

Центральноукраїнський національний технічний університет
Механіко-технологічний факультет
Кафедра кібербезпеки та програмного забезпечення

”Допущено до захисту”
Завідувач кафедри кібербезпеки
та програмного забезпечення
д.т.н., професор
_____ Олексій СМІРНОВ
“ ____ ” _____ 2022 р.

ВИПУСКНА КВАЛІФІКАЦІЙНА РОБОТА
за другим (магістерським) рівнем вищої освіти
на тему
“Дослідження та програмна реалізація системи довідково-інформаційного сервісу комп’ютерної мережі відділу підприємства”

Виконав здобувач вищої освіти
II курсу, групи КІ-21М-1,4
ОПП «Комп’ютерна інженерія»
спеціальності 123 «Комп’ютерна інженерія»
_____ Борзенко В.Ю.
« ____ » _____ 2022 р.

Керівник проекту
кандидат технічних наук
_____ Буравченко К.О.
« ____ » _____ 2022 р.
Рецензент _____

Центральноукраїнський національний технічний університет
Факультет Механіко-технологічний
Кафедра Кібербезпеки та програмного забезпечення
Рівень вищої освіти магістр
Галузь знань 12 "Інформаційні технології"
Спеціальність 123 "Комп'ютерна інженерія"
Освітньо-професійна (освітньо-наукова) програма "Комп'ютерна інженерія"

ЗАТВЕРДЖУЮ

Завідувач кафедри

д.т.н., проф.

Олексій СМІРНОВ

« 6 » вересня 2022 року

ЗАВДАННЯ НА ВИПУСКНУ КВАЛІФІКАЦІЙНУ РОБОТУ ЗА ДРУГИМ (МАГІСТЕРСЬКИМ) РІВНЕМ ВИЩОЇ ОСВІТИ ЗДОБУВАЧА ВИЩОЇ ОСВІТИ

Борзенку В`ячеславу Юрійовичу

(прізвище, ім'я, по батькові)

1. Тема роботи

Дослідження та програмна реалізація системи довідково-інформаційного сервісу комп'ютерної мережі відділу підприємства

2. Керівник роботи

Буравченко Костянтин Олегович, канд. техн. наук

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

затверджені наказом вищого навчального закладу № 19-13 від 17.08.2022 року

3. Строк подання студентом роботи до захисту 10.12.2022 р.

4. Мета та завдання випускної кваліфікаційної роботи: *Метою розробки є дослідження та програмна реалізація системи довідково-інформаційного сервісу комп'ютерної мережі відділу підприємства*

5. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити)

1. Призначення та область використання.

6. Наукова новизна.

2. Перегляд аналогічних існуючих систем.

7. Економічна ефективність розробленої програми.

3. Опис і обґрунтування проектних рішень.

8. Заходи з охорони праці та техніки безпеки.

4. Етапи програмування системи.

9. Висновки.

5. Впровадження системи в промислову експлуатацію

6. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)

Наукова новизна

1 аркуш

Структурна схема системи

1 аркуш

Функціональна схема системи

1 аркуш

Діаграма процесів

1 аркуш

Блок-схема алгоритму роботи додатку

2 аркуша

Показники економічної ефективності

1 аркуш

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Економічний	Савеленко Г.В.	05.10.2022	14.11.2022
Охорона праці	Оришака О.В.	06.10.2022	16.11.2022

7. Дата видачі завдання « 6 » вересня 2022 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти	Строк виконання етапів випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти	Примітка
1.	Аналіз існуючих систем	10.10.2022 р.	
2.	Постановка задачі, оформлення ТЗ	15.10.2022 р.	
3.	Розробка моделі компонента	20.10.2022 р.	
4.	Розробка структур даних	25.10.2022 р.	
5.	Розробка алгоритмів зв'язку та відображення	30.10.2022 р.	
6.	Програмування алгоритмів	10.11.2022 р.	
7.	Розрахунок економічної ефективності	13.11.2022 р.	
8.	Розрахунки з охорони праці та техніки безпеки	15.11.2022 р.	
9.	Оформлення ПЗ	17.11.2022 р.	
10.	Попередній захист роботи	10.12.2022 р.	

Дата видачі завдання
« 6 » вересня 2022 р.

Підпис керівника

Буравченко К.О.
(прізвище та ініціали)Завдання прийнято до виконання
« 6 » вересня 2022 р.

Підпис здобувача

Борзенко В.Ю.
(прізвище та ініціали)

АНОТАЦІЯ

Борзенко В.Ю. Дослідження та програмна реалізація системи довідково-інформаційного сервісу комп'ютерної мережі відділу підприємства. 123 Комп'ютерна інженерія. Центральноукраїнський національний технічний університет. Кропивницький. 2022.

В даній випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти розроблено програмне забезпечення, яке призначено для системи довідково-інформаційного сервісу комп'ютерної мережі відділу підприємства.

Метою розробки є дослідження та програмна реалізація системи довідково-інформаційного сервісу комп'ютерної мережі відділу підприємства.

Об'єктом дослідження є процес довідково-інформаційного сервісу комп'ютерної мережі відділу підприємства.

Предметом дослідження є методи довідково-інформаційного сервісу комп'ютерної мережі відділу підприємства.

Методи дослідження базуються на методах теорії телекомунікацій, методах математичної статистики, методах розробки програмного забезпечення.

Результат роботи – програмна реалізація системи довідково-інформаційного сервісу комп'ютерної мережі відділу підприємства.

В процесі роботи над програмною моделлю виконано аналіз існуючих апаратних та програмних засобів. В повній мірі описані всі компоненти розробленого програмного забезпечення.

Розроблено зручний інтерфейс користувача. Наведені інструкції по роботі з програмними засобами.

Програма може використовуватися на ПЕОМ архітектури IBM PC з ОС Windows 10/11.

Програму розроблено в середовищі RAD Studio Delphi 10.4.

Ключові слова: комп'ютерна інженерія, довідково-інформаційний сервіс, комп'ютерна мережа

ABSTRACT

Borzenko V.Yu. Research and software implementation of the reference and information service system of the computer network of the enterprise department. 123 Computer engineering. Central Ukrainian National Technical University. Kropyvnytskyi. 2022.

In this graduation thesis for the second (master's) level of higher education, software was developed, which is intended for the reference and information service system of the computer network of the enterprise department.

The purpose of the development is the research and software implementation of the reference and information service system of the computer network of the enterprise department.

The object of the study is the process of reference and information service of the computer network of the enterprise department.

The subject of the study is the methods of reference and information service of the computer network of the enterprise department.

Research methods are based on telecommunications theory methods, mathematical statistics methods, and software development methods.

The result of the work is the software implementation of the reference and information service system of the computer network of the enterprise department.

In the process of working on the software model, an analysis of existing hardware and software was performed. All components of the developed software are fully described.

A convenient user interface has been developed. Instructions for working with software tools are provided.

The program can be used on PCs of IBM PC architecture with Windows 10/11 OS.

The program was developed in the RAD Studio Delphi 10.4 environment.

Keywords: computer engineering, reference and information service, computer network

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ І ТЕРМІНІВ	3
ВСТУП.....	4
1 ПРИЗНАЧЕННЯ ТА ОБЛАСТЬ ВИКОРИСТАННЯ	7
1.1 Призначення системи.....	7
1.2 Область застосування.....	9
2 ПЕРЕГЛЯД АНАЛОГІЧНИХ ІСНУЮЧИХ СИСТЕМ	11
2.1 Огляд існуючих систем, технологій, архітектур та програмних рішень за профілем теми випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти.....	11
2.2 Обґрунтування вибору засобів для побудови системи та мови програмування.....	19
2.3 Розгорнута постановка завдання	25
3 ОПИС І ОБҐРУНТУВАННЯ ПРОЕКТНИХ РІШЕНЬ	27
3.1 Опис функціонування системи	27
3.2 Розробка структурної схеми.....	33
3.3 Розробка функціональної схеми	37
3.4 Розробка діаграми процесів.....	39
4 РЕАЛІЗАЦІЯ РОБОТИ. РОЗРАХУНКИ І ЕКСПЕРИМЕНТАЛЬНІ ДАНІ, ЩО ПІДТВЕРДЖУЮТЬ ВІРНІСТЬ ПРОЕКТНИХ ТА ПРОГРАМНИХ РІШЕНЬ.....	41
4.1 Розробка блок-схем та опис алгоритмів функціонування системи.....	41
4.2 Захист розробленого програмного забезпечення.....	52
5 ВПРОВАДЖЕННЯ СИСТЕМИ В ПРОМИСЛОВУ ЕКСПЛУАТАЦІЮ	53
6 НАУКОВА НОВИЗНА	56

					ВКРМ-123.22.0003.00.00.ПЗ			
Вим	Арк.	№ докум.	Підп.	Дата	<i>Дослідження та програмна реалізація системи довідково-інформаційного сервісу комп'ютерної мережі відділу підприємства</i>	Лім.	Аркуш	Аркушів
<i>Розроб.</i>	<i>Борзенко В.Ю.</i>					М	1	98
<i>Перев.</i>	<i>Буравченко К.О.</i>							
Н.контр.	<i>Гермак В.С.</i>					<i>ЦНТУ КІ-21М-1,4</i>		
Затв.	<i>Смірнов О.А.</i>							

7 ЕКОНОМІЧНА ЕФЕКТИВНІСТЬ РОЗРОБЛЕНОЇ ПРОГРАМИ.....	57
7.1 Техніко економічне обґрунтування теми випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти.....	57
7.2 Розрахунок трудомісткості розробки програмної продукції.....	59
7.3 Визначення чисельності виконавців і планового фонду зарплати.....	61
7.4 Розрахунок капітальних вкладень та амортизаційних відрахувань у розробника.....	66
7.5 Визначення собівартості розробки та ціни програмної продукції.....	70
7.6 Визначення об'єму капітальних вкладень та експлуатаційних витрат у споживача програмної продукції.....	73
7.7 Визначення експлуатаційних витрат.....	73
7.8 Визначення економічної ефективності програмної продукції.....	75
7.9 Висновок.....	77
8 ЗАХОДИ З ОХОРОНИ ПРАЦІ ТА ТЕХНІКИ БЕЗПЕКИ	78
8.1 Вступ.....	78
8.2 Аналіз умов праці на робочому місці ІТ-фахівця.....	79
8.3 Пропозиції щодо підвищення працездатності ІТ-фахівця.....	81
8.4 Пожежна безпека.....	83
8.5 Розрахункова частина	84
8.6 Висновки до розділу.....	86
9 ОСНОВНІ ВИСНОВКИ.....	87
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	89

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ І ТЕРМІНІВ

АДІС	–	автоматизований довідково-інформаційний сервіс
БД	–	база даних
ВП	–	відеопам'ять
ОЗП	–	оперативний запам'ятовувальний пристрій
ОП	–	оперативна пам'ять
ПЕОМ	–	персональна електронно-обчислювальна машина
ПЗ	–	програмне забезпечення
ПК	–	персональний комп'ютер
ПП	–	програмний продукт
СУБД	–	система управління базами даних

					ВКРМ-123.22.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		3

ВСТУП

Актуальність теми. Для виробництва продукції підприємства, незалежно від форм власності, для свого нормального й стабільного функціонування мають потребу в сировині, комплектуючих, запчастинах і т.п. Іншими словами для організації своєї діяльності підприємства зіштовхуються із проблемою своєчасної поставки продукції. Для цих цілей підприємства державної форми власності використовують постачальницькі організації (у цей час даними організаціями користуються спеціалізовані державні підприємства), підприємства ж приватних форм власності проводять відповідні заходи, спрямовані на своєчасне й безперервне забезпечення необхідною сировиною або матеріалами. Але в цей час практично всі підприємства будь-якої форми власності самостійно займаються пошуком підприємств-постачальників, а також у міру необхідності організацією поставок. Як правило цими питаннями займаються відділи логістики підприємств.

Одним з показників підприємства, що характеризують роботу, є товарообіг, що являє собою планово організаційний процес обігу засобів виробництва [2], від якого багато в чому залежать і інші економічні показники. У загальний обсяг товарообігу включають всі товари, реалізовані підприємством, тобто отримані від підприємств постачальників продукції. Також товарообіг показує наскільки швидко підприємство використовує отриману продукцію, тобто якими темпами воно здійснює свою діяльність, чим більше на підприємство здійснюється поставок, тим більш стабільно працює дане підприємство.

При здійсненні поставок на підприємство виробляється обробка й зберігання великої кількості інформації, пов'язаної з поставками, що у себе включає:

					ВКРМ-123.22.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		4

– своєчасне й правильне оформлення документів і контроль за кожною операцією надходження товарів від постачальників, з переробки й інших джерел, виявлення розбіжності фактичної наявності й кількості, зазначеного в супровідних документах;

– контроль за своєчасним, повним і правильним оприбуткуванням товарів, що надійшли;

– своєчасне й правильне оформлення документації й контроль за кожною операцією відпустки, відвантаження або реалізації товару;

– контроль за дотриманням нормативів запасу товарів.

У зв'язку із цим для надійного функціонування системи поставок необхідно вести їх систематичний і безперервний облік, що й буде виконувати розроблювальне ПЗ відділу логістики підприємства.

У зв'язку з тим, що сучасні відділи логістики, як правило це великі структури, мають внутрішню комп'ютерну мережу, що має вихід в Інтернет, то розробка програмного забезпечення цієї комп'ютерної мережі має велике значення.

Розроблювальний програмний продукт буде відрізнятися від аналогічного програмного забезпечення можливістю застосування на сучасній електронно-обчислювальній техніці [1], зручним інтерфейсом, низькою вартістю, можливістю його використання на будь-якому підприємстві.

Мета й завдання дослідження. Метою роботи є дослідження та програмна реалізація системи довідково-інформаційного сервісу комп'ютерної мережі відділу підприємства.

Для досягнення поставленої мети визначена програма дослідження, що складається з наступних завдань:

– Огляд існуючих систем довідково-інформаційного сервісу комп'ютерної мережі відділу підприємства.

– Дослідження системи довідково-інформаційного сервісу комп'ютерної мережі відділу підприємства.

					ВКРМ-123.22.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		5

1 ПРИЗНАЧЕННЯ ТА ОБЛАСТЬ ВИКОРИСТАННЯ

1.1 Призначення системи

Автоматизований довідково-інформаційний сервіс комп'ютерної мережі відділу логістики (АДІС) “Облік поставок” буде використовуватися на підприємствах різних форм власності й забезпечувати контроль і облік поставок вироблених на підприємство. Також при використанні даного ПЗ буде бути можливість складання звітності про поставки на підприємство, виявлення заборгованості по оплаті поставок. Розроблювальне ПЗ може бути використане як керівником підприємства, для здійснення контролю вироблених поставок на підприємство, так і начальниками цехів, для ведення обліку поставок.

Метою магістерського проектування є розробка й створення програмного продукту комп'ютерної мережі відділу логістики “Облік поставок”. Дане програмне забезпечення призначене для контролю, обліку, автоматизації й систематизації інформації про поставки різного виду продукції на підприємство будь-якої форми власності, що займається будь-яким видом виробництва або діяльності.

Розроблювальний програмний продукт комп'ютерної мережі відділу логістики повинен забезпечувати створення інформаційної бази про здійснені поставки на підприємство, а також здійснювати створення наступних документів:

- бланк договору підприємства замовника з фірмою-постачальником (із вказівкою найменування і юридичних адрес сторін, що беруть участь у договорі, асортиментів продукції для поставок, її кількості, можливої вартості, умови й терміни дії договору);
- заявку на поставку необхідної продукції (вказується кількість, найменування, номенклатура, строки поставки, сума поставки);
- замовлення на поставку.

Комерційна версія програмного продукту дозволить робити:

- більш повний контроль і організацію обліку про поставки на

					ВКРМ-123.22.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		7

підприємство;

- автоматизувати процес оформлення поставок на підприємство;
- зменшити часові витрати на оформлення документів, пов'язаних з

поставками;

- обчислювати заборгованість по оплаті здійснених поставок на зазначений період;

- забезпечити користувача системою допомоги як по поняттях предметної області, так і по користуванню програмним продуктом.

Розроблювальний автоматизована система комп'ютерної мережі відділу логістики повинна буде реалізувати наступні функції:

- забезпечення введення даних про поставки на підприємство;
- аналіз введеної інформації;
- підрахунок заборгованості підприємства за здійснені поставки;
- визначати оптимальний рахунок-фактуру з погляду “кількість-ціна”;
- робить печать документації, пов'язаної з організацією поставок (бланк договору, замовлення, заявки).

Розроблювальна АДС комп'ютерної мережі відділу логістики повинна забезпечувати автоматизований контроль, а так само облік поставок на підприємство (цех цього підприємства), для цього створювана система повинна:

- забезпечувати уведення, пов'язаних з поставками на підприємство й обробку цих даних;

- створювати звітні документи й документи для організації грузопоставок;

- мати систему допомоги по програмі;

- при введенні даних про найменування товарів повинен використовуватися довідник “Номенклатура товарів”;

- створювані документи повинні відповідати галузевим стандартам, прийнятим на підприємстві.

У результаті виконаної роботи передбачається досягти наступних ефектів:

- зменшення часу необхідного для обліку поставок зроблених на підприємство;

					ВКРМ-123.22.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		8

- автоматизація контролю поставок;
- можливість тривалого зберігання інформації про поставки на підприємство великого строку давнини, для можливості більше повного розрахунку ефективності діяльності підприємства;
- постійна популярність про строки оплати здійснених поставок.

1.2 Область застосування

Будь-яке підприємство, здійснюючи свою діяльність, для одержання продукції від постачальників повинне укласти з останніми договір на поставку продукції. Звичайно на одноіменну продукцію підприємство-замовник містить кілька договорів з підприємствами-постачальниками. Потім замовник, у міру потреби в певній продукції, висилає постачальникові заявку на поставку продукції й одержує від останній рахунок-фактуру, у якому зазначене найменування продукції і її відпускна ціна. На підставі цих рахунків підприємство-замовник визначає оптимальну заявку й висилає постачальникові замовлення на поставку продукції. Після одержання замовленої продукції замовник відправляє рахунок у бухгалтерію, що оплачує його в банку в плинні строку, передбаченого договором. Тому для документального забезпечення процесу поставок на підприємство програма повинна створювати (роздруковувати) наступні необхідні документи:

- бланк договору підприємства-замовника з фірмою-постачальником (із вказівкою найменування і юридичних адрес сторін, асортиментів продукції для поставок, її кількості й можливої вартості, а так само умови й терміни дії договору);
- замовлення на поставку необхідної продукції (вказується кількість, найменування, номенклатура, строки поставки).

Також створювана автоматизована система за наявним даними про постачальників і знову отриманим даним повинна визначати оптимальний рахунок-фактуру з погляду кількість-ціна. Будь-яку поставку підприємство-замовник зобов'язаний оплатити у встановленим договором термін, тому АС

					ВКРМ-123.22.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		9

повинна здійснювати підрахунок суми боргу (грошей до виплати) на поточну дату.

На рисунку 1.1 представлена функціональна схема здійснення поставок на підприємство.

У такий спосіб для розробки АДІС необхідно виконати наступні задачі:

- реалізація керування доступом до АДІС;
- створити СУБД для роботи АДІС;
- розробити довідкову інформацію з АДІС;
- виконати аналіз і обліку поставок.

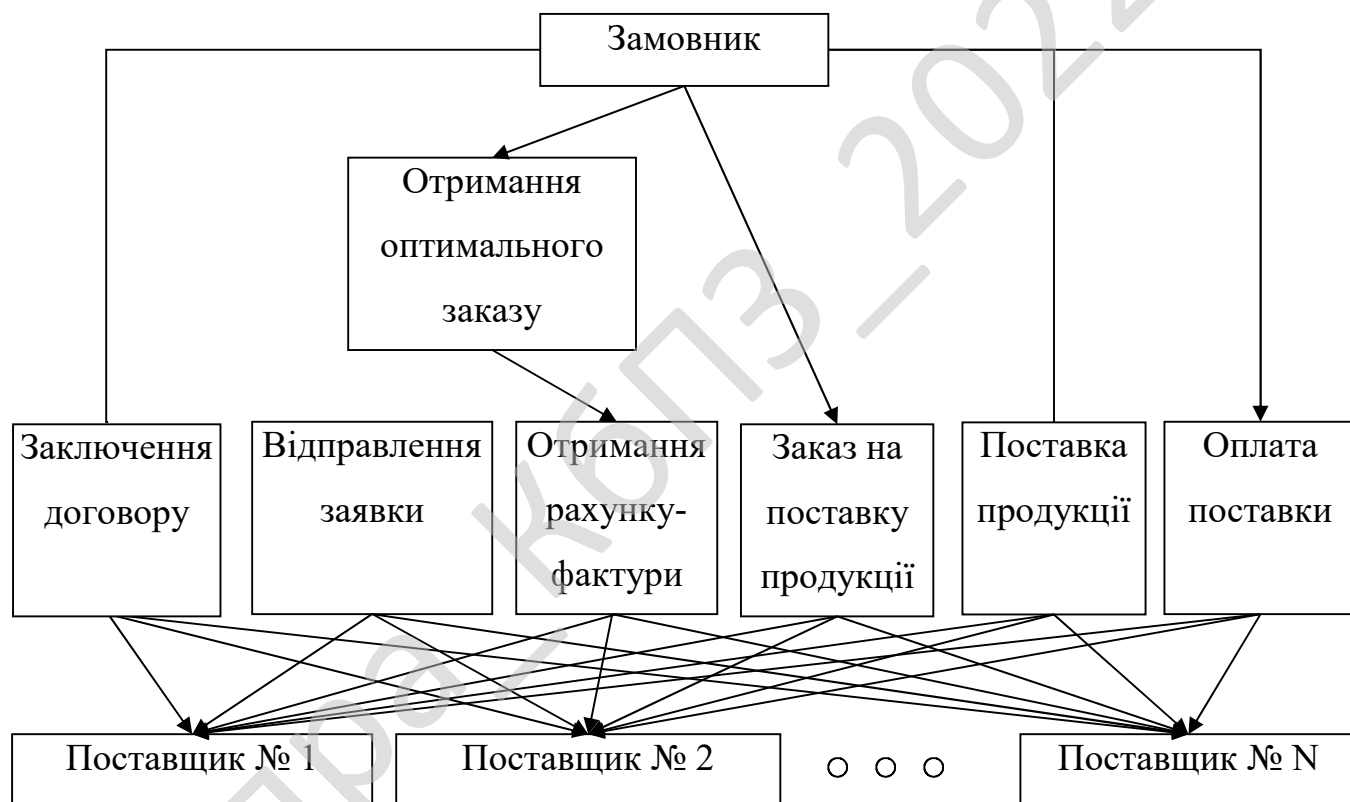


Рисунок 1.1 – Функціональна схема здійснення поставок на підприємство

Таким чином, виходячи з вищеперерахованого, дослідження та програмна реалізація системи довідково-інформаційного сервісу комп'ютерної мережі відділу підприємства, є актуальною задачею, яка потребує вирішення у даній випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти.

способами розміщення сегментів установлюється порядок доступу до них. Установлений порядок доступу до сегментів спричиняється процедурність язику запитів і жадає від користувача знання шляхів доступу до даних, що проходять по галузях дерева ієрархічного запису. Що є одним з недоліків даної моделі. Як інші недоліки можна відзначити наступні:

- складність реалізації “багато до багатьох”, що вимагає надмірності даних на фізичному рівні, що приведе до небажаного й не виправданому збільшенню БД;

- вимога підвищеної коректності до операції видалення, оскільки видалення вихідного сегмента спричиняє видалення породжених;

- доступ до будь-якого породженого сегмента можливий тільки через вихідний, що збільшує час відповіді а запит до БД.

У зв'язку з тим, що ієрархічна модель має більшу кількість недоліків вона не буде застосовуватися для моделювання розроблювальної АДІС.

Мережна модель даних

Мережа [5] – більш загальна структура в порівнянні з ієрархією. Вузлами мережі є окремі екземпляри запису. Вузли запису є одиницею доступу до БД. Оскільки окремий вузол може мати трохи безпосередньо старших вузлів, так само, як і трохи безпосередньо підлеглих, те дана структура забезпечує пряме подання відносини “багато до багатьох”. Для зв'язку між записами-вузлами існує сполучний запис, всі екземпляри якого містяться в ланцюжок для зв'язку двох екземплярів.

Основною конструкцією мережної моделі даних є набір. Для кожного типу набору, обумовленого в схемі, повинен бути зазначений певний тип запису власника набору, а так само довільне число типів запису членів набору. Кожний екземпляр набору складається з одного екземпляра-власника й одного або більше екземплярів записів-членів.

Кожний екземпляр запису-набору представляє ієрархічні зв'язки між екземпляром запису-власника й відповідних екземплярів записів-членів. Це є

					ВКРМ-123.22.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		12

наслідком того обмеження, що жоден екземпляр запису-члена з набору не може належати більш, ніж одному екземпляру набору. Спосіб, яким кожний екземпляр запису власника зв'язується з відповідними екземплярами записів-членів, визначається в схемі мережі. Одним зі способів організації таких зв'язків є встановлення ланцюжка покажчиків, що виходять із екземпляра запису-власника, що проходять через всі екземпляри записів-членів і, що вертаються назад до екземпляра запису-власника, що забезпечує високу швидкість обробки запитів.

Головний недолік мережної моделі полягає в складності структури пам'яті. Користувач повинен знати, які ланцюжки існують і які відсутні. У результаті язик запитів процедурний і вимагає програмістських навичок.

Реляційна модель даних

Реляційна модель – множинне відношення яке являє собою підмножину декартового добутку списку доменів [27,20]. Домен – це безліч значень, з якого витягають значення для даного атрибута. Іншими словами в основі реляційної моделі лежать прості таблиці, які задовольняють певним обмеженням, а тому можуть розглядатися як математичні відносини. Рядки таких таблиць називаються кортежами, імена стовпців – атрибутами. Слід зазначити, що всі кортежі різні, а порядок стовпців довільний, чим спрощується процес обробки кортежів. У відношенні (таблиці) виділяється кілька атрибутів, однозначно ідентифікуючі кортежі й називаних ключами.

Особливість реляційної моделі полягає в тім, що у відмінності від мережної й ієрархічної моделей реальні об'єкти й взаємозв'язки між ними представляються в базі даних одноманітно у вигляді нормалізованих відносин [24].

Основний недолік реляційної моделі даних зв'язується з низькою продуктивністю реляційної СУБД. Але розробка сучасних СУБД таких як, ORACLE, InterBase, Access і ін. дозволило перебороти й цей недолік.

Перевага реляційної моделі можна розділити на дві групи:

- 1) перевага для користувачів:

					ВКРМ-123.22.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		13

- реляційна БД являє собою набір таблиць із якими користувач звик працювати;
 - не потрібно пам'ятати шляхи доступу до даних і будувати алгоритми й процедури обробки свого запиту;
 - реляційні мови легкі для вивчення й освоєння, у той час як мови спілкування з ієрархічною й мережною моделями призначені для програмістів і мало придатні для користувачів;
- 2) перевага обробки даних реляційної БД:
- зв'язність – реляційне подання дає ясну картину взаємозв'язків атрибутів з різних відносин;
 - точність – спрямовані зв'язки в реляційній БД відсутні. Відносини по своїй природі мають більше точний зміст і піддаються маніпулюванню з використанням таких засобів, як алгебра й вираження відносин [5], що забезпечують наочність і гнучкість моделі даних;
 - гнучкість – операції проєкції й об'єднання [17] дозволяють розрізати й склеювати відносини, так що програміст може одержувати різноманітні файли в потрібній формі;
 - таємність – контроль таємності спрощується. Для кожного відношення є можливість завдання правомірності доступу, засекречені показники можна виділити в окремі відносини з перевіркою прав доступу.
 - простота впровадження – фізичне розміщення однорідних (табличних) файлів набагато простіше, ніж розміщення ієрархічних і мережних структур.
 - незалежність даних – БД повинна допускати можливість розширення, тобто додавання нових атрибутів і відносин.

Оскільки серед перерахованих логічних моделей даних реляційна має значні переваги й малі недоліки, то вона й буде взята в основу для побудови СУБД.

Вибір концептуальної моделі

Для вибору концептуальної моделі даних розглянемо три їхні різновиди:

- семантична модель;
- фрейми;

					ВКРМ-123.22.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		14

У зв'язку з тим, що модель “сутність-зв'язок” найбільш близька по принципах організації до реляційної моделі й реалізація останньої на основі першої найбільш зручна, то як концептуальна модель обрана модель “ сутність-зв'язок”.

Процес моделювання

Виділення сутностей

Сутність “постачальник” є стрижневою сутністю розроблювальної моделі. З постачальником полягає договір, на підставі якого ведеться вся інша діяльність підприємстві по поставках, відправлення заявки постачальникам, одержання від них рахунку-фактури, відправлення замовлення на поставку, одержання товару, оплата поставки. Як ключ для даної сутності вводиться атрибут № Постачальника.

Всі сутності, їхні атрибути й ключі представлені в таблиці 2.1.

Таблиця 2.1 – Всі сутності , їхні атрибути й ключі

Назва сутності	Атрибут	Ключ
Договір	№ Договору, дата договору, сума договору, термін дії.	№ Договору
Постачальник	№ Постачальника, найменування постачальника, адреса, телефон.	№ Постачальника
Асортименти товарів	№ Товару, найменування товару.	№ Товару
Заявка	№ Заявки, асортименти заявки, номер договору, дата заявки.	№ Заявки
Замовлення	№ Замовлення, № Договору, асортименти замовлення, дата замовлення, номер рахунку.	№ Замовлення
Рахунок-фактура	№ Рахунку, асортименти рахунку, ціна за одиницю товару, сума рахунку.	№ Рахунку

Виділення зв'язків між сутностями

Виділення зв'язків між сутностями здійснюється на підставі аналізу предметної області. Всі виділені зв'язки представлені на рисунку 2.1.

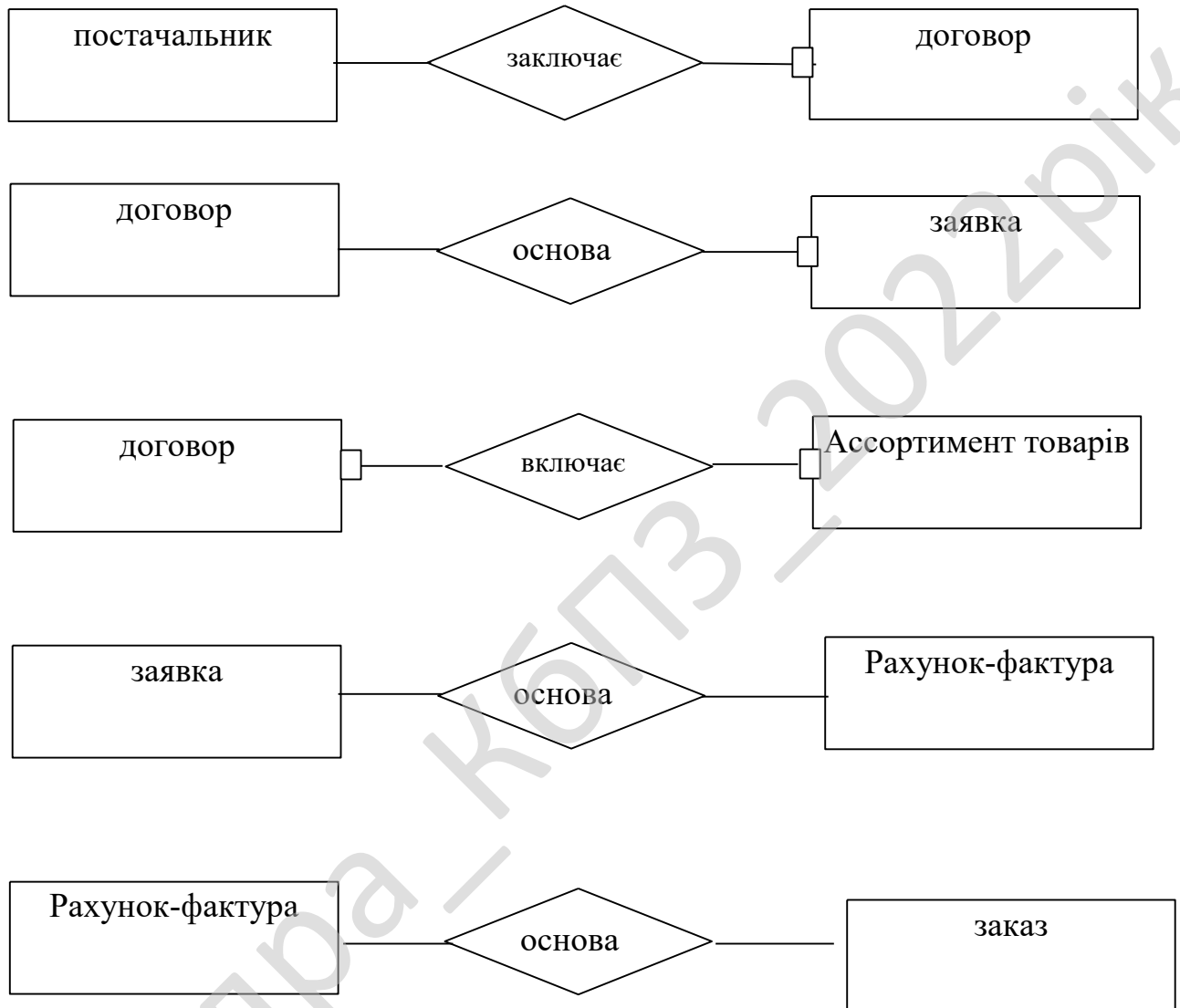


Рисунок 2.1 – Зв'язки між сутностями

Побудова логічної моделі

Виконавши аналіз сутностей і зв'язків меду ними побудуємо логічну модель, у вигляді відносин (таблиця 2.2).

Таблиця 2.2 – Логічна модель, у вигляді відносин

Назва сутності	Атрибут	Ключ
Договір	№ Договору, дата договору, сума договору, термін дії.	№ Договору
Постачальник	№ Постачальника, найменування постачальника, адреса, телефон.	№ Постачальника
Асортименти товарів	№ Товару, найменування товару.	№ Товару
Заявка	№ Заявки, номер договору, дата заявки.	№ Заявки
Заявка	№ Заявки, № товару, кількість.	№ Заявки, № Товару
Асортименти заявки	№ Замовлення, № Договору, дата замовлення, номер рахунку.	№ Замовлення
Асортименти замовлення	№ Замовлення, № Заявки, № товару.	№ Замовлення, № Заявки, № товару.
Рахунок-фактура	№ Рахунку, сума рахунку.	№ Рахунку
Ціни постачальника	№ Рахунку, № Заявки, № Товару.	№ Рахунку, № Заявки, № Товару.

Для побудови логічної моделі даних використовувався case – засіб [17] ER-Win, що дозволяє проектувати реляційні моделі даних як на фізичному рівні (ER-діаграми), так і на фізичному (проектування таблиць БД).

Логічна модель даних представлена у вигляді ER-діаграми на рисунку 2.2.

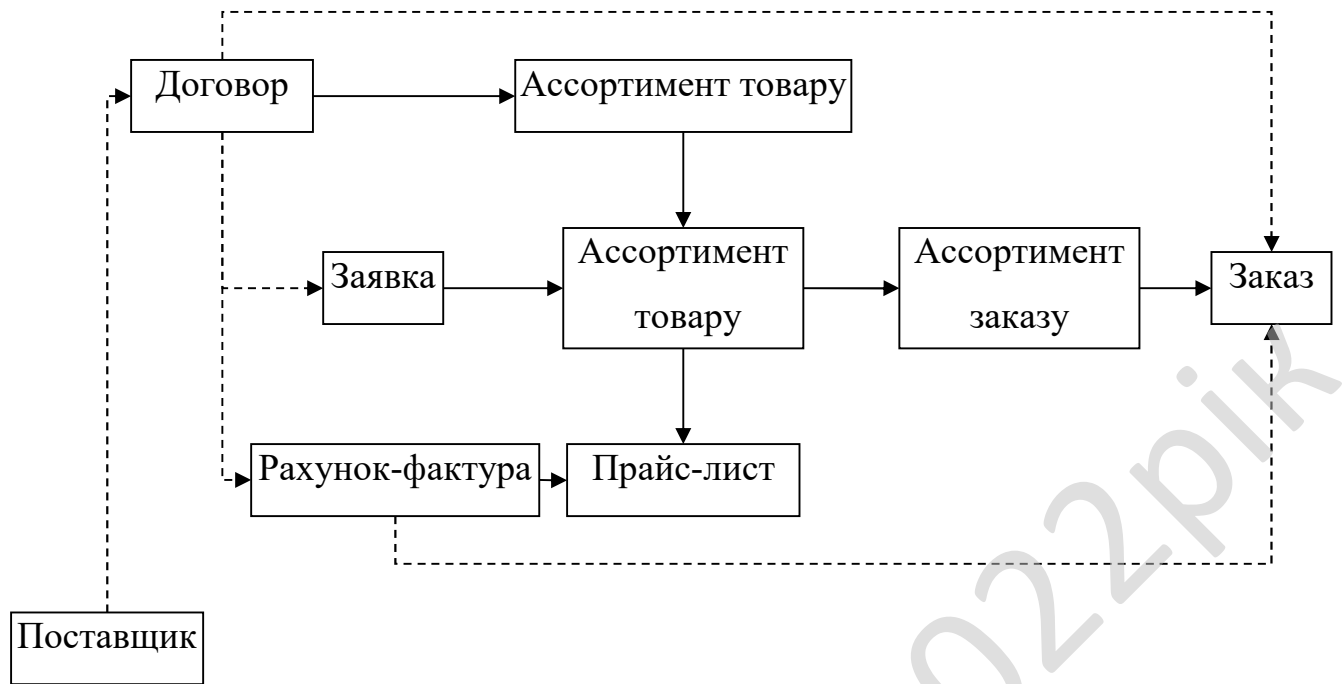


Рисунок 2.2 – ER-діаграма моделі даних АДІС “Облік поставок”

2.2 Обґрунтування вибору засобів для побудови системи та мови програмування

Embarcadero Delphi, раніше Borland Delphi і Codegear Delphi, – інтегроване середовище розробки ПЗ для Microsoft Windows, Mac OS, iOS і Android мовою Delphi (що раніше носила назву Object Pascal), створена спочатку фірмою Borland і на даний момент приналежна й розроблювальна Embarcadero Technologies. Embarcadero Delphi є частиною пакета Embarcadero RAD Studio і поставляється в чотирьох редакціях: Community (поширюється безкоштовно й має обмежену ліцензію на використання в комерційних цілях), Professional, Enterprise і Architect.

Delphi 10.4 Sydney

Випущено 26 травня 2020 року. RAD Studio Delphi 10.4 забезпечує значно поліпшену високопродуктивну нативну підтримку Windows, кращу продуктивність розробки, миттєві підказки code completion, прискорення виконання коду із синтаксисом керованих записів, поліпшення виконання паралельних завдань на сучасних багатоядерних CPU, а також містить більш

1000 виправлень багів, поліпшення продуктивності середовища й бібліотек і багато чого крім того.

Основні можливості Delphi 10.4.1:

– Істотні розширення для Windows: поліпшення для застосунків на моніторах 4K High DPI, інтеграція з новим WebView2 на базі Chromium, використання розширених title bars, таких же, як в Office, Explorer, Google Chrome.

– Керування пам'яттю в Delphi тепер стандартизоване на всіх підтримуваних платформах – мобільних, настільних і серверних – використовувачи класичну реалізацію керування пам'яттю об'єктів.

– Істотне поліпшення Delphi Code Insight (без можливого блокування IDE – в окремому процесі), що допоможе при роботі з великими проектами.

– Тип даних Delphi «record» тепер підтримуватимуть довільні ініціалізацію, фіналізацію й операції копіювання.

– Розширена підтримка бібліотек C++: ZeroMQ, SDL2, SOCI, libSIMDpp і Nematode.

– Відладник Win 64 (на LLDB) і збирач для C++.

– Поліпшення для C++: Включена велика кількість поліпшень STL з Dinkumware.

– Підтримка Metal Driver GPU для macOS і iOS.

– Вбудований Fmxlinux.

– Компонент Twebbrowser для iOS тепер реалізований на Wkwebview API.

Реалізація компонента Media Player для macOS тепер використовує Avfoundation.

Реалізований заново стилізуємий FMX компонент TМемо на платформі Windows значно поліпшений і тепер має відмінну підтримку ІМЕ.

– Численні поліпшення швидкості й стабільності роботи нашої бібліотеки The Parallel Programming Library (PPL).

– Додані оновлені драйвери для FireBird, PostgreSQL і SQLite.

					ВКРМ-123.22.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		20

Істотне поліпшення Delphi Code Insight

Як найбільше й головне поліпшення інструментів програмування Delphi за багато років, в 10.4 Delphi Code Insight реалізований через Language Server Protocol (LSP). LSP – це технологія генерації результатів для code completion, навігації й інших сервісів в окремому процесі. Це значить, що code completion і Code Insight одержать більш точні результати без блокування IDE. 10.4 забезпечує набагато більш високу продуктивність розроблювачів, які працюють із більшими проектами, що містять мільйони рядків коду.

Delphi Custom Managed Records

Ключове розширення мови Delphi: тип даних Delphi «record» тепер підтримуть довільні ініціалізацію, фіналізацію й операції копіювання. Управляйте тем, як ці структури створюються, копіюються й звільнюються з допомогу вашого коду, який буде виконуватися у відповідний момент.

Це розширює потужність конструкцій records в Delphi, які використовуються щоб одержати більшу ефективність у порівнянні із класами.

Єдине керування пам'яттю

Керування пам'яттю в Delphi тепер стандартизоване на всіх підтримуваних платформах – мобільних, настільних і серверних – використовувачи класичну реалізацію керування пам'яттю об'єктів.

У порівнянні з Automatic Reference Counting (ARC), це дає кращу сумісність із існуючим кодом і спрощує написання компонентів, бібліотек і застосунків.

ARC модель керування пам'яттю model залишилася для керування рядками й посиланнями на тип інтерфейсу на всіх платформах. Для C++ це означає, що при створенні й звільненні Delphi-style класів в C++ використовується звичайне керування пам'яттю, як у будь-якого heap-allocated класу C++, що значно знижує складність коду.

					ВКРМ-123.22.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		22

масштабується під відповідне до монітора дозвіл для показу форми. Був оновлений API стилізації для підтримки стилів high DPI.

Кожний графічний елемент UI може бути обраний з наборів різних масштабів і масштабований до потрібного DPI, що дає чітке зображення елементів UI на всіх моніторах.

Нові High DPI стилі й стилізація окремих VCL компонент

Обновлено велике число вбудованих і преміальних VCL стилів для підтримки нового режиму стилізації High-dpi. Це дозволяє вам створювати застосунку з відмінним дизайном для всіх моніторів.

Розроблювачі VCL застосунків тепер можуть використовувати трохи VCL стилів на різних формах в одному застосунку або в різних компонентах на одній формі. Це також включає стилізацію компонентів загальною темою для платформи. Крім застосункової гнучкості використання стилів, це дозволяє використовувати нестилізовані компоненти із зовнішніх бібліотек в VCL застосунках, що використовують стиль.

Поліпшена кроссплатформеність

- Додана підтримка Metal Driver GPU для macOS і iOS.
- Крім підтримки останнього iOS SDK, в RAD Studio 10.4 розроблювачі можуть задовольнити нові вимоги Apple до набору стартових екранів.
- Реалізований заново стилізуємий FMX компонент TМемо на платформі Windows значно поліпшений і тепер має відмінну підтримку IME.
- Користувачам редакцій Enterprise або Architect доступна повна інтеграція Fmxlinux з IDE для створення клієнтських застосунків Linux з GUI.
- Компонент Twebbrowser для iOS тепер реалізований на Wkwebview API.
- Реалізація компонента Media Player для macOS тепер використовує Avfoundation.

					ВКРМ-123.22.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		24

Оновлений менеджер пакетів Getit

Менеджер пакетів Getit в IDE був значно вдосконалений.

Дати випуску релізів пакетів тепер видні, і можливе сортування списку по цих датах; відбір тільки встановлених пакетів, контенту, доступного тільки при наявності підписки, багато чого іншого.

Універсальний інсталятор для установки Online і Offline

В 10.4 включений новий універсальний інсталятор, який використовує технологію на базі Getit. Цей інсталятор підтримує як online, так і offline (з ISO) варіанти установки.

Тепер обоє варіанта установки дозволяють вам указати початковий набір можливостей RAD Studio для установки, наприклад, свою комбінацію мов програмування й цільових платформ, мов інтерфейсу, і додавати до нього або видаляти непотрібне в будь-який момент.

2.3 Розгорнута постановка завдання

Згідно з технічним завданням на випуск кваліфікаційну роботу за другим (магістерським) рівнем вищої освіти, реалізації підлягає програмне забезпечення, яке призначено для системи довідково-інформаційного сервісу комп'ютерної мережі відділу підприємства.

В процесі розробки випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти необхідно виконати наступний обсяг роботи:

а) провести аналіз існуючих систем-аналогів для виявлення їх позитивних і негативних якостей. Результати аналізу врахувати в подальших розробках;

б) вибрати та обґрунтувати методикку побудови системи контролю роботи технологічного обладнання на виробництві в автоматизованому режимі.

Розробити функціональну та структурну схеми системи;

в) розробити програмне забезпечення системи, що дозволить реалізувати поставлену технічним завданням задачу. Побудувати блок-схеми алгоритмів

					ВКРМ-123.22.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		25

програми та підпрограми;

г) організувати інтерфейс користувача з метою формування та виводу на екран ЕОМ повідомлень про некоректні дії користувача та нестандартні ситуації в роботі технологічного обладнання;

д) розробити рекомендації по організаційних та методичних заходах, які забезпечать впровадження системи в промислову експлуатацію та її подальшу успішну експлуатацію;

е) провести розрахунки по визначенню економічної ефективності розробленої системи;

ж) розробити заходи по охороні праці при впровадженні та експлуатації системи, а також розробити заходи з цивільного захисту;

з) сформуванати висновки про виконаний обсяг робіт та одержані результати.

					ВКРМ-123.22.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		26

3 ОПИС І ОБҐРУНТУВАННЯ ПРОЕКТНИХ РІШЕНЬ

3.1 Опис функціонування системи

Проектування алгоритмів програмного забезпечення системи довідково-інформаційного сервісу комп'ютерної мережі відділу підприємства

Алгоритмізація в самому загальному виді може бути визначена як процес спрямованої дії проектувальника (групи проектувальників), необхідний для вироблення алгоритмів, достатніх для реалізації створюваного об'єкта (системи), що задовольняє заданим вимогам [19]. Завершальним етапом алгоритмізації є випуск набору алгоритмів, що відображає рішення, прийняті проектувальником, у формі, необхідної для виробництва об'єкта (системи). При проектуванні системи я використовував три класи алгоритмів:

- алгоритми, пов'язані із проектуванням АДІС;
- алгоритми реляційної алгебри, необхідні для роботи із БД;
- алгоритми розрахунку необхідних показників (обчислення заборгованості підприємства по оплаті поставок, визначення оптимального рахунку-фактури).

Вибір методу проектування АДІС

Метод – це послідовний процес створення моделей, які описують цілком певними засобами різні сторони розроблюваної програмної системи [18]. Методи важливі з кількох причин. По-перше, вони впорядковують процес створення складних програмних систем. По-друге, вони дозволяють менеджерам у процесі розробки оцінити ступінь просування й ризик.

Звичайно методи проектування діляться на три основні групи;

- метод проектування зверху долілиць;
- метод потоків даних;
- об'єктно-орієнтоване проектування.

					ВКРМ-123.22.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		27

Для структурного проектування характерна алгоритмічна декомпозиція. Слід зазначити, що більшість програм написана відповідно до цього методу. Проте структурний підхід не дозволяє виділити абстракції й забезпечити обмеження доступу до даних; він також не надає достатніх засобів для організації паралелізму. Структурний метод не може забезпечити створення гранично складних систем, і він, як правило, неефективний в об'єктних і об'єктно-орієнтованих мовах програмування. Тому даний метод не використовувався для проектування АДІС "Облік поставок".

У методі потоків даних програмна система розглядається як перетворювач вхідних потоків у вихідні. Метод потоків даних з успіхом застосовувався при рішенні ряду складних задач, зокрема, у системах інформаційного забезпечення, де існують прямі зв'язки між вхідними й вихідними потоками системи й де не потрібно приділяти особливу увагу швидкодії. Але оскільки одним з основних вимог пропонує до проектуваного АДІС є збільшення швидкості автоматизації обліку поставок і зменшення тимчасових витрат на оформлення поставок на підприємстві, то застосування даного методу також недоцільно для проектування АДІС.

Об'єктно-орієнтоване проектування (object-oriented design, OOD) – це підхід в основі якого лежить подання про те, що програмну систему потрібно проектувати як сукупність взаємодіючих один з одним об'єктів, розглядаючи кожний об'єкт як екземпляр певного класу, причому класи утворюють ієрархію. Об'єктно-орієнтований підхід відбиває топологію новітніх мов високого рівня, таких як Object Pascal, C++, Smalltalk [23] і ін. Моделі, для проектування якої використовується вищезгаданий підхід проектування властиві чотири головних елементи:

- абстрагування;
- інкапсуляція;
- модульність;
- ієрархія.

Абстрагування дозволяє виділити істотні характеристики проектного об'єкта, що відрізняють його від інших об'єктів;

Інкапсуляція – процес відділення друг від друга елементів об'єкта, що визначають його пристрій і поведження. Вона дозволяє ізолювати контрактні зобов'язання абстракції від їхньої реалізації.

Модульність – властивість системи, що була розкладена на внутрішньо зв'язкові, але слабо зв'язані між собою модулі.

Ієрархія – упорядкування абстракцій, розташування їх по рівнях.

Абстракція й інкапсуляція доповнюють один одного. Абстрагування спрямоване на спостереження поведження об'єкта ззовні, а інкапсуляція визначає чіткі границі між різними абстракціями, тобто спостереження за поведженням об'єкта зсередини.

Використання цих елементів проектування й дозволяє значно збільшити продуктивність будь-якої проектової системи.

Таким чином, для проектування АДІС використовується об'єктно-орієнтований підхід.

Аналіз алгоритмів роботи з базою даних

Система керування розробленої БД використовує реляційний підхід для побудови бази даних [20]. Подібні системи засновані на реляційній моделі даних, які використовуються для моделювання взаємозв'язків між об'єктами реального миру й для зберігання даних про ці об'єкти. Застосування реляційної моделі даних [27] обумовлено використанням реляційної алгебри й відповідних алгоритмів і операцій для виконання дій над даними. Використання алгоритмів реляційної алгебри [20] дозволяє забезпечити високу продуктивність роботи з базою даних.

Основні операції реляційної алгебри були вперше запропоновані Коддом [26]. Він довів, що запити, формулюємі за допомогою мови виражування можуть бути сформульовані в мовах реляційної алгебри й навпаки [18], т.е запити

					ВКРМ-123.22.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		29

представлені за допомогою мови реляційної алгебри можуть бути використані для виконання запитів до розробленого БД. Нижче наведений ряд запитів до БД:

```
SELECT nomer_dogovora, postav.nomer_postav, dogovor.nomer_postav,  
       naimen_post
```

```
FROM   postav, dogovor
```

```
WHERE  postav.nomer_postav=dogovor.nomer_postav
```

```
SELECT nomer_zajavki, zajavka.nomer_dogovora,  
       dogovor.nomer_dogovora, naimen_post, postav.nomer_postav,  
       dogovor.nomer_postav
```

```
FROM   from zajavka, dogovor, postav
```

```
WHERE  (zajavka.nomer_dogovora=dogovor.nomer_dogovora)
```

```
       AND (postav.nomer_postav=dogovor.nomer_postav)
```

```
SELECT nomer_zakaza, zakaz.nomer_dogovora, dogovor.nomer_dogovora,  
       naimen_post, postav.nomer_postav, dogovor.nomer_postav
```

```
FROM   zakaz, dogovor, postav
```

```
WHERE  (zakaz.nomer_dogovora=dogovor.nomer_dogovora)
```

```
       AND (postav.nomer_postav=dogovor.nomer_postav)
```

Розглянемо чотири операції над відносинами [20]:

- селекція;
- проекція;
- теоретико-множинне об'єднання;
- з'єднання.

Селекція (selected_on – піддані селекції по) зменшує кількість рядків у таблиці, і її можна представити як результат розрізування таблиці по горизонталі й видалення непотрібних кортежів. Формально селекція записується так:

R selected_on [<предикат>] {синтаксис мови запитів (SQL)}

Тут <предикат> – це логічне вираження, що може містити порівняння значень одних атрибутів зі значеннями інших у тім же кортежі або з константами. У результаті зберігаються тільки рядки, що задовольняють <предикату>.

Операція селекції відповідає програмам, які вибирають запису з файлів і друкують ці записи. Однак умови відбору можуть ставиться тільки до окремо взятих записів. Наприклад, неможливо вибрати запис, виходячи з того, що

					ВКРМ-123.22.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		30

вилучені з відношення-результату. Дана операція аналогічна об'єднанню множин в алгебрі, але вона є додатковою стосовно обмеження, тому що є можливість відновити стосунки шляхом об'єднання двох доповнюючих друг друга результатів операції селекції.

Операція теоретико-множинного відношення відповідає відомій операції “злиття” файлів. Якщо відомо, що файли не перетинаються, і якщо порядок записів не грає ролі, то досить скопіювати один файл наприкінці іншого. Однак, як правило, файли підтримуються в порядку первинних ключів, і тоді використовуються прості алгоритми злиття., що зчитують по черзі запису з кожного файлу залежно від того, у якому з файлів запис має ключ із меншим значенням полів, так що в новий файл запису також будуть міститися в порядку первинних ключів.

З'єднання (`joined_to` – з'єднання `c`) має два операнда; воно визначено для будь-яких двох таблиць. Якщо ці дві таблиці не мають стовпців зі співпадаючими іменами, то з'єднання поводить, як декартовий добуток, з'єднуючи кожний рядок першої таблиці по черзі з кожним рядком другої таблиці. Якщо імена всіх стовпців цих двох таблиць збігаються, то з'єднання поводить, як теоретико-множинне перетинання, і створює таблицю, що складається з тих рядків, які зустрічаються в кожній з розглянутих двох таблиць (така таблиця може бути й порожній, аналогічно порожній множині). Якщо у двох таблиць-операндів збігаються лише деякі імена стовпців, то в результаті з'єднання виходить таблиця, що містить усі імена стовпців першої таблиці, а також всі ті імена стовпців другій таблиці, які не зустрілися в першій. Рядки результату вибираються з першої таблиці, а додаткові значення конкатенуються (приєднуються) з тих рядків другої таблиці, у яких значення в загальних стовпцях збігаються. Певною мірою з'єднання є доповненням проекції, якщо здійснити проекцію “вихідного” відношення так, щоб вийшов набір відносин, кожне з яких зберігає первинний ключ вихідного, то з'єднання цього відношення відновить вихідне при додатковій

					ВКРМ-123.22.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		32

умові, що кожний стовпець вихідного відношення зустрічається хоча б в одній із проєкцій.

При формулюванні запитів операція з'єднання є вирішальною, якщо в запиті використовується більше одного відношення. Як правило, для формування запиту використовується з'єднання декількох таблиць, а потім селекція необхідних рядків, і, нарешті, проєкція на необхідні стовпці при печатці.

Операція з'єднання найбільше відповідає операції “селективної вибірки”, при виконанні якої список ключів представлений у вигляді записів у файлі транзакцій [19], і потрібно вибрати або записати у вихідний файл відповідні записи з основного файлу. Ключі у файлі транзакцій можуть збігатися, наприклад, зі стороннім ключем в основному файлі або ж із частиною первинного ключа, і в цих випадках для кожного запису у файлі транзакцій може бути обрано кілька записів з основного файлу. Таким чином, використовується з'єднання як узагальнене перетинання [20].

Алгоритми, які виконують перераховані вище операції, реалізуються на рівні системи керування базою даних. Їхній зміст формується на основі визначень цих операцій. Для їхньої реалізації використовуються або стандартні функції мови програмування, або формується SQL-запит.

3.2 Розробка структурної схеми

Автоматизований довідково-інформаційний сервіс комп'ютерної мережі відділу логістики (АДІС) “Облік поставок” призначена для автоматизації процесів керування матеріальними, інформаційними й фінансовими потоками на підприємстві. До задач підсистеми ставляться організація й контроль діяльності підрозділів підприємства за висновком договорів, організації надходження, переміщення й складування матеріальних цінностей, а також збуту готової продукції.

					ВКРМ-123.22.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		33

Структурна схема автоматизованої довідково-інформаційної системи комп'ютерної мережі відділу логістики (АДІС) "Облік поставок" зображена на рисунку 3.1.



Рисунок 3.1 – Структурна схема системи

Підсистема "Логістика" включає у свій склад наступні модулі:

- Модуль "Ведення договорів".
- Модуль "Постачання".
- Модуль "Збут".
- Модуль "Складський облік".

Модуль "Постачання" призначений для автоматизації діяльності підрозділів, що займаються забезпеченням підприємства товарами, матеріалами й комплектуючими.

Модуль виконує наступні функціональні задачі:

- Ведення довідкових цін і строків поставки матеріалів і комплектуючих по постачальниках.
- Одержання інформації про наявність товарів на складах.
- Визначення об'єму закупівель на основі планів діяльності, стану складу й запасів у цехах.
- Вибір постачальника, розподіл замовлень на закупівлю між постачальниками.
- Формування замовлення, прив'язка замовлення до договору.
- Передача інформації про очікувані поставки на склад.
- Одержання інформації про оплату й поточний стан замовлення на закупівлю.
- Перегляд списку пов'язаних із замовленням документів.
- Підготовка звітних форм.

Модуль "Збут" призначений для автоматизації діяльності підрозділів, що займаються збутому продукції підприємства.

Модуль виконує наступні функціональні задачі:

- Ведення прайс-аркушів на продукцію.
- Формування замовлення, прив'язка замовлення до договору.
- Формування заявки на відвантаження зі складу.
- Виписка рахунків (формування документів-підстав).
- Формування накладних на відвантаження продукції.
- Формування рахунків-фактур.
- Одержання інформації про оплату й поточний стан замовлення.
- Підготовка звітних форм.

Модуль "Складський облік" призначений для проведення складських операцій (приходу, витрати й внутрішнього переміщення товару), а також для обліку наявності товарів на логічних, фізичних складах і всього підприємства в цілому.

					ВКРМ-123.22.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		35

Модуль виконує наступні функціональні задачі:

- Ведення складського обліку в розрізі підприємства, фізичних і/або логічних складів.
- Формування й облік поставки товарів.
- Формування й облік відвантаження товарів.
- Формування й облік заявок на поставку товарів.
- Формування й облік документів внутрішнього переміщення товару.
- Розрахунок залишків і наявності товарів на складах.
- Списання товару при проведенні складських операцій приходу, витрати й внутрішнього переміщення товару.
- Можливість формування первинних документів (прибуткова накладна, видаткова накладна, накладна внутрішнього переміщення).
- Одержання звітності про стан складу.

Модуль "Ведення договорів" призначений для автоматизації діяльності підрозділів підприємства, що займаються роботою з договорами. Даний модуль дозволяє проводити весь цикл робіт за договором, починаючи від етапу розгляду комерційної пропозиції й підготовки договору, до моменту його закриття. Крім цього контролюються етапи проходження договору, його оплата, існує можливість знаходження всіх пов'язаних з договором документів.

Модуль "Договору" виконує наступні функціональні задачі:

- Внесення відомостей про договір.
- Виділень етапів договору.
- Підготовка специфікації до договору у формі замовлення.
- Перегляд інформації про оплати й виконання договору.
- Перегляд пов'язаних з договором документів.
- Підготовка звітних форм.

					ВКРМ-123.22.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		36

3.3 Розробка функціональної схеми

На рисунку 3.2 наведена функціональна схема системи.

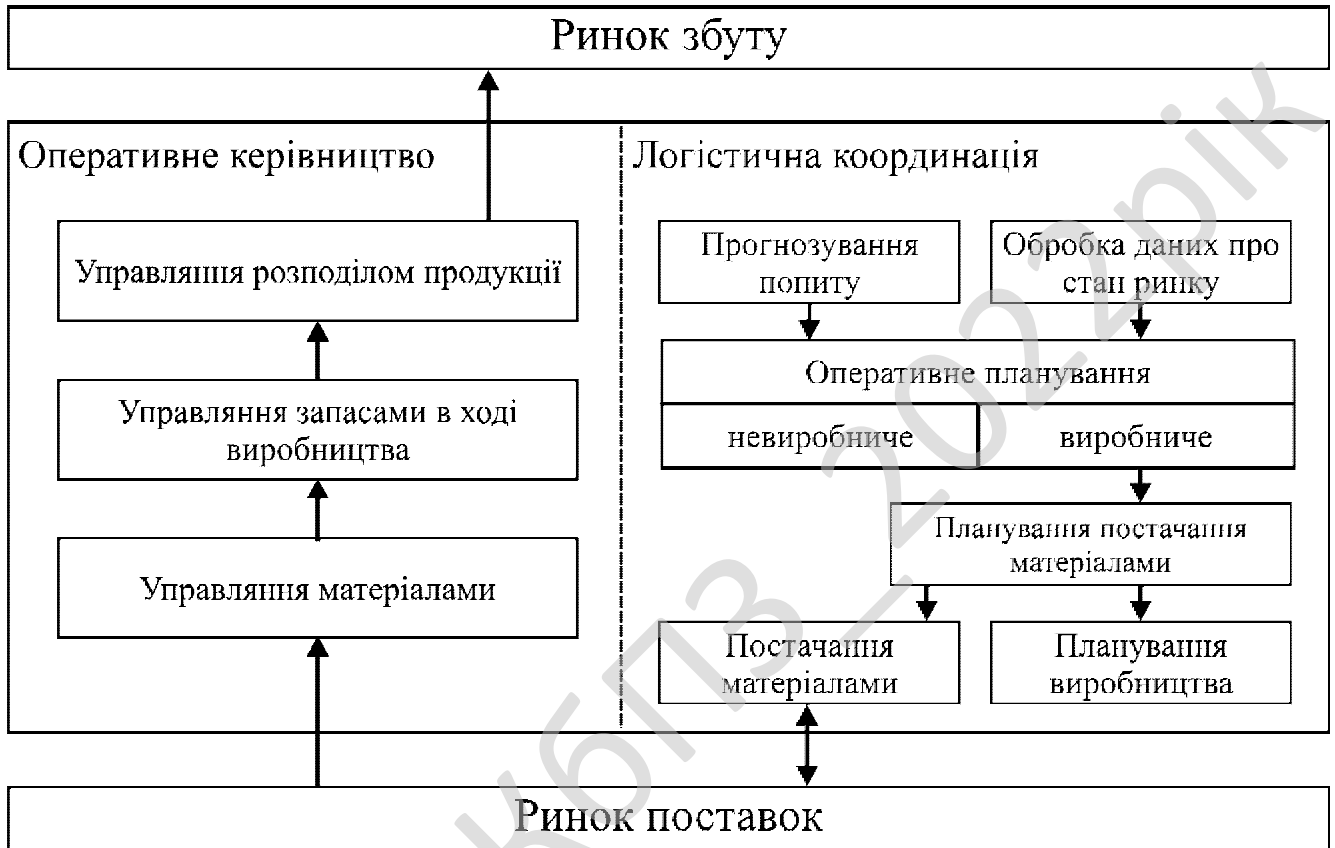


Рисунок 3.2 – Функціональна схема системи

Одне з основних завдань логістики складається у створенні інтегрованої ефективної системи регулювання й контролю матеріальних і інформаційних потоків, що забезпечує високу якість поставки продукції. З цим завданням самим тісним образом сполучене рішення таких проблем, як:

- відповідність один одному матеріальних і інформаційних потоків;
- контроль за матеріальним потоком і передача даних про нього в єдиний центр;
- визначення стратегії й технології фізичного переміщення товарів;
- розробка способів керування операціями руху товарів;

- встановлення норм стандартизації напівфабрикатів і упакування;
- визначення обсягу виробництва, транспортування й складування;
- розбіжність між наміченими цілями й можливостями закупівлі й виробництва.

Відповідно до сучасних завдань логістики розрізняють два види її функцій: оперативні й координаційні

Оперативний характер функцій пов'язаний з безпосереднім керуванням рухом матеріальних цінностей у сфері постачання, виробництва й розподілу, по суті, мало чим відрізняється від функцій традиційного матеріально-технічного забезпечення.

До функцій у сфері постачання відноситься керування рухом сировини й матеріалів, окремих частин або запасів готової продукції від постачальника або пункту їхнього придбання до виробничих підприємств, складам або торговельним сховищам.

У фазі виробництва функцією логістики стає управління запасами, що включає контроль руху напівфабрикатів і компонентів через всі стадії виробничого процесу, а також переміщення готової продукції на оптові склади й роздрібні ринки збуту.

Функції керування розподілом продукції охоплюють оперативну організацію потоків кінцевої продукції від підприємства-виробника до споживачів.

До числа функцій логістичної координації (див. мал.3) ставляться: виявлення й аналіз потреб у матеріальних ресурсах різних фаз і частин виробництва; аналіз ринків, на яких діє підприємство, і прогнозування поведінки інших джерел цих ринків; обробка даних, що відносяться до замовлень і потреб клієнтури (рисунок 3.2). Перераховані функції логістики укладаються в координації попиту та пропозиції на товар. У цьому змісті маркетинг і логістика тісно взаємозалежні, а формула, що затвердилася, – «маркетинг формує попит, а логістика його реалізує» – має під собою вагому

					ВКРМ-123.22.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		38

підставу. Певною мірою формула застосовна й до координації взаємин логістики й виробництва. Таким чином, логістика займається «стикуванням» двох сфер: пропонуваного ринком попиту й висуваної компанією пропозиції, що базується на відповідній інформації. У рамках координаційних функцій логістики виділилося ще один з її напрямків – оперативне планування, продиктоване прагненням зменшити запаси, не знижуючи ефективності виробничої й збутової діяльності фірм. Суть його полягає в тому, що на підставі прогнозу попиту, коректируемого пізніше при надходженні реальних замовлень, розробляються графіки перевезень і в цілому порядок керування запасами готової продукції, що у підсумку й визначає планування виробництва, розробку програм постачання його сировиною й комплектуючими виробами.

3.6 Розробка діаграми процесів

На рисунку 3.3 зображена діаграма процесів розробленої системи.

З неї ми бачимо, що у системі існують три основні процеси:

- з'єднання з базою даних;
- роз'єднання з базою даних;
- робота з базою даних.

Для роботи з базою даних потрібно, щоб після процесу з'єднання з базою даних, виконався процес авторизації. Більш детально, цей процес буде розглянуто у наступному розділі.

Робота з базою даних дозволяє взаємодіяти з наступними процесами:

- процес заключення договорів;
- процес формування та заповнення заявок;
- процес виставлення рахунку-фактури;
- процес формування замовлення тог, або іншого товару;
- процес друкування змін у базі даних, та друкування потрібних документів.

					ВКРМ-123.22.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		39

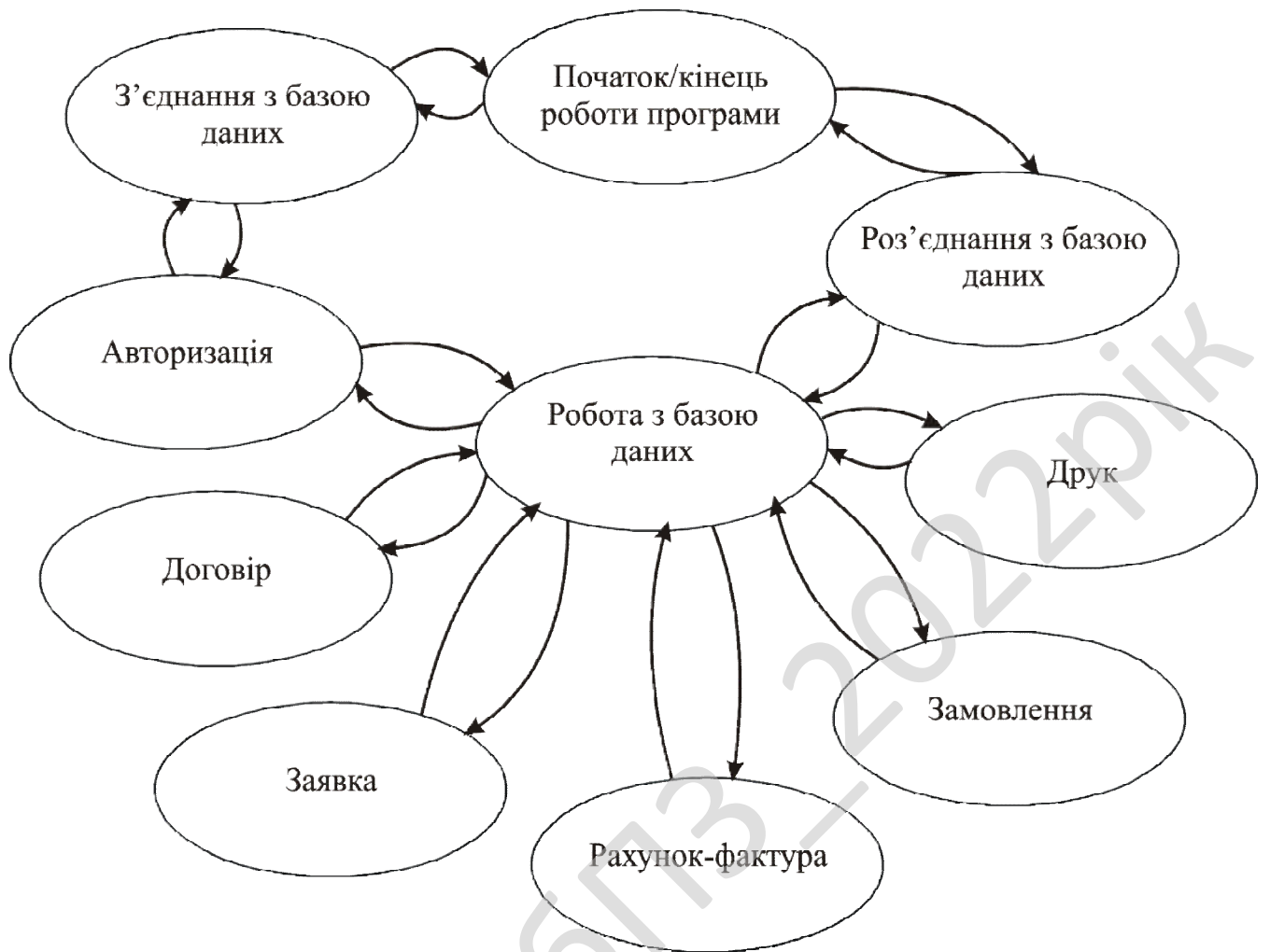


Рисунок 3.3 – Діаграма процесів системи

Таким чином, розглянувши опис системи, структурну, функціональну схеми системи, та діаграму взаємодії процесів перейдемо до опису блок-схем основної програми, та підпрограм, які використовуються, для реалізації системи.

4 РЕАЛІЗАЦІЯ РОБОТИ. РОЗРАХУНКИ І ЕКСПЕРИМЕНТАЛЬНІ ДАНІ, ЩО ПІДТВЕРДЖУЮТЬ ВІРНІСТЬ ПРОЕКТНИХ ТА ПРОГРАМНИХ РІШЕНЬ

4.1 Блок-схеми та опис алгоритмів функціонування системи

Одним із найважливіших етапів при написанні програмного забезпечення є побудова блок-схеми алгоритму, на основі якої розробляється та будується програма.

Блок-схема основної програми зображена на рисунку 4.1. Вона досить повно показує хід роботи програми.

Робота з програмою починається з виведення вікна авторизації, на якому потрібно ввести логін та пароль.

Користувач вводить пароль й програма визначає які права доступу має користувач:

- права адміністратора;
- права користувача;
- відсутність будь яких прав доступу.

Якщо права доступу відсутні, то програма повертається до вікна авторизації, або закінчується робота з програмою.

При наданні прав адміністратора або користувача розблоковуються, у першому випадку, усі функції, у другому випадку, функції доступні для користувача.

Після отримання відповідних прав, виводиться головне вікно програми.. На цьому вікні виводиться база даних підприємства.

Після цього починається робота з базою даних підприємства.

У програмі це виглядає наступним чином.

```
procedure ShowRecord;  
//створення форм для введення записів до БД
```

					ВКРМ-123.22.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		41

```

begin
  Form3.Edit1.Text:=NoteData.org;
  Form3.RichEdit1.Text:=NoteData.adr;
  Form3.Edit2.Text:=NoteData.tel;
  Form3.Edit3.Text:=NoteData.kont1;
  Form3.Edit6.Text:=NoteData.email;
  Form3.ComboBox2.Text:=Notedata.reg;
  Form3.ComboBox3.Text:=Notedata.vidd;
  Form3.ComboBox4.Text:=Notedata.osnprod;
  Form3.ComboBox5.Text:=Notedata.meneg;
end;
procedure ClearData;
//очищення даних у БД
begin
  Form3.Edit1.Text:='';
  Form3.RichEdit1.Text:='';
  Form3.Edit2.Text:='';
  Form3.Edit3.Text:='';
  Form3.Edit6.Text:='';
  Form3.ComboBox2.Text:='';
  Form3.ComboBox3.Text:='';
  Form3.ComboBox4.Text:='';
  Form3.ComboBox5.Text:='';
end;
// Pos починається з нуля - 0,1,2,3,4,5,6,...
// FileSize починається з одиниці - 1,2,3,4,5,6,7,8,...
function GetProgramPath : String;
//встановлення шляху до БД
begin
  GetProgramPath:=ExtractFilePath(ParamStr(0));
end;

procedure TForm3.FormCreate(Sender: TObject);
var
  pPos, cPos, i, j: integer;
  s: string;
begin
  AssignFile(NoteFile,GetProgramPath+'data\'+'adsbaz.dat'); // зв'язуємо змінну з файлом
  if not FileExists(GetProgramPath+'data\'+'adsbaz.dat') then // якщо файл не існує
    begin
      ForceDirectories(GetProgramPath+'data');
    end;
end;

```

					ВКРМ-123.22.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		42

```

Rewrite(NoteFile); // створення файлу з БД у якому зберігається БД
_Pos:=0;
end
else
begin // якщо файл з БД існує, то... (CopyFile - резервне копіювання)

CopyFile(pchar(GetProgramPath+'data\'+'adsbaz.dat'),pchar(GetProgramPath+'data\'+'
adsbaz_.dat'),false);
Reset(NoteFile); // відкриття файлу з БД
_Pos:=0;
if not Eof(NoteFile) then // якщо у файлі БД є дані
begin
Seek(NoteFile,_Pos); // переміщення на Pos
Read(NoteFile,NoteData); // читання даних з БД
ShowRecord; // показати дані
if _Pos=0 then Previous.Enabled:=false;
Delete.Enabled:=true;
if FileSize(NoteFile)>=2
then Next.Enabled:=true; // якщо даних більше або дорівнює 2
end;
end;

```

Роглянемо які дії відбуваються при натисненні на відповідні кнопки.

Натиснення кнопка «Документ». У цьому випадку відбувається перехід до роботи з документами.

Натиснення кнопка «Заявка». У цьому випадку відбувається робота з заявками.

Натиснення кнопка «Рахунок-фактура». У цьому випадку відбувається робота з відповідними рахунками.

Натиснення кнопка «Замовлення». У цьому випадку відбувається робота з замовленнями на підприємстві.

Натиснення кнопка «Друк». У цьому випадку відбувається робота з друком вказаних сторінок.

Натиснення кнопка «Вихід». У цьому випадку відбувається закінчення роботи з програмою, або перехід до головного вікна програми, в залежності від потреби.

					ВКРМ-123.22.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		43

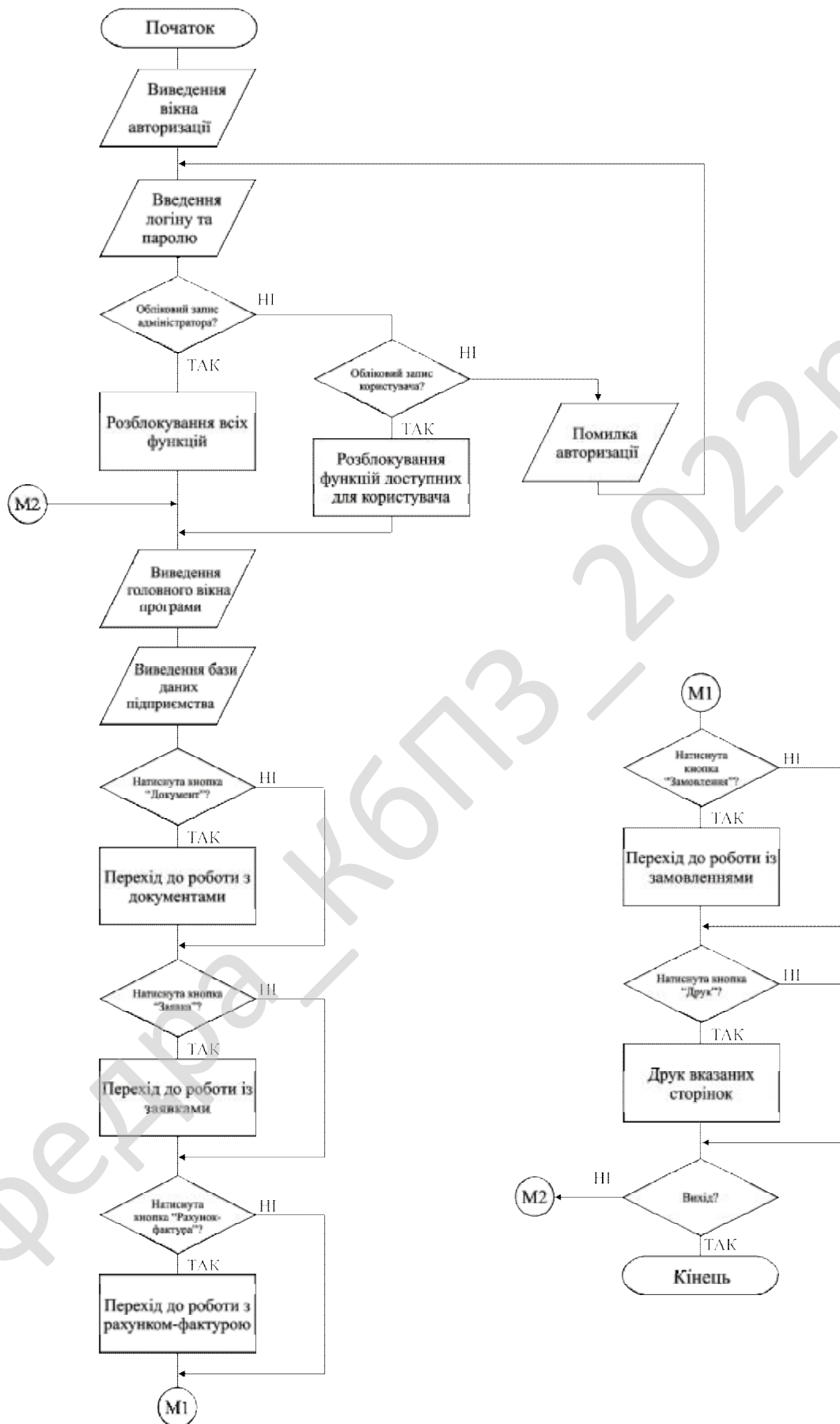


Рисунок 4.1 – Блок-схема роботи основної програми

На рисунку 4.2 представлена блок-схема підпрограми обчислення залишку товару на складі. Вона працює за нижче описаним алгоритмом.

Спочатку зменшується кількість товару на вітрині у відповідності до числа, вказаного при замовленні. Далі перевіряється чи не досяг залишок товару на вітрині допустимого мінімального значення – якщо кількість товару допустима, здійснюється повернення до основної програми. Якщо ж ні, то обчислюється число недостатньої кількості товару для поповнення на вітрині, і на визначену кількість зменшується число товару на складі і збільшується на вітрині.

Дані про код та кількість товару додаються до списку товарів, що потрібно відправити на вітрину. Задля уникнення дефіциту товару на складі перевіряється чи не досяг його запас допустимого мінімального значення. Якщо кількість товару допустима, то здійснюється повернення до основної програми, якщо ж відбулося досягнення рівня мінімального запасу, то перевіряється чи не було вже зроблено замовлення на дану партію товару.

У разі виявлення, що очікується прибуття нової партії потрібного товару, здійснюється повернення до основної програми. Якщо ж товар не очікується, то обчислюється недостатня кількість товару на складі і формується лист замовлення постачальнику на потрібний товар. Для цього виконується запит до бази даних і визначаються дані про постачальника товару та номер договору, який було з ним укладено.

Сформований лист відправляється на електронну адресу постачальнику, а для товару, що було замовлено, встановлюється прапорець очікування. Для контролю за замовленням, виконаним в автоматичному режимі, менеджеру електронною поштою надсилається звіт про доставку.

На рисунку 4.3 зображена блок-схема роботи підпрограми прийому товару.

					ВКРМ-123.22.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		45

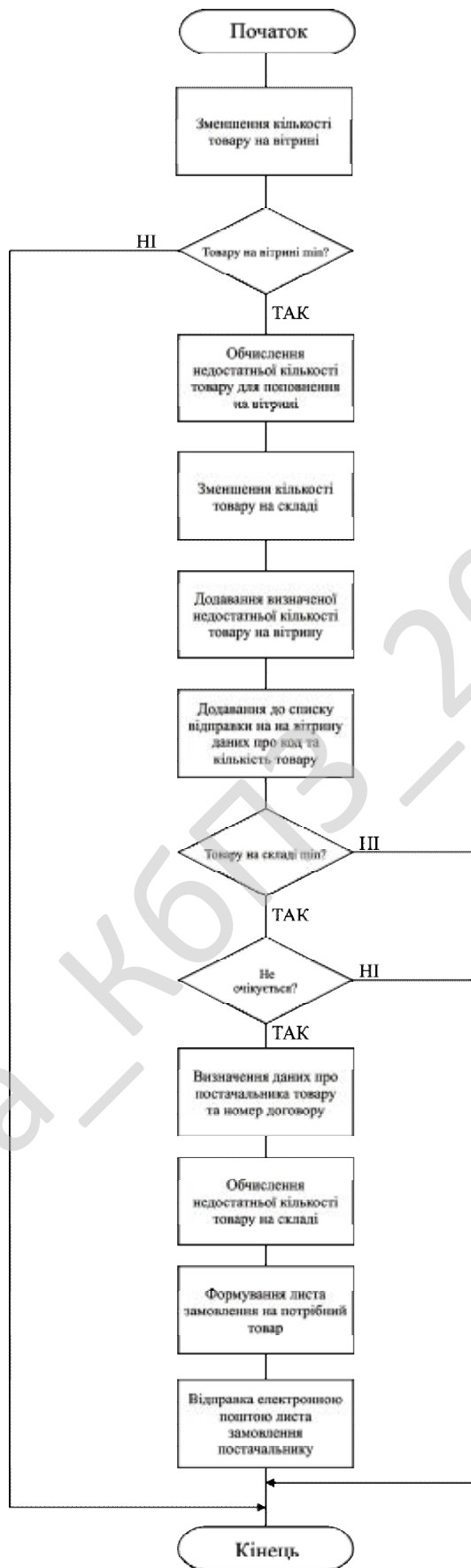


Рисунок 4.2– Блок-схема роботи підпрограми обчислення залишку товару на складі

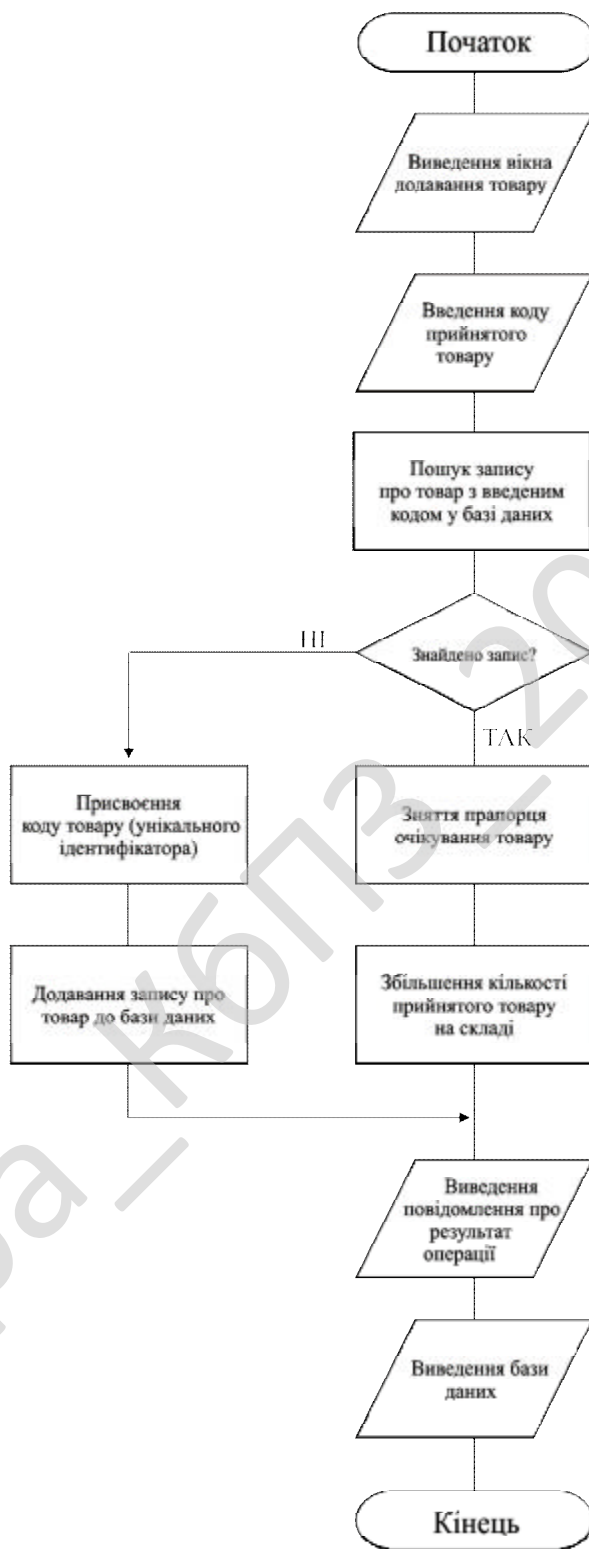


Рисунок 4.3– Блок-схема роботи підпрограми прийому товару

Після введення коду прийнятого товару виконується запит до БД з метою знаходження запису про товари із отриманим кодом. Якщо пошук закінчився успішно, то для даного товару знімається прапорець очікування і збільшується кількість товару на складі на стільки одиниць, скільки було прийнято даного товару.

У випадку, якщо в результаті пошуку в БД не було знайдено запису, тоді прийнятому товару присвоюється внутрішній код, який являється унікальним ідентифікатором.

Нижче описана робота із програмним продуктом розробленим у результаті виконання завдання на магістерське проектування.

Робота з договорами

Робота з договорами містить у собі:

- робота з постачальниками;
- робота з договорами;
- робота з товарами;
- робота з укладеними договорами;
- робота з асортиментами договорів.

Договір укладається підприємством-замовником з підприємством-постачальником на поставку певного виду й асортиментів продукції. З одним постачальником може бути укладено кілька договорів. Як атрибути договори є наступні поля: номер договору, код постачальника, дата договору, сума договору, термін дії договору. Всі атрибути, крім терміну дії договори є обов'язковими для заповнення. На підставі договору виробляється подальша діяльність по поставках на підприємстві. Вона полягає в:

- робота із заявками;
- робота з рахунками;
- робота із замовленнями.

Для автоматизації використання АДС “Облік поставок” реалізована можливість печатки бланків документів договору, заявки, замовлення.

					ВКРМ-123.22.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		48

Додавання нового договору здійснюється шляхом вибору відповідної закладки й уведенні тексту в поля-атрибути таблиці. Додавання за умови, що для договору, що додається, відомий постачальник.

Редагування відбувається при натисканні клавіші Enter на обраному записі. Відбувається автоматична зміна всіх полів інших таблиць пов'язаних з номером договору, що редагується. Це зміна необхідно для підтримки послідовної цілісності в БД.

Для видалення певного договору необхідно два рази клацнути правою кнопкою миші на видаляється договорі, що. Автоматично вийдуть всі записи пов'язані з видаляється договором, що (заявки, рахунку-фактури, замовлення).

Робота з постачальниками

Робота з постачальниками складається в додаванні нового постачальника, його атрибутів, видаленні постачальника, редагуванні атрибутів постачальника: код постачальника (для кожного постачальника код унікальний), найменування постачальника, адреса й телефон постачальника. Всі атрибути, крім телефону є обов'язковими для заповнення, у випадку їхнього незаповнення виникає помилка.

Додавання постачальника виробляється в такий спосіб: користувач вибирає відповідну таблицю й заповнює атрибути постачальника.

Для редагування таблиці “постачальники” потрібно вибрати запис для редагування, натиснути клавішу Enter і змінити необхідну інформацію. Змінені атрибути постачальника автоматично змінюються в інших таблицях.

Видалення запису “постачальник” відбувається шляхом подвійного клацання мишею на видаляється записи, що. При цьому потрібен запит на підтвердження видалення запису.

Робота з товарами

Таблиця “товари” являє собою довідник товарів, які поставляються на підприємство. Атрибути цієї таблиці містять унікальний код для кожного товару й найменування товару. При висновку кожного нового договору необхідно заповнити таблицю асортименти договору.

					ВКРМ-123.22.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		49

Додавання нового запису в таблицю здійснюється шляхом уведення інформації про товар у рядки таблиці товари. Редагування – натисканням клавіші Enter на редагується строке, що, і зміні інформації.

Видалення – подвійним клацанням миші на строці, що видаляється.

Робота з укладеними договорами

Робота з даною таблицею для користувача обмежена, оскільки даними для її заповнення служать раніше заповнені таблиці (договір, постачальник).

Робота з асортиментами договорів

Робота з асортиментами договорів полягає в додаванні, редагуванні й видаленні найменування товару або товарів, які постачальник зобов'язується поставити замовникові на підставі переліку товарів, що поставляються, що вказується в укладеному договорі. Вищевказані операції проводяться аналогічно операціям у роботі з договорами.

Робота із заявками

Робота із заявками являє собою роботу із трьома закладками:

- Заявка.
- Асортименти заявки.
- Всі заявки.

Закладка “заявка” містить таблицю з даними про заявки, які зробив замовник постачальникові по одному з укладених договорів. Таблиця заявка містить атрибути: номер заявки, номер договору, дата заявки. Заповнення всіх атрибутів є обов'язковим. Номер договору один з ув'язнених, у протилежному випадку виникає помилка.

Користувач має можливість додавати, редагувати й видаляти записи.

Додати запис можна у випадку коли таблиця активна, тобто користувач здійснює роботу з нею. Таблиця автоматично переводиться в режим додавання записів при натисканні користувачем клавіші на порожньому рядку, або натисканням клавіші Insert. Для редагування необхідно вибрати запис для редагування й, нажавши клавішу Enter зробити редагування необхідного поля

					ВКРМ-123.22.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		50

запису. Видалення відбувається шляхом подвійного клацання мишею на обраній для видалення запису.

Закладка “асортименти товарів” містить таблицю з даними про зроблену заявку, а саме: номер заявки, номер товару, кількість замовленої продукції в прийнятих одиницях виміру (шт., кг., л., і т.п.). всі атрибути є обов'язковими до заповнення. Крім того, номер товару (код товару) може бути обраний тільки з номерів товару, які зазначені в довіднику товарів.

Видалення, додавання й редагування записів відбувається аналогічно закладці заявка.

Робота з рахунками

Для робота з рахунками пропонується закладка “рахунок-фактура”, що містить таблицю рахунку й поле для визначення оптимального рахунку. Таблиця “рахунку” включає атрибути: номер рахунку, номер заявки, номер договору, сума рахунку. Всі атрибути обов'язкові для заповнення. Асортименти рахунку відповідає асортиментам заявки. На закладку виводиться інформація (або надається для уведення) тільки по одному з укладених договорів, номер якого обраний у таблиці асортименти договорів.

Робота із замовленнями

Для роботи із замовленнями пропонується дві закладки:

- Замовлення.
- Всі замовлення.

У закладку “замовлення” включені таблиця “замовлення” з атрибутами: номер замовлення, номер договору, номер рахунку, отримано, оплачено, і поле для визначення заборгованості підприємства по оплаті виконаних поставок. Користувачеві надається можливість додавання, редагування й видалення записів. Всі операції із записами здійснюються для певного договору, зазначеного в закладці асортименти договору. Всі атрибути таблиці обов'язкові до заповнення. Заповнення полів таблиці оплачене й отримано можна здійснювати зі списку, що випадає, із двома рядками (так, немає).

					ВКРМ-123.22.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		51

5 МЕТОДИКА ВПРОВАДЖЕННЯ СИСТЕМИ В ПРОМИСЛОВУ ЕКСПЛУАТАЦІЮ

Після запуску файлу `postavki.exe` на виконання на моніторі з'являється головне меню (рисунок 5.1):

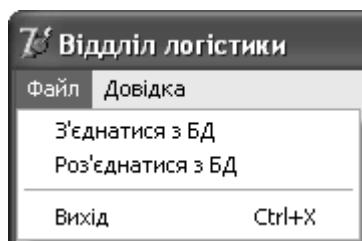


Рисунок 5.1 – Головне меню АДІС

Для початку роботи із програмою необхідно з'єднатися з базою даних, для чого клацнути по команді меню з'єднається із БД. Якщо на комп'ютері користувача встановлений InterBase Local Server і створена база даних, то з'явиться запит на підтвердження права доступу до БД (рисунок 5.2):

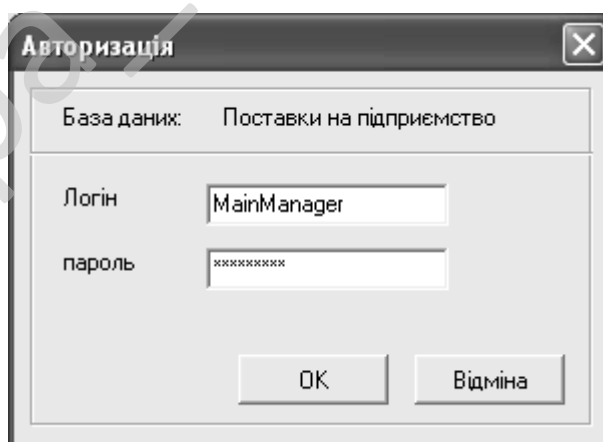


Рисунок 5.2 – Вікно уведення пароля

Пароль доступу QWERTY.

У випадку, якщо з'єднання пройшло успішно, то користувач допускається до роботи з АДІС.

Робоче вікно АДІС виглядає в такий спосіб (рисунок 5.3):

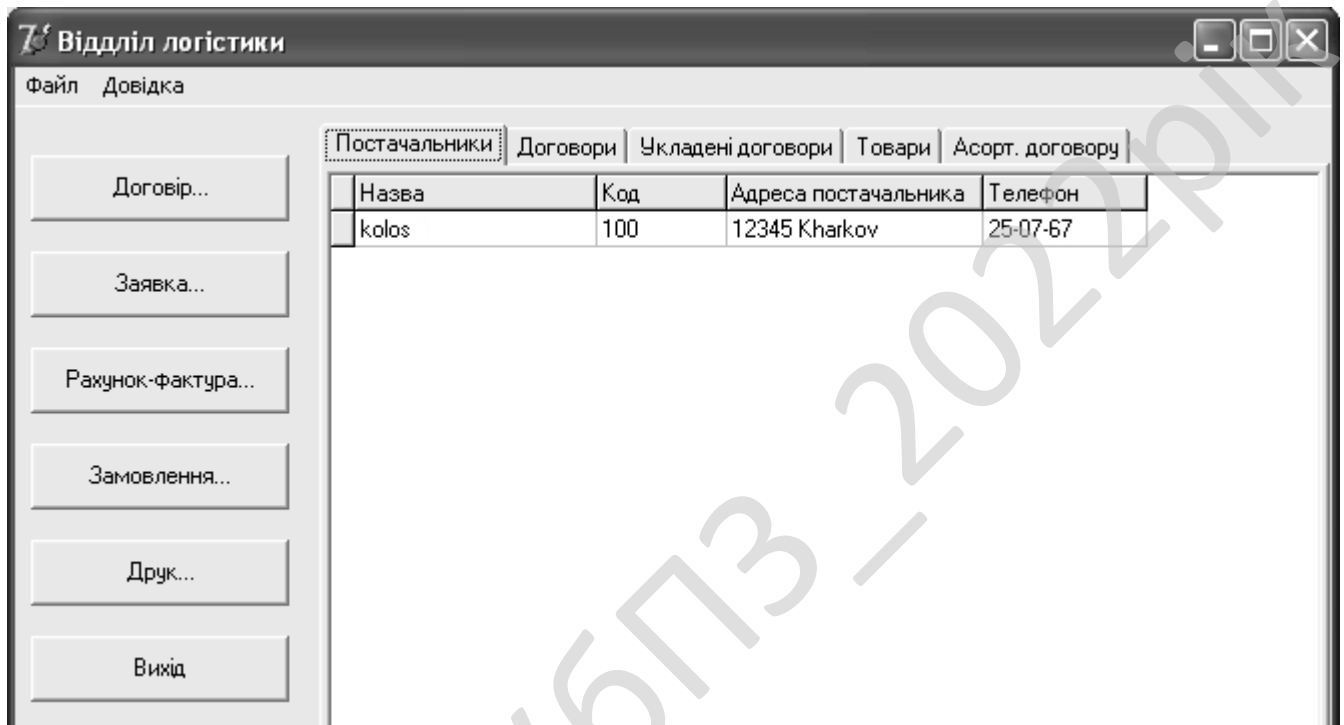


Рисунок 5.3 – Робоча область АДІС

Коротку довідку про розроблену програму та її автора можна переглянути вибравши підпункт «Про програму...» з меню «Довідка», після чього з'явиться вікно зображене на рисунку 5.4.

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Центральноукраїнський національний технічний
університет
Кафедра кібербезпеки та програмного забезпечення
ВИПУСКНА КВАЛІФІКАЦІЙНА РОБОТА
за другим (магістерським) рівнем вищої освіти
на тему: "Дослідження та програмна реалізація системи
довідково-інформаційного сервісу комп'ютерної мережі відділу
підприємства"
ОПП «Комп'ютерна інженерія»
спеціальності 123 «Комп'ютерна інженерія»
Виконав: Борзенко В.Ю.
Науковий керівник: Буравченко К.О.
Кропивницький - 2022

ок

Рисунок 5.4 – Довідка про програму

					ВКРМ-123.22.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		55

6 НАУКОВА НОВИЗНА

У випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти розроблено програмне забезпечення, яке призначено для системи довідково-інформаційного сервісу комп'ютерної мережі відділу підприємства.

Метою розробки є дослідження та програмна реалізація системи довідково-інформаційного сервісу комп'ютерної мережі відділу підприємства.

Об'єктом дослідження є процес довідково-інформаційного сервісу комп'ютерної мережі відділу підприємства.

Предметом дослідження є методи довідково-інформаційного сервісу комп'ютерної мережі відділу підприємства.

Методи дослідження базуються на методах теорії телекомунікацій, методах математичної статистики, методах розробки програмного забезпечення.

Наукова новизна отриманих результатів. У процесі рішення завдань, обумовлених цілями дослідження, отримані наступні результати:

– Удосконалено метод довідково-інформаційного сервісу комп'ютерної мережі відділу підприємства.

– Розроблено вітчизняний продукт довідково-інформаційного сервісу комп'ютерної мережі відділу підприємства, який має більш широкі можливості, на відміну від існуючих аналогів.

					ВКРМ-123.22.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		56

7 ДАНІ ПРО ЕКОНОМІЧНУ ЕФЕКТИВНІСТЬ РОЗРОБЛЕНОЇ ПРОГРАМИ

7.1 Техніко-економічне обґрунтування теми випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти

Після ознайомлення з підприємством та засобами розробки програмної продукції був розроблений план розробки програми. Був підрахований необхідний час для розробки та впровадження програми. Цей час склав 48 днів (два місяці).

В магістерській роботі проведено дослідження та виконана програмна реалізація системи довідково-інформаційного сервісу комп'ютерної мережі відділу підприємства. Розроблене програмне забезпечення має достатню надійність і задовольняє усім поставленим умовам, а саме:

- а) невеликий розмір;
- б) невеликі системні потреби;
- в) незалежність від встановлених на комп'ютері баз даних;
- г) зручність у користуванні та надійність

Таблиця 7.1 – Початкові данні

Показники	Позначення	Характеристика або величина
1	2	3
1. Кількість розроблених програм період, шт	N	1
2. Кількість екземплярів програм, шт	Ne	57
3. Запланований термін розробки, днів	Fpq	48 (2 місяці)
4. Група задачі підсистеми управління (1-6)	–	1
5. Ступінь новизни задачі (А, Б, В, Г)	–	Б
6. Складність алгоритму (1, 2, 3)	–	2
7. Кількість макетів вхідної інформації	–	3

Продовження таблиці 7.1

1	2	3
8. Кількість форм вихідної інформації.	–	4
9. Мова програмування (1-6)	–	2
10. Попередній досвід (1-6)	–	3
11. Гнучкість проекту ПП (1-6)	–	3
12. Детальність проекту ПП (1-6)	–	2
13. Рівень спрацьованості колективу (1-6)	–	2
14. Ступінь вимірності процесів (1-6)	–	3
15. Необхідна надійність програмного забезпечення (1-6)	–	2
16. Розмір бази даних (порівняно з розміром програми) (1-6)	–	2
17. Складність кінцевого програмного продукту (1-6)	–	2
18. Необхідний рівень забезпечення повторного використання (1-6)	–	2
19. Документованість відповідно до планованого життєвого циклу (1-6)	–	2
20. Вимоги до швидкодії ПП (1-6)	–	2
21. Обмеження на розміри основного сховища даних (1-6)	–	2
22. Різноманітність використовуваних обчислювальних платформ (1-6)	–	2
23. Професійний рівень аналітиків (1-6)	–	2
24. Професійний рівень програмістів (1-6)	–	2
25. Постійність складу команди розробників (1-6)	–	2
26. Досвід розробки додатків (1-6)	–	2
27. Досвід роботи з обчислювальною платформою (1-6)	–	2

Продовження таблиці 7.1

1	2	3
28. Досвід роботи з мовою і інструментами середовища розробки (1-6)	–	2
29. Досвід роботи з програмними інструментами розробки (1-6)	–	3
30. Розробка ПО для декількох серверів одночасно (1-6)	–	2
31. Вимоги до дотримання встановленого графіка робіт (1-6)	–	2
32. Вартість ПЗ у розробника (НМА), грн	–	60000
33. Норматив додаткової зарплати, % :	Н _д	10
34. Норматив відрахувань у соціальні фонди, %	Н _с	22
35. Норматив загальногосподарських витрат, %	Н _г	15
36. Норматив витрат на освоєння нових мов програмування, %	Н _п	15
37. Рівень рентабельності програмної продукції, %	Р _е	50
38. Ставка податку на додану вартість, %	Н _{дв}	20

7.2 Розрахунок трудомісткості розробки програмної продукції

Значення трудомісткості розробки програмного забезпечення для стадій ТЗ, ЕК, ТП та ВП визначаємо по типовим нормам часу приведеним в додатках МВ. Стадія РП є найбільш тривалою і трудомісткою, що робить значний вплив на інші стадії проекту.

Визначимо трудомісткість розробки ПЗ для стадії РП.

Обчислюємо номінальні трудовитрати, люд-міс.:

$$T_{ном} = A \text{ Size}^B \quad (7.1)$$

де А – коефіцієнт Боєма, А=2,45;

					ВКРМ-123.22.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		59

Size – загальний об'єм відлагодженого програмного коду, тис. рядків;

B – показник ступеня, що визначається співвідношенням

$$B = 1,01 + 0,001 \sum W_i \quad (7.2)$$

де W_i – сумарне значення п'яти показників (МВ, додаток 2), що відображають особливості розробки проекту програмного продукту (ПП) і колективу розробників.

$$B = 1,01 + 0,001(2,43 + 3,64 + 3,38 + 3,95 + 2,73) = 1,026$$

$$T_{ном} = 2,45 \cdot 2,7^{1,026} = 6,78 \text{ люд-міс.}$$

Визначаємо уточнені (з урахуванням приведених в МВ додатку 3 сімнадцяти додаткових коефіцієнтів) трудовитрати, люд-міс.:

$$T_{уточн} = T_{ном} \prod V_j, \quad (7.3)$$

де $\prod V_j$ – добуток сімнадцяти додаткових коефіцієнтів, приведених в МВ додатку 3.

$$T_{уточн} = 6,78 \cdot (0,88 \cdot 0,93 \cdot 0,88 \cdot 0,91 \cdot 0,95 \cdot 1,1 \cdot 0,87 \cdot 1,22 \cdot 1,16 \cdot 1,1 \cdot 1,1 \cdot 1,12 \cdot 1,1 \cdot 1,1 \cdot 1,1) = 9,37 \text{ люд-міс.}$$

Ці коефіцієнти дозволяють диференційовано оцінювати результати роботи програмістів, беручи до уваги швидкодію програми, використання різноманітних обчислювальних платформ і інструментів розробки, взаємодію декількох серверів, вимоги до об'ємів баз даних і ін.

Визначаємо підсумкові трудовитрати по стадії робочий проект, люд-дні:

$$T_{РП} = 0,3 C T_{уточн}^{0,33+0,2(B-1,01)} S, \quad (7.4)$$

де C – визначений емпірично коефіцієнт, запропонований авторами методики, (МВ, додаток 4); S – коефіцієнт стиснення (або подовження) графіка робіт %, що дозволяє коректувати терміни розробки ПО згідно встановленим вимогам. Вибираємо в межах (25...350)%

$$T_{РП} = 0,3 \cdot 2,66 \cdot 9,37^{0,33+0,2(1,026-1,01)} \cdot 65 = 109 \text{ люд/день}$$

Для зручності визначення загальної трудомісткості на розробку програмного забезпечення результати розрахунків по стадіям зводимо до таблиці 7.2.

					ВКРМ-123.22.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		60

Таблиця 7.2 – Визначення трудомісткості розробки програмного забезпечення

Стадії розробки	Трудомісткість за типовими нормами та розрахунками	
	Величина, люд/дні	Підстава
Технічне завдання	9	Д5
Ескізний проект	10	Д6
Технічний проект	9	Д7
Робочий проект	109	Ф 7.1-7.4
Впровадження	13	Д13
Всього	150	–

7.3 Визначення чисельності виконавців і планового фонду зарплати

Чисельність ставок інженерів-програмістів для розробки програмного забезпечення визначається за формулою

$$Ч = \frac{T_{nz} N}{F_{pq} - H_{ев}}, \quad (7.5)$$

де F_{pq} – плановий фонд робочого часу одного спеціаліста, днів,
 T_{nz} – трудомісткість розробки програмного забезпечення люд-дні,

$$Ч = \frac{150 \cdot 1}{48 \cdot 5} = 3,5 \text{ ставки}$$

Чисельність інженерів-електронщиків для проведення технічного обслуговування та ремонту комп'ютерних мереж визначається в залежності від наявності технічних засобів і норм витрат часу на виконання профілактичних робіт на протязі року.

Визначаємо затрати часу на виконання профілактичних робіт по обслуговуванню обладнання за період розробки. Результати розрахунку зводимо до таблиці 7.3

Таблиця 7.3 – Затрати часу на виконання профілактичних робіт по обслуговуванню обладнання за розрахунковий період

Найменування обладнання	Профілактичне обслуговування			
	Кількість хв. на один. обл.	Кількість обладнання	Затрати часу в хв.	Затрати часу в год.
Системний блок ПК	90	6	540	9
Монітор	60	6	360	6
Клавіатура	30	6	180	3
Маніпулятор «мишка»	30	6	180	3
Принтер матричний	60	0	0	0,0
Принтер лазерний	120	1	120	2
Принтер струминний	60	1	60	1
Сканер	20	1	20	0,33
Концентратор-маршрутизатор	30	1	30	0,5
Кабельні господарства ЛОМ на 1 м.п.	2,5	140	350	5,83
Копіювальний апарат	140	1	140	2,33
Усього за рік:			3 _ч	32,99

Час на профілактику обладнання в загальному балансі робочого часу інженерів-електронщиків не повинен складати більше 10%

Виходячи з цього фонд робочого часу інженерів-електронщиків складає:

$$\Phi_{op}^c = \frac{3_{ч} \cdot n_{mic}}{1,2} \quad (7.6)$$

$$\Phi_{op}^c = \frac{33 \cdot 3}{1,2} = 82,5 \text{ год}$$

					ВКРМ-123.22.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		62

Визначаємо необхідну кількість ставок штатного персоналу сектора ТО:

$$Ч_{ел} = \frac{\Phi_{др}^c}{F_{др} \cdot T_{зм}} \quad (7.7)$$

$$Ч_{ел} = 82,5 / (48 \cdot 8) = 0,2 \text{ ставки}$$

Для забезпечення нормального технічного обслуговування засобів ТО та мереж, необхідно прийняти найбільше ціле значення розрахункової чисельності інженерів–електронщиків.

Чисельність інженерів-системотехніків, адміністраторів мережі, дизайнерів WEB вузлів, системних програмістів (аналітиків), бухгалтерів-економістів визначається за потребою в залежності від функціональних обов'язків. Після визначення чисельності персоналу складається штатний розклад.

Таблиця 7.4 – Розрахунок чисельності штатного персоналу сектору системного та адміністративного обслуговування засобів ОТ та комп'ютерних мереж

Посада	Вид роботи	Час	К-ть штатних одиниць
Адміністратор загальної мережі, аналітик	Адміністрування локальної мережі, поштового та серверу DNS (OC FreeBSD), маршрутизатора Cisco, доменного контролера Windows Server 2019, серверу доступу ADSL (OC Linux), налаштування ADSL, VPN, PPPoE, Frame Relay, Wi-Fi	0,8	0,2
	Налаштування і конфігурування базової станції безпроводного зв'язку (CMTS)	0,2	
	Розробка та впровадження проектів з організації зв'язку між віддаленими об'єктами, ЛОМ	0,2	
	Забезпечення цілодобової роботи зв'язку клієнтів до мережі Інтернет	0,4	
Всього		1,6	

Продовження таблиці 7.4

Посада	Вид роботи	Час	Кількість штатних одиниць
Продакт-менеджер	Презентації нової продукції, пошук каналів збуту	2	0,5
	Підтримка постійних клієнтів	1	
	Оформлення договорів, ведення тендерів	0,5	
	Контроль взаєморозрахунків з постачальниками	0,5	
Всього		4	
Дизайнер WEB	Розробка концепції оформлення та інтерфейсу сайту, оптимізація дизайну існуючих, проектує їх структуру та навігацію	0,5	0,2
	Створення графічних і стилістичних елементів сайту	0,5	
	Оформлення банерів і промо-сторінок	0,3	
	Розміщення графіки і контенту на Інтернет сторінках	0,3	
Всього		1,6	
Інженер верстальник	Розробка та верстка макетів рекламної продукції та технічної документації	1	0,2
	Верстка друкованих видань	0,2	
	Додрукова підготовка макетів	0,2	
	Розміщення графіки і контенту на Інтернет сторінках	0,2	
Всього		1,6	

Складемо штатний розклад виконавців у таблицю 7.5.

Таблиця 7.5 – Штатний розклад виконавців

Посада	Кількість ставок	Середньо-місячний оклад, грн.	Всього за період розробки, грн.
Керівник (ІТ-менеджер)	1	19103	38206
Продакт-менеджер	0,5	17000	17000
Інженер-програміст	3,5	20000	140000
Інженер-електронщик	0,2	15000	6000
Інженер-системотехнік	0,2	15000	6000
Адміністратор мережі	0,2	11000	4400
Системний програміст	0,2	9000	3600
Дизайнер WEB	0,2	9000	3600
Інженер-верстальник	0,2	9000	3600
Бухгалтер-економіст	0,2	10000	4000
Всього за період розробки	$R_{cn}=6,4$	-	$\Phi_{роб}=226406$

Розрахуємо середньоденну зарплату одного виконавця:

$$z_{cd} = \frac{\Phi_{роб}}{R_{cn} F_{pq}}, \quad (7.8)$$

де $\Phi_{роб}$ – загальна сума зарплати за плановий період, грн.

$$z_{cd} = \frac{226406}{6,4 \cdot 48} = 737 \text{ грн.}$$

7.4 Розрахунок капітальних вкладень та амортизаційних відрахувань у розробника

Балансова вартість будівель визначається з урахуванням кількості робочих місць виконавців, питомої площі на одне робоче місце, та вартості одного квадратного метра виробничої площі:

					ВКРМ-123.22.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		65

$$B_{y\delta} = R_{cn}^1 S_y C_{nl}, \quad (7.9)$$

де R_{cn}^1 – кількість робочих місць виконавців, шт. Приймаємо 8 робочих місць.

S_y – питома площа на одне робоче місце, m^2 ,

C_{nl} – вартість одного квадратного метра площі, грн.

Згідно даних інтернет ресурсу DOM.RIA (<https://dom.ria.com>) ціна одного квадратного метра площі, вік якої не перевищує 30 років, по місту складає 500...1600 у.о./ m^2 . Враховуючи, що курс складає 1 у.о. = 38 грн. приймаємо для розрахунку вартість одного метра квадратного рівною 20000 грн./ m^2 . На кожне робоче місце у середньому потрібно $8 m^2$. З урахуванням цього:

$$B_{y\delta} = 8 \cdot 8 \cdot 20000 = 1280000 \text{ грн.}$$

Вартість передавальних пристроїв складає 10% від вартості будівель, і у даному випадку вона складе: 128000 грн.

Балансова вартість інвентарю розраховується за нормою 3500 грн на одне робоче місце. Тобто

$$I_{нв} = R_{cn}^1 \cdot C_m, \quad (7.10)$$

де C_m – ціна меблів для одного робочого місця, грн.

$$I_{нв} = 8 \cdot 3500 = 28000 \text{ грн}$$

Балансова вартість обчислювальної техніки визначається по оптовим цінам постачальника з врахуванням витрат на транспортування.

Специфікація на обчислювальну техніку наведена в таблиці 7.7. Дані по оптовій ціні на обладнання та комплектуючі вибирались за комерційною пропозицією фірми Brain за 14.11.22 – джерело <http://brain.com.ua/>

					ВКРМ-123.22.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		66

Таблиця 7.6 – Специфікація

Найменування комплектуючої або обладнання	Тип	Оптова ціна
Персональний комп'ютер		11771
Системний блок		7771
Процесор	Intel Core i7-2700K (S1155/4x3.5GH /5GT/s/8MB/95 Вт)	2500
Системна плата	Huanan B75 (s1155, Intel B75, PCI-Ex16 DVI/VGA/HDMI	1100
Жорсткий диск	HDD 500 Gb SAMSUNG Barracuda HD502HJ (3.5", 500ГБ, 16МБ, SATA II-300)	1290
Оперативна пам'ять	DIMM 4096Mb DDR3 PC3-12800 Kingstor 1600MHz, 512M x 64, CL9-9-9-27, 1.65V w/heatsink, HyperX	834
DVD-привод	DVD±RW ASUS DRW-24B5ST Black Bulk	416
Корпус	Logicpower 8702 – 550w 12cm	1411
Кардрідер внутрішній	Transcend TS-RDF8K USB 3.0	220
інше	Клавіатура, мишка	Подаруно
Монітор	Монітор BenQ GL2450HM Black	2600
Принтер лазерний	Canon i-SENSYS LBP6030W	2700
Принтер струминний	Epson Stylus Photo P50 (C11CA45341) + USB cable	5500
Сканер	Epson Perfection V37	2800
Копіювальний апарат	Canon i-SENSYS MF217W with Wi-Fi	5965
Пристрій безперебійног живлення	Powercom BNT-600AP USB	1400

					ВКРМ-123.22.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		67

Витрати на транспорт, монтаж та випробування можуть бути прийняті в межах до 10% від оптової ціни.

Для визначення необхідної кількості капітальних вкладень складемо таблицю 7.8.

Таблиця 7.7 – Балансова вартість обчислювальної техніки

Найменування обчислювальної техніки	Кількість, шт.	Ціна за одиницю, грн.	Витрати на транспортування, монтаж та випробування.	Загальна вартість, грн.
Персональні комп'ютери	8	11771	9416,8	103584,8
Принтер лаз.	2	2700	540	5940
Принтер струм.	1	5500	550	6050
Сканери	1	2800	280	3080
Копіюв. апарат	1	5965	596,5	6561,5
Всього	–	–	–	125216,3

Таблиця 7.8 – Вартість основних фондів та амортизаційні відрахування розробника

Групи та види основних фондів	Балансова вартість, грн.	Амортизація	
		Норма, %	Відрахування, грн.
1	2	3	4
Група 3			
1. Будівлі	1280000	-	-
2. Передавальні пристрої	128000	-	-
Всього по групі	1408000	5	70400

Продовження таблиці 7.8

1	2	3	4
Група 4			
3. Обчислювальна техніка	125216	-	-
Всього по групі	125216	50	62608
Група 5,6			
4. Вимірювальні пристрої	5190	-	-
5. Транспортні засоби	143000	-	-
6. Господарський інвентар	28000	-	-
Всього по групі	176190	20	35238
7. Нематеріальні активи	60000	10	6000
Разом	$K_p = 1769406$		$A_p = 174246$

Примітка: вартість автомобіля Chevrolet Aveo 2008 взята за даними сайту «Авто-Ріа», джерело https://auto.ria.com/uk/auto_chevrolet_aveo_32795647.html, складає 143000 грн.

7.5 Визначення собівартості розробки та ціни програмної продукції

Визначимо основну зарплату виконавців:

$$z_o = \frac{z_{cd} \cdot T_{nz}}{N_e}, \quad (7.11)$$

де N_e – Кількість екземплярів програм, шт.

$$z_o = 737 \cdot 150 / 57 = 1940 \text{ грн}$$

Визначимо додаткову зарплату (оплата відпусток, виконання державних та суспільних обов'язків) на рівні 10%

					ВКРМ-123.22.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		69

$$Z_o = Z_o \cdot H_q \cdot 0,01, \quad (7.12)$$

де H_q – норматив додаткової зарплати, %

$$Z_o = 1940 \cdot 10 \cdot 0,01 = 194 \text{ грн}$$

Відрахування на соціальні потреби за нормативом $H_c = 22\%$ від суми основної та додаткової зарплати

$$C_{oc} = 0,01 \cdot H_c (Z_o + Z_o), \quad (7.13)$$

де H_c – відрахування на соціальні потреби, %

$$C_{oc} = 0,01 \cdot 22(1940 + 194) = 470 \text{ грн}$$

Визначимо загальногосподарські витрати (електроенергію, ремонт і утримання приміщень і т.д) за нормативом $H_z = 15\%$ від основної зарплати

$$G_{ocn} = Z_o \cdot H_z \cdot 0,01, \quad (7.14)$$

де H_z – загальногосподарські витрати, %

$$G_{ocn} = 1940 \cdot 15 \cdot 0,01 = 291 \text{ грн}$$

Визначимо витрати на матеріали для розробки програмної продукції за нормами споживання та діючими цінами за одиницю виміру:

$$Z_M = (Z_{M1} + Z_{M2} + Z_{M3}) / N_e, \quad (7.15)$$

де Z_{M1} – вартість паперу, грн., Z_{M2} – вартість запам'ятовуючих пристроїв, грн., Z_{M3} – вартість фарби, картриджей, тонеру, грн., N_e – кількість екземплярів програм, шт.

Згідно прийнятих норм на підприємстві $n_{вум}$ приймаємо 0,4 пачки паперу на період розробки. Тоді, враховуючи, що вартість пачки паперу складає $C_n = 200$ грн., визначаємо вартість паперу за період розробки:

$$Z_{M1} = C_n \cdot N_m. \quad (7.16)$$

$$Z_{M1} = 200 \cdot 0,4 = 80 \text{ грн.}$$

Згідно прийнятих норм по комплектації до вартості запам'ятовуючих пристроїв входить вартість CD/DVD дисків. Їх кількість дорівнює кількості коробочних версій запропонованого продукту (приймаємо 10):

$$Z_{M2} = \sum C_d, \quad (7.17)$$

					ВКРМ-123.22.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		70

де: C_d – вартість дисків CD/DVD: CDR box – 23,6 грн./шт., DVD-R box – 46,6 грн./шт.

$$Z_{M2} = 46,6 \cdot 10 = 466 \text{ грн.}$$

Згідно норм одноразовій заправці підлягають усі друкуючі пристрої становить:

$$Z_{M3} = \sum C_z, \quad (7.18)$$

де: C_z – вартість розхідних матеріалів друкуючих пристроїв: відновлення та заправка картриджу для Canon i-SENSYS LBP6030W – 574 грн.; картридж для Epson Stylus Photo P50 – 558 грн.; відновлення картриджу для MF217W – 570 грн.

$$Z_{M3} = 574 + 558 + 570 = 1702 \text{ грн.}$$

$$Z_M = (80 + 466 + 1702) / 57 = 39 \text{ грн.}$$

Визначимо витрати на освоєння нових мов програмування або операційни систем за нормативом ($H_n = 15\%$) від основної зарплати виконавців

$$O_n = Z_o \cdot H_n \cdot 0,01, \quad (7.19)$$

де H_n – норматив витрат на освоєння нових мов програмування, %

$$O_n = 1940 \cdot 15 \cdot 0,01 = 291 \text{ грн}$$

Визначимо витрати на амортизацію основних фондів з урахування загальної річної суми амортизаційних відрахувань та кількості екземплярів програ ($N_e = 57$ прим.)

$$A_m = \frac{A_p \cdot N_{mic}}{N_e \cdot 12}, \quad (7.20)$$

де A_p – загальна річна сума амортизаційних відрахувань, грн.

$$A_m = 174246 \cdot 2 / (57 \cdot 12) = 509 \text{ грн}$$

Повна собівартість ПЗ визначається як сума витрат за попередніми статтям калькуляції

$$C_n = Z_o + Z_d + C_{oc} + \Gamma_{ocn} + Z_M + O_n + A_m. \quad (7.21)$$

$$C_n = 1940 + 194 + 470 + 291 + 39 + 291 + 509 = 3734 \text{ грн.}$$

Визначимо плановий прибуток за рівнем рентабельності (Рп) програмно продукції, яка залежить від складності програми та ступеня новизни задачі.

					ВКРМ-123.22.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		71

Для даного програмного забезпечення рівень рентабельності складає 50%

$$P_p = 0,01 \cdot P_n \cdot C_n, \quad (7.22)$$

де P_c – рівень рентабельності, %

$$P_p = 0,01 \cdot 50 \cdot 3734 = 1867 \text{ грн.}$$

Величини ціна підприємства, податок на додану вартість, відпускна ціна програмної продукції визначаються за формулами, приведеними в таблиці 7.9

Таблиця 7.9 – Нормативна калькуляція собівартості розробки програмного забезпечення задачі

Найменування статей витрат	Позначення	Величина, грн.
1	2	3
1. Основна зарплата виконавців	Z_o	1940
2. Додаткова зарплата виконавців	Z_d	194
3. Відрахування на соціальні потреби	C_{oc}	470
4. Загальногосподарські витрати	G_{ocn}	291
5. Витрати на матеріали	Z_M	39
6. Освоєння нових операційних систем, мов програмування	O_n	291
7. Амортизація основних фондів	A_m	509
8. Повна собівартість програмного забезпечення	C_n	3734
9. Плановий прибуток	P_p	1867
10. Ціна підприємства $C_n = C_n + P_p$	C_n	5601
11. Податок на додану вартість $ПДВ = 0,01 \cdot H_{ов} \cdot C_n$	$ПДВ$	1120,2
12. Відпускна ціна програмної продукції $Ц = Ц_n + ПДВ$	$Ц$	6721,2

Витрати на технічне обслуговування:

$$Z_p = T_p \cdot Z_z \cdot (1 + 0,01 \cdot H_q) \cdot (1 + 0,01 \cdot H_c), \quad (7.23)$$

де T_p – кількість годин обслуговування системи за рік, год.,

Z_z – заробітна плата обслуговуючого персоналу, грн/год

Після купівлі нового програмного забезпечення кількість профілактичних годин робіт зменшилася з 250 годин на рік до 150 годин на рік, тому витрати на технічне обслуговування зменшилися з

$$Z_{p \text{ баз}} = 250 \cdot 100 \cdot 1,1 \cdot 1,22 = 33550 \text{ грн.}$$

до

$$Z_{p \text{ нов}} = 150 \cdot 100 \cdot 1,1 \cdot 1,22 = 20130 \text{ грн.}$$

Витрати на електроенергію визначаються з урахуванням спожитої потужності ($P_{ел}$) в кіловатах, часу експлуатації технічних засобів (T_p) в годинах та ціни однієї кіловат-години ($C_{ел}$).

$$Z_{ел} = P_{ел} \cdot T_p \cdot C_{ел}. \quad (7.24)$$

$$Z_{ел \text{ баз}} = 0,2 \cdot 7200 \cdot 3,2 = 4608 \text{ грн}$$

$$Z_{ел \text{ нов}} = 0,2 \cdot 6600 \cdot 3,2 = 4224 \text{ грн}$$

Витрати по амортизації визначаються на основі норм амортизаційних відрахувань, вартості програмної продукції і основних фондів. Для розрахунку складаємо таблицю 7.12.

Таблиця 7.12 – Розрахунок амортизаційних відрахувань

Групи основних фондів	Норма амортизац ії %	Балансова вартість, грн., за варіантами		Сума відрахувань, грн., за варіантами	
		Базовий	Новий	Базовий	Новий
Програмна продукція	50	–	6721	–	3360,5
Всього відрахувань	-	–	6721	–	3360,5

$$T_{cn} = \frac{K_n - K_6}{I_6 - I_n} \quad (7.28)$$

$$T_{cn} = \frac{6721}{38158 - 27715} = 0,6 \text{ років}$$

Показники економічної ефективності програмної продукції зводимо до таблиці 7.13.

Таблиця 7.13 – Показники економічної ефективності програмної продукції

Найменування показників	Одиниця виміру	Величина
1. Кількість екземплярів програми	Прим.	57
2. Повна собівартість розробленої програми	Грн.	3774
3. Ціна розробленої програми	Грн.	5601
4. Плановий прибуток від реалізації розробленої програми	Грн.	1827
5. Рентабельність програмної продукції	%	50
6. Об'єм додаткових капітальних вкладень у виробника програмної продукції	Грн.	1769406
7. Загальний прибуток від реалізації програмної продукції	Грн.	104139
8. Величина економічного ефекту при виготовленні програмної продукції	Грн.	77378
9. Період окупності додаткових капітальних вкладень у виробника програмної продукції	Років	0,6
10. Об'єм додаткових капітальних вкладень у споживача програмної продукції	Грн.	6721
11. Величина економічного ефекту у користувача програмної продукції	Грн.	7083
12. Період окупності додаткових капітальних вкладень у користувача програмної продукції	Років	0,6

7.9 Висновки

Розроблена програма економічно вигідна. За рахунок впровадження програмного забезпечення досягається скорочення часу обробки інформації, підвищується культура праці, підвищення якості приймаючих управлінських рішень.

Кафедра _ КБПЗ _ 2022 рік

					VKPM-123.22.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		77

8 ЗАХОДИ З ОХОРОНИ ПРАЦІ ТА ТЕХНІКИ БЕЗПЕКИ

8.1 Вступ

Охорона праці – система збереження життя і здоров'я працівників у процесі трудової діяльності, що включає правові, соціально-економічні, організаційні, технічні, санітарно-гігієнічні, лікувально-профілактичні, реабілітаційні та інші заходи. Загальні положення державної політики, щодо галузі охорони праці зазначені у Законі України “Про охорону праці”. Цей Закон визначає основні положення щодо реалізації конституційного права працівників на охорону їх життя і здоров'я у процесі трудової діяльності, на належні, безпечні і здорові умови праці, регулює за участю відповідних органів державної влади відносини між роботодавцем і працівником з питань безпеки, гігієни праці та виробничого середовища і встановлює єдиний порядок організації охорони праці в Україні [20]. Законодавство про працю містить норми і вимоги з техніки безпеки і виробничої санітарії, норми, що регулюють робочий час і час відпочинку, звільнення та переведення на іншу роботу, норми праці щодо жінок, молоді, гігієнічні норми і правила тощо. Загальний нагляд за додержанням норм охорони праці покладено на прокуратуру, спеціальний – на професійні спілки. Контроль за безпекою праці здійснюють також, державні й відомчі спеціалізовані інспекції. Науково-технічний прогрес вніс серйозні зміни в умови виробничої діяльності робітників розумової діяльності. Їх праця стала більш інтенсивною, напруженою і вимагає значних витрат розумової, емоційної і фізичної енергії. Це призвело до необхідності у знаходженні комплексного рішення проблем ергономіки, гігієни і організації праці, регламентації режимів праці та відпочинку. Охорона здоров'я робітників, забезпечення безпеки умов праці, ліквідація та профілактика професійних захворювань і виробничого травматизму складає одну з головних турбот людського суспільства.

					ВКРМ-123.22.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		78

- віддалення фази розвитку втоми;
- забезпечення високої продуктивності праці за нормальних фізіологічних затрат.

Комплекс заходів щодо підвищення і збереження працездатності працівників на оптимальному рівні реалізується на техніко-організаційному, соціально-економічному, санітарно-гігієнічному, медико-біологічному, психологічному напрямках.

Вагомим фактором високої працездатності і продуктивності праці є оптимізація трудових навантажень на основі механізації і автоматизації виробничих процесів, удосконалення технології, скорочення і ліквідації важкої ручної праці. Доведено, що при правильній організації праці на легких роботах спостерігається найбільша тривалість фази стійкого стану, а на важких роботах вона нетривала.

Високий рівень працездатності безпосередньо залежить від умов праці, оскільки поліпшення їх супроводжується зменшенням енергетичних затрат організму на подолання несприятливого впливу факторів виробничого середовища.

Важливим напрямком підвищення працездатності працюючих є ритмізація трудових процесів, оптимізація темпу роботи, а також раціоналізація трудових рухів на фізіологічній основі, що сприяє формуванню і закріпленню робочих динамічних стереотипів, а отже зменшенню м'язових і вольових зусиль. Ритмічна робота підвищує функціональні можливості організму, сприяє його тренуваності і забезпечує економізацію енергетичних затрат. [1]

Багатьом програмістам постійно доводиться працювати з великою кількістю програм одночасно. Часте перемикання туди-сюди між IDE та довідкою суттєво зменшує продуктивність фахівця. Однак вирішення цієї проблеми досить просте та очевидне: встановлення більшої кількості моніторів.

Оптимальним варіантом є два монітори. Все ж таки це найпростіший з апаратної точки зору варіант. Крім того, якби їх було більше, то ними було б

важче керувати, та й столі просто не вистачить місця на ще один монітор. Але тут ще залежить розміру моніторів. Є системи із 4 або 6 відносно невеликими екранами, які кріпляться на кронштейні. Але оптимальним є два 27-дюймові монітори, на яких все добре видно, особливо коли працювати доводиться в основному з текстом [3].

8.4 Пожежна безпека

Вимоги до пожежної безпеки на підприємстві неухильно повинен дотримуватися кожен співробітник, а організаційна складова при цьому покладається на посадових осіб за відповідним рішенням керівництва і прописується в посадових інструкціях і положеннях по структурним підрозділам.

Зокрема, вказуються конкретні території, ділянки, зони, об'єкти, цілі будівлі і їх частини, поверхи, на яких відповідального співробітника повинне проводити такі організаційні роботи.

Відповідальні особи зобов'язуються розробити, впровадити та підтримувати в певному інструкцією і положенням на ввірених їм об'єктах протипожежний режим і інструкції відповідно до вимог, викладених в нормативних актах.

Передбачено також створення підрозділу добровільної пожежної охорони та пожежно-рятувальної команди в його складі.

Встановлений режим включає порядки з описом місць спеціального призначення та правила їх користування та утримання, наприклад:

- евакуаційних шляхів;
- так званих «курилок»;
- місць складування продукції та сировини;
- стоянки транспорту.

Також встановлюється порядок роботи та технічного обслуговування:

- вентиляційного устаткування;

					ВКРМ-123.22.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		83

- засобів пожежогасіння і захисту від загорянь;
- нагрівальних приладів;
- електрообладнання.

Розробляються і впроваджуються правила роботи з відкритим вогнем і горючими матеріалами. Створюються графіки проходження інструктажів з пожежної безпеки співробітників, а також порядок і терміни перевірок знань пожежно-технічного мінімуму, в тому числі, тих працівників, які відповідальні за цю ділянку роботи на підприємстві. При цьому можуть передбачатися внутрішні лекції, семінари, тренінги та практичні заняття на підприємстві, а також зовнішні – на базі спеціалізованих навчальних центрів з професійними викладачами.

Важливою складовою протипожежного режиму на будь-якому об'єкті є розробка і впровадження порядку дій при виникненні пожежі. Неодмінно має бути план евакуації, описано, як повинні відключатися електроустановки, що і в якій послідовності необхідно робити співробітникам.

Відповідно, для кожного об'єкта, кожного приміщення (крім коридорів, санвузлів, басейнів і подібних приміщень), окремих видів робіт складаються інструкції, за якими повинен працювати персонал, залучений на певних ділянках і в виконанні окремих видів робіт. За інструкціями проводиться навчання (інструктаж) персоналу з подальшим контролем знань.

Детально про те, як розробити протипожежний режим, прописати порядки та інструкції, пояснюють на тематичних курсах і семінарах. [24]

8.5 Розрахункова частина

Проведемо розрахунок штучного освітлення за методом коефіцієнта використання світлового потоку для приміщення, ширина якого складає 6 м, довжина – 7 м, висота – 3,4 м.

У зазначеному приміщенні працює 7 людей.

					ВКРМ-123.22.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		84

Підставимо всі значення у формулу та визначимо індекс приміщення:

$$i=1,4.$$

Знаючи індекс приміщення, за знаходимо $n = 0,23$ (з табличних даних коефіцієнтів використання світлового потоку (n) світильників з відповідним типом лампам) [8]. Підставимо всі значення у формулу, визначимо світловий потік: $F=90391$ Лм.

Для розрахунку будемо використовувати світлодіодні стельові панелі *Призма-72 6400К*, світловий потік яких $F_{л} = 7200$ Лм.

Число ламп визначається за формулою:

$$N=F/F_{л}$$

де:

F – світловий потік,

$F_{л}$ – світловий потік однієї лампи.

Підставимо всі значення у формулу та визначимо індекса приміщення:

$$N= 90391 / 7200=12,5 \text{ шт.}$$

Приймаємо необхідну кількість світлодіодних світильників 13 шт.

8.6 Висновки до розділу

Дотримання всіх необхідних умов праці не лише сприяє збереженню здоров'я працівників, а також підвищує ефективність виробництва в цілому.

З цих міркувань було здійснено аналіз умов праці, призначеного для праці програмістів, проведено розгляд небезпечних та шкідливих факторів, що негативно впливають на програмістів під час роботи. Виконано розрахунок штучного освітлення, як одного з ключових факторів впливу на працездатність та здоров'я програміста. Розроблено заходи з умов поліпшення охорони праці.

					ВКРМ-123.22.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		86

9 ОСНОВНІ ВИСНОВКИ

Програмне забезпечення, створене в результаті виконання випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти, призначено для системи довідково-інформаційного сервісу комп'ютерної мережі відділу підприємства.

В межах України в недостатній мірі представлені вітчизняні розробки в цій області.

У випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти наведені теоретичне узагальнення й рішення наукового завдання дослідження методів довідково-інформаційного сервісу комп'ютерної мережі відділу підприємства.

Рішення даного завдання полягало у вирішенні наступних задач:

- Був проведений огляд існуючих систем довідково-інформаційного сервісу комп'ютерної мережі відділу підприємства.
- Досліджена система довідково-інформаційного сервісу комп'ютерної мережі відділу підприємства.
- На основі отриманих результатів досліджень створена програмна реалізація системи довідково-інформаційного сервісу комп'ютерної мережі відділу підприємства.

Розроблені під час виконання випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти алгоритми дозволяють успішно вирішувати завдання довідково-інформаційного сервісу комп'ютерної мережі відділу підприємства.

Проведено аналіз предметної галузі в ході якого були виявлені об'єкти, взаємодія яких носить істотний характер для функціональної діяльності предметної галузі, і їхні основні характеристики; побудована алгоритм і вибраний середовище розробки.

					ВКРМ-123.22.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		87

Розроблене програмне забезпечення має простий, дружній та зручний інтерфейс користувача, що забезпечує легкість у освоєнні роботи програмного продукту, зручність у використанні, і не потребує особливих спеціальних знань.

При створенні програмного забезпечення було використано об'єктно-орієнтований підхід, що відповідає сучасним тенденціям у галузі розробки комерційних програмних систем.

Програма реалізована на мові високого рівня RAD Studio Delphi 10.4. Дана мова програмування дозволяє найбільш ефективно обробляти дані. Це дозволило мінімізувати строк розробки програмного забезпечення, і, як слід, зменшити витрати на його розробку. Запропоноване програмне забезпечення ділиться на загальне програмне забезпечення, що поставляється із засобами обчислювальної техніки й спеціальне програмне забезпечення, що спеціально розроблене для даної конкретної системи й включає програми, що реалізують її функції.

Програма призначена для виконання під управлінням багатозадачної операційної системи Windows 10/11.

Даються необхідні рекомендації з установки розробленого програмного забезпечення.

Для підвищення рівня безпеки запропоновано застосовувати алгоритм Khufu.

В цілому створене програмне забезпечення підтверджує правильність використаних проектних рішень та повністю відповідає вимогам технічного завдання. Створене програмне забезпечення має потенційну можливість для подальшого вдосконалення і застосування у різних галузях.

Розроблена програма має реальний економічний ефект від її впровадження у виробництво у сумі 7083 грн. З урахуванням вартості розробки програми та обладнання, строк окуплення становить 0,6 роки.

					ВКРМ-123.22.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		88

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Борзенко В.Ю. Дослідження та програмна реалізація системи довідково-інформаційного сервісу комп'ютерної мережі відділу підприємства // Збірник праць молодих науковців ЦНТУ. – Вип. 13. – Кропивницький: ЦНТУ, 2022.

2. Kovalenko Oleksandr Qualitative risk analysis of software development / Oleksandr Kovalenko, Jamil Al-Azzeh, Oleksii Smirnov, Anna Kovalenko, Serhii Smirnov // Asian Journal of Information Technology. – Volume 17 Issue 3. – Medwell Journals. – 2018. – P. 218-230. ISSN: 1682-3915. URL: <http://medwelljournals.com/abstract/?doi=ajit.2018.218.230> Doi: ajit.2018.218.230

3. Kovalenko Oleksandr, The mathematical model of the testing technology for DOM XSS vulnerabilities / O. Kovalenko, O. Smirnov, A.Kovalenko, S. Smirnov, V. Vialkova // Scientific & practical cyber security journal (SPCSJ) Volume 2 Issue 1, P. 22-28. Georgia. Tbilisi. Scientific Cyber Security Association (SCSA), 2018 ISSN: 2587-4667. URL: <https://journal.scsa.ge/wp-content/uploads/2018/12/04-3-o.kovalenko-a.kovalenko-o.smirnov-s.smirnov-v.vialkova.pdf>

4. Коваленко А.В. Технология тестирования DOM XSS уязвимости / А.В. Коваленко, А.С. Коваленко, А.А. Смирнов, С.А. Смирнов // Scientific & practical cyber security journal (SPCSJ) Volume 1. Issue 1. P. 38-45 Georgia. Tbilisi. Scientific Cyber Security Association (SCSA), 2017 ISSN: 2587-4667. URL: <https://journal.scsa.ge/wp-content/uploads/2018/12/8-dom-xss-testing-technology-vulnerabilities.pdf>

5. Коваленко О.В. Моделі та методи розроблення програмного забезпечення комп'ютерних систем для підвищення безпеки даних: монографія / О.В. Коваленко // К.: Вид. «КОД» – 2019. – 305 с.

6. Коваленко А.В. Методы качественного анализа и количественной

					ВКРМ-123.22.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		89

оценки рисков разработки программного обеспечения / А.В. Коваленко, А.А. Смирнов // Информационные технологии в управлении, образовании, науке и промышленности: монография / Под редакцией профессора В.С. Пономаренко. – Х.: Видавець Рожко С.Г., 2016. – 566 с.

7. Коваленко А.В. Разработка метода управления рисками разработки программного обеспечения / А.В. Коваленко, А.А. Смирнов // Інформаційні технології: проблеми та перспективи: монографія / За загальною редакцією В.С. Пономаренка. – Х.: Видавець Рожко С.Г., 2017. – 447 с.

8. Коваленко А.В. Комплекс математических моделей технологии тестирования web-приложений / А.В. Коваленко, А.А. Смирнов // Інформаційні технології: сучасний стан та перспективи: монографія / За загальною редакцією В.С. Пономаренка. – Х.: ТОВ «ДІСА ПЛЮС», 2018. – 461 с.

9. Коваленко А.В. Задачи распознавания ситуаций в ERP системах / А.В. Коваленко, А.А. Смирнов, А.С. Коваленко // Збірник наукових праць "Системи обробки інформації". – Випуск 4(120). – Х.: ХУПС – 2014. – С. 161-164.

10. Коваленко А.В. Методы качественного анализа и количественной оценки рисков разработки программного обеспечения / А.В. Коваленко, А.А. Смирнов // Збірник наукових праць "Системи обробки інформації". – Випуск 5(142). – Х.: ХУПС – 2016. – С. 153-157.

11. Коваленко А.В. Проблемы анализа и оценки рисков информационной деятельности / А.В. Коваленко, А.А. Смирнов, Н.Н. Якименко, А.П. Доренский // Збірник наукових праць "Системи обробки інформації". – Випуск 3(140). – Х.: ХУПС – 2016. – С. 40-42.

12. Коваленко А.В. Метод качественного анализа рисков разработки программного обеспечения / А.В. Коваленко, А.А. Смирнов, Н.Н. Якименко, А.П. Доренский // Наука і техніка Повітряних Сил Збройних Сил України. – Випуск 2(23). – Харків: ХУПС. – 2016. – С. 150-158.

13. Коваленко А.В. Метод количественной оценки рисков разработки

					ВКРМ-123.22.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		90

программного забезпечення / А.В. Коваленко, А.А. Смирнов, Н.Н. Якименко, А.П. Доренский // Збірник наукових праць Харківського університету Повітряних Сил. Випуск 2 (47). – Харків: ХУПС. – 2016. – С. 128-133.

14. Коваленко А.В. Использование псевдобулевых методов бивалентного программирования для управления рисками разработки программного обеспечения / А.В. Коваленко, А.А. Смирнов // Системи управління, навігації та зв'язку. – Випуск 1 (37). – Полтава: ПолтНТУ. – 2016. – С. 98-103.

15. Коваленко А.В. Метод управления рисками разработки программного обеспечения / А.В. Коваленко // Системи управління, навігації та зв'язку. – Випуск 2 (38). – Полтава: ПолтНТУ. – 2016. – С. 93-100.

16. Коваленко А.В. Технология тестирования уязвимости к SQL инъекциям / А.В. Коваленко // Системи управління, навігації та зв'язку. – Випуск 5 (45). – Полтава: ПолтНТУ. – 2017. – С. 66-71.

17. Коваленко А.В. Масштабирование имитационной модели технологии тестирования безопасности / А.В. Коваленко // Системи управління, навігації та зв'язку. – Випуск 6 (46). – Полтава: ПолтНТУ. – 2017. – С. 181-184.

18. Коваленко А.В. Имитационная модель технологии тестирования безопасности Web-приложений / А.В. Коваленко // Системи управління, навігації та зв'язку. – Випуск 1 (47). – Полтава: ПолтНТУ. – 2018. – С. 114-123.

19. Коваленко О.В. Методи якісного аналізу та кількісної оцінки ризиків розроблення програмного забезпечення/ О.В. Коваленко // Системи управління, навігації та зв'язку. – Випуск 3 (49). – Полтава: ПолтНТУ. – 2018. – С. 116-125.

20. Коваленко О.В. Управління ризиками розроблення програмного забезпечення за умови обмеженості коштів виділених на усунення помилок безпеки/ О.В. Коваленко // Техніка в сільськогосподарському виробництві, галузеве машинобудування, автоматизація. – Випуск 31. – Кропивницький: ЦНТУ. – 2018. – С. 128-140.

21. Коваленко О.В. GERT-мережевий синтез технології тестування на вразливість WEB-додатків/ О.В. Коваленко // Захист інформації. – Випуск 20(2) –

					ВКРМ-123.22.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		91

К.: НАУ. – 2018. – С. 89-94.

22. Коваленко О.В. Імітаційна модель технології тестування безпеки на основі положень теорії масштабування / О.В. Коваленко // Безпека інформації. – Випуск 24 (2). – К.: НАУ. – 2018. – С. 110-117.

23. Коваленко О.В. Оцінка ефективності технології тестування безпеки / О.В. Коваленко // Вчені записки Таврійського національного університету імені В.І. Вернадського. Серія: Технічні науки. Том 29 (68) № 2, 2018. – С. 137-141

24. Коваленко О.В. Методи та засоби управління безпекою додатків / О.В. Коваленко // Інформаційно-керуючі системи на залізничному транспорті. №4, 2018. – С. 41-44.

25. Коваленко О.В. Розробка інформаційної технології передтестової компіляції та розподілу доступу / О.В. Коваленко // Системи управління, навігації та зв'язку. – Випуск 4 (50). – Полтава: ПолтНТУ. – 2018. – С. 115-119.

26. Коваленко О.В. Аналіз та дослідження інформаційних технологій розробки програмного забезпечення/ О.В. Коваленко // Вчені записки Таврійського національного університету імені В.І. Вернадського. Серія: Технічні науки. Том 29 (68) № 5, 2018. – С. 131-137.

27. Коваленко О.В. Удосконалений метод управління ризиками розроблення програмного забезпечення на основі напівмарковської моделі прийняття рішень/ О.В. Коваленко // Сучасні інформаційні системи. – Випуск 2(3). – Харків. – 2018. – С. 41-48.

28. Коваленко О.В. Математичні моделі технології тестування DOM XSS вразливості та вразливості до SQL ін'єкцій / О.В. Коваленко // Вісник Черкаського державного технологічного університету. Серія : Технічні науки №4, 2018. – С. 29-36.

29. Коваленко О.В. Математична модель технології тестування вразливості до SQL ін'єкцій / О.В. Коваленко // Системи управління, навігації та зв'язку. – Випуск 6 (58). – Полтава: ПолтНТУ. – 2019. – С. 43-47.

					ВКРМ-123.22.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		92

25 березня 2016 р. – Кіровоград: КНТУ. – 2016. – С. 71-72.

36. Коваленко А.В. Методы качественного анализа рисков разработки программного обеспечения / А.В. Коваленко, А.А. Смирнов // Збірник тез першої міжнародної науково-практичної конференції «Проблеми науково-технічного та правового забезпечення кібербезпеки у сучасному світі» (ПНПЗК-2016). м. Харків. 30 березня – 1 квітня 2016 р. – Харків: НТУ «ХПІ». – 2016. – С. 6-7.

37. Коваленко А.В. Структурная идентификация рисков разработки программного обеспечения / А.В. Коваленко, А.А. Смирнов // Збірник тез XVIII міжнародного науково-практичного семінару «Комбінаторні конфігурації та їх застосування». м. Кіровоград. 15-16 квітня 2016 р. – Кіровоград: КНТУ. – 2016. – С. 175-182.

38. Коваленко А.В. Исследование разработанной методики структурной идентификации рисков разработки программного обеспечения / А.В. Коваленко, А.А. Смирнов // Збірник тез VIII міжнародної науково-практичної конференції “Проблеми і перспективи розвитку ІТ-індустрії”. м. Харків. 28-29 квітня 2016 р. – Харків: ХНЕУ. – 2016. – С. 49.

39. Коваленко А.В. Исследование дерева рисков разработки программного обеспечения / А.В. Коваленко, А.А. Смирнов // Збірник тез III міжнародної науково-практичної конференції «Інформаційна та економічна безпека» (INFECO-2016)». м. Харків. 28-30 квітня 2016 р. – Харків: ХННІ ДВНЗ «УБС». – 2016. – С. 174-178.

40. Коваленко А.В. Методы качественного анализа и количественной оценки рисков разработки программного обеспечения / А.В. Коваленко, А.А. Смирнов, А.С. Коваленко // Сборник тезисов XII международной конференции "Стратегия качества в промышленности и образовании". г. Варна. Болгария. 30 мая – 02 июня 2016 г – Варна. ТУВ. – 2016. – С. 585-589.

41. Коваленко А.В. Разработка метода управления рисками разработки программного обеспечения / А.В. Коваленко, А.А. Смирнов,

					ВКРМ-123.22.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		94

А.С. Коваленко // Матеріали Всеукраїнської науково-практичної конференції «Кібербезпека в Україні: правові та організаційні питання». м. Одеса, 21 жовтня 2016 р. – Одеса : ОДУВС, 2016. – С.146-148.

42. Коваленко А.В. Метод управления рисками разработки программного обеспечения с использованием псевдобулевых методов бивалентного программирования / А.В. Коваленко, А.А. Смирнов // Матеріали Всеукраїнської науково-практичної конференції «Актуальні задачі та досягнення у галузі кібербезпеки». м. Кропивницький, 23-25 листопада 2016 року – Кропивницький: ЦНТУ, 2016. – С. 162.

43. Коваленко А.В. Псевдобулевые методы бивалентного программирования для управления рисками разработки программного обеспечения / А.В. Коваленко, А.А. Смирнов, А.С. Коваленко, С.А. Смирнов // Збірник наукових праць III міжнародної науково-практичної конференції «Актуальні питання забезпечення кібернетичної безпеки та захисту інформації». м. Київ. 22-25 лютого 2017 р. – Київ: Європейський університет. – 2017. – С. 158-162.

44. Коваленко А.В. Метод управления рисками разработки программного обеспечения / А.В. Коваленко, А.А. Смирнов // Збірник тез II науково-практичної конференції “Проблеми кібербезпеки інформаційно-телекомунікаційних систем”. м. Київ. 23-24 березня 2017 р. – Київ: КНУ ім. Тараса Шевченка – 2017. – С. 203-205.

45. Коваленко А.В. Алгоритмы анализа уязвимостей при управлении рисками разработки программного обеспечения / А.В. Коваленко, А.А. Смирнов, А.С. Коваленко // Conferenta internationala (editia a XIII-a). «Securitatea informationala 2017». Chisinau. Republic of Moldova. 4-5 aprilie 2017. – Chisinau: ADSEM. – 2017. – P. 19-22.

46. Коваленко А.В. Алгоритм анализа DOM XSS уязвимости при управлении рисками разработки программного обеспечения / А.В. Коваленко, А.А. Смирнов, А.С. Коваленко // Збірник тез дев'ятнадцятого міжнародного

					ВКРМ-123.22.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		95

науково-практичного семінару «Комбінаторні конфігурації та їх застосування». м. Кропивницький 7-8 квітня 2017 р. – Кропивницький: ГЛА НАУ. – 2017. – С. 125-127.

47. Коваленко А.В. Алгоритм аналізу уязвимості SQL Injection для управління ризиками розробки програмного забезпечення / А.В. Коваленко, А.А. Смирнов, А.С. Коваленко // Збірник тез другої міжнародної науково-технічної конференції «Проблеми науково-технічного та правового забезпечення кібербезпеки у сучасному світі» (ПНПЗК-2017). м. Харків. 10-12 квітня 2017 р. – Харків: НТУ «ХПІ». – 2017. – С. 27.

48. Коваленко А.В. Метод управління ризиками розробки програмного забезпечення на основі алгоритмів аналізу уязвимостей / А.В. Коваленко, А.А. Смирнов, А.С. Коваленко // Збірник тез Міжнародної науково-практичної конференції «Інформаційна безпека та комп'ютерні технології» (IS&CT). м. Кіровоград. 20-22 квітня 2017 р. Кіровоград: КНТУ. – 2017. – С. 92.

49. Коваленко А.В. Алгоритми аналізу DOM XSS уязвимості та уязвимості SQL Injection при управлінні ризиками розробки програмного забезпечення / А.В. Коваленко, А.А. Смирнов, А.С. Коваленко // Збірник тез ІХ міжнародної науково-практичної конференції “Проблеми і перспективи розвитку ІТ-індустрії”. м. Харків. 20-21 квітня 2017 р. – Харків: ХНЕУ. – 2017. – С. 61.

50. Kovalenko O.V. Method of testing the dom xss vulnerability / Kovalenko Oleksandr, Kovalenko Anna, Smirnov Oleksii, Smirnov Serhii // International Conference «information technologies, systems and networks ITSН-2017». Chisinau, Republic of Moldova. 17 – 18 October 2017. – Chisinau: Academy of Sciences of Moldova, Military Academy of Armed Forces “Alexandru cel Bun”. – 2017. – P. 7.

51. Коваленко О.В. Метод тестування DOM XSS уразливості / О.В. Коваленко, О.А. Смирнов, А.С. Коваленко, С.А. Смирнов // Збірник тез всеукраїнської науково-практичної інтернет-

					ВКРМ-123.22.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		96

конференції «Автоматика та комп'ютерно-інтегровані технології у промисловості, телекомунікаціях, енергетиці та транспорті». м. Кропивницький. 16-17 листопада 2017 р. – Кропивницький: ЦНТУ. – 2017. – С. 198-199.

52. Коваленко О.В. GERT-модель технології тестування DOM XSS уразливості / О.В. Коваленко, А.С. Коваленко, О.А. Смірнов, С.А. Смірнов // Збірник наукових праць IV міжнародної науково-практичної конференції «Актуальні питання забезпечення кібернетичної безпеки та захисту інформації». м. Київ. 21-24 лютого 2018 р. – Київ: Європейський університет. – 2018. – С. 65-70.

53. Коваленко О.В. Технології тестування уразливостей Web-застосунків з використанням GERT-моделі / О.В. Коваленко, А.С. Коваленко, О.А. Смірнов, С.А. Смірнов // Збірник тез всеукраїнської науково-практичної конференції "Комп'ютерні інтелектуальні системи та мережі (КІСМ-2018)". м. Кривий Ріг. 21-23 березня 2018 р. – Кривий Ріг.: ДВНЗ КНУ – 2018. – С. 227-230.

54. Коваленко А.В. Тестирование уязвимости Web-приложений к атаке вида межсайтовый скриптинг / А.В. Коваленко, А.С. Коваленко, А.А. Смирнов, С.А. Смирнов // Збірник тез «Securitea internationala 2018». Conferenta internationala (editia a XIV-a). Chisinau. Moldova. 20-21 martie 2018. – Chisinau: ADSEM. – 2018. – Р. 54-56.

55. Коваленко А.В. Комплекс математических моделей технологии тестирования web-приложений / А.В. Коваленко, А.С. Коваленко, А.А. Смирнов, С.А. Смирнов // Збірник тез X міжнародної науково-практичної конференції “Проблеми і перспективи розвитку ІТ-індустрії”. м. Харків. 19-20 квітня 2018 р. – Харків: ХНЕУ. – 2018. – С. 38.

56. Державні будівельні норми України: ДБН В.2.5-28:2018. – Режим доступу до ресурсу: <https://goo.su/9AkQ>

57. Державні санітарні правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин: ДСанПІН 3.3.2-

					ВКРМ-123.22.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		97

Додаток А
(обов'язковий)

Технічне завдання

Зміст

1 Найменування та область застосування.....	2
2 Підстава для розробки.....	2
3 Мета та призначення розробки.....	2
4 Джерела розробки.....	2
5 Технічні вимоги.....	2
5.1 Вміст проекту.....	2
5.2 Показники призначення.....	3
5.3 Вимоги до функціональних характеристик.....	3
5.4 Вимоги до архітектури.....	3
5.5 Вимоги до надійності.....	3
5.6 Умови експлуатації.....	4
5.7 Вимоги до складу та параметрів технічних засобів.....	4
5.8 Вимоги до інформаційної і програмної сумісності.....	4
5.8.1 Обладнання.....	4
5.8.2 Мова програмування.....	4
5.8.3 Вхідні дані.....	5
5.8.4 Вихідні дані.....	5
6 Вимоги до програмної документації.....	5
7 Економічні вимоги.....	5
8 Вимоги щодо охорони праці.....	5
9 Перелік документів, що розробляються.....	6
10 Етапи розробки.....	6
11 Порядок контролю та приймання.....	6

					ВКРМ-123.22.0003.00.00.ТЗ		
Вим.	Арк.	№ документа	Підпис	Дата			
Розробив	Борзенко В.Ю.				Літ.	Аркуш	Аркушів
Перевірів	Буравченко К.О.			М			
Н. Контр.	Гермак В.С.				ЦНТУ КІ-21М-1,4		
Затв.	Смірнов О.А.						

1 Найменування та область застосування

Це технічне завдання розповсюджується на дослідження та програмну реалізацію системи довідково-інформаційного сервісу комп'ютерної мережі відділу підприємства.

2 Підстава для розробки

Підставою для розробки служить завдання на випускну кваліфікаційну роботу за другим (магістерським) рівнем вищої освіти, видане на кафедрі кібербезпеки та програмного забезпечення (нак. № 19-13 від 17.08.2022 року).

3 Мета та призначення розробки

Метою випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти є дослідження та програмна реалізація системи довідково-інформаційного сервісу комп'ютерної мережі відділу підприємства.

4 Джерела розробки

Джерелом цієї випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти є стосовна до теми література і існуючі аналоги.

5 Технічні вимоги

5.1 Склад продукції

Складниками розробки є:

- вибір і обґрунтування методів реалізації проекту;
- розробка програмної частин системи, а також розробка взаємодії системи з ОС та з користувачем;

					ВКРМ-123.22.0003.00.00.ТЗ	Арк.
Вим.	Арк.	№ документа	Підпис	Дата		2

- техніко-економічне обґрунтування доцільності прийнятого до розробки програмного забезпечення;
- аналіз умов праці;
- розробка програми, що реалізує спроектовані алгоритми роботи системи.

5.2 Показники призначення

Система повинна забезпечувати:

- програмну реалізацію системи довідково-інформаційного сервісу комп'ютерної мережі відділу підприємства;
- цілісність даних у процесі роботи та при зберіганні;
- простий, інтуїтивно зрозумілий інтерфейс.

5.3 Вимоги до функціональних характеристик

Розроблене програмне забезпечення не повинно мати обмежень на версію драйверів та операційної системи.

5.4 Вимоги до архітектури

Компонент, що розробляється повинен використовувати системні засоби та апаратні засоби, що на даному етапі розвитку обчислювальної техніки найбільше поширені.

5.5 Вимоги до надійності

Програмні модулі написані по всім правилам, які стосуються стандартних викликів процедур, функцій, методів і форм, визначених технічною документацією на середовище розробки.

					ВКРМ-123.22.0003.00.00.ТЗ	Арк.
Вим.	Арк.	№ документа	Підпис	Дата		3

5.6 Умови експлуатації

Робочі місця користувачів ПЗ повинні задовольняти наступним умовам експлуатації:

- температура повітря: 19-20 град. по Цельсію;
- відносна вологість повітря до 80%;
- атмосферний тиск 107 кПа.

5.7 Вимоги до складу та параметрів технічних засобів

Програмне забезпечення повинно бути реалізоване на ПЕОМ архітектури IBM PC, працювати в ОС Windows 10/11 і з сумісними з цією платформою пристроями і прикладним програмним забезпеченням.

5.8 Вимоги до інформаційної і програмної сумісності

Переносність програмного забезпечення повинна бути забезпечена за рахунок його реалізації стандартного інтерфейсу взаємодії з ОС, що працюють під управлінням ОС Windows 10/11.

5.8.1 Обладнання

Комп'ютер Intel® Celeron/8 Mb/1.2 Gb/SVGA 14" 1Mb або сумісні з ним.

5.8.2 Мова програмування

Середовище RAD Studio Delphi 10.4.

					ВКРМ-123.22.0003.00.00.ТЗ	Арк.
Вим.	Арк.	№ документа	Підпис	Дата		2

5.8.3 Вхідні дані

Опис алгоритму роботи запропонованої системи.

5.8.4 Вихідні дані

Робоча програма.

6 Вимоги до програмної документації

Програмна продукція повинна бути представлена у виді опису структури даних, схем та опису алгоритму, а також текстів вихідних модулів програмного забезпечення згідно ЄСПД .

7 Економічні вимоги

7.1 Для ПЗ необхідно виробити функціонально-вартісний аналіз варіантів розробки.

7.2 Виконати розрахунок витрат показників економічного ефекту з урахуванням цін на 3 вересня 2022 року.

8 Вимоги щодо охорони праці

В частині охорони праці випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти повинен бути розглянутий аналіз умов праці на робочому місці ІТ-фахівця.

					ВКРМ-123.22.0003.00.00.ТЗ	Арк.
Вим.	Арк.	№ документа	Підпис	Дата		5

9 Перелік документів, що розробляються

- Наукова новизна – 1 аркуш.
- Структурна схема системи – 1 аркуш.
- Функціональна схема системи – 1 аркуш.
- Діаграма процесів – 1 аркуш.
- Блок-схема алгоритму роботи програми – 2 аркуша.
- Показники економічної ефективності – 1 аркуш.
- Пояснювальна записка – 98 аркушів.

10 Етапи розробки

10.1 Збір і обробка інформації по темі випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти. Постановка задачі на виконання випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти (складання ТЗ).

10.2 Проведення досліджень або експериментальних робіт для уточнення основних положень випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти.

10.3 Розробка функціональних схем, блок схем алгоритмів роботи програмного забезпечення.

10.4 Побудова схем взаємодії даних.

10.5 Створення прототипу ПЗ.

10.6 Віднаходження ПЗ, аналіз отриманих результатів.

10.7 Робота над питанням охорони праці і техніки безпеки.

10.8 Розрахунок з техніко-економічного обґрунтування.

10.9 Оформлення пояснювальної записки і виконання робіт по графічній частині.

11 Порядок контролю та приймання

11.1 Подання випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти на попередній захист 10.12.2022 р.

11.2 Подання випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти на захист 19.12.2022 р.

					ВКРМ-123.22.0003.00.00.ТЗ	Арк.
Вим.	Арк.	№ документа	Підпис	Дата		6

Додаток Б
(обов'язковий)

Міністерство освіти і науки України
Центральноукраїнський національний технічний університет

ЗАТВЕРДЖУЮ

Керівник випускної кваліфікаційної роботи за
другим (магістерським) рівнем вищої освіти

_____ Буравченко К.О.

*Дослідження та програмна реалізація
системи довідково-інформаційного сервісу комп'ютерної мережі відділу
підприємства*

Лістинг програми

Код документу 12

Носій: CD/DVD-диск / USB-флеш-накопичувач

Загальна кількість аркушів: 28

Літера: РП

Кропивницький – 2022 року

Файл Manager_Base.dpr - головний файл проекту

```
program Manager_Base;

uses
  Forms,
  paswd in 'avt.pas' {Form2},
  adsbaz in 'adsbaz.pas' {Form3},
  about in 'about.pas' {Form4},
  usprint in 'usprint.pas' {Form1};

{$R *.res}

begin
  Application.Initialize;
  Application.CreateForm(TForm2, Form2);
  Application.CreateForm(TForm3, Form3);
  Application.CreateForm(TForm4, Form4);
  Application.CreateForm(TForm1, Form1);
  Application.Run;
end.
```

Кафедра _ КБПЗ _ 2022 рік

Файл adsbaz.pas - основна програма

```
unit adsbaz;

interface

uses
  Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
  StdCtrls, Buttons, ComCtrls, ExtCtrls, Grids, Printers;

type
  TForm3 = class(TForm)
    Label1: TLabel;
    Label2: TLabel;
    Label3: TLabel;
    Edit1: TEdit;
    Edit2: TEdit;
    Edit3: TEdit;
    Previous: TButton;
    Next: TButton;
    New: TButton;
    Delete: TButton;
    Save: TButton;
    Button1: TButton;
    Label5: TLabel;
    RichEdit1: TRichEdit;
    Button2: TButton;
    Bevel1: TBevel;
    Label4: TLabel;
    Button4: TButton;
    Label8: TLabel;
    Label9: TLabel;
    Label10: TLabel;
    Label11: TLabel;
    Label12: TLabel;
    Edit6: TEdit;
    ComboBox2: TComboBox;
    ComboBox3: TComboBox;
    ComboBox4: TComboBox;
    ComboBox5: TComboBox;
    Bevel2: TBevel;
    Button6: TButton;
    Button7: TButton;
    Button8: TButton;
    Edit7: TEdit;
    Bevel3: TBevel;
    Bevel5: TBevel;
    Bevel6: TBevel;
    Bevel4: TBevel;
    Label6: TLabel;
    Bevel7: TBevel;
    procedure FormCreate(Sender: TObject);
    procedure NewClick(Sender: TObject);
    procedure SaveClick(Sender: TObject);
    procedure PreviousClick(Sender: TObject);
    procedure NextClick(Sender: TObject);
    procedure DeleteClick(Sender: TObject);
    procedure Button1Click(Sender: TObject);
    procedure Button2Click(Sender: TObject);
    procedure FormClose(Sender: TObject; var Action: TCloseAction);
    procedure Button4Click(Sender: TObject);
    procedure Button7Click(Sender: TObject);
    procedure Button6Click(Sender: TObject);
    procedure Button8Click(Sender: TObject);
    procedure Edit7KeyPress(Sender: TObject; var Key: Char);
  private
    { Private declarations }
  public
```

```

    { Public declarations }
end;

var
    Form3: TForm3;

implementation

{$R *.DFM}

uses usprint, paswd;

type
    TNote=record
        org: string[200];
        adr: string[255];
        tel: string[150];
        kont1: string[150];
        email: string[100];
        reg: string[200];
        vidd: string[200];
        osnprod: string[200];
        meneg: string[150];
    end;

var
    NoteFile: file of TNote;
    NoteData: TNote;
    _Pos: integer;

procedure ShowRecord;
//створення форм для введення записів до БД
begin
    Form3.Edit1.Text:=NoteData.org;
    Form3.RichEdit1.Text:=NoteData.adr;
    Form3.Edit2.Text:=NoteData.tel;
    Form3.Edit3.Text:=NoteData.kont1;
    Form3.Edit6.Text:=NoteData.email;
    Form3.ComboBox2.Text:=Notedata.reg;
    Form3.ComboBox3.Text:=Notedata.vidd;
    Form3.ComboBox4.Text:=Notedata.osnprod;
    Form3.ComboBox5.Text:=Notedata.meneg;
end;

procedure ClearData;
//очищення даних у БД
begin
    Form3.Edit1.Text:='';
    Form3.RichEdit1.Text:='';
    Form3.Edit2.Text:='';
    Form3.Edit3.Text:='';
    Form3.Edit6.Text:='';
    Form3.ComboBox2.Text:='';
    Form3.ComboBox3.Text:='';
    Form3.ComboBox4.Text:='';
    Form3.ComboBox5.Text:='';
end;

// Pos починається з нуля - 0,1,2,3,4,5,6,...
// FileSize починається з одиниці - 1,2,3,4,5,6,7,8,...

function GetProgramPath : String;
//встановлення шляху до БД
begin
    GetProgramPath:=ExtractFilePath(ParamStr(0));
end;

procedure TForm3.FormCreate(Sender: TObject);
var

```



```

Seek(NoteFile,cPos);
Read(NoteFile,NoteData); // читання даних з БД
s:=NoteData.vidd;
for i:=0 to FileSize(NoteFile)-1 do
begin
  if s<>' ' then
  begin
    ComboBox3.Items.Add(s);
    break;
  end;
  inc(cPos);
  Seek(NoteFile,cPos);
  Read(NoteFile,NoteData); // читання даних з БД
  s:=NoteData.vidd;
end;
for cPos:=0 to FileSize(NoteFile)-1 do
begin
  Seek(NoteFile,cPos);
  Read(NoteFile,NoteData); // читання даних з БД
  s:=NoteData.vidd;
  if s<>' ' then
  begin
    for i:=0 to ComboBox3.Items.Count-1 do
    if pos(s,ComboBox3.Items.Strings[i])=0
    then j:=2
    else
    begin
      j:=1;
      Break;
    end;
  end;
  if j=2 then ComboBox3.Items.Add(s);
  j:=0;
end;
ComboBox3.Sorted:=true;
////////// Основна продукція
cPos:=0;
Seek(NoteFile,cPos);
Read(NoteFile,NoteData); // читання даних з БД
s:=NoteData.osnprod;
for i:=0 to FileSize(NoteFile)-1 do
begin
  if s<>' ' then
  begin
    ComboBox4.Items.Add(s);
    break;
  end;
  inc(cPos);
  Seek(NoteFile,cPos);
  Read(NoteFile,NoteData); // читання даних з БД
  s:=NoteData.osnprod;
end;
for cPos:=0 to FileSize(NoteFile)-1 do
begin
  Seek(NoteFile,cPos);
  Read(NoteFile,NoteData); // читання даних з БД
  s:=NoteData.osnprod;
  if s<>' ' then
  begin
    for i:=0 to ComboBox4.Items.Count-1 do
    if pos(s,ComboBox4.Items.Strings[i])=0
    then j:=2
    else
    begin
      j:=1;
      Break;
    end;
  end;
  if j=2 then ComboBox4.Items.Add(s);

```

```

    j:=0;
    end;
    ComboBox4.Sorted:=true;
    //////////// Менеджер
    cPos:=0;
    Seek(NoteFile,cPos);
    Read(NoteFile,NoteData); // читання даних з БД
    s:=NoteData.meneg;
    for i:=0 to FileSize(NoteFile)-1 do
    begin
        if s<>' ' then
        begin
            ComboBox5.Items.Add(s);
            break;
        end;
        inc(cPos);
        Seek(NoteFile,cPos);
        Read(NoteFile,NoteData); // читання даних з БД
        s:=NoteData.meneg;
    end;
    for cPos:=0 to FileSize(NoteFile)-1 do
    begin
        Seek(NoteFile,cPos);
        Read(NoteFile,NoteData); // читання даних з БД
        s:=NoteData.meneg;
        if s<>' ' then
        begin
            for i:=0 to ComboBox5.Items.Count-1 do
            if pos(s,ComboBox5.Items.Strings[i])=0
            then j:=2
            else
            begin
                j:=1;
                Break;
            end;
            end;
            if j=2 then ComboBox5.Items.Add(s);
            j:=0;
        end;
        ComboBox5.Sorted:=true;
        //////////////////////////////////////
        pPos:=FileSize(NoteFile); // визначення кіл-ті компонентів файлу з БД
        end;
        Label4.Caption:='Запис № '+IntToStr(_pos+1)+' з '+IntToStr(pPos);
    end;

    procedure TForm3.NewClick(Sender: TObject);
    var
        pPos: integer;
    begin
        ClearData; //очищення вікна
        Edit1.SetFocus;
        _Pos:=FileSize(NoteFile); //визначення кіл-ті компонентів файлу з БД
        Seek(NoteFile,_Pos); //переміщення покажчика на кінець файлу з БД
        Button2.Enabled:=false;
        Previous.Enabled:=false;
        Next.Enabled:=false;
        Delete.Enabled:=false;
        Save.Enabled:=true;
        //////////////////////////////////////
        pPos:=FileSize(NoteFile); //визначення кіл-ті компонентів файлу з БД
        end;
        Label4.Caption:='Запис № '+IntToStr(_pos+1)+' з '+IntToStr(pPos+1);
    end;

    procedure TForm3.SaveClick(Sender: TObject);
    begin
        NoteData.org:=Edit1.Text;
        NoteData.adr:=RichEdit1.Text;

```

```

NoteData.tel:=Edit2.Text;
NoteData.kont1:=Edit3.Text;
NoteData.email:=Edit6.Text;
Notedata.reg:=ComboBox2.Text;
Notedata.vidd:=ComboBox3.Text;
Notedata.osnprod:=ComboBox4.Text;
Notedata.meneg:=ComboBox5.Text;
Write(NoteFile,NoteData); // запис у файл
Button2.Enabled:=true;
Next.Enabled:=false;
Save.Enabled:=false;
Delete.Enabled:=true;
if FileSize(NoteFile)>=2 then
Previous.Enabled:=true;
end;

procedure TForm3.PreviousClick(Sender: TObject);
var
  pPos: integer;
begin
  Next.Enabled:=true;
  if _Pos<>0 then dec(_Pos); // якщо покажчик не дорівнює 0, то Pos-1
  Seek(NoteFile,_Pos); // переміщення на Pos-1
  Read(NoteFile,NoteData); // читання Pos-1
  ShowRecord; // показати запис
  if _Pos=0 then Previous.Enabled:=false; //якщо номер-0, то...
  //////////////////////////////////////
  pPos:=FileSize(NoteFile); // визначення кіл-ті компонентів файлу з БД   end;
  Label4.Caption:='Запис № '+IntToStr(_Pos+1)+' з '+IntToStr(pPos);
end;

procedure TForm3.NextClick(Sender: TObject);
var
  pPos:integer;
begin
  Previous.Enabled:=true;
  if _Pos=FileSize(NoteFile)-2 // якщо номер позиції= кількості
  then Next.Enabled:=false; // номерів, то...
  inc(_Pos); // Pos+1
  Seek(NoteFile,_Pos); // переміщення покажчика на Pos+1
  Read(NoteFile,NoteData); // читання даних з БД
  ShowRecord; // показати запис
  //////////////////////////////////////
  pPos:=FileSize(NoteFile); // визначення кіл-ті компонентів файлу з БД
  end;
  Label4.Caption:='Запис № '+IntToStr(_Pos+1)+' з '+IntToStr(pPos);
end;

procedure TForm3.DeleteClick(Sender: TObject);
var
  i:integer;
  pPos: integer;
begin
  if MessageDlg('Ви дійсно хочете видалити запис?', mtWarning,
  [mbYes, mbNo],0)=mrYes
  then
  begin
  begin
  if _Pos=FileSize(NoteFile)-1 then // якщо елемент що видаляється останній
  begin
  Seek(NoteFile,_Pos);
  Truncate(NoteFile);
  Next.Enabled:=false;
  if _Pos<>0
  then dec(_Pos) // якщо є дані, то Pos-1
  else
  begin
  Delete.Enabled:=false;
  ClearData;
  exit;
  end;
  end;
  end;
  end;
  end;

```

```

        end;
    end
else
begin
    for i:=_Pos+1 to FileSize(NoteFile)-1 do
    begin
        Seek(NoteFile,i); // перехід на Pos+1
        Read(NoteFile,NoteData); // читання
        Seek(NoteFile, i-1); // перехід на Pos
        Write(NoteFile,Notedata); // запис Pos+1 в Pos
    end;
    Seek(NoteFile, FileSize (NoteFile) -1);
    Truncate (NoteFile);
    end;
    Seek (NoteFile, _Pos);
    Read (NoteFile, NoteData);
    ShowRecord;
    if _Pos=0 then Previous.Enabled:=false;
    //////////////////////////////////////
    pPos:=FileSize (NoteFile); // визначення кіл-ті компонентів файлу з БД
end;
    Label4.Caption:='Запис № ' +IntToStr (_Pos+1)+' з '+IntToStr (pPos);
    end
    else
end;

procedure TForm3.Button2Click(Sender: TObject);
begin
    NoteData.org:=Edit1.Text;
    NoteData.adr:=RichEdit1.Text;
    NoteData.tel:=Edit2.Text;
    NoteData.kont1:=Edit3.Text;
    NoteData.email:=Edit6.Text;
    Notedata.reg:=ComboBox2.Text;
    Notedata.vidd:=ComboBox3.Text;
    Notedata.osnprod:=ComboBox4.Text;
    Notedata.meneg:=ComboBox5.Text;
    if _Pos=0
    then _Pos:=0
    else _Pos:=FilePos (NoteFile) -1; // визначення розташування покажчика
    Seek (NoteFile, _Pos);
    Write (NoteFile, NoteData); // запис у файл
end;

procedure TForm3.Button1Click(Sender: TObject);
begin
    Close;
end;

procedure TForm3.FormClose(Sender: TObject; var Action: TCloseAction);
begin
    try
        CloseFile (NoteFile);
    except
        MessageDlg ('Помилка закриття файлу з БД!', mtError, [mbOK], 0);
    end;
    Form2.Close;
end;

procedure TForm3.Button4Click(Sender: TObject);
begin
    Form1.Show;
end;

procedure TForm3.Button7Click(Sender: TObject);
var
    pPos, p: integer;
begin
    ClearData; // очищення вікна

```

```

Edit1.SetFocus;
_Pos:=0; // визначення кіл-ті компонентів файлу з БД
Seek(NoteFile,_Pos); // переміщення покажчика на кінець файлу з БД
Read(NoteFile,NoteData); // читання Pos-1
ShowRecord; // показати запис
Previous.Enabled:=false;
Next.Enabled:=true;
Delete.Enabled:=true;
Save.Enabled:=false;
////////////////////////////////////
pPos:=FileSize(NoteFile)-1; // визначення кіл-ті компонентів файлу з БД
Label4.Caption:='Запис № '+IntToStr(_Pos+1)+' з '+IntToStr(pPos+1);
end;

procedure TForm3.Button6Click(Sender: TObject);
var
  pPos: integer;
begin
  ClearData; // очищення вікна
  Edit1.SetFocus;
  _Pos:=FileSize(NoteFile)-1;
  Seek(NoteFile,_Pos); // переміщення покажчика на кінець файлу з БД
  Read(NoteFile,NoteData);
  ShowRecord;
  Previous.Enabled:=true;
  Next.Enabled:=false;
  Delete.Enabled:=true;
  Save.Enabled:=false;
  //////////////////////////////////////
  pPos:=FileSize(NoteFile); // визначення кіл-ті компонентів файлу з БД
  Label4.Caption:='Запис № '+IntToStr(_Pos+1)+' з '+IntToStr(pPos);
end;

procedure TForm3.Button8Click(Sender: TObject);
var
  pPos: integer;
begin
  pPos:=FileSize(NoteFile);
  if StrToInt(Edit7.Text)>pPos
  then
    begin
      MessageDlg('Введене число ('+Edit7.Text+') більше, ніж кількість записів
('+IntToStr(pPos)+') !',mtError,[mbOK],0);
      Edit7.Text:='1';
      Edit7.SetFocus;
      Edit7.SelStart:=0;
      Edit7.SelLength:=1;
      Exit;
    end;
  if StrToInt(Edit7.Text)=0 then
    begin
      MessageDlg('Введеное число ('+Edit7.Text+') менше 1!',mtError,[mbOK],0);
      Edit7.Text:='1';
      Edit7.SetFocus;
      Edit7.SelStart:=0;
      Edit7.SelLength:=1;
      Exit;
    end;
  ClearData; // очищення вікна
  Edit1.SetFocus;
  _Pos:=StrToInt(Edit7.Text);
  Seek(NoteFile,_Pos-1);
  Read(NoteFile,NoteData);
  ShowRecord; // показати запис
  if _Pos-1=FileSize(NoteFile)-1 // якщо номер позиції= кількості
  then // номерів, то...
    begin
      Next.Enabled:=false;
      Previous.Enabled:=true;
    end;
end;

```

```

end;
if _ Pos-1=0                                // якщо номер позиції= кількості
then                                         // номерів, то...
begin
  Next.Enabled:=true;
  Previous.Enabled:=false;
end;
if (_ Pos-1>0) and (_ Pos-1<FileSize(NoteFile)-1)
then
begin
  Next.Enabled:=true;
  Previous.Enabled:=true;
end;
Delete.Enabled:=true;
Save.Enabled:=false;
Button2.Enabled:=true;
pPos:=FileSize(NoteFile);
dec(_ Pos);
////////////////////////////////////
pPos:=FileSize(NoteFile); // визначення кіл-ті компонентів файлу з БД
Label4.Caption:='Запис № '+IntToStr(_ Pos+1)+' з '+IntToStr(pPos);
end;

procedure TForm3.Edit7KeyPress(Sender: TObject; var Key: Char);
begin
  if Key=#13 then
  begin
    Edit7.Text:='1';
    Edit7.SelStart:=0;
    Edit7.SelLength:=1;
  end;
  if (Key<'0') or (Key>'9') then Key:=#0;
  if Length(Edit7.Text)>=4 then Key:=#0;
end;

end.

```

Кафедра _ КБІЗ _ 2022 рік

Файл usprint.pas - формування звітів

```
unit usprint;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, Grids, StdCtrls, Printers, ComCtrls, DBGrids, Clipbrd,
  OleServer, EXCEL97, EXCEL2000, ComObj, Active, ExtCtrls, ExcelXP;

type
  TForm1 = class(TForm)
    Button1: TButton;
    Button2: TButton;
    StringGrid1: TStringGrid;
    Button5: TButton;
    Button6: TButton;
    SaveDialog1: TSaveDialog;
    Button7: TButton;
    Button10: TButton;
    Button11: TButton;
    XLApp: TExcelApplication;
    Panel1: TPanel;
    Label1: TLabel;
    Bevel1: TBevel;
    Bevel3: TBevel;
    Bevel2: TBevel;
    RichEdit1: TRichEdit;
    RichEdit2: TRichEdit;
    Button3: TButton;
    ComboBox1: TComboBox;
    Button8: TButton;
    Button9: TButton;
    ComboBox2: TComboBox;
    Button4: TButton;
    Button12: TButton;
    procedure Button1Click(Sender: TObject);
    procedure Button2Click(Sender: TObject);
    procedure FormCreate(Sender: TObject);
    procedure Button3Click(Sender: TObject);
    procedure Button5Click(Sender: TObject);
    procedure Button6Click(Sender: TObject);
    procedure Button8Click(Sender: TObject);
    procedure Button7Click(Sender: TObject);
    procedure Button9Click(Sender: TObject);
    procedure Button10Click(Sender: TObject);
    procedure Button11Click(Sender: TObject);
    procedure ComboBox1Change(Sender: TObject);
    procedure Button4Click(Sender: TObject);
    procedure Button12Click(Sender: TObject);
    procedure StringGrid1Click(Sender: TObject);
    procedure StringGrid1DbClick(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  Form1: TForm1;

implementation

uses adsbaz;

{$R *.dfm}
```



```

x:=a;
Delete(x,1,1);
if (StrToInt(x)<10)
then a:=a+' ';
if (StrToInt(x)>=10) and (StrToInt(x)<100)
then a:=a+' ';
////////////////////////////////////
RichEdit2.Lines.Add('| '+a+' | '+b+' | '+c+' | '+d+' | '+e+'
|');
end;
RichEdit2.Lines.Add('');-----i-|');
StringGrid1.Row:=1;
end;

procedure TForm1.FormCreate(Sender: TObject);
begin
AssignFile(NoteFile,GetProgramPath+'data\'+'adsbaz.dat'); // зв'язуємо змінну з файлом
Reset(NoteFile); // відкриття файлу з БД
_Pos:=0;
if not Eof(NoteFile) then // якщо у файлі є дані
begin
Read(NoteFile,NoteData); // читання даних з БД
_Pos:=FilePos(NoteFile)-1; // визначення розташування покажчика
ShowRecord; // показати дані
end;
StringGrid1.ColWidths[0]:=10;
StringGrid1.ColWidths[1]:=50;
StringGrid1.ColWidths[2]:=100;
StringGrid1.ColWidths[3]:=100;
StringGrid1.ColWidths[4]:=98;
StringGrid1.ColWidths[5]:=100;
StringGrid1.ColWidths[6]:=95;
StringGrid1.ColWidths[7]:=95;
StringGrid1.ColWidths[8]:=100;
StringGrid1.ColWidths[9]:=100;
StringGrid1.ColWidths[10]:=100;
StringGrid1.RowHeights[0]:=20;
StringGrid1.RowCount:=2;
StringGrid1.Cells[1,0]:='№';
StringGrid1.Cells[2,0]:=' Організація';
StringGrid1.Cells[3,0]:='Адреса';
StringGrid1.Cells[4,0]:='Телефон';
StringGrid1.Cells[5,0]:='Контактна особа';
StringGrid1.Cells[6,0]:=' Е-mail';
StringGrid1.Cells[7,0]:='Регіон';
StringGrid1.Cells[8,0]:='Вид діяльності';
StringGrid1.Cells[9,0]:='Основна продукція';
StringGrid1.Cells[10,0]:='Менеджер';
Form1.Caption:='Вибірка... (0 записів з '+IntToStr(FileSize(NoteFile))+')';
end;

procedure TForm1.Button3Click(Sender: TObject);
Label Thenext1;
Label Thenext2;
Label Thenext3;
var
pPos,i,ch,zap: integer;
s: string;
begin
ch:=0;
for i:=1 to StringGrid1.RowCount-1 do
StringGrid1.Rows[i].Clear;
StringGrid1.RowCount:=2;
StringGrid1.FixedCols:=1;
StringGrid1.FixedRows:=1;
////////////////////////////////////
if ComboBox1.Text='Організація'
then

```

```

begin
  zap:=0;
  for pPos:=0 to FileSize(NoteFile)-1 do
  begin
    Seek(NoteFile,pPos);
    Read(NoteFile,NoteData); // читання даних з БД
    if NoteData.org=''
    then goto Thenext2;
    s:=NoteData.org;
    if pos(ComboBox2.Text,s)<>0
    then
      begin
        ch:=ch+1;
        StringGrid1.Cells[1,ch]:=IntToStr(ch)+'. ';
        StringGrid1.Cells[2,ch]:=NoteData.org;
        StringGrid1.Cells[3,ch]:=NoteData.adr;
        StringGrid1.Cells[4,ch]:=NoteData.tel;
        StringGrid1.Cells[5,ch]:=NoteData.kontl;
        StringGrid1.Cells[6,ch]:=NoteData.email;
        StringGrid1.Cells[7,ch]:=NoteData.reg;
        StringGrid1.Cells[8,ch]:=NoteData.vidd;
        StringGrid1.Cells[9,ch]:=NoteData.osnprod;
        StringGrid1.Cells[10,ch]:=NoteData.meneg;
        StringGrid1.RowCount:=StringGrid1.RowCount+1;
        zap:=zap+1;
        Form1.Caption:='Вибірка ('+IntToStr(zap)+' записів з
'+IntToStr(FileSize(NoteFile))+')';
        Application.ProcessMessages;
      end;
      Thenext2:
      end;
      Form1.Caption:='Вибірка... ('+IntToStr(zap)+' записів з
'+IntToStr(FileSize(NoteFile))+')';
      StringGrid1.RowCount:=StringGrid1.RowCount-1;
      end;
      //////////////////////////////////////
      if ComboBox1.Text='Період'
      then
        begin
          zap:=0;
          for pPos:=0 to FileSize(NoteFile)-1 do
          begin
            Seek(NoteFile,pPos);
            Read(NoteFile,NoteData); // читання даних з БД
            if NoteData.reg='' then goto Thenext1;
            s:=NoteData.reg;
            if s=ComboBox2.Text then
              begin
                ch:=ch+1;
                StringGrid1.Cells[1,ch]:=IntToStr(ch)+'. ';
                StringGrid1.Cells[2,ch]:=NoteData.org;
                StringGrid1.Cells[3,ch]:=NoteData.adr;
                StringGrid1.Cells[4,ch]:=NoteData.tel;
                StringGrid1.Cells[5,ch]:=NoteData.kontl;
                StringGrid1.Cells[6,ch]:=NoteData.email;
                StringGrid1.Cells[7,ch]:=NoteData.reg;
                StringGrid1.Cells[8,ch]:=NoteData.vidd;
                StringGrid1.Cells[9,ch]:=NoteData.osnprod;
                StringGrid1.Cells[10,ch]:=NoteData.meneg;
                StringGrid1.RowCount:=StringGrid1.RowCount+1;
                zap:=zap+1;
                Form1.Caption:='Вибірка... ('+IntToStr(zap)+' записів з
'+IntToStr(FileSize(NoteFile))+')';
                Application.ProcessMessages;
              end;
              Thenext1:
              end;
              Form1.Caption:='Вибірка... ('+IntToStr(zap)+' записів з
'+IntToStr(FileSize(NoteFile))+')';

```



```

Button2.Enabled:=true;
Button6.Enabled:=true;
Button6.Enabled:=true;
Button8.Enabled:=true;
Button9.Enabled:=true;
Button12.Enabled:=true;
end;

procedure TForm1.Button5Click(Sender: TObject);
var
  pPos: integer;
begin
  for pPos:=0 to StringGrid1.RowCount-2 do
    begin
      Seek(NoteFile,pPos); // переміщення покажчика на кінець файлу з БД
      NoteData.org:=StringGrid1.Cells[2,pPos+1];
      NoteData.adr:=StringGrid1.Cells[3,pPos+1];
      NoteData.tel:=StringGrid1.Cells[4,pPos+1];
      NoteData.kontl:=StringGrid1.Cells[5,pPos+1];
      NoteData.email:=StringGrid1.Cells[6,pPos+1];
      Notedata.reg:=StringGrid1.Cells[7,pPos+1];
      Notedata.vidd:=StringGrid1.Cells[8,pPos+1];
      Notedata.osnprod:=StringGrid1.Cells[9,pPos+1];
      Notedata.meneg:=StringGrid1.Cells[10,pPos+1];
      Write(NoteFile,NoteData); // запис у файл
    end;
  Form3.Close;
end;

procedure TForm1.Button6Click(Sender: TObject);
var
  EditFile:string;
begin
  Button2.Click;
  if SaveDialog1.Execute then
    begin
      EditFile:=SaveDialog1.FileName;
      RichEdit2.Lines.SaveToFile(EditFile);
    end;
end;

procedure AutoSizeGridColumn(Grid: TStringGrid; column: integer);
var
  i: integer;
  temp: integer;
  max: integer;
begin
  max:=0;
  for i:=0 to Grid.RowCount-1 do
    begin
      temp:=Grid.Canvas.TextWidth(Grid.Cells[column, i]);
      if temp>max
      then max:=temp;
    end;
  Grid.ColWidths[column]:=Max+Grid.GridLineWidth+6;
end;

procedure TForm1.Button8Click(Sender: TObject);
var
  i: integer;
begin
  for i:=0 to StringGrid1.ColCount-1 do
    AutoSizeGridColumn(StringGrid1, i);
  end;

procedure TForm1.Button7Click(Sender: TObject);
var
  n: integer;
begin

```

```

RichEdit1.Lines.LoadFromFile(GetProgramPath+'data\'+num.txt');
for n:=0 to RichEdit1.Lines.Count-1 do
begin
StringGrid1.Cells[1,n+1]:=RichEdit1.Lines.Strings[n];
StringGrid1.RowCount:=StringGrid1.RowCount+1;
end;
StringGrid1.RowCount:=StringGrid1.RowCount-1;

RichEdit1.Lines.LoadFromFile(GetProgramPath+'data\'+org.txt');
for n:=0 to RichEdit1.Lines.Count-1 do
begin
StringGrid1.Cells[2,n+1]:=RichEdit1.Lines.Strings[n];
end;

RichEdit1.Lines.LoadFromFile(GetProgramPath+'data\'+adr.txt');
for n:=0 to RichEdit1.Lines.Count-1 do
begin
StringGrid1.Cells[3,n+1]:=RichEdit1.Lines.Strings[n];
end;

RichEdit1.Lines.LoadFromFile(GetProgramPath+'data\'+tel.txt');
for n:=0 to RichEdit1.Lines.Count-1 do
begin
StringGrid1.Cells[4,n+1]:=RichEdit1.Lines.Strings[n];
end;

RichEdit1.Lines.LoadFromFile(GetProgramPath+'data\'+kont1.txt');
for n:=0 to RichEdit1.Lines.Count-1 do
begin
StringGrid1.Cells[5,n+1]:=RichEdit1.Lines.Strings[n];
end;

RichEdit1.Lines.LoadFromFile(GetProgramPath+'data\'+email.txt');
for n:=0 to RichEdit1.Lines.Count-1 do
begin
StringGrid1.Cells[6,n+1]:=RichEdit1.Lines.Strings[n];
end;

RichEdit1.Lines.LoadFromFile(GetProgramPath+'data\'+reg.txt');
for n:=0 to RichEdit1.Lines.Count-1 do
begin
StringGrid1.Cells[7,n+1]:=RichEdit1.Lines.Strings[n];
end;

RichEdit1.Lines.LoadFromFile(GetProgramPath+'data\'+vidd.txt');
for n:=0 to RichEdit1.Lines.Count-1 do
begin
StringGrid1.Cells[8,n+1]:=RichEdit1.Lines.Strings[n];
end;

RichEdit1.Lines.LoadFromFile(GetProgramPath+'data\'+osnprod.txt');
for n:=0 to RichEdit1.Lines.Count-1 do
begin
StringGrid1.Cells[9,n+1]:=RichEdit1.Lines.Strings[n];
end;
RichEdit1.Lines.LoadFromFile(GetProgramPath+'data\'+meneg.txt');
for n:=0 to RichEdit1.Lines.Count-1 do
begin
StringGrid1.Cells[10,n+1]:=RichEdit1.Lines.Strings[n];
end;
end;

procedure TForm1.Button9Click(Sender: TObject);
var
i,j,cl,rt,cr,rb: integer;
s: string;
CopySel: Boolean;
begin

```

```

// CopySel:=true - копіювати виділення
// CopySel:=false - всі осередки
CopySel:=false;
CL:=-1;
RT:=-1;
CR:=-1;
RB:=-1;
s:='';
with StringGrid1 do
begin
  if CopySel then
  begin
    CL:=Selection.Left;
    CR:=Selection.Right;
    RT:=Selection.Top;
    RB:=Selection.Bottom;
  end;
  // при необхідності FixedRows і FixedCols можна замінити на 0
  if (CL<FixedCols) or (CL>CR) or (CL>=ColCount) then CL:=FixedCols;
  if (CR<FixedCols) or (CL>CR) or (CR>=ColCount) then CR:= ColCount-1;
  if (RT<FixedRows) or (RT>RB) or (RT>=RowCount) then RT:=FixedRows;
  if (RB<FixedCols) or (RT>RB) or (RB>=RowCount) then RB:= RowCount-1;
  for i:=RT to RB do
  begin
    for j:=CL to CR do
    begin
      s:=s+Cells[j,i];
      if j<CR then s:=s+#9;
    end;
    s:=s+#13#10;
  end;
  Clipboard.AsText:=s;
end;

function IsOLEObjectInstalled(Name: String): boolean;
var
  ClassID: TCLSID;
  Rez: HRESULT;
begin
  // шукаємо CLSID OLE-Об'єкта
  Rez:=CLSIDFromProgID(PWideChar(WideString(Name)), ClassID);
  if Rez=S_OK
  then Result:=true // об'єкт знайдений
  else Result:=false;
end;

procedure TForm1.Button11Click(Sender: TObject);
var
  XL: variant; // Змінна в якій створюється об'єкт EXCEL
  i,j,n: integer;
begin
  // Створення документа (об'єкт Excel)
  XL:=CreateOleObject('Excel.Application');
  // Щоб не ставив запитання про збереження документа
  XL.DisplayAlerts:=false;
  // новий документ
  XL.WorkBooks.Add;
  // або завантажуюмо його
  XL.WorkBooks.Open('akt.xls');
  // Робимо його видимим
  XL.Visible:=true;
  // Коли прога вже оттестирована, краще це робити наприкінці,
  // швидше працює, а поки ні, краще на початку
  // Лівого й правого поля відступу для печатки
  XL.WorkBooks[1].Worksheets[1].PageSetup.LeftMargin:=30;
  XL.WorkBooks[1].Worksheets[1].PageSetup.RightMargin:=10;
  // Даємо назву сторінці
  XL.WorkBooks[1].Worksheets[1].Name:='Прайс аркуш';

```

```

// Рядок з'являється на кожному аркуші при печаті
XL.WorkBooks[1].WorkSheets[1].PageSetup.PrintTitleRows:='$3:$3';
XL.WorkBooks[1].WorkSheets[1].PageSetup.PrintTitleColumns:='$A:$A';
// формат числа
for i := 4 to 13 do
  XL.WorkBooks[1].WorkSheets[1].Columns[i].NumberFormat:='0,00';
XL.WorkBooks[1].WorkSheets[1].Columns[4].NumberFormat:='0';
// Таким способом можна задавати ширину стовпчика
XL.WorkBooks[1].WorkSheets[1].Columns[1].ColumnWidth:=4.5;
XL.WorkBooks[1].WorkSheets[1].Columns[2].ColumnWidth:=50;
for i := 3 to 13 do
  XL.WorkBooks[1].WorkSheets[1].Columns[i].ColumnWidth := 8;
// Шрифт жирний
XL.WorkBooks[1].WorkSheets[1].Rows[1].Font.Bold:=True;
XL.WorkBooks[1].WorkSheets[1].Rows[1].Font.Color:=clBlack;
XL.WorkBooks[1].WorkSheets[1].Rows[1].Font.Size:=16;
XL.WorkBooks[1].WorkSheets[1].Rows[1].Font.Name:='Times New Roman';
XL.WorkBooks[1].WorkSheets[1].Cells[1,4]:='Прайс аркуш';
// Вирівнюємо по центру по вертикалі
XL.WorkBooks[1].WorkSheets[1].Rows[1].VerticalAlignment:=2;
// Вирівнюємо по центру по горизонталі
XL.WorkBooks[1].WorkSheets[1].Rows[1].HorizontalAlignment:=3;
// Об'єднуємо осередку
XL.WorkBooks[1].WorkSheets[1].Range['A1:D1'].Merge;
// Вирівнюємо по центру по вертикалі
XL.WorkBooks[1].WorkSheets[1].Rows[3].VerticalAlignment:=2;
// Вирівнюємо по центру по горизонталі
XL.WorkBooks[1].WorkSheets[1].Rows[3].HorizontalAlignment:=3;
// Вирівнюємо по лівому краю
XL.WorkBooks[1].WorkSheets[1].Cells[3,2].HorizontalAlignment:=2;
XL.WorkBooks[1].WorkSheets[1].Cells[3,3].HorizontalAlignment:=2;
// Вирівнюємо по правому краю
XL.WorkBooks[1].WorkSheets[1].Cells[3,4].HorizontalAlignment:=4;
XL.WorkBooks[1].WorkSheets[1].Rows[3].Font.Color:=clBlack;
XL.WorkBooks[1].WorkSheets[1].Rows[3].Font.Name:='Times New Roman';
XL.WorkBooks[1].WorkSheets[1].Rows[3].Font.Size:=12;
XL.WorkBooks[1].WorkSheets[1].Rows[3].Font.Bold:=True;
XL.WorkBooks[1].WorkSheets[1].Cells[3,1]:='№';
XL.WorkBooks[1].WorkSheets[1].Cells[3,2]:='Найменування продукції';
XL.WorkBooks[1].WorkSheets[1].Cells[3,3]:='Одиниця виміру';
// окреслення діапазону осередків тільки знизу
// Borders[1] .... [4] - це край осередку ColorIndex -4142 - порожній колір i i
n - змінні

XL.WorkBooks[1].WorkSheets[1].Range['A'+IntToStr(i)]:'+chr(ord('C')+n)+IntToStr
(i)].Borders.LineStyle:=1;

XL.WorkBooks[1].WorkSheets[1].Range['A'+IntToStr(i)]:'+chr(ord('C')+n)+IntToStr
(i)].Borders.Weight:=2;

XL.WorkBooks[1].WorkSheets[1].Range['A'+IntToStr(i)]:'+chr(ord('C')+n)+IntToStr
(i)].Borders[4].ColorIndex := 1;

XL.WorkBooks[1].WorkSheets[1].Range['A'+IntToStr(i)]:'+chr(ord('C')+n)+IntToStr
(i)].Borders[1].ColorIndex := -4142;

XL.WorkBooks[1].WorkSheets[1].Range['A'+IntToStr(i)]:'+chr(ord('C')+n)+IntToStr
(i)].Borders[2].ColorIndex := -4142;

XL.WorkBooks[1].WorkSheets[1].Range['A'+IntToStr(i)]:'+chr(ord('C')+n)+IntToStr
(i)].Borders[3].ColorIndex := -4142;
// окреслення діапазону осередків

XL.WorkBooks[1].WorkSheets[1].Range['A3:'+chr(ord('C')+n)+IntToStr(i)].Borders.L
ineStyle:=1;

XL.WorkBooks[1].WorkSheets[1].Range['A3:'+chr(ord('C')+n)+IntToStr(i)].Borders.W
eight:=2;

```

```

XL.WorkBooks[1].Worksheets[1].Range['A3:'+chr(ord('C')+n)+IntToStr(i)].Borders.ColorIndex:=1;
// присвоєння осередку значення
XL.WorkBooks[1].Worksheets[1].Cells[i,j]:=' До-В';
// Повертати слова, писати вертикально, під кутом і т.д.
XL.WorkBooks[1].Worksheets[1].Rows[2].Orientation:=90;
XL.WorkBooks[1].Worksheets[1].Range['A2:B2'].Orientation:=0;
end;

```

```

procedure TForm1.ComboBox1Change(Sender: TObject);

```

```

var

```

```

  cPos, i, j: integer;

```

```

  s: string;

```

```

begin

```

```

  if ComboBox1.Text='Всі'

```

```

  then ComboBox2.Enabled:=false

```

```

  else ComboBox2.Enabled:=true;

```

```

  //////////////////////////////////////// Організація

```

```

  ComboBox2.Clear;

```

```

  ComboBox2.Text:='ТОВ';

```

```

  ComboBox2.Items.Add('ТОВ');

```

```

  ComboBox2.Items.Add('ВАТ');

```

```

  ComboBox2.Items.Add('ЗАТ');

```

```

  ComboBox2.Items.Add('ГВП');

```

```

  ComboBox2.Items.Add('НДІ');

```

```

  ComboBox2.Items.Add('МІІ');

```

```

  ComboBox2.Items.Add('МУП');

```

```

  ComboBox2.Items.Add('ДНЦ');

```

```

  ComboBox2.Items.Add('НПО');

```

```

  ComboBox2.Items.Add('ІЧП');

```

```

  ComboBox2.Items.Add('ДОУ');

```

```

  ComboBox2.Items.Add('ПБЮЛ');

```

```

  //////////////////////////////////////// Рєгiон

```

```

  if ComboBox1.Text='Рєгiон' then

```

```

  begin

```

```

    ComboBox2.Text:='';

```

```

    ComboBox2.Items.Clear;

```

```

    cPos:=0;

```

```

    Seek(NoteFile,cPos);

```

```

    Read(NoteFile,NoteData); // читання даних з БД

```

```

    s:=NoteData.reg;

```

```

    for i:=0 to FileSize(NoteFile)-1 do

```

```

    begin

```

```

      if s<>' ' then

```

```

      begin

```

```

        ComboBox2.Text:=s;

```

```

        ComboBox2.Items.Add(s);

```

```

        break;

```

```

      end;

```

```

    inc(cPos);

```

```

    Seek(NoteFile,cPos);

```

```

    Read(NoteFile,NoteData); // читання даних з БД

```

```

    s:=NoteData.reg;

```

```

  end;

```

```

  for cPos:=0 to FileSize(NoteFile)-1 do

```

```

  begin

```

```

    Seek(NoteFile,cPos);

```

```

    Read(NoteFile,NoteData); // читання даних з БД

```

```

    s:=NoteData.reg;

```

```

    if s<>' ' then

```

```

    begin

```

```

      for i:=0 to ComboBox2.Items.Count-1 do

```

```

      if pos(s,ComboBox2.Items.Strings[i])=0

```

```

      then j:=2

```

```

      else

```

```

      begin

```

```

        j:=1;

```

```

        Break;

```

```

        end;
    end;
    if j=2 then ComboBox2.Items.Add(s);
    j:=0;
end;
end;
////////// Менеджер
if ComboBox1.Text='Менеджер' then
begin
    ComboBox2.Text:='';
    ComboBox2.Items.Clear;
    cPos:=0;
    Seek(NoteFile,cPos);
    Read(NoteFile,NoteData); // читання даних з БД
    s:=NoteData.meneg;
    for i:=0 to FileSize(NoteFile)-1 do
    begin
        if s<>' ' then
        begin
            ComboBox2.Text:=s;
            ComboBox2.Items.Add(s);
            break;
        end;
        inc(cPos);
        Seek(NoteFile,cPos);
        Read(NoteFile,NoteData); // читання даних з БД
        s:=NoteData.meneg;
    end;
    for cPos:=0 to FileSize(NoteFile)-1 do
    begin
        Seek(NoteFile,cPos);
        Read(NoteFile,NoteData); // читання даних з БД
        s:=NoteData.meneg;
        if s<>' ' then
        begin
            for i:=0 to ComboBox2.Items.Count-1 do
            if pos(s,ComboBox2.Items.Strings[i])=0
            then j:=2
            else
            begin
                j:=1;
                Break;
            end;
        end;
        if j=2 then ComboBox2.Items.Add(s);
        j:=0;
    end;
end;
//////////
ComboBox2.Sorted:=true;
end;

procedure TForm1.Button10Click(Sender: TObject);
var
    i,j,r,c: integer;
    WorkBk: _WorkBook; // визначаємо Workbook
    WorkSheet: _WorkSheet; // визначаємо WorkSheet
    iIndex: OleVariant;
    TabGrid: Variant;
begin
    iIndex:=1;
    r:=StringGrid1.RowCount; // кількість рядків
    c:=StringGrid1.ColCount; // кількість стовпців
    // Створюємо масив-матрицю
    TabGrid:=VarArrayCreate([0,(r-1),0,(c-1)],VarOleStr);
    i:=0;
    // Визначаємо цикл для заповнення масиву-матриці
    repeat
        for j:=0 to (c-1) do // заповнення TabGrid з StringGrid1

```

```

    TabGrid[i,j]:=StringGrid1.Cells[j+1,i];
    inc(i,1);
until i>( r-1);
// З'єднуємося із сервером TExcelApplication
XLApp.Connect;
// Додаємо WorkBooks в ExcelApplication
XLApp.WorkBooks.Add(xlWBatWorkSheet,0); // не працює в XP !!!
// Вибираємо першу Workbook
WorkBk:=XLApp.WorkBooks.Item[iIndex];
// Визначаємо перший Worksheet
Worksheet:=WorkBk.WorkSheets.Get_Item(1) as _Worksheet;
// Зіставляємо Delphi масив-матрицю з матрицею в Worksheet
// Worksheet.Range['A1',Worksheet.Cells.Item[r,c]].Value:=TabGrid;
// Заповнюємо властивості Worksheet
Worksheet.Name:='Звіт';
// Колонтитули (дробові числа розділяються тільки '.')
Worksheet.PageSetup.HeaderMargin:=XLApp.CentimetersToPoints(0.5);
Worksheet.PageSetup.RightHeader:='Сторінка &[Сторінка] з &[Сторінок] -
&[Дата]';
// Орієнтація сторінки: книжкова (xlPortrait(:=1)) або альбомна
Worksheet.PageSetup.Orientation:=xlLandscape; // альбомна (:=2)
Worksheet.Columns.WrapText:=true; // переносити за словами
Worksheet.Columns.AutoFit; // автовисота
Worksheet.Columns.HorizontalAlignment:=xlRight;
Worksheet.Columns.VerticalAlignment:=xlCenter;
Worksheet.Columns.ColumnWidth:=12;
Worksheet.Columns.Font.Size:=8;
// Обрамлення (xlDouble,...)
Worksheet.Columns.Borders.LineStyle:=xlContinuous;
// Ширина першого рядка
Worksheet.Range['A'+IntToStr(1),'A'+IntToStr(R)].ColumnWidth:=6;
// Розмір шрифту 1-го стовпця
Worksheet.Range['A'+IntToStr(1),'M'+IntToStr(1)].Font.Size:=10;
// Вирівнювання першого стовпця

Worksheet.Range['A'+IntToStr(1),'A'+IntToStr(R)].HorizontalAlignment:=xlAlignLe
ft;
// Вирівнювання першого рядка
Worksheet.Range['A'+IntToStr(1),'M'+IntToStr(1)].HorizontalAlignment:=xlCenter;
// Показуємо Excel
XLApp.Visible[0]:=true;
XLApp.ScreenUpdating[0]:=true;
// Розриваємо зв'язок із сервером
XLApp.Disconnect;
// Unassign the Delphi Variant Matrix
TabGrid:=Unassigned;
end;

procedure TForm1.Button12Click(Sender: TObject);
var
    i,j,r,c: integer;
    Excel,Workbook,Sheet,TabGrid: Variant;
    iIndex: OleVariant;
begin
    iIndex:=1;
    r:=StringGrid1.RowCount; // кількість рядків
    c:=StringGrid1.ColCount; // кількість стовпців
    // Створюємо масив-матрицю
    TabGrid:=VarArrayCreate([0,( r-1),0,( c-1)],VarOleStr);
    i:=0;
    // Визначаємо цикл для заповнення масиву-матриці
    repeat
        for j:=0 to ( c-1) do // заповнення TabGrid з StringGrid1
            TabGrid[i,j]:=StringGrid1.Cells[j+1,i];
            inc(i,1);
        until i>( r-1);
    // Завантажуємо Excel
    try
        Excel:=CreateOleObject('Excel.Application'); // для інших

```

```

except
  Excel:=CreateOleObject('Excel.Application.10'); // для Office XP
end;
Excel.SheetsInNewWorkbook:=1;
WorkBook:=Excel.WorkBooks.Add;
Sheet:=WorkBook.WorkSheets[1];
Sheet.Name:='База менеджерів - Звіт';
Sheet.Range['A6',Sheet.Cells.Item[r+5, c-1]].Value:=TabGrid;
// Орієнтація сторінки: книжкова (xlPortrait або :=1)
Sheet.PageSetup.CenterHorizontally:=true;
Sheet.PageSetup.Orientation:=xlLandscape; // альбомна (або :=2)
// Поля (дробові числа розділяються тільки '.')
Sheet.PageSetup.LeftMargin:=Excel.CentimetersToPoints(1);
Sheet.PageSetup.RightMargin:=Excel.CentimetersToPoints(1);
Sheet.PageSetup.TopMargin:=Excel.CentimetersToPoints(1);
Sheet.PageSetup.BottomMargin:=Excel.CentimetersToPoints(1);
// Колонтитули (дробові числа розділяються тільки '.')
Sheet.PageSetup.HeaderMargin:=Excel.CentimetersToPoints(0.5);
// &З, &ДО, &Д - сторінка, кількість (усього), дата (dd,mm,yy)
// тільки для російського MS-Office; для English - &P, &N, &D
Sheet.PageSetup.RightHeader:='Сторінка &Із із &ДО - &Д';
// Властивості таблиці
Sheet.Columns.WrapText:=true; // переносити за словами
Sheet.Cells.Columns.AutoFit; // автовисота рядків
Sheet.Range['A'+IntToStr(6), 'J'+IntToStr(6)].RowHeight:=28;
Sheet.Columns.HorizontalAlignment:=xlLeft;
Sheet.Columns.VerticalAlignment:=xlCenter;
Sheet.Columns.ColumnWidth:=12;
Sheet.Columns.Font.Size:=8;
// Шапка над таблицею ([3,4] - рядок, стовпець)
Sheet.Cells[3,4]:='ТОВ " Адс-Континент" - База менеджерів за '+DateToStr(Date);
Sheet.Range['A3', 'J3'].HorizontalAlignment:=xlHAlignCenter;
Sheet.Range['A3', 'J3'].Font.Name:='Times New Roman';
Sheet.Range['A3', 'J3'].Font.Bold:=true;
Sheet.Range['A3', 'J3'].Font.Size:=14;
Sheet.Range['A3', 'J3'].Columns.WrapText:=false;
// Обрамлення осередків (xlDouble,...)

Sheet.Range['A'+IntToStr(6), 'J'+IntToStr(r+5)].Columns.Borders.LineStyle:=xlCont
inuou;
// Ширина стовпців
Sheet.Range['A'+IntToStr(6), 'A'+IntToStr(r+5)].ColumnWidth:=6;
Sheet.Range['B'+IntToStr(6), 'B'+IntToStr(r+5)].ColumnWidth:=24;
Sheet.Range['C'+IntToStr(6), 'C'+IntToStr(r+5)].ColumnWidth:=22;
Sheet.Range['D'+IntToStr(6), 'D'+IntToStr(r+5)].ColumnWidth:=14;
Sheet.Range['E'+IntToStr(6), 'E'+IntToStr(r+5)].ColumnWidth:=12;
Sheet.Range['F'+IntToStr(6), 'F'+IntToStr(r+5)].ColumnWidth:=12;
Sheet.Range['G'+IntToStr(6), 'G'+IntToStr(r+5)].ColumnWidth:=8;
Sheet.Range['H'+IntToStr(6), 'H'+IntToStr(r+5)].ColumnWidth:=12;
Sheet.Range['I'+IntToStr(6), 'I'+IntToStr(r+5)].ColumnWidth:=12;
Sheet.Range['J'+IntToStr(6), 'J'+IntToStr(r+5)].ColumnWidth:=12;
// Розмір шрифту 6-го стовпця
Sheet.Range['A'+IntToStr(6), 'J'+IntToStr(6)].Font.Size:=10;
// Вирівнювання першого стовпця

Sheet.Range['A'+IntToStr(6), 'A'+IntToStr(r+5)].HorizontalAlignment:=xlHAlignCent
er;
Sheet.Range['A'+IntToStr(6), 'A'+IntToStr(r+5)].Font.Bold:=true;
// Вирівнювання першого рядка
Sheet.Range['A'+IntToStr(6), 'J'+IntToStr(6)].HorizontalAlignment:=xlCenter;
Sheet.Range['A'+IntToStr(6), 'J'+IntToStr(6)].Font.Bold:=true;
// Показуємо Excel
Excel.Visible:=True;
Excel.ScreenUpdating:=true;
//
Excel:=Unassigned;
WorkBook:=Unassigned;
Sheet:=Unassigned;
TabGrid:=Unassigned;

```

```
end;

procedure TForm1.Button4Click(Sender: TObject);
begin
  Close;
end;

procedure TForm1.StringGrid1Click(Sender: TObject);
begin
  StringGrid1.ShowHint:=false;
end;

procedure TForm1.StringGrid1DbClick(Sender: TObject);
var
  i,j:integer;
begin
  StringGrid1.ShowHint:=true;
  j:=StringGrid1.Row;
  StringGrid1.Hint:=StringGrid1.Cells[1,j];
  for i:=2 to StringGrid1.ColCount-1 do
    StringGrid1.Hint:=StringGrid1.Hint+' - '+StringGrid1.Cells[i,j];
  end;
end.

end.
```

Кафедра _ КБПЗ _ 2022 рік

Файл avt.pas - авторизація

```

unit paswd;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, StdCtrls;

type
  TForm2 = class(TForm)
    Button1: TButton;
    Edit1: TEdit;
    Label1: TLabel;
    procedure Button1Click(Sender: TObject);
    procedure FormClose(Sender: TObject; var Action: TCloseAction);
    procedure Edit1KeyPress(Sender: TObject; var Key: Char);
    procedure FormCreate(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  Form2: TForm2;

implementation

uses adsbaz, usprint;

{$R *.dfm}

procedure TForm2.Button1Click(Sender: TObject);
begin
  if Edit1.Text='QWERTY'
  then
    begin
      Screen.Cursor:=crHourGlass;
      Sleep(400);
      Screen.Cursor:=crDefault;
      Form3.Show;
      Form2.Hide;
    end
  else
    begin
      MessageDlg('Невірний пароль!', mtError, [mbOK], 0);
      Close;
    end;
end;

procedure TForm2.FormClose(Sender: TObject; var Action: TCloseAction);
begin
  Action:=caFree;
end;

procedure TForm2.Edit1KeyPress(Sender: TObject; var Key: Char);
begin
  if Key=#13 then Button1.Click;
end;

procedure TForm2.FormCreate(Sender: TObject);
begin
  // пароль як при вході в Win XP
  Edit1.Font.Name:='Wingdings';
  Edit1.PasswordChar:='l'; // символ "крапка"
  //

```

```
Application.Title:='База підприємств і менеджерів v0.99';  
end;  
  
end.
```

Кафедра _ КБПЗ _ 2022 рік

Файл about.pas - довідка про програму

```
unit about;

interface

uses
  Classes, Graphics, Controls, Forms, StdCtrls, ExtCtrls, ShellApi, jpeg,
  Buttons;

type
  TfmAbout = class(TForm)
    Label1: TLabel;
    Label2: TLabel;
    Label3: TLabel;
    Label4: TLabel;
    Label5: TLabel;
    Label6: TLabel;
    Label7: TLabel;
    Label8: TLabel;
    BitBtn1: TBitBtn;
    Image1: TImage;
    procedure Label7Click(Sender: TObject);
    procedure BitBtn1Click(Sender: TObject);
    procedure FormCreate(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  fmAbout: TfmAbout;

implementation

{$R *.DFM}

procedure TfmAbout.BitBtn1Click(Sender: TObject);
begin
  fmAbout.Close;
end;

procedure TfmAbout.FormCreate(Sender: TObject);
begin
  Label1.Caption:='МАГІСТЕРСЬКА РОБОТА';
  Label2.Caption:='на тему:';
  Label3.Caption:='Дослідження та програмна реалізація системи довідково-
інформаційного сервісу комп'ютерної мережі відділу підприємства';
  Label4.Caption:='';
  Label5.Caption:='Розробив: студент Борзенко В`ячеслав Юрійович';
  Label6.Caption:='гр. КІ-21М-1,4';
  Label7.Caption:=' Керівник: Буравченко К.О.';
  Label8.Caption:='м. Кропивницький 2022';
end;
end.
```