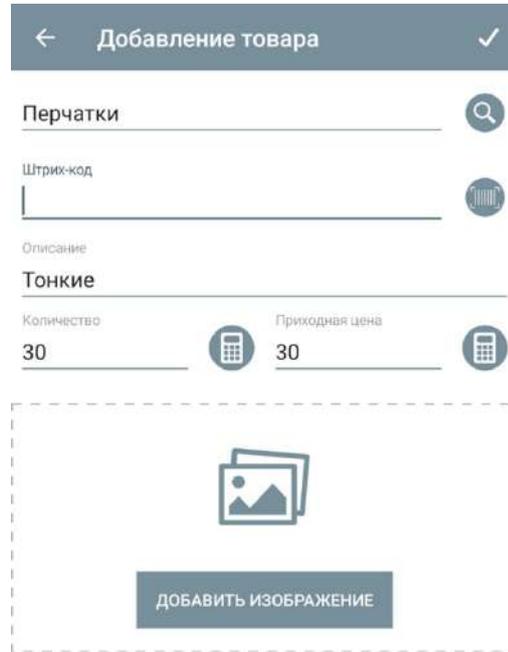


Рисунок 3.5 – Заповнення картки товару

Додавання товарів та їх відображення на екрані прибуткових документів зображено на рисунках 3.6 та 3.7 відповідно.

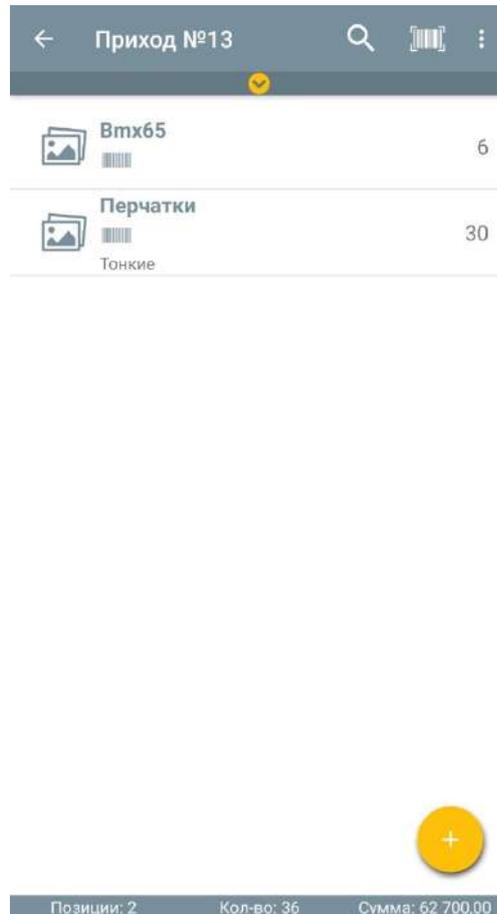


Рисунок 3.6 – Додавання товарів в прибутковий документ

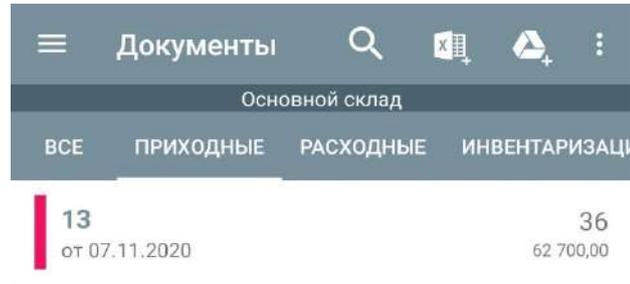


Рисунок 3.7 – Відображення створеного документу

Прийняті товари згідно прибуткового документу починають відображатися у списку товарів складу, це зображено на рисунку 3.8.

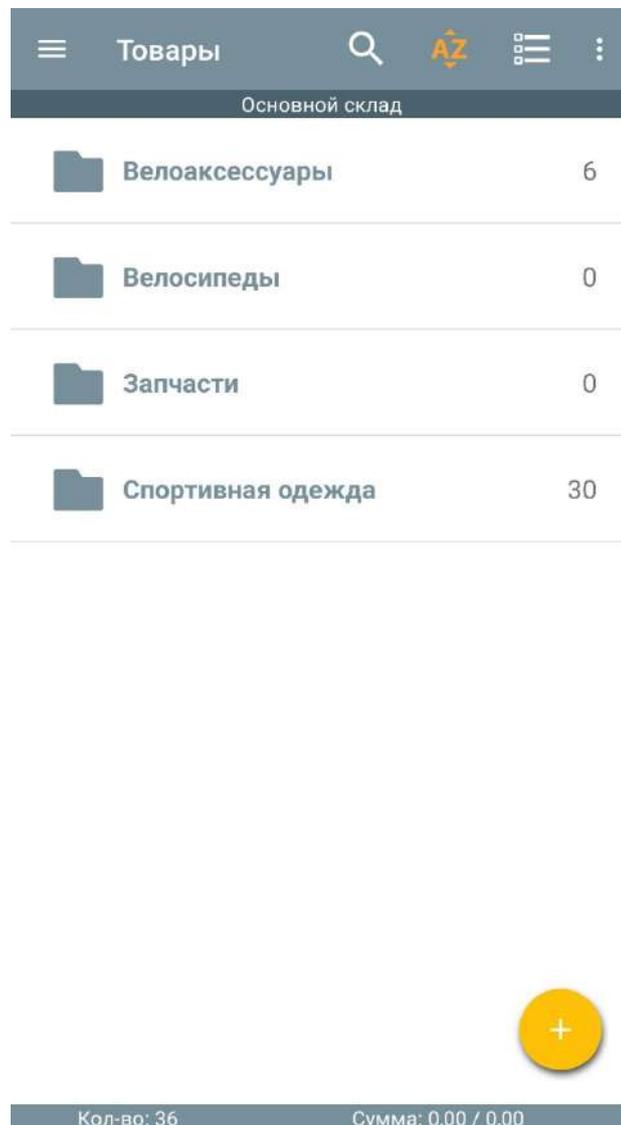


Рисунок 3.8 – Перегляд залишків товарів після додавання товару за прибутковим документом

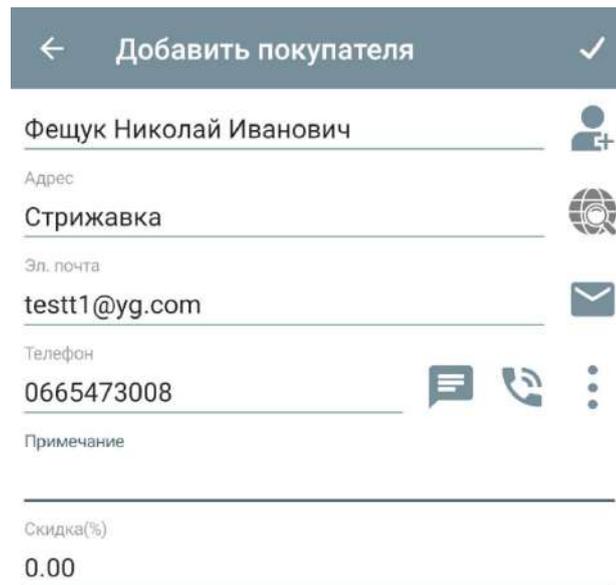
Мобільний додаток розробленої автоматизованої системи дозволяє вести облік Постачальників і Покупців. Для цього потрібно додати їх до відповідних

довідників.

Програма дозволяє імпортувати дані Постачальників і Покупців зі списку Контактів мобільного пристрою.

Довге натиснення на імені контрагента в списку або "свайп" вліво дозволяє переглянути всі транзакції даного контрагента. Крім цього в розділі Документи доступні Звіти по Постачальникам і Покупцям.

На рисунку 3.9 зображено процес додавання даних про нового покупця та на рисунку 3.10 зображено список існуючих покупців.



The screenshot shows a mobile application interface for adding a buyer. At the top, there is a dark blue header with a back arrow, the text "Добавить покупателя" (Add buyer), and a checkmark icon. Below the header, the form contains several fields with labels and values:

- Имя:** Фещук Николай Иванович (Name: Feshuk Nikolai Ivanovich) with a person icon and a plus sign.
- Адрес:** Стрижавка (Address: Strizhivka) with a globe icon.
- Эл. почта:** testt1@yg.com (Email: testt1@yg.com) with an envelope icon.
- Телефон:** 0665473008 (Phone: 0665473008) with a speech bubble icon, a phone icon, and a vertical ellipsis icon.
- Примечание:** (Note) - This field is currently empty.
- Скидка(%):** 0.00 (Discount(%): 0.00)

Рисунок 3.9 – Додавання даних про нового покупця

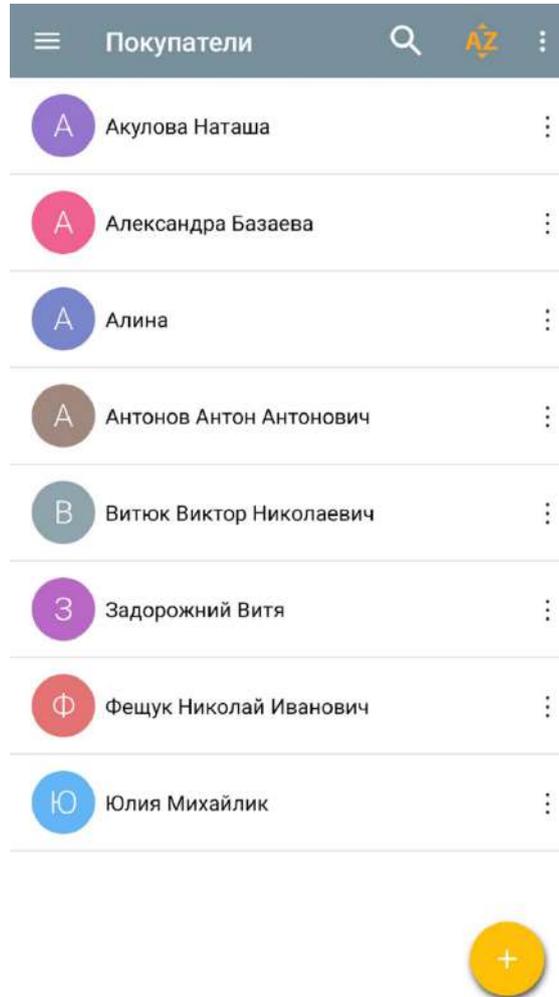


Рисунок 3.10 – Відображення списку покупців

Створення Постачальників та відображення списку із створеними контрагентами зображено на рисунках 3.11 та 3.12.

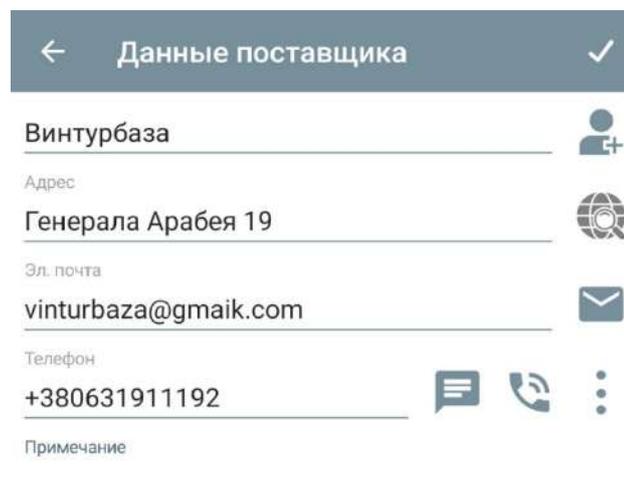


Рисунок 3.11 – Додавання даних про постачальника

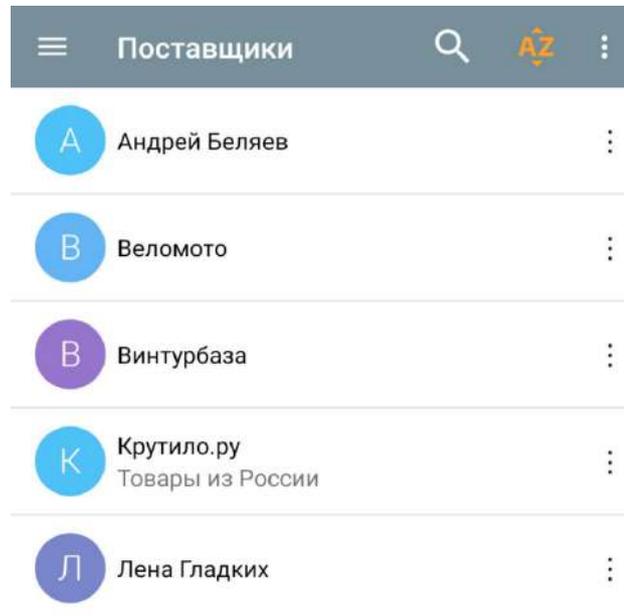


Рисунок 3.12 – Відображення списку постачальників

Основні дані вже занесені до бази даних автоматизованої системи, і тепер є можливість протестувати створення видаткового документу.

Екран створення нового видаткового документу зображено на рисунку 3.13.



Рисунок 3.13 – Створення нового видаткового документу

Заповнений видатковий документ зображено на рисунку 3.14.

The screenshot displays a receipt form titled 'Расход №17'. The form includes the following fields:

- Дата документа:** 09.11.2020
- Номер документа:** 17
- Покупатель:** Витюк Виктор Николаевич
- Скидка(%):** 0.00
- Примечание:** (empty field)

Below the form, there is a list of items:

Image	Item Name	Quantity
	Перчатки Тонкие	2
	Втх65	1

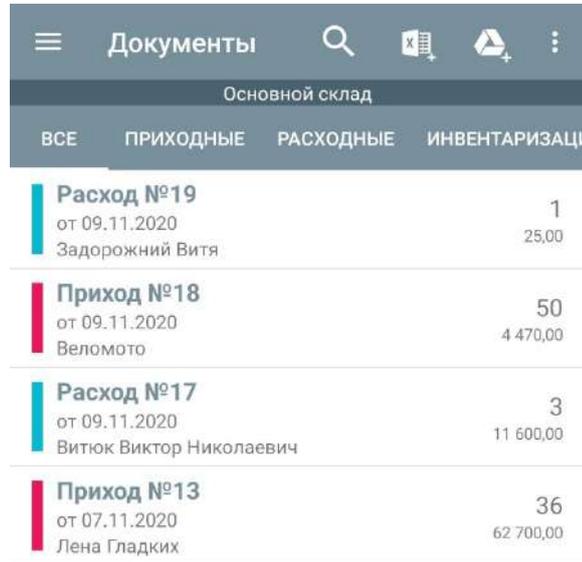
At the bottom of the screen, there is a summary bar with the following information:

- Позиции: 2
- Кол-во: 3
- Сумма: 11 600,00

A yellow circular button with a plus sign is located above the summary bar.

Рисунок 3.14 – Заповнений видатковий документ

Відображення списку прибуткових та видаткових документів зображено на рисунку 3.15.



Основной склад			
ВСЕ	ПРИХОДНЫЕ	РАСХОДНЫЕ	ИНВЕНТАРИЗАЦИЯ
Расход №19	от 09.11.2020	Задорожний Витя	1 25,00
Приход №18	от 09.11.2020	Веломото	50 4 470,00
Расход №17	от 09.11.2020	Витюк Виктор Николаевич	3 11 600,00
Приход №13	от 07.11.2020	Лена Гладких	36 62 700,00

Рисунок 3.15 – Відображення документів в системі

У мобільному додатку існує 3 типи документів:

- Прибутковий - оформляється при прийомі товару на склад. Кількість з документа додається до кількості в наявності на складі.
- Видатковий - оформляється при списанні / відвантаження товару зі складу. Кількість, вказана в документі, віднімається з кількості на складі.
- Інвентаризація - оформляється при проведенні інвентаризації або переобліку. Кількість, вказана в документі, заміщає кількість в наявності на складі.

При додаванні товару в документ можна вибрати товар з довідника товарів або ж просто зчитати його штрих-код.

Якщо в момент додавання товару в документ, такого товару не існувало в довіднику товарів, то він буде доданий в довідник товарів. Потрібно зауважити, що неможливо додати в видатковий документ товар, якого раніше не було в довіднику, так як неможливо списати позицію, кількість якої на складі дорівнює 0.

Для видалення документа або рядків документа можна скористатись контекстним меню, яке викликається довгим натисканням на потрібному рядку, або ж жестом "свайп".

При видаленні відбувається відповідна зміна кількості товарів на складі. Документ інвентаризації можна видалити тільки в тому випадку, якщо він є

останнім документом, так як при видаленні більш ранніх документів інвентаризації неможливо правильно підрахувати кількість в наявності на складі.

Зберегти документи в файл, відправити на e-mail або в Dropbox можна 2-ма способами: викликати контекстне меню довгим натисканням на потрібному рядку або ж перейти в режим позначки (вибору) документів. Для позначки документів натисніть на плаваючу кнопку "Плюс" і виберіть пункт "Вибрати документи" – тепер можна відзначити потрібні документи "галочками", а внизу з'являться кнопки, відповідні можливим діям з ними.

У верхньому меню знаходяться кнопки, що дозволяють відфільтрувати документи за датою і задати сортування списку.

На основі інформації із бази даних є можливість формувати звіти, повний перелік яких зображено на рисунку 3.16.

Отчеты	
Основной склад	
Доказ товаров	☆ ⋮
Журнал транзакций	☆ ⋮
Затраты	☆ ⋮
Оборот товара	☆ ⋮
По покупателям	☆ ⋮
По поставщикам	☆ ⋮
Прибыли и убытки	☆ ⋮
Продажи по дням	☆ ⋮
Продажи по месяцам	☆ ⋮
Продажи по товарам	☆ ⋮
Список документов	☆ ⋮

Рисунок 3.16 – Перелік звітів, що доступні для формування

На рисунках 3.17 – 3.21 відображено деякі звіти на основі створених дій в

системі.

← **Оборот товара** PDF X

За период с 01.11.2020 по 09.11.2020 по складу Основной склад

Наименование	Остаток на начало	Приход	Перемещен
Велоаксессуары			
Впх65	0	6	
Итого:	0	6	
Запчасти			
Трос	0	40	
Итого:	0	40	
Спортивная одежда			
Куртка	0	10	
Перчатки	0	30	
Итого:	0	40	
ВСЕГО:	0	86	

Рисунок 3.17 – Звіт по товарообороту за вказаний період

← **По покупателям** PDF X

За период с 01.11.2020 по 09.11.2020 по складу Основной склад

Наименование	Кол-во	Сумма	Себестоимо
Витюк Виктор Николаевич			
Впх65	1	11 500,00	0
Перчатки	2	100,00	0
Итого:	3	11 600,00	0
Задорожний Витя			
Трос	1	25,00	0
Итого:	1	25,00	0
ВСЕГО:	4	11 625,00	0

Рисунок 3.18 – Звіт з продажу відсортований за списком покупців

← **Журнал транзакций** PDF X

За период с 01.11.2020 по 09.11.2020 по складу Основной склад

Дата	№	Примечание	Тип	Конт
07.11.2020	13		Приход	Лена Гладки
07.11.2020	13		Приход	Лена Гладки
09.11.2020	18		Приход	Веломото
09.11.2020	18		Приход	Веломото
09.11.2020	17		Расход	Витюк Викто
09.11.2020	17		Расход	Витюк Викто
09.11.2020	19		Расход	Задорожний
ВСЕГО:				

Рисунок 3.19 – Журнал транзакцій

← По поставщикам		
За период с 01.11.2020 по 09.11.2020 по складу Основной склад		
Наименование	Кол-во	Сумма
Андрей Беляев		
Велопарковка нас два вел	20	0,00
Велосипед трюковый	10	0,00
Итого:	30	0,00
Веломото		
Куртка	10	3 990,00
Трос	40	480,00
Итого:	50	4 470,00
Винтурбаза		
Корзина	45	0,00
Итого:	45	0,00
Крутило.ру		
Велопарковка нас два вел	34	11 900,00
Корзина	20	4 000,00
Итого:	54	15 900,00
Лена Гладких		
Втх65	6	61 800,00
Перчатки	30	900,00
Итого:	36	62 700,00
ВСЕГО:	215	83 070,00

Рисунок 3.20 – Звіт з надходження товарів відсортований за списком постачальників

← Список документов				
За период с 01.11.2020 по 09.11.2020 по складу Основной склад				
Дата	№	Примечание	Тип	Конт
07.11.2020	13		Приход	Лена Гладких
09.11.2020	18		Приход	Веломото
09.11.2020	17		Расход	Витюк Викто
09.11.2020	19		Расход	Задорожный
ВСЕГО:				

Рисунок 3.21 – Список документів

В даному розділі є можливість вибору пріоритету конкретним звітам позначивши їх зіркою. Такі звіти будуть відображатись завжди на початку списку, що зображено на рисунку 3.22.

Отчеты		
Основной склад		
Журнал транзакций	★	⋮
Оборот товара	★	⋮
Продажи по дням	★	⋮
Дозаказ товаров	☆	⋮
Затраты	☆	⋮
По покупателям	☆	⋮
По поставщикам	☆	⋮
Прибыли и убытки	☆	⋮
Продажи по месяцам	☆	⋮
Продажи по товарам	☆	⋮
Список документов	☆	⋮

Рисунок 3.22 – Надання звітам пріоритету

На рисунку 3.23 зображено можливості з експорту звітів.

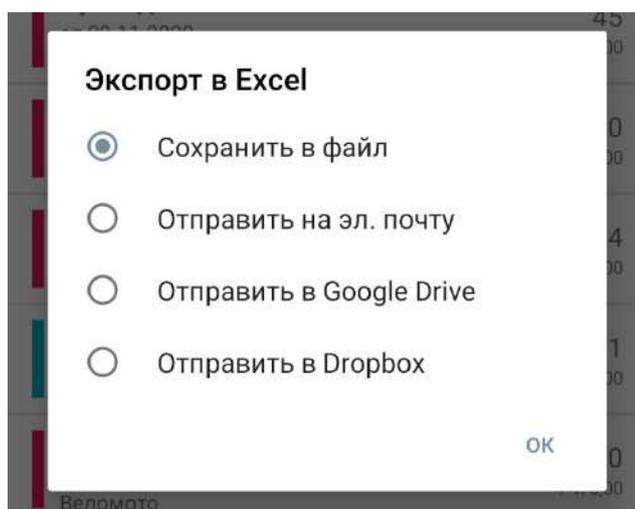


Рисунок 3.23 – Варіанти експорту звітів

На рисунку 3.24 зображено звіт, що збережений в документ Excel.

4	Дата	№	Примечание	Тип	Контрагент	Группа	Наименование	Описание	Штрих-код	Кол-во	Сумма прих.	Сумма расх.
5	07.11.20	13		Приход	Лена Гладких	Велоаксессуары	Впх85			6	61800	0
6	07.11.20	13		Приход	Лена Гладких	Спортивная одежда	Перчатки	Тонкие		30	900	0
7	09.11.20	18		Приход	Веломото	Запчасти	Трос		5784387	40	480	0
8	09.11.20	18		Приход	Веломото	Спортивная одежда	Куртка			10	3680	0
9	09.11.20	17		Расход	Виток Виктор Николаевич	Велоаксессуары	Впх85			1	0	11500
10	09.11.20	17		Расход	Виток Виктор Николаевич	Спортивная одежда	Перчатки	Тонкие		2	0	100
11	09.11.20	19		Расход	Задорожний Витя	Запчасти	Трос		5784387	1	0	25
12	ВСЕГО:									90	67170.00	11625.00

Рисунок 3.24 – Звіт, що збережений в документ Excel

На рисунку 3.25 зображено звіт, що збережений у pdf форматі.

Журнал транзакций
За период с 01.11.2020 по 09.11.2020 по складу Основной склад

Дата	№	Примечание	Тип	Контрагент	Группа	Наименование	Описание	Штрих-код	Кол-во	Сумма прих.	Сумма расх.
07.11.2020	13		Приход	Лена Гладких	Велоаксессуары	Вмх65			6	61800.00	0.00
07.11.2020	13		Приход	Лена Гладких	Спортивная одежда	Перчатки	Тонкие		30	900.00	0.00
09.11.2020	18		Приход	Веломото	Запчасти	Трос		5784367	40	480.00	0.00
09.11.2020	18		Приход	Веломото	Спортивная одежда	Куртка			10	3990.00	0.00
09.11.2020	17		Расход	Витюк Николаевич	Велоаксессуары	Вмх65			1	0.00	11500.00
09.11.2020	17		Расход	Витюк Николаевич	Спортивная одежда	Перчатки	Тонкие		2	0.00	100.00
09.11.2020	19		Расход	Задорожный Витя	Запчасти	Трос		5784367	1	0.00	25.00
ВС ЕГО:									90	67170.00	11625.00

Рисунок 3.25 – Звіт, що збережений у pdf форматі

Перевіривши правильність роботи програми, заповнення таблиць бази даних та виводу інформації із них на екрани програми, зроблено висновок про працездатність розробленого мобільного додатку автоматизованої системи обліку складських товарів та підтримки торгівлі.

Новизною та перевагою даного додатку автоматизованої системи є його універсальність та одночасно не обтяженість додатковими функціональними можливостями, що відволікають увагу та час користувачів при роботі з програмою.

Даний мобільний додаток автоматизованої системи обліку складських товарів та підтримки торгівлі дозволить автоматизувати рутинні завдання та мінімізувати вплив помилок людського фактору на роботу підприємства, задля зменшення втрат та ефективного розподілу вивільнених робочих ресурсів на інші області роботи компанії.

4 ЕКОНОМІЧНА ЧАСТИНА

4.1 Комерційний та технологічний аудит науково-технічної розробки

Комплексна магістерська кваліфікаційна робота з розробки та дослідження «Розробка автоматизованої системи обліку складських товарів і підтримки торговельної діяльності. Частина 2. Розробка мобільного застосунку» відноситься до науково-технічних робіт, які орієнтовані на виведення на ринок (або рішення про виведення науково-технічної розробки на ринок може бути прийнято у процесі проведення самої роботи), тобто коли відбувається так звана комерціалізація науково-технічної розробки. Цей напрямок є пріоритетним, оскільки результатами розробки можуть користуватися інші споживачі, отримуючи при цьому певний економічний ефект. Але для цього потрібно знайти потенційного інвестора, який би взявся за реалізацію цього проекту і переконати його в економічній доцільності такого кроку.

Для наведеного випадку нами мають бути виконані такі етапи робіт:

- 1) проведено комерційний аудит науково-технічної розробки, тобто встановлення її науково-технічного рівня та комерційного потенціалу;
- 2) розраховано витрати на здійснення науково-технічної розробки;
- 3) розрахована економічна ефективність науково-технічної розробки у випадку її впровадження і комерціалізації потенційним інвестором і проведено обґрунтування економічної доцільності комерціалізації потенційним інвестором.

Дана магістерська робота присвячена розробці прикладного інструменту для оптимізації процесів у сфері оптової та роздрібною торгівлі. Проектований мобільний застосунок має гнучку модель комерціалізації: він може функціонувати як автономний продукт (за моделлю підписки) або ж інтегруватися як мобільний терміналів у вже існуючі ERP-системи підприємств.

В якості аналогів на ринку можна розглядати наприклад Trini POS, орієнтовна вартість якого складає 350 доларів США на користувача на рік або приблизно 14 800 гривень на рік.

Для проведення комерційного та технологічного аудиту до оцінювання залучають не менше трьох незалежних експертів. Рекомендується оцінювати науково-технічний рівень розробки та її комерційний потенціал за п'ятибальною системою, застосовуючи 12 критерій у відповідності із табл. 4.1.

Таблиця 4.1 – Рекомендовані критерії оцінювання комерційного потенціалу розробки та їх можлива бальна оцінка

Бали (за 5-ти бальною шкалою)					
Критерій	0	1	2	3	4
Технічна здійсненність концепції					
1	Достовірність концепції не підтверджена	Концепція підтверджена експертними висновками	Концепція підтверджена розрахунками	Концепція перевірена на практиці	Перевірено роботоздатність продукту в реальних умовах
2	Багато аналогів на малому ринку	Мало аналогів на малому ринку	Кілька аналогів на великому ринку	Один аналог на великому ринку	Продукт не має аналогів на великому ринку
Ринкові переваги					
3	Ціна продукту значно вища за ціни аналогів	Ціна продукту дещо вища за ціни аналогів	Ціна продукту приблизно дорівнює цінам аналогів	Ціна продукту дещо нижче за ціни аналогів	Ціна продукту значно нижче за ціни аналогів
4	Технічні та споживчі властивості продукту значно гірші, ніж в аналогів	Технічні та споживчі властивості продукту трохи гірші, ніж в аналогів	Технічні та споживчі властивості продукту на рівні аналогів	Технічні та споживчі властивості продукту трохи кращі, ніж в аналогів	Технічні та споживчі властивості продукту значно кращі, ніж в аналогів
5	Експлуатаційні витрати значно вищі, ніж в аналогів	Експлуатаційні витрати дещо вищі, ніж в аналогів	Експлуатаційні витрати на рівні експлуатаційних витрат аналогів	Експлуатаційні витрати трохи нижчі, ніж в аналогів	Експлуатаційні витрати значно нижчі, ніж в аналогів
Ринкові перспективи					

6	Ринок малий і не має позитивної динаміки	Ринок малий, але має позитивну динаміку	Середній ринок з позитивною динамікою	Великий стабільний ринок	Великий ринок з позитивною динамікою
7	Активна конкуренція великих компаній	Активна конкуренція	Помірна конкуренція	Незначна конкуренція	Конкурентів немає
Практика на здійсненість					
8	Відсутні фахівці як з технічної, так і з комерційної реалізації ідеї	Необхідно наймати фахівців або витратити значні кошти та час на навчання наявних фахівців	Необхідне незначне навчання фахівців та збільшення їх штату	Необхідне незначне навчання фахівців	Є фахівці з питань як з технічної, так і з комерційної реалізації ідеї
9	Потрібні значні фінансові ресурси, які відсутні. Джерела фінансування ідеї відсутні	Потрібні незначні фінансові ресурси. Джерела фінансування відсутні	Потрібні значні фінансові ресурси. Джерела фінансування є	Потрібні незначні фінансові ресурси. Джерела фінансування є	Не потребує додаткового фінансування
10	Необхідна розробка нових матеріалів	Потрібні матеріали, що використовуються у військово-промисловому комплексі	Потрібні дорогі матеріали	Потрібні досяжні та дешеві матеріали	Всі матеріали для реалізації ідеї відомі та давно використовуються у виробництві
11	Термін реалізації ідеї більший за 10 років	Термін реалізації ідеї більший за 5 років. Термін окупності інвестицій більше 10-ти років	Термін реалізації ідеї від 3-х до 5-ти років. Термін окупності інвестицій більше 5-ти років	Термін реалізації ідеї менше 3-х років. Термін окупності інвестицій від 3-х до 5-ти років	Термін реалізації ідеї менше 3-х років. Термін окупності інвестицій менше 3-х років

12	Необхідна розробка регламентних документів та отримання великої кількості дозвільних документів на виробництво та реалізацію продукту	Необхідно отримання великої кількості дозвільних документів на виробництво та реалізацію продукту, що вимагає значних коштів та часу	Процедура отримання дозвільних документів для виробництва та реалізації продукту вимагає незначних коштів та часу	Необхідно тільки повідомлення відповідним органам про виробництво та реалізацію продукту	Відсутні будь-які регламентні обмеження на виробництво та реалізацію продукту
----	---	--	---	--	---

Усі дані по кожному параметру занесено в таблиці 4.2

Експертами було обрано Юхимчук М. С., Ковтун В. В. та Дубовой В.М., оскільки вони є висококваліфікованими викладачами ВНТУ та водночас керівниками магістерських кваліфікаційних робіт. Їхній фаховий досвід, наукові здобутки та практична компетентність у відповідних галузях забезпечують об'єктивну, професійну й обґрунтовану експертизу результатів дослідження.

Таблиця 4.2 – Результати оцінювання комерційного потенціалу розробки

Критерії оцінювання	ПІБ експертів		
	Юхимчук М. С.	Ковтун В. В.	Дубовой В.М.
	Бали		
Технічна здійсненність концепції	4	4	4
Наявність аналогів на ринку	3	3	4
Цінова політика	4	4	4
Технічні та споживчі властивості виробу	4	3	4
Експлуатаційні витрати	4	4	3
Ринок збуту	4	3	4
Конкурентоспроможність	3	4	3
Фахівці з технічної і комерційної реалізації	4	3	3
Фінансування	4	4	3
Матеріально-технічна база	3	3	3
Термін реалізації ідеї	4	4	3
Супровідна документація	4	3	3
Сума	45	42	41
Середньоарифметична сума балів	$(45+42+41) / 3 = 42,67$		

За даними таблиці 4.2 можна зробити висновок щодо рівня комерційного потенціалу даної розробки. Для цього доцільно скористатись рекомендаціями, наведеними в таблиці 4.3.

Таблиця 4.3 – Рекомендаціями рівня комерційного потенціалу даної

Середньоарифметична сума балів, розрахована на основі висновків експертів	Рівень комерційного потенціалу розробки
0-10	Низький
11-20	Нижче середнього
21-30	Середній
31-40	Вище середнього
41-48	Високий

Як свідчать результати оцінювання, комерційний потенціал нового програмного продукту є високим. Це зумовлено підвищенням рівня безпеки ІКС завдяки інтеграції адаптованих методів управління ризиками інформаційної безпеки. Такий підхід дозволяє визначати оптимальні стратегії оцінки ризиків для підприємств у межах функціонування розробленої комп'ютеризованої системи моніторингу безпеки об'єктів.

4.2 Прогнозування витрат на виконання науково-дослідної (дослідно-конструкторської) роботи.

Основна заробітна плата розробників, яка розраховується за формулою:

$$Z_o = \frac{M}{T_p} \cdot t \quad (4.1)$$

де M – місячний посадовий оклад конкретного розробника (дослідника), грн.;

T_p – число робочих днів за місяць, 23 днів;

t – число днів роботи розробника (дослідника).

Результати розрахунків зведемо до таблиці 4.4. Так як в даному випадку розробляється програмний продукт, то розробник виступає одночасно і основним розробником, і тестувальником розроблюваного програмного продукту.

Таблиця 4.4 — Основна заробітна плата розробників

Найменування посади	Місячний посадовий оклад, грн.	Оплата за робочий день, грн.	Число днів роботи	Витрати на заробітну плату, грн.
Керівник проекту	20 000,00	1 000,00	5	5 000,00
Програміст	35 000,00	1 750,00	25	43 750,00
UX / UI дизайнер	23 000,00	1 150,00	25	28 750,00
Системний адміністратор	23 000,00	1 150,00	25	28 750,00
Всього				106 250,00

Додаткова заробітна плата розробників, які брати участь в розробці обладнання/програмного продукту.

Додаткову заробітну плату прийнято розраховувати як 12 % від основної заробітної плати розробників та робітників:

$$Z_d = Z_o \cdot 12 \% / 100 \% \quad (4.2)$$

$$Z_d = (106250,00 \cdot 12 \% / 100 \%) = 12750,00 \text{ (грн.)}$$

Нарахування на заробітну плату розробників.

Згідно діючого законодавства нарахування на заробітну плату складають 22 % від суми основної та додаткової заробітної плати.

$$H_z = (Z_o + Z_d) \cdot 22 \% / 100\% \quad (4.3)$$

$$H_z = (106250,00 + 12750,00) \cdot 22 \% / 100 \% = 26180,00 \text{ (грн.)}$$

Оскільки для розроблювального пристрою не потрібно витратити матеріали та комплектуючі, то витрати на матеріали і комплектуючі дорівнюють нулю.

Амортизація обладнання, яке використовувалось для проведення розробки.

Амортизація обладнання, що використовувалось для розробки в спрощеному вигляді розраховується за формулою:

$$A = \frac{Ц}{T_{\text{в}}} \cdot \frac{t_{\text{вик}}}{12} \text{ [грн.]} \quad (4.4)$$

де Ц – балансова вартість обладнання, грн.;

T – термін корисного використання обладнання згідно податкового законодавства, років;

$t_{\text{вик}}$ – термін використання під час розробки, місяців.

Розрахуємо, для прикладу, амортизаційні витрати на ноутбук балансова вартість якого становить 22000 грн., термін його корисного використання згідно податкового законодавства – 2 роки, а термін його фактичного використання – 2 міс.

$$A_{\text{обл}} = \frac{22000}{2} \times \frac{2}{12} = 1833,33 \text{ грн.}$$

Аналогічно визначаємо амортизаційні витрати на інше обладнання та приміщення. Розрахунки заносимо до таблиці 4.5. Так як вартість ліцензійної операційної системи та спеціалізованих ліцензійних нематеріальних ресурсів є меншою за 20 000 грн, такий нематеріальний актив не підлягає амортизації, $B_{\text{нем.ак.}} = 6100 \text{ грн.}$

Таблиця 4.5 – Амортизаційні відрахування на матеріальні та нематеріальні ресурси для розробників

Найменування обладнання	Балансова вартість, грн.	Строк корисного використання, років	Термін використання обладнання, місяців	Амортизаційні відрахування, грн.
Комп'ютер та комп'ютерна периферія	22 000	2	2	1 833,33
Офісне обладнання (меблі)	25 000	4	2	1 041,67
Приміщення	1 600 000	20	2	13 333,33
Всього				16 208,33

Тарифи на електроенергію для побутових споживачів (промислових підприємств) відрізняються від тарифів на електроенергію для населення. При цьому тарифи на розподіл електроенергії у різних постачальників (енергорозподільних компаній), будуть різними. Крім того, розмір тарифу залежить від класу напруги (1-й або 2-й клас). Тарифи на розподіл електроенергії для всіх енергорозподільних компаній встановлює Національна комісія з регулювання енергетики і комунальних послуг (НКРЕКП). Витрати на силову електроенергію розраховуються за формулою:

$$V_e = V \cdot P \cdot \Phi \cdot K_n, \quad (4.5)$$

де V – вартість 1 кВт-години електроенергії для 1 класу підприємства з ПДВ в 2025 році для Вінницької області за даними Енера-Вінниця, $V = 12,69$ грн./кВт;

P – встановлена потужність обладнання, кВт. $P = 0,3$ кВт;

Φ – фактична кількість годин роботи обладнання, годин;

K_n – коефіцієнт використання потужності, $K_n = 0,9$;

$$V_e = 0,9 \cdot 0,3 \cdot 8 \cdot 50 \cdot 12,69 = 1370,52 \text{ (грн.)}$$

Інші витрати та загальновиробничі витрати.

До статті «Інші витрати» належать витрати, які не знайшли відображення у зазначених статтях витрат і можуть бути віднесені безпосередньо на собівартість досліджень за прямими ознаками. Витрати за статтею «Інші

витрати» розраховуються як 50...100% від суми основної заробітної плати дослідників:

$$I_e = (Z_o + Z_p) \cdot \frac{H_{iv}}{100\%}, \quad (4.6)$$

де H_{iv} – норма нарахування за статтею «Інші витрати».

$$I_e = 106250,00 * 78\% / 100\% = 82875,00 \text{ (грн.)}$$

До статті «Накладні (загальновиробничі) витрати» належать: витрати, пов'язані з управлінням організацією; витрати на винахідництво та раціоналізацію; витрати на підготовку (перепідготовку) та навчання кадрів; витрати, пов'язані з набором робочої сили; витрати на оплату послуг банків; витрати, пов'язані з освоєнням виробництва продукції; витрати на науково-технічну інформацію та рекламу та ін. Витрати за статтею «Накладні (загальновиробничі) витрати» розраховуються як 100...150% від суми основної заробітної плати дослідників:

$$H_{нзв} = (Z_o + Z_p) \cdot \frac{H_{нзв}}{100\%}, \quad (4.7)$$

де $H_{нзв}$ – норма нарахування за статтею «Накладні (загальновиробничі) витрати».

$$H_{нзв} = 106250,00 * 135\% / 100\% = 143437,50 \text{ (грн.)}$$

Витрати на проведення науково-дослідної роботи.

Сума всіх попередніх статей витрат дає загальні витрати на проведення науково-дослідної роботи:

$$B_{заг} = 106250,00 + 12750,00 + 26180,00 + 16208,33 + 6100,00 + 1370,52$$

$$+82875,00+143437,50 = 395171,35 \text{ грн.}$$

Розрахунок загальних витрат на науково-дослідну (науково-технічну) роботу та оформлення її результатів.

Загальні витрати на завершення науково-дослідної (науково-технічної) роботи та оформлення її результатів розраховуються за формулою:

$$ЗВ = \frac{B_{заг}}{\eta} \quad (\text{грн}), \quad (5.8)$$

де η – коефіцієнт, який характеризує етап (стадію) виконання науково-дослідної роботи.

Так, якщо науково-технічна розробка знаходиться на стадії: науково-дослідних робіт, то $\eta=0,1$; технічного проектування, то $\eta=0,2$; розробки конструкторської документації, то $\eta=0,3$; розробки технологій, то $\eta=0,4$; розробки дослідного зразка, то $\eta=0,5$; розробки промислового зразка, то $\eta=0,7$; впровадження, то $\eta=0,9$. Оберемо $\eta = 0,5$, так як розробка, на даний момент, знаходиться на стадії дослідного зразка:

$$ЗВ = 395171,35 / 0,5 = 790342,70 \text{ грн.}$$

4.3 Розрахунок економічної ефективності науково-технічної розробки за її можливої комерціалізації потенційним інвестором

В ринкових умовах узагальнювальним позитивним результатом, що його може отримати потенційний інвестор від можливого впровадження результатів цієї чи іншої науково-технічної розробки, є збільшення у потенційного інвестора величини чистого прибутку. Саме зростання чистого прибутку забезпечить потенційному інвестору надходження додаткових коштів, дозволить покращити фінансові результати його діяльності, підвищить конкурентоспроможність та може позитивно вплинути на ухвалення рішення щодо комерціалізації цієї розробки.

Для того, щоб розрахувати можливе зростання чистого прибутку у потенційного інвестора від можливого впровадження науково-технічної розробки необхідно:

а) вказати, з якого часу можуть бути впроваджені результати науково-технічної розробки;

б) зазначити, протягом скількох років після впровадження цієї науково-технічної розробки очікуються основні позитивні результати для потенційного інвестора (наприклад, протягом 3-х років після її впровадження);

в) кількісно оцінити величину існуючого та майбутнього попиту на цю або аналогічні чи подібні науково-технічні розробки та назвати основних суб'єктів (зацікавлених осіб) цього попиту;

г) визначити ціну реалізації на ринку науково-технічних розробок з аналогічними чи подібними функціями.

При розрахунку економічної ефективності потрібно обов'язково враховувати зміну вартості грошей у часі, оскільки від вкладення інвестицій до отримання прибутку минає чимало часу. При оцінюванні ефективності інноваційних проектів передбачається розрахунок таких важливих показників:

— абсолютного економічного ефекту (чистого дисконтованого доходу);

— внутрішньої економічної дохідності (внутрішньої норми дохідності);

— терміну окупності (дисконтованого терміну окупності).

Аналізуючи напрямки проведення науково-технічних розробок, розрахунок економічної ефективності науково-технічної розробки за її можливої комерціалізації потенційним інвестором можна об'єднати, враховуючи визначені ситуації з відповідними умовами.

Розробка чи суттєве вдосконалення програмного засобу (програмного забезпечення, програмного продукту) для використання масовим споживачем.

В цьому випадку майбутній економічний ефект буде формуватися на основі таких даних:

$$\Delta\Pi_i = (\pm\Delta\Pi_0 \cdot N + \Pi_0 \cdot \Delta N)_i \cdot \lambda \cdot \rho \cdot \left(1 - \frac{\vartheta}{100}\right), \quad (4.9)$$

де $\pm\Delta\Pi_0$ – зміна вартості програмного продукту (зростання чи зниження) від впровадження результатів науково-технічної розробки в аналізовані періоди часу;

N – кількість споживачів які використовували аналогічний продукт у році до впровадження результатів нової науково-технічної розробки;

Π_0 – основний оціночний показник, який визначає діяльність підприємства у даному році після впровадження результатів наукової розробки, $\Pi_0 = \Pi_0 \pm \Delta\Pi_0$;

Π_0 – вартість програмного продукту у році до впровадження результатів розробки;

ΔN – збільшення кількості споживачів продукту, в аналізовані періоди часу, від покращення його певних характеристик;

λ – коефіцієнт, який враховує сплату податку на додану вартість. Ставка податку на додану вартість дорівнює 20%, а коефіцієнт $\lambda = 0,8333$.

ρ – коефіцієнт, який враховує рентабельність продукту;

ϑ – ставка податку на прибуток, у 2025 році $\vartheta = 18\%$.

Припустимо, що при прогнозованій ціні 5600 грн. за одиницю виробу, термін збільшення прибутку складе 3 роки. Після завершення розробки і її вдосконалення, можна буде підняти її ціну на 400 грн. Кількість одиниць реалізованої продукції також збільшиться: протягом першого року – на 13000 шт., протягом другого року – на 12000 шт., протягом третього року на 11000 шт. До моменту впровадження результатів наукової розробки реалізації продукту не було:

$$\Delta\Pi_1 = (0 \cdot 5500 + (5500 + 500) \cdot 13000) \cdot 0,8333 \cdot 0,26 \cdot (1 - 0,18) = 1385400 \text{ грн.}$$

$$\Delta\Pi_2 = (0 \cdot 5500 + (5500 + 500) \cdot (13000 + 12000)) \cdot 0,8333 \cdot 0,26 \cdot (1 - 0,18) = 26601000 \text{ грн}$$

$$\Delta\Pi_3 = (0 \cdot 5500 + (5500 + 500) \cdot (13000 + 12000 + 11000)) \cdot 0,8333 \cdot 0,26 \cdot (1 - 0,18) = 38257000 \text{ грн.}$$

Отже, комерційний ефект від реалізації результатів розробки за три роки складе 78712000, 00 грн.

4.4 Розрахунок ефективності вкладених інвестицій та періоду їх окупності.

Розраховуємо приведену вартість збільшення всіх чистих прибутків $ПП$, що їх може отримати потенційний інвестор від можливого впровадження та комерціалізації науково-технічної розробки:

$$ПП = \sum_1^T \frac{\Delta\Pi_i}{(1+\tau)^t}, \quad (4.10)$$

де $\Delta\Pi_i$ – збільшення чистого прибутку у кожному із років, протягом яких виявляються результати виконаної та впровадженої науково-дослідної (науково-технічної) роботи, грн;

T – період часу, протягом якою виявляються результати впровадженої науково-дослідної (науково-технічної) роботи, роки;

τ – ставка дисконтування, за яку можна взяти щорічний прогнозований рівень інфляції в країні, $\tau = 0,05 \dots 0,15$;

t – період часу (в роках).

Збільшення прибутку ми отримаємо, починаючи з першого року:

$$ПП = (1385400,00/(1+0,1)^1) + (26601000,00/(1+0,1)^2) + (38257000,00/(1+0,1)^3) = 1259454,55 + 21983471,07 + 28744850,49 = 49\,999\,878,90 \text{ грн.}$$

Далі розраховують величину початкових інвестицій PV , які потенційний інвестор має вкласти для впровадження і комерціалізації науково-технічної розробки. Для цього можна використати формулу:

$$PV = k_{инв} \times ЗВ, \quad (4.11)$$

де $k_{инв}$ – коефіцієнт, що враховує витрати інвестора на впровадження науково-технічної розробки та її комерціалізацію, це можуть бути витрати на підготовку приміщень, розробку технологій, навчання персоналу, маркетингові заходи тощо; зазвичай $k_{инв} = 2 \dots 5$, але може бути і більшим;

ZB – загальні витрати на проведення науково-технічної розробки та оформлення її результатів, грн.

$$PV = 3 \cdot 790342,70 = 2\,371\,028,10 \text{ грн.}$$

Тоді абсолютний економічний ефект E_{abc} або чистий приведений дохід (NPV , *Net Present Value*) для потенційного інвестора від можливого впровадження та комерціалізації науково-технічної розробки становитиме:

$$E_{abc} = III - PV, \quad (4.12)$$

$$E_{abc} = 49\,999\,878,90 - 2\,371\,028,10 = 47\,628\,850\,748,8 \text{ грн}$$

Оскільки $E_{abc} > 0$ то вкладання коштів на виконання та впровадження результатів даної науково-дослідної (науково-технічної) роботи може бути доцільним.

Для остаточного прийняття рішення з цього питання необхідно розрахувати внутрішню економічну дохідність або показник внутрішньої норми дохідності (IRR , *Internal Rate of Return*) вкладених інвестицій та порівняти її з так званою бар'єрною ставкою дисконтування, яка визначає ту мінімальну внутрішню економічну дохідність, нижче якої інвестиції в будь-яку науково-технічну розробку вкладати буде економічно недоцільно.

Розрахуємо відносну (щорічну) ефективність вкладених в наукову розробку інвестицій E_{ϵ} . Для цього використаємо формулу:

$$E_{\epsilon} = \sqrt[T_{жс}]{1 + \frac{E_{abc}}{PV}} - 1, \quad (4.13)$$

$T_{жс}$ – життєвий цикл наукової розробки, роки.

$$E_{\epsilon} = \sqrt[3]{1 + 47\,628\,850\,748,8 / 2\,371\,028,10} - 1 = 1,64$$

Визначимо мінімальну ставку дисконтування, яка у загальному вигляді визначається за формулою:

$$\tau = d + f, \quad (4.14)$$

де d – середньозважена ставка за депозитними операціями в комерційних банках; в 2025 році в Україні $d = (0,09...0,15)$;

f – показник, що характеризує ризикованість вкладень; зазвичай, величина $f = (0,05...0,5)$.

$$\tau_{\min} = 0,14 + 0,05 = 0,19$$

Так як $E_v > \tau_{\min}$, то інвестор може бути зацікавлений у фінансуванні даної наукової розробки.

Розрахуємо термін окупності вкладених у реалізацію наукового проекту інвестицій за формулою:

$$T_{ок} = \frac{1}{E_v}, \quad (4.15)$$

$$T_{ок} = 1 / 1,64 = 0,61 \text{ р.}$$

Оскільки $T_{ок} < 3$ -х років, а саме термін окупності рівний 0,61 роки, то фінансування даної наукової розробки є доцільним.

4.5 Висновки до розділу

Економічна частина даної роботи містить розрахунок витрат на розробку нового програмного продукту, сума яких складає 790 342,70 гривень. Було спрогнозовано орієнтовану величину витрат по кожній з статей витрат. Також розраховано чистий прибуток, який може отримати виробник від реалізації нового технічного рішення, розраховано період окупності витрат для інвестора та економічний ефект при використанні даної розробки. В результаті аналізу розрахунків можна зробити висновок, що розроблений програмний продукт за ціною дешевший за аналог і є висококонкурентоспроможним. Період окупності складе близько 0,61 роки.

ВИСНОВКИ

У ході дослідження підтверджено, що автоматизація бізнес-процесів є ключовим чинником підвищення ефективності підприємств будь-якого масштабу. Застосування систем обліку і спеціалізованого програмного забезпечення дозволяє не лише скоротити витрати часу на рутинні операції, а й зменшити вплив людського фактора, підвищуючи точність ведення обліку та оперативність прийняття рішень. Аналіз існуючих проблем у сфері складського обліку, таких як накопичення нереалізованих товарів, неточності інвентаризації та затримки в закупівлях, показав, що без автоматизації ефективне управління торговельною діяльністю є складним, особливо при масштабуванні бізнесу.

Виконане дослідження дозволило визначити ключові функції мобільного додатку для автоматизованого обліку товарів, розробити структуру бази даних, яка забезпечує збереження та обробку інформації у режимі реального часу, та створити програмний комплекс, що пройшов тестування і підтвердив правильність своєї роботи. Використання Agile-підходу та методів Scrum дозволило забезпечити гнучкість розробки та ефективне вирішення завдань на кожному етапі проекту.

Практична реалізація мобільного додатку демонструє його здатність оптимізувати управлінські процеси торговельної мережі, підвищити продуктивність персоналу та забезпечити контроль над товарами у всіх філіях підприємства. Результати роботи мають практичне значення для підприємств, що прагнуть автоматизувати складський облік і підтримку торговельної діяльності, а сам підхід до розробки може бути використаний у подальших дослідженнях і проектах у сфері інформаційних систем.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Гетьман О. О., Шаповал В. М. Економіка підприємства: навчальний посібник / О. О. Гетьман, В. М. Шаповал. — 2-ге вид. — Київ: Центр учбової літератури, 2010. — 488 с.
2. Бартольдї Дж. К., Хекман С. Warehouse & Distribution Science / John J. Bartholdi III, Steven T. Hackman. — Atlanta: Supply Chain and Logistics Institute, 2014. — 420 p. (electronic resource). URL: <https://www.warehouse-science.com/book/editions/wh-sci-0.98.1.pdf?>
3. Клеппман М. Designing Data-Intensive Applications: The Big Ideas Behind Reliable, Scalable, and Maintainable Systems / Martin Kleppmann. — O'Reilly Media, 2017. — 616 p. URL: <https://martin.kleppmann.com/2017/03/27/designing-data-intensive-applications.html>
4. Сілбершатц А., Кейт Г., Сундарас Дж. Database System Concepts / Abraham Silberschatz, Henry F. Korth, S. Sudarshan. — McGraw-Hill Education, 2019. — 1376 p. URL: <https://www.db-book.com>
5. Берtrand М. Logistics & Supply Chain Management: Strategy, Operation and Practice / Martin Christopher. — 5th ed. — Pearson, 2016. — 312 p. URL: <https://lib.hpu.edu.vn/handle/123456789/35774>
6. Barth, P. Mobile Warehouse Management: Modern WMS and Mobile Computing / Paul Barth. — Springer, 2018. — 210 p. URL:
7. Phillips W., Stewart N., Marsicano C. Android Programming: The Big Nerd Ranch Guide / Bill Phillips, Chris Stewart, Kristin Marsicano. — 4th ed. — Big Nerd Ranch Guides, 2020. — 432 p. URL: https://openlibrary.org/books/OL28701975M/Android_Programming?
8. Deitel H., Deitel P. Android™ for Programmers: An App-Driven Approach / Paul Deitel, Harvey Deitel. — 3rd ed. — Pearson, 2017. — 768 p. URL: <https://www.oreilly.com/library/view/androidtm-for-programmers/9780132160261/>
9. Ray R. Kotlin Programming: The Big Nerd Ranch Guide / Josh Skeen, David Greenhalgh. — Big Nerd Ranch Guides, 2018. — 320 p. URL:
10. Elements of Kotlin / Mark L. Murphy — 2021 — URL: <https://freecomputerbooks.com/Elements-of-Kotlin.html>
11. Дайновський Ю. А. 505 прийомів бізнесу / Ю. А. Дайновський. — Київ: Знання, 2008. — 256 с.
12. Василенко В. А. Теорія і практика розробки управлінських рішень: навч. посібник / В. А. Василенко. — Київ: Знання, 2002. — 224 с.

13. Кузнецова М. Г. Забезпечення захищеності інформаційних ресурсів у розподілених системах / М. Г. Кузнецова. — Київ: ІПРИ НАНУ, 2004. — 112 с.
14. Шиян А. А. Економічна кібернетика: вступ до моделювання соціальних і економічних систем / А. А. Шиян. — Львів: Магнолія, 2006. — 228 с.
15. Стрембіцький П., Юхимчук М., Лесько В., Перепелиця С. (2025). Централізований моніторинг інфраструктури з використанням системи THANOS: перспективи та виклики. Herald of Khmelnytskyi National University. Technical Sciences, 347(1),417-422. <https://doi.org/10.31891/2307-5732-2025-347-57>.
16. Перепелиця С., Юхимчук М., Лесько В. Моделювання кіберфізичних систем управління в умовах негативних зовнішніх факторів / С. Перепелиця, М. Юхимчук, В. Лесько // Measuring and Computing Devices in Technological Processes. — 2025. — № 1. — С. 7–12. — URL: <https://doi.org/10.31891/2219-9365-2025-81-125>.
17. Юхимчук М., С., Стрембіцький П.П., Перепелиця С.С. Використання віддаленого моніторингу для підвищення якості та ефективності у виробничому середовищі. Herald of Khmelnytskyi National University. Technical Sciences, 335(3(1), 330-334. — URL: <https://doi.org/10.31891/2307-5732-2024-335-3-44>
18. Коркошка О. В. Інформаційні системи та технології управління: навч. посібник / О. В. Коркошка. — Київ: Кондор, 2012. — 320 с.
19. Биков І. І. Інформаційні технології в управлінні підприємством / І. І. Биков. — Київ: Ліра-К, 2013. — 256 с.
20. Пілат Н. Електронний документообіг і архівування: практичний посібник / Н. Пілат. — Київ: ДІП, 2015. — 168 с.
21. Бондарчук О. Є. Системи електронного документообігу в Україні: аналіз практик / О. Є. Бондарчук. — Київ: Видавництво КНЕУ, 2019. — 140 с.
22. Хоменко С. П. Захист інформації в комп'ютерних системах / С. П. Хоменко. — Київ: Наукова думка, 2011. — 312 с.
23. Кириченко О. В. Проєктування баз даних: навчальний посібник / О. В. Кириченко. — Київ: КНТ, 2018. — 248 с.
24. Пасічник В. О. Управління запасами та логістичні операції: підручник / В. О. Пасічник. — Львів: Видавничий центр, 2017. — 320 с.
25. Сергієнко Т. О. Мобільні додатки: розробка та тестування / Т. О. Сергієнко. — Харків: IT-Books, 2020. — 220 с.
26. Литвиненко О. М. Управління інформаційними системами підприємства / О. М. Литвиненко. — Київ: Наукова думка, 2015. — 280 с.

27. Захарчук І. П. Організація матеріально-технічного постачання: навч. посібник / І. П. Захарчук. — Вінниця: Нова Книга, 2016. — 252 с.
28. Замула І. В., Селецька Д. О. «Облікове забезпечення управління товарами на митному складі». Проблеми теорії та методології бухгалтерського обліку, контролю і аналізу, 2023. — 27–33с
29. Yerlan Berdaliyev, Alex Pappachen James. «RFID-Cloud Smart Cart System». arXiv, 2016. — URL: <https://arxiv.org/abs/1608.03724>
30. Chunan Tong. «An Efficient Intelligent Semi-Automated Warehouse Inventory Stocktaking System». arXiv, 2023 — URL: <https://arxiv.org/abs/2309.12365>
31. Abhi Desai. «Enhancing Inventory Management with Progressive Web Applications (PWAs): A Scalable Solution for Small and Large Enterprises». arXiv, 2025 — URL: <https://arxiv.org/abs/2506.11011>
32. Шапарець М. С. Дослідження технічних особливостей RFID-систем : стаття / КПІ, 2019
33. Резнік Н. Використання RFID-технології в логістиці та управлінні ланцюгами постачань. Molodiy Vcheniy, 2022.
34. Олійник В. Досвід використання цифрових RFID технологій в управлінні операціями підприємств : стаття, 2025.
35. Самсонова С. Ю. Розроблення автоматизованої системи управління складськими логістичними процесами, 2022, 5-10с
36. Android Developers — Introduction to activities, 2025 — URL: <https://developer.android.com/guide/components/activities/intro-activities>
37. Android API Reference — Intent, 2025. — URL: <https://developer.android.com/reference/android/content/Intent>
38. Firebase — Realtime Database, 2025 — URL: <https://firebase.google.com/docs/database>
39. John W. Toomey Inventory Management: Principles, Concepts and Techniques —
40. Ramgovind P., Engelbrecht W. Business, Warehouse and Inventory Management, 2021, 73-87с, URL — <https://www.vanschaiknet.com/catalogue/business-warehouse-and-inventory-management/>
41. Луценко І. С. Логістичне управління запасами: навчально-методичний комплекс дисципліни, 2021 р. — URL: <https://ela.kpi.ua/items/a5493d9d-e295->

[4bf3-9863-d18c1cad506e](https://ela.kpi.ua/items/a5493d9d-e295-4bf3-9863-d18c1cad506e)

42. С.В. Смерічевська Логістичний менеджмент: навчально-методичний комплекс, 2022 р. КПІ ім. Ігоря Сікорського — URL: <https://ela.kpi.ua/items/a5493d9d-e295-4bf3-9863-d18c1cad506e>

43. Олена О. Григоревська, В. П'явка Управлінський облік виробничих запасів: логістичний підхід, 2017 р. —URL: <https://er.knutd.edu.ua/handle/123456789/9561>

44. Крикавський Є.В. Економіка логістики: навч.посіб. / Є.В. Крикавський, О.А. Похильченко, Н.В. Чернописька, О.С. Костюк, Н.Б. Савіна, С.М. Нікчич, Л.Я. Якимишин; за заг. ред. Є.В. Крикавського, О.А. Похильченко. – Львів: Видавництво «Львівської політехніки», 2014. – 640с.

45. Антоненко О.М. Формування систем електронної торгівлі і логістика»: теорія та практика організацій / О.М. Антоненко, І.П. Міщук, О.О. Хамула // Вісник Національного університету «Львівська політехніка». – 2012, №749. – С. 349-356

46. Окландер М.А. Маркетингові дослідження збутової функції логістичних систем підприємств, 2012. – с.43-50.

47. Мельникова Н.В. Моделі управління прямими та зворотними матеріальними потоками в логістичній системі промислового підприємства, 2014. – 20 с.

48. Dwyer T. The Three Pillars of Effective Returns Management, 2012 – URL: <http://www.inboundlogistics.com/cms/article/the-three-pillars-of-effective-returns-management/>

49. Andrew Stellman & Jennifer Greene Learning Agile: Understanding Scrum, XP, Lean and Kanban, 2014 – URL: <https://pdfroom.com/books/oreilly-learning-agile-2015pdf/E1d4DnrvdOb>

50. Михайленко Павло, Серeda Катерина. Розробка автоматизованої системи обліку складських товарів та підтримки торгівлі. Всеукраїнська науково-технічна конференція факультету інтелектуальних інформаційних технологій та автоматизації, 2025 — URL: <https://ir.lib.vntu.edu.ua/handle/123456789/48183>

Додатки

Додаток А

ПРОТОКОЛ ПЕРЕВІРКИ КВАЛІФІКАЦІЙНОЇ РОБОТИ

Назва роботи: Розробка автоматизованої системи обліку складських товарів і підтримки торговельної діяльності. Частина 2. Розробка мобільного застосунку.

Тип роботи: магістерська кваліфікаційна робота
(бакалаврська кваліфікаційна робота / магістерська кваліфікаційна робота)

Підрозділ кафедра КСУ
(кафедра, факультет, навчальна група)

Коефіцієнт подібності текстових запозичень, виявлених у роботі системою StrikePlagiarism (КП1) 8,17 %

Висновок щодо перевірки кваліфікаційної роботи (відмітити потрібне)

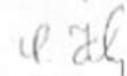
- Запозичення, виявлені у роботі, оформлені коректно і не містять ознак академічного плагіату, фабрикації, фальсифікації. Роботу прийняти до захисту.
- У роботі не виявлено ознак плагіату, фабрикації, фальсифікації, але надмірна кількість текстових запозичень та/або наявність типових розрахунків не дозволяють прийняти рішення про оригінальність та самостійність її виконання. Роботу направити на доопрацювання.
- У роботі виявлено ознаки академічного плагіату та/або в ній містяться навмисні спотворення тексту, що вказують на спроби приховування недобросовісних запозичень. Робота до захисту не приймається.

Експертна комісія:

Ковтун В.В., завідувач кафедри КСУ
(прізвище, ініціали, посада)


(підпис)

Ковалюк О.О., доцент кафедри КСУ
(прізвище, ініціали, посада)


(підпис)

Особа, відповідальна за перевірку 
(підпис)

Дубової В.М.
(прізвище, ініціали)

З висновком експертної комісії ознайомлений(-на)

Керівник 
(підпис)

Юхимчук М.С., професор кафедри КСУ
(прізвище, ініціали, посада)

Здобувач К. Sereday
(підпис)

Середа К.Я.
(прізвище, ініціали)

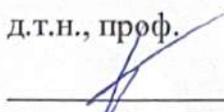
Додаток Б

ТЕХНІЧНЕ ЗАВДАННЯ МКР

Міністерство освіти і науки України
Вінницький національний технічний університет
Факультет інтелектуальних інформаційних технологій та автоматизації
Кафедра комп'ютерних систем управління

ЗАТВЕРДЖЕНО

Завідувач кафедри КСУ ВНТУ,
д.т.н., проф.

 В'ячеслав Ковтун

“ 17 ” жовтня 2025 р.

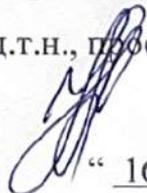
ТЕХНІЧНЕ ЗАВДАННЯ

на виконання комплексної магістерської кваліфікаційної роботи
Розробка автоматизованої системи обліку складських товарів і підтримки
торговельної діяльності. Частина 2. Розробка мобільного застосунку.

08-33.МКР.11.00.000 ТЗ

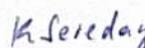
Керівник роботи:

д.т.н., професор кафедри КСУ

 Марія ЮХИМЧУК

“ 16 ” жовтня 2025р.

Виконавець:

 ст. гр. Катерина СЕРЕДА

“ 16 ” жовтня 2025р.

Вінниця 2025

1. Назва та галузь застосування:

1.1. Назва – Розробка автоматизованої системи обліку складських товарів та підтримки торгівлі. Частина 2. Розробка мобільного додатку.

1.2. Галузь застосування – Комп'ютеризовані системи автоматизації управлінської діяльності.

2. Підстава для проведення розробки:
Тема магістерської кваліфікаційної роботи затверджена наказом по ВНТУ № 313 від 24.09.2025р.

3. Мета та призначення розробки:

Метою магістерської кваліфікаційної роботи є підвищення ефективності роботи торгової мережі за рахунок створення мобільного додатку автоматизованої системи обліку складських товарів та торгівлі, що дозволить автоматизувати рутинні завдання та мінімізувати вплив помилок людського фактору на роботу підприємства, задля зменшення втрат та ефективного розподілу вивільнених робочих ресурсів на інші області роботи компанії.

4. Джерела розробки:

Магістерська кваліфікаційна робота виконується вперше. В ході проведення розробки повинні використовуватись такі документи:

4.1. Бондаренко, О. О. Розробка програмного забезпечення засобами Java / О. О. Бондаренко. – Харків: ХНУРЕ, 2021. – 210 с.

4.2. Крамаренко, А. О. Android-програмування: навчальний посібник / А. О. Крамаренко, С. М. Бондар. – Київ: КНЕУ, 2020. – 176 с.

4.3. Стеценко, В. М. Автоматизація обліку на підприємствах торгівлі / В. М. Стеценко. – Київ: Ліра-К, 2019. – 234 с.

4.4. Шевченко, І. В. Розробка мобільного додатку для обліку товарів на складі / І. В. Шевченко, Н. М. Коваль // Вісник ХНУРЕ. – 2022. – № 3. – С. 85–90.

4.5. Ярова, І. С. Інформаційні системи підтримки торгівлі: сучасні підходи / І. С. Ярова // Системи обробки інформації. – 2021. – № 1(159). – С. 40–44.

4.6. Тарасов, Ю. М. Мобільна розробка з Kotlin і Android Studio / Ю. М. Тарасов. – Львів: ЛНУ, 2023. – 198 с.

5. Вимоги до розробки:

5.1. Перелік головних функцій:

5.1.1. управління товарними запасами;

5.1.2. управління документами та звітністю;

5.1.3. управління списком контрагентів компанії;

- 5.1.4. управління підрозділами компанії;
- 5.1.5. прийом та відправлення розрахунків;
- 5.1.6. здійснення логістики товарів між підрозділами компанії.
- 5.2. Основні технічні вимоги до розробки:
 - 5.2.1. Вимоги до програмної платформи:
 - 5.2.1.1. WINDOWS 7\8\10;
 - 5.2.1.2. Версія ОС Android вище API 14: Android 4.0 (IceCreamSandwich).
 - 5.2.2. Умови експлуатації системи:
 - 5.2.2.1. робота на стандартних ПЕОМ в приміщеннях зі стандартними умовами;
 - 5.2.2.2. Можливість цілодобового функціонування системи;
 - 5.2.2.3. Текст програмного забезпечення є цілком закритим.
- 6. Стадії та етапи розробки:
 - 6.1. Пояснювальна записка:
 - 6.1.1. Дослідження актуальності поставленої Задачі «19» _жовтня_ 2025 р.
 - 6.1.2. Загальний огляд систем управління обліку складських товарів «19» жовтня_ 2025 р.
 - 6.1.3. Аналіз функцій автоматизованих систем управління обліку складських товарів та торгівлі та порівняння програм аналогів «_1_» листопада 2025 р.
 - 6.1.4. Розробка структури програмного забезпечення Системи «5» листопада 2025 р.
 - 6.2. Графічні матеріали:
 - 6.2.1. Розробка UML-діаграм «_10» листопада 2025 р.
 - 6.2.2. Тестування програмного забезпечення «20» листопада 2025 р.
- 7. Порядок контролю і приймання
 - 7.1. Хід виконання роботи контролюється керівником роботи. Рубіжний контроль провести до «14» листопада 2025 р.
 - 7.2. Атестація проекту здійснюється на попередньому захисті. Попередній захист магістерської кваліфікаційної роботи провести до «10» грудня 2025 р.
 - 7.3. Підсумкове рішення щодо оцінки якості виконання роботи приймається на засіданні ЕК. Захист магістерської кваліфікаційної роботи провести до «18» _грудня_ 2025 р.

Додаток В

Лістинг програмного забезпечення

AndroidManifest.xml

```

<manifest
xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:tools="http://schemas.android.com/tools"
package="com.vntu.techpay">

    <permission
android:name="com.vntu.techpay.permission.MAPS_RECEIVE"
android:protectionLevel="signature">
    </permission>

    <uses-permission
android:name="android.permission.INTERNET" />
    <uses-permission
android:name="android.permission.ACCESS_NETWORK_STATE"
/>
    <uses-permission
android:name="android.permission.ACCESS_WIFI_STATE" />
    <!-- <uses-permission
android:name="com.google.android.providers.gsf.permission.READ
_GSERVICES"/> -->
    <uses-permission
android:name="com.vntu.techpay.permission.MAPS_RECEIVE"/>
    <uses-permission android:name="android.permission.CAMERA"
/>
    <uses-permission
android:name="android.permission.CALL_PHONE" />
    <uses-permission
android:name="android.permission.ACCESS_NETWORK_STATE"
/>
    <uses-permission
android:name="android.permission.ACCESS_WIFI_STATE" />
    <!-- External storage for caching. -->
    <uses-permission
android:name="android.permission.READ_EXTERNAL_STORAG
E" />
    <uses-permission
android:name="android.permission.WRITE_EXTERNAL_STORAG
E" />
    <!-- My Location -->
    <uses-permission
android:name="android.permission.ACCESS_COARSE_LOCATIO
N" />
    <uses-permission
android:name="android.permission.ACCESS_FINE_LOCATION"/>
    <uses-permission
android:name="android.permission.GET_ACCOUNTS" />
    <uses-permission
android:name="com.google.android.c2dm.permission.RECEIVE" />
    <!-- Maps API needs OpenGL ES 2.0. -->
    <uses-feature
        android:glEsVersion="0x00020000"
        android:required="true"/>
    <uses-feature android:name="android.hardware.camera"
        android:required="true" />
    <!--
=====GCM=====
=====-->
    <uses-permission
android:name="android.permission.WAKE_LOCK" />
    <permission
android:name="com.vntu.techpay.permission.C2D_MESSAGE"
        android:protectionLevel="signature" />
    <uses-permission
android:name="by.keycard.admin.test.permission.C2D_MESSAGE"
/>
    <!--
=====END_GCM=====
=====-->
</manifest>

<application
android:name="com.vntu.techpay.AppStart"
android:allowBackup="true"
android:hardwareAccelerated="true"
android:supportRtl="true"
android:largeHeap="true"
android:icon="@drawable/logo"
android:label="@string/app_name"
android:configChanges="locale">
    <!--=====Facebook=====-->
    <meta-data android:name="com.facebook.sdk.ApplicationId"
        android:value="@string/facebook_app_id"/>
    <activity android:name="com.facebook.FacebookActivity"
        android:configChanges=
"keyboard|keyboardHidden|screenLayout|screenSize|orientation"
        android:label="@string/app_name" />
    <activity
        android:name="com.facebook.CustomTabActivity"
        android:exported="true">
        <intent-filter>
            <action android:name="android.intent.action.VIEW" />
            <category
android:name="android.intent.category.DEFAULT" />
            <category
android:name="android.intent.category.BROWSABLE" />
            <data
android:scheme="@string/fb_login_protocol_scheme" />
        </intent-filter>
        </activity>
    <!--=====End_Facebook=====-->
    <meta-data
android:name="com.google.android.maps.v2.API_KEY"
android:value="AIzaSyByVBP4wOop6WBUlzo3gtoQ5xYsCB4IKQ
c"/>
    <!--<meta-data-->
    <!--android:name="com.google.android.gms.version"-->
    <!--android:value="@integer/google_play_services_version"
/>-->
    <service android:name="com.vntu.techpay.MyService" />
    <activity
        android:name="com.vntu.techpay.Activities.MainActivity"
        android:theme="@style/MyMaterialTheme"
        android:screenOrientation="portrait"
        android:label="@string/app_name"
        android:windowSoftInputMode="stateAlwaysHidden"
        android:configChanges="locale">
        <intent-filter>
            <action android:name="android.intent.action.MAIN" />
            <category
android:name="android.intent.category.LAUNCHER" />
        </intent-filter>
        </activity>
    <activity
        android:name="com.vntu.techpay.Activities.LoginActivity"
        android:theme="@style/MyMaterialTheme"
        android:screenOrientation="portrait"
        android:label="@string/label_login"
        android:configChanges="locale">
        </activity>
    <activity
        android:name="com.vntu.techpay.Activities.RegistrationActivity"
        android:allowBackup="true"
        android:screenOrientation="portrait"
        android:theme="@style/MyMaterialTheme"
        android:label="@string/label_registration"
        android:configChanges="locale">
        </activity>
    <activity
        android:name="com.vntu.techpay.Activities.ViewPagerWishActivity"
        android:theme="@style/MyMaterialTheme"
        android:screenOrientation="portrait"
        android:label="@string/label_viewpager_wish"
    </activity>

```

```

        android:configChanges="locale">
    </activity>
</activity>
android:name="com.vntu.techpay.Activities.RulesForSupportActivit
y"
    android:theme="@style/MyMaterialTheme"
    android:screenOrientation="portrait"
    android:label="@string/label_support"
    android:configChanges="locale">
</activity>
</activity>
android:name="com.vntu.techpay.Activities.QuestionsForSupportAc
tivity"
    android:theme="@style/MyMaterialTheme"
    android:screenOrientation="portrait"
    android:label="@string/label_support"
    android:configChanges="locale">
</activity>
</activity>
android:name="com.vntu.techpay.Activities.WishWhatActivity"
    android:theme="@style/MyMaterialTheme"
    android:screenOrientation="portrait"
    android:label="@string/label_viewpager_wish"
    android:configChanges="locale">
</activity>
</activity>
    android:name="com.vntu.techpay.Activities.PushActivity"
    android:theme="@style/MyMaterialTheme"
    android:screenOrientation="portrait"
    android:label="@string/label_push"
    android:configChanges="locale">
</activity>
</activity>
    android:name="com.vntu.techpay.Activities.ShareActivity"
    android:theme="@style/MyMaterialTheme"
    android:screenOrientation="portrait"
    android:label="@string/label_share"
    android:configChanges="locale">
</activity>
</activity>
android:name="com.vntu.techpay.Activities.PartnersCategoryActivit
y"
    android:theme="@style/MyMaterialTheme"
    android:screenOrientation="portrait"
    android:label="@string/label_partners"
    android:configChanges="locale">
</activity>
</activity>
android:name="com.vntu.techpay.Activities.PartnersServicesActivit
y"
    android:theme="@style/MyMaterialTheme"
    android:screenOrientation="portrait"
    android:label="@string/label_services"
    android:configChanges="locale">
</activity>
</activity>
android:name="com.vntu.techpay.Activities.ViewPagerPartnerActivit
y"
    android:theme="@style/MyMaterialTheme"
    android:screenOrientation="portrait"
    android:label="@string/label_partners"
    android:configChanges="locale">
</activity>
</activity>
android:name="com.vntu.techpay.Activities.SocialNetworkActivity"
    android:theme="@style/MyMaterialTheme"
    android:screenOrientation="portrait"
    android:label="@string/label_social"
    android:configChanges="locale">
</activity>
<!--
=====GCM=====
-->

```

```

<receiver
    android:name="com.google.android.gms.gcm.GcmReceiver"

```

```

        android:exported="true"
    android:permission="com.google.android.c2dm.permission.SEND">
        <intent-filter>
            <action
                android:name="com.google.android.c2dm.intent.RECEIVE"/>
            <category android:name="by.keycard.admin.test"/>
        </intent-filter>
    </receiver>
</service>
    android:name="com.vntu.techpay.GCM.GCMPushReceiverService"
    android:exported="false">
        <intent-filter>
            <action
                android:name="com.google.android.c2dm.intent.RECEIVE"/>
        </intent-filter>
    </service>
</service>
    android:name="com.vntu.techpay.GCM.GCMRegistrationIntentServ
ice" android:exported="false">
        <intent-filter>
            <action
                android:name="com.google.android.gms.iid.InstanceID"/>
        </intent-filter>
    </service>
<!--
=====END_GCM=====
-->
</application>
</manifest>

```

Gradle, module app

```

apply plugin: 'com.android.application'
apply plugin: 'com.google.gms.google-services'

```

```

android {
    compileSdkVersion 23
    buildToolsVersion "25.0.0"
    useLibrary 'org.apache.http.legacy'

    defaultConfig {
        applicationId "com.vntu.techpay"
        minSdkVersion 16
        targetSdkVersion 23
        versionCode 1
        versionName "1.0"
    }
    buildTypes {
        release {
            minifyEnabled false
            proguardFiles getDefaultProguardFile('proguard-android.txt'),
'proguard-rules.pro'
        }
    }
    dexOptions {
        javaMaxHeapSize "2g"
    }
}

```

```

dependencies {
    compile fileTree(dir: 'libs', include: ['*.jar'])
    testCompile 'junit:junit:4.12'
    compile 'com.android.support:appcompat-v7:23.1.1'
    compile "com.google.android.gms:play-services-location:8.3.0"
    compile "com.google.android.gms:play-services-maps:8.3.0"
    compile "com.google.android.gms:play-services-gcm:8.3.0"
    compile 'com.android.support:design:23.2.0'
    compile 'com.android.support:recyclerview-v7:23.1.1'
    compile 'com.android.support:cardview-v7:23.1.1'
    //Виджеты и сторонние библиотеки
    compile 'com.github.rey5137:material:1.2.2'
    compile 'org.apache.httpcomponents:httpcore:4.4.1'
    compile 'com.phlox.widget:android-date-picker:0.2.1'
    compile 'com.github.bumptech.glide:glide:3.7.0'
    compile 'jp.wasabeef:glide-transformations:2.0.2'
    compile 'de.hdodenhof:circleimageview:2.1.0'
    //Slider
    compile 'com.squareup.picasso:picasso:2.5.2'
    compile 'com.nineoldandroids:library:2.4.0'
    compile 'com.daimajia.slider:library:1.1.5@aar'
}

```

```
//facebook
compile 'com.facebook.android:facebook-android-sdk:4.2.0'
//VK
compile 'com.vk:androidsdk:1.6.5'
}
```

AppStart.java

```
package com.vntu.techpay;

import android.app.Application;
import android.content.Context;
import android.content.Intent;
import android.content.res.Configuration;
import android.telephony.TelephonyManager;

import com.facebook.FacebookSdk;
import com.facebook.appevents.AppEventsLogger;
import com.vk.sdk.VKAccessToken;
import com.vk.sdk.VKAccessTokenTracker;
import com.vk.sdk.VKSdk;
import com.vk.sdk.util.VKUtil;

import java.util.Locale;

import com.vntu.techpay.Activities.MainActivity;

public class AppStart extends Application {

    VKAccessTokenTracker vkAccessTokenTracker = new
VKAccessTokenTracker() {
        @Override
        public void
onVKAccessTokenChanged(VKAccessToken oldToken,
VKAccessToken newToken) {
            if (newToken == null) {
                Intent intent = new
Intent(AppStart.this, MainActivity.class);

                intent.setFlags(Intent.FLAG_ACTIVITY_NEW_TASK|In
tent.FLAG_ACTIVITY_CLEAR_TOP);

                startActivity(intent);
                // VKAccessToken is
invalid
            }
        }
    };

    @Override
    public void onCreate() {
        super.onCreate();
        App.initInstance();
        TelephonyManager tm =
(TelephonyManager) getSystemService(Context.TELEPHONY_SER
VICE);

        String countryCode =
tm.getNetworkCountryIso();
        if (countryCode.contains("by") ||
countryCode.contains("be")) {
            setLocale(0, getBaseContext());
        } else if (countryCode.contains("ru")) {
            setLocale(1, getBaseContext());
        } else if (countryCode.contains("ua")) {
            setLocale(1, getBaseContext());
        } else {
            setLocale(2, getBaseContext());
        }
        //=====FB=====

        FacebookSdk.sdkInitialize(getApplicationContext());
        AppEventsLogger.activateApp(this);
        //=====VK=====
        VKSdk.initialize(getApplicationContext());
        String[] fingerprints =
VKUtil.getCertificateFingerprint(this, this.getPackageName());
        System.out.println("vkToken " +
fingerprints.length + " " + fingerprints[0]);
    }

    public void setLocale(int numb_locale, Context context) {
        Locale locale;
        Configuration configuration;
```

```
switch (numb_locale) {
    case 0:
        locale = new
Locale("be");

        Locale.setDefault(locale);

        Configuration();

        locale;

        context.getResources().updateConfiguration(configuration
, null);

        break;

    case 1:
        locale = new
Locale("ru");

        Locale.setDefault(locale);

        Configuration();

        locale;

        context.getResources().updateConfiguration(configuration
, null);

        break;

    case 2:
        locale = new
Locale("en");

        Locale.setDefault(locale);

        Configuration();

        locale;

        context.getResources().updateConfiguration(configuration
, null);

        break;
}
```

DBHelper.java

```
package com.vntu.techpay.DB;

import android.content.Context;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;

import com.vntu.techpay.App;

public class DatabaseHelper extends SQLiteOpenHelper {

    public static final String DATABASE_NAME =
"keycard_user_database_test7.db";
    private static final int DATABASE_VERSION = 68;

    public static final String TABLE_PARTNER = "partner";
    public static final String PARTNER_ID = "id";
    public static final String PARTNER_NAME = "name";
    public static final String PARTNER_DESCRIPTION =
"description";
    public static final String PARTNER_ADDRESS =
"address";
    public static final String PARTNER_SITE = "site";
    public static final String PARTNER_LOGO = "logo";
    public static final String PARTNER_LON = "lon";
    public static final String PARTNER_LAT = "lat";
    public static final String PARTNER_WORKTIME =
"worktime";
    public static final String PARTNER_WORKDAYS =
"workdays";
    public static final String PARTNER_CATEGORY_ID =
"category_id";
    public static final String
PARTNER_SUBCATEGORY_ID = "subcategory_id";
    public static final String PARTNER_DISCOUNT =
"discount";
    public static final String PARTNER_PHONE = "phone";
```

```

        public static final String PARTNER_PHONE2 =
"phone2";
        public static final String PARTNER_FAVOURITE =
"favourite";
        public static final String PARTNER_STATUS = "status";
        public static final String PARTNER_RAITING =
"raiting";
        public static final String PARTNER_SERVICE_ID =
"service_id";
        public static final String PARTNER_COUNTRY_ID =
"country_id";
        public static final String PARTNER_REGION_ID =
"region_id";
        public static final String PARTNER_CITY_ID =
"city_id";

        public static final String TABLE_CATEGORY =
"category";
        public static final String CATEGORY_ID = "id";
        public static final String CATEGORY_TITLE = "title";
        public static final String CATEGORY_PARENT_ID =
"parent_id";
        public static final String CATEGORY_LOGO = "logo";
        public static final String
CATEGORY_PARTNER_COUNT = "partner_count";
        public static final String CATEGORY_IS_DELETE =
"is_delete";

        public static final String
TABLE_PRODUCT_CATEGORY = "product_category";
        public static final String ID_PRODUCT_CATEGORY =
"id_product_category";
        public static final String
PRODUCT_CATEGORY_NAME= "product_category_name";
        public static final String
PRODUCT_CATEGORY_PARENT_ID =
"product_category_parent_id";
        public static final String
PRODUCT_CATEGORY_STATUS = "product_category_status";

        public static final String TABLE_HISTORY = "history";
        public static final String HISTORY_ID_SERVER =
"id_server";
        public static final String HISTORY_USER_ID =
"user_id";
        public static final String HISTORY_PARTNER_ID =
"partner_id";
        public static final String HISTORY_DATE = "date";
        public static final String HISTORY_AMOUNT =
"amount";
        public static final String HISTORY_DISCOUNT =
"discount";
        public static final String HISTORY_DEVICE_ID=
"device_id";
        public static final String HISTORY_CALC = "calc";
        public static final String HISTORY_STATUS = "status";

        public static final String TABLE_WISH = "wish";
        public static final String WISH_ID = "wish_id";
        public static final String WISH_CATEGORY_ID =
"category_id";
        public static final String WISH_SUBCATEGORY_ID =
"subcategory_id";
        public static final String WISH_PRODUCT_NAME =
"product_name";
        public static final String WISH_TIME_ID =
"time_purchase";
        public static final String WISH_STATUS = "status";

        public static final String TABLE_PRODUCT =
"product";
        public static final String PRODUCT_ID = "product_id";
        public static final String PRODUCT_CATEGORY_ID =
"product_category_id";
        public static final String PRODUCT_PARTNER_ID =
"product_partner_id";
        public static final String PRODUCT_NAME =
"product_name";
        public static final String PRODUCT_LOGO =
"product_logo";

        public static final String
PRODUCT_DESCRIPTION_SHORT =
"product_description_short";
        public static final String PRODUCT_DESCRIPTION =
"product_description";
        public static final String PRODUCT_PRICE =
"product_price";
        public static final String PRODUCT_CURRENCY_ID =
"product_currency_id";
        public static final String PRODUCT_STATUS =
"product_status";

        public static final String TABLE_SERVICE = "service";
        public static final String SERVICE_ID = "service_id";
        public static final String SERVICE_PARTNER_COUNT
= "partner_count";
        public static final String SERVICE_NAME =
"service_name";
        public static final String SERVICE_LOGO =
"service_logo";
        public static final String SERVICE_IS_DELETE =
"service_is_delete";

        public static final String TABLE_SHARE = "share";
        public static final String SHARE_ID = "id";
        public static final String SHARE_PARTNER_ID =
"partner_id";
        public static final String SHARE_HEADER = "header";
        public static final String SHARE_LOGO = "logo";
        public static final String SHARE_DISCRIPTION =
"discription";
        public static final String SHARE_STATUS = "status";

        public static final String TABLE_PARTNER_IMAGE=
"partner_image";
        public static final String PARTNER_IMAGE_ID =
"partner_id";
        public static final String PARTNER_IMAGE_NAME =
"name";

        public static final String TABLE_PUSH = "push";
        public static final String PUSH_ID = "push_id";
        public static final String PUSH_PARTNER_ID =
"push_partner_id";
        public static final String PUSH_TITLE = "push_title";
        public static final String PUSH_SHORT_DESCRIPTION
= "push_short_description";
        public static final String PUSH_CONTENT =
"push_content";
        public static final String PUSH_STATUS =
"push_status";
        public static final String PUSH_CREATETIME =
"push_createtime";

        public static final String TABLE_FAQ = "faq";
        public static final String FAQ_ID = "faq_id";
        public static final String FAQ_QUESTION =
"faq_question";
        public static final String FAQ_ANSWER = "faq_answer";
        public static final String FAQ_LOGO = "faq_logo";
        public static final String FAQ_CREATETIME =
"faq_createtime";
        public static final String FAQ_STATUS = "faq_status";

        public static final String TABLE_ADVERTISING =
"advertising";
        public static final String ADVERTISING_ID =
"advertising_id";
        public static final String ADVERTISING_TYPE =
"advertising_type";
        public static final String ADVERTISING_LOGO =
"advertising_logo";
        public static final String ADVERTISING_CREATETIME
= "advertising_createtime";
        public static final String ADVERTISING_URL =
"advertising_url";
        public static final String ADVERTISING_PAGES =
"advertising_pages";
        public static final String ADVERTISING_IS_DELETE =
"advertising_is_delete";

```

```

        private static final String SQL_CREATE_PARTNER =
"CREATE TABLE IF NOT EXISTS "
        + TABLE_PARTNER + " (" +
PARTNER_ID + " INTEGER PRIMARY KEY
AUTOINCREMENT,"
        + PARTNER_NAME + "
VARCHAR(40),"
        + PARTNER_DESCRIPTION + "
TEXT,"
        + PARTNER_ADDRESS + "
VARCHAR(100),"
        + PARTNER_SITE + "
VARCHAR(100),"
        + PARTNER_LOGO + "
VARCHAR(200),"
        + PARTNER_LON + "
VARCHAR(40),"
        + PARTNER_LAT + "
VARCHAR(40),"
        + PARTNER_WORKTIME + "
VARCHAR(20),"
        + PARTNER_WORKDAYS + "
VARCHAR(20),"
        + PARTNER_CATEGORY_ID + "
INTEGER(11),"
        +
PARTNER_SUBCATEGORY_ID + " INTEGER(11),"
        + PARTNER_DISCOUNT + "
VARCHAR(40),"
        + PARTNER_PHONE + "
VARCHAR(20),"
        + PARTNER_PHONE2 + "
VARCHAR(20),"
        + PARTNER_FAVOURITE + "
INTEGER(1),"
        + PARTNER_STATUS + "
INTEGER(1),"
        + PARTNER_RAITING + "
REAL,"
        + PARTNER_SERVICE_ID + "
INTEGER(11),"
        + PARTNER_COUNTRY_ID + "
INTEGER(11),"
        + PARTNER_REGION_ID + "
INTEGER(11),"
        + PARTNER_CITY_ID + "
INTEGER(11));";

        private static final String SQL_CREATE_CATEGORY =
"CREATE TABLE IF NOT EXISTS "
        + TABLE_CATEGORY + " (" +
CATEGORY_ID + " INTEGER PRIMARY KEY
AUTOINCREMENT,"
        + CATEGORY_TITLE + "
VARCHAR(40),"
        + CATEGORY_PARENT_ID + "
INTEGER(11),"
        + CATEGORY_LOGO + "
VARCHAR(255),"
        +
CATEGORY_PARTNER_COUNT + " INTEGER(11),"
        + CATEGORY_IS_DELETE + "
INTEGER(1));";

        private static final String
SQL_CREATE_PRODUCT_CATEGORY = "CREATE TABLE IF
NOT EXISTS "
        +
TABLE_PRODUCT_CATEGORY + " (" +
ID_PRODUCT_CATEGORY + " INTEGER PRIMARY KEY
AUTOINCREMENT,"
        +
PRODUCT_CATEGORY_NAME + " VARCHAR(40),"
        +
PRODUCT_CATEGORY_PARENT_ID + " INTEGER(11),"
        +
PRODUCT_CATEGORY_STATUS + " INTEGER(1));";

        public static final String SQL_CREATE_HISTORY =
"CREATE TABLE IF NOT EXISTS "
        + TABLE_HISTORY + " ("
        + HISTORY_ID_SERVER + " INTEGER(11),"
        + HISTORY_USER_ID + " INTEGER(11),"
        + HISTORY_PARTNER_ID + "
INTEGER(11),"
        + HISTORY_DATE + "
VARCHAR(40),"
        + HISTORY_AMOUNT + "
INTEGER(11),"
        + HISTORY_DISCOUNT + "
INTEGER(11),"
        + HISTORY_DEVICE_ID + "
INTEGER(11),"
        + HISTORY_STATUS + "
INTEGER(1),"
        + HISTORY_CALC + "
INTEGER(11));";

        public static final String SQL_CREATE_WISH =
"CREATE TABLE IF NOT EXISTS "
        + TABLE_WISH + " ("
        + WISH_ID + " INTEGER(11),"
        + WISH_CATEGORY_ID + "
INTEGER(11),"
        + WISH_SUBCATEGORY_ID + "
INTEGER(11),"
        + WISH_PRODUCT_NAME + "
VARCHAR(11),"
        + WISH_STATUS + "
VARCHAR(1),"
        + WISH_TIME_ID + "
VARCHAR(40));";

        public static final String SQL_CREATE_PRODUCT =
"CREATE TABLE IF NOT EXISTS "
        + TABLE_PRODUCT + " ("
        + PRODUCT_ID + "
INTEGER(11),"
        + PRODUCT_CATEGORY_ID + "
INTEGER(11),"
        + PRODUCT_PARTNER_ID + "
INTEGER(11),"
        + PRODUCT_NAME + "
VARCHAR(255),"
        + PRODUCT_LOGO + "
VARCHAR(255),"
        +
PRODUCT_DESCRIPTION_SHORT + " VARCHAR(255),"
        + PRODUCT_DESCRIPTION + "
TEXT,"
        + PRODUCT_PRICE + "
VARCHAR(255),"
        + PRODUCT_STATUS + "
INTEGER(1),"
        + PRODUCT_CURRENCY_ID + "
INTEGER(11));";

        public static final String SQL_CREATE_SERVICE =
"CREATE TABLE IF NOT EXISTS "
        + TABLE_SERVICE + " ("
        + SERVICE_ID + "
INTEGER(11),"
        + SERVICE_PARTNER_COUNT
+ " INTEGER(11),"
        + SERVICE_NAME + "
VARCHAR(255),"
        + SERVICE_LOGO + "
VARCHAR(255),"
        + SERVICE_IS_DELETE + "
INTEGER(11));";

        public static final String SQL_CREATE_SHARE =
"CREATE TABLE IF NOT EXISTS "
        + TABLE_SHARE + " ("
        + SHARE_ID + " INTEGER(11),"
        + SHARE_PARTNER_ID + "
INTEGER(11),"
        + SHARE_HEADER + "
VARCHAR(255),"
        + SHARE_LOGO + "
VARCHAR(255),"
        + SHARE_STATUS + "
INTEGER(1),"

```

```

        + SHARE_DISCRIPTION + "
TEXT);";

        public static final String
SQL_CREATE_PARTNER_IMAGE = "CREATE TABLE IF NOT
EXISTS "
        + TABLE_PARTNER_IMAGE + "
("
        + PARTNER_IMAGE_ID + "
INTEGER(11),"
        + PARTNER_IMAGE_NAME + "
VARCHAR(255));";

        public static final String SQL_CREATE_PUSH =
"CREATE TABLE IF NOT EXISTS "
        + TABLE_PUSH + " ("
        + PUSH_ID + " INTEGER(11),"
        + PUSH_PARTNER_ID + "
INTEGER(11),"
        + PUSH_TITLE + "
VARCHAR(256),"
        + PUSH_SHORT_DESCRIPTION
+ " VARCHAR(256),"
        + PUSH_CONTENT + " TEXT,"
        + PUSH_STATUS + "
INTEGER(11),"
        + PUSH_CREATETIME + "
VARCHAR(16));";

        public static final String SQL_CREATE_FAQ =
"CREATE TABLE IF NOT EXISTS "
        + TABLE_FAQ + " ("
        + FAQ_ID + " INTEGER(11),"
        + FAQ_QUESTION + "
VARCHAR(256),"
        + FAQ_ANSWER + "
VARCHAR(256),"
        + FAQ_LOGO + "
VARCHAR(256),"
        + FAQ_STATUS + "
INTEGER(1),"
        + FAQ_CREATETIME + "
VARCHAR(40));";

        public static final String SQL_CREATE_ADVERTISING
= "CREATE TABLE IF NOT EXISTS "
        + TABLE_ADVERTISING + " ("
        + ADVERTISING_ID + "
INTEGER(11),"
        + ADVERTISING_TYPE + "
INTEGER(1),"
        + ADVERTISING_LOGO + "
VARCHAR(256),"
        + ADVERTISING_CREATETIME
+ " VARCHAR(256),"
        + ADVERTISING_URL + "
VARCHAR(256),"
        + ADVERTISING_PAGES + "
VARCHAR(256),"
        + ADVERTISING_IS_DELETE +
" INTEGER(1));";

        private static final String SQL_DELETE_PARTNER =
"DROP TABLE IF EXISTS "
        + TABLE_PARTNER;

        private static final String SQL_DELETE_CATEGORY =
"DROP TABLE IF EXISTS "
        + TABLE_CATEGORY;

        private static final String
SQL_DELETE_PRODUCT_CATEGORY = "DROP TABLE IF
EXISTS "
        + TABLE_CATEGORY;

        private static final String SQL_DELETE_HISTORY =
"DROP TABLE IF EXISTS "
        + TABLE_HISTORY;

        private static final String SQL_DELETE_WISH =
"DROP TABLE IF EXISTS "
        + TABLE_WISH;

        private static final String SQL_DELETE_PRODUCT =
"DROP TABLE IF EXISTS "
        + TABLE_PRODUCT;

        private static final String SQL_DELETE_SERVICE =
"DROP TABLE IF EXISTS "
        + TABLE_SERVICE;

        private static final String SQL_DELETE_SHARE =
"DROP TABLE IF EXISTS "
        + TABLE_SHARE;

        private static final String
SQL_DELETE_PARTNER_IMAGE = "DROP TABLE IF EXISTS
"
        + TABLE_PARTNER_IMAGE;

        private static final String SQL_DELETE_PUSH =
"DROP TABLE IF EXISTS "
        + TABLE_PUSH;

        private static final String SQL_DELETE_FAQ = "DROP
TABLE IF EXISTS "
        + TABLE_FAQ;

        private static final String
SQL_DELETE_ADVERTISING = "DROP TABLE IF EXISTS "
        + TABLE_ADVERTISING;

        public DatabaseHelper(Context context) {
            super(context, DATABASE_NAME, null,
DATABASE_VERSION);
        }

        @Override
        public void onCreate(SQLiteDatabase db) {
            App.log("Создание базы данных");
            db.execSQL(SQL_CREATE_PARTNER);
            db.execSQL(SQL_CREATE_CATEGORY);

            db.execSQL(SQL_CREATE_PRODUCT_CATEGORY);
            db.execSQL(SQL_CREATE_HISTORY);
            db.execSQL(SQL_CREATE_WISH);
            db.execSQL(SQL_CREATE_PRODUCT);
            db.execSQL(SQL_CREATE_SERVICE);
            db.execSQL(SQL_CREATE_SHARE);

            db.execSQL(SQL_CREATE_PARTNER_IMAGE);
            db.execSQL(SQL_CREATE_PUSH);
            db.execSQL(SQL_CREATE_FAQ);

            db.execSQL(SQL_CREATE_ADVERTISING);
        }

        @Override
        public void onUpgrade(SQLiteDatabase db, int
oldVersion, int newVersion) {
            App.log("Обновление базы данных с версии
" + oldVersion + " до версии " + newVersion + ", которое удалит
все старые данные");
            db.execSQL(SQL_DELETE_PARTNER);
            db.execSQL(SQL_DELETE_CATEGORY);

            db.execSQL(SQL_DELETE_PRODUCT_CATEGORY);
            db.execSQL(SQL_DELETE_HISTORY);
            db.execSQL(SQL_DELETE_WISH);
            db.execSQL(SQL_DELETE_PRODUCT);
            db.execSQL(SQL_DELETE_SERVICE);
            db.execSQL(SQL_DELETE_SHARE);

            db.execSQL(SQL_DELETE_PARTNER_IMAGE);
            db.execSQL(SQL_DELETE_PUSH);
            db.execSQL(SQL_DELETE_FAQ);

            db.execSQL(SQL_DELETE_ADVERTISING);
            onCreate(db); // Создаём новый экземпляр
таблицы
        }
    }
}

```

```

package com.vntu.techpay;

import android.app.Activity;
import android.app.Application;
import android.content.Context;
import android.content.SharedPreferences;
import android.content.SharedPreferences.Editor;

public class AppPreferences extends Application {

    public static final String
KEY_PREFS_USER_CONNECTION_DATA =
"user_connection_data_v44";
    public static final String KEY_PREFS_UPDATE_TIME =
"update_time_v44";
    public static final String KEY_PREFS_USER_BALANCE =
"user_balance_v44";
    public static final String
KEY_PREFS_USER_BALANCE_AVAILABLE =
"user_balance_available_v44";
    public static final String KEY_PREFS_CMS_TOKEN =
"cms_token_v6";
    //=====PROFILE=====
    public static final String KEY_PREFS_USER_ID = "user_id_v6";
    public static final String KEY_PREFS_USER_EMAIL =
"user_email_v6";
    public static final String KEY_PREFS_USER_PHONE =
"user_phone_v6";
    public static final String KEY_PREFS_USER_LOGO =
"user_logo_v6";
    public static final String KEY_PREFS_USER_COUNTRY_ID =
"user_country_id_v6";
    public static final String KEY_PREFS_USER_REGION_ID =
"user_region_id_v6";
    public static final String KEY_PREFS_USER_CITY_ID =
"user_city_id_v6";
    public static final String KEY_PREFS_USER_FIRSTNAME =
"user_firstname_v6";
    public static final String KEY_PREFS_USER_LASTNAME =
"user_lastname_v6";
    public static final String KEY_PREFS_USER_MIDDLENAME =
"user_middlename_v6";
    public static final String KEY_PREFS_USER_SEX =
"user_sex_v6";
    public static final String
KEY_PREFS_USER_MARITAL_STATUS =
"user_marital_status_v6";
    public static final String KEY_PREFS_USER_IS_AUTO =
"user_is_auto_v6";
    public static final String
KEY_PREFS_USER_SOCIAL_NETWORK =
"user_social_network_v6";
    public static final String
KEY_PREFS_USER_MOBILE_OPERATOR =
"user_mobile_operator_v6";
    public static final String KEY_PREFS_USER_BIRTH_DAY =
"user_birth_day_v6";
    public static final String KEY_PREFS_USER_BIRTH_MONTH
= "user_birth_month_v6";
    public static final String KEY_PREFS_USER_BIRTH_YEAR =
"user_birth_year_v6";
    public static final String KEY_PREFS_USER_HAS_CHILDRED
= "user_has_children_v6";
    public static final String KEY_PREFS_USER_KITCHEN =
"user_kitchen_v6";
    public static final String KEY_PREFS_USER_INTERESTS =
"user_interests_v6";
    public static final String KEY_PREFS_USER_SOCIAL_STATUS
= "user_social_status_v6";
    public static final String KEY_PREFS_USER_FREQ_TRAVEL =
"user_freq_travel_v6";

    private static final String APP_SHARED_PREFS =
AppPreferences.class.getSimpleName();
    private SharedPreferences _sharedPrefs;
    private Editor _prefsEditor;

    public AppPreferences(Context context) {
        this._sharedPrefs =
context.getSharedPreferences(APP_SHARED_PREFS,
Activity.MODE_PRIVATE);
        this._prefsEditor = _sharedPrefs.edit();
    }

    // ConnectionData - данные которые приходят с сервера
    // строкой в json формате
    public String getConnectionData() {
        return
_sharedPrefs.getString(KEY_PREFS_USER_CONNECTION_DATA, "");
    }

    public void setConnectionData(String text) {
        _prefsEditor.putString(KEY_PREFS_USER_CONNECTION_DATA, text);
        _prefsEditor.commit();
    }

    // дата последнего обновления приложения, данные
    // сохраняются в unixtime
    public String getUpdateTime() {
        return
_sharedPrefs.getString(KEY_PREFS_UPDATE_TIME, "1");
    }

    public void setUpdateTime(String update_time) {
        _prefsEditor.putString(KEY_PREFS_UPDATE_TIME,
update_time);
        _prefsEditor.commit();
    }

    // balance - баланс пользователя
    public String getBalance() {
        return
_sharedPrefs.getString(KEY_PREFS_USER_BALANCE, "");
    }

    public void setBalance(String text) {
        _prefsEditor.putString(KEY_PREFS_USER_BALANCE, text);
        _prefsEditor.commit();
    }

    // balance Available - баланс пользователя доступный
    public String getBalanceAvailable() {
        return
_sharedPrefs.getString(KEY_PREFS_USER_BALANCE_AVAILABLE, "");
    }

    public void setBalanceAvailable(String text) {
        _prefsEditor.putString(KEY_PREFS_USER_BALANCE_AVAILABLE,
text);
        _prefsEditor.commit();
    }

    // CMS Token
    public String getToken() {
        return _sharedPrefs.getString(KEY_PREFS_CMS_TOKEN,
"");
    }

    public void setToken(String token) {
        _prefsEditor.putString(KEY_PREFS_CMS_TOKEN, token);
        _prefsEditor.commit();
    }

    //=====PROFILE=====
    // UserId
    public String getUserId() {
        return _sharedPrefs.getString(KEY_PREFS_USER_ID, "");
    }

    public void setUserId(String token) {
        _prefsEditor.putString(KEY_PREFS_USER_ID, token);
        _prefsEditor.commit();
    }

    // userEmail
    public String getUserEmail() {
        return _sharedPrefs.getString(KEY_PREFS_USER_EMAIL,
"");
    }
}

```

```

public void setUserEmail(String token) {
    _prefsEditor.putString(KEY_PREFS_USER_EMAIL, token);
    _prefsEditor.commit();
}
// UserPhone
public String getUserPhone() {
    return _sharedPrefs.getString(KEY_PREFS_USER_PHONE,
    "");
}

public void setUserPhone(String token) {
    _prefsEditor.putString(KEY_PREFS_USER_PHONE, token);
    _prefsEditor.commit();
}
// UserLogo
public String getUserLogo() {
    return _sharedPrefs.getString(KEY_PREFS_USER_LOGO, "");
}

public void setUserLogo(String token) {
    _prefsEditor.putString(KEY_PREFS_USER_LOGO, token);
    _prefsEditor.commit();
}
// UserCountryId
public String getUserCountryId() {
    return
    _sharedPrefs.getString(KEY_PREFS_USER_COUNTRY_ID, "");
}

public void setUserCountryId(String token) {
    _prefsEditor.putString(KEY_PREFS_USER_COUNTRY_ID,
    token);
    _prefsEditor.commit();
}
// UserRegionId
public String getUserRegionId() {
    return
    _sharedPrefs.getString(KEY_PREFS_USER_REGION_ID, "");
}

public void setUserRegionId(String token) {
    _prefsEditor.putString(KEY_PREFS_USER_REGION_ID,
    token);
    _prefsEditor.commit();
}
// UserCityId
public String getUserCityId() {
    return _sharedPrefs.getString(KEY_PREFS_USER_CITY_ID,
    "");
}

public void setUserCityId(String token) {
    _prefsEditor.putString(KEY_PREFS_USER_CITY_ID, token);
    _prefsEditor.commit();
}
// UserFirstname
public String getUserFirstname() {
    return
    _sharedPrefs.getString(KEY_PREFS_USER_FIRSTNAME, "");
}

public void setUserFirstname(String token) {
    _prefsEditor.putString(KEY_PREFS_USER_FIRSTNAME,
    token);
    _prefsEditor.commit();
}
// UserLastname
public String getUserLastname() {
    return
    _sharedPrefs.getString(KEY_PREFS_USER_LASTNAME, "");
}

public void setUserLastname(String token) {
    _prefsEditor.putString(KEY_PREFS_USER_LASTNAME,
    token);
    _prefsEditor.commit();
}
// UserMiddlename
public String getUserMiddlename() {
    return
    _sharedPrefs.getString(KEY_PREFS_USER_MIDDLENAME, "");
}

public void setUserMiddlename(String token) {
    _prefsEditor.putString(KEY_PREFS_USER_MIDDLENAME,
    token);
    _prefsEditor.commit();
}
// UserSex
public String getUserSex() {
    return _sharedPrefs.getString(KEY_PREFS_USER_SEX, "");
}

public void setUserSex(String token) {
    _prefsEditor.putString(KEY_PREFS_USER_SEX, token);
    _prefsEditor.commit();
}
// UserMaritalStatus
public String getUserMaritalStatus() {
    return
    _sharedPrefs.getString(KEY_PREFS_USER_MARITAL_STATUS,
    "");
}

public void setUserMaritalStatus(String token) {
    _prefsEditor.putString(KEY_PREFS_USER_MARITAL_STATUS,
    token);
    _prefsEditor.commit();
}
// UserIsAuto
public String getUserIsAuto() {
    return _sharedPrefs.getString(KEY_PREFS_USER_IS_AUTO,
    "");
}

public void setUserIsAuto(String token) {
    _prefsEditor.putString(KEY_PREFS_USER_IS_AUTO, token);
    _prefsEditor.commit();
}
// UserSocialNetwork
public String getUserSocialNetwork() {
    return
    _sharedPrefs.getString(KEY_PREFS_USER_SOCIAL_NETWORK,
    "");
}

public void setUserSocialNetwork(String token) {
    _prefsEditor.putString(KEY_PREFS_USER_SOCIAL_NETWORK,
    token);
    _prefsEditor.commit();
}
// UserMobileOperator
public String getUserMobileOperator() {
    return
    _sharedPrefs.getString(KEY_PREFS_USER_MOBILE_OPERATO
    R, "");
}

public void setUserMobileOperator(String token) {
    _prefsEditor.putString(KEY_PREFS_USER_MOBILE_OPERATO
    R, token);
    _prefsEditor.commit();
}
// UserBirthDay
public String getUserBirthDay() {
    return
    _sharedPrefs.getString(KEY_PREFS_USER_BIRTH_DAY, "");
}

public void setUserBirthDay(String token) {
    _prefsEditor.putString(KEY_PREFS_USER_BIRTH_DAY,
    token);
    _prefsEditor.commit();
}
// UserBirthMonth
public String getUserBirthMonth() {
    return
    _sharedPrefs.getString(KEY_PREFS_USER_BIRTH_MONTH, "");
}

public void setUserBirthMonth(String token) {

```

```

    _prefsEditor.putString(KEY_PREFS_USER_BIRTH_MONTH,
token);
    _prefsEditor.commit();
}
// UserBirthYear
public String getUserBirthYear() {
    return
_sharedPrefs.getString(KEY_PREFS_USER_BIRTH_YEAR, "");
}

public void setUserBirthYear(String token) {
    _prefsEditor.putString(KEY_PREFS_USER_BIRTH_YEAR,
token);
    _prefsEditor.commit();
}
// UserHasChildren
public String getUserHasChildren() {
    return
_sharedPrefs.getString(KEY_PREFS_USER_HAS_CHILDRED,
"");
}

public void setUserHasChildren(String token) {
_prefsEditor.putString(KEY_PREFS_USER_HAS_CHILDRED,
token);
    _prefsEditor.commit();
}
// UserKitchen
public String getUserKitchen() {
    return _sharedPrefs.getString(KEY_PREFS_USER_KITCHEN,
"");
}

public void setUserKitchen(String token) {
    _prefsEditor.putString(KEY_PREFS_USER_KITCHEN,
token);

```

```

    _prefsEditor.commit();
}
// UserInterests
public String getUserInterests() {
    return
_sharedPrefs.getString(KEY_PREFS_USER_INTERESTS, "");
}

public void setUserInterests(String token) {
    _prefsEditor.putString(KEY_PREFS_USER_INTERESTS,
token);
    _prefsEditor.commit();
}
// UserSocialStatus
public String getUserSocialStatus() {
    return
_sharedPrefs.getString(KEY_PREFS_USER_SOCIAL_STATUS,
"");
}

public void setUserSocialStatus(String social_status) {
_prefsEditor.putString(KEY_PREFS_USER_SOCIAL_STATUS,
social_status);
    _prefsEditor.commit();
}
// UserFreqTravel
public String getUserFreqTravel() {
    return
_sharedPrefs.getString(KEY_PREFS_USER_FREQ_TRAVEL, "");
}

public void setUserFreqTravel(String freq_travel) {
    _prefsEditor.putString(KEY_PREFS_USER_FREQ_TRAVEL,
freq_travel);
    _prefsEditor.commit();
}

```

ІЛЮСТРАТИВНА ЧАСТИНА

РОЗРОБКА АВТОМАТИЗОВАНОЇ СИСТЕМИ ОБЛІКУ СКЛАДСЬКИХ
ТОВАРІВ І ПІДТРИМКИ ТОРГОВЕЛЬНОЇ ДІЯЛЬНОСТІ. ЧАСТИНА 2.
РОЗРОБКА МОБІЛЬНОГО ЗАСТОСУНКУ.

Перелік ілюстративних матеріалів:

1. Слайд 1 — Титульний слайд
2. Слайд 2 — Актуальність і мета дослідження
3. Слайд 3 — Концепція рішення
4. Слайд 4 — Структурна схема бази даних
5. Слайд 5 — UML-діаграма варіантів використання
6. Слайд 6 — UML-діаграма компонентів
7. Слайд 7 — Середовище розробки
8. Слайд 8 — Реалізація та інтерфейс програмного забезпечення
9. Слайд 9 — Процес додавання даних про нового покупця
10. Слайд 10 — Висновки
11. Слайд 11 — Фінальний слайд

Керівник роботи:

д.т.н., професор кафедри КСУ



Марія ЮХИМЧУК

Виконавець:

Ст. гр. 1АКІТР-24м



Катерина СЕРЕДА

РОЗРОБКА АВТОМАТИЗОВАНОЇ СИСТЕМИ ОБЛІКУ СКЛАДСЬКИХ ТОВАРІВ І ПІДТРИМКИ ТОРГОВЕЛЬНОЇ ДІЯЛЬНОСТІ. ЧАСТИНА 2. РОЗРОБКА МОБІЛЬНОГО ЗАСТОСУНКУ

Студент 2 курсу, групи 1АКІТР-24м спеціальності 174 – Автоматизація та комп'ютерно-інтегровані технології та роботехніка Катерина СЕРЕДА

Керівник: д.т.н., професор кафедри КСУ Марія ЮХИМЧУК

Слайд 1 — Титульний слайд

- **Актуальність теми.** Автоматизація бізнес-процесів є ключовим чинником успішного розвитку підприємств. Із розширенням компанії зростає потреба у контролі великої кількості працівників і швидкому аналізі даних, що й обумовлює необхідність впровадження систем обліку та іншого програмного забезпечення [1]. Наразі автоматизація охоплює не лише великі компанії, але й середній та малий бізнес, де конкурентні переваги здобувають ті, хто впроваджує ефективні інструменти.
- **Мета і завдання дослідження.** Метою роботи є підвищення ефективності торгової мережі шляхом розробки мобільного додатку для автоматизованого обліку складських товарів і торгівлі. Це дозволить мінімізувати людський фактор та ефективно використовувати ресурси.
- **Завдання:**
 - Аналіз існуючих рішень у сфері обліку і торгівлі;
 - Дослідження технологій реалізації автоматизованих систем;
 - Вивчення практичних аспектів функціонування систем обліку;
 - Розробка мобільного додатку в режимі реального часу;
 - Проведення досліджень і тестування результатів.
- **Об'єкт дослідження:** процес складського обліку додатку.
- **Предмет дослідження:** методи і засоби програмування для реалізації складського обліку.
- **Методи дослідження:** алгоритмічна теорія, Agile-підхід, Scrum, мови HTML, JavaScript і CSS.

Слайд 2 — Актуальність і мета дослідження

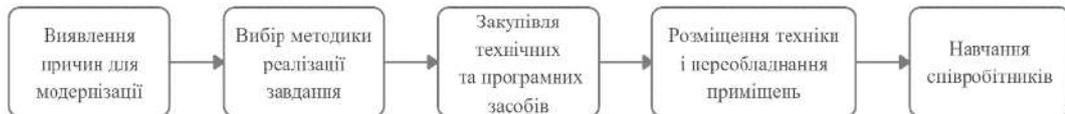
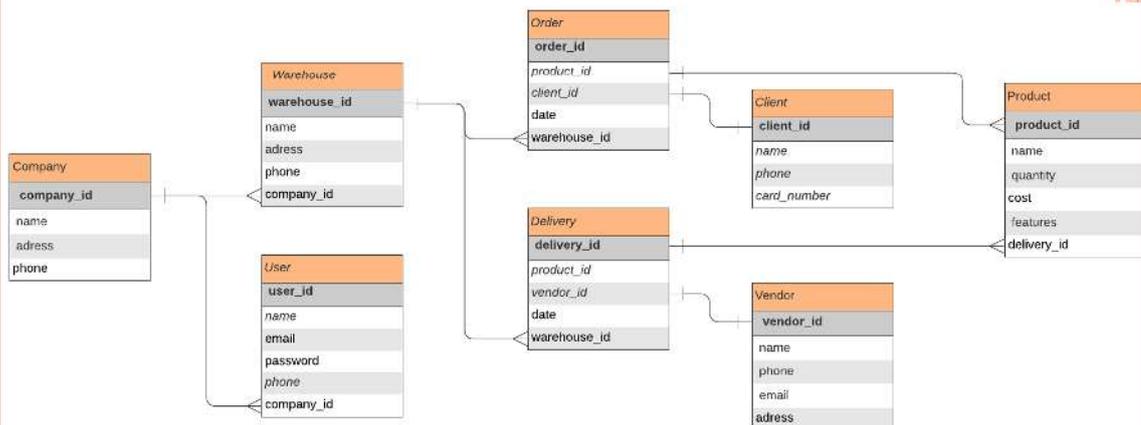


Рисунок 1.1 – Етапи переходу на технологію електронної форми документації

Дослідивши тенденції розвитку сучасного світу програмних продуктів для автоматизації бізнес-процесів, зокрема управління складськими товарами, поставлено за мету розробити власний мобільний додаток для обліку складських товарів та торгівлі, що буде забезпечувати безперебійний доступ до бази даних товарів компанії та управління ними, а також інтеграцію сучасних функцій для комунікації та відстеження документообігу, що прискорить процес виконання рутинних завдань співробітниками компанії, а також сприятиме зменшенню витрат та ефективного розподілу вивільнених робочих ресурсів на інші області роботи компанії

Слайд 3 — Концепція рішення

Продублюємо структурну схему бази даних, що роз'яснює зв'язки між сутностями та містить перелік атрибутів, необхідних для утворення цих зв'язків



Структурна схема бази даних системи

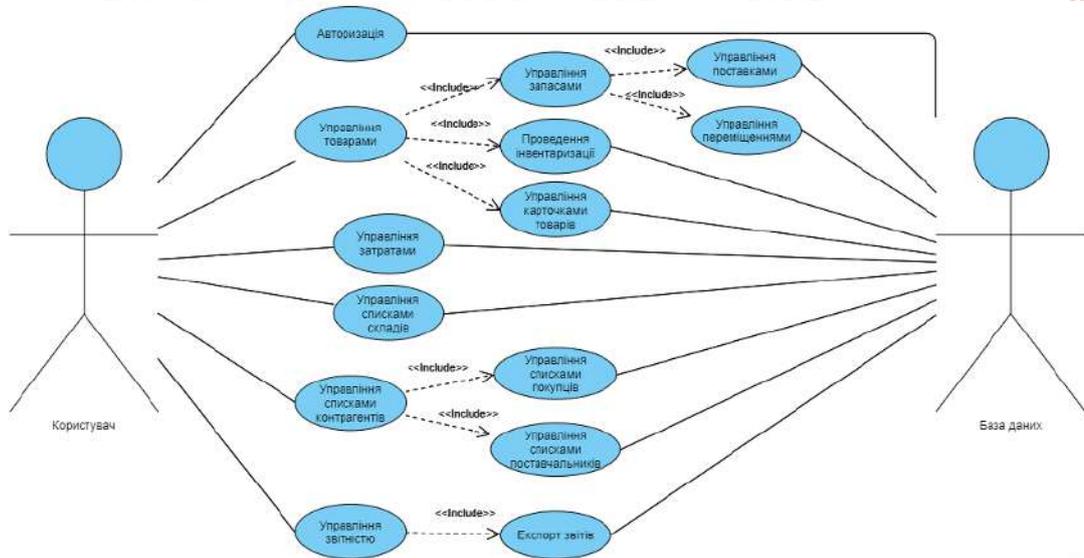
Слайд 4 — Структурна схема бази даних

Для користувача передбачено наступні прецеденти по управлінню:

- Товарами;
- Затратами.
- Списками складів
- Списками контрагентів.
- Звітністю.

Кожен із зазначених вище варіантів використання передбачає звернення до Бази даних, тому вона на діаграмі винесена окремим актором, що обробляє запити по перегляду, створенню та редагуванню інформації в системі.

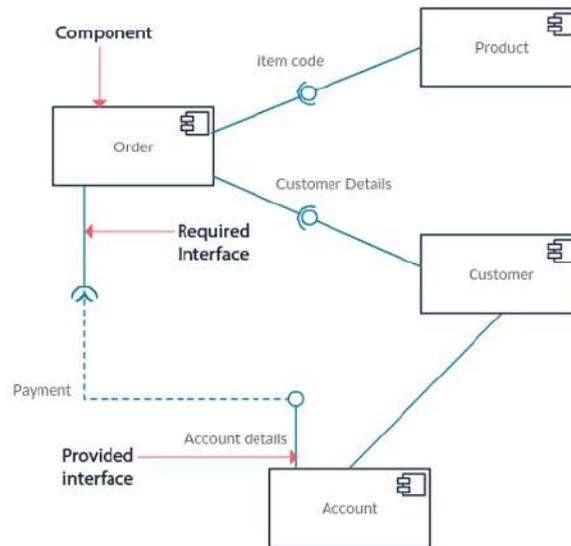
UML-діаграма варіантів використання системи, що розробляється, представлена на рисунку



UML-діаграма варіантів використання

Слайд 5 — UML-діаграма варіантів використання

Діаграма компонентів відображає структурні взаємозв'язки компонентів програмного рішення, та, в основному, використовується при роботі зі складними системами з багатьма компонентами. Компоненти спілкуються між собою за допомогою інтерфейсів, а інтерфейси пов'язані за допомогою роз'ємів. На рисунку UML-діаграма компонентів, автоматизованої системи що розробляється.



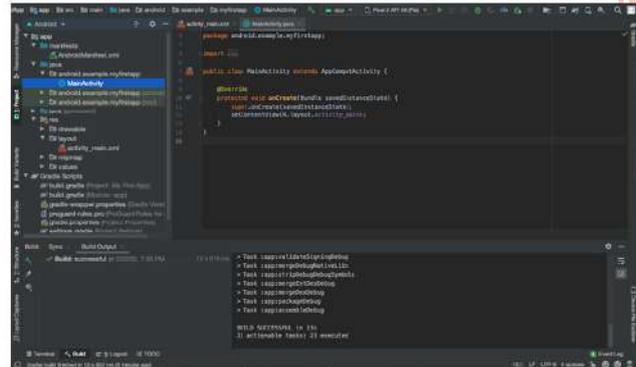
UML-діаграма компонентів

Слайд 6 — UML-діаграма компонентів

Для створення мобільного додатку автоматизованої системи спочатку необхідно було завантажити останню версію Android Studio та обрати Створення нового проекту, що зображено на рисунку



Стартове вікно середовища програмування Android Studio

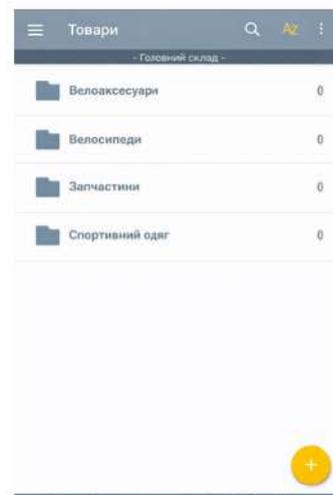


Головне вікно зі створеним новим додатком

Слайд 7 — Середовище розробки



Головний екран мобільного додатку автоматизованої системи обліку складських товарів та підтримки торгівлі



Створені групи товарів

Слайд 8 — Реалізація та інтерфейс програмного забезпечення

← Додати покупця ✓

Фещук Микола Іванович

Адреса
Стрижівка

Ел. пошта
testt1@yg.com

Телефон
0665473008

Прок. код

Зв'язок(%)
0.00

Зв'язок(%)
0.00

Процес додавання даних про
нового покупця

Дата	Тип	Контр. партія	Вид	Вартість	Сума	Сума
27.11.20	Група	Вексел	Вексел	Вексел	0	41820
27.11.20	Група	Вексел	Вексел	Вексел	30	300
28.11.20	Група	Вексел	Вексел	Вексел	5766367	40
28.11.20	Група	Вексел	Вексел	Вексел	10	3660
28.11.20	Група	Вексел	Вексел	Вексел	1	1000
28.11.20	Група	Вексел	Вексел	Вексел	2	100
28.11.20	Група	Вексел	Вексел	Вексел	5766367	1
28.11.20	Група	Вексел	Вексел	Вексел	0	26
28.11.20	Група	Вексел	Вексел	Вексел	0	47470
28.11.20	Група	Вексел	Вексел	Вексел	0	10000

Журнал транзакцій За період...

Звіт, що
збережений в
документ Excel

Слайд 9 — Процес додавання даних про нового покупця

ВИСНОВКИ

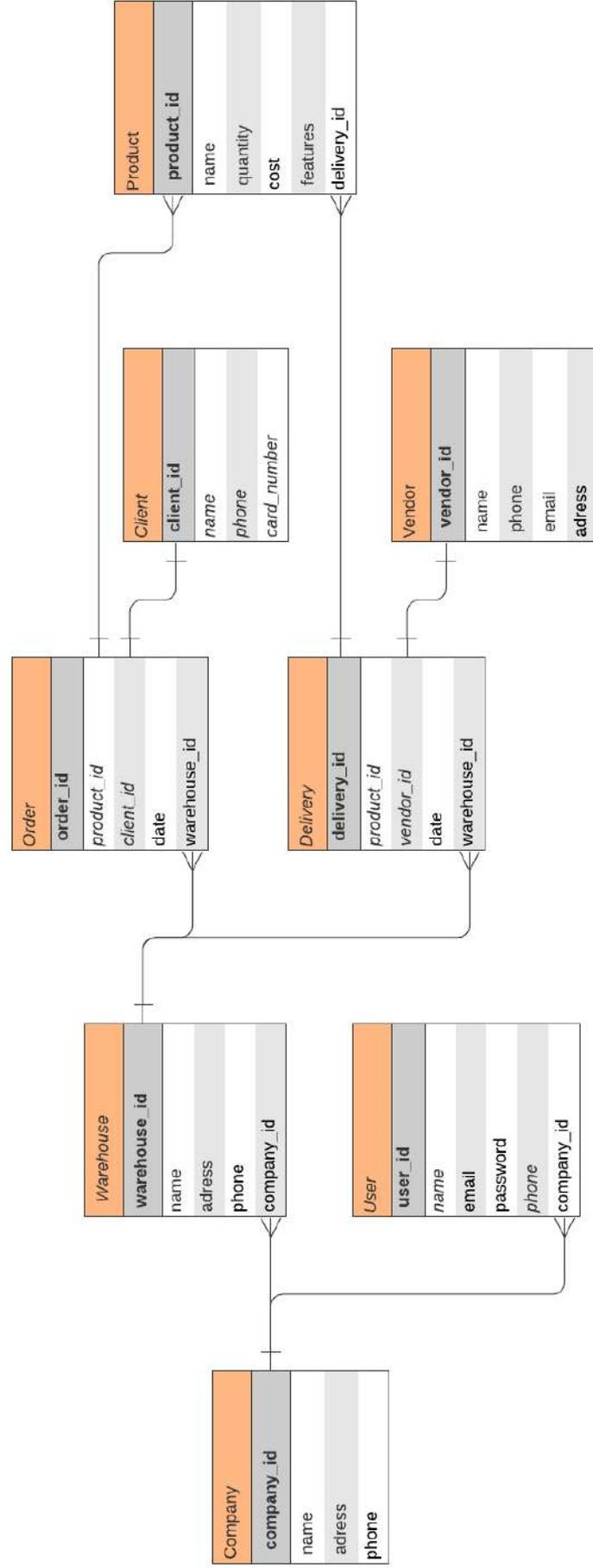
- У комплексній магістерській кваліфікаційній роботі підтверджено, що автоматизація бізнес-процесів є ключовим чинником підвищення ефективності підприємств будь-якого масштабу. Застосування систем обліку і спеціалізованого програмного забезпечення дозволяє не лише скоротити витрати часу на рутинні операції, а й зменшити вплив людського фактора, підвищуючи точність ведення обліку та оперативність прийняття рішень. Аналіз існуючих проблем у сфері складського обліку, таких як накопичення нереалізованих товарів, неточності інвентаризації та затримки в закупівлях, показав, що без автоматизації ефективного управління торговельною діяльністю є складним, особливо при масштабуванні бізнесу.
- Виконане дослідження дозволило визначити ключові функції мобільного додатку для автоматизованого обліку товарів, розробити структуру бази даних, яка забезпечує збереження та обробку інформації у режимі реального часу, та створити програмний комплекс, що пройшов тестування і підтвердив правильність своєї роботи. Використання Agile-підходу та методів Scrum дозволило забезпечити гнучкість розробки та ефективне вирішення завдань на кожному етапі проекту.
- Практична реалізація мобільного додатку демонструє його здатність оптимізувати управлінські процеси торговельної мережі, підвищити продуктивність персоналу та забезпечити контроль над товарами у всіх філіях підприємства. Результати роботи мають практичне значення для підприємств, що прагнуть автоматизувати складський облік і підтримку торговельної діяльності, а сам підхід до розробки може бути використаний у подальших дослідженнях і проектах у сфері інформаційних систем.

Слайд 10 — Висновки

- Дякую за увагу, доповідь завершено!



Слайд 11 — Фінальний слайд



Структурна схема бази даних СИСТЕМИ

UML-діаграма компонентів

