

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ХАРЧОВИХ ТЕХНОЛОГІЙ**

Інститут (факультет) Автоматизації і комп'ютерних систем
Кафедра Інформаційних систем

«До захисту в ЕК»
Директор інституту(декан факультету)
Форсюк А.В.
(підпис) (прізвище та ініціали)

« ___ » _____ 2021р.

«До захисту допущено»
Завідувач кафедри
Чумаченко С. М.
(підпис) (прізвище та ініціали)

« ___ » _____ 2021р.

**КВАЛІФІКАЦІЙНА РОБОТА
НА ЗДОБУТТЯ ОСВІТНЬОГО СТУПЕНЯ БАКАЛАВРА**

зі спеціальності 122 «Комп'ютерні науки»
(код та назва спеціальності)
освітньо-професійної програми «Комп'ютерні науки»

на тему: «Розробка інформаційної системи роботи магазину роздрібної торгівлі»

Виконав: здобувач 5 курсу, групи 2

Шилюк Ольга Ігорівна
(прізвище, ім'я, по батькові повністю)

_____ (підпис)

Керівник доц. Бойко Регіна Олегівна
(прізвище, ім'я, по батькові повністю)

_____ (підпис)

Консультанти _____
(прізвище та ініціали)

_____ (підпис)

_____ (прізвище та ініціали)

_____ (підпис)

_____ (прізвище та ініціали)

_____ (підпис)

Рецензент проф. Ладанюк Анатолій Петрович
(прізвище та ініціали)

_____ (підпис)

Засвідчую, що в цій кваліфікаційній
роботі немає запозичень із праць
інших авторів без відповідних
посилань.

Здобувач _____
(підпис)

Київ - 2021р.

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ХАРЧОВИХ ТЕХНОЛОГІЙ

Інститут (факультет) Автоматизації і комп'ютерних систем

Кафедра Інформаційних систем

Освітній ступінь бакалавр

Спеціальність _____

Освітньо-професійна програма «Комп'ютерні науки»
(код і назва)
(назва)

ЗАТВЕРДЖУЮ

Завідувач кафедри

інформаційних систем

с.н.с. С. М. Чумаченко

“ _____ ” _____ 20__ року

З А В Д А Н Н Я

НА КВАЛІФІКАЦІЙНУ РОБОТУ ЗДОБУВАЧА

Шилюк Ольга Ігорівна

(прізвище, ім'я, по батькові)

1. Тема роботи «Розробка інформаційної системи роботи магазину роздрібної торгівлі»

керівник роботи Бойко Регіна Олегівна, доцент

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затвержені наказом закладу вищої освіти від “9” листопада 2020 року №932-кс

2. Строк подання здобувачем роботи 01.02.2021

3. Вихідні дані до роботи 1) замовлення

2) замовники

3) товар

4. Зміст пояснювальної записки (перелік питань, які потрібно розробити)

1) Вступ

2) Розділ 1. Системний аналіз об'єкту автоматизації та постановка задачі на проектування.

3) Розділ 2. Технічне завдання на проектування.

4) Розділ 3. Опис комплексу задач автоматизації.

5) Розділ 4. Охорона праці.

б) Висновки.

5. Перелік графічного матеріалу

1. Функціональна модель діяльності відділу управління товарів _____

2. Концептуальна модель діяльності відділу управління товарів _____

3. Фізична схема бази даних _____

4. Схеми згенерованих баз даних _____

5. Інтерфейс користувача _____

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Розділ 1	Доц. Бойко Р.О.		
Розділ 2	Доц. Бойко Р.О.		
Розділ 3	Доц. Бойко Р.О.		
Розділ 4	Доц. Бойко Р.О.		

7. Дата видачі завдання 9 листопада 2020 року

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів виконання кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1	Предпроектне дослідження та системний аналіз діяльності відділу управління товарів	15.11.2020	
2	Розробка функціональної та концептуальної моделей діяльності відділу управління товарів	30.11.2020	
3	Визначення та реалізація функцій підсистеми	05.12.2020	
4	Розробка плану організації праці	25.01.2021	
5	Оформлення роботи та розробка презентації	01.02.2021	

Здобувач _____

(підпис)

_____ (прізвище та ініціали)

Керівник роботи _____

(підпис)

_____ (прізвище та ініціали)

АНОТАЦІЯ

Кваліфікаційна робота «Розробка інформаційної системи роботи магазину роздрібної торгівлі» розроблена Шилюк О. І., складається з 64 сторінки, 1 таблиця, 36 рисунків, 4 додатки, 18 літературних джерел.

Метою цієї кваліфікаційної роботи є розробка та створення інформаційної системи для клієнт-менеджера компанії «Матролікс».

ІС надаватиме інформацію про поставки та залишок товарів, а також про співробітників, товари загалом та постачальників з якими співпрацює компанія. Також була розроблена інформаційна система для клієнта компанії, що дає змогу комфортне оформлення замовлення, зручний каталог асортименту та змогу з'єднатись з менеджером.

КЛЮЧОВІ СЛОВА: КЛІЄНТ-МЕНЕДЖЕР, БАЗА ДАНИХ, КЛІЄНТ, ОХОРОНА ПРАЦІ, ІНФОРМАЦІЙНА СИСТЕМА

SUMMARY

Qualification work "Development of an information system of the retail store" developed by O. I. Shyliuk, consists of 64 pages, 1 table, 36 drawings, 4 application, 18 literary sources.

The purpose of this qualification work is to develop and create an information system for the customer service manager of the «Matrolix» store.

The IS will provide information on deliveries and balances of goods, as well as on employees, goods in general and suppliers with whom the store cooperates. An information system developed for the customer of the store, which allows for comfortable ordering, a convenient catalog of the range and the ability to connect with the manager.

KEY WORDS: CLIENT MANAGER, DATABASE, CLIENT, LABOR PROTECTION, INFORMATION SYSTEM

ЗМІСТ

ВСТУП	7
РОЗДІЛ 1. СИСТЕМНИЙ АНАЛІЗ ОБ'ЄКТУ АВТОМАТИЗАЦІЇ ТА ПОСТАНОВКА ЗАДАЧ НА ПРОЕКТУВАННЯ	8
1.1 Загальна характеристика компанії «Матролікс»	8
1.2 Організаційна структура компанії, роль і взаємодія підрозділів.....	9
1.2.1 Загальна схема організаційної структури	9
1.2.2. Структура відділу управління товару.....	10
1.3. Аналіз нинішнього стану комп'ютеризації	12
1.4. Розроблення функціональної моделі та аналіз існуючих бізнес-процесів	13
1.4.1. Розробка функціональної моделі.....	13
1.4.2. Виявлені проблеми.....	13
1.5. Огляд існуючих рішень для розв'язання виявлених проблем	14
1.5.1. RetailCRM	14
1.5.2. 1С:УНФ	16
1.5.3. MS SQL Server 2008, Telegram	17
1.6. Обґрунтування доцільності проектування й розроблення системи	20
1.7. Концептуальна модель системи	20
РОЗДІЛ 2. ТЕХНІЧНЕ ЗАВДАННЯ НА ПРОЕКТУВАННЯ	21
2.1. Постановка задачі на проектування.....	21
2.2. Призначення та цілі створення системи.....	22
2.3. Вимоги до функцій та системи в цілому	22
РОЗДІЛ 3. ОПИС КОМПЛЕКСУ ЗАДАЧ АВТОМАТИЗАЦІЇ	24
3.1. Інформаційне забезпечення системи	24
3.2. Алгоритмізація та реалізація комплексу задач автоматизації.....	26
3.3. Інструкція користувача	39
3.4 Технічне та системне забезпечення розробки	48
3.4.1 Обґрунтування вибору технічних засобів.....	48
РОЗДІЛ 4. ОХОРОНА ПРАЦІ	49
4.1 Заходи пожежної безпеки у службових приміщеннях	49
4.2 Вимоги до робочого місця офісного працівника	50
4.2.1 Безпека під час роботи з персональним комп'ютером.....	52
ВИСНОВКИ	53

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	54
ДОДАТКИ.....	56
ДОДАТОК А «ФУНКЦІОНАЛЬНА МОДЕЛЬ (AS-IS)».....	56
ДОДАТОК Б «ФУНКЦІОНАЛЬНА МОДЕЛЬ(TO BE)».....	57
ДОДАТОК В «ФІЗИЧНА СХЕМА БАЗИ ДАНИХ».....	58
ДОДАТОК Д «ФРАГМЕНТИ КОДУ ПРОГРАМИ»	59

ВСТУП

Створення сучасних інформаційних систем являє собою найскладнішу задачу, вирішення якої вимагає застосування спеціальних методик та інструментів.

Не дивно, що останнім часом серед системних аналітиків і розробників значно збільшився інтерес до CASE (Computer-Aided Software / System Engineering) - технологій і інструментальних CASE-засобів, що дозволяє максимально систематизувати і автоматизувати всі етапи розробки програмного забезпечення.

Інформаційна система — сукупність організаційних і технічних засобів для збереження та обробки інформації з метою забезпечення інформаційних потреб користувачів.

Метою даної кваліфікаційної роботи є створення інформаційної системи для магазину компанії роздрібною торгівлі “Матролікс”.

Завданням даної роботи є:

- аналіз діяльності та вивчення специфіки роботи магазину роздрібною торгівлі “Матролікс”;
- аналіз існуючого бізнес-процесу та обґрунтування доцільності розробки інформаційної системи;
- визначення основних вимог до створюваної системи, функцій, які вона повинна виконувати, та їх значення у постановці задачі на розроблення системи.

РОЗДІЛ 1. СИСТЕМНИЙ АНАЛІЗ ОБ'ЄКТУ АВТОМАТИЗАЦІЇ ТА ПОСТАНОВКА ЗАДАЧІ НА ПРОЕКТУВАННЯ

1.1 Загальна характеристика компанії «Матролікс»

Компанія «Матролікс» пропонує в широкому асортименті непродовольчі товари. Матролікс пропонує широкий асортимент товарів декоративної косметики, товари для догляду за обличчям і тілом, біжутерію, аксесуари, побутові товари, а також товари для догляду за дітьми. У певні дні магазин пропонує покупцям зробити покупки і разом допомогти новонародженим дітям. Частина суми від продажу перераховується на закупівлю медичного обладнання для пренатальних центрів України. В результаті проведення акцій проекту закуплено і передано медичне обладнання пренатальним центрам на суму близько 1 млн грн. Всі представлені товари характеризуються високою якістю, тому кожен покупець може бути впевнений в належному стані куплених предметів. «Матролікс» займається роздрібною торгівлею з 2018 року, штат компанії налічує до 50 осіб, та на протязі 3х років втілює маленькі мрії та грандіозні плани мільйонів людей. У нас можна знайти буквально все. Продає за справедливою ціною та надає гарантію, бо вважає, що онлайн-шопінг має бути максимально зручним і безпечним. Вірить, що речі існують для того, щоб робити життя простішим, приємнішим і добрішим. Тому й пошук тієї самої речі повинен бути швидким, зручним і приємним. Не просто продає товари для догляду за обличчям, прикраси або аксесуари. Допомагає знайти саме те, що треба, в одному місці та без зайвих хвилювань, щоб ви не витрачали життя на пошуки, а просто жили щасливо. «Матролікс» – це універсальна відповідь на будь-який запит, початок пошуку та його кінцева зупинка, справжній помічник. Назавжди позбавляє своїх покупців від неприємних компромісів, виконує бажання і дає змогу мріяти сміливіше. Завдяки розумному пошуку та чесному сервісу ми робимо життя наших клієнтів трішки кращим просто зараз.

«Матролікс» - це інтернет-магазин, офіс та склад м. Київ і співробітники, які вільно орієнтуються в асортименті з 2000 одиниць з багатьох країн світу.

Більшість товару закуповується в Європі, також компанія співпрацює з українськими постачальниками.

1.2 Організаційна структура компанії, роль і взаємодія підрозділів

1.2.1 Загальна схема організаційної структури

Лінійна структура управління підприємством складається із взаємно підпорядкованих органів у вигляді ієрархічної драбини. Кожен управлінець підпорядковується тільки одному керівнику і всі зв'язки з вищими рівнями управління відбуваються через нього. Організаційна структура компанії зображена на рис. 1.2.1.

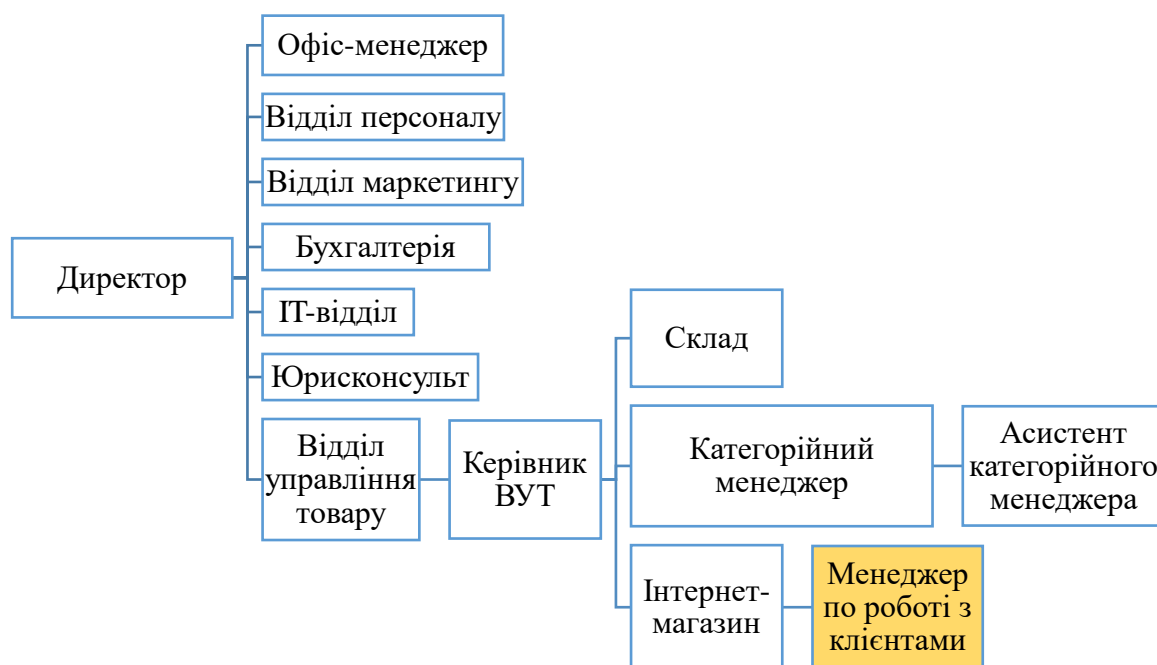


Рис. 1.2.1. Організаційна структура

Керівник відділу управління товару розподіляє обов'язки між співробітниками відділу і затверджує їхні посадові інструкції.

Технологічні процеси пов'язані з логістикою. Під цим поняттям мають на увазі всі операції, що виконуються під час перевезення вантажу (транспортування, розвантаження, зберігання, фасування, упаковка). Ці операції – продовження виробничого процесу, і безпосереднє рух транспорту. Комерційні операції являють собою всі процеси, так чи інакше пов'язані з купівлею-продажом. У цей список також входять організаційно-господарські процеси. Не можна сказати,

що вони безпосередньо пов'язані з купівлею продажом, але ці процеси безумовно впливають на систематизацію торгового потоку.

1.2.2. Структура відділу управління товару

Основні функції відділу управління товару

- Вивчення попиту на товар, його прогноз. Дослідження споживчого попиту на певні групи товарів;
- Пошук і виявлення постачальників;
- Всі дії, пов'язані з формуванням асортименту товарів;
- Управління асортиментом;
- Економічне обґрунтування вибору того чи іншого постачальника;
- Організація зав'язків з постачальниками;
- Організація сервісного обслуговування;
- Укладання та розірвання договорів, вся робота з документацією;
- Вибір маркетингових прийомів для продажу товарів;
- Застосування маркетингу в соціальних мережах, реклама в інтернеті тощо;
- Оцінка та вивчення власної діяльності.

Взаємодія з іншими відділами

Відділ управління товару взаємодіє з усіма іншими відділами в компанії. На склад він передає інформацію про найближчі надходження товарів, отримує від них інформацію щодо надлишків та недостач в прийнятому надходженні. В бухгалтерію передає рахунки на оплату від постачальників. ІТ-відділу передає інформацію про несправність офісної техніки, вони її налагоджують.

Функціональні обов'язки клієнт-менеджера

- Здійснює аналіз аудиторії потенційних клієнтів, виявляє потреби клієнтів, їх рівень і спрямованість;
- Розробляє методики пошуку клієнтів, планує роботу з клієнтами, складає схеми звернення до клієнтів;
- Безпосередньо здійснює пошук клієнтів всіма доступними способами (шляхом розміщення реклами, участі у виставках, ярмарках, презентаціях, направлення

пропозицій по засобам комунікацій, електронною поштою, смс, інстаграм, телеграм тощо);

- Прогнозує фінансову і матеріальну забезпеченість потенційних клієнтів;
- Проводить попередні переговори з клієнтами, що зацікавилися пропозиціями;
- Уточнює потреби кожного конкретного клієнта і готує пропозицію;
- Переконає клієнтів у вигідності пропозиції, пропонує на обговорення і узгодження товари;
- Пропонує клієнтам шляхи рішення не узгоджених питань, що виникли після здійснення покупки;
- Підтримує контакт з існуючими клієнтами, організовує роботу з ними;
- Розробляє схеми взаємин з найбільш вигідними і перспективними клієнтами (пропозиції особливих умов, систем знижок і індивідуального обслуговування, прискорених термінів і особливих умов виконання замовлень);
- Розробляє і дає клієнтам рекомендації та консультації щодо найбільш ефективного використання товарів; забезпечує можливість відвідування клієнтами виставок, ярмарків, презентацій нових товарів;
- Налагоджує зворотній зв'язок з клієнтами (вивчає їхні вимоги до продукції, встановлює причини незадоволеності клієнта спільною роботою, аналізує претензії клієнтів і вживає всіх заходів щодо їх вирішення та збереження репутації компанії);
- Формує базу даних клієнтів, своєчасно вносить до неї зміни;
- Вивчає і аналізує політику конкурентів у взаєминах з клієнтами;
- Контролює фінансові надходження від клієнтів в компанію та відправлення замовлень до клієнтів.

Аналіз включає в себе декілька важливих критеріїв в роботі менеджера по роботі з клієнтами:

- **Результативність.** Показник розраховується як відношення обсягу реальних продажів до того обсягу, який було заявлено планом;

- Робота з цільовою аудиторією. Тут варто оцінити ставлення нових клієнтів до тих, кого компанія втратила, а також число тих, хто став постійним клієнтом;
- Робота з фінансовими надходженнями. Мається на увазі діяльність по ефективному погашенню дебіторської заборгованості. Слід оцінити, наскільки швидко клієнти проплачують свої замовлення і як менеджер впливає на це;
- Наскільки добре менеджер знає свою компанію, її асортимент;
- Чи знає менеджер портрет потенційного клієнта, конкурентів, нішу на ринку та інші маркетингові показники;
- Чи володіє менеджер технікою продажів, а також наскільки він комунікабельний;
- Рівень професійної підготовки;
- Вмотивованість і залученість в загальну справу;
- Відданість традиціям компанії і корпоративну культуру в цілому;
- Постійне вдосконалення своїх навичок, а також безперервне вивчення ринку;
- Щоденне спілкування з клієнтами і його ефективність.

1.3. Аналіз нинішнього стану комп'ютеризації

Компанія займається торгівлею непродовольчих товарів, тому проаналізувавши стан програмного забезпечення яке використовує відділ управління товарів можна сказати, що воно не є надійним і недостатньо автоматизованим. Програмне забезпечення не має стабільного збереження даних, тому працівники повинні більшу частину інформації, що надійшла зберігати в інших, таких як Excel з пакету програм Microsoft Office. Для виправлення помилок в роботі програмного забезпечення використовується TeamViewer, також всі комп'ютери захищені від вірусів за допомогою антивірусної системи ESET NOD32.

1.4. Розроблення функціональної моделі та аналіз існуючих бізнес-процесів

1.4.1. Розробка функціональної моделі

Розроблена функціональна модель для відділу управління товару компанії «Матролікс» яка відображає суть процесів.

Входи моделі:

- Товар на складі;
- Візит клієнта;
- Заявка на онлайн покупку від клієнта.

Виходи із моделі:

- Рахунки на оплату;
- Замовлення на виготовлення товару;
- Проданий товар;
- Замовлення нового товару.

Управління моделі:

- Права і закони України;
- Вказівки.

Механізми:

- Інформаційна система;
- Комірник;
- Водій;
- Менеджери.

Контекстна діаграма має один рівень декомпозиції, та наведена на рис.1 у Додатку А.

1.4.2. Виявлені проблеми

При аналізі роботи нинішньої інформаційної системи було встановлено, що вона не має стабільного збереження даних, відсутня можливість формувати звіти та здійснювати пошук, фільтрації, вибірки з бази даних. Менеджер повинен це

все робити сам, через це втрачає багато часу на дії, які могли би бути автоматизовані.

1.5. Огляд існуючих рішень для розв'язання виявлених проблем

1.5.1. RetailCRM

RetailCRM – це CRM-система для управління інтернет-магазином в режимі «одного вікна». За допомогою цієї системи можна автоматизувати процес продажів:

- Отримувати офлайн і онлайн замовлення в єдину базу;
- Агрегувати комунікації з клієнтами з усіх каналів;
- Комплектувати замовлення, управляти переміщенням товарів між складами і магазинами;
- Виставляти рахунки по смс і пошті, фіскалізовані платежі по ФП, зберігати чеки оплат і повернень;
- Розраховувати вартість доставки, відправляти заявки в службу доставки, оповіщати клієнтів про місцезнаходження замовлення в автоматичному режимі, управляти власними кур'єрами, друкувати наклейки і супроводжуючі документи;
- Прискорювати обробку замовлень і знижувати кількість помилок за рахунок повної автоматизації бізнес-процесів.

Використовувати CRM-маркетинг для:

- Сегментації клієнтів з поведінки, різними показниками і призначенням для користувача полях;
- Повторних продажів на автопілоті за допомогою інтелектуального помічника ActionAssistant;
- Email та смс розсилок;
- Відстеження результатів маркетингу.

Аналізувати результати в розрізі:

- Замовлень;

- Клієнтів;
- Товарів;
- Комунікацій;
- Фінансів.

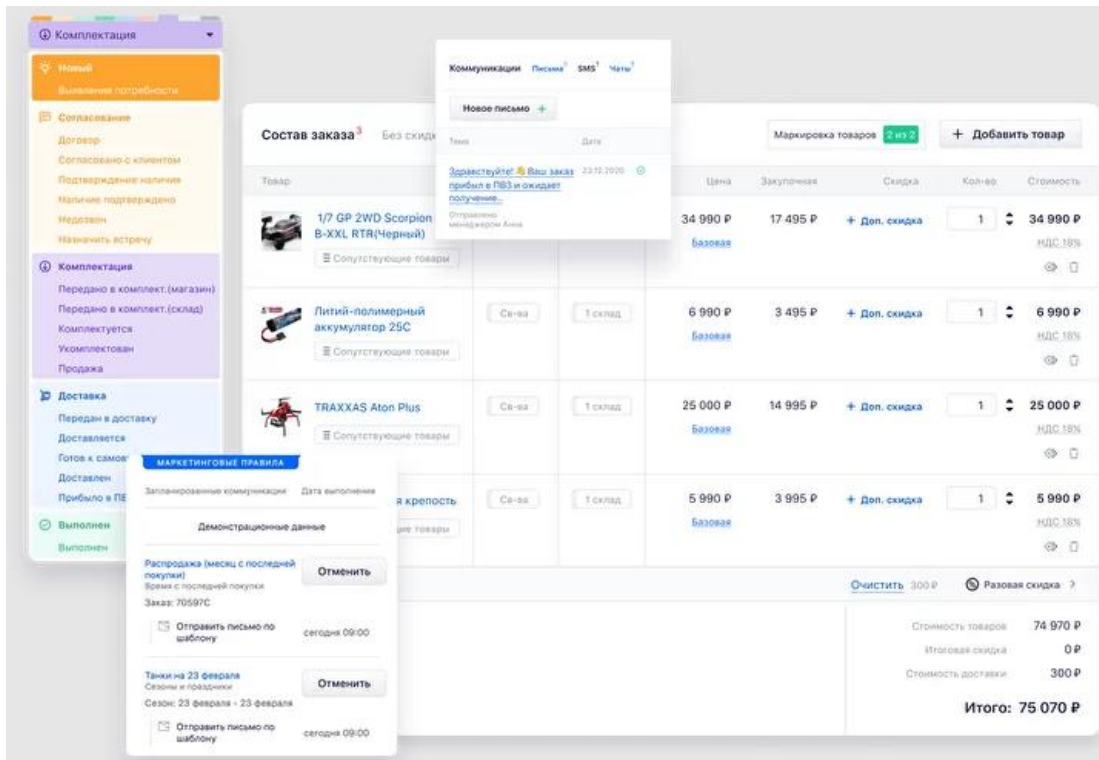


Рис.1.5.1. Вікно замовлення програми RetailCRM

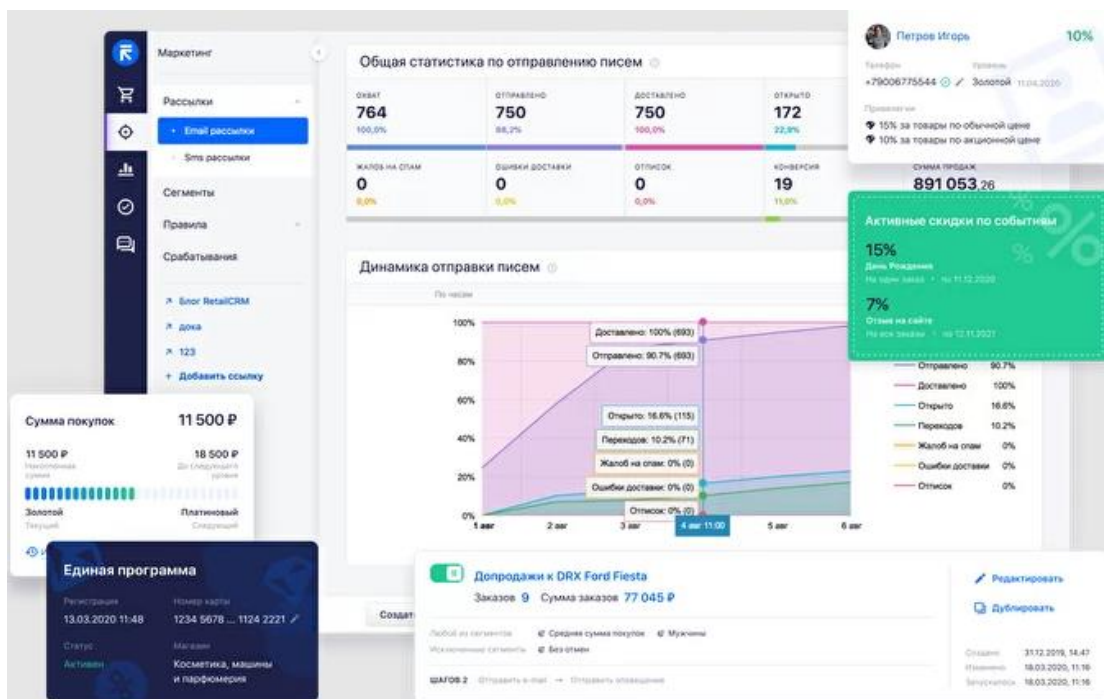


Рис.1.5.2. Вікно розділу маркетингу програми RetailCRM

1.5.2. 1С:УНФ

Інтернет-торгівля в 1С:УНФ – повноцінна і ефективна робота з інтернет-магазином. Система дозволяє організувати діяльність інтернет-магазину, а саме:

- Автоматизацію процесу продажів з використанням відповідної схеми торгівлі;
- Детальне планування продажів в натуральному і вартісному вираженні;
- Ведення кількісно-сумового або сумового обліку;
- Формування всіх потрібних звітів;
- Підключення торгового обладнання;
- Автоматичне резервування товарів при завантаженні замовлення з інтернет-магазину;
- Автоматична відправка чека клієнтові при покупці в інтернет-магазині;
- Детальна картка клієнта з повними даними і можливістю моніторингу співробітництва;
- Воронка продажів: кількість замовлень на кожному етапі, середній час проходження стану, сума замовлень, конверсія;
- Управління відділом продажів з докладною налаштуванням автоматизації робочих процесів;
- Аналіз роботи менеджерів.

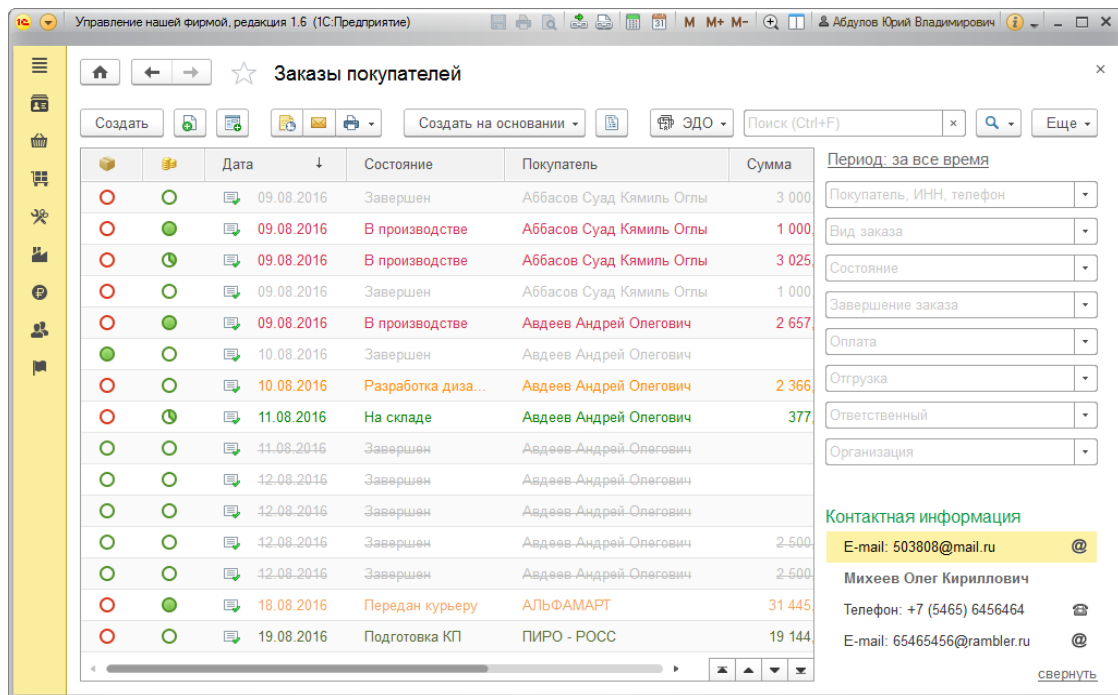


Рис.1.5.3. Вікно замовлень програми 1С:УНФ

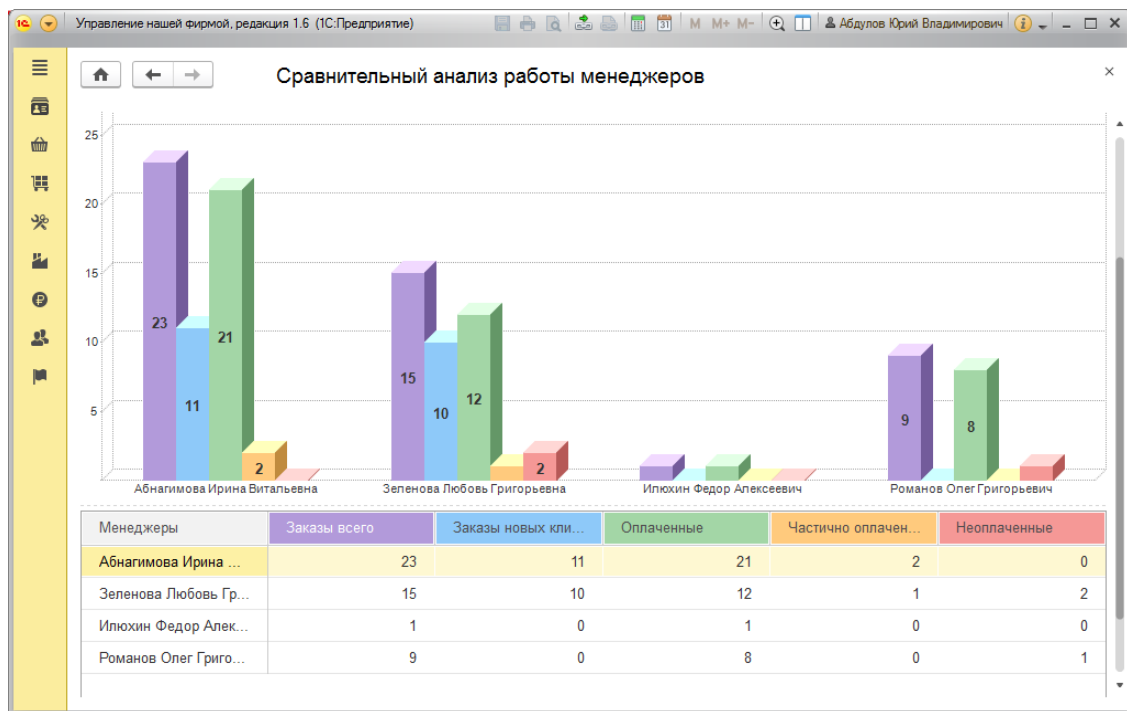


Рис.1.5.4. Вікно аналізу роботи менеджерів програми 1С:УНФ

1.5.3. MS SQL Server 2008, Telegram

База даних в SQL Server складається з колекції таблиць, в якій зберігається особливий набір структурованих даних. Таблиця містить колекцію рядків, так званих записів або кортежів і стовпців, також званих атрибутами. Кожен стовпець у таблиці призначений для зберігання конкретного типу даних,

наприклад дат, імен, грошових сум або чисел. Обов'язками користувача, яким може бути менеджер, робітник складу або керівник магазину, є занесення до БД інформації про продажі, надходження товару та залишок товару в наявності. Інформаційна система виконує такі функції:

- Збереження даних про товару, надходження, залишок товару та співробітників магазину.
- Здійснення пошуку, фільтрації, вибірки з БД.
- Формування звітів.

Telegram bot – це програма, яка працює в месенджері Telegram і часто імітує живе спілкування з користувачем в чаті. Чат-бот запускає різні варіанти встановлених ланцюжків у відповідь на натискання певних кнопок і на введення слів в чаті. Чат-бот може запитувати інформацію від користувача – номер телефону, пошту, місто, день народження та інші дані, щоб переносити їх в змінні. За допомогою чат-бота в Telegram можна відправляти розсилки за всім списком контактів бота, приймати платежі, створювати каталог товарів.

Результати порівняння систем-аналогів представлені в таблиці вище.

Характеристика	RetailCRM	1С:УНФ	Telegram bot, MS SQL
Прийом платежів	-	-	+
База клієнтів	+	+	+
Управління замовленнями	+	+	+
Моніторинг ефективності персоналу	+	+	+
Система лояльності	+	-	+
Управління тех. Підтримкою	+	+	-
Операційні системи	Веб-додаток	Веб-додаток, Android, iOS	Веб-додаток, Десктопна версія додатку, Android, iOS
Вартість	50\$/місяць	25\$/місяць	100\$ одноразова виплата

Таблиця 1.5.1. Порівняння систем-аналогів

З приведених систем найкраще себе показала система RetailCRM.

Її недоліками є ціна, неможливість приймати платежі, система підтримується тільки в веб-додатку. У результаті аналізу вищерозглянутих систем приходимо до висновку, що цю систему не слід впроваджувати для клієнт-менеджерів.

1.6. Обґрунтування доцільності проектування й розроблення системи

З наведеної вище таблиці робимо висновок, що розглянуті системи є дуже дорогими або мають суттєві недоліки, що робить їх не практичними в використанні. Це означає, що створення системи на основі MS SQL та веб-додаток в месенджері Telegram згідно потреб працівників – є актуальним і обґрунтованим завданням.

1.7. Концептуальна модель системи

Для того, щоб відобразити можливість створення звітів та всі можливі функції сучасної інформаційної системи для комфортної та швидкої роботи працівника – використовуємо модель TO-BE, яка зображена в додатку Б. Вона має один рівень декомпозиції та нове управління – посадові інструкції. Також на виході з'явилися звіти, а в механізмах правильна інформаційна система.

РОЗДІЛ 2. ТЕХНІЧНЕ ЗАВДАННЯ НА ПРОЕКТУВАННЯ

2.1. Постановка задачі на проектування

Метою даної роботи є забезпечення інформаційної підтримки та діджиталізація роботи магазину роздрібної торгівлі “Матролікс”.

Головною задачею системи є перенесення інформаційної бази роботи магазину до цифрового формату, зберігання в одному місці інформації про поставки та залишок товарів, а також про співробітників, товари загалом та постачальників з якими співпрацює магазин. Особливістю реалізації функцій, які покладені на інформаційну систему є необхідність зручного введення даних в таблиці за рахунок зручного інтерфейсу користувача з кнопками, підказками та іншими необхідними речами.

Обов'язками користувача, яким може бути касир, робітник складу або керівник магазину, є занесення до БД інформації про продажі, надходження товару та залишок товару в наявності.

В якості СУБД використовується MS SQL Server 2008, що дає змогу адміністратору робити архівну копію даних на випадок аварійної ситуації.

Користувачами даної системи виступають, касири, робітники складу або керівник магазину.

Інформаційна система магазину роздрібної торгівлі “Матролікс” виконує такі функції:

- Збереження даних про товару, надходження, залишок товару та співробітників магазину;
- Здійснення пошуку, фільтрації, вибірки з БД ;
- Формування звітів;

В якості клієнтської версії використовується месенджер Telegram зі створеним Telegram-ботом який створено за допомогою мови програмування JavaScript, підключеним фреймворком Node.js та node telegram bot api. База даних створена в документо-орієнтованій системі MongoDB з відкритим вихідним кодом, яка не потребує опису схеми таблиць. Що дає змогу користувачу комфортне

оформлення замовлення, зручний каталог асортименту магазину, консультацію з менеджером для вибору товарів.

2.2. Призначення та цілі створення системи

Інформаційні системи розроблюються для автоматизації роботи клієнт-менеджера та підвищення ефективності продажів в інтернеті. Основні цілі яких потрібно притримуватись при розробці даної системи є:

- Збільшення прибутку компанії;
- Підвищення продуктивності персоналу;
- Можливість формування звітів, пошуку, фільтрації інформації.

2.3. Вимоги до функцій та системи в цілому

Створений апаратно-програмний комплекс має забезпечити зберігання даних, зручний інтерфейс для перегляду замовлень, які надходять від клієнтів і на основі цих даних клієнт-менеджер буде створювати накладні для робітників складу які будуть збирати та відправляти замовлення клієнтам.

Даний апаратно-програмний комплекс буде призначений для інформаційної підтримки роботи компанії. Він забезпечить можливість вибору найоптимальнішого та найбільш вигідного для компанії варіанту оформлення замовлень.

Розроблювальний комплекс повинен забезпечувати безпомилкове збереження даних, підтримку популярними операційними системами, мати низьку ціну порівняно з існуючими системами, захищений доступ до даних, web інтерфейс, автоматичне формування звітів.

Вимоги до додатку для взаємодії з працівниками:

- Наявність Бази даних;
- Автоматичне формування звітів;
- Захист доступу до даних;
- Зберігання в одному місці інформації про поставки та залишок товарів;
- Зручний інтерфейс користувача з кнопками та підказками;
- Можливість пошуку, фільтрації по різним критеріям.

Вимоги до web-додатку:

- Додаток, що підтримується актуальними версіями браузерів, комп'ютерів, мобільних пристроїв;
- Інтуїтивно зрозумілий інтерфейс користувача;
- Каталог, який дає можливість швидко зорієнтуватись в категоріях товарів;
- Можливість додати товари до кошика і оформити замовлення.

РОЗДІЛ 3. ОПИС КОМПЛЕКСУ ЗАДАЧ АВТОМАТИЗАЦІЇ

3.1. Інформаційне забезпечення системи

Після виробничої практики, маючи всі необхідні дані, проводимо детальний аналіз магазину роздрібної торгівлі «Матролік».

Спочатку необхідно побудувати логічно-фізичну модель даних за допомогою CASE-засобу AllFusion ERWin Data Modeler. Фізична модель наведена в додатку В.

Далі на основі створеної моделі генеруємо базу даних в MS SQL Server 2008(Tools/Forward Engineer/Schema Generation), перед цим створивши порожню базу даних. Генерація структури БД на основі створеного SQL коду відбувається після натиснення кнопки Generate. Діалог зв'язку з БД і виконання SQL коду відбувається в результаті натиснення кнопки Connect. Отримуємо згенеровану базу даних в середовищі MS SQL Server 2008. Вигляд схеми даних наведений в додатку В.

Для під'єднання БД до середовища Microsoft Visual Studio 2015 використовується компонент джерело даних, потім обираємо додати нове джерело далі вводимо назву сервера та обираємо нашу БД , після перевірки підключення можна починати роботу. Схема БД в Microsoft Visual Studio 2015 має наступний вигляд.

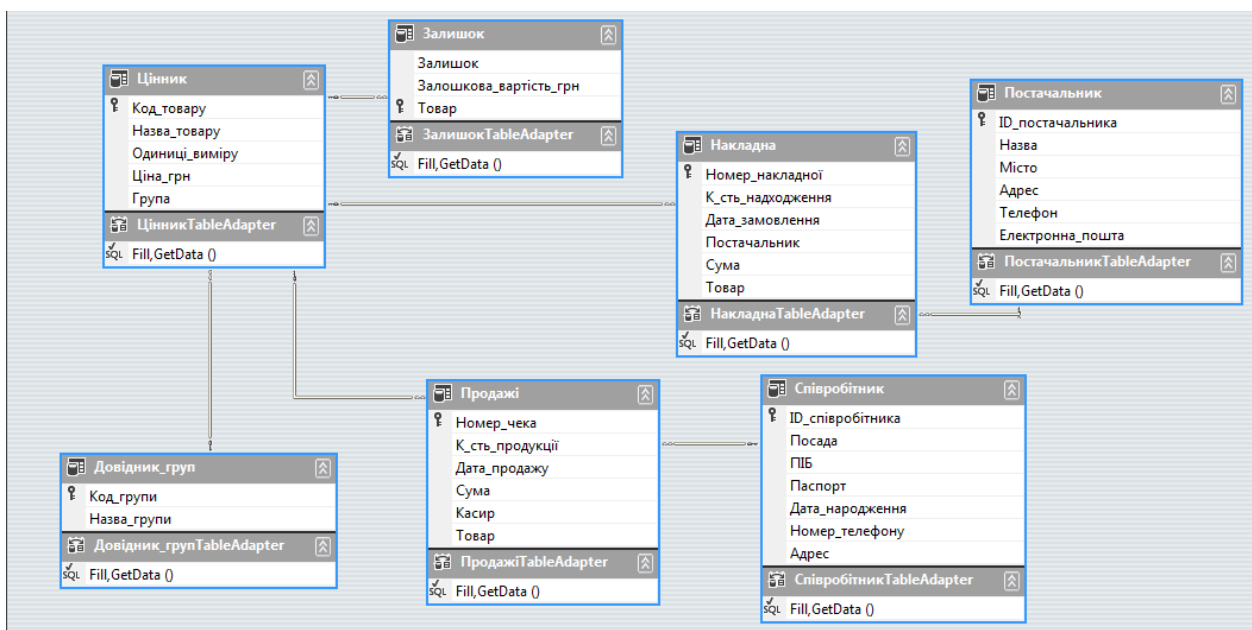


Рис. 3.1.1. Схема у вигляді XSD

Для створення web-додатку необхідно зробити підключення до серверу через командну строку, та згенерувати базу даних у середовищі MongoDBCompass, перед цим створивши порожню базу даних. Під'єднання структури бази даних в проект відбувається завдяки підключенню хосту в код, та зберіганню файлу з базою даних в дереві проекту.

База даних web-додатку складається з двох колекцій – картки товарів, користувачі.

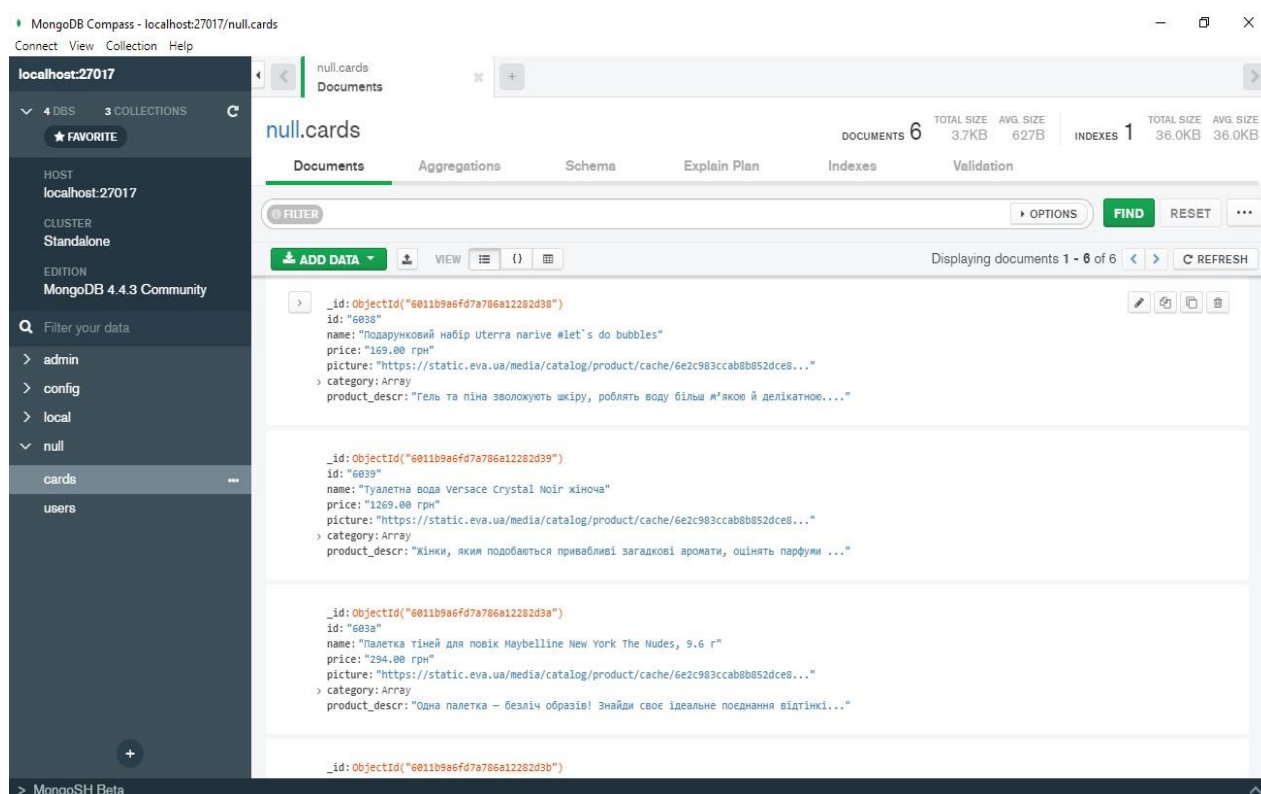


Рис.3.1.2. Вікно карточок в середовищі MongoDBCompass

Інформація про картки товарів знаходяться в колекції «cards», яка містить поля:

- Назва товару;
- Ціна;
- Опис товару;
- Рисунок;

- Категорія.

3.2. Алгоритмізація та реалізація комплексу задач автоматизації

Для організації управління додатком за допомогою меню, на головній формі розміщуємо елемент MenuStrip та додаємо необхідні вкладки:

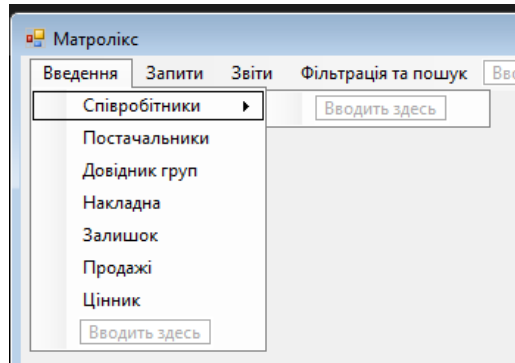


Рис. 3.2.1. Створення меню проекту

Для можливості переходу між формами для кожної кнопки меню використовуємо наступний програмний код:

```
private void співробітникиToolStripMenuItem_Click(object sender, EventArgs e)
{
    worker N = new worker();
    N.ShowDialog();
}
```

Аналогічний код, зі зміною назви класу, який описує вікно, застосовуємо для всіх пунктів меню.

Для того, щоб помістити поля на форми введення даних у меню Дані застосовуємо команду «Показати джерела даних» або використовуємо гарячі клавіші *SHIFT+ALT+D*. Обираємо необхідну таблицю, та спосіб представлення, та перетягуємо її на форму.

Для додавання зв'язаних таблиць треба перетянути відповідну дочірню таблицю в меню «Джерела даних» в списку полів батьківської таблиці.

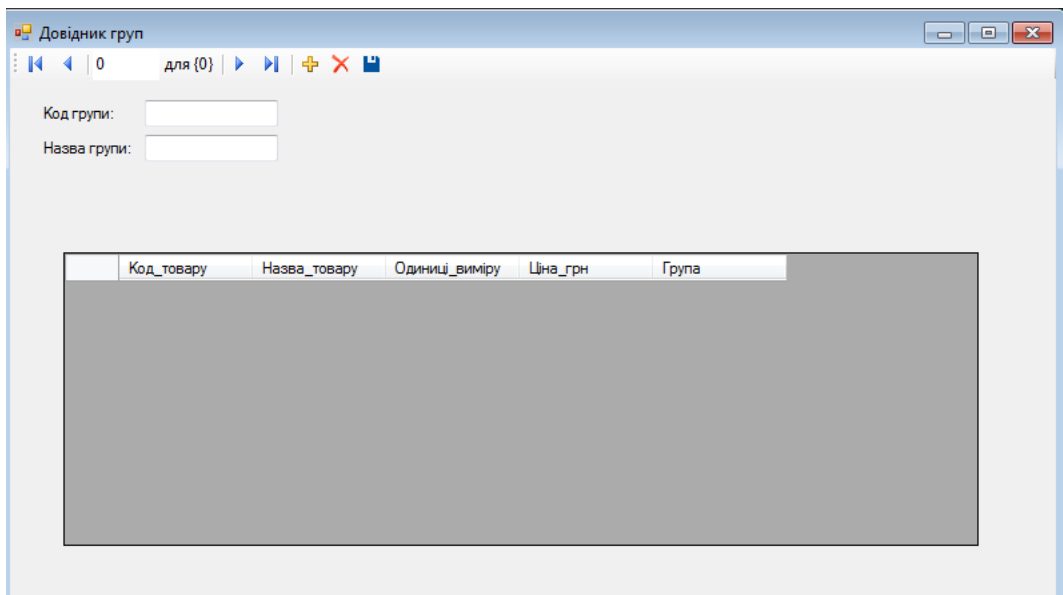


Рис.3.2.2. Додавання підпорядкованої таблиці «Цінник» до батьківської таблиці «Групи товарів»

Для зручності введення інформації в підпорядкованих таблицях заміняємо поле «код_групи» тощо, на випадючі списки з відповідними найменуваннями (наприклад: «код_групи» -> «Група»).

Налаштуємо властивості компонентів ComboBox для зв'язку з відповідними таблицями і формування списків як показано на рис. 4.

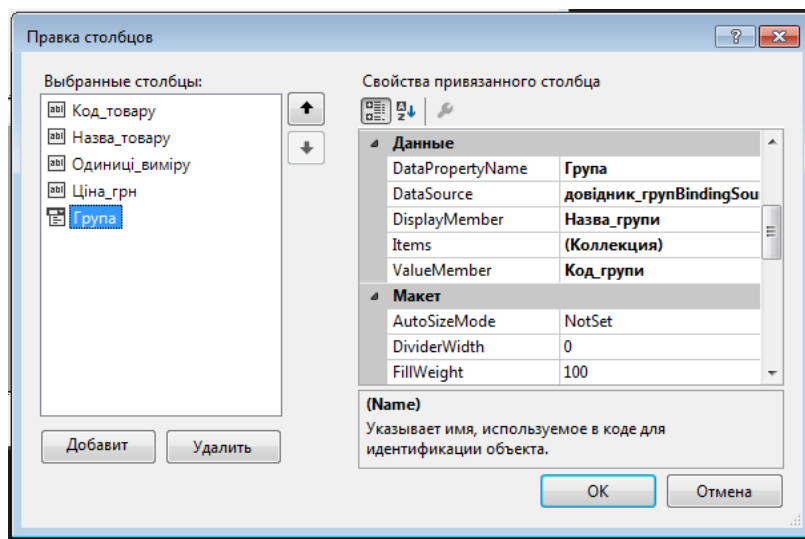


Рис.3.2.3. Налаштування властивостей поля зі списком

Для зручності пошуку, створимо параметричний запит.

Впершу чергу для створення параметричного запиту нам потрібно створити новий tableadapter в своїй схемі даних KURS_OLDataSet.xsd.

Для цього на пустому місці поля схеми даних натиснемо ПКМ та оберемо пункт додати->tableadapter.

Після чого натиснема на створеному tableadapter ПКМ на оберемо пункт налаштувати.

Відкриється вікно запиту даного tableadapter.

За допомогою «Построитель запросов», створимо пошук чеків обраного касира:

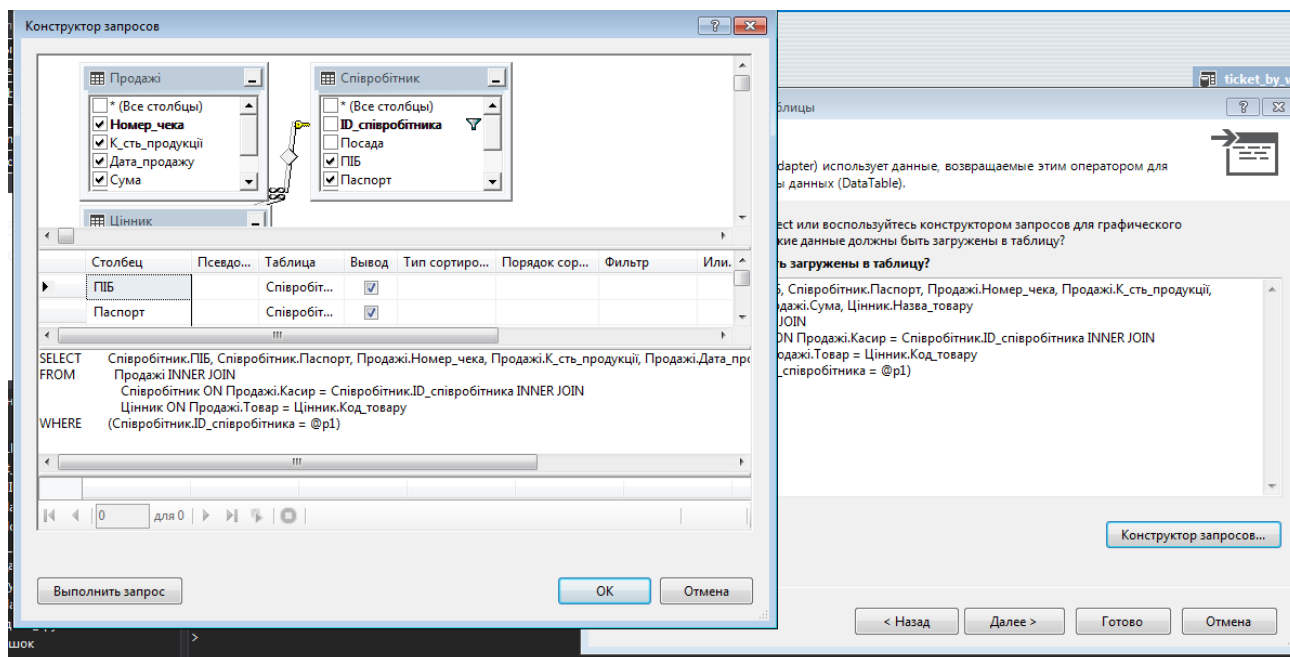


Рис.3.2.4. Створення параметричного запиту

В результаті отримуємо TableAdepter такого вигляду:

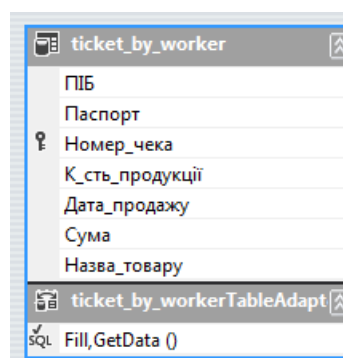


Рис.3.2.5. TableAdepter параметричного запиту

Для зручності організуємо поле пошуку у вигляді випадуючого списку (ComboBox), та прив'яжемо його до даних:.

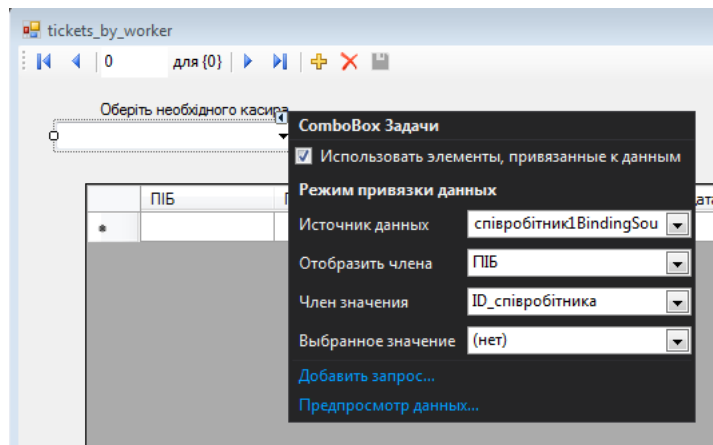


Рис.3.2.6. Налаштування властивостей ComboBox, привязка до даних

Так як в процедурі пошуку ми беремо дані з списку за допомогою функції *SelectedValue*, яка приймає дані з *ValueMember*, це означає, що в полі *Член значення*, треба обрати значення, за яким ми шукаємо (у даному випадку, поле – *ПІБ працівника*).

Код кнопки пошуку має такий вигляд:

```
private void button1_Click(object sender, EventArgs e)
{
    try
    {
        this.ticket_by_workerTableAdapter.Fill(this.kURS_OLDataSet.ticket_by_worker,
        comboBox1.Text);
    }
    catch (System.Exception ex)
    {
        System.Windows.Forms.MessageBox.Show(ex.Message);
    }
}
```

}

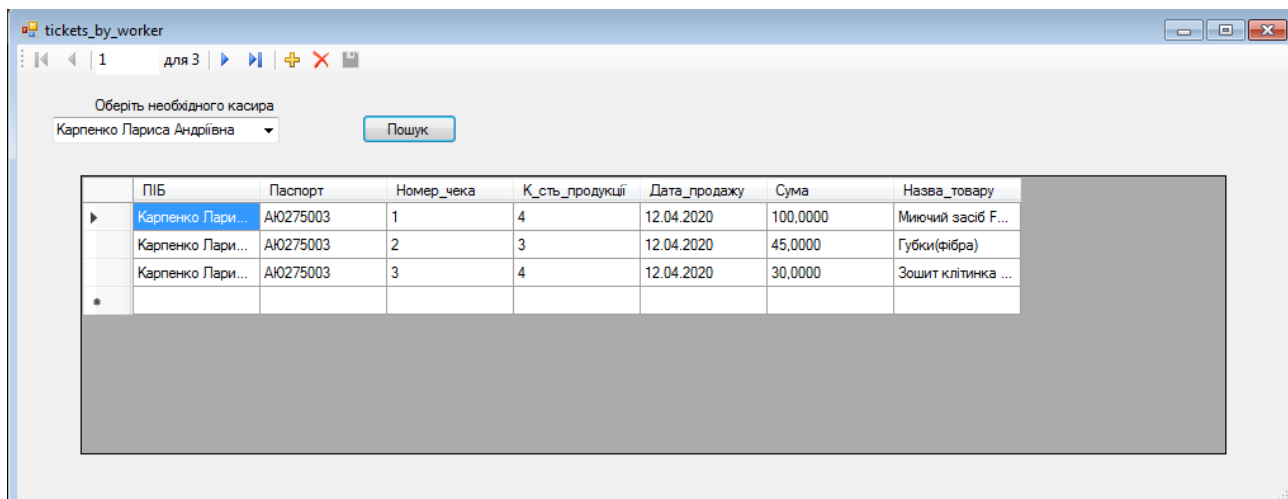


Рис.3.2.7. Форма реалізації параметричного пошуку

Аналогічним способом забезпечено пошук накладних постачальника, результат має такий вигляд:

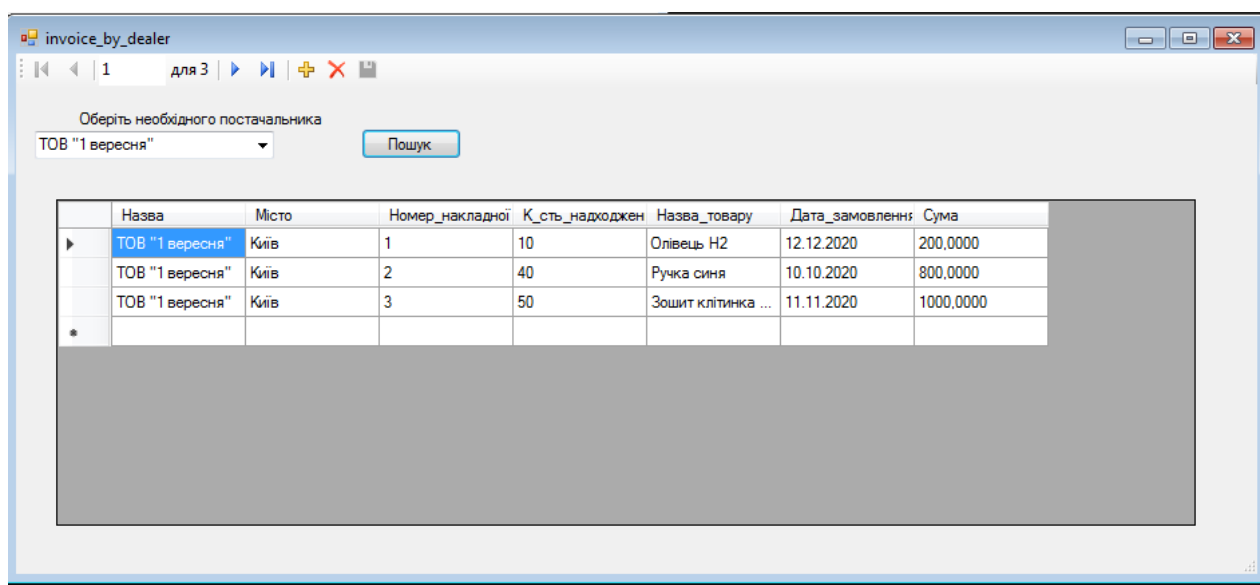


Рис.3.2.8. Пошук накладних обраного постачальника

Для можливості швидко знаходити товари, яких не залишилося, або за введеним параметром, скільки залишилося товару використовуємо фільтр для даних.

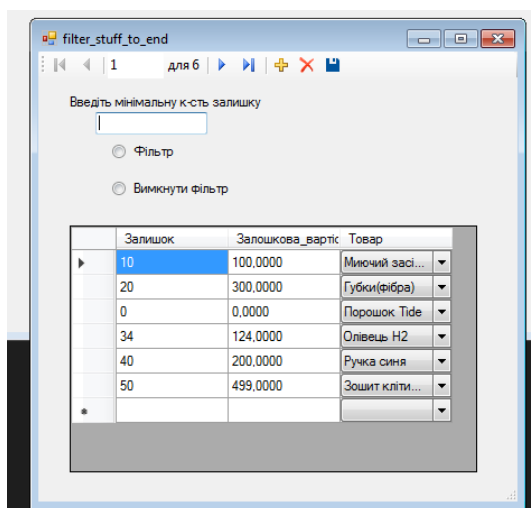


Рис.3.2.9. Форма з реалізованим фільтром

В даному фільтрі умова фільтра має наступний вигляд:

```
this.залишокBindingSource.Filter = "Залишок <= " + textBox1.Text;
```

Це означає, що при активації даного фільтра, на екрані будуть виводитися остатки тих продуктів, чий остаток менший або дорівнює введеному числу.

Це дозволить нам знаходити остатки товарів з різними значеннями, адже для кожного товару, різною є норма остатку на складі.

Код `RadioButton` для ввімкнення та вимкнення режиму фільтрації, має такий вигляд:

```
private void radioButton1_CheckedChanged(object sender, EventArgs e)
    {
        this.залишокBindingSource.Filter = "Залишок <= " + textBox1.Text;
    }
private void radioButton2_CheckedChanged(object sender, EventArgs e)
    {
        this.залишокBindingSource.RemoveFilter();
    }
```

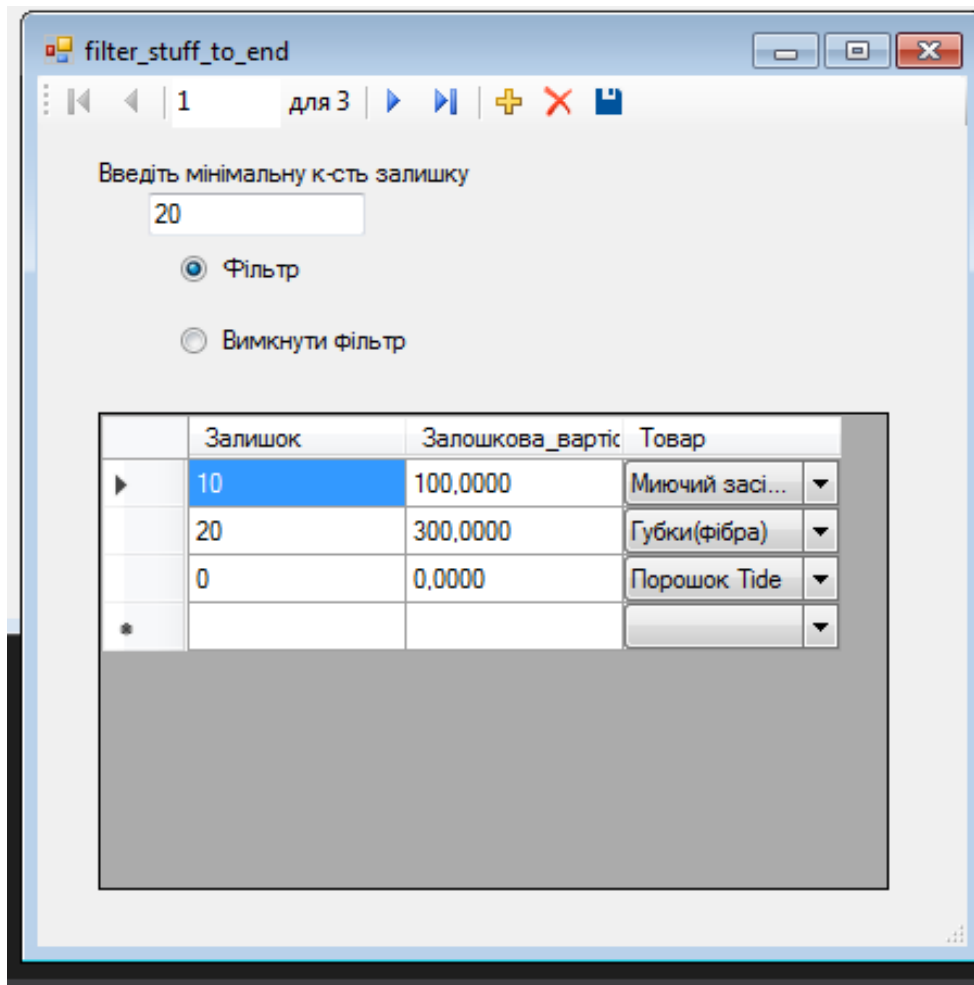


Рис.3.2.10. Приклад виконання фільтрації

Пошук накладних обраного постачальника.

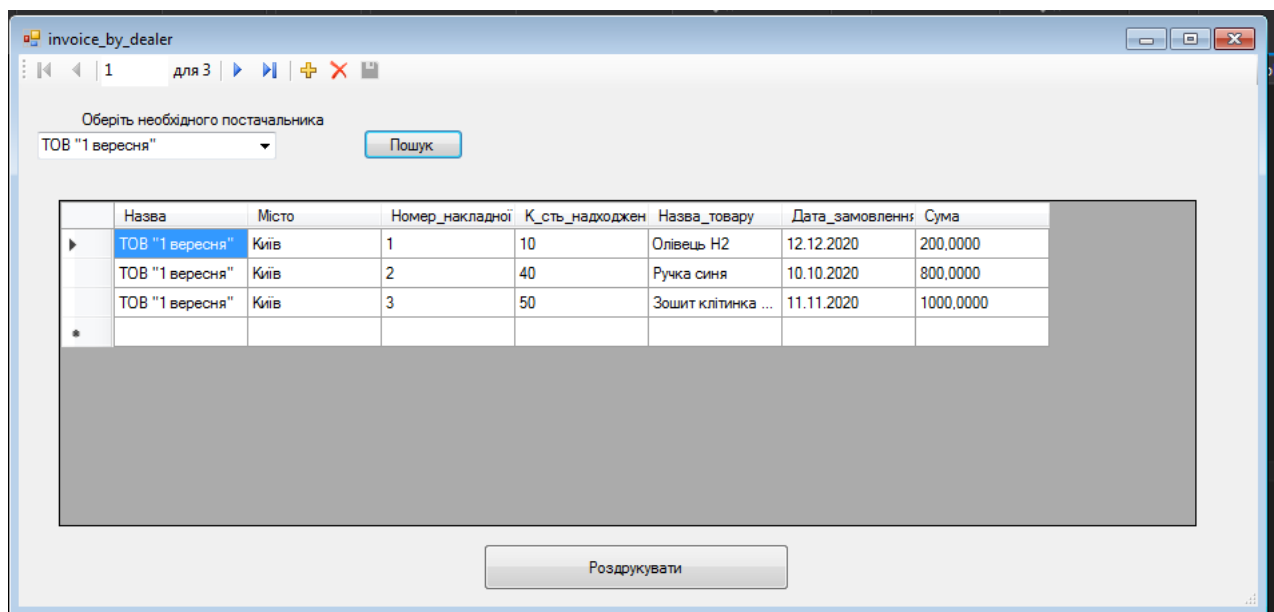


Рис.3.2.11. Приклад пошуку квитанції за номером угоди

Як бачите на цій формі є кнопка для друку, її код, також код клавiшi для пошуку наведений на Рис 15.

```
private void button1_Click(object sender, EventArgs e)
{
    try
    {
        this.invoice_by_dealerTableAdapter.Fill(this.kURS_OLDDataSet.invoice_by_dealer, comboBox1.Text);
    }
    catch (System.Exception ex)
    {
        System.Windows.Forms.MessageBox.Show(ex.Message);
    }
}

private void button2_Click(object sender, EventArgs e)
{
    PrintDialog dlg = new PrintDialog();
    dlg.ShowDialog();
}
}
```

Рис.3.2.12. Коди клавiш на формi

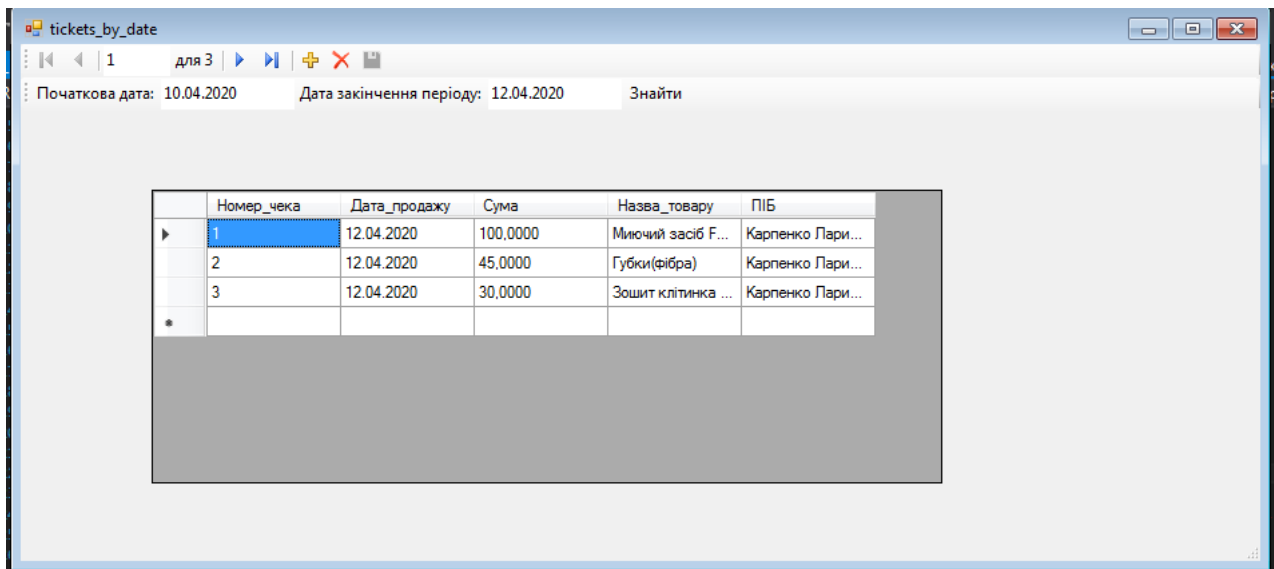


Рис.3.2.13. Пошук чеків заданого періоду

Sql-код:

```
SELECT          Продажі.Номер_чека,  Продажі.Дата_продажу,  Продажі.Сума,
Цінник.Назва_товару,  Спiвробiтник.ПІБ
FROM          Продажі INNER JOIN
              Цінник ON Продажі.Товар = Цінник.Код_товару INNER JOIN
              Спiвробiтник ON Продажі.Касир = Спiвробiтник.ID_спiвробiтника
```

WHERE (Продажі.Дата_продажу >= @p1) AND (Продажі.Дата_продажу <= @p2)

```
private void fillToolStripButton_Click(object sender, EventArgs e)
{
    try
    {
        this.tickets_by_dateTableAdapter.Fill(this.kURS_OLDataSet.tickets_by_date, p1ToolStripTextBox.Text, p2ToolStripTextBox.Text);
    }
    catch (System.Exception ex)
    {
        System.Windows.Forms.MessageBox.Show(ex.Message);
    }
}
```

Рис.3.2.14. Код клавiші пошуку

Для створення привітального повідомлення в web-додатку і виведення основного меню в інтерфейсі користувача потрібно додати елемент `.onText` та `.sendMessage`:

```
//привітальне повідомлення
bot.onText(/start/, msg => {
    const text = `Привіт, ${msg.from.first_name} ! Тебе вітає магазин Matrolix!\n\nВ нас ти можеш придбати круті аксесуари, та подаруночки для своїх близьких людей 🎁\n\nНавігація м агазину знаходиться трішки нижче Якщо виникнуть запитання по роботі бота або потрібн а буде консультація по асортименту - клікай по кнопці "Зв'язатись з менеджером" 🗺️♀`
    //Основне меню
    bot.sendMessage(helper.getChatId(msg), text, {
        reply_markup: {
            keyboard: keyboard.home,
            resize_keyboard: true
        }
    })
})
```

Рис.3.2.15 Привітальне повідомлення та основне меню

Для можливості переходу між формами для кожної кнопки меню використовуємо наступний програмний код:

```
bot.on('message', msg => {
    const chatId = helper.getChatId(msg)
    switch (msg.text) {
        case keyboardButtons.home.inline:
            bot.sendMessage(chatId, `Виберіть категорію товару`, {
                reply_markup: { keyboard: keyboard.inline }
            })
    }
})
```

```

    })
    break
case keyboardButtons.inline.valentine:
    sendCardsByQuery(chatId, { category: 'valentine' })
    break
case keyboardButtons.inline.bracletes:
    sendCardsByQuery(chatId, { category: 'bracletes' })
    break
case keyboardButtons.inline.office:
    sendCardsByQuery(chatId, { category: 'office' })
    break
case keyboardButtons.inline.pins:
    sendCardsByQuery(chatId, { category: 'pins' })
    break
case keyboardButtons.home.bag:
    showBagCards(chatId, msg.from.id)
    break
case keyboardButtons.back:
    bot.sendMessage(chatId, `Що бажаєте придбати?`, {
      reply_markup: {
        keyboard: keyboard.home,
        resize_keyboard: true
      }
    })
    break
}
})

```

Відображення всіх карток в категорії обробляються методом `.on` з коду вище, наступною функцією:

```

function sendCardsByQuery(chatId, query) {
  Card.find(query).then(cards => {
    const html = cards.map((f, i) => {
      return `>${i + 1}</b> ${f.name} - ${f.price} /f${f._id}`
    }).join('\n')
  })
}

```

```

    sendHTML(chatId, html, 'cards')
  })
}

```

Telegram-боти мають два види клавіатур – основна, яка відображається на протязі всієї сесії в боті, та інлайн-клавіатура яка прив’язана до конкретного повідомлення бота.

При бажанні основну клавіатуру можна сховати, але основна навігація знаходиться в основній клавіатурі, тому приховувати її не бажано. Інлайн клавіатура використовується в додатку лише раз в картці товару – для можливості додати товар в кошик, та кнопка-посилання для можливості замовити товар швидко, тобто вона веде одразу в чат з клієнт-менеджером.

У меню всіх видимих карток категорії користувач може вибрати будь-яку картку натиснувши на посилання, яке перенесе його в картку з детальним описом товару:

```

//Детальне зображення картки
bot.onText(/\/f(.+)/, (msg, [source, match]) => {
  const cardId = helper.getItemId(source)
  const chatId = helper.getChatId(msg)

  Promise.all([
    Card.findOne({ _id: cardId }),
    User.findOne({ telegramId: msg.from.id })
  ]).then(([cards, users]) => {
    let isBag = false
    if (users) {
      isBag = users.card.indexOf(cards._id) !== -1
    }
    const bagText = isBag ? 'Видалити з кошика' : 'Додати у кошик'
    const caption = `Назва: ${cards.name}\nЦіна: ${cards.price}\nОпис товару: ${cards.produ
ct_descr}`

    //Налаштування інлайн-клавіатури
    bot.sendPhoto(chatId, cards.picture, {
      caption: caption,

```

```

reply_markup: {
  inline_keyboard: [
    [
      {
        text: bagText,
        callback_data: JSON.stringify({
          type: ACTION_TYPE.TOGGLE_BAG_CARD,
          cardId: cards._id,
          isBag: isBag
        })
      }
    ],
    [
      {
        text: 'Замовити зараз!',
        url: 'https://t.me/matrolya'
      }
    ]
  ]
}
}).catch(e => console.log(e))
})

```

Функції роботи кошику описані нижче:

//додавання та видалення з кошику

```

function toggleBagCard(userId, queryId, { cardId, isBag }) {
  let userPromise
  User.findOne({ telegramId: userId })
  .then(user => {
    if (user) {
      if (isBag) {
        user.cards = user.card.filter(cId => cId !== cardId)
      } else {
        user.card.push(cardId)
      }
    }
  })
}

```

```

    userPromise = user
  } else {
    userPromise = new User({
      telegramId: userId,
      cards: [cardId]
    })
  }
  const answerText = isBag ? 'Видалено' : 'Додано до кошика'
  userPromise.save().then(_ => {
    bot.answerCallbackQuery(queryId, {
      callback_query_id: queryId,
      text: answerText
    })
  }).catch(err => console.log(err))
})
}
//кошик
function showBagCards(chatId, telegramId) {
  User.findOne({ telegramId })
  .then(user => {
    if (user) {
      Card.find({ _id: { '$in': user.card } }).then(card => {
        let html
        if (card.length) {
          html = card.map(c => {
            return `${c.name} - <b>${c.price}</b>`
          }).join('\n')
          html = `<b>Товари у кошику:</b>\n${html}`
        } else {
          html = 'Додайте товар до кошика'
        }
        sendHTML(chatId, html, 'home')
      })
    } else {
      sendHTML(chatId, 'Додайте товар до кошика', 'home')
    }
  })
}

```

```
}  
}).catch(e => console.log(e))  
}
```

3.3. Інструкція користувача

При запускові програми з'являється головне меню, що забезпечує навігацію по додаткові, необхідно лиш обрати функцію, яку необхідно виконати: введення, запити, звіти або фільтрація:

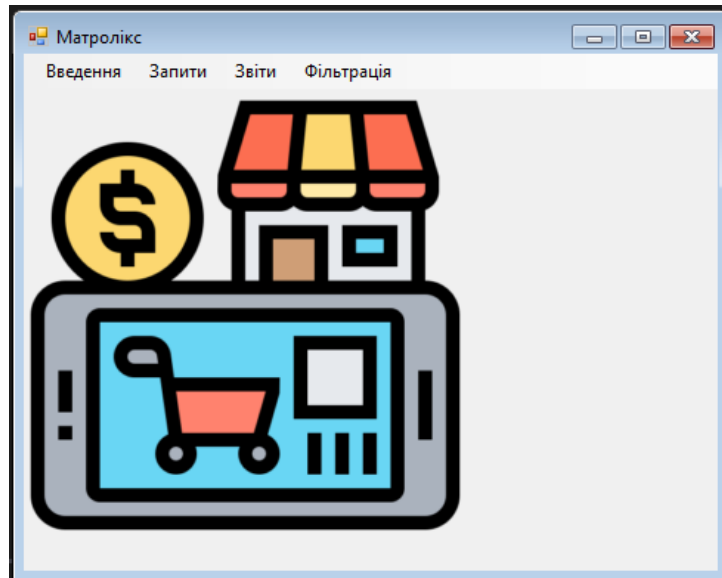


Рис.3.3.1. Головне меню

Навігація забезпечує перехід за такими вкладками:

Введення:

- Співробітники
- Постачальники/Накладні
- Довідник товарів
- Залишок
- Продажі

Запити:

- Пошук “Чеки касира”
- Пошук “Товари з малим залишком”
- Пошук “Накладні постачальника”
- Пошук “Чеки заданого періода”

- Пошук “Накладні заданого періода”

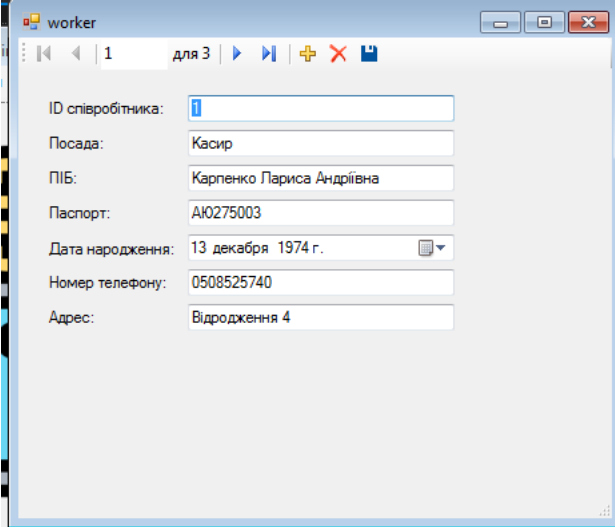
Звіти:

- Звіт “Чеки касирів”
- Звіт “Чеки за обраний період”
- Звіт “Накладні постачальників”
- Звіт “Накладні за обраний період”
- Звіт “Залишок продукції”

Фільтрація:

- Товари, що скінчилися
- Список касирів
- Товари обраної категорії

Для введення даних про співробітників оберіть необхідний пункт меню (*Введення/співробітники*):



The image shows a screenshot of a software application window titled "worker". The window contains a form for entering employee data. The form fields are as follows:

ID співробітника:	<input type="text"/>
Посада:	Касир
ПІБ:	Карпенко Лариса Андріївна
Паспорт:	АЮ275003
Дата народження:	13 декабря 1974 г.
Номер телефону:	0508525740
Адрес:	Відродження 4

Рис.3.3.2. Форма співробітників

У формі *Залишок* доступ до даних аналогічний.

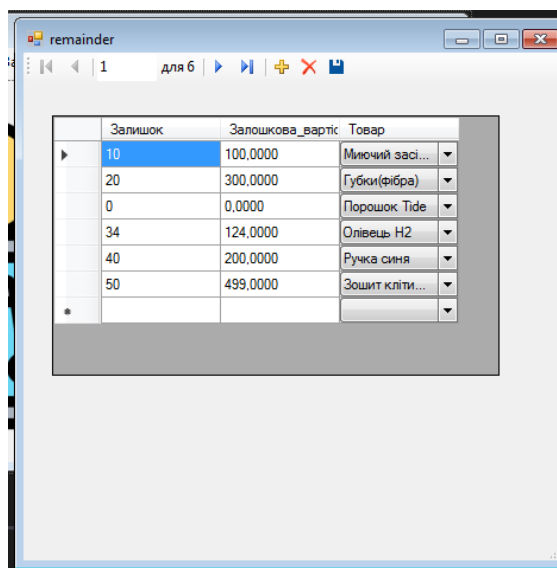


Рис.3.3.3. Форма залишку

У формі Постачальники/Накладні доступ введення та перегляду даних, відрізняється, тут ми бачимо область №1 для введення даних про постачальників, а також область №2 для введення накладних цих постачальників.

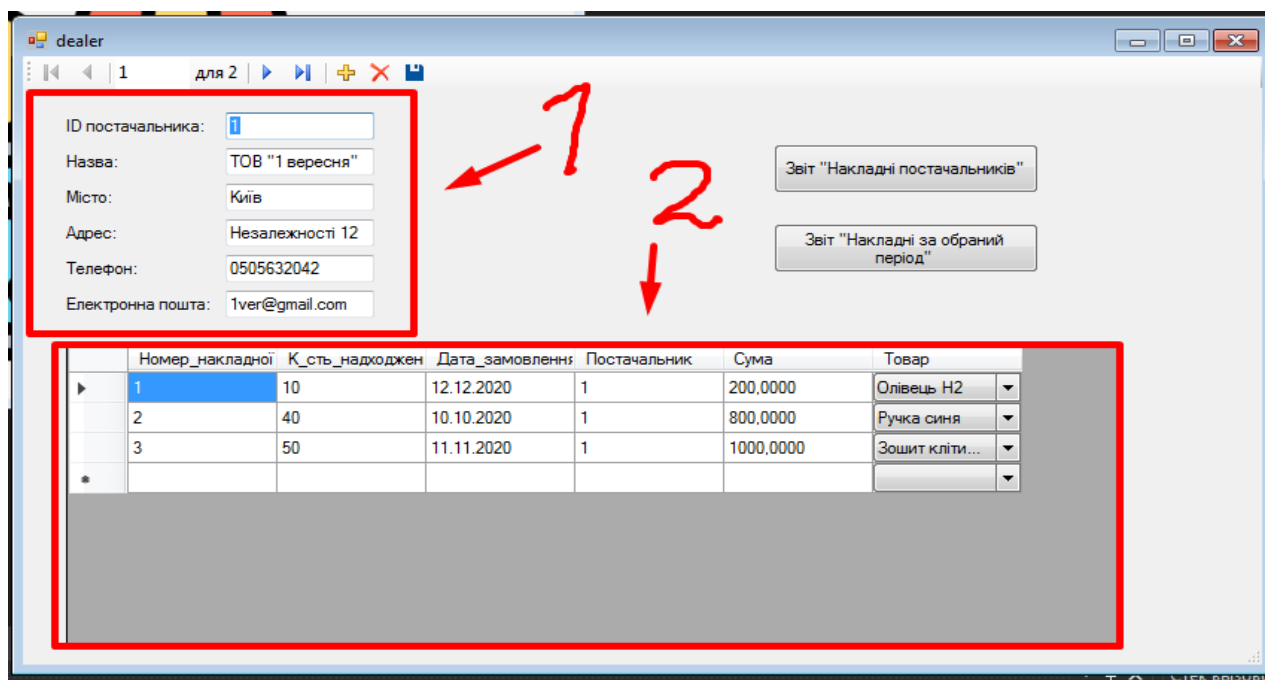


Рис.3.3.4. Форма накладної

При натисненні на кнопку «Звіт "Накладні постачальників"», відкривається звіт:

Назва	Дата замовлення	Назва товару	Сума
ТОВ "1 вересня"	10.10.2020 0:00:00	Ручка синя	800,0000
		Всього	800,0000
	11.11.2020 0:00:00	Зошит клітинка 12арк	1000,0000
		Всього	1000,0000
	12.12.2020 0:00:00	Олівець H2	200,0000
		Всього	200,0000
Всього		2000,0000	
ТОВ "УКРХІМ"	10.11.2020 0:00:00	М'яч Fairy 1л	300,0000
		Всього	300,0000
	12.11.2020 0:00:00	Порошок Tide	1200,0000
		Всього	1200,0000
	Всього		1500,0000

Рис.3.3.5. Звіт «Накладні постачальників»

Тобто 2 клавіші зі світами дублюють дії, які можна виконати з вкладки, Звіти, що спрощує користування додатком.

Для пошуку чеків касира необхідно відкрити форму *Пошук/Чеки касира*, в якому обрати з випадаючого списку необхідного касира та натиснути Пошук.

	ПІБ	Паспорт	Номер_чека	К_сть_продукції	Дата_продажу	Сума	Назва_товару
▶	Карпенко Париса Андріївна	АЮ275003	1	4	12.04.2020	100,0000	М'яч Fairy 1л
	Карпенко Париса Андріївна	АЮ275003	2	3	12.04.2020	45,0000	Губки(фібра)
	Карпенко Париса Андріївна	АЮ275003	3	4	12.04.2020	30,0000	Зошит клітинка ...
*							

Рис.3.3.6. Пошук чеків касира

При відкритті вікна “Фільтр “Товари що закінчуються” відкриється вікно фільтрації товарів, де ми можемо обрати, яка к-сть товарів на складі нам необхідна, і після активації фільтра за допомогою радіокнопки, нам будуть виведені на екран всі товари в яких залишок нижчий або дорівнює введеному, це спростить процес моніторингу товарів, що закінчуються:

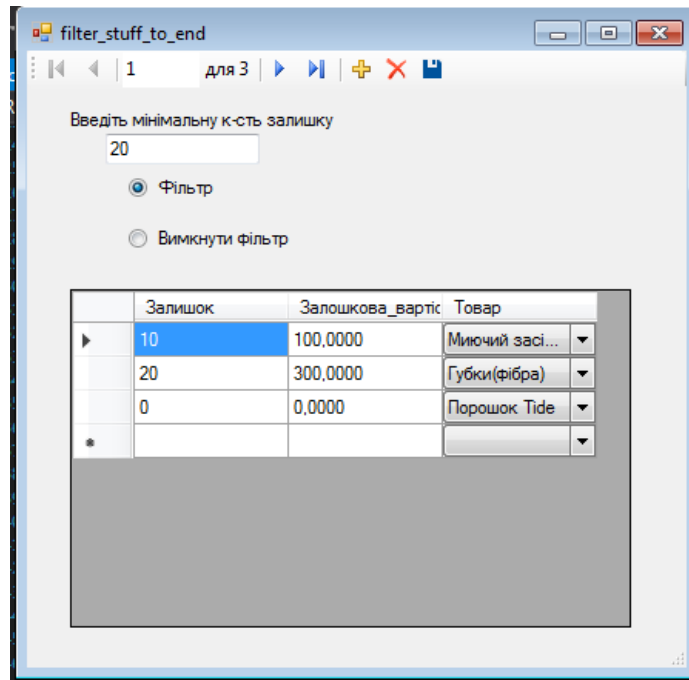


Рис.3.3.7. Вікно «Фільтр»

При запускові web-додатку в месенджері Telegram, бот просить нас запусити додаток:

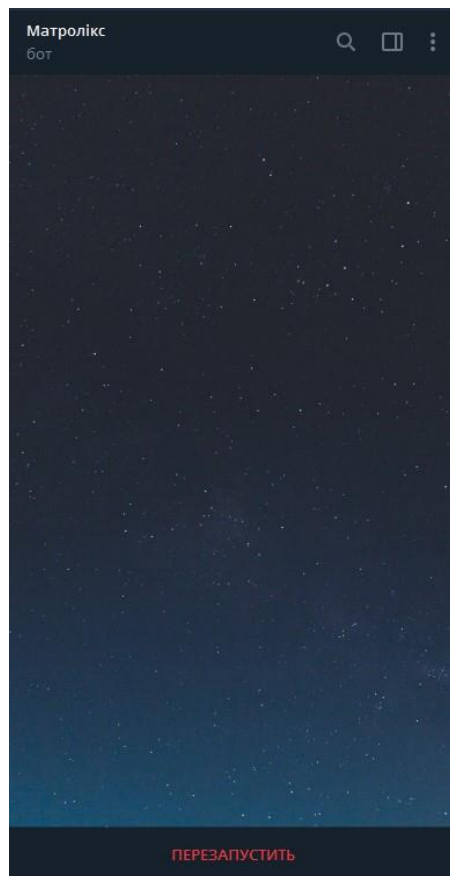


Рис.3.3.8. Запуск Telegram-бота

Після чого з'являється привітальне повідомлення та навігація:

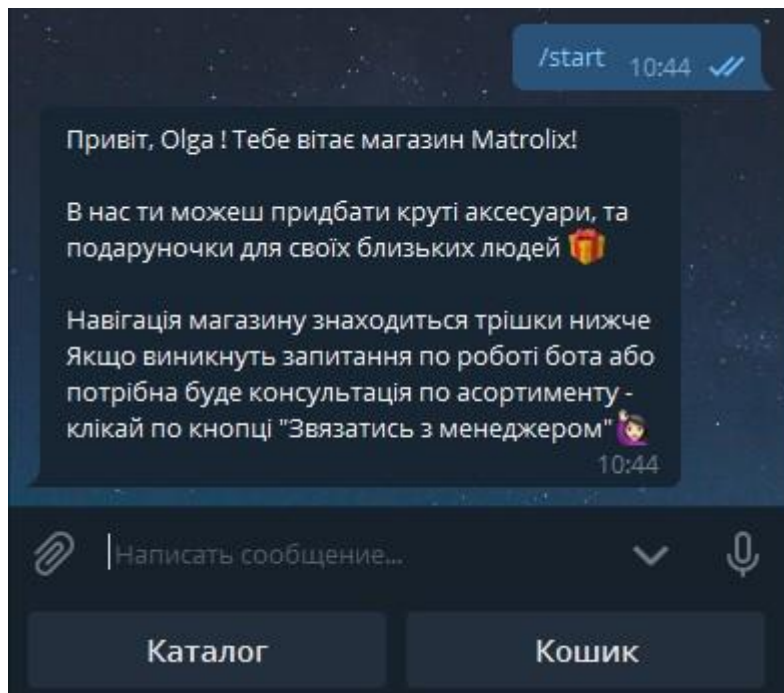


Рис.3.3.9. Головне меню

В каталозі є можливість вибору категорії товару:

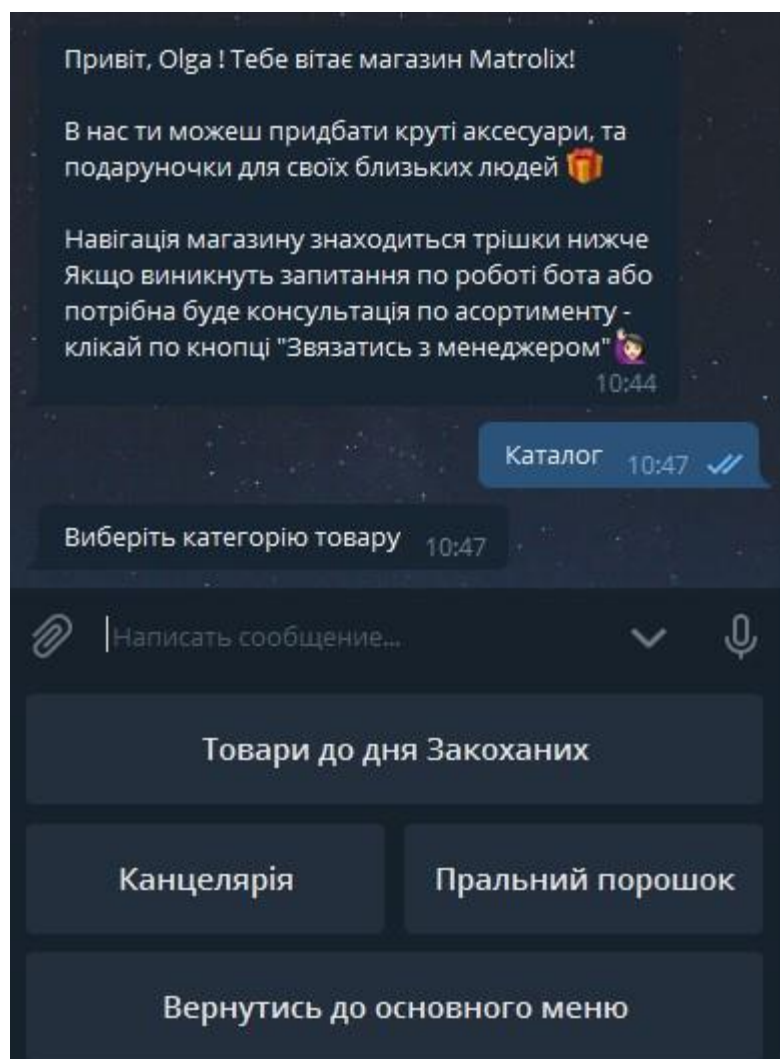


Рис.3.3.10. Каталог товарів

В меню всіх видимих карток категорії ми бачимо порядковий номер картки товару, назву, ціну та посилання, натиснувши на яке відкриється картка товару з можливістю додати товар в кошик або замовити товар швидко у клієнт-менеджера, натиснувши клавішу інлайн-клавіатури «Замовити зараз!», якщо товар вже є у кошику клавіша інлайн-клавіатури «Додати у кошик» - зміниться на «Видалити з кошика»:

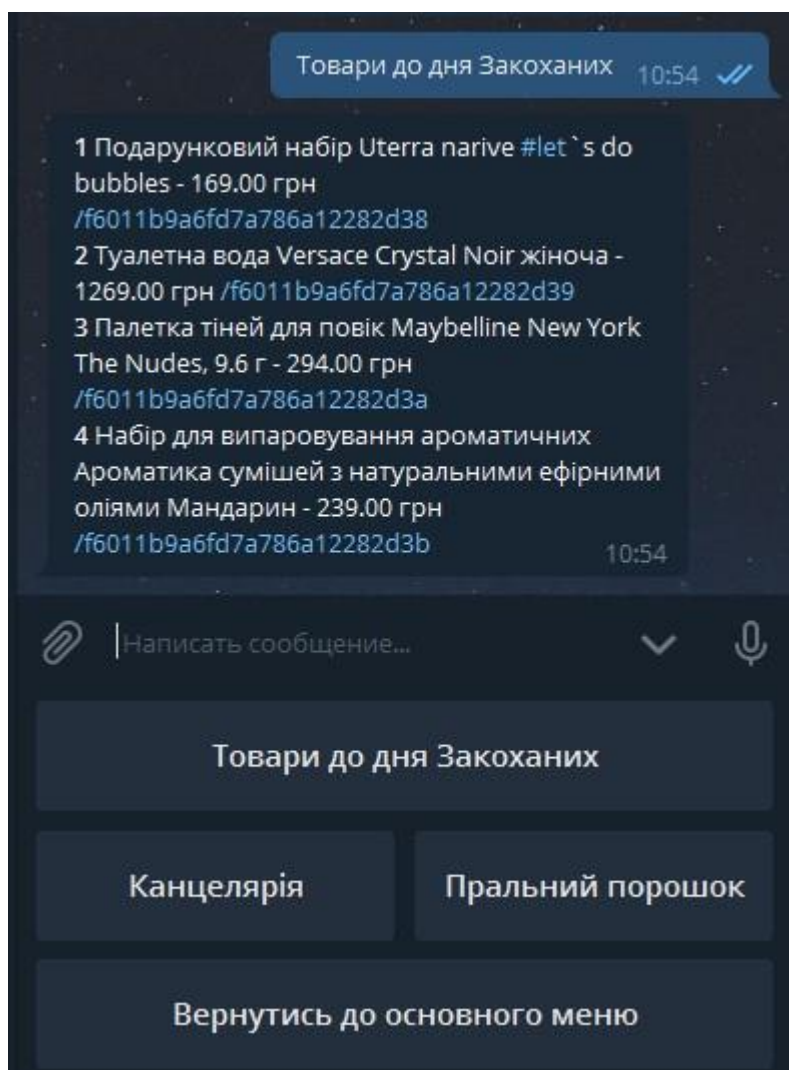


Рис.3.3.11. Меню всіх видимих карток

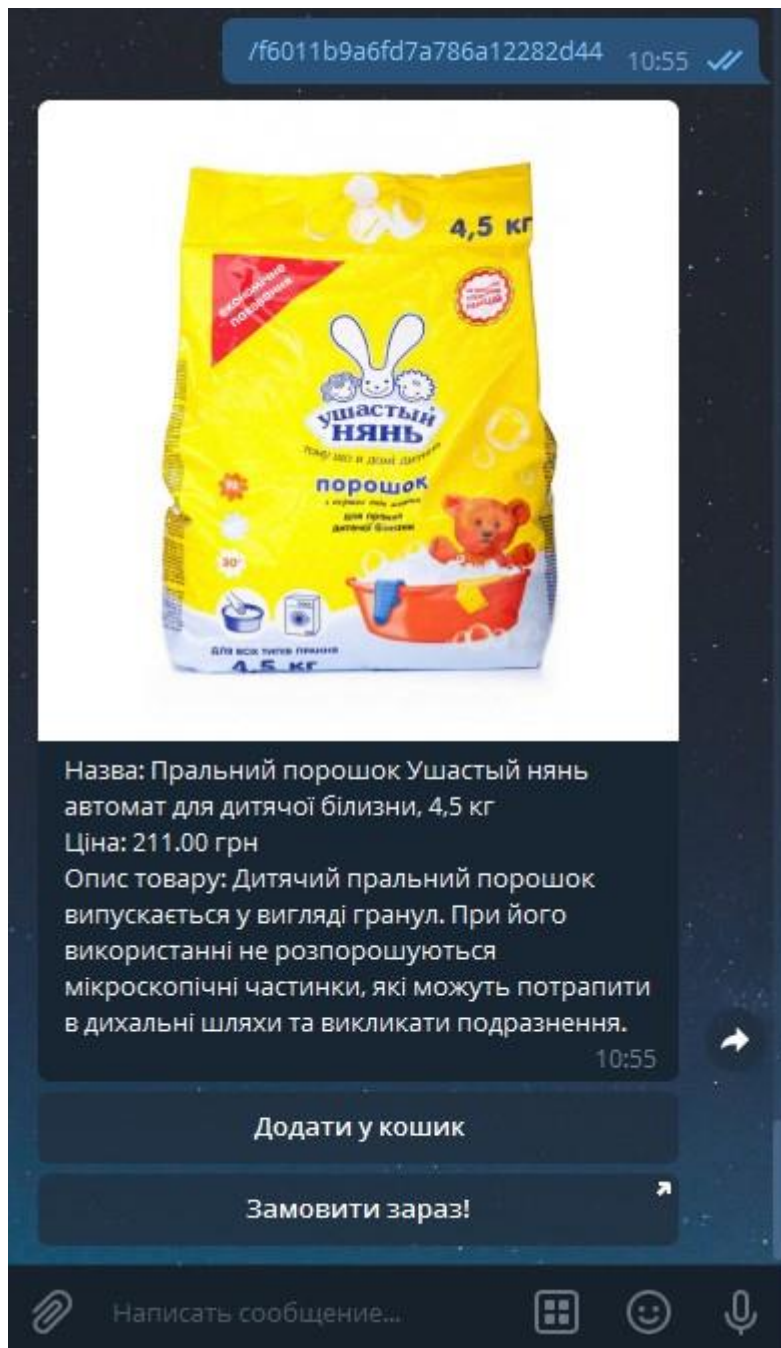


Рис.3.3.12. Вид картки з можливістю додавання товару в кошик

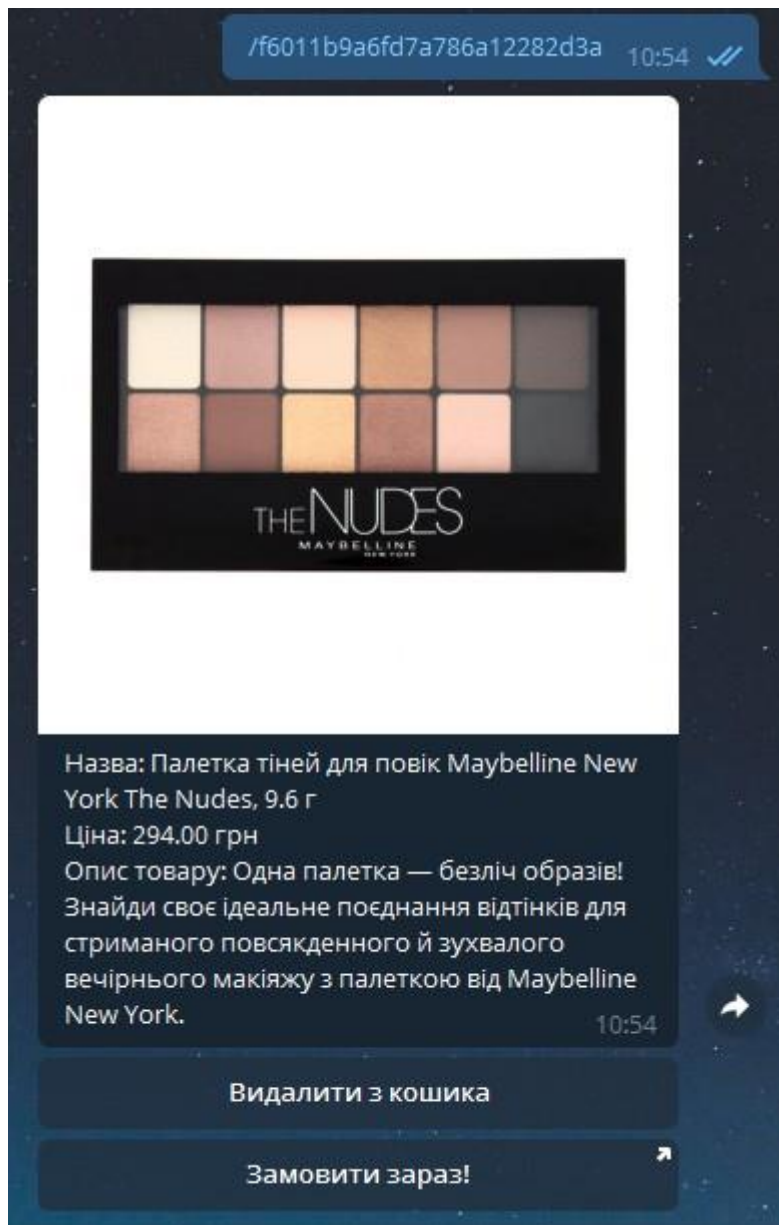


Рис.3.3.13. Вид картки з можливістю видалити товар з кошику

Товари в кошику виглядають наступним чином:

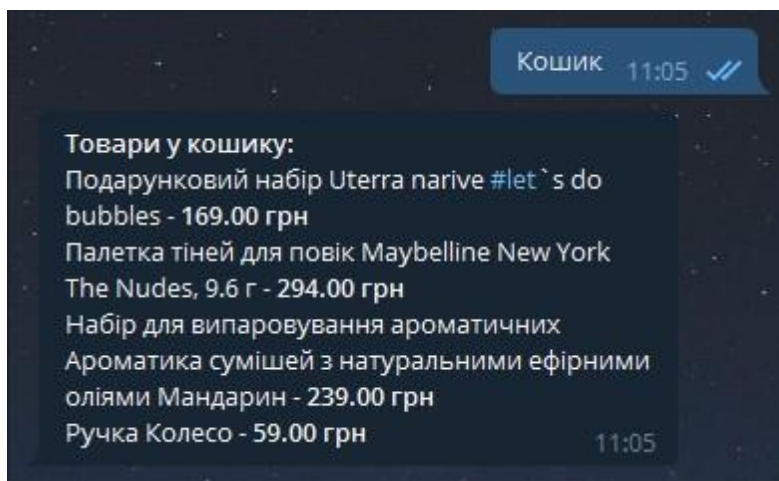


Рис.3.3.14. Меню кошика

3.4 Технічне та системне забезпечення розробки

3.4.1 Обґрунтування вибору технічних засобів

В даний час основна маса вітчизняних виробництв потребують автоматизації в тому чи іншому вигляді. В сучасних умовах ринку виживають ті, хто найбільш конкурентної здатні, хто вміє організувати свої виробничі процеси з найбільшою ефективністю і найменшими затратами. Але як знизити витрати на виробництво без автоматизації компанії в цілому, а також і автоматизації технологічних процесів? Очевидно, що сучасній компанії, для того щоб рости і розвиватися, необхідна «нова кров» у вигляді нових технологій і рішень. Комп'ютеризовані системи все більше охоплюють виробничий сектор, на робочих місцях менеджерів починають з'являтися прості та ефективні засоби для оперативного контролю та управління технологічними процесами. Не викликає сумнівів те, що грамотно побудована система – це не тільки набір програмних і апаратних засобів. В більшості випадків – це організаційно-технічне планування і розподілення функцій клієнт-менеджера та технологічного устаткування. Перш за все, це аналіз поточного стану об'єкта автоматизації і вибір саме тих вузлів, які дозволять дійсно на порядок збільшити ті чи інші якісні показники. Під структурою управління розуміють сукупність частин автоматичної системи, на які вона може бути розділена по визначеній ознаці. Вибір структури управління об'єктом автоматизації робить вагомий вплив на ефективність його роботи, зниження відносної вартості системи управління, її надійності, ремонт здатності і т.д.

Розробка організаціями стратегічних планів, забезпечення підтримки діяльності співробітників, ідентифікація проблем, що виникають, та взаємодія з іншими організаціями багато в чому визначаються рівнем якості інформації. Один із способів огляду якості інформації полягає в оцінці її характеристик з точки зору їх значення для організації. В число значимих відмінних властивостей інформації входять час, зміст та форма.

РОЗДІЛ 4. ОХОРОНА ПРАЦІ

4.1 Заходи пожежної безпеки у службових приміщеннях

Вимоги пожежної безпеки Меблі та обладнання мають розміщатися так, щоб забезпечити евакуаційний прохід до виходу з приміщення (заввишки не менше 1,0 м). Евакуаційні шляхи та виходи слід завжди утримувати вільними, нічим не захащувати. В міру накопичення горючих відходів, а також після закінчення роботи їх слід прибрати у спеціально відведені сміттєзбірники. Електромережі, електроприлади та арматура повинні експлуатуватися тільки у справному стані з урахуванням вказівок і рекомендацій заводів – виробників. У разі пошкоджень електромереж, вимикачів, розеток та інших електроприладів слід негайно вимкнути їх і вжити необхідних заходів щодо приведення до безпечного стану. Документи, папір та інші горючі матеріали слід зберігати на відстані від електрощитів, електрокабелів, проводів – не менше 1,0 м, від світильника – 0,5 м від приладів опалення – 0,25 м. Засоби протипожежного захисту слід утримувати у справному стані.

Всі працівники в офісі зобов'язані вміти користуватися наявними вогнегасниками, іншими первинними засобами пожежогасіння та внутрішніми пожежними кранами, знати місця їх розташування. Відстань від найвіддаленішого місця офісу до вогнегасника не повинна перевищувати 20 м. У службових приміщеннях забороняється: Улаштовувати тимчасові електромережі, застосовувати саморобні некалібровані плавкі вставки в запобіжниках і саморобні подовжувачі, які не відповідають вимогам Правил улаштування електроустановок, прокладати електричні проводи безпосередньо по горючій основі, експлуатувати світильники зі знятими ковпаками (розсіювачами); Використовувати вимикачі та штепсельні розетки для розвішування на них одягу або інших предметів, обгортати електролампи й світильники папером, заклеювати ділянки електропроводки горючою тканиною, папером; Користуватись побутовими електрокип'ятильниками, електрочайниками й т.і. (окрім спеціально відведених та обладнаних місць). Залишати без нагляду ввімкнені в електромережу кондиціонери, комп'ютери,

лічильні та друкарські машинки тощо; Захаращувати підходи до засобів пожежогасіння, використовувати пожежні крани, рукава й пожежний інвентар не за призначенням; Використовувати для зберігання документів, різних матеріалів, предметів та інвентарю шафи (інші) інженерних комунікацій; Курити (окрім спеціально відведених місць, позначених написом «Місця для куріння» та обладнаних урною або попільничкою з негорючого матеріалу), проводити зварювальні та інші роботи без оформлення відповідного дозволу (наряду - допуску) застосовувати ЛЗР;

Відповідальний за пожежну безпеку після закінчення роботи зобов'язаний: Ретельно оглянути всі службові приміщення, пересвідчитись у оту, що немає порушень, які можуть призвести до виникнення пожежі; Вимкнути освітлення, знеструмити прилади та обладнання (за винятком електрообладнання, яке відповідно до вимог технології має працювати цілодобово). 3. Обов'язки та дії працівників у разі виникнення пожежі. У разі виявлення пожежі (ознак горіння) кожен працівник зобов'язаний: негайно повідомити про це Державну пожежну охорону (номер телефону для виклику «101», вказати при цьому адресу, кількість поверхів, місце виникнення пожежі, наявність людей, а також своє прізвище); Повідомити про пожежу адміністрацію, також чергового в нічний час (за наявності); Організувати евакуацію людей і матеріальних цінностей; Вимкнути (за необхідності). Струмоприймачі та вентиляцію; Розпочати гасіння пожежі наявними первинними засобами пожежогасіння; Організувати зустріч підрозділів пожежної охорони та надати їм допомогу під час гасіння пожежі.

4.2 Вимоги до робочого місця офісного працівника

Кожен має право на належні, безпечні і здорові умови праці. Це гарантує нам Конституція України (ч. 4 ст. 43). Більш детальні вимоги щодо охорони праці, зокрема охорони праці офісних працівників, містять Кодекс законів про працю, Закон України «Про охорону праці», а також інші підзаконні нормативно-правові акти. У відповідності до вимог ст. 153 Кодексу законів про працю України та ст. 6 Закону України «Про охорону праці» на всіх підприємствах, в установах, організаціях створюються безпечні і нешкідливі умови праці. Забезпечення

безпечних і нешкідливих умов праці покладається на власника або уповноважений ним орган. Умови праці на робочому місці, безпека технологічних процесів, машин, механізмів, устаткування та інших засобів виробництва, стан засобів колективного та індивідуального захисту, що використовуються працівником, а також санітарно-побутові умови повинні відповідати вимогам нормативних актів про охорону праці. Власник або уповноважений ним орган повинен впроваджувати сучасні засоби техніки безпеки, які запобігають виробничому травматизму, і забезпечувати санітарно-гігієнічні умови, що запобігають виникненню професійних захворювань працівників. Стаття 158 Кодексу законів про працю України встановлює обов'язок власника або уповноваженого ним органу вживати заходів щодо полегшення і оздоровлення умов праці працівників шляхом впровадження прогресивних технологій, досягнень науки і техніки, засобів механізації та автоматизації виробництва, вимог ергономіки, позитивного досвіду з охорони праці, зниження та усунення запиленості та загазованості повітря у виробничих приміщеннях, зниження інтенсивності шуму, вібрації, випромінювань тощо. А згідно з ч. 1 ст. 13 Закону України «Про охорону праці» роботодавець зобов'язаний створити на робочому місці в кожному структурному підрозділі умови праці відповідно до нормативно-правових актів, а також забезпечити додержання вимог законодавства щодо прав працівників у галузі охорони праці. Робочі місця офісних працівників, обладнані персональними комп'ютерами (далі – робочі місця), повинні відповідати вимогам «Правил охорони праці під час експлуатації електронно-обчислювальних машин», затверджених Наказом Державного комітету України з промислової безпеки, охорони праці та гірничого нагляду від 26.03.2010 року № 65 (Правила), та «Державних санітарних правил і норм роботи з візуальними дисплейними терміналами електронно-обчислювальних машин», затверджених постановою Головного державного санітарного лікаря України від 10.12.98 N 7 (ДСанПіН 3.3.2-007- 98). Правила поширюються на всіх суб'єктів господарювання незалежно від форм власності, які у своїй діяльності здійснюють роботу, пов'язану з персональними

комп'ютерами, у тому числі на тих, які мають робочі місця, обладнані персональними комп'ютерами і периферійними пристроями. Зазначені нормативно-правові акти встановлюють санітарногігієнічні вимоги до приміщення, в якому розташоване робоче місце, власне до робочого місця, освітлення, рівнів вібрації і шуму, мікроклімату в приміщенні тощо.

4.2.1 Безпека під час роботи з персональним комп'ютером

Щодня перед початком роботи необхідно очищати монітор від пилу та інших забруднень. Після закінчення роботи персональний комп'ютер і периферійні пристрої повинні бути відключені від електричної мережі. У разі виникнення аварійної ситуації необхідно негайно відключити персональний комп'ютер і периферійні пристрої від електричної мережі. Не допускається: виконувати обслуговування, ремонт та налагодження персонального комп'ютеру та периферійних пристроїв безпосередньо на робочому місці оператора; зберігати біля персонального комп'ютеру та периферійних пристроїв папір, будь-які носії інформації (диски, флешки тощо), запасні блоки, деталі тощо, якщо вони не використовуються для поточної роботи; відключати захисні пристрої, самочинно проводити зміни у конструкції та складі персонального комп'ютеру та периферійних пристроїв або їх технічне налагодження; працювати з персональним комп'ютером, у яких під час роботи з'являються нехарактерні сигнали, нестабільне зображення на моніторі тощо; працювати з матричним принтером за відсутності вібраційного килимка та зі знятою (піднятою) верхньою кришкою.

ВИСНОВКИ

Даний проект розроблявся для компанії «Матролікс», який було виконано у середовищі Microsoft Visual Studio 2015 при використанні мови програмування C # та СУБД MS SQL Server 2008, та Web-додатку у месенджері Telegram з використанням мови програмування JavaScript, його фреймворку Node.js та Node Telegram Bot Api та створенням бази даних в MongoDBCompass.

Дана система повинна значно полегшити роботу клієнт-менеджерів та комірників при моніторингу наявності товарів, та обліку залишку товарів, а також спростити ведення звітності по продажам, а саме, забезпечити зручний перегляд даних, додавання та вилучення даних та виконання пошуку та фільтрації потрібних записів за допомогою запитів, та збереження всієї необхідної інформації в одній БД. Збільшити прибуток компанії, підвищення ефективності продажів в інтернеті завдяки залученню нових клієнтів в Web-додаток.

Виконання даної роботи покращило навички та уміння в розробці та проектуванні інтерфейсу користувача у середовищі Microsoft Visual Studio 2015 з використанням об'єктно-орієнтованої мови програмування C# та динамічної мови програмування JavaScript закріпило та поглибило знання з дисциплін «Проектування інформаційних систем» та «Об'єктно-орієнтоване програмування».

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. <https://www.gd.ru/articles/8809-kommercheskiy-otdel>
2. https://jobs.ua/rus/job_description/view/260
3. <https://www.unitcon.ru/stati-i-dokumenty/interesnye-stati/84-personal/706-otsenka-raboty-menedzhera-po-prodazham.html>
4. <https://startpack.ru/application/retailcrm>
5. <https://v8.1c.ru/small.biz/internet-prodazhi-v-1s-unf/>
6. <https://docs.microsoft.com/ru-ru/sql/relational-databases/databases/databases?view=sql-server-ver15>
7. <https://ru.osvita.ua/vnz/reports/management/13733/>
8. <https://uk.wikipedia.org/wiki/>
9. <https://studfile.net/preview/5195286/page:2/>
10. <https://www.vspu.edu.ua/content/instruct/in7/a6.pdf>
11. <https://dnipr.kyivcity.gov.ua/files/2014/11/13/ohoronapr.pdf>
12. <https://uk.wikipedia.org/wiki/JavaScript>
13. Нейгел К., Ивѐн Б.. Professional C# 5.0 and .NET 4.5
14. Методичні вказівки до виконання курсових проектів з дисципліни «Автоматизоване проектування інформаційних систем» для студентів спеціальності 6.050101 «Комп'ютерні науки» денної та заочної форми навчання / Уклад. О.М. М'якшило К.: НУХТ, 2011.
15. Закон України "Про охорону праці" в редакції від 21 листопада 2002 р
16. Законодавство України про охорону праці / Збірник нормативних документів. У 4 т. - К.: Держнагляд охорони праці; Основа, 1995.
17. Автоматизоване проектування інформаційних систем: Методичні вказівки до виконання лабораторних робіт для студентів напряму 6050101 «Комп'ютерні науки» денної та заочної форми навчання. Частина 1 / уклад. О.М. М'якшило, О.В. Харкянен – К.: НУХТ, 2011. – 15 с.
18. Управління ІТ проектами [Електронний ресурс]: лабораторний практикум для студентів напряму підготовки 6.050101 "Комп'ютерні науки"

денної та заочної форм навч. / уклад. О. А. Хлобистова, М. В. Гладка. - К.:
НУХТ, 2013. – 108 с.

ДОДАТКИ

ДОДАТОК А «ФУНКЦІОНАЛЬНА МОДЕЛЬ (AS-IS)»

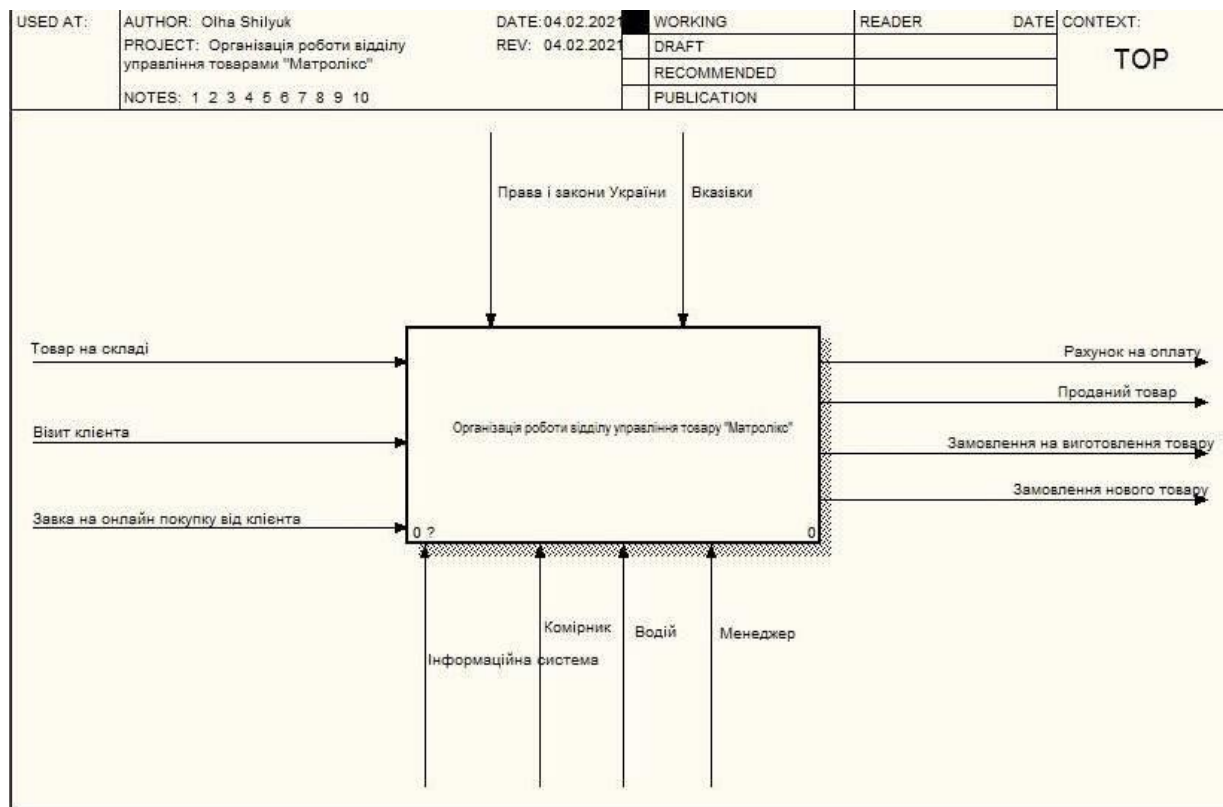


Рис.А.1. Функціональна схема моделі AS-IS

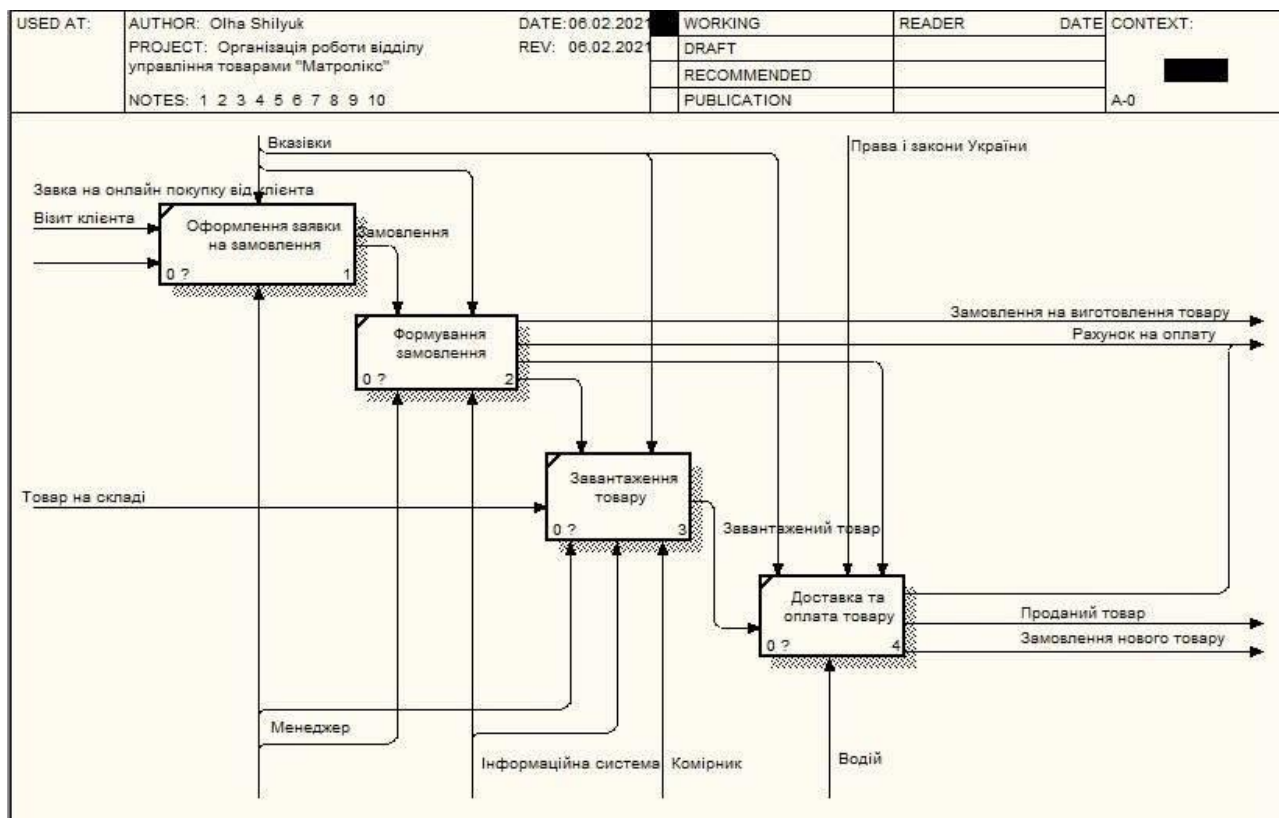


Рис.А.2. Перший рівень декомпозиції AS-IS

ДОДАТОК Б «ФУНКЦІОНАЛЬНА МОДЕЛЬ(ТО ВЕ)»

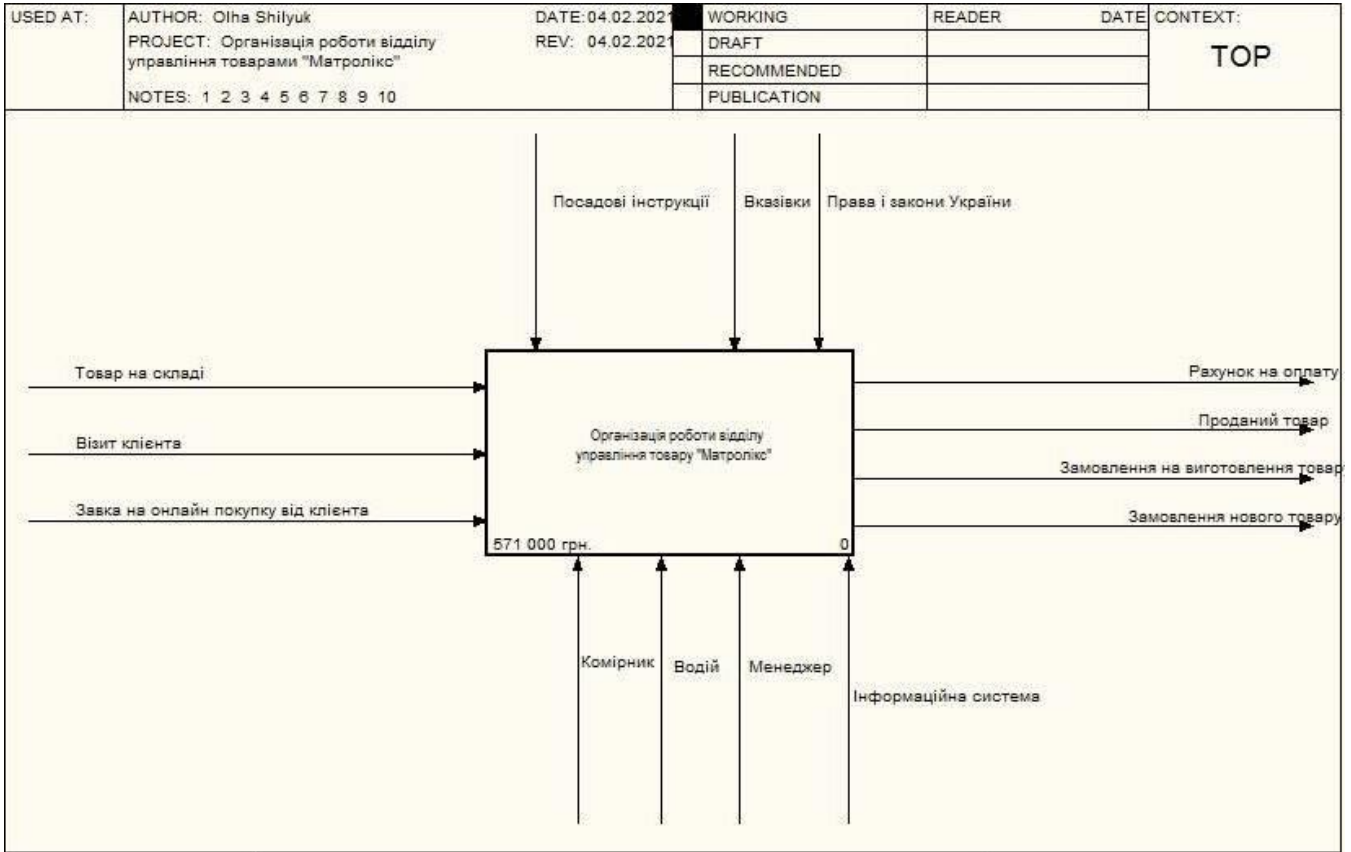


Рис.Б.1. Функціональна модель ТО-ВЕ

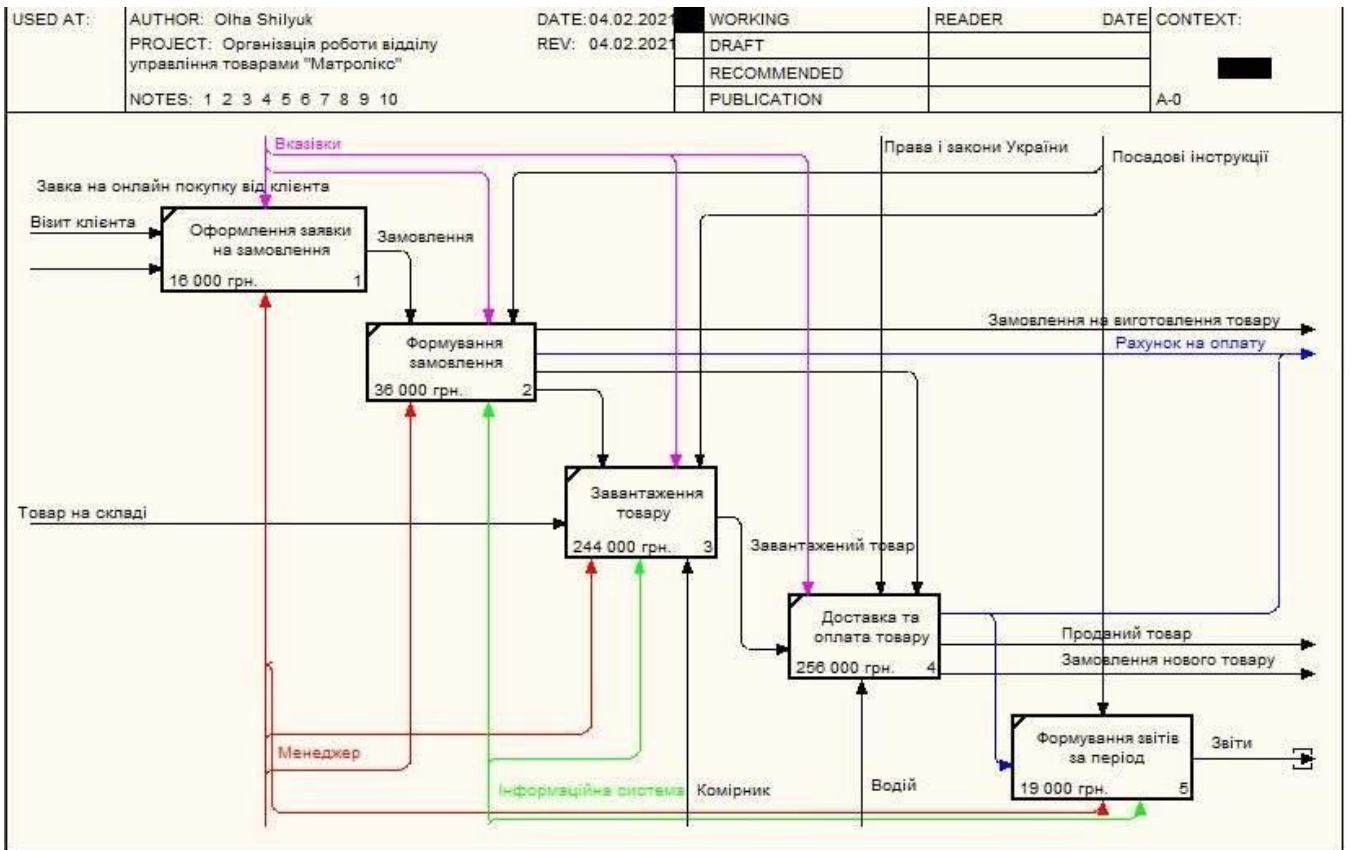


Рис.Б.2. Перший рівень декомпозиції діаграми ТО-ВЕ

ДОДАТОК В «ФІЗИЧНА СХЕМА БАЗИ ДАНИХ»

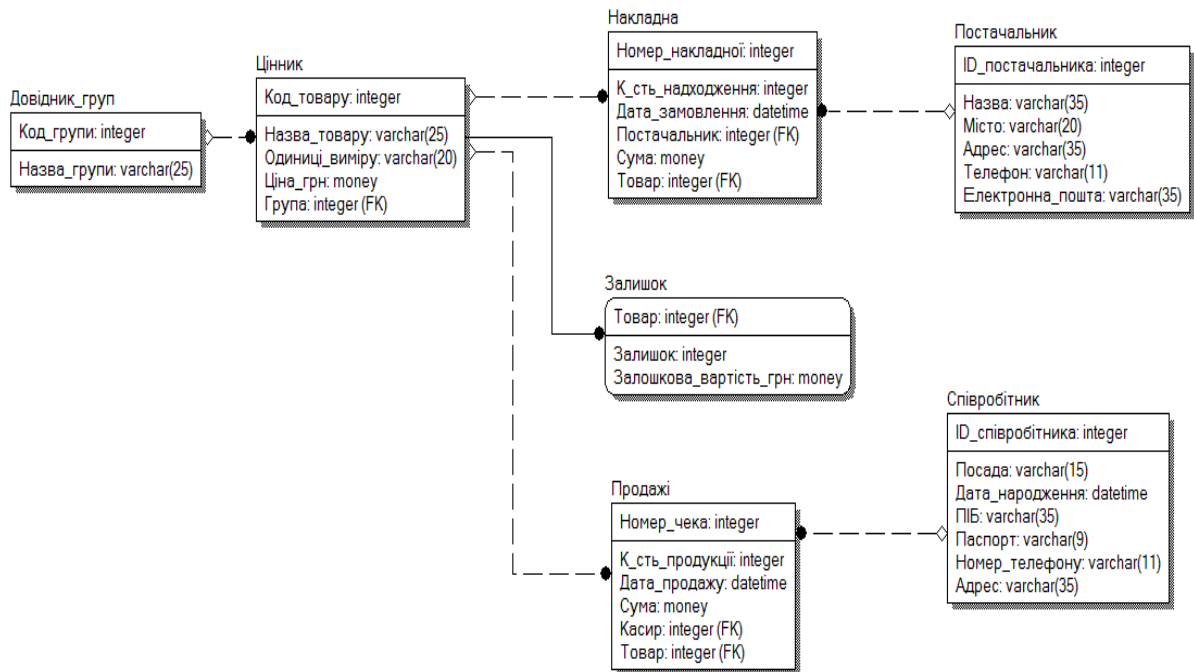


Рис.В.1. Фізична модель БД

ДОДАТОК Д «ФРАГМЕНТИ КОДУ ПРОГРАМИ»

1. Код інформаційної системи

```
private void button1_Click(object sender, EventArgs e)
{
    try
    {
        this.invoice_by_dealerTableAdapter.Fill(this.kURS_OLDataSet.invoice_by_dealer, comboBox1.Text);
    }
    catch (System.Exception ex)
    {
        System.Windows.Forms.MessageBox.Show(ex.Message);
    }
}

private void button2_Click(object sender, EventArgs e)
{
    PrintDialog dlg = new PrintDialog();
    dlg.ShowDialog();
}
}
```

Рис.Д.1. Коди клавіш на формі

```
private void fillToolStripButton_Click(object sender, EventArgs e)
{
    try
    {
        this.tickets_by_dateTableAdapter.Fill(this.kURS_OLDataSet.tickets_by_date, p1ToolStripTextBox.Text, p2ToolStripTextBox.Text);
    }
    catch (System.Exception ex)
    {
        System.Windows.Forms.MessageBox.Show(ex.Message);
    }
}
}
```

Рис.Д.2. Код клавіші пошуку

```
private void снівробітникиToolStripMenuItem_Click(object sender, EventArgs e)
{
    worker N = new worker();
    N.ShowDialog();
}

private void button1_Click(object sender, EventArgs e)
{
    try
    {
        this.ticket_by_workerTableAdapter.Fill(this.kURS_OLDataSet.ticket_by_worker,
        comboBox1.Text);
    }
    catch (System.Exception ex)
    {
        System.Windows.Forms.MessageBox.Show(ex.Message);
    }
}
```

```

private void radioButton1_CheckedChanged(object sender, EventArgs e)
{
    this.залишокBindingSource.Filter = "Залишок <= " + textBox1.Text;
}
private void radioButton2_CheckedChanged(object sender, EventArgs e)
{
    this.залишокBindingSource.RemoveFilter();
}

```

2. Код web-додатку:

```

//привітальне повідомлення
bot.onText(/start/, msg => {
    const text = `Привіт, ${msg.from.first_name} ! Тебе вітає магазин Matrolix!\n\nВ нас ти можеш придбати круті аксесуари, та подаруночки для своїх близьких людей 🎁\n\nНавігація магазину знаходиться трішки нижче Якщо виникнуть запитання по роботі бота або потрібна буде консультація по асортименту - клікай по кнопці "Зв'язатись з менеджером" 🗣️♀`
    //Основне меню
    bot.sendMessage(helper.getChatId(msg), text, {
        reply_markup: {
            keyboard: keyboard.home,
            resize_keyboard: true
        }
    })
})
bot.on('message', msg => {
    const chatId = helper.getChatId(msg)
    switch (msg.text) {
        case keyboardButtons.home.inline:
            bot.sendMessage(chatId, `Виберить категорію товару`, {
                reply_markup: { keyboard: keyboard.inline }
            })
            break
        case keyboardButtons.inline.valentine:
            sendCardsByQuery(chatId, { category: 'valentine' })
            break
    }
}

```

```

case keyboardButtons.inline.bracletes:
  sendCardsByQuery(chatId, { category: 'bracletes' })
  break
case keyboardButtons.inline.office:
  sendCardsByQuery(chatId, { category: 'office' })
  break
case keyboardButtons.inline.pins:
  sendCardsByQuery(chatId, { category: 'pins' })
  break
case keyboardButtons.home.bag:
  showBagCards(chatId, msg.from.id)
  break
case keyboardButtons.back:
  bot.sendMessage(chatId, `Що бажаєте придбати?`, {
    reply_markup: {
      keyboard: keyboard.home,
      resize_keyboard: true
    }
  })
  break
}
})
function sendCardsByQuery(chatId, query) {
  Card.find(query).then(cards => {
    const html = cards.map((f, i) => {
      return ` ${i + 1} </b> ${f.name} - ${f.price} /f${f._id}`
    }).join('\n')
    sendHTML(chatId, html, 'cards')
  })
}
//Детальне зображення картки
bot.onText(/\/f(.+)/, (msg, [source, match]) => {
  const cardId = helper.getItemId(source)
  const chatId = helper.getChatId(msg)

```

```

Promise.all([
  Card.findOne({ _id: cardId }),
  User.findOne({ telegramId: msg.from.id })
]).then(([cards, users]) => {
  let isBag = false
  if (users) {
    isBag = users.card.indexOf(cards._id) !== -1
  }
  const bagText = isBag ? 'Видалити з кошика' : 'Додати у кошик'
  const caption = `Назва: ${cards.name}\nЦіна: ${cards.price}\nОпис товару: ${cards.produ
ct_descr}`
  //Налаштування інлайн-клавіатури
  bot.sendPhoto(chatId, cards.picture, {
    caption: caption,
    reply_markup: {
      inline_keyboard: [
        [
          {
            text: bagText,
            callback_data: JSON.stringify({
              type: ACTION_TYPE.TOGGLE_BAG_CARD,
              cardId: cards._id,
              isBag: isBag
            })
          }
        ],
        [
          {
            text: 'Замовити зараз!',
            url: 'https://t.me/matrolya'
          }
        ]
      ]
    }
  })
})

```

```

    }).catch(e => console.log(e))
  })
  //додавання та видалення з кошику
  function toggleBagCard(userId, queryId, { cardId, isBag }) {
    let userPromise
    User.findOne({ telegramId: userId })
      .then(user => {
        if (user) {
          if (isBag) {
            user.cards = user.cards.filter(cId => cId !== cardId)
          } else {
            user.cards.push(cardId)
          }
          userPromise = user
        } else {
          userPromise = new User({
            telegramId: userId,
            cards: [cardId]
          })
        }
        const answerText = isBag ? 'Видалено' : 'Додано до кошика'
        userPromise.save().then(_ => {
          bot.answerCallbackQuery(queryId, {
            callback_query_id: queryId,
            text: answerText
          })
        }).catch(err => console.log(err))
      })
  }
  //кошик
  function showBagCards(chatId, telegramId) {
    User.findOne({ telegramId })
      .then(user => {
        if (user) {
          Card.find({ _id: { '$in': user.cards } }).then(card => {

```

```
let html
if (card.length) {
  html = card.map(c => {
    return `${c.name} - <b>${c.price}</b>`
  }).join('\n')
  html = `<b>Товари у кошику:</b>\n${html}`
} else {
  html = 'Додайте товар до кошика'
}
sendHTML(chatId, html, 'home')
})
} else {
  sendHTML(chatId, 'Додайте товар до кошика', 'home')
}
}).catch(e => console.log(e))
}
```