РЕФЕРАТ

Кваліфікаційна робота спеціаліста (бакалавра) за темою «Програмна система автоматизації складського обліку комп'ютерної техніки та її комплектуючих» зі спецчастиною «Розробка таблиць бази даних, програмних модулів і інтерфейсу користувача засобами Microsoft Access».

Пояснювальна записка кваліфікаційної роботи містить: 91сторінок, 58 рисунків, 18 таблиць, додатків, використаних джерел.

Об'єкт дослідження роботи – програмна система для автоматизації складського обліку комп’ютерної техніки та її комплектуючих

Предмет дослідження – автоматизації обліку і контролю товарообороту на складі.

Мета кваліфікаційної роботи – розробка програмної системи, яка дозволить оптимально організувати облік товарів, автоматизувати процеси видачі та прийому товарів та підвищити ефективність обслуговування на складі.

Результатом кваліфікаційної роботи – досліджений об’єкт автоматизації, проведений процес проектування програмного засобу за допомогою уніфікованої мови моделювання UML, розроблена програмна система, яка автоматизує робоче місце робітника складу, проведено тестування розробленої програми.

ЗМІСТ

Стор.

[Вступ 8](#_Toc485023984)

[1 Аналіз програмної галузі та постановка задачі 10](#_Toc485023985)

[1.1 Завдання програмного забезпечення для автоматизації складського обліку 10](#_Toc485023986)

[1.2 Типи програмного забезпечення для автоматизації складського обліку 13](#_Toc485023987)

[1.2.1 Desktop (Настільні) 13](#_Toc485023988)

[1.2.2 Web-based (Веб-додатки) 13](#_Toc485023989)

[1.2.3 Однокористувацькі 13](#_Toc485023990)

[1.2.4 Стратегії 14](#_Toc485023991)

[1.3 Проблеми автоматизації 14](#_Toc485023992)

[1.4 Аналіз існуючих рішень 14](#_Toc485023993)

[1.5 Огляд програм складського обліку 15](#_Toc485023994)

[1.5.1 Коротка характеристика програми "СуперСклад" 15](#_Toc485023995)

[1.5.2 Коротка характеристика програми "Атонекс" Склад 16](#_Toc485023996)

[1.5.3 Коротка характеристика програми "Клас365" 17](#_Toc485023997)

[2 Проектування програної системи 19](#_Toc485023998)

[2.1 Проектування об’єктних модулів 19](#_Toc485023999)

[2.1.1 Діаграма прецендентів користувача. 19](#_Toc485024000)

[2.1.2 Діаграма прецендентів ролі користувача. 19](#_Toc485024000)

[2.1.3 Діаграма прецендентів ролі адміністратора. 20](#_Toc485024000)

[2.1.4 Діаграма варіантів використання для складського обліку. 21](#_Toc485024000)

[2.1.5 Діаграма об’єктів для складського обліку, об’єкт користувач.. 21](#_Toc485024000)

[2.1.6 Діаграма об’єктів для складського обліку, об’єкт БД товарів.. 22](#_Toc485024000)

[2.1.7 Діаграма об’єктів для складського обліку, об’єкт бланк документа товарів на складі…………………………………………………………………22](#_Toc485024000)

[2.1.8 Діаграма класів для складського обліку.. 24](#_Toc485024000)

[2.1.9 Діаграма послідовності для складського обліку. 24](#_Toc485024000)

[2.1.10 Діаграма послідовності для об’єкту складського обліку БД товарів.. 25](#_Toc485024000)

[2.1.11 Діаграма послідовності для об’єкту складського обліку бланку документа.. 26](#_Toc485024000)

[2.1.12 Діаграма послідовності для складського обліку... 27](#_Toc485024000)

[2.1.13 Діаграма взаємодії для класу користувач... 27](#_Toc485024000)

[2.1.14 Діаграма взаємодії для класу БД товарів... 28](#_Toc485024000)

[2.1.15 Діаграма взаємодії для класу бланк документа.. 28](#_Toc485024000)

[2.1.16 Діаграма взаємодії для складського обліку.. 29](#_Toc485024000)

[2.1.17 Діаграма діяльності для складського обліку. 30](#_Toc485024000)

[2.1.18 Діаграма розгортання для складського обліку.. 31](#_Toc485024000)

[2.2 Вимоги до програмного і апаратного забезпечення 31](#_Toc485024001)

[3 Розробка програмного засобу 33](#_Toc485024004)

[3.1 Розробка структури бази даних 33](#_Toc485024005)

[3.2 Запити до БД 37](#_Toc485024006)

[3.4 Створення форм 48](#_Toc485024007)

[3.5 Створення звітів 51](#_Toc485024008)

[4 Результати роботи 54](#_Toc485024009)

[4.1 Опис роботи розробленої програми 54](#_Toc485024010)

[4.2 Автоматизація складського обліку 54](#_Toc485024011)

[4.2.1 Група комплектуючих 54](#_Toc485024012)

[4.2.2 Додати контрагента 55](#_Toc485024013)

[4.2.3 Договір 56](#_Toc485024014)

[4.2.4 Користувач 57](#_Toc485024015)

[4.2.5 Прихід-розхід 58](#_Toc485024016)

[4.2.6 Огляд контрагенту 58](#_Toc485024017)

[4.2.7 Тип операції 59](#_Toc485024018)

[4.2.8 Роль користувача 60](#_Toc485024019)

[5 Тестування програмної системи 62](#_Toc485024020)

[5.1 Тестування форми Form1 62](#_Toc485024021)

[5.2 Тестування форми контрагент 64](#_Toc485024022)

[5.3 Тестування форми договір 65](#_Toc485024023)

[Висновки 67](#_Toc485024024)

[Список викоростаних джерел 68](#_Toc485024025)

[Додаток А Лістінг програми 69](#_Toc485024025)

[Додаток Б Презентація для захисту 72](#_Toc485024025)

[Додаток В Екрані форми програми 83](#_Toc485024025)

[Додаток Г Зауваження нормоконтролера 87](#_Toc485024025)

# ВСТУП

На сьогоднішній день сучасний склад - складне технічне спорудження, призначене для управління запасами на різних ділянках логістичного ланцюга і виконання конкретних функцій по зберіганню і перетворенню матеріального потоку в цілому.

У кожного підприємства є проблеми в роботі складського обладнання, їх може бути безліч, і всі разом вони призводять до значних сукупних втрат, змушуючи керівництво задуматися про шляхи їх подолання. Один з таких шляхів - запровадження автоматизованої системи[1]

В умовах активного використання інформаційних технологій, розвитку комунікаційних мереж, розширення складських приміщень, актуальним залишається питання забезпечення відповідного рівня обслуговування користувачів. Існування сучасного світу неможливе без використання розвинутої структури інформаційних технологій. Використання комп'ютерів в області складських підприємств прискорює та підвищує рівень обслуговування користувачів. Сучасні автоматизовані технології відкривають нові можливості великих масивів інформації, що надається базами даних, шляхом створення нових спеціалізованих інформаційних продуктів[2]

Сучасні робітники складу затрачують багато ресурсів на виконання роботи, тому введення автоматизованої програми допоможе їм, швидше працювати та менше втомлюватися. В контексті сказаного, проблема вивчення автоматизації складського обліку в інформаційному обслуговуванні користувачів має велике значення і є актуальною на сьогодні.

Мета – розробка програмної системи для автоматизації обліку і контролю товару на складі, автоматизувати процеси прийому та видачі товарів складу, підвищити ефективність обслуговування клієнтів і як наслідок збільшити користувачів складу.

Організаційна стратегія інтеграції виробництва і операцій, управління трудовими ресурсами, орієнтована на безперервну балансування і оптимізацію ресурсів підприємства за допомогою спеціалізованого інтегрованого пакета прикладного програмного забезпечення, що забезпечує загальну модель даних і процесів[3]

Виходячи з актуальності теми, предмета і мети дослідження визначені задачі роботи:

* вивчити об’єкт дослідження;
* провести аналіз та виявити проблеми автоматизації складського обліку;
* сформувати вимоги до програмної системи та описати процес її проектування;
* розробити структуру бази даних, програмний засіб для автоматизації обліку складу комп’ютерної техніки та її комплектуючих;
* провести тестування розробленої програмної системи[4]

# 1 АНАЛІЗ ПРОГРАМНОЇ ГАЛУЗІ ТА ПОСТАНОВКА ЗАДАЧІ

## 

## 1.1 Завдання програмного забезпечення для автоматизації складського обліку.

## Розглянемо основні причини, які спонукають торгові підприємства впроваджувати у себе системи автоматизації складського обліку і управління складом. Найбільш часто зустрічаються три укрупнені групи завдань, які вирішуються за допомогою автоматизації:[5]

## - базові проблеми (це задоволення потреби в повної, своєчасної та достовірної інформації про товар, забезпечення достатньої швидкості виконання складських операцій при підтримці належної якості роботи з товаром, організація системи контролю персоналу і отримання даних для обґрунтованої мотивації співробітників);

## - проблеми переїзду складу;

## - проблеми розвитку складського господарства.

## Завдання базового рівня стоять перед усіма підприємствами незалежно від їх профілю, однак для торгових компаній потреба в їх вирішенні вище, ніж у виробничників. Потреба в інформації, створення «прозорого» складу - найперше завдання автоматизації складського господарства торгового підприємства. Інформація про реальну кількість товарів на складі, яку може надати складська система, критично важлива при розподілі товарів на замовлення та їх резервування за багатьма покупцями[6]

Корпоративна інформаційна система найчастіше або не має в своїй структурі подібної функціональності, або, як у випадку з реальними залишками, не встигає відображати фактичний стан складу в кожен момент часу. Найчастіше задоволення перерахованих потреб буває досить, щоб ефективність роботи складу помітного зросла. Потім вже підприємство може приступити до роботи над розвитком свого складського обладнання

Окремим пунктом слід позначити проблему переїзду складу на нове місце. З цим завданням найчастіше стикаються саме торговельні підприємства як найбільш мобільні. Лише досить великі організації можуть дозволити собі придбати склад у власність. У цих випадках впровадження системи автоматизації представляється вдалим виходом з положення. Однак необхідно відзначити, що автоматизація складу в момент переїзду має свої особливості. Звичайно потрібно виконати проект в стислі терміни у відносно невизначеною середовищі (новий склад, нове обладнання, нові технології вантажопереробки та ін.).

Зазвичай виділяють наступні етапи автоматизації складу:

1. Постановка завдання. В даний процес повинні бути залучені відразу кілька підрозділів компанії, діяльність яких має те чи інше відношення до подій на складі процесам: менеджери з продажу, бухгалтерія, IT-підрозділ, логісти та т.д., а також самі розробники.

2. Вибір постачальника та програмного забезпечення. Тут необхідно звернути увагу на співвідношення ціна-якість, термін використання продукту на ринку і на конкретні розроблені рішення[7]

3. Основні моменти, на які слід звернути увагу при виборі ПО:

- Оснащеність повної довідкової системою. Це необхідно для швидкого освоєння програми працівниками. Наявність гарної довідкової системи вказує на серйозність підходу до своєї діяльності фірми-виробника програми, тому що розробка документації вимагає чималих витрат. Набагато, зручно користуватися повною довідковою системою, а не простим описом в текстовому файлі, це набагато ускладнить процес при необхідності використання довідкової інформації під час роботи програми.

- Здатність формувати звіти. Одним з головних вимог є отримання в наочному і зрозумілому вигляді звітів. Тому необхідно заздалегідь визначитися на які питання повинна вміти відповідати програма відповідно до специфіки діяльності фірми. Однак, складно знайти програму, яка зможе дати відповіді на всі питання, але можливо врахувати даний факт при замовленні програми.

- Можливість зручного введення даних в програму. Деякі програми влаштовані так, що для зазначення найменування товару необхідно відкрити форму або вікно для вибору, значить, доведеться при необхідності ввести 150 найменувань товару, відкривати 150 раз форму. В інших програмах дана операція може зажадати відкривати дві форми, а, отже, процес введення буде займати багато часу, і інформація буде містити помилки. Розробники пропонують безліч рішень для швидкого введення даних, одним з них є можливість створення списку, з якого можна буде вибирати необхідні.

- Здатність відображення програмою реального стану справ. Наприклад, товари на складі можуть бути бракованими, некондиційними і т.д. Також можуть мати і інші параметри (артикул). Якщо програма не мати можливість вести облік таких станів, то і відповідей на питання, звіти отримати буде неможливо. На багатьох підприємствах необхідно, щоб звіти були як в цінах з ПДВ, так і без урахування ПДВ, проте не всі програми це роблять. Не варто вимагати від програми вирішення будь-яких ситуацій, досить щоб вона дозволяла вводити примітки та позначки. Є програми, що передбачають облік кожної партії товару з присвоєнням коду або номера. На практиці досить часто необхідно зробити звіт по партіях, а також це дозволяє вести більш строгий облік[8]

- Можливість створення резервної копії бази даних. Всі програми складського обліку використовують бази даних при роботі, однак, при перебою живлення і по інших технічних причин є ймовірність пошкодження даних. Ця функція полегшить відновлення втраченої інформації.

3. Розробка дослідного екземпляра програмного забезпечення. Виходячи з формулювання завдання програмісти розробляють ПО, саме, тому дуже важливий перший етап автоматизації.

4. Організація IT-структури (тобто установка необхідного обладнання).

5. Виділення тестового зразка. Тестування ПЗ, найважливіший етап розробки.

6. Впровадження. Після виправлень всіх нюансів починається впровадження системи.

Таким чином, завдяки автоматизації можна вирішити такі завдання:

- Систематизувати номенклатурний довідник;

- Можливість отримувати інформацію про наявність товару на складі в режимі реального часу;

- Управління складським простором;

- Збільшення числа складських операцій;

- Підвищення продуктивності складу в цілому.

## 

## 1.2 Типи програмного забезпечення для автоматизації складського обліку

## 1.2.1 Desktop (Настільні)

Програмне забезпечення знаходиться на робочому столі кожного користувача. Це надає найбільш гнучкий інтерфейс. Такі програми зазвичай дозволяють зберігати інформацію в файл, який надалі може бути викладений в загальний доступ для інших користувачів або ж дані зберігаються в центральній базі даних[9]

## 

## 1.2.2 Web-based (Веб-додатки)

Програмне забезпечення є веб-додатком, доступ до якого здійснюється за допомогою браузера.

Плюси і мінуси:

- Доступ може бути здійснений з будь-якого комп'ютера, не потрібна установка додаткових додатків;

- Простий контроль доступу;

- Багатокористувацький доступ;

- Тільки одна програма, яка встановлена на центральному сервері;

- Швидкість роботи нижче, ніж у звичайних додатків;

- Проблеми з доступом до сервера або його вихід з ладу ведуть до повної недоступності інформації[10]

## 

## 1.2.3 Однокористувацькі

Однокористувацькі системи можуть використовуватися в якості персональних або для управління невеликими магазинами.

## 

## 1.2.4 Стратегії

Призначені для координації дій кількох десятків або сотень користувачів. Зазвичай будуються за технологією клієнт-сервер.

## 1.3 Проблеми автоматизації

Ринок сучасного ПЗ для автоматизації складського обліку представлений широким спектром продуктів, що розрізняються ступенем задоволення вимог до користувача. Користувач сам обирає, яким типом програмного забезпечення користуватись.

Автоматизація складського обліку переслідує 3 основні цілі:

- скоротити час обслуговування покупців;

- організувати облік товарів на складі;

- полегшити проведення інвентаризацій.

Проблеми на шляху підвищення ефективності складського обладнання можна розділити на чотири групи:

- організаційні (розподіл функцій, повноважень, зон відповідальності персоналу і підрозділів);

- технологічні (послідовність і методологія виконання операцій);

- інформаційні (інформаційні системи обліку та обробки даних, засоби комунікації);

- технічні (наявність, адекватність і ступінь зношеності технічних ресурсів).

## 

## 1.4 Аналіз існуючих рішень

Автоматизація підприємств сфери складського обліку дозволяє виконати їх, вирішуючи такі завдання.

1. Збільшує пропускну спроможність складу, тим самим підвищує прибуток.

2. Робить процес інвентаризації простіше для співробітників, облік і перевірка займає в рази менше часу.

3. Додає багато можливостей, завдяки ПЗ товарного системі. Наприклад, дозволяє зручно працювати з ціноутворенням, цінниками і штрих-кодами товарів, автоматично проводити списання, робити аналітику щодо руху товарів і грошових коштів.

4. Фіксація діяльності касирів знижує кількість помилок, пов'язаних з людським фактором, майже повністю виключає ризик обрахунку покупців і злодійства.

## 1.5 Огляд програм складського обліку

Системи управління проектами можна розділити на «безплатні» і «платні» (з доповненим функціоналом).

## 

## 1.5.1 Коротка характеристика програми "СуперСклад"

## Основне призначення програми автоматизація обліку товарно-матеріальних цінностей і грошей, а її відмінна риса простата використання і відсутність бухгалтерських термінів в інтерфейсі.

Основні функції програми :

- Облік наявності і руху товарів. Оформлення і друк первинних складських документів для прийому та відпуску товарів.

- Облік наявності і руху грошових коштів в готівковій та безготівковій формі. Оформлення і друк касових документів.

- Формування і друк звітів про рух товарів в кількісному і сумовому вираженні, звітів про взаємодію з постачальниками і покупцями, звітів про рух та наявність грошових коштів.

- Експорт звітів в "MS Excel" і "OpenOffice.Calc" для подальшого аналізу.

- Формування, друк і графічне представлення аналітичних звітів по динаміці і структурі продажів.

- Ведення до 100 складів в єдиній базі даних з можливістю внутрішнього переміщення товарів між складами.

- Розмежування доступу користувачів до різної інформації, для кожного користувача можна встановити до яких звітів і документів він має доступ.

- Налаштування друкованого виду первинних документів за допомогою вбудованого генератора первинних документів.

- Фіксація і ведення протоколу дій користувачів, завжди відомо, хто і коли зробив остання зміна товарного або грошового документа.

## 

## 1.5.2 Коротка характеристика програми "Атонекс" Склад

Проста і зручна програма обліку в торгівлі. Дана програма оптимально підійде малому і середньому бізнесу, так як поряд з простотою містить весь основний функціонал, необхідний для обліку в торгівлі: складський облік, продаж товару, каса, набір звітів для аналізу фінансових показників, а так же ревізії залишків та інше. Програму можна використовувати абсолютно безкоштовно. Але поряд з цим існує платна версія, яка володіє більш широким функціоналом.

Можливості програми.

Складський облік.

- Прихід товару

- Витрата товару

- Ведення декількох складів

- Операції передачі товару між складами

- Списання товару

- Звітність по складу і складських операціях (прихід, витрата, залишки і т.д.)

Модуль продажу товарів з можливістю підключення фіскального реєстратора.

Модуль Каса. У зв'язку зі скасуванням обов'язкового використання касового апарату, для ведення обліку руху грошових коштів, можна використовувати модуль Каса. Даний модуль дозволяє вести облік по касових операціях в межах підприємства, введення приходу, витрати грошових коштів, закриття змін, звіти касирів по змінах, залишок в касі. Даний модуль призначений в основному для внутрішнього контролю.

Управління персоналом

- Ведення бази даних по співробітниках

- Звіти для розрахунку заробітної плати співробітникам

Звітність Модуль формування звітності під конкретного замовника. Гнучка система звітів дозволяє створювати звіт будь-якої форми за наявними в базі даних без внесення змін в саму програму, що істотно спрощує і здешевлює доопрацювання комплексу під замовника. З програмою поставляється основний набір звітів[11]

## 

## 1.5.3 Коротка характеристика програми "Клас365"

"Клас365" легко налаштовується під завдання користувача. Систему можна застосовувати в роздрібній, оптовій торгівлі або ж в сфера послуг.

Основні можливості:

- Взаємовідносини з клієнтами, вбудований поштовий клієнт, модуль SMS розсилки;

- Робота з усіма популярними сканерами штрих коду, резервування товару на підставі замовлення;

- Завантаження довідника товарів з 1С або Excel;

- Контроль термінів зберігання партій;

- Управління запасами;

- Автоматичне формування накладних;

- Звіт по залишках;

- Необмежена кількість складів;

- Формування файлу платежів для клієнт-банку;[12]

# 2 ПРОЕКТУВАННЯ ПРОГРАМНОЇ СИСТЕМИ

## 2.1 Проектування об’єктних модулів

На рисунку 2.1 зображено які ролі може використовувати користувач.



Рисунок 2.1 - Діаграма прецендентів користувача

На рисунку 2.2 зображено які функції може використовувати користувач.



Рисунок 2.2 - Діаграма прецендентів ролі користувача

На рисунку 2.3 зображено які функції може використовувати адміністратор.



Рисунок 2.3 - Діаграма прецендентів ролі адміністратора

На рисунку 2.4 приведена діаграма використання, спроектована в середовищі Rational Rose. Діаграма варіантів використання показує, основні функції системи (видати товар, заказати товар, оформити заказ, реєстрація, тощо), взаємодію об'єктів при виконанні вищеперелічених функцій. Що дає загальне уявлення про систему. Основними дійовими особами користувач і контрагент.

Користувач виконує наступні дії (у зв’язку з роллю):

- реєструє у системі нового користувача;

- виконує пошук и перегляд інформації про товари на складі;

- перегляд замовленого товару;

- здійснює видачу товару зі складу контрагенту;

- переглядає список товарів їх суму та кількість;

- здійснює операцію повернення товару;

- переглядає контрагентів які здійснювали договори;

- виписує скидки на конкретні договори

- редагує існуючу інформацію о товарах на складі

****

Рисунок 2.4 - Діаграма варіантів використання для складського обліку.

На об’єктах відображені аргументи та методи користувача та його ролі.



Рисунок 2.5 - Діаграма об’єктів для складського обліку, об’єкт користувач.

На об’єктах відображені аргументи та методи для роботи з БД товарів.



Рисунок 2.6 - Діаграма об’єктів для складського обліку, об’єкт БД товарів.

На рисунку 2.7 на об’єктах відображені аргументи та методи для роботи з таблицею договір.



Рисунок 2.7 - Діаграма об’єктів для складського обліку, об’єкт бланк документа товарів на складі.

На рисунку 2.8 приведена діаграма класів. Основними класами є клас користувач, БД товарів, бланк документа. «Користувач» - включає в себе функції у зв’язку з роллю. «Адміністратор» - має більш повний функціонал: «Група комплектуючих», дозволяє додавати, змінювати, та видаляти дані з таблиці «Групи комплектуючих», також має два запити к БД – на вибірку групи комплектуючих, та перехресний групи комплектуючих. «Додати контрагента», дозволяє додавати, змінювати, та видаляти дані з таблиці «Контрагент», також має два запити к БД – підсумковий контрагент і додати контрагента (даний запит був створений для відображення багатофункціональності використовування запитів). «Договір», дозволяє додавати, змінювати, та видаляти дані з таблиці «Договір», також має два запити к БД – на видалення договору та на оновлення договору по скидці.

«Користувач», дозволяє додавати, змінювати, та видаляти дані з таблиці «Користувач», також має два запити к БД – підсумковий користувач і на створення таблиці «Адміністратор». «Прихід-розхід», дозволяє додавати, змінювати, та видаляти дані з таблиці «Прихід-розхід». «Огляд контрагента», дозволяє оглянути контрагентів і договори які вони заключили. «Тип операції», дозволяє додавати, змінювати, та видаляти дані з таблиці «Тип операції», також має запит к БД – на вибірку ти операції. «Роль користувача», дозволяє додавати, змінювати, та видаляти дані з таблиці «Роль користувача». Роль «Користувач», має менший функціонал, «Додати контрагента», «Прихід-розхід» і «Огляд контрагента».

****

Рисунок 2.8 - Діаграма класів для складського обліку.

На рисунку 2.9 зображено послідовність входу у кабінет у зв’язку з роллю користувача.



Рисунок 2.9 - Діаграма послідовності для об’єкту складського обліку користувач.

На рисунку 2.10 зображено послідовність виконання методів програмної системи.



Рисунок 2.10 - Діаграма послідовності для об’єкту складського обліку БД товарів.

На рисунку 2.11 зображено послідовність створення договору складського обліку.



Рисунок 2.11 - Діаграма послідовності для об’єкту складського обліку бланку документа.

Діаграма послідовності представлена на рисунку 2.12 відображає випадок коли користувач оформляє заказ на товар, залишає товар на складі та видає його контрагенту. Виділяються шість об’єктів: Адміністратор, Користувач, Оформлення заказу, Заказ товару, Видача товару, Бланк документа. Стрілками показані взаємодії цих об'єктів, а саме пересилання повідомленнями між цими об'єктами, необхідних для виконання поставленого завдання.



## 

## Рисунок 2.12 - Діаграма послідовності для складського обліку.

На рисунку 2.13 зображені взаємодії між користувачами.



Рисунок 2.13 - Діаграма взаємодії для класу користувач.

На рисунку 2.14 зображені взаємодії у програмній системі між методами з БД.



Рисунок 2.14 - Діаграма взаємодії для класу БД товарів.

На рисунку 2.15 зображені взаємодії у програмній системі між договором та методами користувача.



Рисунок 2.15 - Діаграма взаємодії для класу бланк документа.



Рисунок 2.16 - Діаграма взаємодії для складського обліку.

На рисунку 2.17 приведена діаграма діяльності, яка показує послідовність дій, необхідних для видачі та прийому товару до складу. Після входу до програмної системи користувача зі зв’язком з роллю з’являється певний функціонал за допомогою якого користувач взаємодіє з контрагентом.



Рисунок 2.17 - Діаграми діяльності для складського обліку.

На рисунку 2.18 приведена діаграма розміщення для розроблюваної системи, яка описує апаратну складову необхідну для її функціонування.

Сервер БД є окремою машиною на якій стоїть База даних MS SQL . За допомогою мережі сервер з'єднується Сервер БД є окремою машиною на якій стоїть База даних MySQL. За допомогою мережі сервер з'єднується з робочою станцією працівників складу, на яких встановлена програма MS Access, яка виробляє всі необхідні операції звертаючись лише до сервера за інформацією з БД.



Рисунок 2.18 – Діаграма розгортання для складського обліку.

## 2.2 Вимоги до програмного і апаратного забезпечення[13]

### Таблиця 2.1 - Вимоги до програмного і апаратного забезпечення.

|  |  |
| --- | --- |
| Компонент | Office 2010 |
| Комп’ютер и процесор | 500 МГц |
| Пам’ять | 256 МБ |
| Жорсткий диск | 3 ГБ |
| Монітор | 1024 × 576\* |

### Таблиця 2.2 - Вимоги до програмного і апаратного забезпечення.

|  |  |
| --- | --- |
| Компонент | Вимога |
| ПК и процесор | Процесор з тактовою частотою 500 МГц або вище; для Microsoft Outlook з диспетчером контактів потрібно процесор з тактовою частотою 1 ГГц. |
| Пам’ять | Не менш 256 МБ оперативної пам'яті; для роботи з графікою і використання миттєвого пошуку Microsoft Outlook, Microsoft Outlook з диспетчером контактів, Communicator, а також інших розширених функцій рекомендується 512 МБ оперативної пам'яті. |
| Жорсткий диск | 3,5 ГБ вільного дискового простору |
| Монітор | Монітор з роздільною здатністю 1024 х 768 або більше. |
| Додатково | Вимоги та функціональні можливості продуктів можуть різнитися залежно від конфігурації системи та операційної системи. |

# 3 РОЗРОБКА ПРОГРАМНОГО ЗАСОБУ

## 3.1 Розробка структури бази даних

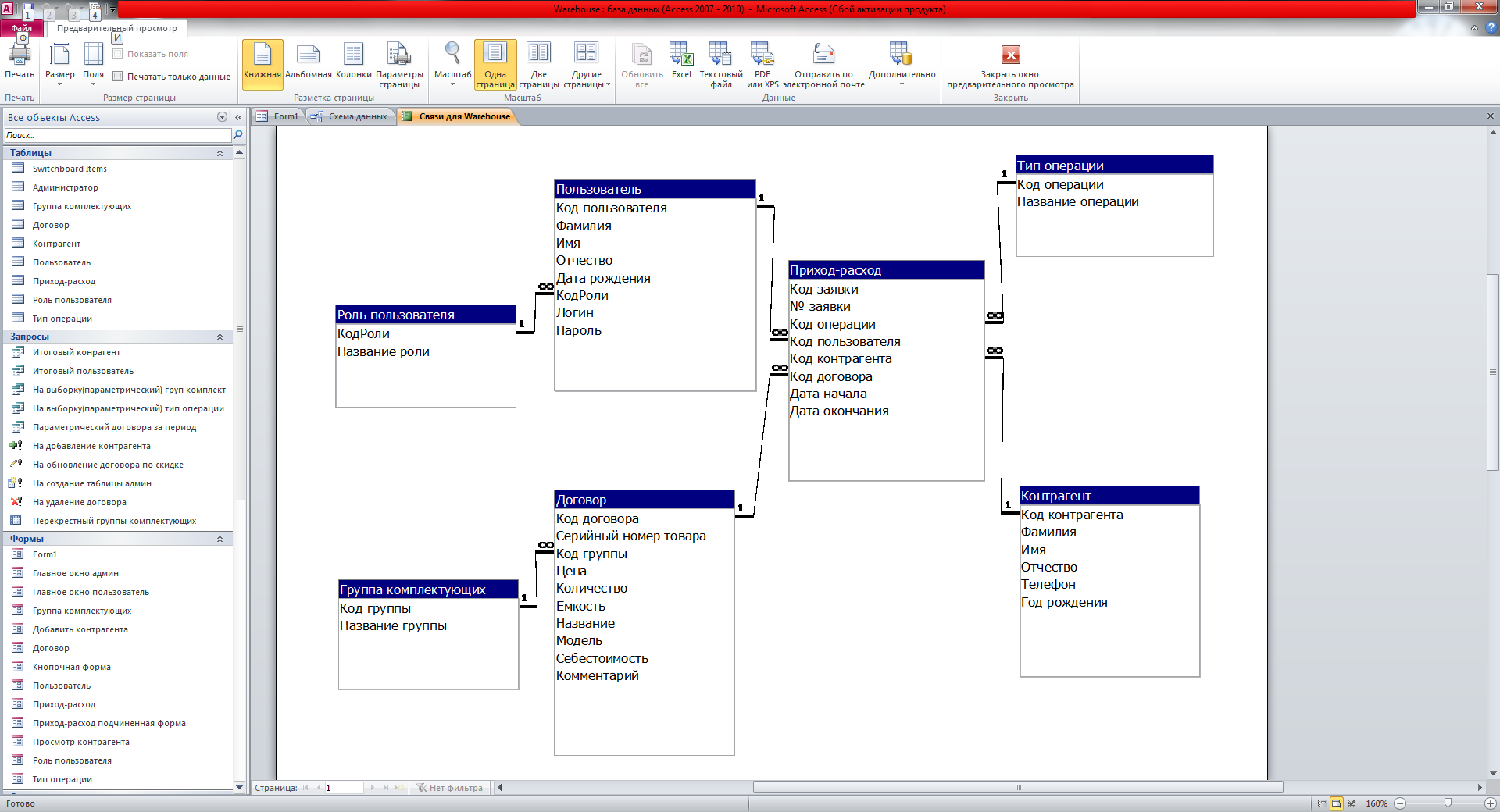


Рисунок 3.1 Схема даних складського обліку.

Розроблена база даних містить такі відомості про користувачів: код користувача, ПІБ, рік народження, код ролі, логін, пароль. Відомості про договір: код договору, серійний номер, код групи, ціна, кількість, ємкість, найменування, модель, собівартість, коментарі. Відомості про контрагентів: ПІБ, телефон, рік народження. Відомості про прихід-розхід: код заявки, № заявки, код операції, код користувача, код контрагента, код договору, дата початку, дата закінчення. Відомості про роль користувача: код ролі, найменування ролі. Відомості про групи комплектуючих: код групи, найменування групи. Відомості про тип операції, код операції, найменування операції[14]

Дані об'єкти предметної області представляються наступними базовими таблицями (табл. 3.1 - 3.7).

Таблиця 3.1 – Користувач

|  |  |  |
| --- | --- | --- |
| Ім'я поля | Тип поля | Примітка |
| Код користувача | Лічильник | Первинний ключ. Номер користувача |
| Прізвище | Текстовий(30) | Прізвище |
| Ім'я | Текстовий(20) | Ім'я |
| По батькові | Текстовий(20) | По батькові |
| Рік народження | Дата/час | Рік народження |
| КодРолі | Числовий | Зовнішній ключ. Найменування ролі |
| Логін | Текстовий(32) | Логін користувача |
| Пароль | Текстовий(16) | Пароль користувача |

У таблиці використовую маску вводу для прізвища, ім'я, по батькові – «>L<??????????????????», яка означає перша літера велика, наступні маленькі. Маска вводу для логіну «AAAaaaaaaaaaaaaaaaaaaaaaaaaaaaaa» - користувач повинен ввести 3 букви та може ввести останні, використовую умову для вводу даних «Not Like "\*[а-я]\*"» яка означає що користувач повинен ввести значення не російською літерою, а англійською. Маска вводу для паролю – пароль, умова вводу даних аналогічна як у логіні.

Таблиця 3.2 – Договір

|  |  |  |
| --- | --- | --- |
| Ім'я поля | Тип поля | Примітка |
| Код договору | Лічильник | Первинний ключ. Номер договору |
| Серійний номер | Текстовий(8) | Серійний номер |
| Ім'я поля | Тип поля | Примітка |
| Код групи | Числовий | Зовнішній ключ. Найменування групи |
| Ціна | Грошовий | Ціна договору |
| Кількість | Числовий | Кількість товару |
| Ємкість | Числовий | Ємкість товару |
| Найменування | Текстовий(30) | Найменування товару |
| Модель | Текстовий(20) | Модель товару |
| Собівартість | Обчислюваний | Собівартість товару |
| Коментарі | Текстовий(150) | Коментарі до товару |

У таблиці використовую маску вводу для серійного номеру «00000000» - користувач повинен ввести цифру [0,9], ціна формат – «0,00"грн."» умова – «>0», кількість, умова – «>0», ємкість формат – «0,00" Гб"» умова – «>0», собівартість вираження [Ціна]\*[Кількість], формат - 0,00"грн.".

Таблиця 3.3 – Контрагент

|  |  |  |
| --- | --- | --- |
| Ім'я поля | Тип поля | Примітка |
| Код контрагента | Лічильник | Первинний ключ. Номер контрагента |
| Прізвище | Текстовий(30) | Прізвище |
| Ім'я | Текстовий(20) | Ім'я |
| По батькові | Текстовий(20) | По батькові |
| Телефон | Текстовий(15) | Телефон |
| Рік народження | Дата/час | Рік народження |

У таблиці використовую маску вводу для прізвища, ім'я, по батькові – «>L<??????????????????», яка означає перша літера велика, наступні маленькі. Маска вводу для телефону «!\(#999")-"00\-00\-00\-000""» - може ввести «+», може ввести три цифри, та повинен ввести інші[15]

Таблиця 3.4 – Прихід-розхід

|  |  |  |
| --- | --- | --- |
| Ім'я поля | Тип поля | Примітка |
| Код заявки | Лічильник | Первинний ключ. Номер заявки |
| № заявки | Текстовий(8) | № заявки |
| Код операції | Числовий | Зовнішній ключ. Найменування операції. |
| Код користувача | Числовий | Зовнішній ключ. Номер користувача. |
| Код контрагента | Числовий | Зовнішній ключ. Номер контрагента. |
| Код договору | Числовий | Зовнішній ключ. Номер договору. |
| Дата початку | Дата/час | Дата початку договору |
| Дата закінчення | Дата/час | Дата закінчення договору |

У таблиці використовую маску вводу для № заявки – «99999999» може ввести цифру [0,9].

Таблиця 3.5 – Роль користувача

|  |  |  |
| --- | --- | --- |
| Ім'я поля | Тип поля | Примітка |
| КодРолі | Лічильник | Первинний ключ. Номер ролі |
| Найменування ролі | Текстовий(20) | Найменування ролі |

Таблиця 3.6 – Тип операції

|  |  |  |
| --- | --- | --- |
| Ім'я поля | Тип поля | Примітка |
| Код операції | Лічильник | Первинний ключ. Номер операції |
| Найменування операції | Текстовий(20) | Найменування операції |

Таблиця 3.7 – Група комплектуючих

|  |  |  |
| --- | --- | --- |
| Ім'я поля | Тип поля | Примітка |
| Код групи | Лічильник | Первинний ключ. Номер групи |
| Найменування групи | Текстовий(25) | Найменування групи |

Довідниками є таблиці «Група комплектуючих», «Тип операції», «Роль користувача».

Таблиці «Роль користувача» і «Користувач» перебувають у відношенні «1 : n» (так як одна роль може бути у кількох користувачів). Таблиці «Група комплектуючих» і «Договір» » перебувають у відношенні «1 : n» (так як одна група може бути у кількох договорах). Таблиці «Користувач» і «Прихід-розхід» перебувають у відношенні «1 : n» (так як один користувач може бути у кількох прихід-розходах). Таблиці «Договір» і «Прихід-розхід» перебувають у відношенні «1 : n» (так як один договір може бути у кількох прихід-розходах). Таблиці «Прихід-розхід» і «Тип операції» перебувають у відношенні «n : 1» (так як у кількох прихід-розходах може бути один тип операції). Таблиці «Прихід-розхід» і «Контрагент» перебувають у відношенні «n : 1» (так як у кількох прихід-розходах може бути один контрагент)[16]

## 3.2 Запити до БД

## Запит на вибірку – для отримання даних із таблиці та виконання яких либо обчислень з даними. (Підсумковий контрагент, підсумковий користувач, на вибірку групи комплектуючих(параметричний), на вибірку тип операції (параметричний), договори за період(параметричний), перехресний групи комплектуючих).

## Запит на змінення – добавити, змінити либо видалити дані в таблицях. (На добавлення контрагента, на обновлення договору по скидці, на створення таблиці «Адміністратор», на видалення таблиці).

## Дані запити предметної області представляються наступними базовими таблицями (табл. 3.8 - 3.16).

3.8 Таблиця - Підсумковий контрагент

|  |  |  |
| --- | --- | --- |
| Поле | Ім'я таблиці | Групова операція |
| Прізвище | Контрагент | Угрупування |
| Ім'я | Контрагент | Угрупування |
| По батькові | Контрагент | Угрупування |
| Собівартість | Договір | Sum |
| Кількість товарів | Договір | Sum |

Для створення запиту використовувалися три таблиці «Контрагент», «Прихід-розхід» і «Договір»

У процесі розробки програмної системи був створений запит Підсумковий контрагент(рис. 3.2)

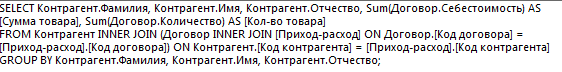


Рисунок 3.2 – Запит підсумковий контрагент

3.9 Таблиця - Підсумковий користувач

|  |  |  |
| --- | --- | --- |
| Поле | Ім'я таблиці | Групова операція |
| Прізвище | Користувач | Угрупування |
| Ім'я | Користувач | Угрупування |
| По батькові | Користувач | Угрупування |
| КодРолі | Користувач | Угрупування |
| Кількість договорів | Договір | Count |

Для створення запиту використовувалися три таблиці «Користувач», «Прихід-розхід» і «Договір»

У процесі розробки програмної системи був створений запит Підсумковий користувач(рис. 3.3)

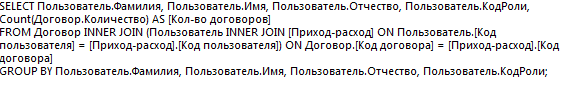


Рисунок 3.3 – Запит підсумковий користувач

3.10 Таблиця - На вибірку групи комплектуючих (параметричний)

|  |  |  |
| --- | --- | --- |
| Поле | Ім'я таблиці | Умова відбору |
| Прізвище | Контрагент | - |
| Ім'я | Контрагент | - |
| По батькові | Контрагент | - |
| Телефон | Контрагент | - |
| Найменування групи | Група комплектуючих | Ввести найменування групи |

Для створення запиту використовувалися чотири таблиці «Контрагент», «Прихід-розхід» , «Договір» і «Група комплектуючих»

У процесі розробки програмної системи був створений запит На вибірку групи комплектуючих (параметричний) ( рис 3.4)

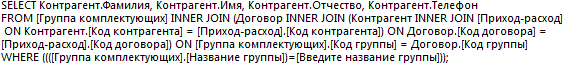


Рисунок 3.4 – Запит На вибірку групи комплектуючих (параметричний)

3.11 Таблиця - На вибірку тип операції (параметричний)

|  |  |  |
| --- | --- | --- |
| Поле | Ім'я таблиці | Умова відбору |
| Прізвище | Користувач | - |
| Ім'я | Користувач | - |
| По батькові | Користувач | - |
| КодРолі | Користувач | - |
| Прізвище | Контрагент | - |
| Ім'я | Контрагент | - |
| По батькові | Контрагент | - |
| Найменування операції | Тип операції | Ввести тип операції |

Для створення запиту використовувалися чотири таблиці «Контрагент», «Прихід-розхід» , «Користувач» і «Тип операції»

У процесі розробки програмної системи був створений запит На вибірку тип операції (параметричний) (рис 3.5)

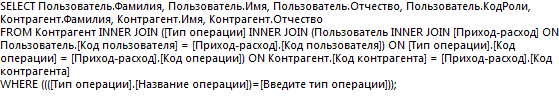


Рисунок 3.5 – Запит На вибірку тип операції (параметричний)

3.12 Таблиця – Договори за період (параметричний)

|  |  |  |
| --- | --- | --- |
| Поле | Ім'я таблиці | Умова відбору |
| Прізвище | Контрагент | - |
| Ім'я | Контрагент | - |
| По батькові | Контрагент | - |
| Прізвище | Користувач | - |
| Ім'я | Користувач | - |
| По батькові | Користувач | - |
| КодРолі | Користувач | - |
| Дата початку | Прихід-розхід | Between [Ввести дату початку] And [Ввести дату закінчення] |
| Дата закінчення | Прихід-розхід | Between [Ввести дату початку] And [Ввести дату закінчення] |
| Код договору | Прихід-розхід | - |

Для створення запиту використовувалися три таблиці «Контрагент», «Прихід-розхід» , «Користувач»

У процесі розробки програмної системи був створений запит Договори за період (параметричний) (рис. 3.6)

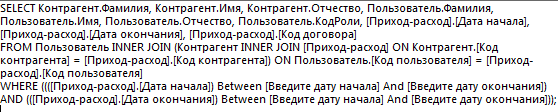


Рисунок 3.6 – Запит Договори за період (параметричний)

3.13 Таблиця - На добавлення контрагента

|  |  |
| --- | --- |
| Поле | Добавлення |
| Ввести прізвище | Прізвище |
| Ввести ім'я | Ім'я |
| Ввести по батькові | По батькові |
| Ввести телефон | Телефон |
| Ввести дату народження | Дата народження |

Для створення запиту використовувалися таблиця «Контрагент»

У процесі розробки програмної системи був створений запит На добавлення контрагента (див. рис 3.7)



Рисунок 3.7 – Запит На добавлення контрагента

3.14 Таблиця - На обновлення договору по скидці

|  |  |  |  |
| --- | --- | --- | --- |
| Поле | Ім'я таблиці | Обновлення | Умова відбору |
| Ціна | Договір | [Ціна]-[Ціна]\*[Ввести скидку в %]/100 | - |
| Код договору | Договір | - | [Ввести код договору] |

Для створення запиту використовувалися таблиця «Договір»

У процесі розробки програмної системи був створений запит На обновлення договору по скидці (рис. 3.8)



Рисунок 3.8 – Запит На обновлення договору по скидці

3.15 Таблиця - На створення таблиці адміністратор

|  |  |  |
| --- | --- | --- |
| Поле | Ім’я таблиці | Умова відбору |
| Прізвище | Користувач | - |
| Ім’я | Користувач | - |
| По батькові | Користувач | - |
| Дата народження | Користувач | - |
| Логін | Користувач | - |
| Пароль | Користувач | - |
| Найменування ролі | Роль користувача | “Адміністратор” |

Для створення запиту використовувалися дві таблиці «Користувач» і «Роль користувача»

У процесі розробки програмної системи був створений запит На створення таблиці адміністратор (рис. 3.8)[17]

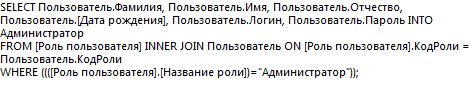


Рисунок 3.8 – Запит На створення таблиці адміністратор

3.15 Таблиця - На видалення договору

|  |  |  |  |
| --- | --- | --- | --- |
| Поле | Ім'я таблиці | Видалення | Умова відбору |
| Код договору | Договір | Умова | [Ввести код договору] |

Для створення запиту використовувалися таблиця «Договір»

У процесі розробки програмної системи був створений запит На видалення договору (рис. 3.9)



Рисунок 3.9 – Запит На видалення договору

3.16 Таблиця - Перехресний групи комплектуючих

|  |  |  |  |
| --- | --- | --- | --- |
| Поле | Ім'я таблиці | Групова операція | Перехресна таблиця |
| Найменування групи | Група комплектуючих | Угрупування | Заголовок строк |
| Найменування | Договір | Угрупування | Заголовок стовбців |
| Кількість | Договір | Sum | Значення |

Для створення запиту використовувалися дві таблиці «Договір» і «Група комплектуючих»

У процесі розробки програмної системи був створений запит Перехресний групи комплектуючих (рис. 3.10)

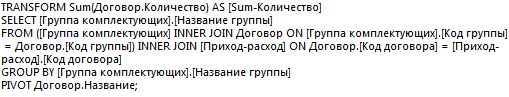


Рисунок 3.10 – Запит Перехресний групи комплектуючих

3.3 Описання запитів програмної системи.

1) Запит «Підсумковий контрагент» – служить для вибірки всіх контрагентів. Запит повертає дані у виді наступних полів:[18]

- ПІБ

- сума

- кількість товарів

2) Запит «Підсумковий користувач» - служить для вибірки всіх користувачів. Запит повертає дані у виді наступних полів:

- ПІБ

- код ролі

- кількість договорів

3) Запит «На вибірку груп комплектуючих (параметричний)» - служить для вибірки конкретної групи. Запит повертає дані у виді наступних полів:

- ПІБ

- телефон

- найменування групи

4) Запит «На вибірку(параметричний) тип операції» – служить для вибірки конкретної операції. Запит повертає дані у виді наступних полів:

- ПІБ користувача

- ПІБ контрагента

- найменування операції

5) Запит «Параметричний договори за період» – служить для вибірки всіх договорів у період конкретної дати. Запит повертає дані у виді наступних полів:

- ПІБ користувача

- ПІБ контрагента

- код ролі

- дата початку

- дата закінчення

- код договору

6) Запит «На додавання контрагенту» – служить для додавання конкретного контрагента Запит повертає дані у виді наступних полів:

- ПІБ

- телефон

- рік народження

7) Запит «На оновлення договору по скидці» – служить для оновлення конкретного договору, за введеною скидкою. Запит повертає дані у виді наступних полів:

- Ціна

- код договору

8) Запит «На створення таблиці адміністратор» - служить для створення конкретних користувачів (адміністраторів). Запит повертає дані у виді наступних полів:

- ПІБ

- дата народження

- логін

- пароль

- найменування ролі

9) Запит «На видалення договору» - служить для видалення конкретного договору. Запит видаляє дані:

- Код договору

- серійний номер

- код групи

- ціна

- кількість

- ємкість

- найменування

- модель

- собівартість

- коментарі

10) Запит «Перехресний груп комплектуючих» – служить для відображення всіх найменування груп, товарів і їх кількостей з якими використовувались операції. Запит повертає дані у виді наступних полів:

- Найменування групи

- Найменування товару

- Кількість товару

## 3.4 Створення форм

## У процесі розробки програмної системи була створена форма введення паролю і логіна(див. рис. 3.11)[19]

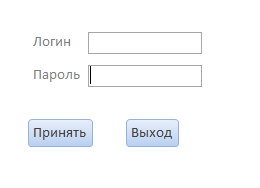


Рисунок 3.11 - початкова форма входу в програмну систему

Для роботи початкової форми програмної системи використовується VBA макрос, який за допомоги порівняння вхідних даних з даними таблиці користувач визначає яке головне вікно відкривати у відповідності з кодом ролі користувача. Якщо данні введені некоректно, тоді видається помилка вводу даних.

У процесі розробки програмної системи була створена форма головного кабінету адміністратора(рис. 3.12)

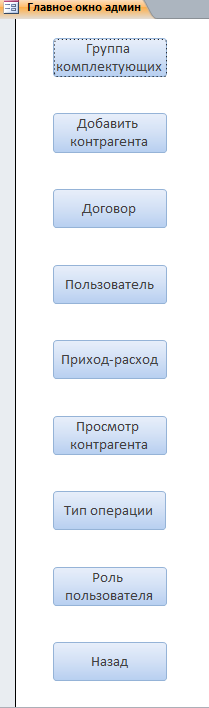


Рисунок 3.12 - «Головне вікно адмін» форма входу в програмну систему при ролі адміністратор.

У процесі розробки програмної системи була створена форма головного кабінету користувача(рис. 3.13)

Користувач під роллю адміністратора має більш розвинений функціонал для роботи на складі.

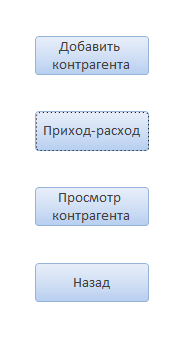


Рисунок 3.13 - «Головне вікно користувач» форма входу в програмну систему під роллю користувача.

У процесі розробки програмної системи була створена форма введення договору(див. рис. 3.14)

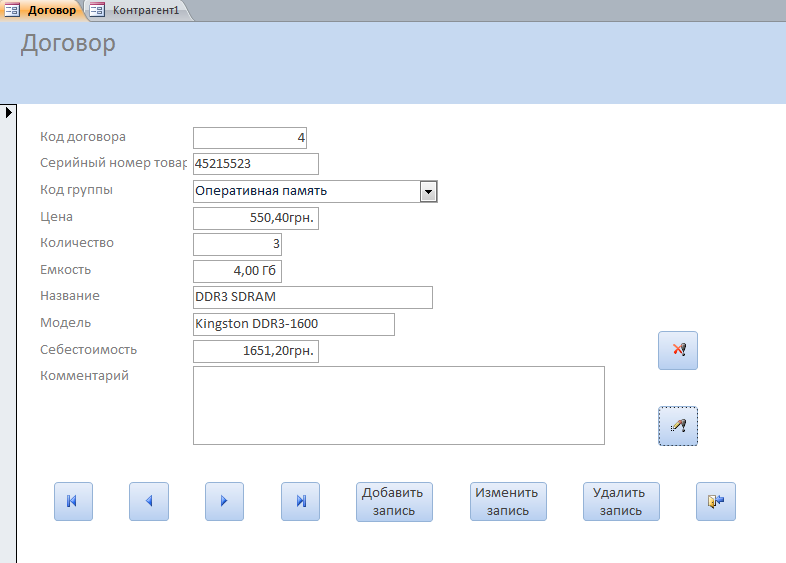


Рисунок 3.14 - «Договір» форма створення, змінення та видалення договору.

У формі використовуються запити, які відображенні у вигляді кнопок. У даній формі можна виконати два запити, на видалення договору та на оновлення ціни товару за указаною скидкою.

У процесі розробки програмної системи була створена форма огляду контрагента (рис. 3.15)

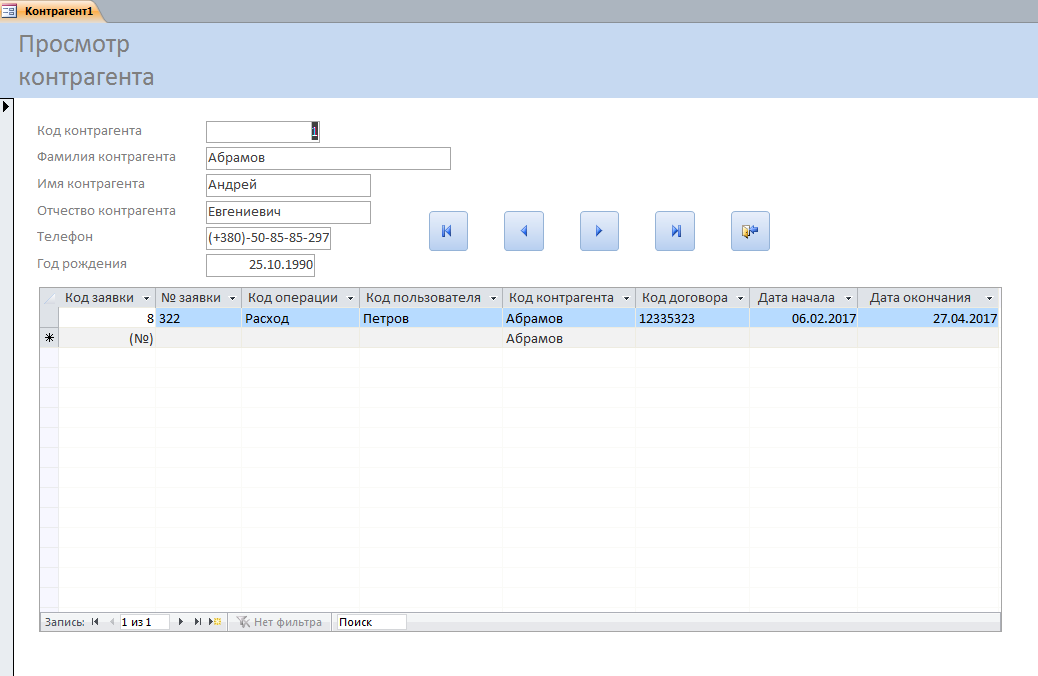


Рисунок 3.15 - «Огляд контрагенту» форма огляду контрагентів та їх договорів.

## 3.5 Створення звітів

У програмній системі були створені звіти на основі запитів, для представлення інформації о товарах і операцій над ними.

У процесі розробки програмної системи був створений звіт підсумковий контрагент(рис. 3.16)

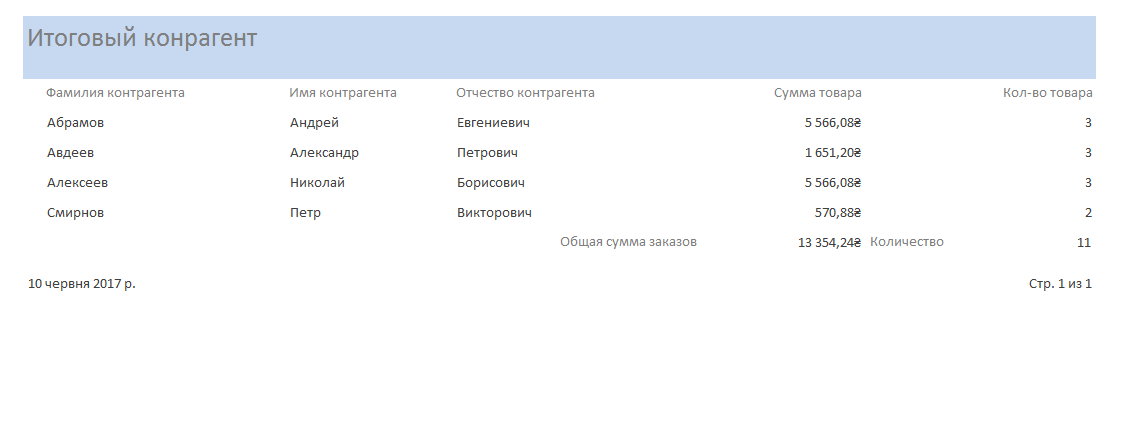


Рисунок 3.16 - «Підсумковий контрагент» звіт о контрагентах та підсумкова сума та кількість товару.

У процесі розробки програмної системи був створений звіт прихід-розхід(див. рис. 3.17)

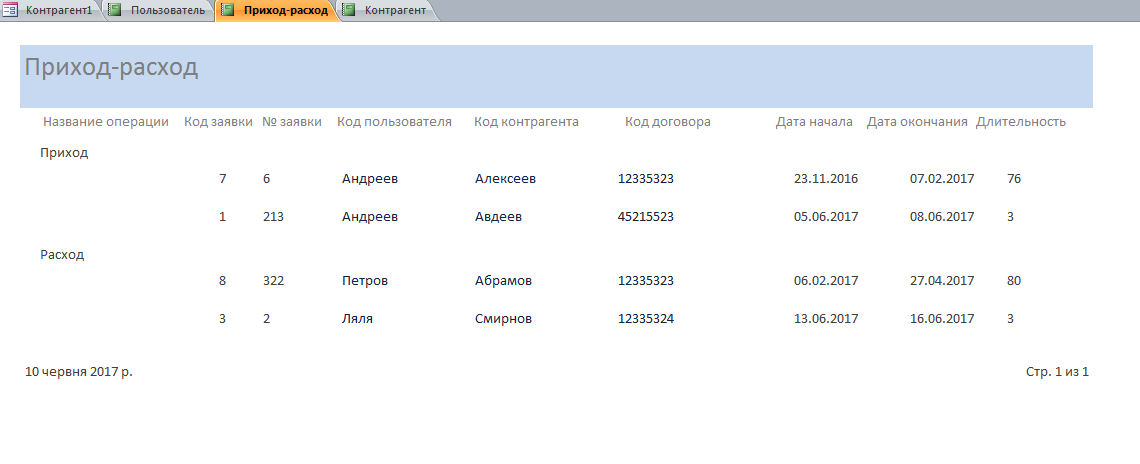


Рисунок 3.17 - «Прихід-розхід» - звіт о контрагентах, коду договору та виконання операції (кількість днів).

У процесі розробки програмної системи був створений звіт контрагент(див. рис. 3.18)



Рисунок 3.18 - «Контрагент» звіт о контрагентах.

# 4 РЕЗУЛЬТАТИ РОБОТИ

## 4.1 Опис роботи розробленої програми

## Для роботи з програмою, запускаємо програму MS Access, відкриваємо базу даних Warehouse. Перед нами з’являється головна форма програмної системи «Form1» (див. рис. 3.11). Користувач повинен ввести дані у поля логіну та паролю, при невірному вводі, вискакую помилка, якщо дані коректні, у зв’язку з роллю, користувач направляється у свій кабінет, де будуть відображенні методи, які він може виконати. Для виконання користувач вибирає метод в кабінеті, натискає на нього, відкривається головне вікно, де користувач може виконувати певні дії. Користувач може додавати, змінювати та видаляти дані з БД.

## 4.2 Автоматизація складського обліку

Якщо користувач увійшов у програму під роллю адміністратора він потрапляє до головної форми адміну див. рис. 3.12.

На формі відображені такі методи: група комплектуючих, додати контрагента, договір, користувач, прихід-розхід, огляд контрагенту, тип операції, роль користувача, назад.

## 4.2.1 Форма «група комплектуючих»

Метод дозволяє добавити, змінити та видалити групу комплектуючих, також метод включає два запити, параметричний на вибірку групи комплектуючих та перехресний запит.

На рисунку 4.1 зображена форма групи комплектуючих. Вона містить список груп комплектуючих. Справу відображені запити, знизу кнопки навігації, кнопки додавання, зміни та видалення груп і кнопка виходу з форми. Для перегляду груп необхідно використовувати кнопки навігації.

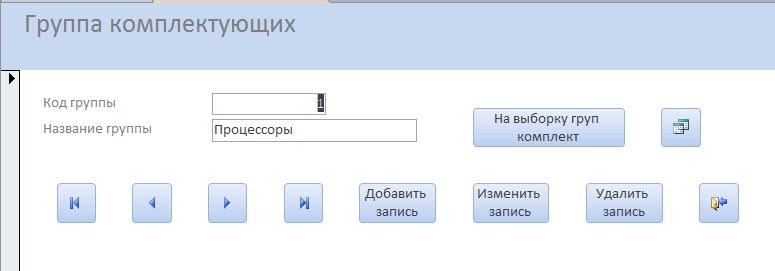


Рисунок 4.1 - Група комплектуючих.

## 4.2.2 Форма «додати контрагента»

Форма дозволяє добавити, змінити та видалити контрагента, також метод включає два запити, підсумковий контрагент, додати контрагента.

На рисунку 4.2 зображена форма контрагент. Вона містить список контрагентів. Справу відображені запити, знизу кнопки навігації, кнопки додавання, зміни та видалення контрагентів і кнопка виходу з форми. Для перегляду контрагентів необхідно використовувати кнопки навігації.

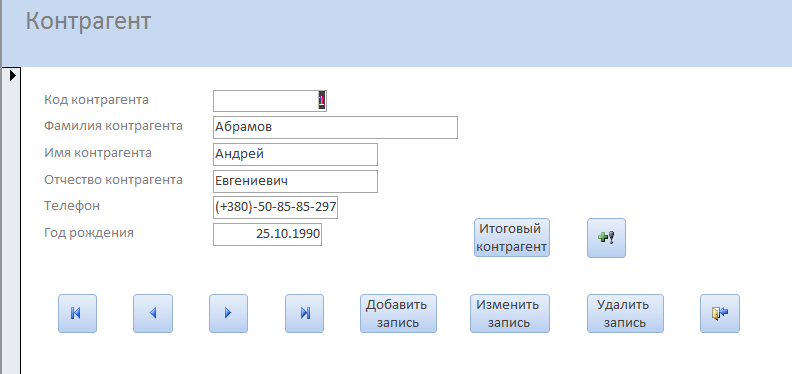


Рисунок 4.2 – Додати контрагента

## 4.2.3 Форма «договір»

Форма дозволяє добавити, змінити та видалити договір, також метод включає два запити, видалити договір, оновити договір по введеної скидці.

На рисунку 4.3 зображена форма договір. Вона містить список договорів. Справу відображені запити, знизу кнопки навігації, кнопки додавання, зміни та видалення договорів і кнопка виходу з форми. Для перегляду договорів необхідно використовувати кнопки навігації.

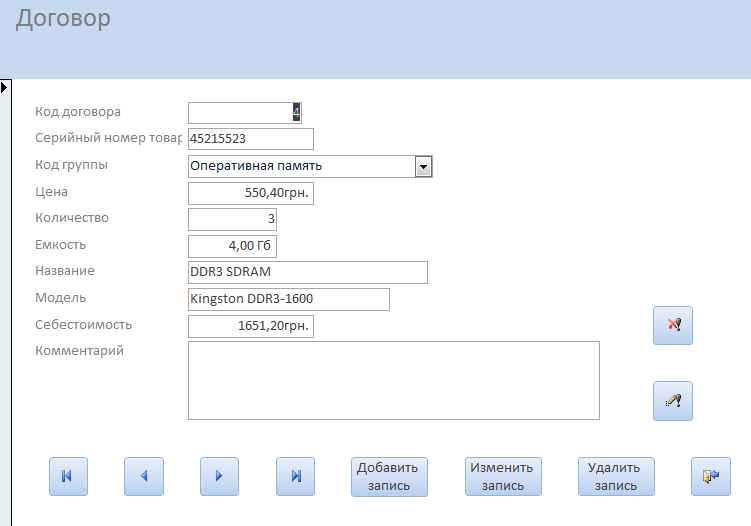


Рисунок 4.3 – Договір

## 4.2.4 Форма «користувач»

Форма дозволяє добавити, змінити та видалити користувача, також метод включає два запити, підсумковий користувач, на створення таблиці «адміністратор».

На рисунку 4.4 зображена форма користувач. Вона містить список користувачів. Справу відображені запити, знизу кнопки навігації, кнопки додавання, зміни та видалення користувачів і кнопка виходу з форми. Для перегляду користувачів необхідно використовувати кнопки навігації.

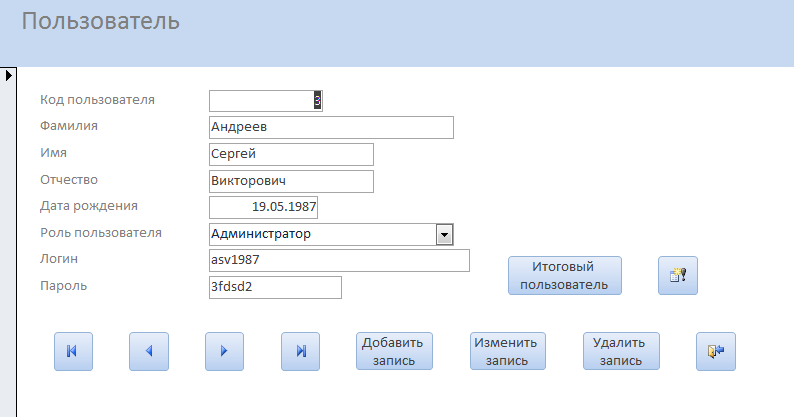


Рисунок 4.4 – Користувач

## 4.2.5 Форма «прихід-розхід»

Форма дозволяє добавити, змінити та видалити заявку.

На рисунку 4.5 зображена форма договір. Вона містить список приходів-розходів. Знизу кнопки навігації, кнопки додавання, зміни та видалення приходів-розходів і кнопка виходу з форми. Для перегляду приходів-розходів необхідно використовувати кнопки навігації.

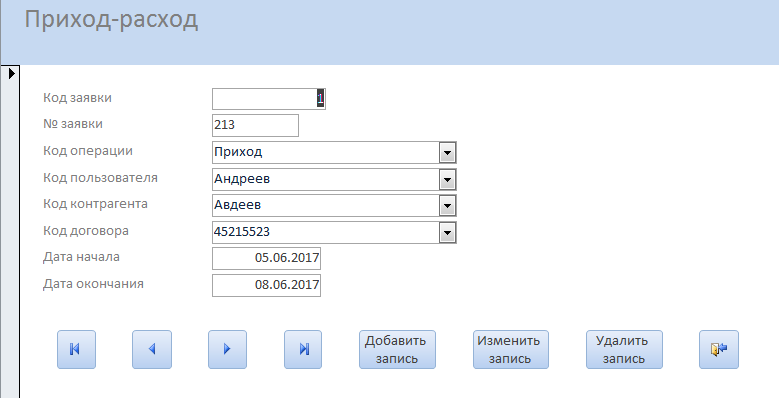


Рисунок 4.5 – форма прихід-розхід

## 4.2.6 Форма «огляд контрагенту»

Форма дозволяє оглянути контрагентів та їх договори.

На рисунку 4.6 зображена форма огляду контрагента. Вона містить список контрагентів. Знизу кнопки навігації, кнопки додавання, зміни та видалення груп і кнопка виходу з форми. Для перегляду контрагентів необхідно використовувати кнопки навігації.

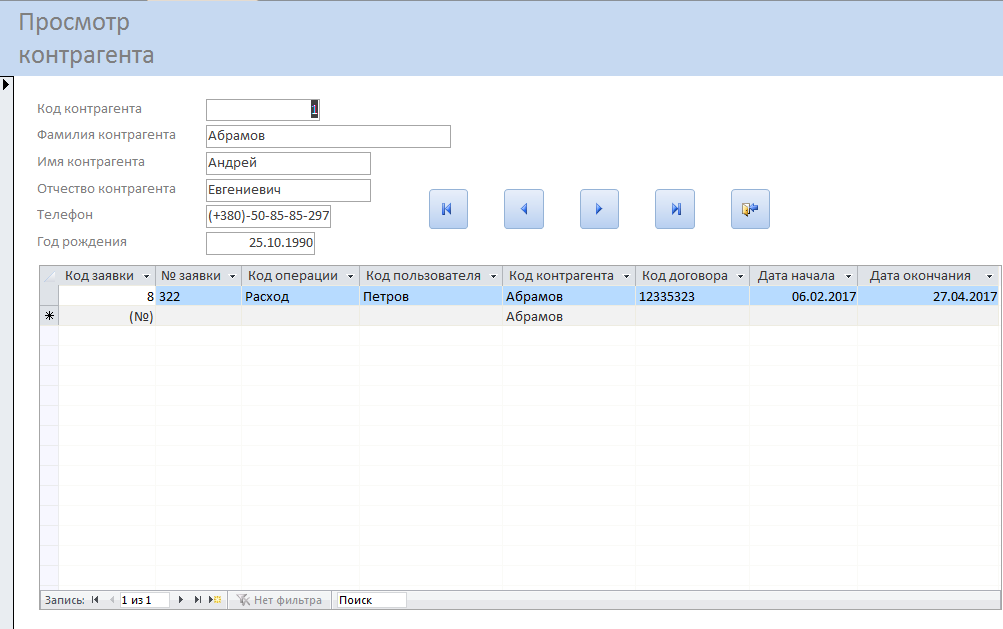


Рисунок 4.6 – Огляд контрагенту

## 4.2.7 Форма «тип операції»

Форма дозволяє добавити, змінити та видалити операцію, також метод включає запит, параметричний на вибірку типу операції.

На рисунку 4.7 зображена форма огляду тип операції. Вона містить список операцій. Справа кнопка запиту, знизу кнопки навігації, кнопки додавання, зміни та видалення груп і кнопка виходу з форми. Для перегляду типів операцій необхідно використовувати кнопки навігації.

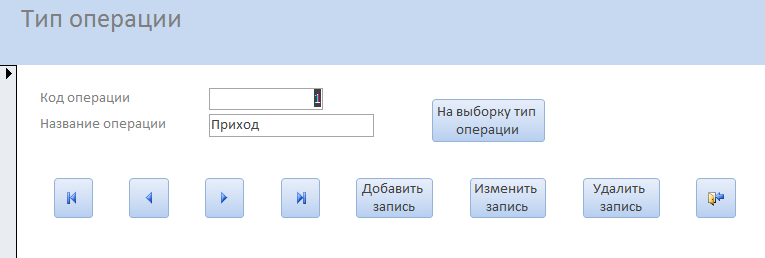


Рисунок 4.7 – Тип операції

## 4.2.8 Форма «роль користувача»

Форма дозволяє добавити, змінити та видалити користувача.

На рисунку 4.8 зображена форма роль користувача. Вона містить список контрагентів. Знизу кнопки навігації, кнопки додавання, зміни та видалення груп і кнопка виходу з форми. Для перегляду ролей користувача необхідно використовувати кнопки навігації.

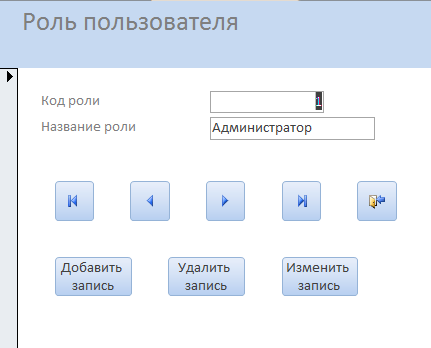


Рисунок 4.8 – Роль користувача

# 5 ТЕСТУВАННЯ ПРОГРАМНОЇ СИСТЕМИ

Для тестування розробленої програми була застосована методика функціонального тестування програмного продукту - тестування методом «чорної скриньки» [20]

При тестуванні програмного продукту методом «чорної скриньки» розглядаються системні характеристики програм, і ігнорується їх внутрішня логічна структура. Основне місце тестів «чорної скриньки» - інтерфейс програмного забезпечення.

Цей тест демонструє:

* як виконуються функції програми;
* як приймаються вихідні дані;
* як виробляються результати;
* як зберігається цілісність зовнішньої інформації.

Тестування «чорної скриньки» (функціональне тестування) дозволяє отримати комбінації вхідних даних, що забезпечують повну перевірку всіх функціональних вимог до програми.

## 5.1 Тестування форми «Form1»

Проведемо тестування функції входу до кабінету користувача. Відкриємо форму та заповнимо не всі поля(рис. 5.1,5.3). Результат тестування зображення на рисунках (рис. 5.2-5.4). Відкриємо форму та заповнимо некоректні дані (рис. 5.5) Відгук о помилці (рис. 5.6).

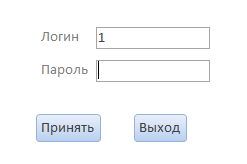


Рисунок 5.1 – Form1

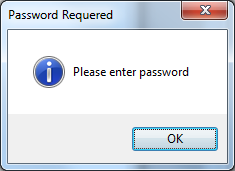


Рисунок 5.2 – тестування без вводу паролю

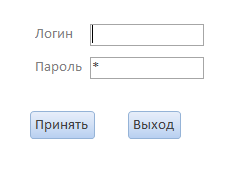


Рисунок 5.3 – Form1

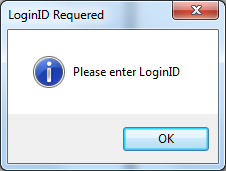


Рисунок 5.4 – тестування без вводу логіну

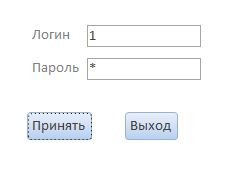


Рисунок 5.5 – Form1

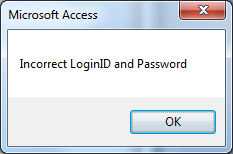


Рисунок 5.6 – тест некоректні дані

## 5.2 Тестування форми контрагент

Проведемо тестування функції додавання контрагента. Відкриємо форму введемо у поле прізвище цифру ( рис. 5.7). Проведемо тест форми контрагент до вводу телефона ( рис. 5.9)

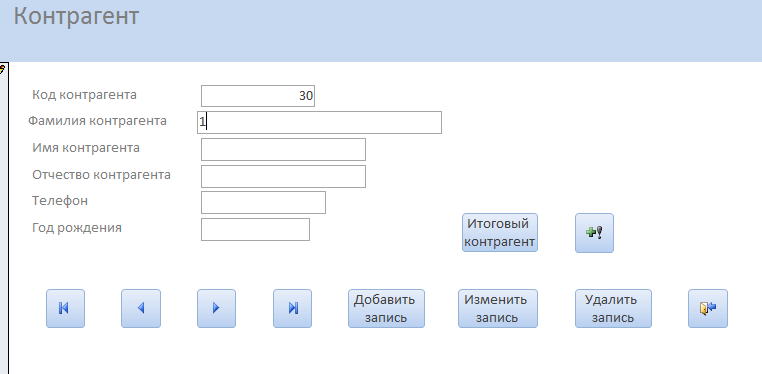


Рисунок 5.7 – форма контрагент

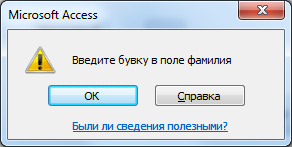


Рисунок 5.8 – тест некоректні дані

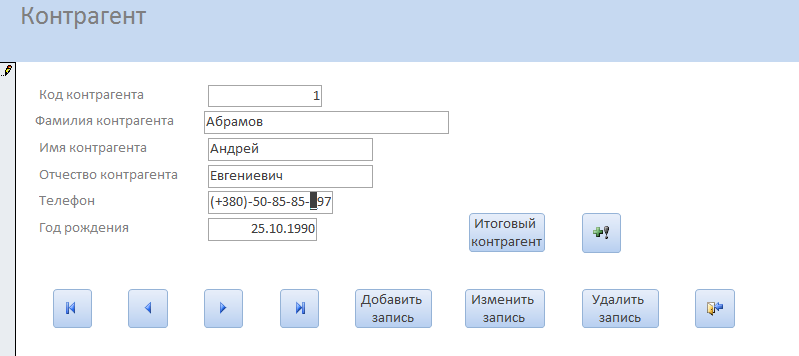


Рисунок 5.9 – форма контрагент

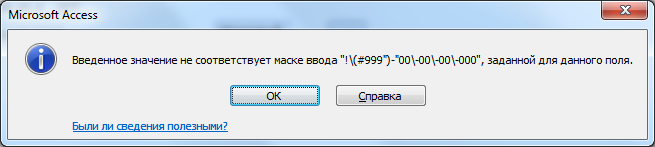


Рисунок 5.10 – помилка до вводу даних

## 5.3 Тестування форми договір

Проведемо тестування на формі договір при некоректному вводі серійного номеру (рис. 5.11). Відгук о помилці серійний номер на відповідає масці вводу (рис. 5.12)

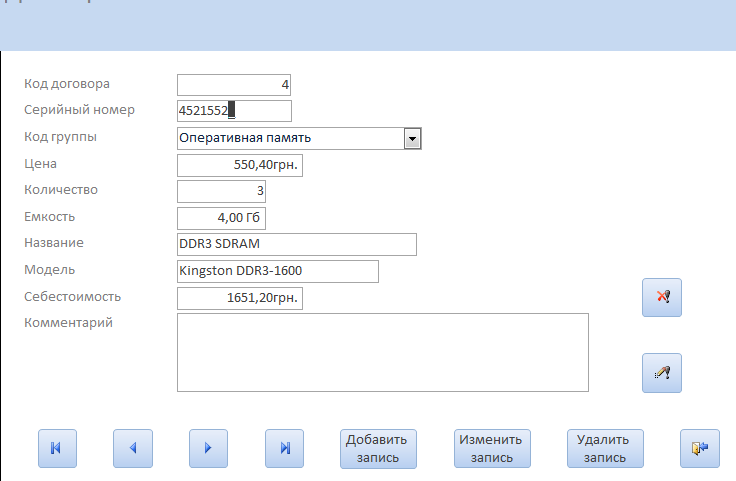


Рисунок 5.11 – форма договір

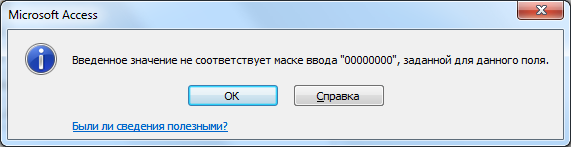


Рисунок 5.12 – відгук о помилці у формі договір

Проведемо тестування до вводу даних у поле логін форми користувач (рис. 5.13). Відгук о помилці до вводу логіна на відповідає умові вводу (рис. 5.14).

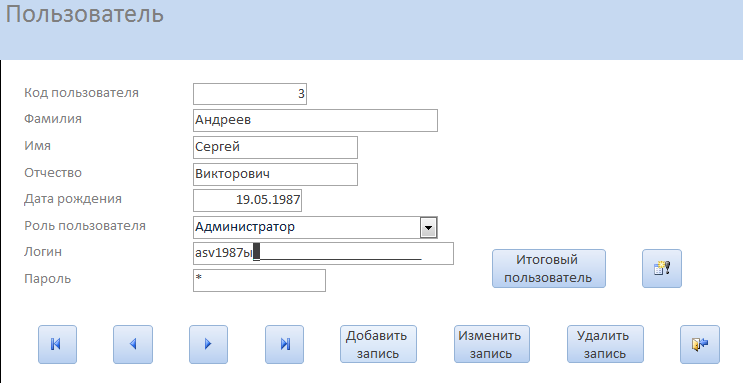


Рисунок 5.13 – форма користувач

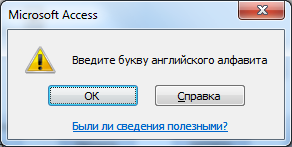


Рисунок 5.14 – відгук о помилці у формі користувач

# ВИСНОВКИ

В ході виконання дипломної роботи була спроектована та розроблена програмна система для автоматизації складського обліку комп’ютерної техніки та її комплектуючих.

У процесі виконання даної роботи був досліджений об'єкт автоматизації складського обліку, розглянуті основні проблеми автоматизації в даній галузі та проаналізовані існуючи інформаційні системи для складських підприємств.

Також були описані особливості предметної області з точки зору реалізації дій різних користувачів, проведений процес проектування програмного засобу за допомогою уніфікованої мови моделювання UML, описані варіанти використання програмної системи, апаратна складова, та принципи взаємодії дійових осіб, розроблена програмна система для автоматизації складського обліку комп’ютерної техніки та її комплектуючих, проведено тестування розробленої програми.

Впровадження розробленої програмної системи в складські підприємства підвищить ефективність робот складів, та використання комп’ютерних технологій прискорити та підвищити якість обслуговування користувачів**.**

# СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Програми автоматизації складського обліку[Електронний ресурс]

http://sklad-prog.ru/obzor/o\_ind.htm/

1. Толстенко Д.С. Егорова И.Е.[Документ]

Опис автоматизації складського обліку торгового підприємства

1. Уроки по MS Access[Електронний ресурс]

https://edugalaxy.intel.ru/index.php?act=attach&type=blogentry&id=3847

4. Cправка по MS Access [Електронний ресурс]

https://support.office.com

5. Урок по VBA [Електронний ресурс]

http://access.my-study.info/urok-online/etapy-izuchenia.php

6. Урок по масці вводу [Електронний ресурс]

https://support.office.com/ru-ru/article/%D0%A1%D0%BE%D0%B7%D0%B4%D0%B0%D0%BD%D0%B8%D0%B5-%D0%B8-%D1%80%D0%B0%D0%B1%D0%BE%D1%82%D0%B0-%D1%81-%D0%BC%D0%B0%D1%81%D0%BA%D0%B0%D0%BC%D0%B8-%D0%B2%D0%B2%D0%BE%D0%B4%D0%B0-%D0%B2-Access-e125997a-7791-49e5-8672-4a47832de8da

7. Складський облік [Електронний ресурс]

http://www.infopolis.com.ua/help.aspx?cut=3/7

8. Принципи реалізації обліку [Електронний ресурс]

http://www.infopolis.com.ua/help.aspx?cut=3

9. Поняття господарської операції. Первинний облік. [Електронний ресурс]

http://www.infopolis.com.ua/help.aspx?cut=3/4

10. Відповідність первинного та бухгалтерського обліків [Електронний ресурс]

http://www.infopolis.com.ua/help.aspx?cut=3/6

11. Інформаційний обмін. [Електронний ресурс]

http://www.infopolis.com.ua/help.aspx?cut=3/14

12. Азбука складського обліку. [Електронний ресурс]

http://www.buhgalteria.com.ua/Hit.html?id=1330

13. Складський облік: творчість прийому товарів. [Електронний ресурс]

http://class365.ru/skladskoi-uchet/oprihodovanie-tovarov

14. Складський облік: система онлайн обліку для бізнесу [Електронний ресурс]

http://bigbird.ru/blog/2013/06/skladskoy-uchet

15. Способи ведення складського обліку [Електронний ресурс]

http://www.xcomp.biz/4-3-1-sposoby-vedeniya-skladskogo-ucheta.html

16. Складський облік 1С [Електронний ресурс]

http://business-soft.net/ru/1c-inventory-control.html

17. «Супер-склад» проста програма складського обліку [Електронний ресурс]

http://sklad-prog.ru/

18. «Атонекс» Склад програма складського обліку. [Електронний ресурс]

http://sklad-prog.ru/obzor/prog\_001.htm

19. «Клас365» програма складського обліку. [Електронний ресурс]

http://sklad-prog.ru/obzor/o\_ind.htm/

20. 1С: Бухгалтерія 8 - програма складського обліку. [Електронний ресурс]

http://sklad-prog.ru/obzor/prog\_014.htm

# ДОДАТОК А

## Лістинг програми

Запит підсумковий контрагент

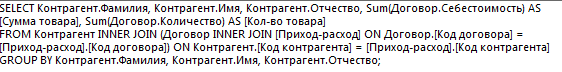


Рисунок А.1

Запит підсумковий користувач

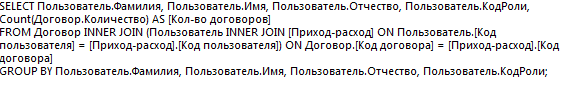


Рисунок А.2

Запит на вибірку групи комплектуючих (параметричний)

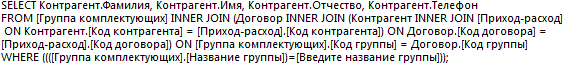


Рисунок А.3

Запит на вибірку тип операції (параметричний)

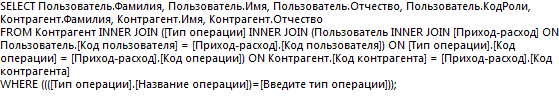


Рисунок А.4

Запит договори за період

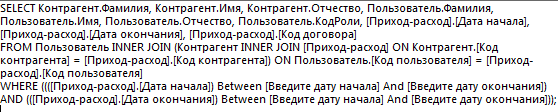


Рисунок А.5

Запит на добавлення контрагента



Рисунок А.6

Макрос на перевірку до входу у кабінету користувача

Option Compare Database

Private Sub Кнопка21\_Click()

Dim UserLevel As Integer

Dim TempPass As String

Dim ID As Integer

If IsNull(Me.txtLoginID) Then

MsgBox "Please enter LoginID", vbInformation, "LoginID Requered"

Me.txtLoginID.SetFocus

ElseIf IsNull(Me.txtPassword) Then

MsgBox "Please enter password", vbInformation, "Password Requered"

Me.txtPassword.SetFocus

Else

'process the job

If (IsNull(DLookup("Логин", "Пользователь", "Логин='" & Me.txtLoginID.Value & "'"))) Or \_

(IsNull(DLookup("Пароль", "Пользователь", "Пароль='" & Me.txtPassword.Value & "'"))) Then

MsgBox "Incorrect LoginID and Password"

Else

UserLevel = DLookup("КодРоли", "Пользователь", "Логин = '" & Me.txtLoginID.Value & "'")

UserLevel = DLookup("КодРоли", "Пользователь", "Пароль = '" & Me.txtPassword.Value & "'")

'TempPass = DLookup("Пароль", "Пользователь", "Логин = '" & Me.txtLoginID.Value & "'")

DoCmd.Close

If UserLevel = 1 Then

'MsgBox "LoginID and Password Correct"

DoCmd.OpenForm "Главное окно админ"

Else

DoCmd.OpenForm "Главное окно пользователь"

End If

End If

End If

End Sub

ДОДАТОК Б

Презентація для захисту

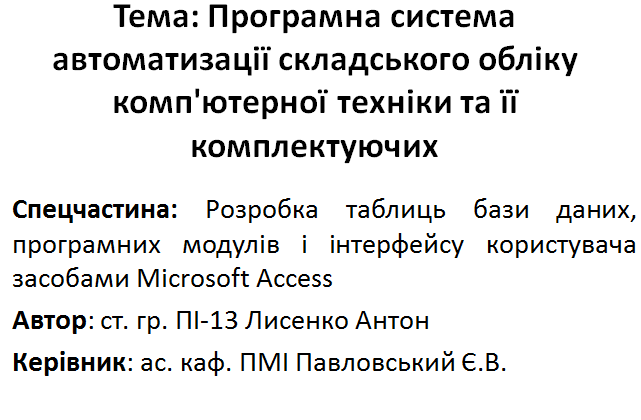


Рисунок Б.1

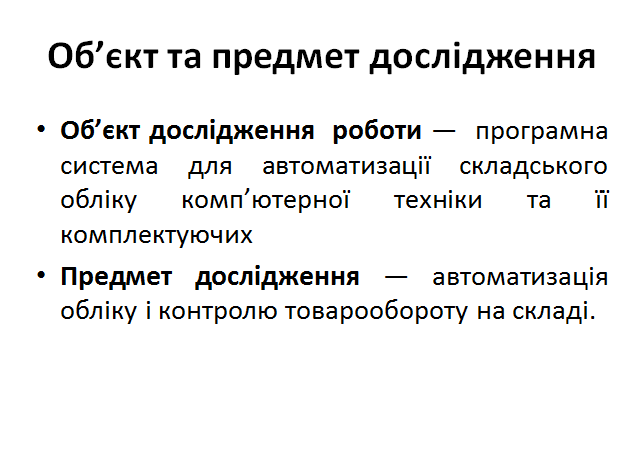


Рисунок Б.2

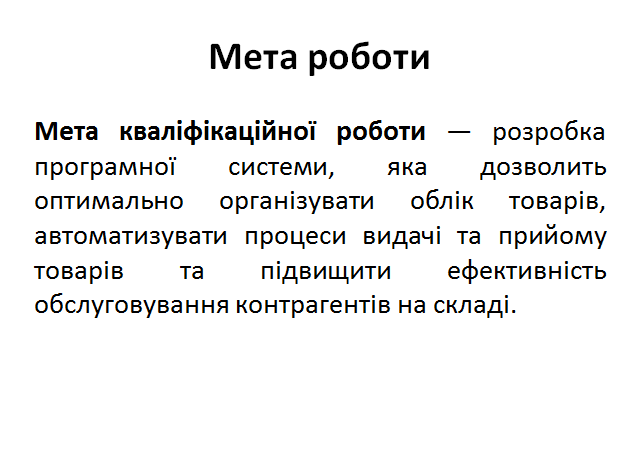


Рисунок Б.3

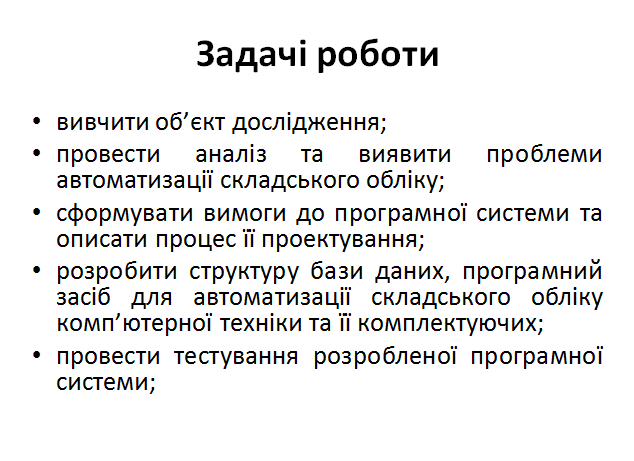


Рисунок Б.4

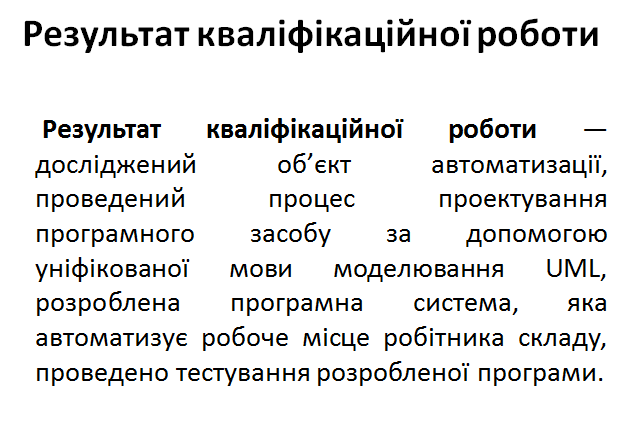


Рисунок Б.5

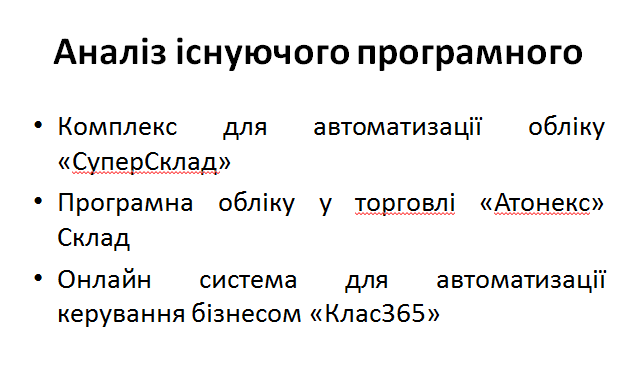


Рисунок Б.6

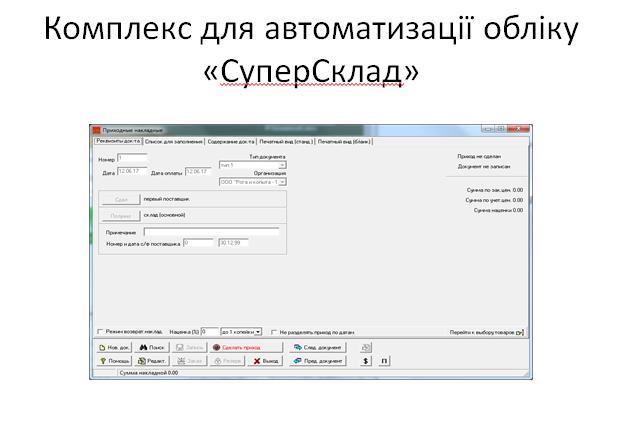


Рисунок Б.7

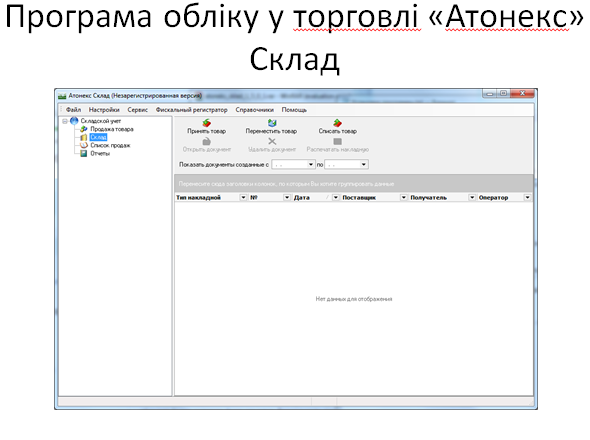


Рисунок Б.8

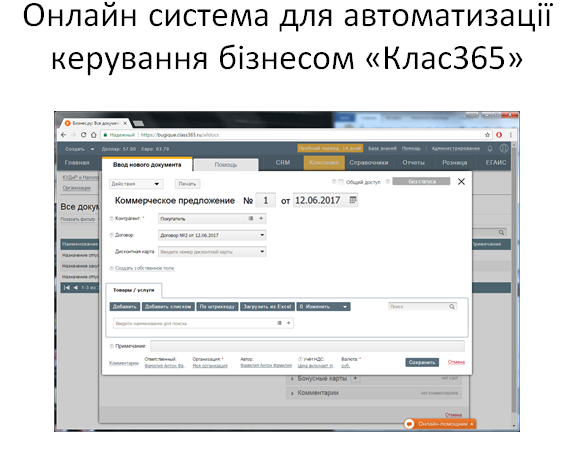


Рисунок Б.9



Рисунок Б.10



Рисунок Б.11

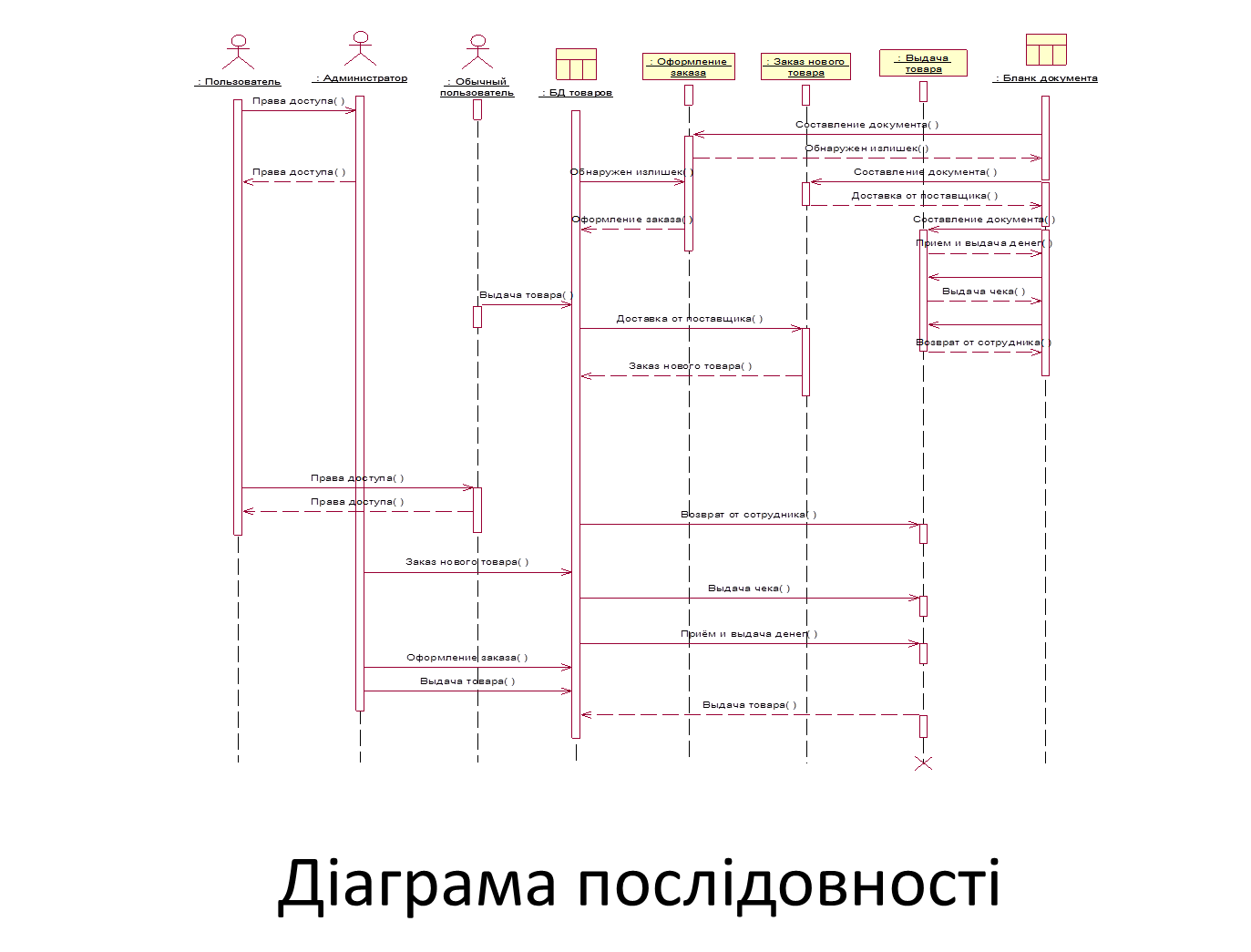


Рисунок Б.12



Рисунок Б.13



Рисунок Б.14



Рисунок Б.15

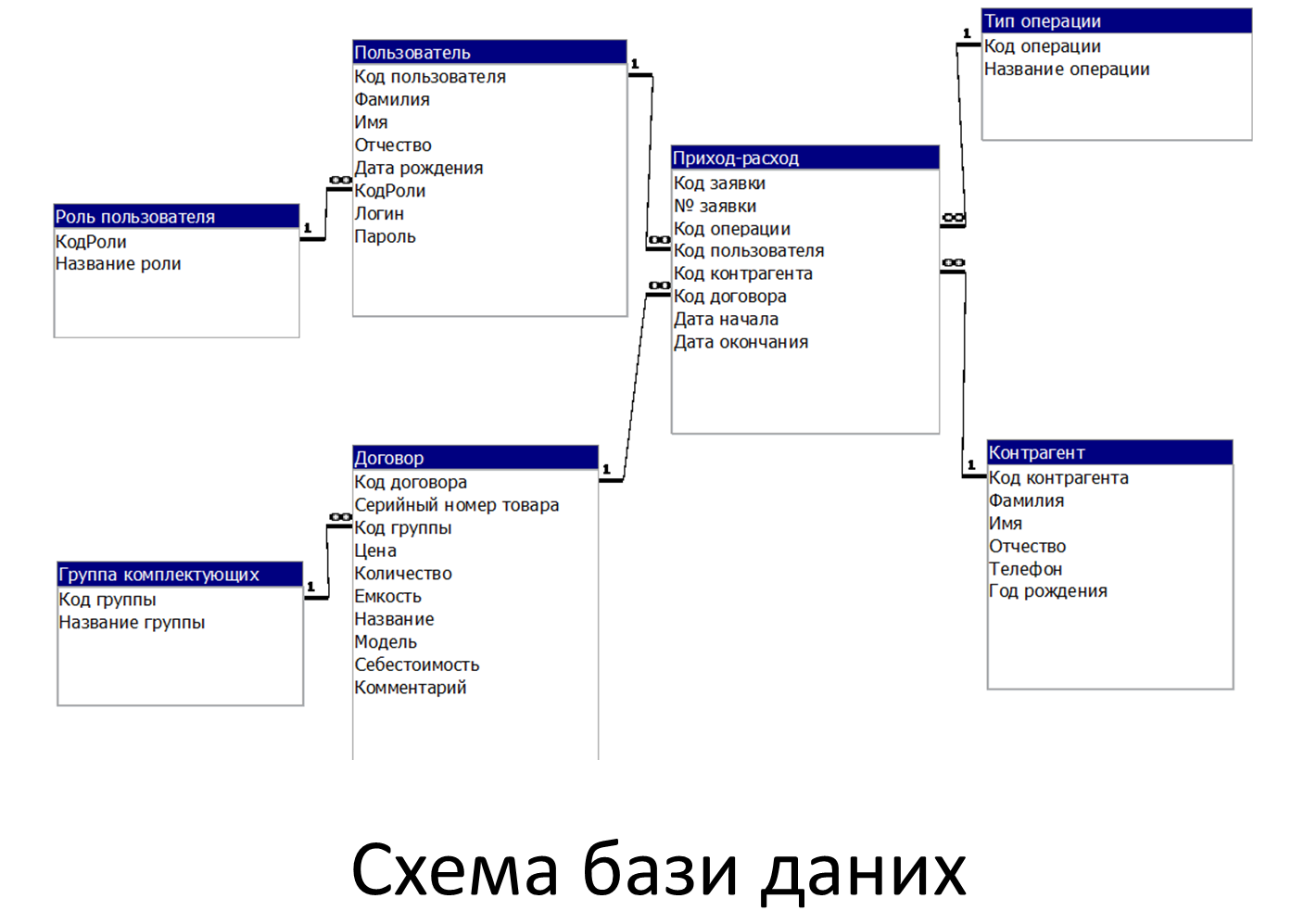


Рисунок Б.16

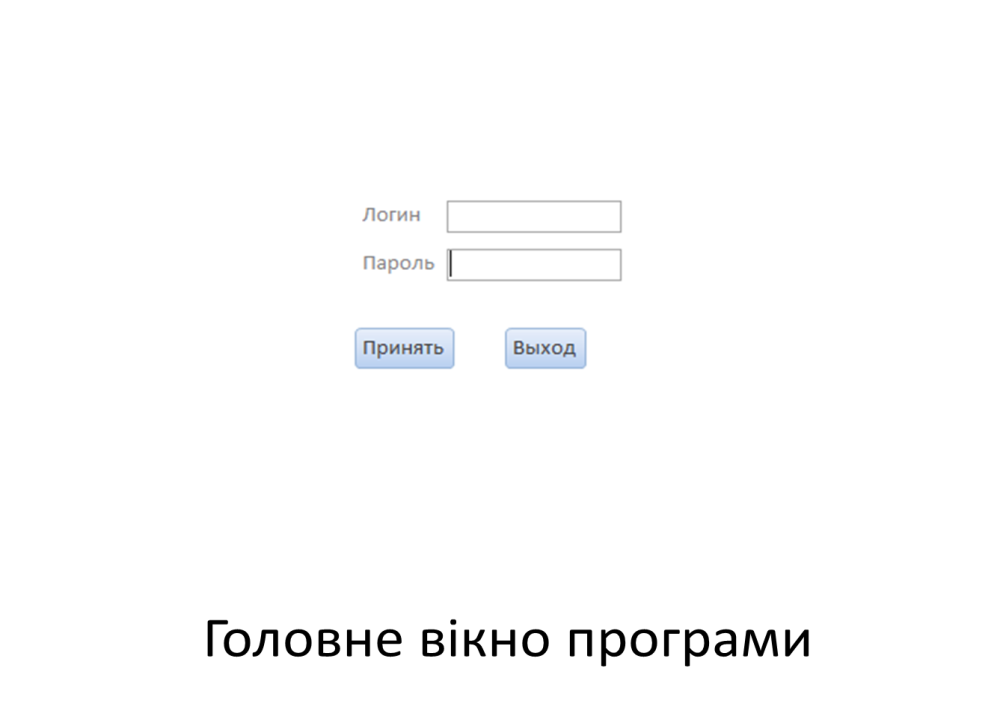


Рисунок Б.17

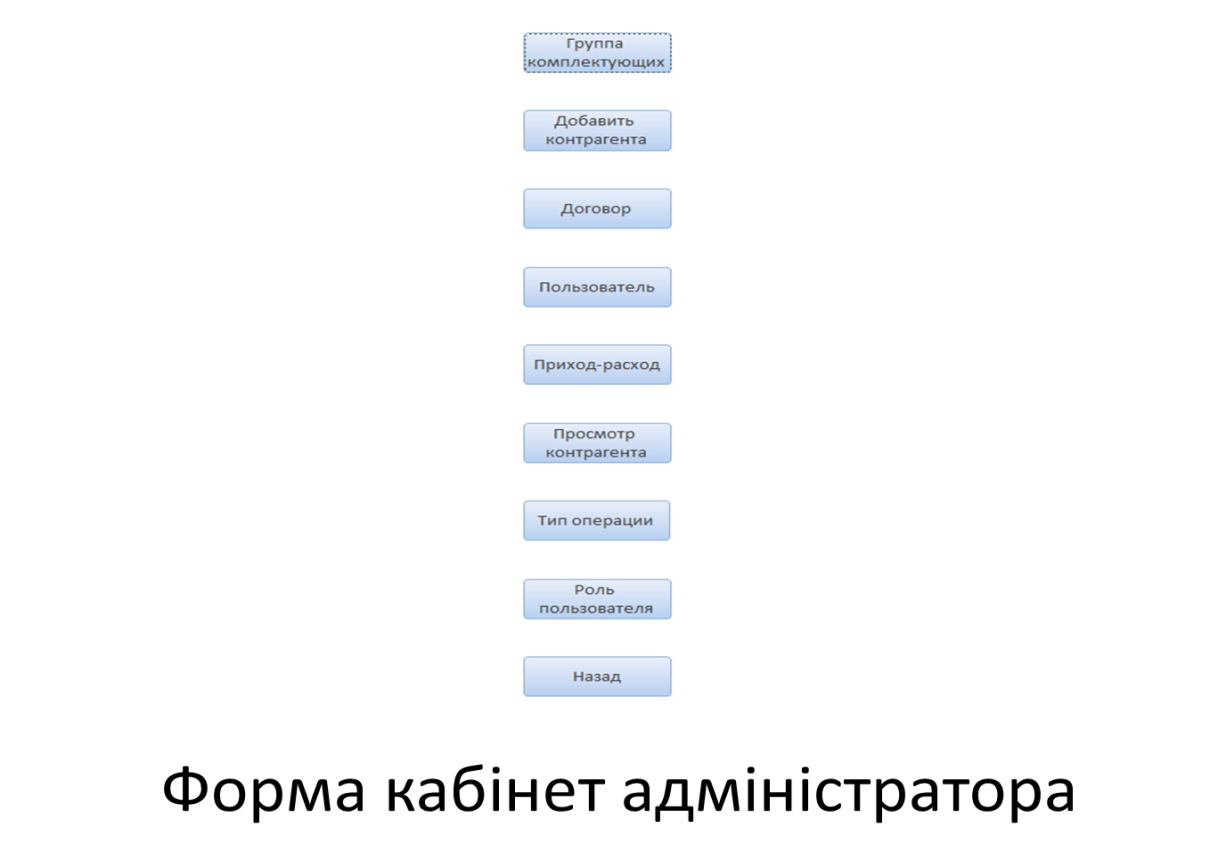


Рисунок Б.18

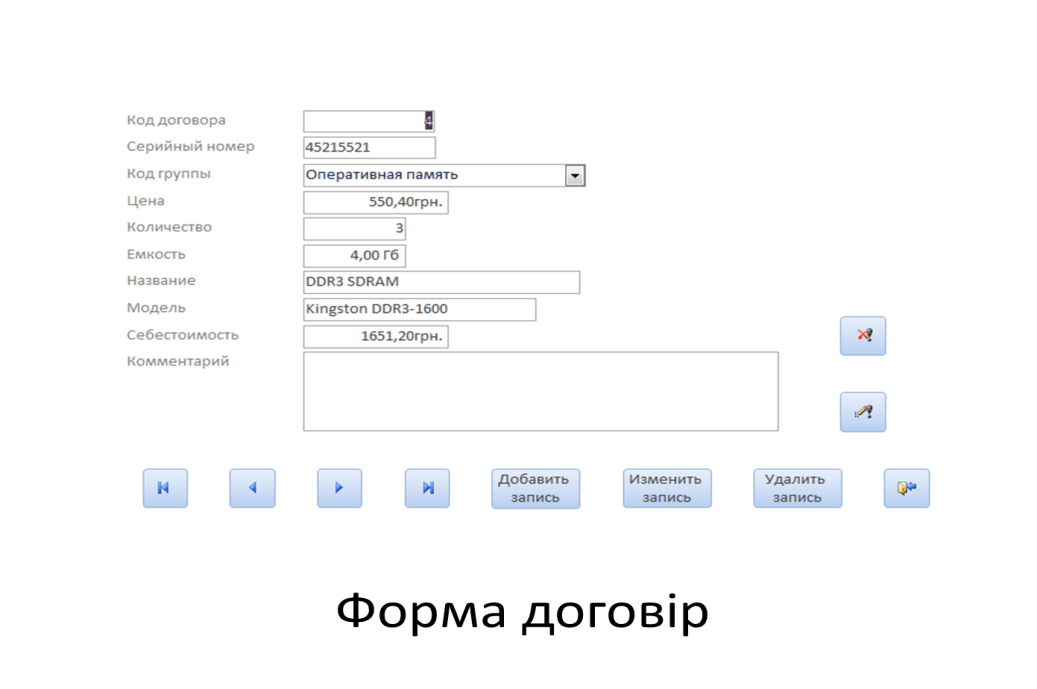


Рисунок Б.19

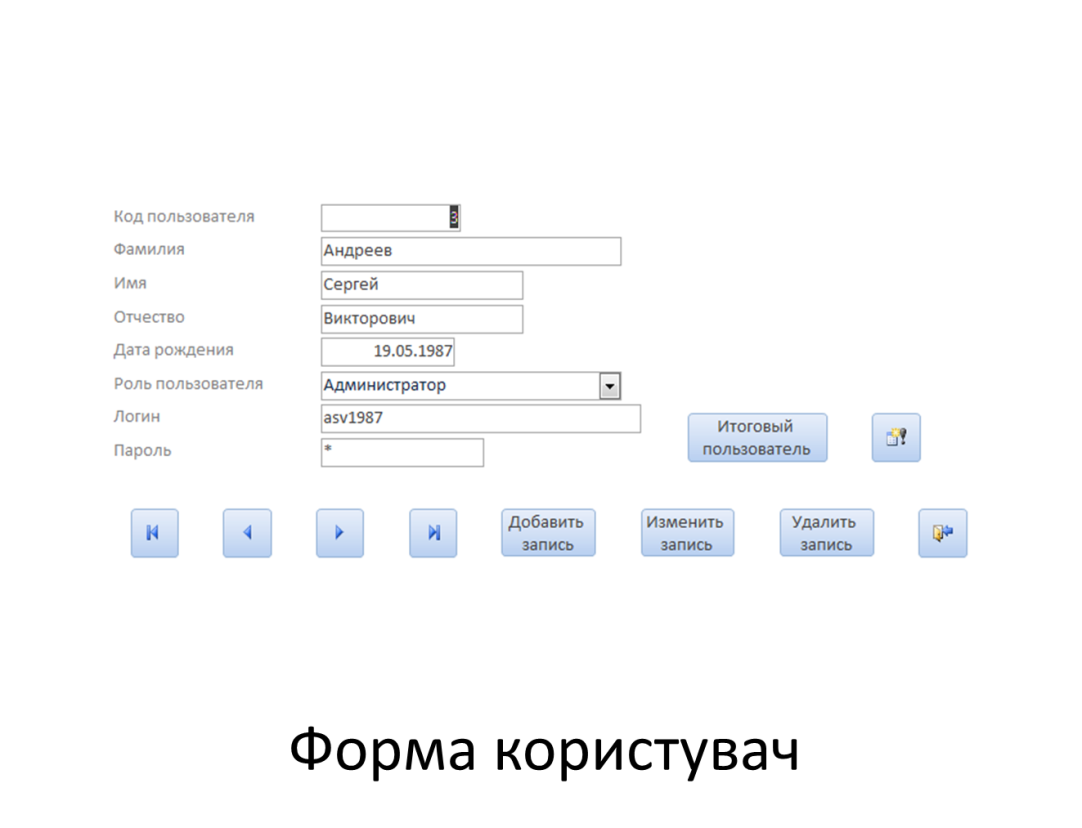


Рисунок Б.20

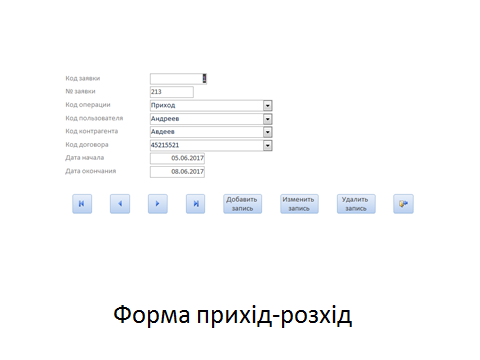


Рисунок Б.21

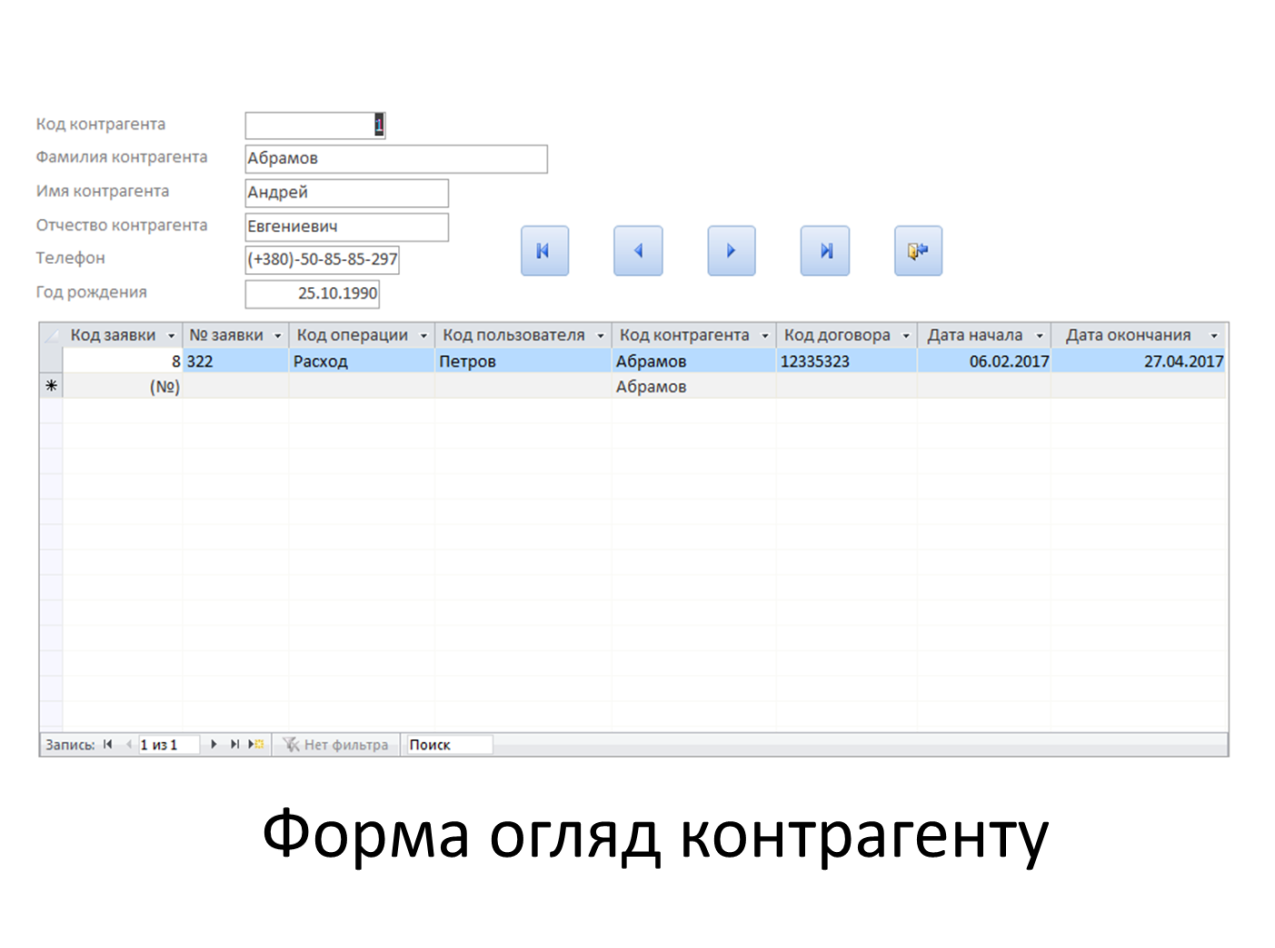


Рисунок Б.22

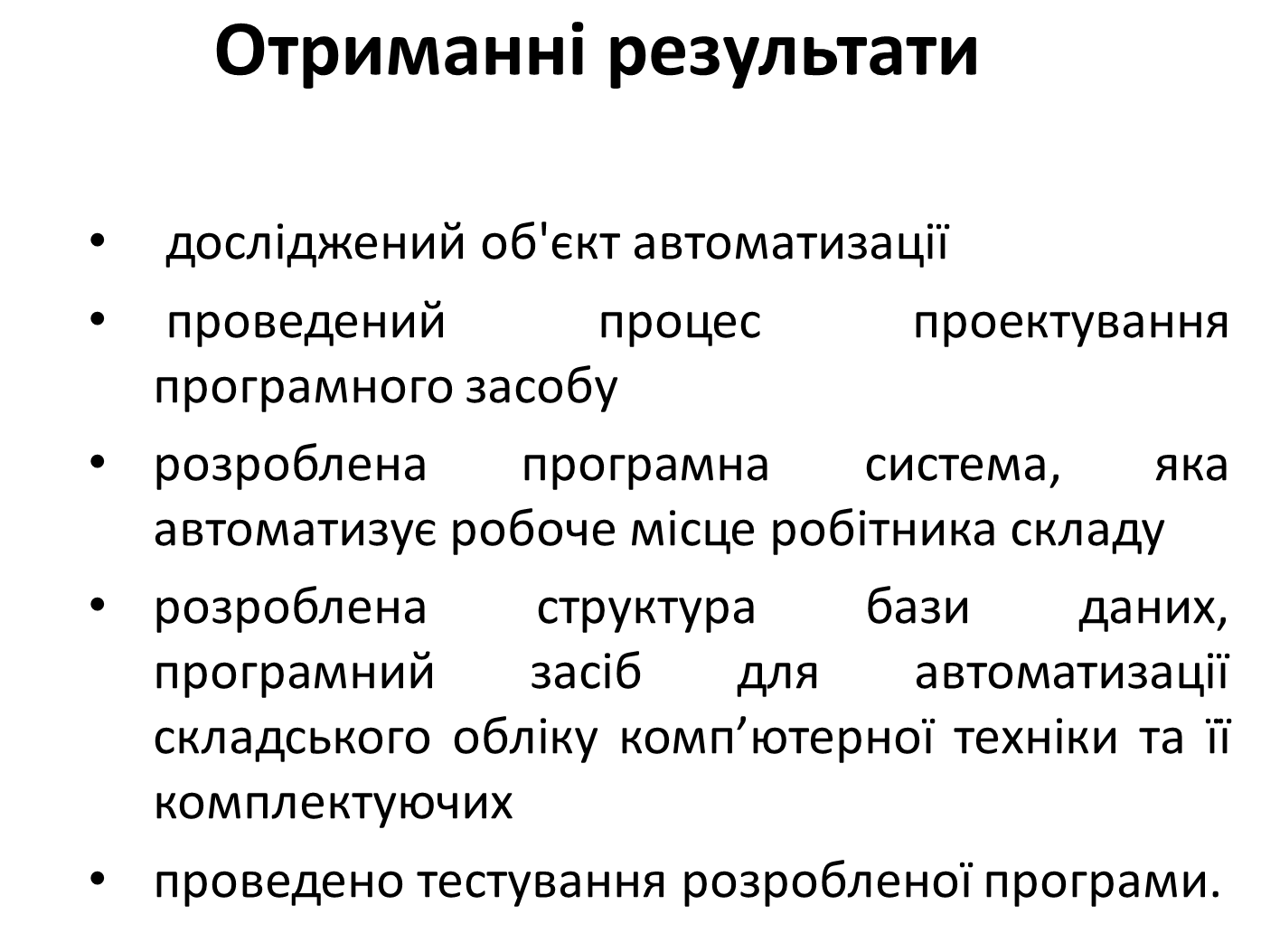


Рисунок Б.23



Рисунок Б.24

ДОДАТОК В

Екранні форми програми

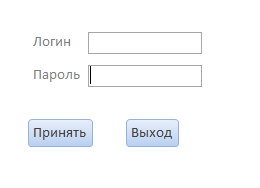


Рисунок В.1- початкова форма входу в програмну систему

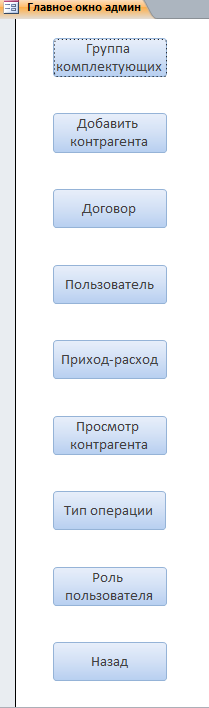


Рисунок В.2 - форма входу в програмну систему при ролі адміністратор.

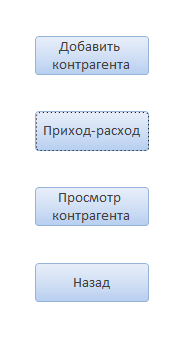


Рисунок В.3 - форма входу в програмну систему під роллю користувача.

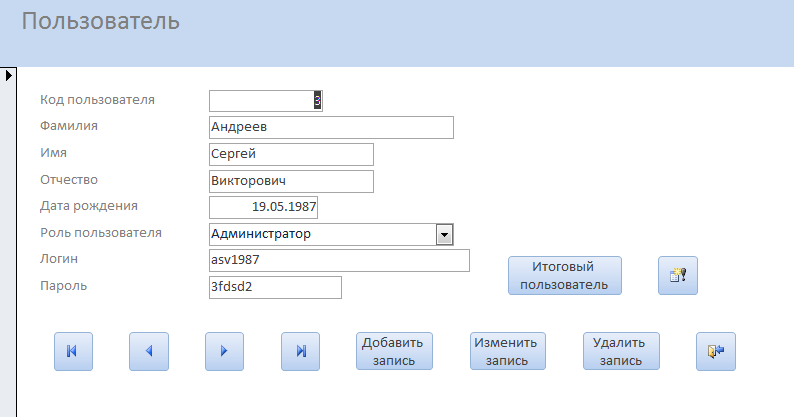


Рисунок В.4 – форма «Користувач»

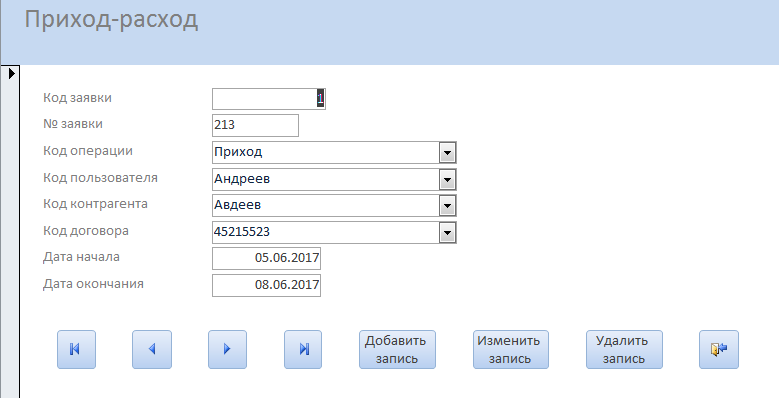


Рисунок В.5 – форма «Прихід-розхід»

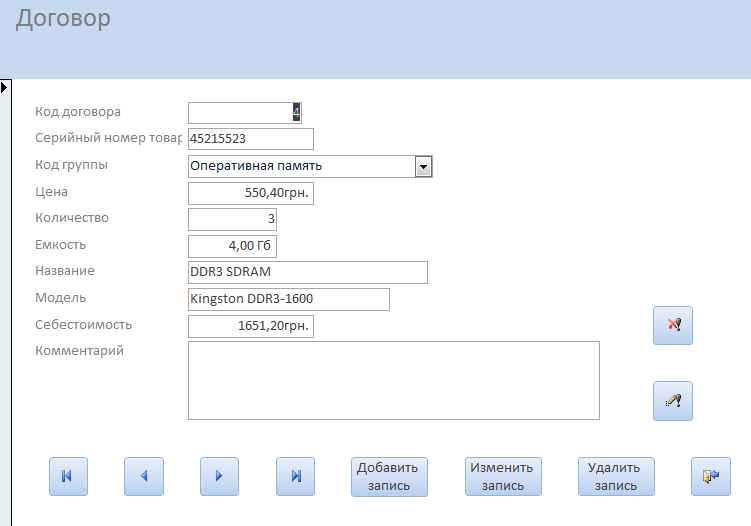


Рисунок В.6 – «Форма договір»

Додаток Г

Зауваження нормоконтролера

Таблиця Д.1 – Зауваження нормоконтролера

|  |  |  |
| --- | --- | --- |
| Сторінка | Позначення | Зміст зауваження |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |