

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
СХІДНОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМ. В. ДАЛЯ
ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ ТА ЕЛЕКТРОНІКИ
КАФЕДРА КОМП'ЮТЕРНИХ НАУК ТА ІНЖЕНЕРІЇ

До захисту допускається
Завідувач кафедри
Скарга-Бандурова І.С.
« ____ » _____ 2017 р.

МАГІСТЕРСЬКА РОБОТА

НА ТЕМУ:

Дослідження та розробка програмних засобів обміну інформацією в
розподіленій системі

Освітньо-кваліфікаційний рівень “Магістр”
Спеціальність 123 - “Комп’ютерна інженерія”
(освітня програма - “Системне програмування”)

Науковий керівник роботи:

(підпис)

Щербакова М.Є.
(ініціали, прізвище)

Консультант з охорони праці:

(підпис)

Критська Я.О.
(ініціали, прізвище)

Студент:

(підпис)

Крючков В.С.
(ініціали, прізвище)

Група:

СП-16ДМ

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
СХІДНОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ВОЛОДИМИРА ДАЛЯ

Факультет Інформаційних технологій та електроніки
Кафедра Комп'ютерних наук та інженерії
Освітньо-кваліфікаційний рівень “магістр”
Напрямок підготовки _____
Спеціальність 123 “Комп'ютерна інженерія” (освітня програма - “Системне програмування”)

ЗАТВЕРДЖУЮ

Зав. кафедри комп'ютерної інженерії

д.т.н., доц. Скарга-Бандурова І.С.

” ” _____ 201_ р.

**З А В Д А Н Н Я
НА МАГІСТЕРСЬКУ РОБОТУ СТУДЕНТУ**

Крючкова Володимира Сергійовича

(прізвище, ім'я, по батькові)

1. Тема роботи Дослідження та розробка програмних засобів обміну інформацією в розподіленій системі

керівник проекту (роботи) к.т.н., доц. Щербакова М.Є.

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

затверджені наказом СНУ ім. В. Даля від “___” _____ 20__ року №__

2. Строк подання студентом роботи _____

3. Вихідні дані до роботи _____

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

1. Аналіз існуючої проблеми 2. Побудова математичної моделі системи 3. Практична реалізація.

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

6. Консультанти роботи, з вказівкою розділів, що до них відносяться

Розділ	Консультант	Підпис, дата	
		Завдання видав	Завдання прийняв
Охорона праці	<i>Критська Я.О</i>		

7. Дата видачі завдання ____.

Керівник _____

(підпис)

Завдання прийняв до виконання _____

(підпис)

КАЛЕНДАРНИЙ ПЛАН

№ п/п	Найменування етапів магістерської роботи	Термін виконання етапів роботи	Примітка
1	<i>Сбір інформації для диплома</i>		
2	<i>Аналіз сучасних методів вирішення проблеми</i>		
3	<i>Аналіз існуючих програмних засобів</i>		
4	<i>Побудова математичної моделі та її аналіз</i>		
5	<i>Розробка практичної частини</i>		
6	<i>Розробка заходів з охорони праці</i>		
7	<i>Оформлення пояснювальної записки і графічного матеріалу</i>		
8	<i>Підготовка та подання магістерської роботи до захисту</i>		

Студент _____

(підпис)

Науковий керівник _____

(підпис)

АННОТАЦИЯ

Крючков В.С. Анализ и разработка программных средств обмена информацией на основе распределённой системы.

Рассмотрены особенности обмена информацией в ОП ИВЦ ОАО «Лисичанскуголь». Впервые предложена адаптивная система документооборота на основе распределённой сети. Защита передаваемых файлов обеспечивается путём их кодирования. Разработана клиентская и серверная часть программного обеспечения. В случае выхода серверной части из работы программа переключается в распределённый режим работы.

Ключевые слова: система, документооборот, распределённая сеть, адаптивность, обмен, информация, программное обеспечение.

АНОТАЦІЯ

Крючков В.С. Аналіз і розробка програмних засобів обміну інформацією на основі розподіленої системи.

Розглянуто особливості обміну інформацією в ОП ІОЦ ПАТ «Лисичанськвугілля». Вперше запропоновано адаптивна система документообігу на основі розподіленої мережі. Захист переданих файлів забезпечується шляхом їх кодування. Розроблено клієнтська і серверна частина програмного забезпечення. У разі виходу серверної частини з роботи програма перемикається в розподілений режим роботи.

Ключові слова: система, документообіг, розподілена мережа, адаптивність, обмін, інформація, програмне забезпечення.

ANNOTATION

Kryuchkov V.S. Analysis and development of software tools for information exchange based on a distributed system.

The peculiarities of information exchange in the Information and Analytical Center of PIC "Lisichanskugol" are considered. For the first time, an adaptive document management system based on a distributed network was proposed. Protection of transmitted files is provided by encoding them. The client and server part of the software is developed. If the server part exits the work, the program switches to the distributed mode of operation.

Keywords: system, workflow, distributed network, adaptability, exchange, information, software.

ЗМІСТ

ВСТУП.....	9
1 ОГЛЯД СУЧАСНИХ РОБІТ В СФЕРІ ДОКУМЕНТООБІГУ.....	12
1.1 Загальні відомості.....	12
1.2 Принципи документообігу	13
1.3 Категорії системи документообігу	13
1.4 Мета СЕД	14
1.5 Види систем електронного документообігу	15
1.5.1 Система документообігу Directum	17
1.5.1.1 Функціональність і можливості інтеграції	17
1.5.1.2 Інтерфейс	18
1.5.1.3 Функції пошуку.....	19
1.5.1.4 Адміністративні настройки.....	19
1.5.1.5 Безпека даних і дій.....	20
1.5.2 Система документообігу ELMA.....	20
1.5.2.1 Функціональність і можливості інтеграції	20
1.5.2.2 Інтерфейс	21
1.5.2.3 Функції пошуку.....	22
1.5.2.4 Адміністративні настройки.....	22
1.5.2.5 Безпека даних і дій.....	23
1.5.3 Система документообігу Optima WorkFlow	24
1.5.3.1 Функціональність і можливості інтеграції	24
1.5.3.2 Інтерфейс	25
1.5.3.3 Функції пошуку.....	26
1.5.3.4 Адміністративні налаштування	26
1.5.3.5 Безпека даних і дій.....	27
1.5.4 Система документообігу 1С: Документооборот 8.....	27
1.5.4.1 Функціональність і можливості інтеграції	27
1.5.4.2 Інтерфейс	28
1.5.4.3 Функції пошуку.....	29
1.5.4.4 Адміністративні настройки.....	29
1.5.4.5 Безпека даних і дій.....	29
1.6 Висновки до розділу 1.....	30

2	АНАЛІЗ СИСТЕМИ ОБМІНУ ІНФОРМАЦІЄЮ НА ПІДПРИЄМСТВІ.....	31
2.1	Стабільність роботи	31
2.2	Робота з програмою.....	33
2.3	Коефіцієнт корисної дії.....	35
2.4	Підвищення стабільності	38
2.5	Робота з інформацією.....	38
2.6	Користь системи	40
2.7	Загальна користь.....	42
2.8	Математична модель та її аналіз.....	43
2.9	Висновки до розділу 2.....	44
3	ПРАКТИЧНА РЕАЛІЗАЦІЯ	45
3.1	Загальні відомості про Microsoft Visual Studio.....	45
3.2	Загальні відомості про C++	46
3.2.1	Узагальнена інформація.....	46
3.2.2	Історія розвитку	46
3.2.3	Переваги мови	48
3.3	Загальні відомості про C#.....	48
3.3.1	Узагальнена інформація про C#	48
3.3.2	Історія розвитку C#.....	50
3.3.3	Переваги мови	50
3.4	Розробка серверної частини	51
3.5	Розробка клієнтської частини.....	63
3.6	Висновки до розділу 3.....	69
4	ОХОРОНА ПРАЦІ ТА БЕЗПЕКИ В НАДЗВИЧАЙНИХ СИТУАЦІЯХ. ЕКОЛОГІЯ ..	70
4.1	Загальні питання з охорони праці.....	70
4.2	Правові та організаційні основи охорони праці	70
4.3	Організаційно-технічні заходи з безпеки праці.....	71
4.4	Аналіз стану умов праці.....	72
4.4.1	Вимоги до приміщень.....	72
4.4.2	Вимоги до організації місця праці	72
4.4.3	Навантаження та напруженість процесу праці	73
4.4.4	Виробнича санітарія	74
4.4.5	Аналіз небезпечних та шкідливих факторів.....	74
4.4.6	Пожежна безпека	77
4.4.7	Електробезпека.....	78

4.5	Гігієнічні вимоги до параметрів виробничого середовища	78
4.5.1	Мікроклімат.....	78
4.5.2	Освітлення	79
4.5.3	Шум та вібрація, електромагнітне випромінювання.....	81
4.5.4	Вентилювання	81
4.6	Заходи з організації виробничого середовища та попередження виникнення надзвичайних ситуацій.....	82
4.7	Вимоги безпеки при надзвичайних ситуаціях:	83
4.8	Розрахунок захисного заземлення (забезпечення електробезпеки будівлі).	84
4.9	Охорона навколишнього природного середовища	87
4.9.1	Загальні дані з охорони навколишнього природного середовища	87
4.9.2	Вимоги до збору, пакування та розміщення відходів ІТ галузі	88
4.9.3	Визначення впливу та заходів щодо поводження з відходами ІТ галузі.....	90
4.10	Висновки до розділу.....	90
	ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ	92
	ПЕРЕЛІК ВІКОРИСТАНИХ ДЖЕРЕЛ.....	95
	Додаток А	98
	Додаток Б.....	106
	Додаток В	111

ПЕРЕЛІК СКОРОЧЕНЬ

ОП – Відокремлений підрозділ

ИВЦ – Інформаційно-обчислювальний центр

ОАО – Відкрите акціонерне товариство

ПО – Програмне забезпечення

СЭДО – Система електронного документообігу

ЭД – Електронний документ

ЭДО – Електронний документообіг

ВСТУП

Актуальність теми. Дана тема була обрана у зв'язку з тим, що на ОП ІОЦ ВАТ «Лисичанськвугілля» виникла проблема з обміном інформацією між відділами. Але вже існуючі програмні засоби не задовольняють вимогам. А саме всі існуючі системи документообігу не дуже надійні. Це обумовлено тим, що найчастіше вони є залежними від сервера. І в разі його відключення або ж виходу з ладу система стає непридатною для експлуатації. Є так само і інший тип системи, а саме розподілені. На відміну від попереднього варіанту, вони не потребують серверної частини. Але їх основним мінусом є те, що в разі відсутності сегмента всередині мережі, файли які на ньому зберігаються, стають недоступними. Таким чином, кожна реалізація має свої плюси і мінуси. І дана розробка враховує мінуси обох варіантів, і усуває їх.

Дана робота спрямована на дослідження і розробку програмних засобів для захищеного обміну інформацією та впровадження їх на підприємства. А також вироблення навичок по роботі з сучасними засобами обміну інформацією, дослідження предметної області і підготовку готового програмного забезпечення. Що і мають на увазі навчальні плани.

Для виявлення проблеми на підприємстві був проведений збір і аналіз даних емпіричним методом. А саме було проведене опитування серед співробітників підприємства і його керівництва, вимір часу, який витрачається на дії і порівняння його з передбачуваним. Після чого використавши експериментально-теоретичний метод дослідження, а саме аналіз отриманих даних, було визначено, що надмірно багато часу витрачається на передачу даних між учасниками мережі, в зв'язку з тим, що цей процес не автоматизований і немає спеціалізованого ПЗ.

При проектуванні і розробці даної системи було виявлено, що всі вже існуючі програмні засоби побудовані на одному з двох основних принципів таких як: клієнт-серверні програми та розподілені системи. Дана ж розробка об'єднує в собі плюси обох їх підходів, але при цьому усуває їх недоліки, так само додаючи деякі аспекти, які використовуються в інших галузях. Система авторизації користувачів всередині системи, в той час як в інших програмних засобах авторизація відбувається автоматично при запуску комп'ютера. А в даному випадку необхідно вводити логін і пароль, що допомагає додатково захистити систему від стороннього втручання. Другою відмітною особливістю є шифрування переданих файлів, що виключає можливість отримання їх сторонньою особою, навіть у разі перехоплення інформації. Останньою і головною особливістю

даного програмного забезпечення є можливість автоматично переходити з клієнт-серверного в розподілене, в разі відмови сервера.

Зв'язок роботи з науковими програмами, планами, темами. Магістерська робота виконувалася протягом 2016-2017 рр. згідно з планами науково-дослідних робіт кафедри комп'ютерних наук та інженерії в межах НДР «Методологія та інформаційна технологія управління техногенним ризиком об'єктів підвищеної небезпеки» (№ ДР 0115U004878).

Мета і задачі дослідження. Метою даної роботи є розробка програмного забезпечення, яке допоможе реалізувати безпечний обмін інформацією між різними відділами на ОП ІОЦ ВАТ «Лисичанськвугілля». Створити базу, в якій будуть зберігатися всі призначені для користувача дані необхідні для авторизації. А саме логін і пароль кожного учасника мережі. Крім цього так само в базі повинні міститися відомості про права доступу кожного з учасників до файлів. Розробити систему доступу, яка дозволяє визначати можливість користувачів отримувати права на перегляд, копіювання, зміну і видалення файлів загального доступу. Написати серверну частину програми, яка буде копіювати на сервер всі загальнодоступні файли, зберігаючи при цьому градацію з прав. А також реалізовувати обмін інформацією між учасниками навіть в разі відсутності власника файлів. Клієнтська частина, в свою чергу, повинна надати користувачеві можливість налаштувати права доступу до файлів, і вказати які саме файли, для яких учасників і з якими привілеями доступні. А в разі відсутності сервера або ж виходу його з ладу, переключити систему в розподілений режим. Що дозволить системі продовжити функціонування і коректний обмін інформацією.

Поставлена мета, визначені наступні завдання:

- дослідити предметну область;
- проаналізувати аналогічні програмні продукти;
- вибрати стратегію розробки;
- побудувати математичну модель;
- розробити програмний засіб на мовах програмування C++ та C#.

Об'єкт дослідження. Система документообігу промислового підприємства.

Предмет дослідження. Розподіл обчислювального навантаження між комп'ютерними компонентами мережесистемних обчислювальних систем.

Методи дослідження. Для вирішення поставлених завдань були застосовані наступні методи дослідження:

- Теоретичні (аналіз предметної області, збір необхідних даних для реалізації поставленої мети);
- емпіричні (вибір засобів розробки);

Наукова новизна одержаних результатів. Набули подальшого розвитку методи розробки систем для обміну інформацією, а саме покращення стабільності роботи за рахунок об'єднання режимів праці а також покращення захисту інформації, за рахунок передачі її по захищеному каналу.

Практичне значення одержаних результатів. На основі отриманих результатів було зроблено розробку програмного засобу, а також виявлено найпроблемніші ділянки у даній сфері. Дану розробку можна застосовувати на будь-яких підприємствах, а так само навіть для навчання в школах і інститутах. Адже без обміну інформацією в наш час практично не може обійтися не одна галузь, а в зв'язку з тим, що це ПЗ не є спеціалізованим, це відкриває широкий спектр його використання.

Публікації. По даній роботі було зроблено дві публікації у збірниках:

IT-Ідея – 2017: збірник науково-практичних праць – Северодонецьк. Програмний засіб для обміну інформацією «InfoNet».

Майбутній науковець – 2017” : науково-практична конференція. Східноукр. ун-ту ім. В. Даля. Програмні засоби обміну інформацією в розподіленій системі

Структура та обсяг магістерської роботи. Магістерська робота складається зі вступу, чотирьох розділів, висновків, списку використаних джерел та додатків. Робота викладена на 115 сторінках машинописного тексту, містить 91 сторінку основного тексту, 26 рисунків, 25 таблиць, додатки на 16 сторінках.

1 ОГЛЯД СУЧАСНИХ РОБІТ В СФЕРІ ДОКУМЕНТООБІГУ

У наш час завжди і всюди використовують документацію, але зараз вже сучасні технології дозволяють уникнути паперового обліку. Для цього були придумані і розроблені системи електронного документообігу.

1.1 Загальні відомості

Система автоматизації документообігу, система електронного документообігу - автоматизована розрахована на багато користувачів система, що супроводжує процес управління роботою ієрархічної організації з метою забезпечення виконання цієї організацією своїх функцій. При цьому передбачається, що процес управління спирається на людино-читаються документи, що містять інструкції для співробітників організації, необхідні до виконання.

Документообіг - рух документів в організації з моменту їх створення або отримання до завершення виконання або відправлення (ГОСТ Р 7.0.8-2013); комплекс робіт з документами: прийом, реєстрація, розсилка, контроль виконання, формування справ, зберігання і повторне використання документації, довідкова робота.

Електронний документообіг (ЕДО) - єдиний механізм по роботі з документами, представленими в електронному вигляді, з реалізацією концепції «безпаперового діловодства».

Електронний документ (ЕД) - документ, створений за допомогою засобів комп'ютерної обробки інформації, який може бути підписаний електронним підписом і збережений на машинному носії у вигляді файла відповідного формату.

Електронний підпис - аналог власноручного підпису, який є засобом захисту інформації, що забезпечує можливість контролю цілісності і підтвердження достовірності електронних документів.

1.2 Принципи документообігу

Основними принципами в будь-якій системі документообігу є:

- Одноразова реєстрація документа, що дозволяє однозначно ідентифікувати документ;
- Можливість паралельного виконання операцій, що дозволяє скоротити час руху документів і підвищення оперативності їх виконання;
- Безперервність руху документа, що дозволяє ідентифікувати відповідального за виконання документа (завдання) в кожен момент часу життя документа (процесу);
- Єдина (або узгоджена розподілена) база документної інформації, що дозволяє виключити можливість дублювання документів;
- Ефективно організована система пошуку документа, що дозволяє знаходити документ, володіючи мінімальною інформацією про нього;
- Розвинена система звітності по різних статусах і атрибутах документів, що дозволяє контролювати рух документів по процесах документообігу і приймати управлінські рішення, ґрунтуючись на даних зі звітів.

1.3 Категорії системи документообігу

Системи документообігу поділяються на 3 категорії:

1. Універсальні «коробкові» СЕДО:

- стандартний набір функцій;
- неможливість повної відповідності потребам конкретної організації;
- низькі витрати часу на придбання та установку;
- відносно низька вартість;
- необхідність придбання ліцензії на кожне впроваджуване робоче місце.

2. Індивідуально розробляються СЕДО:

- максимально персоніфікована система;
- великі витрати часу;

- висока вартість розробки;
- супутні витрати: витрати на навчання співробітників, покупку нового обладнання і програмного забезпечення.

3. Комбіновані СЕДО:

- базова платформа, до якої розробляються необхідні додаткові модулі;
- повну відповідність потребам підприємства;
- невеликі тимчасові витрати на розробку і впровадження;
- вартість включає: ціну базової платформи і вартість індивідуальної доопрацювання, залежить від складності замовлення;
- передача замовнику прав на продукт;
- простота освоєння і використання;
- повна локалізація;
- зручний інтерфейс;
- взаємодія з існуючими офісними додатками.

1.4 Мета СЕД

Основними цілями будь якої СЕД є:

- скорочення або повна відмова від паперового документообігу;
- створення єдиної інформаційної бази компанії;
- зниження ризику втрати документа;
- структурування всієї документації по затвердженій номенклатурі;
- підвищення дисципліни серед співробітників завдяки можливості відстеження діяльності виконавця конкретного документа;
- контроль над виконанням документів відповідно до резолюцій керівника;
- підвищення ефективності роботи компанії.

1.5 Види систем електронного документообігу

Хоча будь-яка класифікація є умовною, традиційно виділяють кілька видів СЕД по функціоналу і важливості справ:

- a) Системи діловодства. Вони використовуються в організаціях з жорстко формалізованими правилами документообігу і вертикальним керуванням незалежно від їх розміру, форми власності та виду діяльності. З їх допомогою можна вирішити кілька основних завдань:
 - упорядкувати роботу з кореспонденцією, як виходить, так і входить;
 - оптимально організувати рух внутрішніх документів в компанії;
 - налагодити роботу зі зверненнями клієнтів;
 - організувати внутрішній електронний архів документів.
- b) Електронні архіви. Це системи структурованого зберігання документів, що забезпечують надійність зберігання, розмежування прав доступу, зручний і швидкий пошук. Впровадження систем такого типу зазвичай це пов'язано з потребою структурування електронної інформації, незалежно від виду документів - текстовий, графіка, мультимедіа. Основні функції архіву - оцифровка паперових документів, управління web-контентом, потоковий ввід і швидкий пошук. Завдяки впровадженню архіву можна скоротити час доступу до інформації, знизити ризики псування або втрати важливих документів, підвищити рівні інформаційної безпеки. Електронні архіви зазвичай існують в складі СЕД і окремо використовуються досить рідко.
- c) Workflow-системи забезпечують автоматизацію не окремих функцій, а бізнес-процесів компанії. Дослівний переклад Workflow - «потік робіт». Система Workflow чітко визначає процес: що, хто, коли і як робить, звідки отримує і куди відправляє. Користувачеві не потрібно замислюватися над тим, як створити документ, як його отримати, як обробити, в які терміни і кому відправити - все вже закріплено в системі. Співробітник не зможе неправильно заповнити документ, пропустити якісь терміни, в системі передбачені нагадування, а також повідомлення керівника про те, що на конкретному етапі у конкретного користувача процес обробки документа порушений. Є можливість оцінити причини допущення помилок і своєчасно їх усунути. Workflow-системи в основному встановлюють в компаніях з високим ступенем формалізації бізнес-процесів, документообіг в яких при простій структурі має масовий характер.

Мінусом таких систем є складність і тривалість застосування. Крім того, вони не можуть замінити електронний архів, оскільки зберігають не всі документи, а тільки використовувані в процесі роботи.

- d) ЄСМ-системи використовуються для збору, управління, зберігання інформації (контенту) і надання до неї доступу співробітникам компанії, тобто, на перший погляд, виконують ті ж самі функції, що і попередні системи. Однак істотна відмінність ЄСМ-систем від своїх побратимів в тому, що вони дозволяють працювати як з структурованим, так і з неструктурованим контентом і мають більш гнучкий функціонал. По суті, забезпечення електронного документообігу є тільки однією з функцій ЄСМ разом з управлінням файлами і записами, управлінням знаннями, власне Workflow, і управлінням web-контентом. У деяких ЄСМ-системах також є функція CRM і управління завданнями і дорученнями.

При впровадженні СЕД на підприємство потенційні вигоди виходять наступними:

- a) Зниження матеріальних витрат:
- невелике підприємство - на 5%;
 - велике підприємство з декількома філіями - на 20%.
- b) Економія на базових процесах - вихідні і вхідні документи, організаційно-розпорядчий документообіг, контроль виконання доручень:
- невелика компанія - від 8 до 15%;
 - велике підприємство - до 50%.

Окремо розглядається процес узгодження і виконання доручень. Тут економія може скласти від 6% до 23% в залежності від структури і величини компанії і стилю її роботи. Чим більше формалізовані процеси в компанії, тим вище буде ефект.

- a) Економія на конкретних операціях, не прив'язаних до процесів, - пошук документів, забезпечення доступу до них і т.д. - від 3 до 24% в залежності від стилю роботи з документами і від організації системи їх зберігання. Якщо створити загальнодоступне сховище електронних документів з чітко прописаними регламентами роботи і зберігання, правами доступу до них, то ефект для компанії буде максимальним.
- b) Зниження ризиків. Цей ефект стосується стратегічних показників і найменше піддається формальному розрахунку. У деяких випадках СЕД дозволяє знизити ризики прострочення узгодження та укладення договорів до 60%.

Основними представниками систем документообігу є:

- Directum;
- ELMA;
- Optima WorkFlow;
- 1С: Документообіг 8.

1.5.1 Система документообігу Directum

1.5.1.1 Функціональність і можливості інтеграції

До функціональних переваг системи Directum відноситься не тільки розширений каталог бізнес-рішень. З функцій, які використовуються кожен день в роботі, можна виділити зручні нагадування, можливості настройки нумераторів і роботи зі штрих-кодovими позначками. Наявність предпросмотра вкладених файлів, файлових сховищ, великий вибір заповнених довідників в базовій конфігурації, інтерактивні майстри створення документів.

Негативні сторони цієї СЕД: відсутність посторінкового виведення об'єктів в опис документів і результатів пошуку, недоступність застосування додаткових стилів в опису документа.

Дані функції реалізуються в Docsvision. У середині головного вікна програми відображаються будь-які веб-сторінки і налаштовані звіти. Архівні документи переміщаються в окремі таблиці бази даних. Користувачі системи можуть вибирати варіант аутентифікації. Застосовується XSLT-перетворення до XML опису реєстраційної картки.

1.5.1.2 Інтерфейс

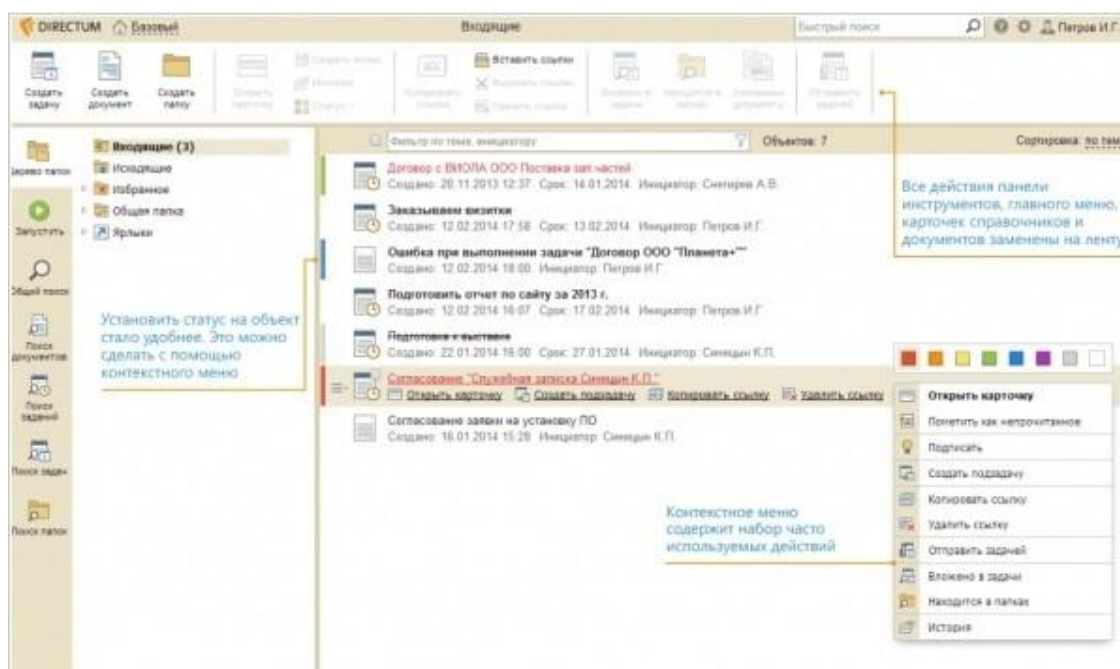


Рисунок 1.1 – Інтерфейс Directum

Інтерфейс системи документообігу Directum можна побачити на рис.1.1. Directum є ЄСМ-систему з можливостями управління документообігом, бізнес-процесами та web-контентом. Система включає в себе функції введення і перетворення документів, управління спільною роботою, довготривалого зберігання документів і забезпечення їх цілісності, доставки інформації. Базовий функціонал надається через web-сервер, для середнього бізнесу з підключенням від 50 до 200 користувачів розроблено хмарне рішення. З функцій, які використовуються кожен день в роботі, можна виділити зручні нагадування, можливості настройки нумераторів і роботи зі штрих-кодovими позначками. Наявність предпросмотра вкладених файлів, файлових сховищ, великий вибір заповнених довідників в базовій конфігурації, інтерактивні майстри створення документів.

Негативні сторони цієї СЕД: відсутність посторінкового виведення об'єктів і результатів пошуку, недоступність застосування додаткових стилів в опису документа.

1.5.1.3 Функції пошуку

До плюсів системи можна віднести наявність повнотекстового пошуку по документам і реквізитами реєстраційних карток (РК). Шаблони пошуку можна налаштувати за реквізитами РК. Діє фільтр опису документів за значеннями колонок. Зручні функції настройки шаблонів пошуків. Можна створювати пошукові папки.

Негативні якості: не можна розподілити права доступу до налаштованим пошуків; немає налаштованих пошуків в веб-клієнта; немає можливості створювати складні пошукові запити, немає посторінкового виведення об'єктів в опис документів і результатів пошуку, не застосовуються додаткові стилі в опису документа.

Функції пошуку системи документообігу можна побачити на рис.1.2.

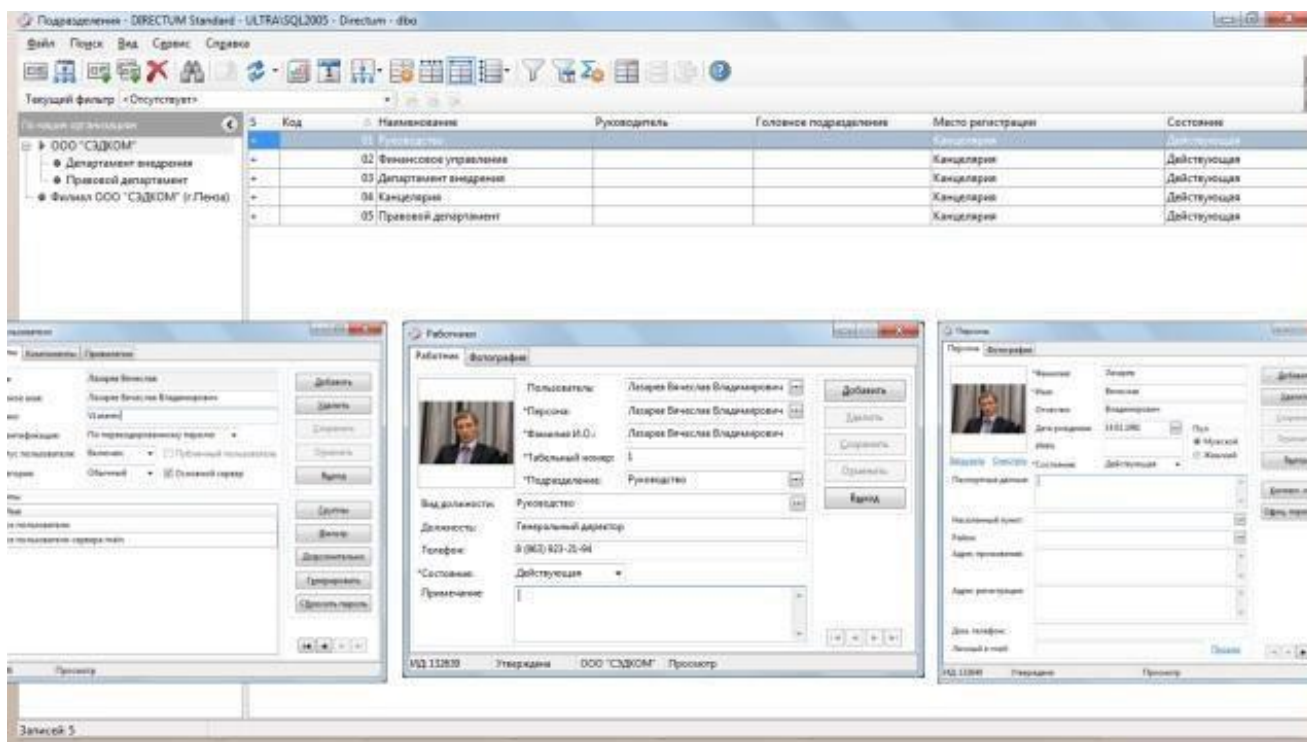


Рисунок 1.2 - Функції пошуку Directum

1.5.1.4 Адміністративні настройки

Система дозволяє одночасно вести організаційно-штатну структуру декількох компаній. Надається детальна інформація про підрозділи і співробітників, її можна розширювати. При необхідності заміщення (повного або асистентом) автоматично створюються папки «вхідні / вихідні». Працює фільтр по підрядку і біжить пошук.

Недоліки: відображення ОШС НЕ наочно, немає можливості сортування елементів в певному порядку, важко створювати і заповнювати довідники, немає локалізації записів довідників. Мінімальні налаштування зовнішнього вигляду і поведінки РК.

1.5.1.5 Безпека даних і дій

Захист даних в системі здійснюється за допомогою логіна-пароля і облікових записів Windows. Можна створювати групи і ролі користувачів. Є дискретна і рольова модель управління доступом, також є можливість видавати повноваження і привілеї. Розмежування прав доступу здійснюється на рівні папки, РК, файлу і довідника. Можна налаштувати передачу прав. Застосовується шифрування і ЕЦП.

Недоліки системи: немає розподілу прав доступу до налаштованим пошуків, немає налаштованих пошуків в веб-клієнті, не можна створювати складні пошукові запити.

Швидкодія панелей перегляду досить низька.

1.5.2 Система документобігу ELMA

1.5.2.1 Функціональність і можливості інтеграції

Систему ELMA відрізняє розширена лінійка бізнес-рішень. У їх числі модуль ЕСМ + (поєднання функцій ВРМ-системи і класичної СЕД), Проекти + - стандартні функції для управління проектами плюс управління процесами; ELMA CRM + - реалізує клієнт-орієнтований підхід через реалізацію наскрізних процесів. І КРІ-рішення, якого немає у системи Directum.

До питання про можливості інтеграції - плагін для роботи з MS Outlook є і у ELMA ЕСМ +, і у двох інших рішень. Крім того, ELMA легко інтегрується з «1С: Підприємство» (з усіма актуальними версіями).

Щоб не плутатися в великій кількості файлів з поправками і змінами, документи в додатку можуть мати необмежену кількість версій, але тільки одну можна призначити актуальною. Вся історія змін документа збирається на картку документа.

Всі необхідні інструменти для роботи з документом і контролю виконавської дисципліни по роботі з цими документами знаходяться у користувача під рукою.

Якщо документ надійшов у компанію в паперовому вигляді, він автоматично зберігається після сканування.

Керівник може переглядати звіт щодо поточної ситуації з виконавською діяльністю, а користувачам надано ряд фільтрів, щоб формувати результати так, як вони вважають за потрібне.

Додаток ЕСМ + можна налаштувати довільно відповідно до потреб окремого підприємства і його організаційної структури.

Для прискорення створення документів можна створювати шаблони. Це доступно для будь-якого користувача, навіть не має навичок в програмуванні.

Ви можете встановити автоматичний внесення штрих-коду в шаблон для моментального розпізнавання в системі.

У всіх розглянутих в огляді систем є свій мобільний додаток для роботи з функціями системи через планшетний комп'ютер або мобільний пристрій. Це дозволяє топ-менеджерам приймати участь в оперативній діяльності компанії з будь-якої точки, де є інтернет. Клієнти для мобільних пристроїв сумісні з усіма мобільними ОС.

1.5.2.2 Інтерфейс

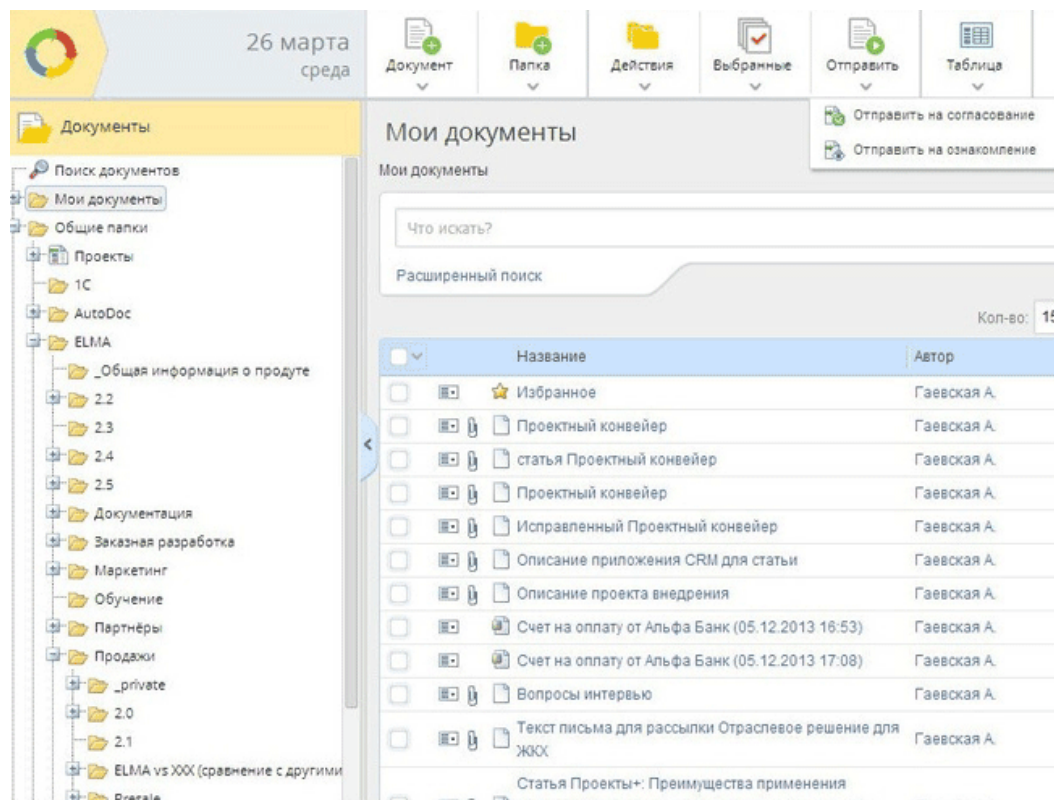


Рисунок 1.3 – Интерфейс ELMA

Інтерфейс системі документообігу ELMA можна побачити на рис.1.3. Зовні інтерфейс ELMA ECM + являє собою стандартне файлове сховище, користувач без досвіду роботи з такими системами цілком може зорієнтуватися самостійно.

Реєстрація документів реалізується в спеціальному майстра в кілька послідовних кроків. При введенні в систему потрібно лише вказати необхідні реквізити документа і прикріпити електронну версію, система автоматично генерує картку документа з усіма даними по ньому.

Вся інформація по організації міститься у внутрішньому порталі. Всі необхідні для роботи інструменти знаходяться в одному місці: архів документів, завдання, календар, стрічка повідомлень, звіти, веб-документи.

Недоліки: немає можливості налаштувати доступ до документів за ролями, а так само суто локальне сховище даних.

1.5.2.3 Функції пошуку

У ELMA ECM + документи збираються в єдиний електронний архів. Пошук документів здійснюється швидко і зручно не тільки за назвами, а й за реквізитами картки документа або РК. Результати пошуку відображаються тільки при наявності прав доступу до документів. Параметри пошуку зберігаються в форматі фільтра, в майбутньому можна швидко шукати документи по збереженим параметрам.

Недолік: відсутність повнотекстового пошуку в документах.

1.5.2.4 Адміністративні настройки

Зручно стежити за ситуацією в філіях великих компаній. Кілька систем об'єднуються в одному вікні, організовуючи єдину інформаційну систему підрозділів і якісний обмін інформацією. Користувач володіє всією необхідною інформацією про зовнішній системі, не покидаючи свою сторінку. Зручно перемикатися між сторінками підрозділів компанії. Крім того, доступна інформація по виконавської діяльності, актуальні новини, перегляд документів, запуск виконання бізнес-процесів.

Загальна інформація по всіх філіях і підрозділах доступна керівникові. Дані про підрозділи і співробітників можна розширювати.

Заміщення співробітників здійснюються в спеціальному розділі. Заміщення знімаються автоматично по завершенні терміну. При установці заміщає завдання по процесам автоматично переносяться на нього. Система повідомляє про разову завдання і про відсутність даного співробітника, пропонуючи вибрати іншого виконавця.

Адміністративні настройки системи ELMA можна побачити на рис.1.4.

The screenshot displays the 'Назначить замещение' (Assign replacement) form in the ELMA system. The interface is in Russian. On the left, there is a sidebar with navigation options: Главная, Администрирование, Настройки, Пользователи, Замещения, График отпусков, Группы, Настройки доступа, Портал, and Инф. каналы. The main form area has a title bar with 'Сохранить' (Save) and 'Отмена' (Cancel) buttons. The form fields are as follows:

- Замещаемый ***: Иванов Иван (Сотрудник отдела Админ)
- Замещающий ***: Петрова Дарья (Бухгалтерия)
- Дата начала ***: 06.03.2012
- Дата завершения ***: 08.03.2012
- Дублировать сообщения**: Нет Да. Подпись: *Дублировать или нет сообщения Замещаемому, если "Да"*
- Предварительное замещение ***: 0. Подпись: *Количество рабочих дней, за которое требуется выполнять т.п.) до фактической даты начала замещения*

Рисунок 1.4 -Адміністративні настройки ELMA

У спеціальному розділі «Графік відсутностей» виводиться список співробітників, яких немає на робочому місці протягом певного терміну.

1.5.2.5 Безпека даних і дій

- Будь-який користувач з будь-якого місця, де є інтернет, може ввести свій логін і пароль і зайти в систему ELMA. Такий вхід в систему налаштований за замовчуванням для всіх користувачів, але все є три основні способи, як захистити вхід в систему:
- Вхід тільки через довірені пристрою. Якщо вхід здійснюється з комп'ютера або мобільного пристрою, якого немає в списку довірених, то система повідомить про неможливість входу.
- Вхід по токени. Токен - пристрій, який прив'язується до облікового запису користувача і буде відповідати тільки йому. Плюс до всього, у окремого користувача є свій пін-код. На токен може бути записано відразу кілька електронних ключів-контейнерів - для входу в різні системи, а не тільки в ELMA, щоб користувачеві не носити цілу в'язку ключів.

- Вхід по Сертифікату. Механізм здійснюється приблизно так само. Користувач прив'язує сертифікат до свого облікового запису. Зайти в систему за сертифікатом можна тільки з певного робочого місця. Така можливість є тільки в браузерях Mozilla Firefox і Internet Explorer.

Також авторизація в ELMA може відбуватися за технологією Single Sign On (SSO) - для входу в систему потрібні ім'я та пароль, з якими користувач увійшов в ОС Windows.

Для узгодження, підписання документа або його версії використовується електронно-цифровий підпис. При цьому жоден інший користувач, навіть дізнавшись пароль, не зможе узгодити або підписати документ.

У ELMA доступна настройка прав доступу на різні об'єкти системи. В основі управління правами - ролі елементів оргструктури і окремих користувачів системи, процес настройки полягає в графічному відображенні посад, відділів і підрозділів, і розподіл відповідальності

1.5.3 Система документообігу Optima WorkFlow

1.5.3.1 Функціональність і можливості інтеграції

До переваг системи можна віднести можливість застосування додаткових стилів в опису документів, висновок зображень в опис документів і область перегляду. Можлива фільтрація завантаженої опису документів. У Optima WorkFlow підтримується настройка рубрикаторів в журналах документів, настройка області перегляду обраної записи в опису документів. Працює кастомизація реєстраційної картки доручення. Сторонні веб-додатки можуть вбудовуватися в головне вікно. Є інтеграція з IBM WebSphere MQ. Для автоматизації діяльності архівних підрозділів розроблений спеціальний модуль Архив.

Недоліки: копіювати посилання на об'єкти не можна. Немає можливості створювати папки користувача.

1.5.3.2 Інтерфейс

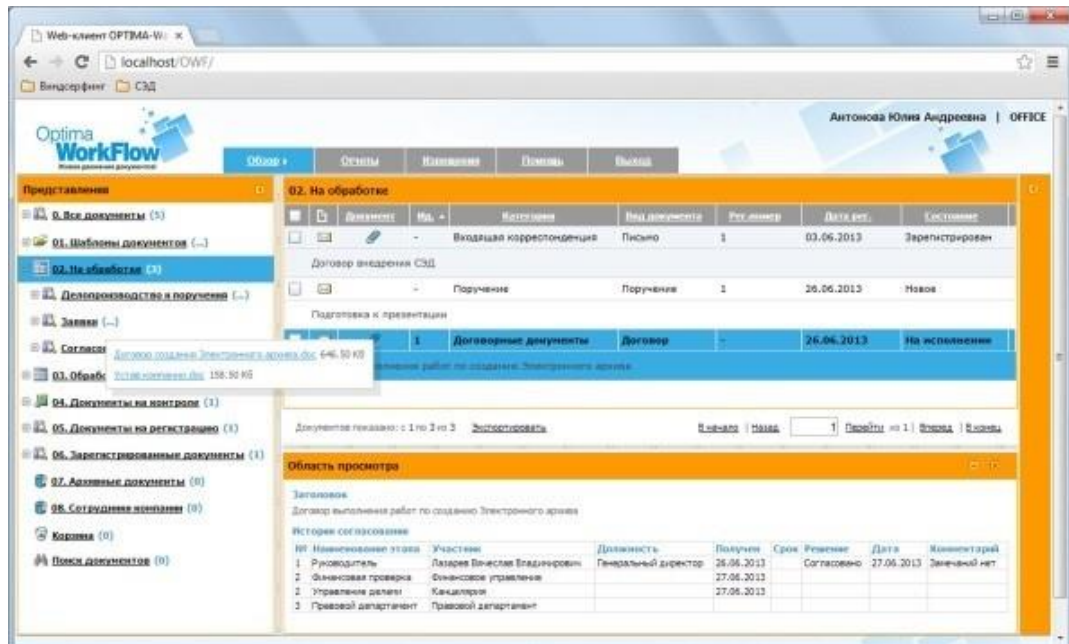


Рисунок 1.5 – Интерфейс Optima Workflow

Інтерфейс Optima Workflow можна побачити на рис. 1.5. Система має інтуїтивно зрозумілим інтерфейсом. В системі налаштовуються стилі оформлення додатка, доступний висновок в опис документів і область перегляду реквізитів і зображень, застосовуються стилі відтворення списку документів (шрифт, заливка, піктограма).

Варто відзначити зручні пошуки в полях-списках реєстраційних карток, наявність візуального контролю обов'язковості заповнення полів, можливість роботи з файлами, зв'язками і рухом документа в єдиному вікні.

Зручно працювати з прикріпленими файлами - робота здійснюється онлайн в єдиному вікні, в режимі check-in & check-out. Наскрізні зв'язку між документами візуалізуються у вигляді дерева.

Недоліки: користувачі не можуть створювати нові журнали, добірки документів, настройки не зберігаються в базу даних. У реєстраційній картці можна налаштувати панель інструментів, локалізувати назви реквізитів. Розмір картки незмінний. Чи не реалізовані механізми drag & drop. Ні базових функцій інтерфейсного вибору карток для зв'язку. Ще один недолік - непрозорість процесу при створенні завдання, для виконавців не можна налаштувати нагадування.

1.5.3.3 Функції пошуку

Застосовується прямий запит до бази даних для формування складного пошукового запиту в журналах документів. Здійснюється повнотекстовий пошук за змістом документа і пошук за зразком реєстраційної картки (не треба налаштовувати шаблони пошуку). Можна вводити пошукові запити за реквізитами реєстраційних карток. Застосовуються накладені фільтри до всіх доступних журналам документів.

Недолік системи в неможливості збереження користувальницьких пошукових шаблонів. Немає повнотекстового пошуку за реквізитами реєстраційних карток документів. Його користувачі не можуть створювати пошукові папки і складні пошукові запити. Відсутня повнотекстовий пошук за реквізитами реєстраційних карток.

1.5.3.4 Адміністративні налаштування

Система підтримує ведення ОШС декількох компаній, структура наочно відображається. Реалізуються функції заміщення (постійний / тимчасовий заступник), є біжить пошук за першими літерами. Елементи можна сортувати в заданому порядку; досить просте заповнення довідника, створення ієрархічних довідників, установка терміну дії запису довідника.

Є дизайнер шаблонів реєстраційних карток, налаштовуються розміри, фон, закладки, поля, шрифти, застосовуються скриптові сценарії щодо подій реєстраційної картки. Адміністративні налаштування Optima WorkFlow можна побачити на рис. 1.6.

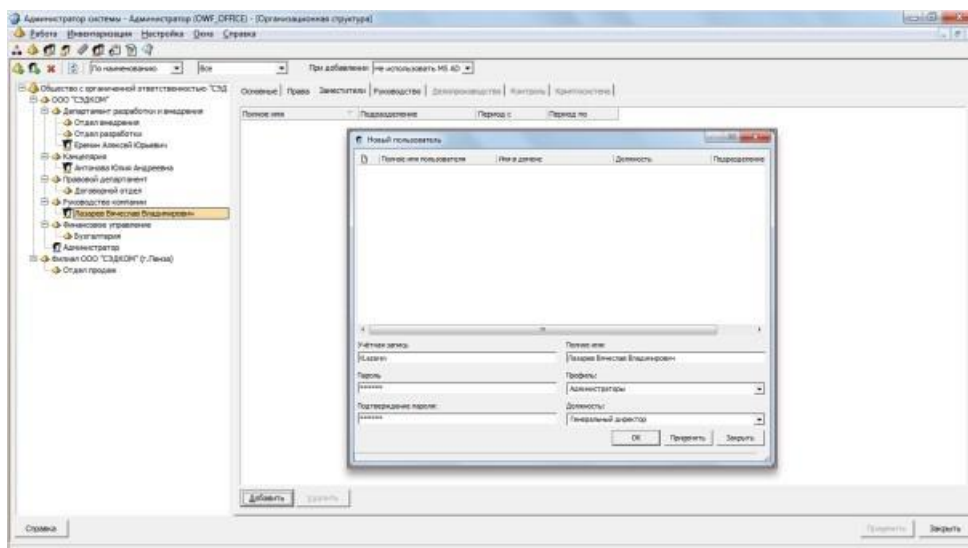


Рисунок 1.6 - Адміністративні налаштування Optima WorkFlow

Недоліки: немає можливості ведення історії довідника, відсутня дзеркальна інформація про заміщення. Інформація про підрозділи і співробітників мінімальна. Не можна створювати типізовані довідники, вести історію довідників, локалізувати записи довідників. Відсутня функція перевірки синтаксису коду скриптових сценаріїв, немає полів типу «радіокнопка».

1.5.3.5 Безпека даних і дій

Захист даних здійснюється за допомогою логіна / пароля, облікового запису Windows. Створюються групи і ролі користувачів, дискретна і рольова модель управління доступом. Є можливість видачі повноважень, розмежування прав доступу на рівні папки, настройка відносних прав доступу по оргштатної структурі організації, застосування ЕЦП.

До окремого файлу не можна розподілити права доступу. Відсутня попередній перегляд файлів.

1.5.4 Система документообігу 1С: Документооборот 8

1.5.4.1 Функціональність і можливості інтеграції

Основні напрямки роботи даного рішення - діловодство, загальний документообіг, управління договірною діяльністю, електронний архів і робота зі зверненнями. Функціонал можна розширювати за допомогою плагінів. При обробці маршрутів документів можна прикріплювати файли і згодом змінювати їх. Є реєстрація Провідника Windows.

Ви можете встановити автоматичний створення, реєстрацію та відправку за маршрутом документів з різних джерел (папка, електронну поштову скриньку, web).

Недоліки: відсутня внутрішня пошта. Не можна створювати завдання для групи і папки з дорученнями. Крім того, у даного рішення немає модуля управління проектами. Неможлива реєстрація з MS Office, Open Office. Чи не підтримується робота з документацією СУЯ. Також немає можливості вести особисті папки користувача.

1.5.4.2 Интерфейс

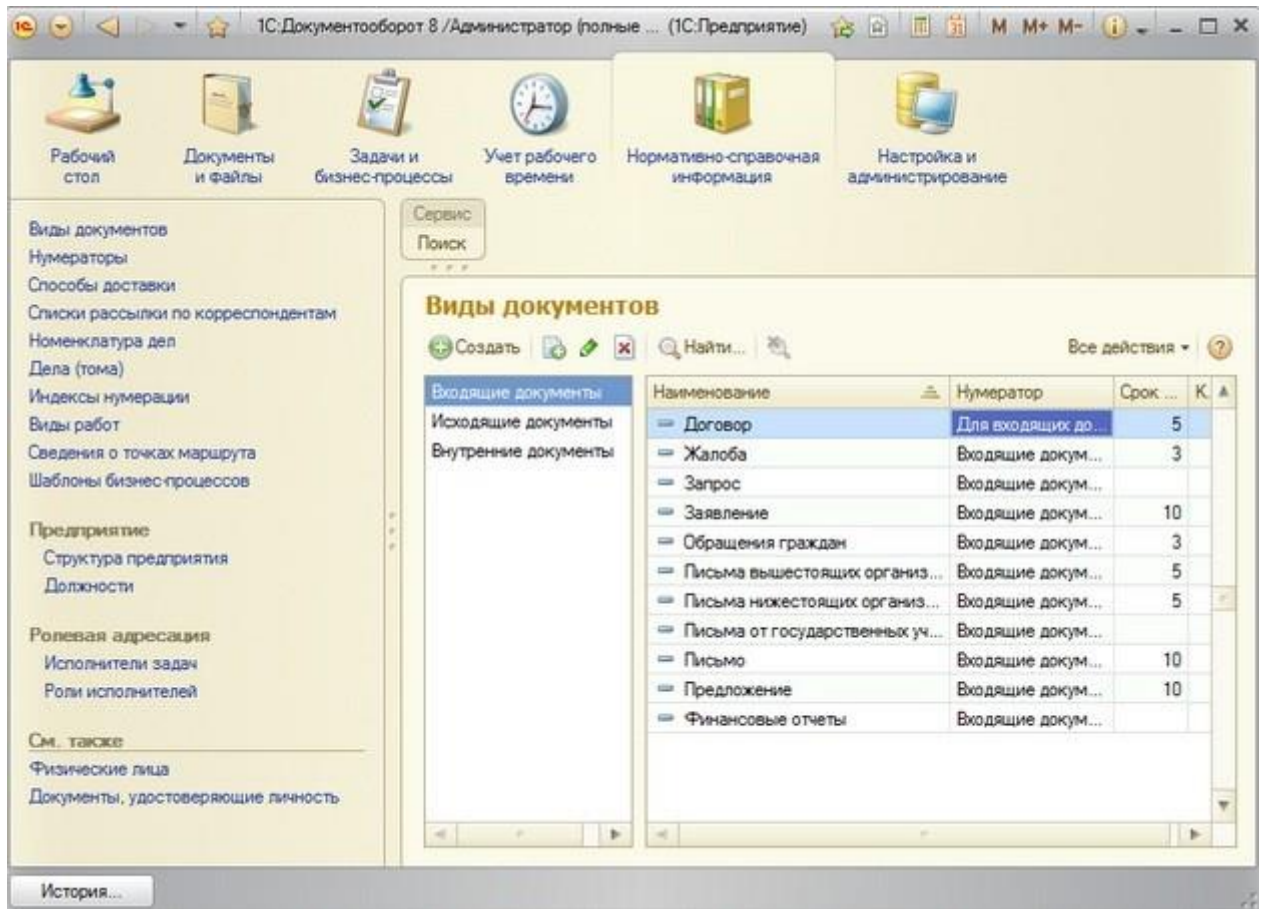


Рисунок 1.7 – Интерфейс 1С: Документооборот 8

Интерфейс 1С: Документооборот 8 можна побачити на рис. 1.8. Інтерфейс налаштовується під типи користувачів. Ви можете переглянути список працюючих в системі користувачів. Є засоби проектування РКК (дизайнер), можливість розширення та налаштування РКК, при цьому можна використовувати складні елементи - таблиці, структури.

Недоліки: немає вбудованих засобів перегляду приєднаних фото, немає перегляду завдань групи, призначення виконавця керівником з групи.

1.5.4.3 Функції пошуку

Здійснюється атрибутивний пошук за реквізитами документа. Можна створювати шаблони пошуку, їх зберігання є ієрархічним і класифікованих. За результатами пошуку проводиться побудова звітів.

В одному запиті можна здійснювати різні види пошуку. Немає настройки набору реквізитів для автоматичного пошуку дублікатів документа при реєстрації.

1.5.4.4 Адміністративні настройки

Робочий стіл користувача налаштовується відповідно до його роллю. Забезпечується швидкий доступ до даних про співробітників, списку телефонів, рекомендацій по складанню кореспонденції та ін. Для керівника відділу можна налаштувати панель повнотекстового пошуку документації будь-якого призначення, керівнику надається панель швидкого доступу до звітів. Інформація розподіляється відповідно до прав доступу. Організаційна структура організації і ролі виконавців можуть використовуватися для вибору виконавців бізнес-процесів.

Підсистема «Завдання і процеси співробітників» містить засоби створення і управління бізнес-процесами, а також налаштування рольової адресації завдань, налаштування матриці заміни / заміщення співробітників.

1.5.4.5 Безпека даних і дій

Є програмні засоби контролю цілісності документів, підтримка різних способів аутентифікації. Користувачеві видаються права на час виконання доручення або документа. Підписані ЕЦП документи відправляються зі збереженням інформації про підписи. Права доступу призначаються відповідно до посади користувача, його роллю, ступенем участі в бізнес-процесах, грифом або індивідуально. Права встановлюються до папок і видам документів і також поширюються на прикріплені файли, вони можуть містити в собі обмеження на читання або на запис. Доступ до звітів також обмежений - співробітнику доступні тільки його звіти, керівнику - звіти свого підрозділу.

Немає розмежування прав доступу на частини документа (поля реєстраційної форми, приєднані файли). Немає системи засобів резервного копіювання і відновлення даних.

Основним підходам даних систем є клієнт-серверне орієнтування. Користувач створює документ на своєму комп'ютері без підключення до сервера, а після його створення або обробки файл надсилається на сервер. Або ж файли відразу створюються на сервері, тим самим позбавляючи учасників від необхідності після створення або обробки робити їх пересилання. І другий варіант підходів може бути розподілена мережа, в якій у кожного з учасників на його комп'ютері зберігатися частина інформації для роботи всієї системи.

1.6 Висновки до розділу 1

Після аналізу існуючого програмного забезпечення та систем, а також методів підходу до вирішення даної проблеми залишилися не вирішеними такі моменти:

- Додатковий захист документів при обміні
- Авторизація користувачів всередині мережі
- Розробка зручного інтерфейсу для даної галузі
- Працездатність системи в разі часткової відмови

Основними завданнями на дану роботу буде:

- Проаналізувати існуючі програмні забезпечення і виявити їх плюси і мінуси
- Розробити власну систему, яка буде враховувати ці недоліки і компенсувати їх
- Розробити серверну частину, що відповідає за зв'язок учасників між собою, авторизацію користувачів, а так само зберігання і передачу даних
- Розробити клієнтську частину, яка зв'язується з сервером для обміну та авторизації, а в разі його виходу з ладу перемикає систему в розподілене режим
- Захистити документи, при передачі використовуючи додаткові методи кодування
- Протестувати ПО на працездатність і налагодити.
- Впровадити систему на підприємства для отримання конкретних даних і результатів

2 АНАЛІЗ СИСТЕМИ ОБМІНУ ІНФОРМАЦІЄЮ НА ПІДПРИЄМСТВІ

На даний момент існує проблема обміну інформацією на ОП ІОЦ ПАТ «Лисичанськвугілля». А саме, при передачі інформації від одного відділу до іншого: через стандартні засоби Windows, або ж з використанням зовнішніх носіїв, дуже часто виникає ситуація втрати інформації. Або ж потрапляння її до сторонніх особистостям. А так же в зв'язку з нестабільністю роботи системи обміну інформацією виникають ситуації, коли обмін неможливий.

2.1 Стабільність роботи

Для вирішення даної проблеми були проведені дослідження в даній області. Насамперед був проведений статистичний збір і аналіз даних про стабільність роботи самої системи за період 2013-2017 рік.

$$S = \frac{T_{\text{общ}}}{T_{\text{кор}}} \quad (2.1)$$

Де стабільність роботи (S) визначається як відношення загального часу роботи ($T_{\text{общ}}$) системи, до коректного часу роботи ($T_{\text{кор}}$).

Стабільність роботи за 2013 – 2017 р. можна побачити на рис. 2.1 або у табл. 2.1.

Таблиця. 2.1 – Стабільність роботи 2013-2017 р.

Рік	% Стабільності
2013	93
2014	85
2015	83
2016	87
2017	76

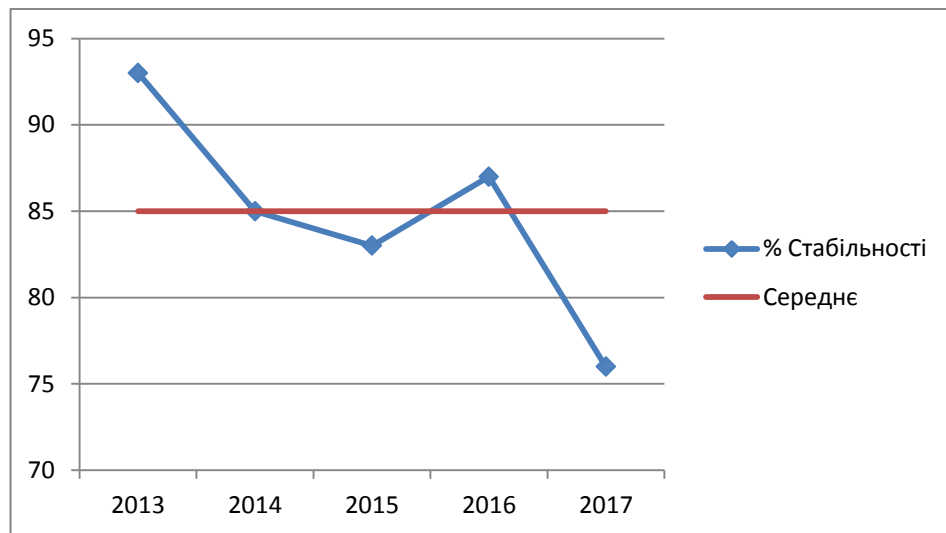


Рисунок. 2.1 – Стабільність роботи 2013 – 2017 р.

З даного графіка видно, що тенденція стабільності з кожним роком стає все менше. Це викликано збільшенням обсягу роботи і вимогами до апаратних засобів. Стабільність роботи за 2017 р. можна побачити на рис. 2.2 або у табл. 2.2.

Таблиця 2.2 – Стабільність роботи 2017 р.

2017	% Стабільності
Січень	80
Лютий	73
Березень	82
Квітень	79
Травень	74
Червень	70
Липень	81
Серпень	77
Вересень	77
Жовтень	70
Листопад	77
Грудень	72

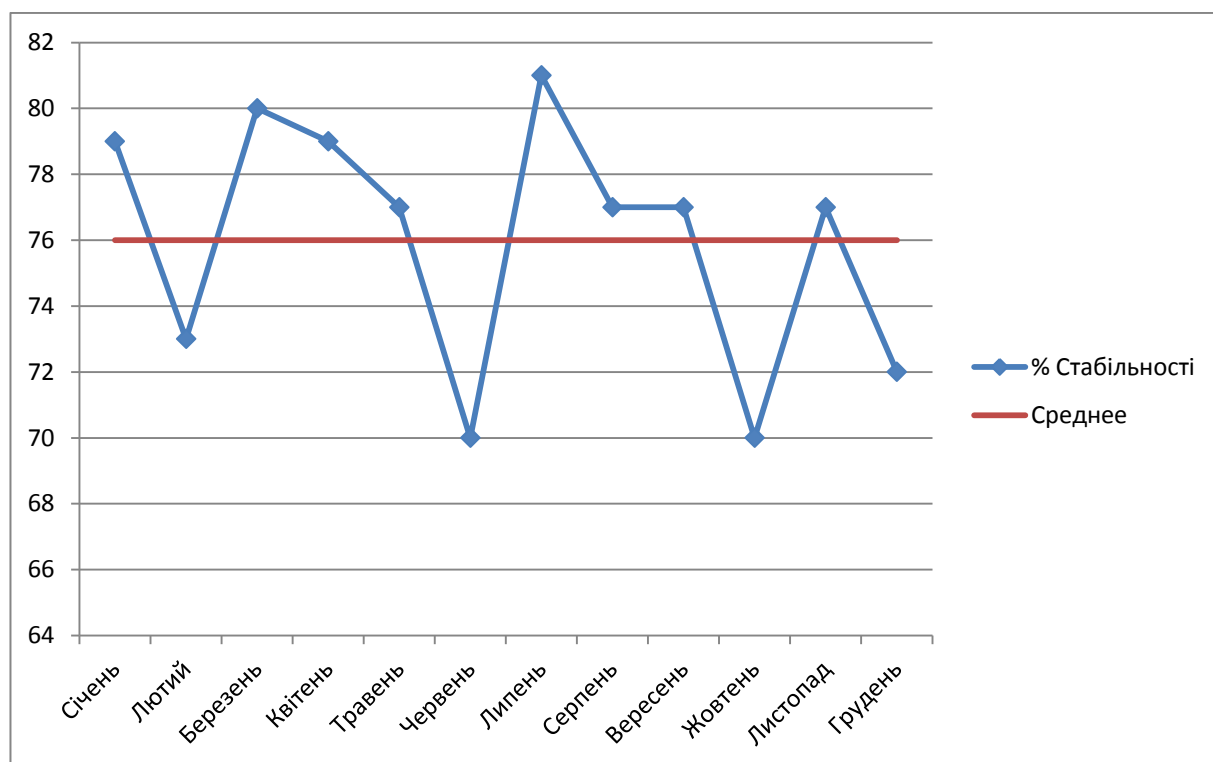


Рисунок. 2.2 – Стабільність роботи 2017 р.

Стабільність роботи системи в 2017 році, як видно на графіку, має 4 значення, які менше середнього. Ці значення якраз припадають на період квартальних звітів організації. У свою чергу це свідчить про те, що програма виходить з ладу не випадково, а терпить сторонні втручання.

2.2 Робота з програмою

У табл. 2.3 або на рис. 2.3. можна побачити відсоток часу яких працівник проводить за роботою з програмою у 2013 – 2017 р.

Таблиця 2.3 – Робота з програмою 2013 – 2017 р.

	% Роботи з програмою	% Роботи без програми
2013	73	27
2014	75	25
2015	78	22
2016	80	20
2017	81	19

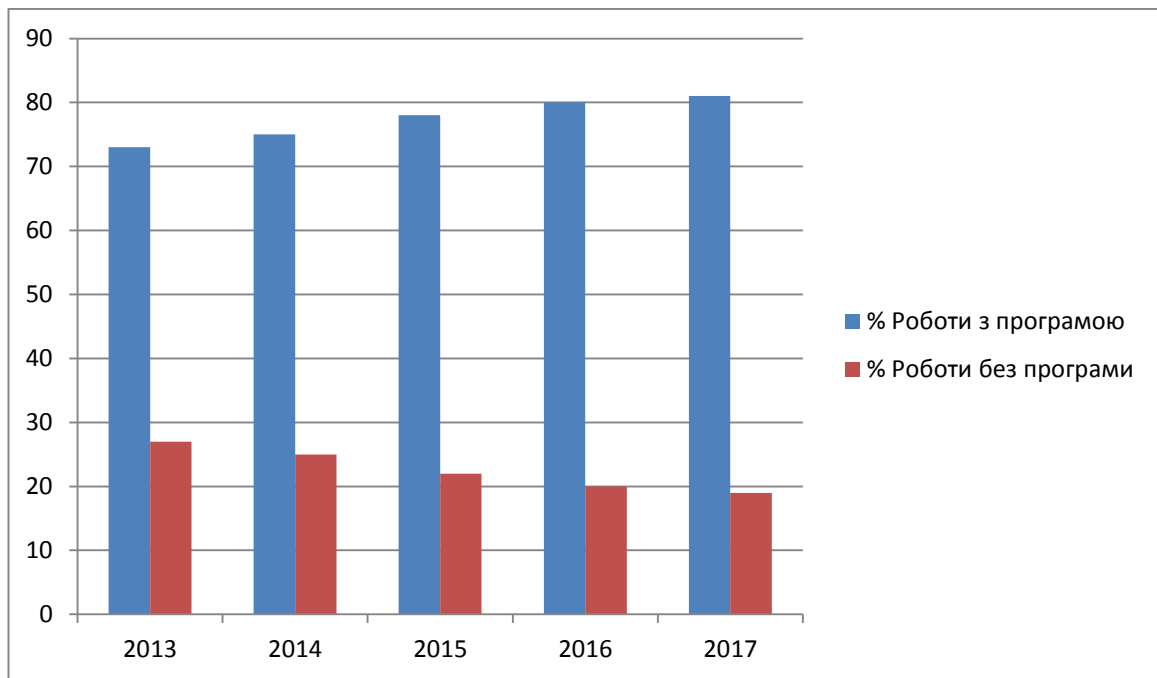


Рисунок. 2.3 – Робота з програмою 2013 – 2017 р.

З кожним роком, вплив програмного забезпечення росте. Що веде до залежності від нього, і в свою чергу вимагає підвищеної уваги до нього.

Таблиця 2.4 – Робота з програмою 2017 р.

2017	% Роботи з програмою	% Роботи без програми
Січень	82	18
Лютий	89	11
Березень	79	21
Квітень	74	26
Травень	78	22
Червень	90	10
Липень	76	24
Серпень	74	26
Вересень	78	22
Жовтень	91	9
Листопад	72	28
Грудень	89	11

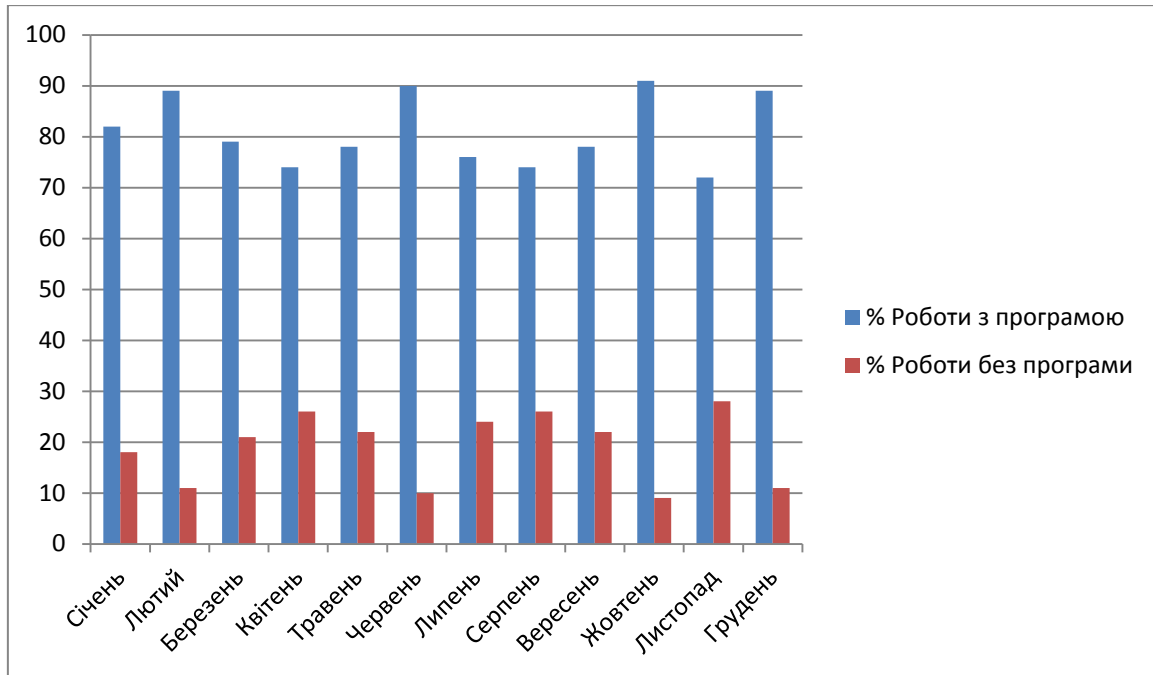


Рисунок. 2.4 – Робота з програмою 2017 р.

У табл. 2.4 та на рис. 2.4 представлено відсоток часу яких працівник проводить за роботою з програмою у 2017 р.

Кількість часу, який людина проводить при роботі з програмою, як видно з графіка, найбільше в кварталні звіти і останній місяць в році. Звідси впливає висновок, що саме в ці періоди ККД співробітників найбільше залежить від стабільності роботи програми.

2.3 Коефіцієнт корисної дії

Зменшення стабільності роботи програми, в свою чергу веде до зниження ККД працівників. Яке можна обчислити за формулою:

$$\eta = T_{\text{пр}} * S + T_{\text{бпр}} \quad (2.2)$$

Де ККД (η) визначається як добуток відсотку часу роботи з програмою ($T_{\text{пр}}$) помножене на стабільність її роботи, плюс час роботи без програми ($T_{\text{бпр}}$).

У таблиці 2.5 а також на рисунку 2.5 представлено ККД у 2013-2017 р.

Таблиця 2.5 - Коефіцієнт корисної дії 2013 - 2017 р.

Рік	ККД
2013	0,9489
2014	0,8875
2015	0,8674
2016	0,896
2017	0,8056

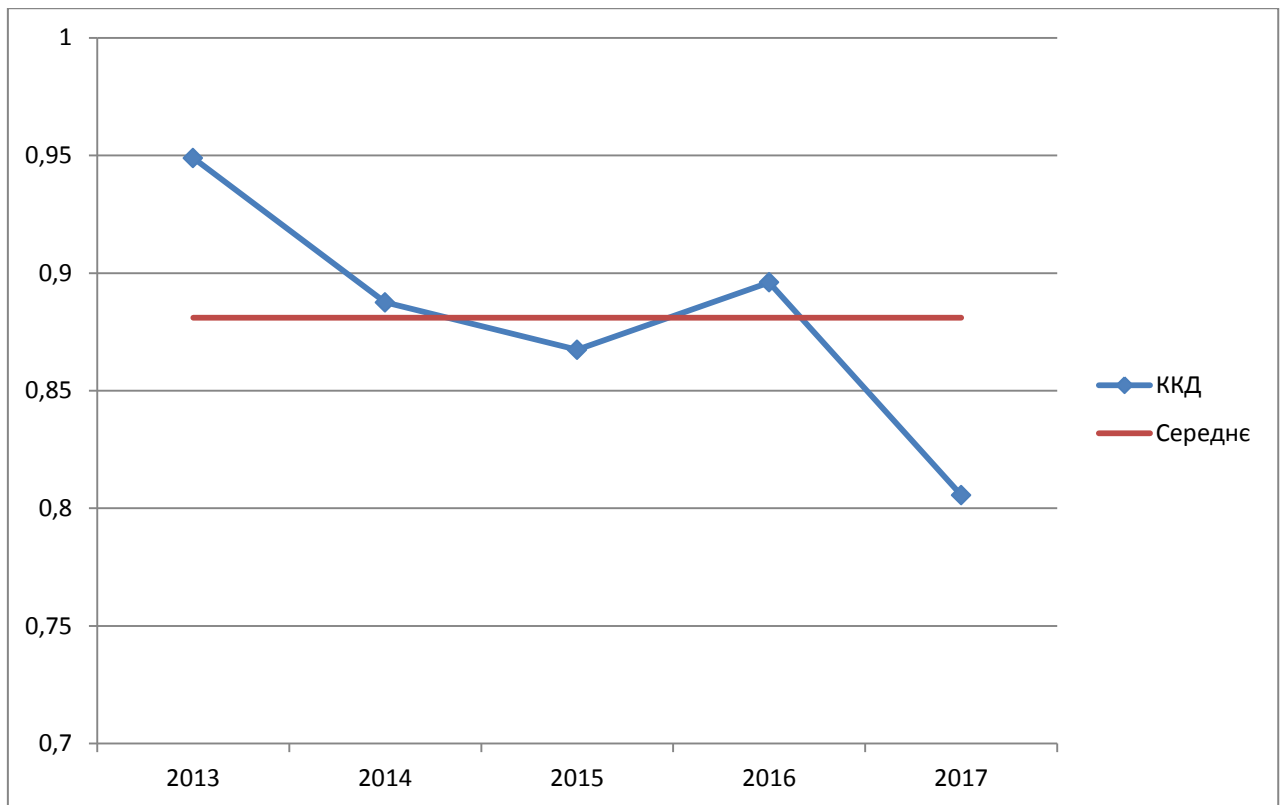


Рисунок. 2.5 - Коефіцієнт корисної дії 2013 - 2017 р.

На графіку видно, що ККД з кожним роком падає і падає, це пов'язано зі стабільністю роботи системи і збільшенням часу роботи з нею.

Таблиця 2.6 - Коефіцієнт корисної дії 2013 р.

2017	ККД
Січень	0,836
Лютий	0,7597
Березень	0,8578
Квітень	0,8446
Травень	0,7972
Червень	0,73
Липень	0,8556
Серпень	0,8298
Вересень	0,8206
Жовтень	0,727
Листопад	0,8344
Грудень	0,7508

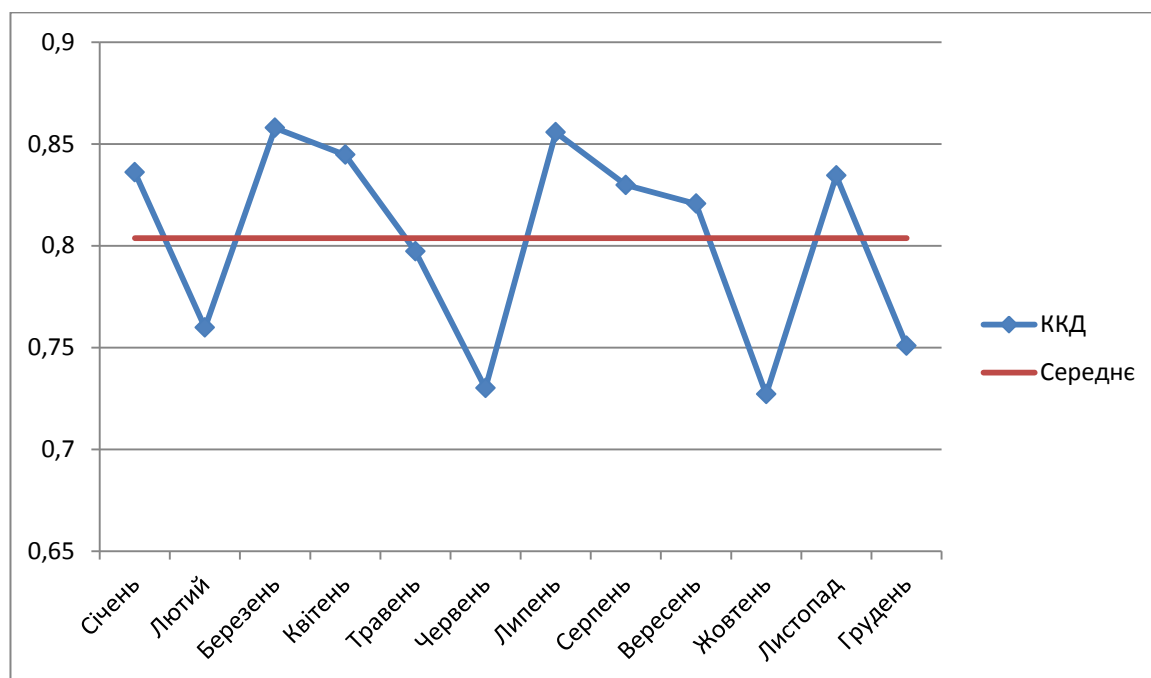


Рисунок. 2.6 - Коефіцієнт корисної дії 2013 р.

2.4 Підвищення стабільності

Для підвищення стабільності системи, необхідно або покращувати продуктивність комп'ютерів, що буває вкрай витратна, або ж модернізацією програми, що призведе до зменшення системних вимог. На Рис. 2.7 можна побачити графік який демонструє різницю вартості програми та вартості обладнання.

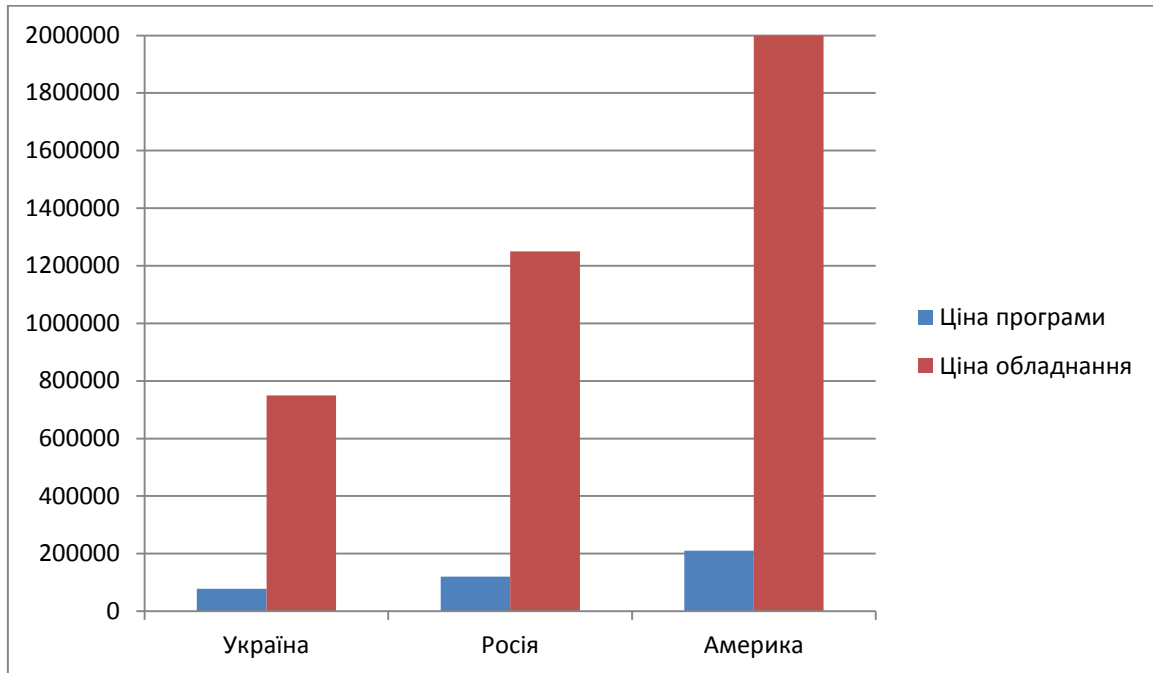


Рисунок. 2.7 – Підвищення стабільності

Чітко видно, що модернізувати програму набагато дешевше, ніж купити нове обладнання. З цього випливає висновок, що купуючи нове програмне забезпечення, компанія економить безліч ресурсів.

2.5 Робота з інформацією

Так само був проведений підрахунок відсотків зміненої, втраченої і розголошеної інформації по відношенню до всієї. Табл. 2.7 та Рис. 2.8 наглядно демонструє який відсоток інформації було пошкоджено за 2013- 2017 р.

Таблиця 2.7 – Робота з інформацією 2013 – 2017 р.

Рік	% Змін	% Втрати	% Витік
2013	5	12	8
2014	7	13	11
2015	7	11	15
2016	8	15	19
2017	10	17	21

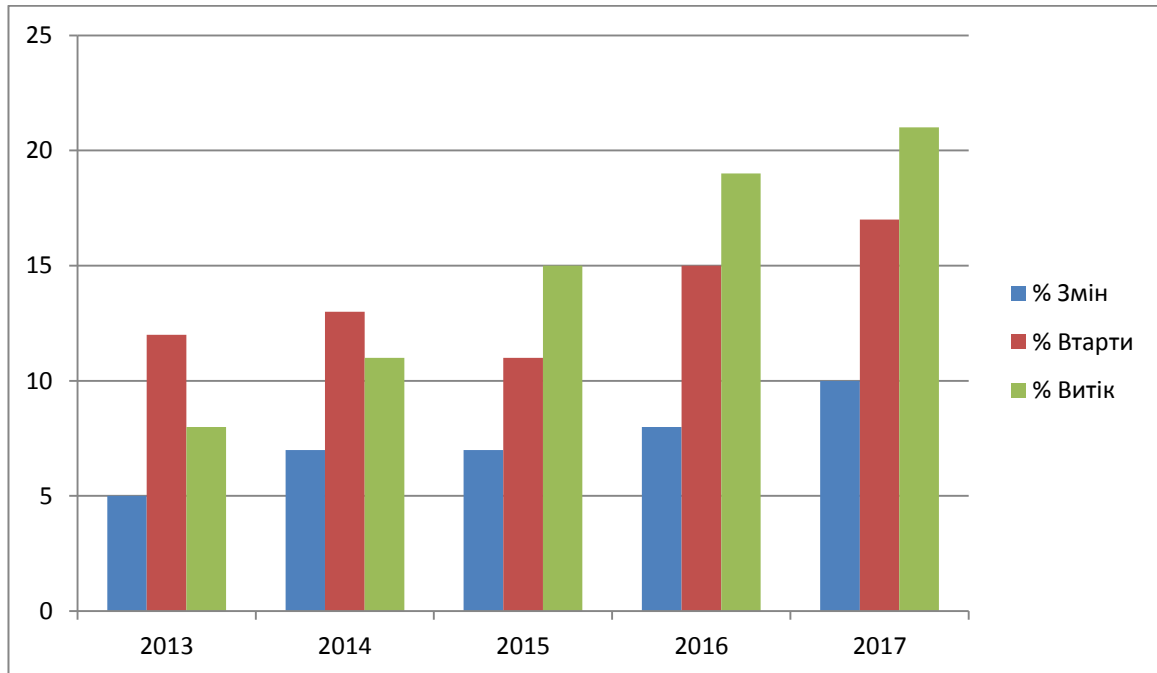


Рисунок. 2.8 – Робота з інформацією 2013 – 2017 р.

З кожним роком процентне співвідношення між всією інформацією та інформацією, яка потрапила в чужі руки, росте. Що свідчить про те, що безпека і конфіденційність даних з кожним роком все гірше, і це викликано сторонніми причинами.

Таблиця 2.8 – Робота з інформацією 2017 р.

2017	% Зміни	% Втрати	% Витік
Січень	15	15	21
Лютий	9	21	25
Березень	11	17	18
Квітень	9	19	19
Травень	9	14	17
Червень	8	13	19
Липень	13	15	20
Серпень	11	13	23
Вересень	9	12	19
Жовтень	10	25	27
Листопад	7	17	21
Грудень	9	23	23

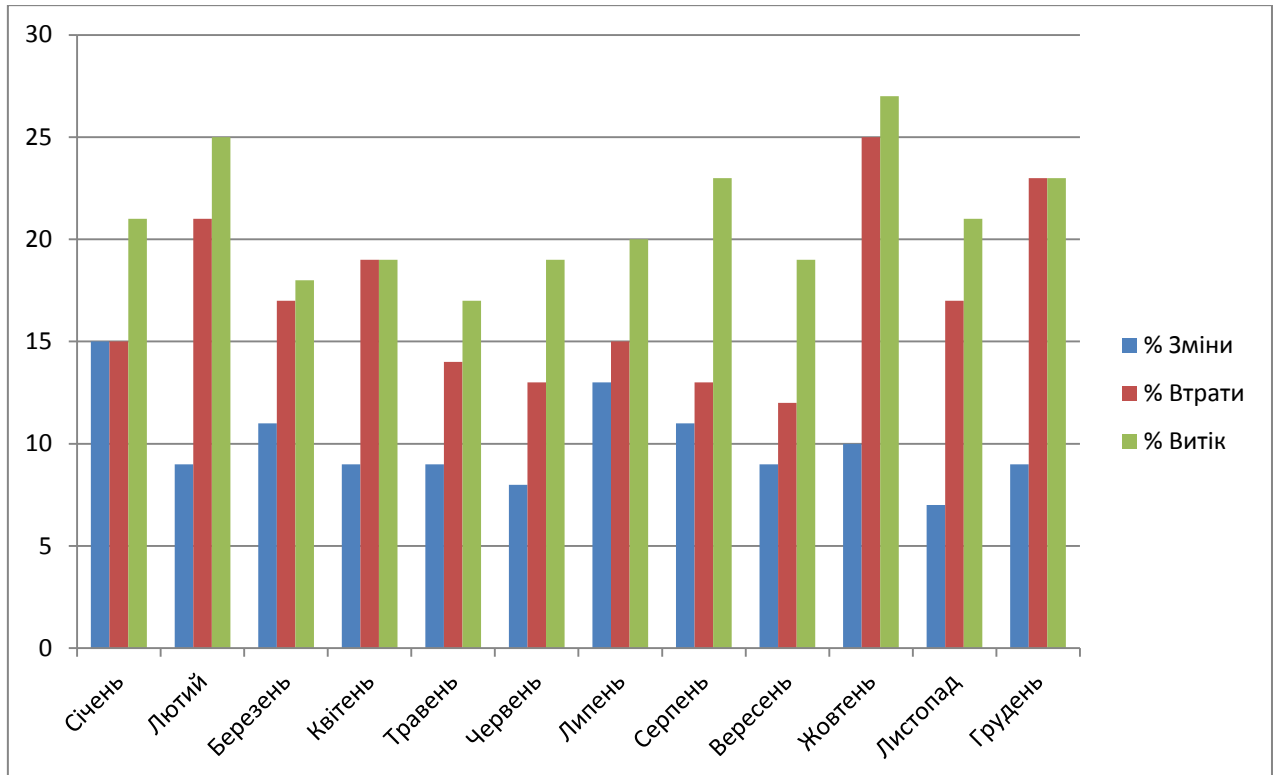


Рисунок. 2.9 – Робота з інформацією 2017 р.

Табл. 2.8 та рис. 2.9 наглядно демонструє який відсоток інформації було пошкоджено за 2017 р. Протягом 2017 року, відсоток зміни інформації коливається в межах 10 - 15 відсотків. А втрата і витік інформації мають, куди більший відсоток, який набагато вище, ніж у змінній інформації. А так же має виражені піки в Лютому, Жовтні, Грудні, а так само Серпні.

2.6 Користь системи

Маючи всі ці параметри можна побудувати графік загальної користі системи спираючись на такі дані:

Таблиця 2.9 – Користь системи 2013 – 2017 р.

Рік	Користь
2013	0,711675
2014	0,612375
2015	0,581158
2016	0,51968
2017	0,418912



Рисунок 2.10 – Користь системи 2013 – 2017 р.

Користь даного програмного забезпечення падає з кожним роком, це чітко видно на графіку. Табл. 2.9 та рис. 2.10 заражують як змінювалась користь програми у 2013 – 2017 р.

Таблиця 2.10 – Користь системи 2017 р.

2017	Користь
Січень	0,40964
Лютий	0,341865
Березень	0,463212
Квітень	0,447638
Травень	0,47832
Червень	0,438
Липень	0,444912
Серпень	0,439794
Вересень	0,49236
Жовтень	0,27626
Листопад	0,45892
Грудень	0,33786

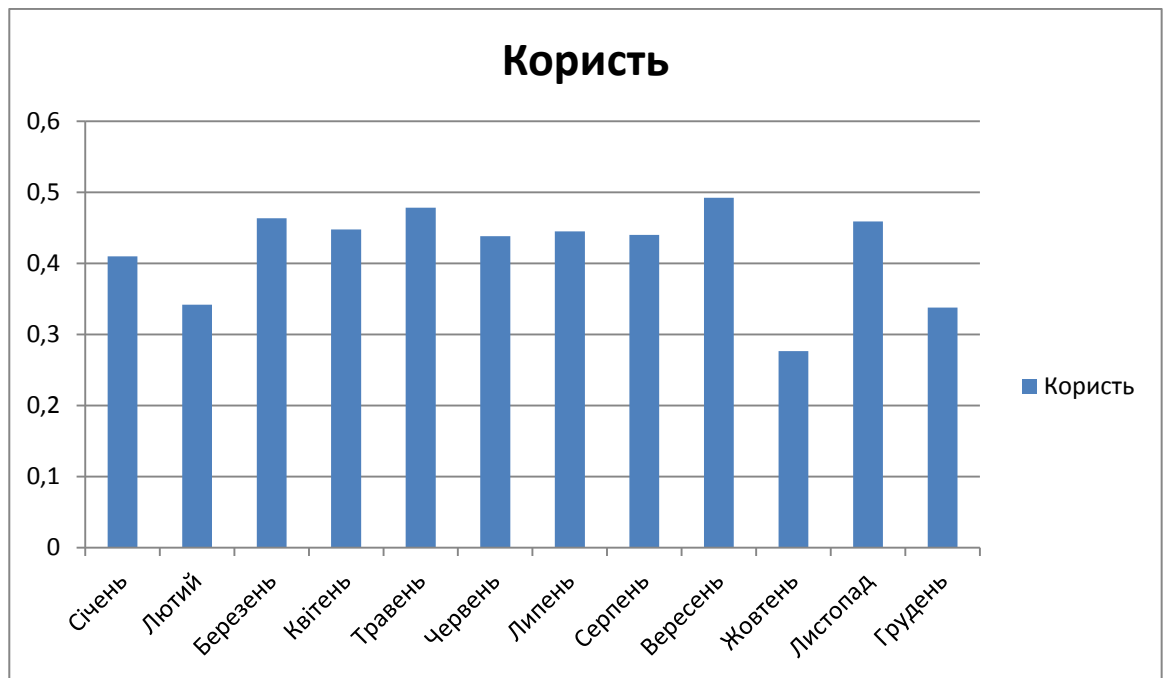


Рисунок. 2.11 – Користь системи 2017 р.

У Лютому, Жовтні і Грудні повна користь програмного забезпечення сильно знижується. Це все викликано сильним падінням ККД в ці періоди, а так само великою втратою даних. Табл. 2.10 та рис. 2.11 заражують як змінювалась користь програми у 2013 – 2017 р.

2.7 Загальна користь

Загальну користь системи можна порахувати за формулою:

$$P = \eta * (1 - C) \quad (2.3)$$

Де P - загальна користь системи, η - ККД системи, а C - загальний відсоток не коректної інформації.

Згадав що $\eta = T_{\text{пр}} * S + T_{\text{бпр}}$, наша формула набуде вигляду:

$$P = (T_{\text{пр}} * S + T_{\text{бпр}}) * (1 - C) \quad (2.4)$$

Також не забуваємо про те, що $S = \frac{T_{\text{общ}}}{T_{\text{кор}}}$, тоді отримуємо:

$$P = \left(T_{\text{пр}} * \frac{T_{\text{общ}}}{T_{\text{кор}}} + T_{\text{бпр}} \right) * (1 - C) \quad (2.5)$$

Наше C , являє собою суму всіх відхилень інформації від норми і розраховується за формулою:

$$C = O_{\text{и}} + O_{\text{п}} + O_{\text{у}} \quad (2.6)$$

Де $O_{И}$ відсоток зміненої інформації, $O_{П}$ – відсоток втраченої інформації, а $O_{У}$ – відсоток інформації яка була витоком.

Тепер якщо замість C , в нашу формулу підставити суму відхилень то вийде такий вираз:

$$P = \left(T_{пр} * \frac{T_{общ}}{T_{кор}} + T_{бпр} \right) * (1 - (O_{И} + O_{П} + O_{У})) \quad (2.7)$$

Трохи перетворивши наше вираз ми отримуємо:

$$P = \left(\frac{T_{общ} * T_{пр}}{T_{кор}} + T_{бпр} \right) * (1 - (O_{И} + O_{П} + O_{У})) \quad (2.8)$$

2.8 Математична модель та її аналіз

$$P = \left(\frac{T_{общ} * T_{пр}}{T_{кор}} + T_{бпр} \right) * (1 - (O_{И} + O_{П} + O_{У})) \quad (2.9)$$

Для аналізу даної математичної моделі системи, проаналізуємо всі її складові.

$T_{общ}$ – Загальний час роботи програми.

$T_{пр}$ – Час роботи співробітника з програмою.

$T_{кор}$ – Коректне час роботи програми.

$T_{бпр}$ – Час роботи співробітника без програми.

$O_{И}$ – Відсоток зміненої інформації.

$O_{П}$ – Відсоток втраченої інформації.

$O_{У}$ – Відсоток інформації, яка потрапила в чужі руки.

$T_{общ}$ – Це час, який система знаходиться в роботі і виконує ті чи інші функції. Причому незалежно від того, чи є поряд користувач чи ні. На цей параметр ми не можемо вплинути, так як він ставиться безпосередньо самим підприємством.

$T_{пр}$ Це час, протягом якого працівник підприємства користується даною системою для виконання службових обов'язків. Воно на пряму залежить від посадової інструкції, і не є можливим змінити його. Тому цей параметр нас не цікавить.

$T_{бпр}$ – Це час, протягом якого працівник виконує свої обов'язки без допомоги програми, тобто самостійно. Це регламентується його посадовою інструкцією, і ми не можемо вплинути на цей параметр.

$T_{кор}$ – Час, протягом якого система повністю виконувала покладені на неї функції, тобто працювала справно. Цей параметр безпосередньо залежить від якості самого програмного

забезпечення, а так же апаратних і програмних вимог. На нього ми можемо вплинути, тому зупинимося трохи докладніше.

O_I – Відсоток інформації, яка зазнала змін, або під впливом некоректності роботи системи, або під зовнішніми впливами. Змінами користувачів або ж стороннім втручанням. Це той параметр, який ми можемо змінити шляхом поліпшення системи.

O_P – Відсоток інформації, яка була загублена або ж видалена з системи в процесі роботи. Дані зміни є безповоротними, внаслідок чого несуть дуже велику загрозу. Вони можуть відбуватися як по недоліків роботи самої системи, так і з-за стороннього втручання. Ми можемо змінити цей параметр в необхідну сторону, для поліпшення роботи системи.

O_U – Відсоток інформації, яка була вкрадена з підприємства або ж потрапила в чужі руки. Розголошення конфіденційної інформації карається законом, а так само штрафується на підприємстві. Витік інформації може бути пов'язана як з співробітниками які некоректно її розпоряджаються, так і зі сторонніми особами які в корисливих цілях хочуть її отримати. Цей фактор так само можна змінити.

$T_{кор}$ це параметр, який відповідає за коректну роботу програми. Ми його можемо змінювати. Як видно з формули $\frac{T_{кор}}{T_{общ}}$, чим він вищий, тим буде краще стабільність роботи нашої системи. Для того, щоб максимально наблизити стабільність роботи нашої системи до одиниці, необхідно, щоб цей параметр, був максимально близький до параметру - $T_{общ}$. Для його збільшення ми можемо стабілізувати роботу нашої системи і збільшити її відмово стійкість. Це досягається шляхом перемикання системи з одного режиму роботи до іншого в разі її відмови.

Для спрощення аналізу змінних об'єднаємо всю роботу з інформацією, а саме змінні: O_I, O_P, O_U в одну змінну C . Вона у нас відповідає за зміну інформації, яка не планувалась. Як видно з формули $(1 - C)$ чим вище наша змінна C , тим більше відхилення нашою інформацією від нормального стану. Для збільшення продуктивності системи, треба цей параметр мінімізувати. Цього можна домогтися шляхом додаткового захисту інформації та її резервного копіювання.

2.9 Висновки до розділу 2

Проаналізувавши отриману математичну модель, і зібрану інформацію можна дійти такого висновку. Для поліпшення роботи системи необхідно збільшити її відмово стійкість шляхом перемикання режимів роботи, в разі виявлення несправностей. А так же додатково захистити інформацію від змін і втрати, за рахунок резервного копіювання та шифрування.

3 ПРАКТИЧНА РЕАЛІЗАЦІЯ

Для реалізації даного завдання була обрана середовище розробки Visual Studio, а так само такі мови програмування як C++ і C#.

3.1 Загальні відомості про Microsoft Visual Studio

Microsoft Visual Studio - лінійка продуктів компанії Microsoft, що включають інтегроване середовище розробки програмного забезпечення і ряд інших інструментальних засобів. Дані продукти дозволяють розробляти як консольні додатки, так і додатки з графічним інтерфейсом, в тому числі з підтримкою технології Windows Forms, а також веб-сайти, веб-додатки, веб-служби як в рідному, так і в керованому кодах для всіх платформ, підтримуваних Windows, Windows Mobile, Windows CE, .NET Framework, Xbox, Windows Phone .NET Compact Framework і Silverlight.

Visual Studio включає в себе редактор вихідного коду з підтримкою технології IntelliSense і можливістю найпростішого рефакторінга коду. Вбудований відладчик може працювати як відладчик рівня вихідного коду, так і відладчик машинного рівня. Решта вбудовуються інструменти включають в себе редактор форм для спрощення створення графічного інтерфейсу додатку, веб-редактор, дизайнер класів і дизайнер схеми бази даних. Visual Studio дозволяє створювати і підключати сторонні додатки (плагіни) для розширення функціональності практично на кожному рівні, включаючи додавання підтримки систем контролю версій вихідного коду (як, наприклад, Subversion і Visual SourceSafe), додавання нових наборів інструментів (наприклад, для редагування і візуального проектування коду на предметно-орієнтованих мовах програмування) або інструментів для інших аспектів процесу розробки програмного забезпечення (наприклад, клієнт Team Explorer для роботи з Team Foundation Server).

3.2 Загальні відомості про C++

3.2.1 Узагальнена інформація

C ++ - компільований, статично типізований мова програмування загального призначення.

Підтримує такі парадигми програмування, як процедурне програмування, об'єктно-орієнтоване програмування, узагальнене програмування. Мова має багату стандартну бібліотеку, яка включає в себе поширені контейнери і алгоритми, введення-виведення, регулярні вирази, підтримку багато поточності і інші можливості. C ++ поєднує властивості як високорівневих, так і низькорівневих мов. У порівнянні з його попередником - мовою C, - найбільшу увагу приділено підтримці об'єктно-орієнтованого і узагальненого програмування.

C ++ широко використовується для розробки програмного забезпечення, будучи одним з найпопулярніших мов програмування. Область його застосування включає створення операційних систем, різноманітних прикладних програм, драйверів пристроїв, додатків для вбудованих систем, високопродуктивних серверів, а також розважальних програм (ігор). Існує безліч реалізацій мови C ++, як безкоштовних, так і комерційних і для різних платформ. Наприклад, на платформі x86 це GCC, Visual C ++, Intel C ++ Compiler, Embarcadero (Borland) C ++ Builder і інші. C ++ зробив величезний вплив на інші мови програмування, в першу чергу на Java і C #.

Синтаксис C ++ успадкований від мови C. Одним з принципів розробки було збереження сумісності з C. Проте, C ++ не є в строгому сенсі надбезліччю C; безліч програм, які можуть однаково успішно транслюватися як компіляторами C, так і компіляторами C ++, досить велике, але не включає всі можливі програми на C.

3.2.2 Історія розвитку

У 1985 році вийшло перше видання «Мови програмування C ++», що забезпечує перший опис цієї мови, що було надзвичайно важливо через відсутність офіційного стандарту. У 1989 році відбувся вихід C ++ версії 2.0. Його нові можливості включали

множинне спадкування, абстрактні класи, статичні функції-члени, функції-константи і захищені члени. У 1990 році вийшло «Кухня країн довідкове керівництво по С ++», покладене згодом в основу стандарту. Останні оновлення включали шаблони, виключення, простору імен, нові способи приведення типів і булевский тип.

Стандартна бібліотека С ++ також розвивалася разом з ним. Першим додаванням до стандартної бібліотеці С ++ стали потоки введення-виведення, що забезпечують засоби для заміни традиційних функцій С printf і scanf. Пізніше самим значним розвитком стандартної бібліотеки стало включення в неї Стандартної бібліотеки шаблонів.

У 1998 році був опублікований стандарт мови ISO / IEC 14882: 1998 (відомий як С ++ 98), розроблений комітетом з стандартизації С ++ (ISO / IEC JTC1 / SC22 / WG21 working group). Стандарт С ++ не описує способи іменування об'єктів, деякі деталі обробки виключень і інші можливості, пов'язані з деталями реалізації, що робить несумісним об'єктний код, створений різними компіляторами. Однак для цього третіми особами створено безліч стандартів для конкретних архітектур і операційних систем.

У 2003 році був опублікований стандарт мови ISO / IEC 14882: 2003, де були виправлені виявлені помилки і недоліки попередньої версії стандарту.

У 2005 році був випущений звіт Library Technical Report (коротко званий TR1). Не будучи офіційно частиною стандарту, звіт описує розширення стандартної бібліотеки, які, як очікувалося авторами, повинні бути включені в наступну версію мови С ++. Ступінь підтримки TR1 поліпшується майже у всіх підтримуваних компіляторах мови С ++.

З 2009 року велася робота по оновленню попереднього стандарту, попередньої версією нового стандарту спершу був С ++ 09, а через рік С ++ 0x, сьогодні [коли?] - С ++ 11, куди були включені доповнення в ядро мови і розширення стандартної бібліотеки, в тому числі велику частину TR1.

С ++ продовжує розвиватися, щоб відповідати сучасним вимогам. Одна з груп, що розробляють мову С ++ і напрямних комітету по стандартизації С ++ пропозиції щодо його поліпшення - це Boost, яка займається, в тому числі, вдосконаленням можливостей мови шляхом додавання в нього особливостей мета програмування.

Ніхто не має права на мову С ++, він є вільним. Однак сам документ стандарту мови (за винятком чернеток) не доступний безкоштовно. В рамках процесу стандартизації, ISO випускає кілька видів видань. Зокрема, технічні доповіді та технічні характеристики публікуються, коли «видно майбутнє, але немає негайної можливості угоди для публікації

міжнародного стандарту.» До 2011 року не було опубліковано три технічних звіту по C ++: TR 19768: 2007 (також відомий як C ++, Технічний звіт 1) для розширень бібліотеки в основному інтегрований в C ++ 11, TR 29124: 2010 року для спеціальних математичних функцій, і TR 24733: 2011 року для десяткової арифметики з плаваючою точкою. Технічна специфікація DTS 18822 :. 2 014 (за файлової системи) була затверджена на початку 2015 року, і інші технічні характеристики знаходяться в стадії розробки і очікують схвалення.

3.2.3 Переваги мови

Найзначнішими перевагами даної мови є:

- Масштабованість. Мовою C ++ розробляють програми для найрізноманітніших платформ і систем.
- Можливість роботи на низькому рівні з пам'яттю, адресами, портами. Що, при необережному використанні, може легко перетворитися на недолік.
- Можливість створення узагальнених алгоритмів для різних типів даних, їх спеціалізація, і обчислення на етапі компіляції, використовуючи шаблони.
-

3.3 Загальні відомості про C#

3.3.1 Узагальнена інформація про C#

C # - об'єктно-орієнтована мова програмування. Розроблено в 1998-2001 роках групою інженерів під керівництвом Андерса Хейлсберг в компанії Microsoft як мова розробки додатків для платформи Microsoft .NET Framework і згодом був стандартизований як ECMA-334 і ISO / IEC 23270.

C # відноситься до сім'ї мов з C-подібним синтаксисом, з них його синтаксис найбільш близький до C ++ і Java. Мова має статичну типізацію, підтримує поліморфізм, перевантаження операторів (в тому числі операторів явного і неявного приведення типу), делегати, атрибути, події, властивості, узагальнені типи і методи, ітератори, анонімні функції з підтримкою замикань, LINQ, виключення, коментарі у форматі XML.

Переїнявши багато від своїх попередників - мов C ++, Pascal, Модула, Smalltalk і, особливо, Java - C #, спираючись на практику їх використання, виключає деякі моделі, що зарекомендували себе як проблематичні при розробці програмних систем, наприклад, C # на відміну від C ++ не підтримує множинне успадкування класів (між тим допускається множинне спадкування інтерфейсів).

Мова C # з'явився на світло в червні 2000 р, в результаті кропіткої роботи великої групи розробників компанії Microsoft, очолюваної Андерсом Хейлсбергом (Anders Hejlsberg). Ця людина відома як автор одного з перших компільованих мов програмування для персональних комп'ютерів IBM - Turbo Pascal. Напевно, на території колишнього Радянського Союзу багато розробників зі стажем, та й просто люди, котрі навчалися в тій чи іншій формі програмування в вузах, випробували на собі чарівність і зручність використання цього продукту. Крім того, під час роботи в корпорації Borland Андерс Хейлсберг прославився створенням інтегрованого середовища Delphi (він керував цим проектом аж до виходу версії 4.0).

Поява мови C # і ініціативи .NET аж ніяк не випадково довелось на початок літа 2000 р Саме до цього моменту компанія Microsoft підготувала промислові версії нових компонентних технологій і рішень в області обміну повідомленнями та даними, а також створення Internet-додатків (COM +, ASP +, ADO +, SOAP, Biztalk Framework). Безсумнівно, найкращим способом просування цих новинок є створення інструментарію для розробників з їх повноцінної підтримки. В цьому і полягає одна з головних задач нової мови C #. Крім того, Microsoft не могла більше розширювати все ті ж інструменти і мови розробки, роблячи їх все більш і більш складними для задоволення конфліктуючих між собою вимог підтримки сучасного обладнання та забезпечення зворотної сумісності з тими продуктами, які були створені на початку 1990-х рр. під час першої появи Windows. Настає момент, коли необхідно почати з чистого аркуша для того, щоб створити простий, але має складну структуру набір мов, середовищ і засобів розробки, які дозволять розробнику легко створювати сучасні програмні продукти.

C # і .NET і є тією самою відправною точкою. Якщо говорити спрощено, то .NET являє собою нову платформу, новий API для програмування в Windows, а C # є нову мову, створений з нуля, для роботи з цією платформою, а також для вилучення всіх вигод з прогресу середовищ розробки і нашого розуміння принципів об'єктно-орієнтованого програмування протягом останніх 20 років.

Необхідно відзначити, що зворотна сумісність не втрачена. Існуючі програми будуть виконуватися, а платформа .NET була спроектована таким чином, щоб вона могла працювати з наявним програмним забезпеченням. Зв'язок між компонентами в Windows зараз майже повністю здійснюється за допомогою COM. З урахуванням цього .NET має здатність створювати оболонки (wrappers) навколо існуючих компонентів COM, так що компоненти .NET можуть спілкуватися з ними, і створювати оболонки навколо компонентів .NET, що дозволяє їм виглядати як звичайні COM-компоненти.

3.3.2 Історія розвитку C#

Автори C # прагнули створити мову, що поєднує простоту і виразність сучасних об'єктно-орієнтованих мов (на кшталт Java) з багатством можливостей і міццю C ++. За словами Андерса Хейлсберг, C # запозичив більшість своїх синтаксичних конструкцій з C ++. Зокрема, в ньому присутні такі зручні типи даних, як структури і перерахування (інший нащадок C ++ - Java - позбавлений цих елементів, що створює певні незручності при програмуванні). Синтаксичні конструкції C # успадковані не тільки від C ++, але і від Visual Basic. Наприклад, в C #, як і в Visual Basic, використовуються властивості класів. Як C ++, C # дозволяє виробляти перевантаження операторів для створених вами типів, Java не підтримує ні ту, ні іншу можливість). C # це фактично гібрид різних мов. При цьому C # синтаксично не менше (якщо не більше) чистий, ніж Java, і так само простий, як Visual Basic, але володіє практично тією ж потужністю і гнучкістю, що і C ++.

3.3.3 Переваги мови

Найзначнішими перевагами даної мови є:

- Повна підтримка класів і об'єктно-орієнтованого програмування, включаючи успадкування інтерфейсів і реалізацій, віртуальних функцій і перевантаження операторів.
- Повний і добре певний набір основних типів.
- Вбудована підтримка автоматичної генерації XML-документації.
- Автоматичне звільнення динамічно розподіленої пам'яті.

- Можливість позначки класів і методів атрибутами, визначеними користувачем. Це може бути корисно при документуванні та здатне впливати на процес компіляції (наприклад, можна помітити методи, які повинні компілюватися тільки в отладочном режимі).
- Повний доступ до бібліотеки базових класів .NET, а також легкий доступ до Windows API (якщо це дійсно необхідно).
- Показчики та прямий доступ до пам'яті, якщо вони необхідні. Однак мова розроблений таким чином, що в переважній більшості випадків можна обійтися і без цього.
- Підтримка властивостей і подій в стилі VB.
- Просте зміна ключів компіляції. Дозволяє отримувати виконувані файли або бібліотеки компонентів .NET, які можуть бути викликані іншим кодом так само, як елементи управління ActiveX (компоненти COM).
- Можливість використання C # для написання динамічних web-сторінок ASP.NET.
-

3.4 Розробка серверної частини

Для включення сервера необхідно ввести реєстраційні дані, а саме логін і пароль. Після чого натиснути на кнопку «Enter». Якщо всі дані заповнені вірно, то відбудеться включення сервера, якщо ж ні, то буде видане повідомлення про помилку. На рис. 3.1 можна побачити вікно логіну серверної частини.

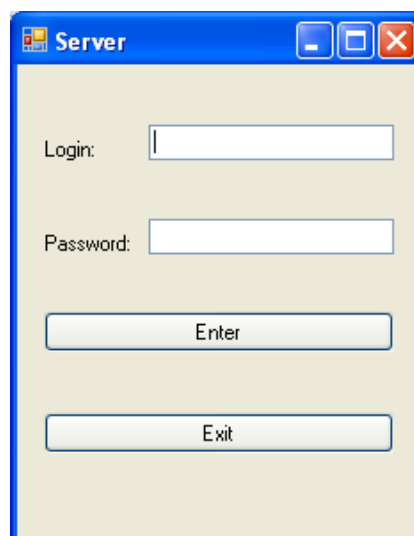


Рисунок 3.1 – Вікно логіну серверної частини

Як виконується процес авторизації можна побачити у наступному кодї:

```
1      FileInfo f = new FileInfo("Server_Login.lg");
2      FileStream s = f.Open(FileMode.OpenorWrite, FileAccess.Read);
3      string login = null;
4      string password = null;
5      login = f.ReadLine();
6      password = f.ReadLine();
7      if ((textBox1.Text != login) || (textBox2.Text != password))
8      {
9          label3.Text = "Wrong login or password. Pleace check information.";
10         label3.Visible = true;
11     }
12     else
13     {
14         System.Diagnostics.Process Proc = new System.Diagnostics.Process();
15         Proc.StartInfo.FileName = "Server.exe";
16         Proc.Start();
17     }
18     f.Close()
```

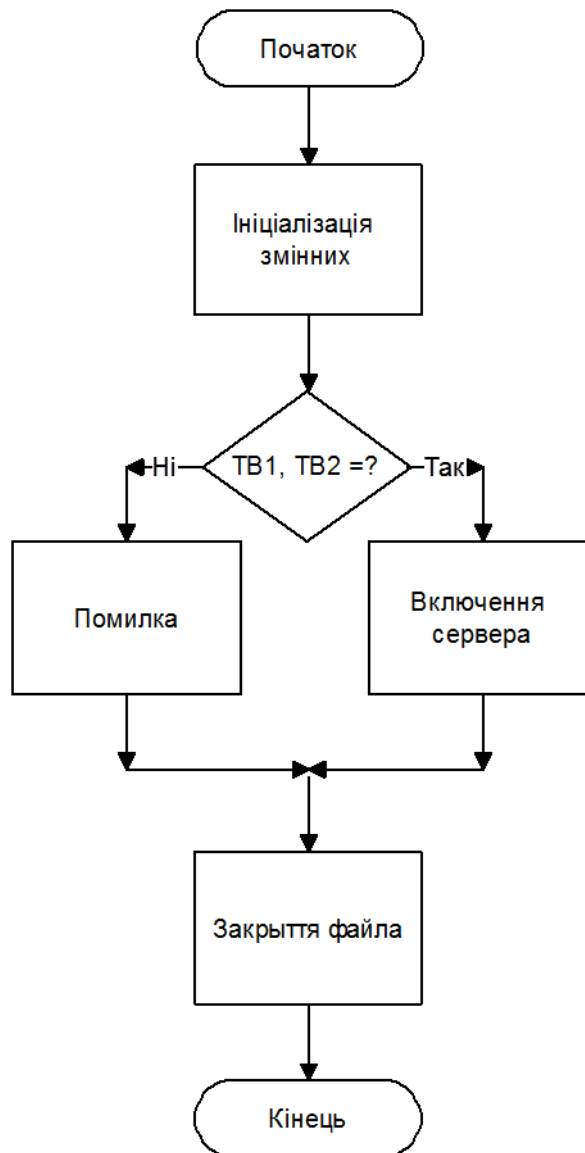


Рисунок 3.2 – Схема роботи процедури авторизації

Server.exe, який є службою Windows, лежить в кореневій папці програми і скомпільовано і зібраний на мові C++ в середовищі розробки Visual Studio.

Для установки сервісу необхідно запустити одну з його функцій а саме InstallService. Лістинг якої можна виявити нижче.

```

1     int InstallService() {
2         SC_HANDLE      hSCManager      =      OpenSCManager(NULL,      NULL,
3     SC_MANAGER_CREATE_SERVICE);
4         if(!hSCManager) {

```

```
5     addLogMessage("Error: Can't open Service Control Manager");
6
7     return -1;
8
9     }
10
11     SC_HANDLE hService = CreateService(
12         hSCManager,
13         serviceName,
14         serviceName,
15         SERVICE_ALL_ACCESS,
16         SERVICE_WIN32_OWN_PROCESS,
17         SERVICE_DEMAND_START,
18         SERVICE_ERROR_NORMAL,
19         servicePath,
20         NULL, NULL, NULL, NULL, NULL
21     );
22
23     if(!hService) {
24         int err = GetLastError();
25         switch(err) {
26             case ERROR_ACCESS_DENIED:
27                 addLogMessage("Error: ERROR_ACCESS_DENIED");
28                 break;
29             case ERROR_CIRCULAR_DEPENDENCY:
30                 addLogMessage("Error: ERROR_CIRCULAR_DEPENDENCY");
```

```
29         break;
30     case ERROR_DUPLICATE_SERVICE_NAME:
31         addLogMessage("Error: ERROR_DUPLICATE_SERVICE_NAME");
32         break;
33     case ERROR_INVALID_HANDLE:
34         addLogMessage("Error: ERROR_INVALID_HANDLE");
35         break;
36     case ERROR_INVALID_NAME:
37         addLogMessage("Error: ERROR_INVALID_NAME");
38         break;
39     case ERROR_INVALID_PARAMETER:
40         addLogMessage("Error: ERROR_INVALID_PARAMETER");
41         break;
42     case ERROR_INVALID_SERVICE_ACCOUNT:
43         addLogMessage("Error: ERROR_INVALID_SERVICE_ACCOUNT");
44         break;
45     case ERROR_SERVICE_EXISTS:
46         addLogMessage("Error: ERROR_SERVICE_EXISTS");
47         break;
48     default:
49         addLogMessage("Error: Undefined");
50     }
51     CloseServiceHandle(hSCManager);
52     return -1;
```

```
53     }  
54     CloseServiceHandle(hService);  
55     CloseServiceHandle(hSCManager);  
56     addLogMessage("Success install service!");  
57     return 0;  
58     }  
59
```

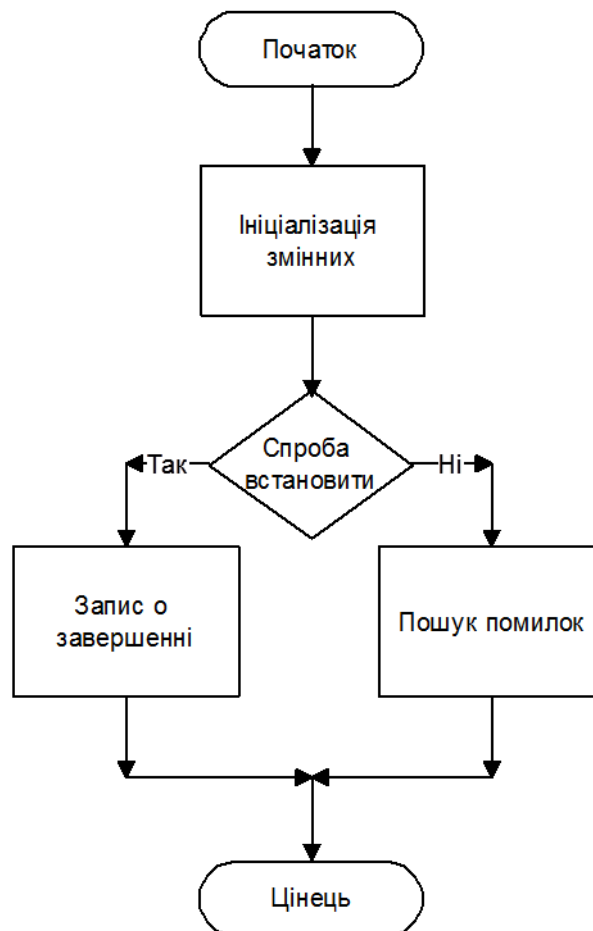


Рисунок 3.3 – Схема роботи процедури InstallService

Ця функція дозволяє встановити сервіс на комп'ютер користувача. У разі успішного встановлення сервісу в лог подій буде зроблено запис про це. В іншому ж випадку при виникненні помилки процедура буде припинена. Рис 3.3 наглядно демонструє принцип функціонування роботи процедури InstallService.

Далі запуск сервісу відбувається шляхом використання його функції StartService, лістинг якої можна виявити трохи нижче.

```
1      Int StartService() {
2          SC_HANDLE      hSCManager      =      OpenSCManager(NULL,      NULL,
3      SC_MANAGER_CREATE_SERVICE);
4          SC_HANDLE      hService      =      OpenService(hSCManager,      serviceName,
5      SERVICE_START);
6          if(!StartService(hService, 0, NULL)) {
7              CloseServiceHandle(hSCManager);
8              addLogMessage("Error: Can't start service");
9              return -1;
10         }
11
12         CloseServiceHandle(hService);
13         CloseServiceHandle(hSCManager);
14         Return 0;
15     }
```

У разі якщо виникає помилка в процесі запуску сервера, то в лог файл робиться позначка про те, що сервіс не може бути запущений. В іншому випадку відбувається запуск.

А так само видалення сервісу в разі потреби можна виконати використовую його функцію видалення, а саме RemoveService. Сама ж функція має такий вигляд:

```
1  SC_HANDLE      hSCManager      =      OpenSCManager(NULL,      NULL,
2  SC_MANAGER_ALL_ACCESS);

3  if(!hSCManager) {

4      addLogMessage("Error: Can't open Service Control Manager");

5      return -1;

6  }

7  SC_HANDLE hService = OpenService(hSCManager, serviceName, SERVICE_STOP |
8  DELETE);

9  if(!hService) {

10     addLogMessage("Error: Can't remove service");

11     CloseServiceHandle(hSCManager);

12     return -1;

13 }

14

15 DeleteService(hService);

16 CloseServiceHandle(hService);

17 CloseServiceHandle(hSCManager);

18 addLogMessage("Success remove service!");

19 return 0;

20 }
```

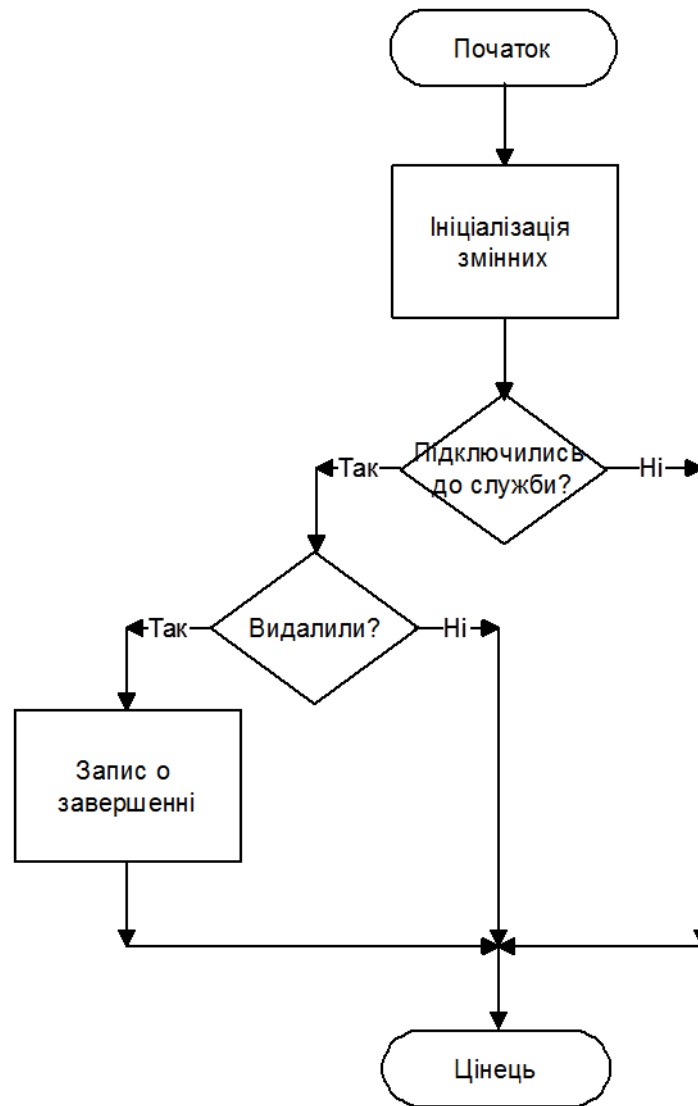


Рисунок 3.4 – Схема роботи процедури RemoveService

Рис 3.4 наглядно демонструє принцип функціонування роботи процедури RemoveService. В даному випадку можливі помилки двох типів. Або немає можливості відкрити Service Control Manager, який є стандартною службою Windows, або немає можливості провести видалення даного сервісу з системи.

Так само у сервера є функція авторизації підключених клієнтів - ToClient. У разі підключення декількох клієнтів вони ставати в чергу і чекають завершення перевірки на авторизацію.

Основна функція сервера забезпечує їх підключення до системи а також дозволяє проводити обмін інформацією між сервером і клієнтами, і має такий вигляд:

```
1  Int t_main ()
2  {
3  char buff[1024];
4      printf("TCP SERVER \n");
5      if (WSAStartup(0x0202, (WSADATA *)&buff[0]))
6      {
7          printf("Error WSAStartup %d\n", WSAGetLastError());
8          return -1;
9      }
10     SOCKET mysocket;
11     if ((mysocket = socket(AF_INET, SOCK_STREAM, 0)) < 0)
12     {
13         printf("Error socket %d\n", WSAGetLastError());
14         WSACleanup();
15         return -1;
16     }
17
18     sockaddr_in local_addr;
19     local_addr.sin_family = AF_INET;
20     local_addr.sin_port = htons(MY_PORT);
21     local_addr.sin_addr.s_addr = 0;
22     if (bind(mysocket, (sockaddr *)&local_addr, sizeof(local_addr)))
23     {
24         printf("Error bind %d\n", WSAGetLastError());
```

```
25     closesocket(mysocket);
26     WSACleanup();
27     return -1;
28 }
29 if (listen(mysocket, 0x100))
30 {
31     printf("Error listen %d\n", WSAGetLastError());
32     closesocket(mysocket);
33     WSACleanup();
34     return -1;
35 }
36
37 printf("ozidanie podkluceniy...\n");
38
39 SOCKET client_socket;
40 sockaddr_in client_addr;
41 int client_addr_size = sizeof(client_addr);
42 while ((client_socket = accept(mysocket, (sockaddr *)&client_addr, \
43     &client_addr_size)))
44 {
45     nclients++;
46     HOSTENT *hst;
47     hst = gethostbyaddr((char *)&client_addr.sin_addr.s_addr, 4, AF_INET);
48     printf("+%s [%s] new connect!\n",
```

```

49     (hst) ? hst->h_name : "", inet_ntoa(client_addr.sin_addr));
50     PRINTUSERS
51     DWORD thID;
52     CreateThread(NULL, NULL, ToClient, &client_socket, NULL, &thID);
53 }
54 return 0;
55 }
56     DWORD WINAPI ToClient(LPVOID client_socket)
57 {
58     SOCKET my_sock;
59     my_sock = ((SOCKET *)client_socket)[0];
60     char buff[BUFSIZ]; //char buff[20 * 1024];
61     #define SHELLO "SOCKET PODKLUCHEN\r\n"
62     send(my_sock, SHELLO, sizeof(SHELLO), 0);
63     int bytes_recv;
64     while ((bytes_recv = recv(my_sock, &buff[0], sizeof(buff), 0)) &&
65     bytes_recv != SOCKET_ERROR)
66         /*tmp_buff = buff[0];
67         //changeWords();
68     send(my_sock, &buff[0], bytes_recv, 0);
69     nclients--;
70     printf("-disconnect\n"); PRINTUSERS
71     closesocket(my_sock);
72     return 0;

```

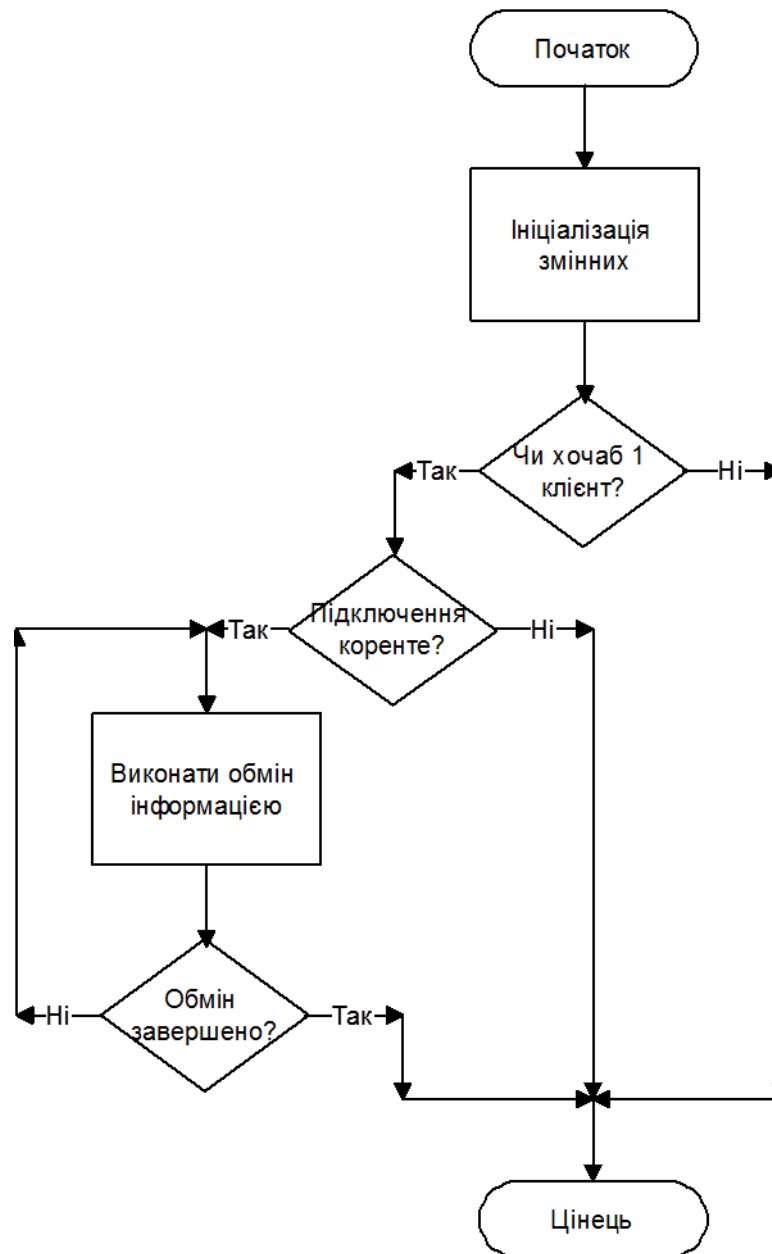


Рисунок 3.5 – Схема роботи процедури t_main

На схемі роботи процедури t_main рис 3.5 можна побачити в яких випадках система майже не виконує ніяких функцій, а знаходиться у режимі очікування.

3.5 Розробка клієнтської частини

Для включення клієнта необхідно ввести логін і пароль. Після чого натиснути на кнопку «Enter». Якщо всі дані заповнені вірно, то відбудеться підключення до сервера,

якщо ж ні, то буде видане повідомлення про помилку. Вікно логіну клієнта можна побачити на рис. 3.6.

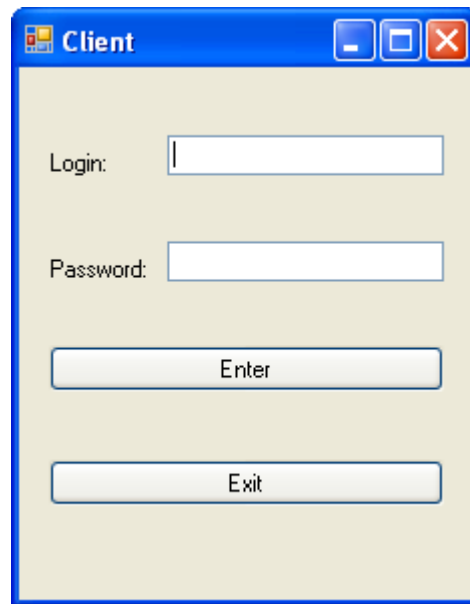


Рисунок 3.6 – Вікно логіну клієнта

```
1      FileInfo f = new FileInfo("Client_Login.lg");
2      FileStream s = f.Open(FileMode.OpenOrCreate, FileAccess.Read);
3      string read = null;
4      string login = null;
5      string password = null;
6      bool z = false;
7      while ((read = s.ReadLine()) != null)
8      {
9          login = read;
10         password = s.ReadLine();
11         if ((textBox1.Text == login) & (textBox2.Text == password))
12         {
13             z = true;
14         }
```



```

15     }
16     f.Close();
17     if (z == false)
18     {
19         label3.Text = "Wrong login or password. Pleace check information.";
20         label3.Visible = true;
21     }
22     else
23     {
24         Form2 form2= new Form2();
25         form2.Show();
26         System.Diagnostics.Process Proc = new System.Diagnostics.Process();
27         Proc.StartInfo.FileName = "Client.exe";
28         Proc.Start()

```

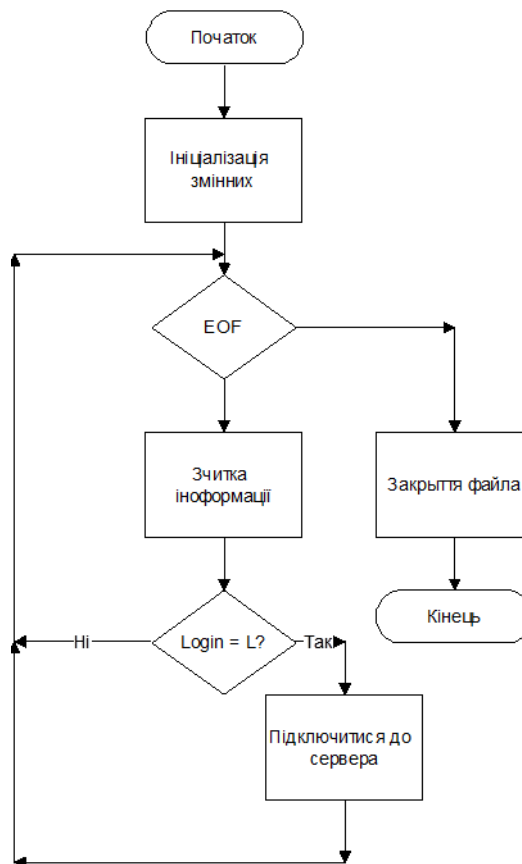


Рисунок 3.7 – Схема роботи авторизації користувача

Після початку роботи програма виробляє початкову ініціалізацію змінних. Далі відкривається файл з реєстраційними даними користувачів і відбувається циклічна перевірка введених даних до записаним у файлі. У разі якщо хоча б один запис збіглася, відбувається підключення клієнта до сервера для обміну інформацією. Схема роботи авторизації користувача представлена на Рис. 3.7.

Одна з функцій клієнтської частини полягає в тому, що б з'єднатися із сервером і обмінюватися з ним інформацією.

```
1  int _tmain(int argc, _TCHAR* argv[])
2      {
3          char buff[BUFSIZ];
4          printf("TCP CLIENT\n");
5          if (WSAStartup(0x202, (WSADATA *)&buff[0]))
6              {
7                  printf("WSAStart error %d\n", WSAGetLastError());
8                  return -1;
9              }
10         SOCKET my_sock;
11         my_sock = socket(AF_INET, SOCK_STREAM, 0);
12         if (my_sock < 0)
13             {
14                 printf("Socket() error %d\n", WSAGetLastError());
15                 return -1;
16             }
17         sockaddr_in dest_addr;
18         dest_addr.sin_family = AF_INET;
```

```
19     dest_addr.sin_port = htons(PORT);
20     HOSTENT *hst;
21     if (inet_addr(SERVERADDR) != INADDR_NONE)
22         dest_addr.sin_addr.s_addr = inet_addr(SERVERADDR);
23     else
24     {
25         if (hst = gethostbyname(SERVERADDR))
26             ((unsigned long *)&dest_addr.sin_addr)[0] =
27             ((unsigned long **)hst->h_addr_list)[0][0];
28         else
29         {
30             printf("Invalid address %s\n", SERVERADDR);
31             closesocket(my_sock);
32             WSACleanup();
33             return -1;
34         }
35     }
36     if (connect(my_sock, (sockaddr *)&dest_addr, sizeof(dest_addr)))
37     {
38         printf("Connect error %d\n", WSAGetLastError());
39         ChangeWorkingType(WSAGetLastError, -1);
40         return -1;
41     }
```

```
42     printf("Soedinenie s %s uspeshno ustanovlenno\n \
43         Type quit for quit\n\n", SERVERADDR);
44     int nsize;
45     while ((nsize = recv(my_sock, &buff[0], sizeof(buff) - 1, 0)) != SOCKET_ERROR)
46     {
47         buff[nsize] = 0;
48         printf("S=>C:%s", buff);
49         printf("S<=C:"); fgets(&buff[0], sizeof(buff) - 1, stdin);
50         if (!strcmp(&buff[0], "quit\n"))
51         {
52             printf("Exit...");
53             closesocket(my_sock);
54             WSACleanup();
55             return 0;
56         }
57         send(my_sock, &buff[0], strlen(&buff[0]), 0);
58     }
59     printf("Recv error %d\n", WSAGetLastError());
60     closesocket(my_sock);
61     WSACleanup();
62     return -1;
63     //return 0;
```

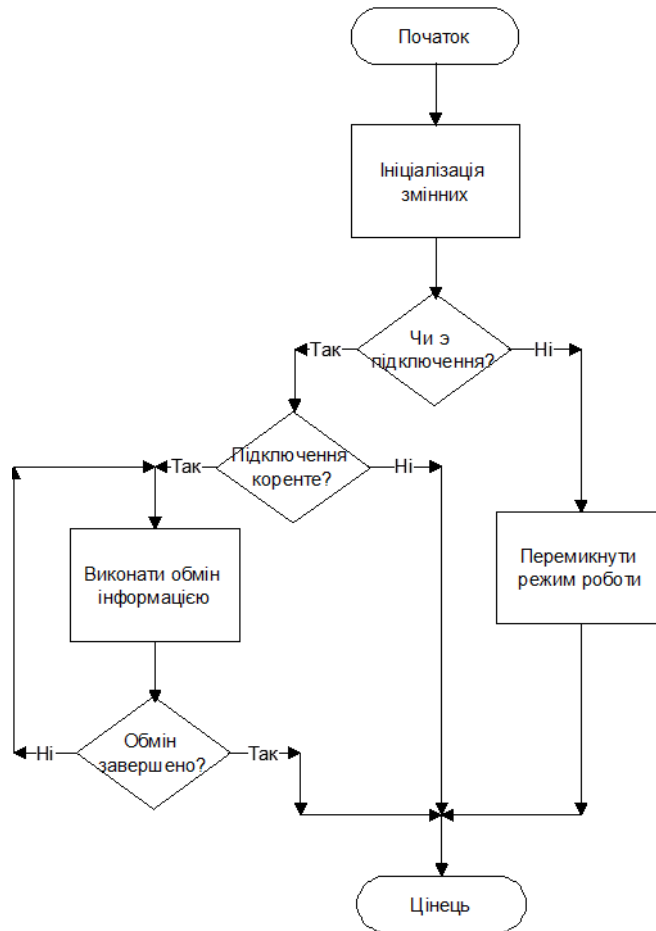


Рисунок 3.8 – Схема роботи основної функції клієнта

Схема роботи основної функції клієнта наглядно представлена на рис. 3.8. Після запуску клієнтської частини, відбувається перевірка, чи є підключення до сервера. У разі його наявності, воно перевіряється на коректність, і якщо перевірка пройдена, починається обмін інформацією. Якщо ж хоча б одна перевірка не пройшла, тоді програма завершує свою роботу до наступного підключення. У разі якщо з'єднання із сервером немає, програма переходить в розподілене режим роботи.

3.6 Висновки до розділу 3

- Сформовано графічні інтерфейси для клієнтської і серверної частини.
- Реалізована система авторизації користувачів.
- Реалізовано захист інформації за рахунок закритого каналу обміну.
- Збільшена стабільність роботи шляхом поєднання клієнт-північного режиму і розподіленого.

4 ОХОРОНА ПРАЦІ ТА БЕЗПЕКИ В НАДЗВИЧАЙНИХ СИТУАЦІЯХ. ЕКОЛОГІЯ

В даному розділі проведено аналіз потенційних небезпечних та шкідливих виробничих факторів, причин пожеж. Розглянуті заходи, які дозволяють забезпечити гігієну праці і виробничу санітарію. На підставі аналізу розроблені заходи з техніки безпеки та рекомендації з пожежної профілактики.

Аналіз потенційно небезпечних і шкідливих виробничих чинників виконується для персонального комп'ютера, на якому буде виконуватися розробка.

4.1 Загальні питання з охорони праці

Умови праці на робочому місці, безпека технологічних процесів, машин, механізмів, устаткування та інших засобів виробництва, стан засобів колективного та індивідуального захисту, що використовуються працівником, а також санітарно-побутові умови повинні відповідати вимогам нормативних актів про охорону праці. В законі України «Про охорону праці» визначається, що охорона праці - це система правових, соціально-економічних, організаційно-технічних, санітарно-гігієнічних і лікувально-профілактичних заходів та засобів, спрямованих на збереження життя, здоров'я і працездатності людини у процесі трудової діяльності.

4.2 Правові та організаційні основи охорони праці

Основним організаційним напрямом у здійсненні управління в сфері охорони праці є усвідомлення пріоритету безпеки праці і підвищення соціальної відповідальності держави, і особистої відповідальності працівників.

4.3 Організаційно-технічні заходи з безпеки праці

В організації/підприємстві проводиться навчання і перевірка знань з питань охорони праці відповідно до вимог Типового положення про порядок проведення навчання і перевірки знань з питань охорони праці, затвердженого наказом Держнаглядохоронпраці України від 26.01.2005 N 15, зареєстрованого в Міністерстві юстиції України 15.02.2005 за N 231/10511 НПАОП 0.00-4.12-05 [15].

Також впроваджені організаційні заходи з пожежної безпеки - навчання і перевірку знань відповідно до вимог Типового положення про інструктажі, спеціальне навчання та перевірку знань з питань пожежної безпеки на підприємствах, в установах та організаціях України, затвердженого наказом Міністерства України з питань надзвичайних ситуацій та у справах захисту населення від наслідків Чорнобильської катастрофи від 29.09.2003 N 368, зареєстрованого в Міністерстві юстиції України 11.12.2003 за N 1148/8469 НАПБ Б.02.005-2003[18]. Обов'язковими вимогами враховане наступне:

- не слід допускати до роботи осіб, що в установленому порядку не пройшли навчання, інструктаж та перевірку знань з охорони праці, пожежної безпеки та цих Правил.
- на підприємстві/організації, де експлуатуються ЕОМ з відео дисплейними терміналами (ВДТ) і периферійними пристроями (ПП), розробляється інструкція з охорони праці відповідно до Положення про розробку інструкцій з охорони праці [16].
- ознайомлення з правилами безпеки праці, одержання відповідних інструктажів засвідчується у журналі інструктажів.
- перед допуском до самостійної роботи кожен працівник має право на навчання з питань охорони праці і роботодавець зобов'язаний, і проводить таке навчання у вигляді двох інструктажів з питань охорони праці: вступного та первинного.

4.4 Аналіз стану умов праці

Робота над проектом проходитиме в приміщенні багатоквартирного будинку. Для даної роботи достатньо однієї людини, для якої надано робоче місце зі стаціонарним комп'ютером.

4.4.1 Вимоги до приміщень

Геометричні розміри приміщення зазначені в Табл. 5.1.

Табл. 4.1 – Розміри приміщення

Найменування	Значення
Довжина, м	5
Ширина, м	5
Висота, м	3
Площа, м ²	25
Об'єм, м ³	75

Згідно з [14] розмір площі для одного робочого місця оператора персонального комп'ютера має бути не менше 6 кв. м, а об'єм — не менше 20 куб. м. Отже, дане приміщення цілком відповідає зазначеним нормам.

4.4.2 Вимоги до організації місця праці

При порівнянні відповідності характеристик робочого місця нормативним основні вимоги до організації робочого місця за [12] (Табл. 5.2) і відповідними фактичними значеннями для робочого місця, констатуємо повну відповідність.

Табл. 4.2 - Характеристики робочого місця

Найменування параметра	Фактичне значення	Нормативне значення
Висота робочої поверхні, мм	750	680 ÷ 800
Висота простору для ніг, мм	730	не менше 600
Ширина простору для ніг, мм	660	не менше 500
Глибина простору для ніг, мм	700	не менше 650
Висота поверхні сидіння, мм	470	400 ÷ 500
Ширина сидіння, мм	400	не менше 400
Глибина сидіння, мм	400	не менше 400
Висота поверхні спинки, мм	600	не менше 300
Ширина опорної поверхні спинки, мм	500	не менше 380
Радіус кривини спинки в горизонтальній площині, мм	400	400
Відстань від очей до екрану дисплея, мм	800	700 ÷ 800

Приміщення кабінету знаходиться на другому поверсі трьох поверхової будівлі і має об'єм 78 м³, площу – 18 м². У цьому кабінеті обладнано три місця праці, з яких два укомплектовані ПК.

Температура в приміщенні протягом року коливається у межах 18–24°C, відносна вологість — близько 50%. Швидкість руху повітря не перевищує 0,2 м/с. Шум в лабораторії знаходиться на рівні 50 дБА. Система вентилявання приміщення — природна неорганізована, а опалення — централізоване.

4.4.3 Навантаження та напруженість процесу праці

Під час виконання випускної роботи: за фізичним навантаженням робота відноситься до категорії легкі роботи (Ia), її виконують сидячи з періодичним ходінням. Щодо

характеру організування виконання дипломної роботи, то він підпадає під нав'язаний режим, оскільки певні розділи роботи необхідно виконати у встановлені конкретні терміни.

Рекомендовано застосування екранних фільтрів, локальних світлофільтрів (засобів індивідуального захисту очей) та інших засобів захисту, а також інші профілактичні заходи.

Роботу за дипломним проектом визнано, такою, що займає 50% часу робочого дня та за восьмигодинної робочої зміни рекомендовано встановити додаткові регламентовані перерви - для розробників програм тривалістю 15 хв. через кожну годину роботи;

4.4.4 Виробнича санітарія

На підставі аналізу небезпечних та шкідливих факторів при виробництві (експлуатації), пожежної безпеки можуть бути надалі вирішені питання необхідності забезпечення працюючих достатньою кількістю освітлення, вентиляції повітря, організації заземлення, тощо.

4.4.5 Аналіз небезпечних та шкідливих факторів

Аналіз небезпечних та шкідливих виробничих факторів виконується у табличній формі (Табл. 5.3). Роботу, пов'язану з ЕОП з ВДТ, у тому числі на тих, які мають робочі місця, обладнані ЕОМ з ВДТ і ПП, виконують із забезпеченням виконання НПАОП 0.00.-1.28-10 [20], які встановлюють вимоги безпеки до обладнання робочих місць, до роботи із застосуванням ЕОМ з ВДТ і ПП. Основними робочими характеристиками персонального комп'ютера є:

- робоча напруга $U=+220\text{В} \pm 5\%$;
- робочий струм $I=2\text{А}$;
- споживана потужність $P=350\text{ Вт}$.

Таблиця 4.3 – Аналіз небезпечних і шкідливих виробничих факторів

Небезпечні і шкідливі виробничі фактори	Джерела факторів (види робіт)	Кількісна оцінка	Нормативні документи
1	2	3	4
<i>фізичні</i>			
- підвищена температура поверхонь обладнання	експлуатація ЕОМ, принтерів, сканерів чи/або серверного обладнання для роботи	2	[14]
- підвищений рівень шуму на робочому місці	-//-	2	[13]
- підвищений рівень вібрації	-//-	2	[14] [27]
- підвищена або знижена вологість повітря	-//-	2	[14]
- підвищена або знижена рухливість повітря	-//-	1	[14]
- підвищений рівень іонізуючого випромінювання в робочій зоні	-//-	2	[14] [24]
- підвищений рівень електромагнітного випромінювання	-//-	2	[24]
Продовження Табл. 4.3			
- підвищений рівень напруги електричної мережі, замикання якої може відбутися через тіло людини	-//-	4	[25] [14]
- підвищений рівень статичної електрики	-//-	2	[25]

Продовження Табл. 4.3			
- підвищена напруженість електричного поля	-//-	2	[24]
- підвищена напруженість магнітного поля	-//-	2	[24]
- недостатність природного світла	порушення умов праці (вимог до приміщень)	2	[11]
- недостатнє освітлення робочої зони	порушення гігієнічних параметрів виробничого середовища	3	[11]
- підвищена яскравість світла	порушення умов праці (організації місця праці-налагодження моніторів)	1	[12]
- понижена контрастність	-//-	1	[12]

<i>психофізіологічні:</i>			
- нервово-психічна перевантаження (розумове, перенапруження аналізаторів-зорових)	- пошук інформації для постановки теми; - пошук та аналіз аналогів і літератури; - пошук наявних технологій, моделювання та аналіз алгоритмів; - виконання роботи за темою диплома, тестування; - оформлення роботи	4	[26] [12]
- фізичні (статичне – сидіння)	порушення умов праці (організації місця праці - сидіння користувача) та організації робочого часу - безпервна робота)	2	[26] [12]

Робочі місця мають відповідати вимогам Державних санітарних правил і норм роботи з візуальними дисплейними терміналами електронно-обчислювальних машин, затверджених постановою Головного державного санітарного лікаря України від 10.12.98 N 7 [12].

4.4.6 Пожежна безпека

Небезпека розвитку пожежі на обчислювальному центрі обумовлюється застосуванням розгалужених систем електроживлення ЕОМ, вентиляції і кондиціонування.

Запобігти утворенню горючого середовища (замінити горючі речовини і матеріали на негорючі і важкогорючі) не надається технічно можливим. Тому проектом передбачаються способи і засоби запобігання утворення (або внесення) в горюче середовище джерел запалювання, таких як:

- застосування електроустаткування, відповідної пожежонебезпечної і вибухонебезпечної зонам відповідно до ПУЕ;
- застосування в конструкції швидкодійних засобів захисного відключення можливих джерел запалення;
- виключення можливості появи іскрового розряду в горючому середовищі з енергією, рівної і вище мінімальної енергії запалення.

Таке приміщення, площею 25 м², відноситься до категорії "В" (пожежонебезпечної) та для протипожежного захисту в ньому проектом передбачено устаткування автоматичною пожежною сигналізацією із застосуванням датчиків-сповіщувачів РІД-1 (сповіщувач димовий ізоляційний) в кількості 1 шт., і застосуванням первинних засобів пожежогасіння.

Простори усередині приміщень в межах, яких можуть утворюватися або знаходитися пожежонебезпечні речовини і матеріали відповідно до [18] відносяться до пожежонебезпечної зони класу П-Па. Це обумовлено тим, що в приміщенні знаходяться тверді горючі та важкозаймисті речовини та матеріали. Приміщенню, у якому розташоване робоче місце, присвоюється II ступень вогнестійкості.

Продуктами згорання, що виділяються на пожежі, є: окис вуглецю; сірчистий газ; окис азоту; синильна кислота; акромін; фосген; хлор і ін. При горінні пластмас, окрім звичних продуктів згорання, виділяються різні продукти термічного розкладання: хлорангідридні кислоти, формальдегіди, хлористий водень, фосген, синильна кислота, аміак, фенол, ацетон, стирол .

4.4.7 Електробезпека

На робочому місці виконуються наступні вимоги електробезпеки: ПК, периферійні пристрої та устаткування для обслуговування, електропроводи і кабелі за виконанням та ступенем захисту відповідають класу зони за ПУЕ (правила улаштування електроустановок), мають апаратуру захисту від струму короткого замикання та інших аварійних режимів. Лінія електромережі для живлення ПК, периферійних пристроїв і устаткування для обслуговування, виконана як окрема групова три- провідна мережа, шляхом прокладання фазового, нульового робочого та нульового захисного провідників. Нульовий захисний провідник використовується для заземлення (занулення) електроприймачів. Штепсельні з'єднання та електророзетки крім контактів фазового та нульового робочого провідників мають спеціальні контакти для підключення нульового захисного провідника. Електромережа штепсельних розеток для живлення персональних ПК укладено по підлозі поруч зі стінами відповідно до затвердженого плану розміщення обладнання та технічних характеристик обладнання. Металеві труби та гнучкі металеві рукави заземлені. Захисне заземлення включає в себе заземлюючих пристроїв і провідник, який з'єднує заземлюючий пристрій з обладнанням, яке заземлюється - заземлюючий провідник.

4.5 Гігієнічні вимоги до параметрів виробничого середовища

4.5.1 Мікроклімат

Мікроклімат робочих приміщень – це клімат внутрішнього середовища цих приміщень, що визначається діючої на організм людини з'єднанням температури, вологості, швидкості переміщення повітря. Оптимальні значення для температури,

відносної вологості й рухливості повітря для зазначеного робочого місця відповідають і наведені в Табл. 4.4:

Таблиця 4.4 – Норми мікроклімату робочої зони об'єкту

Період року	Категорія робіт	Температура С ⁰	Відносна вологість %	Швидкість руху повітря, м/с
Холодна	легка-1 а	22 - 24	40 – 60	0,1
Тепла	легка-1 а	23 - 25	40 – 60	0,1

Дане приміщення обладнане системами опалення, кондиціонування повітря або припливно-витяжною вентиляцією. У приміщенні на робочому місці забезпечуються оптимальні значення параметрів мікроклімату: температури, відносної вологості й рухливості повітря у відповідності до [13]. Рівні позитивних і негативних іонів у повітрі мають відповідати [13].

Контроль параметрів мікроклімату в холодний і теплий період року здійснюється не менше 3-х разів на зміну (на початку, середині, в кінці).

4.5.2 Освітлення

Світло є природною умовою існування людини. Воно впливає на стан вищих психічних функцій і фізіологічні процеси в організмі. Хороше освітлення діє тонізуюче, створює гарний настрій, покращує протікання основних процесів вищої нервової діяльності.

У приміщенні, де розташовані ЕОМ передбачається природне бічне освітлення, рівень якого відповідає [11]. Джерелом природного освітлення є сонячне світло. Регулярно повинен проводитися контроль освітленості, який підтверджує, що рівень освітленості задовольняє ДБН і для даного приміщення в світлий час доби достатньо природного освітлення.

Розрахунок освітлення.

Для виробничих та адміністративних приміщень світловий коефіцієнт приймається не менше $1/8$, в побутових – $1/10$:

$$S_b = \left(\frac{1}{5} \div \frac{1}{10} \right) \cdot S_n, \quad (4.1)$$

де S_b – площа віконних прорізів, m^2 ;

S_n – площа підлоги, m^2 .

$$S_n = a \cdot b = 5 \cdot 5 = 25 \text{ м}^2,$$

$$S = 1/8 \cdot 25 = 3,125 \text{ м}^2.$$

Приймаємо 2 вікна площею $S=1,6 \text{ м}^2$ кожне.

Розрахунок штучного освітлення виробляється по коефіцієнтах використання світлового потоку, яким визначається потік, необхідний для створення заданої освітленості при загальному рівномірному освітленні. Розрахунок кількості світильників n виробляється по формулі (4.2):

$$n = \frac{E \cdot S \cdot Z \cdot K}{F \cdot U \cdot M}, \quad (4.2)$$

де E – нормована освітленість робочої поверхні, визначається нормами – 300 лк;

S – освітлювана площа, m^2 ; $S = 25 \text{ м}^2$;

Z – поправочний коефіцієнт світильника ($Z = 1,15$ для ламп розжарювання та ДРЛ; $Z = 1,1$ для люмінесцентних ламп) приймаємо рівним 1,1;

K – коефіцієнт запасу, що враховує зниження освітленості в процесі експлуатації – 1,5;

U – коефіцієнт використання, залежний від типу світильника, показника індексу приміщення і т.п. – 0,575

M – число люмінесцентних ламп в світильнику – 2;

F – світловий потік лампи – 5400лм (для ЛБ-80).

Підставивши числові значення у формулу (А.2), отримуємо:

$$n = \frac{300 \cdot 25 \cdot 1,1 \cdot 1,5}{5400 \cdot 0,575 \cdot 2} \approx 2,0 \quad (4.3)$$

Приймаємо освітлювальну установку, яка складається з 2-х світильників, які складаються з двох люмінесцентних ламп потужністю 160 Вт, напругою – 220 В.

4.5.3 Шум та вібрація, електромагнітне випромінювання

Рівень шуму, що супроводжує роботу користувачів персональних комп'ютерів (зумовлений як роботою системних блоків, клавіатури, так і друкуванням на принтерах, а також зовнішніми чинниками), коливається у межах 50–65 дБА [13]. У залах опрацювання інформації та комп'ютерного набору рівні шуму не повинні перевищувати 65 дБА.

Віброізоляція можливо здійснювати за допомогою спеціальної прокладки під системний блок, який послаблює передачу вібрацій робочого столу. Вібрація на робочому місці в приміщенні, що розглядається, відповідає нормам [13].

4.5.4 Вентилювання

У приміщенні, де знаходяться ЕОМ, повітрообмін реалізується за допомогою природної організованої вентиляції (вентиляційні шахти), тобто при V приміщення > 40 м³ на одного працюючого допускається природна вентиляція. Цей метод забезпечує приток потрібної кількості свіжого повітря, що визначається в СНіП.

Також має здійснюватися провітрювання приміщення, в залежності від погодних умов, тривалість повинна бути не менше 10 хв. Найкращий обмін повітря здійснюється при наскрізному провітрюванні.

4.6 Заходи з організації виробничого середовища та попередження виникнення надзвичайних ситуацій

Відповідно до санітарно-гігієнічних нормативів та правил експлуатації обладнання наводимо приклади деяких заходів безпеки.

1) Заходи безпеки під час експлуатації персонального комп'ютера та периферійних пристроїв передбачають:

- правильне організування місця праці та дотримання оптимальних режимів праці та відпочинку під час роботи з ПК;
- експлуатацію сертифікованого обладнання;
- дотримання заходів електробезпеки;
- забезпечення оптимальних параметрів мікроклімату;
- забезпечення раціонального освітлення місця праці (освітленість робочого місця не перевищувала 2/3 нормальної освітленості приміщення);
- облаштовуючи приміщення для роботи з ПК, потрібно передбачити припливно-втяжну вентиляцію або кондиціювання повітря.

2) Заходи безпеки під час експлуатації інших електричних приладів передбачають дотримання таких правил:

- постійно стежити за справним станом електромережі, розподільних щитків, вимикачів, штепсельних розеток, лампових патронів, а також мережевих кабелів живлення, за допомогою яких електроприлади під'єднують до електромережі;
- постійно стежити за справністю ізоляції електромережі та мережевих кабелів, не допускаючи їхньої експлуатації з пошкодженою ізоляцією;
- не тягнути за мережевий кабель, щоб витягти вилку з розетки;
- не закривати меблями, різноманітним інвентарем вимикачі, штепсельні розетки;

- не підключати одночасно декілька потужних електропристроїв до однієї розетки, що може викликати надмірне нагрівання провідників, руйнування їхньої ізоляції, розплавлення і загоряння полімерних матеріалів;
- не залишати включені електроприлади без нагляду;
- не допускати потрапляння всередину електроприладів крізь вентиляційні отвори рідин або металевих предметів, а також не закривати їх та підтримувати в належній чистоті, щоб уникнути перегрівання та займання приладу;
- не ставити на електроприлади матеріали, які можуть під дією теплоти, що виділяється, загорітися (канцелярські товари, сувенірну продукцію тощо).

4.7 Вимоги безпеки при надзвичайних ситуаціях:

- 1) При раптовому припиненні подачі електричної енергії вимкнути всі пристрої ПК в такій послідовності: периферійні пристрої, ВДТ, системний блок, стабілізатор (або блок безперервного живлення). Витягнути вилки з розеток. При наявності ознак горіння (дим, запах горілого) необхідно вимкнути всі пристрої ПК, знайти місце загоряння і виконати всі можливі заходи для його ліквідації, попередивши терміново про це керівництво.
- 2) При замиканні, перевантаженні електричного струму на електричному обладнанні, внаслідок ураження грозової блискавки та ймовірної небезпеки ураженням електричним струмом, приймають наступне:
 - попередження замикання здійснюється правильним вибором, монтажем експлуатації мереж;
 - застосування захисту схем у вигляді швидкодіючих реле, а також вимикачів, плавких запобіжників.

Також застосовують різні **електричні захисні засоби від ураження струмом:**

а) Ізолюючі - ізолюють людини від струмоведучих або заземлених частин, а так-же від землі. Вони діляться на основні та додаткові.

б) *Основні* - володіють ізоляцією, здатної довго витримувати робоче напругу електроустановки і тому ними дозволяється стосуватися струмоведучих частин, знаходячи-трудящих під напругою.

в) *Запобіжні* - володіють ізоляцією нездатною витримати робоча напруга електроустановки, і тому вони не можуть самостійно захищати людину від ураження струмом під цим напругою.

4.8 Розрахунок захисного заземлення (забезпечення електробезпеки будівлі).

Згідно з класифікацією приміщень за ступенем небезпеки ураження електричним струмом [19], приміщення в якому проводяться всі роботи відноситься до першого класу (без підвищеної небезпеки). Коефіцієнт використання вертикальних заземлювачів η_v в залежності від розміщення заземлювачів та їх кількості знаходиться в межах 0,4...0,99. Взаємну екрануючу дію горизонтального заземлювача (з'єднувальної смуги) враховують за допомогою коефіцієнта використання горизонтального заземлювача η_c .

Послідовність розрахунку.

1) Визначається необхідний опір штучних заземлювачів $R_{шт.з.}$:

$$R_{шт.з.} = \frac{R_d \cdot R_{пр.з.}}{R_{пр.з.} - R_d}, \quad (4.4)$$

де $R_{пр.з.}$ – опір природних заземлювачів; R_d – допустимий опір заземлення. Якщо природні заземлювачі відсутні, то $R_{шт.з.} = R_d$.

Підставивши числові значення у формулу (А.3), отримуємо:

$$R_{шт.з.} = \frac{4 \cdot 40}{40 - 4} \approx 4 \text{ Ом} \quad (4.5)$$

- 2) Опір заземлення в значній мірі залежить від питомого опору ґрунту ρ , Ом·м. Приблизне значення питомого опору глини приймаємо $\rho=40$ Ом·м (табличне значення).
- 3) Розрахунковий питомий опір ґрунту, $\rho_{\text{розр}}$, Ом·м, визначається відповідно для вертикальних заземлювачів $\rho_{\text{розр.в}}$, і горизонтальних $\rho_{\text{розр.г}}$, Ом·м за формулою:

$$\rho_{\text{розр.}} = \psi \cdot \rho, \quad (4.6)$$

де ψ – коефіцієнт сезонності для вертикальних заземлювачів І кліматичної зони з нормальною вологістю землі, приймається для вертикальних заземлювачів $\rho_{\text{розр.в}}=1,7$ і горизонтальних $\rho_{\text{розр.г}}=5,5$ Ом·м.

$$\rho_{\text{розр.в}} = 1,7 \cdot 40 = 68 \text{ Ом}\cdot\text{м} \quad (4.7)$$

$$\rho_{\text{розр.г}} = 5,5 \cdot 40 = 220 \text{ Ом}\cdot\text{м} \quad (4.8)$$

- 4) Розраховується опір розтікання струму вертикального заземлювача $R_{\text{в}}$, Ом, за (5.5).

$$R_{\text{в}} = \frac{\rho_{\text{розр.в}}}{2 \cdot \pi \cdot l_{\text{в}}} \cdot \left(\ln \frac{2 \cdot l_{\text{в}}}{d_{\text{ст}}} + \frac{1}{2} \cdot \ln \frac{4 \cdot t + l_{\text{в}}}{4 \cdot t - l_{\text{в}}} \right), \quad (4.9)$$

де $l_{\text{в}}$ – довжина вертикального заземлювача (для труб - 2–3 м; $l_{\text{в}}=3$ м);

$d_{\text{ст}}$ – діаметр стержня (для труб - 0,03–0,05 м; $d_{\text{ст}}=0,05$ м);

t – відстань від поверхні землі до середини заземлювача, яка визначається за ф. (5.6):

$$t = h_{\text{в}} + \frac{l_{\text{в}}}{2}, \quad (4.10)$$

де $h_{\text{в}}$ – глибина закладання вертикальних заземлювачів (0,8 м); тоді $t = 0,8 + \frac{3}{2} = 2,3$ м

$$R_{\text{в}} = \frac{68}{2 \cdot \pi \cdot 3} \cdot \left(\ln \frac{2 \cdot 3}{0,05} + \frac{1}{2} \cdot \ln \frac{4 \cdot 2,3 + 3}{4 \cdot 2,3 - 3} \right) = 18,5 \text{ Ом} \quad (4.11)$$

- 5) Визначається теоретична кількість вертикальних заземлювачів n штук, без урахування коефіцієнта використання $\eta_{\text{в}}$:

$$n = \frac{2 \cdot R_B}{R_d} = \frac{2 \cdot 18,5}{4} = 9,25 \quad (4.12)$$

І визначається коефіцієнт використання вертикальних електродів групового заземлювача без врахування впливу з'єднувальної стрічки $\eta_B = 0,57$ (табличне значення).

- б) Визначається необхідна кількість вертикальних заземлювачів з урахуванням коефіцієнта використання n_B , шт:

$$n_B = \frac{2 \cdot R_B}{R_d \cdot \eta_B} = \frac{2 \cdot 18,5}{4 \cdot 0,57} = 16,2 \approx 16 \quad (4.13)$$

- 7) Визначається довжина з'єднувальної стрічки горизонтального заземлювача l_c , м:

$$l_c = 1,05 \cdot L_B \cdot (n_B - 1), \quad (4.14)$$

де L_B – відстань між вертикальними заземлювачами, (прийняти за $L_B = 3$ м);

n_B – необхідна кількість вертикальних заземлювачів.

$$l_c = 1,05 \cdot 3 \cdot (16 - 1) \approx 48 \text{ м} \quad (4.15)$$

- 8) Визначається опір розтіканню струму горизонтального заземлювача (з'єднувальної стрічки) R_r , Ом:

$$R_r = \frac{\rho_{\text{розр.г}}}{2 \cdot \pi \cdot l_c} \cdot \ln \frac{2 \cdot l_c^2}{d_{\text{см}} \cdot h_r}, \quad (4.20)$$

де $d_{\text{см}}$ – еквівалентний діаметр смуги шириною b , $d_{\text{см}} = 0,95b$, $b = 0,15$ м;

h_r – глибина закладання горизонтальних заземлювачів (0,5 м);

l_c – довжина з'єднувальної стрічки горизонтального заземлювача l_c , м

$$R_r = \frac{220}{2 \cdot \pi \cdot 48} \cdot \ln \frac{2 \cdot 48^2}{0,95 \cdot 0,15 \cdot 0,5} = 8,1 \text{ Ом} \quad (4.21)$$

- 9) Визначається коефіцієнт використання горизонтального заземлювача η_c , відповідно до необхідної кількості вертикальних заземлювачів n_B .

Коефіцієнт використання з'єднувальної смуги $\eta_c=0,3$ (табличне значення).

- 10) Розраховується результуючий опір заземлювального електроду з урахуванням з'єднувальної смуги:

$$R_{\text{заг}} = \frac{R_B \cdot R_\Gamma}{R_B \cdot \eta_c + R_\Gamma \cdot n_B \cdot \eta_B} \leq R_d. \quad (4.22)$$

Висновок: дане захисне заземлення буде забезпечувати електробезпеку будівлі, так як виконується умова: $R_{\text{заг}} < 4 \text{ Ом}$, а саме:

$$R_{\text{заг}} = \frac{18,5 \cdot 8,1}{18,5 \cdot 0,3 + 8,1 \cdot 16 \cdot 0,57} = 1,9 \leq R_d \quad (4.23)$$

- 3) При виникненню пожеж при роботі на ПЕОМ від таких можливими джерел запалювання як:
- іскри і дуги коротких замикань;
 - перегрів провідників, резисторів та інших радіодеталей ПЕОМ, від тривалої перевантаження та наявності перехідного опору;
 - іскри при розмиканні і розмиканні ланцюгів;
 - розряди статичної електрики;
 - необережному поводженню з вогнем, а також вибухи газо-повітряних і паро-повітряних сумішей.

4.9 Охорона навколишнього природного середовища

4.9.1 Загальні дані з охорони навколишнього природного середовища

Діяльність за темою магістерської роботи, а саме, процес виконання якої впливає на навколишнє природне середовище і регламентується нормами діючого законодавства: Законом України «Про охорону навколишнього природного середовища», Законом

України «Про забезпечення санітарного та епідемічного благополуччя населення», Законом України «Про відходи», Законом України «Про охорону атмосферного повітря», Законом України «Про захист населення і територій від надзвичайних ситуацій техногенного та природного характеру», Водний кодекс України.

Основним екологічним аспектом в процесі діяльності за даними спеціальностями є процеси впливу на атмосферне повітря та процеси поводження з відходами, які утворюються, збираються, розміщуються, передаються на знешкодження, утилізацію, тощо в ІТ галузі.

В процесі діяльності виявлення проблем при роботі з даними з соціальних мереж та підвищення точності емоційної класифікації виникають процеси поводження з відходами ІТ галузі. Нижче надано перелік відходів, що утворюються в процесі роботи:

- Відпрацьовані люмінесцентні лампи - I клас небезпеки
- Батарейки та акумулятори (малі) -III клас небезпеки
- Відходи друкуючих пристроїв - IV клас небезпеки
- Макулатура - IV клас небезпеки
- Матеріали пакувальні пластмасові забруднені (ємності з-під тонеру, фарби, інш.) - IV клас небезпеки
- Побутові відходи - IV клас небезпеки

4.9.2 Вимоги до збору, пакування та розміщення відходів ІТ галузі

Наводяться вимоги зберігання виявлених за своєю роботою відходів відповідно до вимог Державних санітарних правил і норм [56].

Відходи в міру їх накопичення збирають у тару, відповідну класу небезпеки, з дотриманням правил безпеки, після чого доставляють до місця тимчасового зберігання відходів відповідно до затвердженої схеми їх розміщення. Зазначені для зберігання відходів місця чи об'єкти повинні використовуватися лише для заявлених відходів.

Не допускається зберігання відходів у невстановлених схемою місцях, а також перевищення норм тимчасового зберігання відходів.

Способи тимчасового зберігання відходів визначаються видом, агрегатним станом і класом небезпеки відходів:

- Відходи I класу небезпеки зберігаються в герметичній тарі (сталеві бочки, контейнери). У міру наповнення тару з відходами закривають герметично сталевий кришкою;
- Відходи II класу небезпеки в залежності від агрегатного стану зберігаються в поліетиленових мішках, бочках, сховищах та інших видах тари, яка запобігає поширенню шкідливих речовин;
- Відходи III класу небезпеки зберігаються в тарі, яка забезпечує локалізацію зберігання, дозволяє виконувати вантажно-розвантажувальні і транспортні роботи і виключає поширення в ОС шкідливих речовин;
- Відходи IV класу небезпеки можуть зберігатися відкрито на промисловому майданчику у вигляді конусоподібної купи, звідки їх автотранспортом перевозять у самоскид і доставляють на місце утилізації або захоронення;

Не допускається змішування відходів різних видів і класів небезпеки з будівельними і побутовими відходами, дерев'яної, металевої, синтетичної тари, відходами текстильних матеріалів (старий спецодяг, ганчірки) і ін.

Особливий контроль наділяється збору і зберіганню відпрацьованих ртутьвмісних ламп (енергоощадних) як відходам I класу небезпеки, що збираються і обов'язково передаються на утилізацію підприємствам, що мають ліцензію на поводження з такими небезпечними відходами.

Всі відходи, що утворюються в процесі діяльності/роботи, підлягають обліку.

Вимоги безпеки при поводженні з відходами:

Під час роботи з відходами (прибирання виробничих приміщень, збір і сортування, навантаження, транспортування, розвантаження та ін.) працівники повинні бути забезпечені засобами індивідуального захисту та дотримуватися вимог інструкцій з охорони праці, що діють на підприємстві.

Наведено перелік деяких відходів, які передаються на утилізацію організаціям, які мають ліцензію на поводження з відходами як вторинної сировини:

- Макулатура;
- Матеріали пакувальні вторинні

Відвантаження таких відходів здійснюється відповідно до договору (контракту).

Побутові та будівельні відходи вивозяться на полігон твердих побутових відходів міста, також відповідно до договору з комунальним дорожньо-експлуатаційним управлінням.

Особи, винні в порушенні встановленого порядку поводження з відходами (порушення правил обліку відходів, самовільне складування і видалення відходів, передача відходів в інші підприємства/організації з порушенням встановлених правил), згідно законодавства несуть дисциплінарну, адміністративну або кримінальну відповідальність.

4.9.3 Визначення впливу та заходів щодо поводження з відходами ІТ галузі

З метою визначення та прогнозування впливу відходів на навколишнє середовище, своєчасного виявлення негативних наслідків, їх запобігання відповідно до Закону України «Про відходи» повинен здійснюватися моніторинг місць утворення, зберігання, і видалення відходів.

Відомості про місце утворення та місце розташування відходів зазначаються на «План схемі місці розміщення відходів організації / виробництва».

4.10 Висновки до розділу

Будь-який керівник і управлінець повинен піклується про свій персонал. Трудові ресурси одні з основних ресурсів в економіці підприємства, від їх якості залежить економічний ефект підприємства. Тому збереження і створення сприятливих умов робочим - це одне з першочергових завдань управлінців.

Одна з найважливіших потреб людини - це безпека. Працівник для того щоб приносити максимальний економічний ефект в обов'язковому порядку повинен відчувати себе в безпеці. Тому керівники повинні створювати всі умови для забезпечення цієї безпеки. Але в сучасному світі трудова діяльність безпосередньо пов'язана з небезпечними факторами і травматизмом.

Для запобігання різного роду нещасних випадків та соціального захисту працівників управлінець повинен ретельно продумувати політику охорони праці на підприємстві. При існуванні високого рівня охорони праці на підприємстві працівник буде ефективніше працювати. Охорона праці складається з безлічі різних елементів, одним з основних її структурних частин є навчання з охорони праці.

Необхідно в обов'язковому порядку навчати персонал охорони праці. Навчання охорони праці - це певна система заходів, що дозволяє передбачити різного роду негативні умови і підготувати людину до дії, якщо він потрапить в дані негативні умови.

Закон про основи охорони праці цілком відповідає вимогам сучасності. У ньому чітко прописано, що якщо на підприємстві працює понад 100 чоловік, то там повинна бути створена служба охорони праці (форма власності не має значення). Якщо в організації - менше ста людей, то роботодавець повинен укласти договір або з фахівцем з охорони праці, або з організацією, яка має право працювати в цій галузі.

У число прямих обов'язків роботодавця, відповідно до законодавства, входить забезпечення безпеки працівника при будівельних роботах, здійсненні технологічних процесів, використання робочих інструментів і матеріалів. При необхідності обов'язково застосування засобів колективного та індивідуального захисту. Для кожного робочого місця вводяться свої стандарти безпеки, і співробітники проходять обов'язковий курс навчання безпечним методам і прийомам робіт. Їх підготовленість перевіряється спеціальними контролюючими інстанціями. Також повинні регулярно проводитися медичні огляди.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. Вендров, А.М. CASE-технологии. Современные методы и средства проектирования информационных систем/ А.М. Вендров. - М.: Финансы и статистика, 1998.–176 с.
2. Маклаков, С.В. BPWin и ERWin. Case-средства разработки информационных систем/ С.В.Маклаков-М.: ДИАЛОГ–МИФИ, 1999.–256с.
3. Орлов, С.А. Технологии разработки программного обеспечения/ С.А. Орлов–СПб.: Питер, 2002.–464 с.
4. Гарсиа-Молина Г., Ульман Дж., Уидом Дж. Системы баз данных. Полный курс / Гарсиа-Молина Г, Ульман Дж, Уидом Дж. — М.: "Вильямс", 2003. – 229 с.
5. Дейт. К. Дж. Введение в системы баз данных / К. Дж. Дейт. — "Вильямс", 2001. – 426 с.
6. Харрингтон Д. Л Проектирование реляционных баз данных. Просто и доступно / Д. Л. Харрингтон. – М.: ЛОРИ, 2000. – 277 с.
7. Коннолли Т. М, Бегг К. Базы данных. Проектирование, реализация и сопровождение. Теория и практика / Т. М. Коннолли, К. Бегг. – М.: Издательский дом "Вильямс", 2003. – 261 с.
8. Калянов Г. Н. CASE. Структурный системный анализ (автоматизация и применение) / Г. Н. Калянов. – М.: "Лори", 2006. – 175 с.
9. Черемных, С.В. Структурный анализ систем: IDEF-технологии. /С.В. Черемных, И.О.Семенов, В.С. Ручкин-М.: Финансы и статистика, 2003.–208 с.
10. ГОСТ 12.1.044-89 ССБТ. Пожежовибухонебезпека речовин і матеріалів. Номенклатура показників і методи їх визначення
11. ДБН В.2.5-28:2015 Природне і штучне освітлення
12. ДСанПіН 3.3.2.007-98 Гігієнічні вимоги до організації роботи з візуальними дисплейними терміналами електронно-обчислювальних машин
13. ДСН 3.3.6.037-99 Санітарні норми виробничого шуму, ультразвуку та інфразвуку
14. ДСН 3.3.6.042-99 Санітарні норми мікроклімату виробничих

15. НПАОП 0.00-4.12-05 Типове положення про порядок проведення навчання і перевірки знань з питань охорони праці
16. НПАОП 0.00-4.15-98 Про розробку інструкцій з охорони праці
17. НПАОП 0.00-6.03-93 Порядок опрацювання та затвердження власником нормативних актів про охорону праці
18. НАПБ Б.03.002-2007 Норми визначення категорій приміщень, будинків та зовнішніх установок за вибухопожежною та пожежною безпекою
19. НПАОП 40.1-1.01-97 Правила безопасной эксплуатации электроустановок
20. НПАОП 40.1-1.32-01 Правила устройства электроустановок. Электрооборудование специальных установок
21. ДСН 3.3.6.039-99 Санітарні норми виробничої загальної та локальної вібрації
22. ДСТУ ГОСТ 12.1.012-90 ССБТ. Вібраційна безпека. Загальні вимоги
23. ДБН В.2.5-67:2013 Опалення, вентиляція та кондиціонування
24. ГОСТ 12.1.006-84 ССБТ. Електромагнітні поля радіочастот. Загальні вимоги безпеки. Допустимі рівні на робочих місцях і вимоги до проведення контролю
25. ГОСТ 12.1.030-81 ССБТ. Електробезпечність. Захисне заземлення. Занулення
26. НПАОП 0.00-1.28-10 Правила охорони праці під час експлуатації електронно-обчислювальних машин

ВИСНОВКИ

Метою даної роботи було проаналізувати предмету область та розробити програму для обміну інформацією в розподіленій мережі.

Для досягнення мети були визначені і вирішені наступні завдання:

- Проведено дослідження в предметній області за останні 5 років;
- Зроблено аналіз найпопулярніших програмних продуктів, які вирішують дану проблему;
- Була обрана стратегія розробки;
- Зібравши інформацію про підприємство та проблемну область, було проведено статистичний аналіз;
- На базі статистичного аналізу була побудована математична модель системи;
- Встановлено основні напрями розробки на базі аналізу математичної моделі;
- Вивчено основні підходи до реалізації даної проблеми, так обрано найбільш продуктивний;
- Створено графічний інтерфейс для серверної та клієнтської частини;
- Розроблена практична частина для реалізації основних функцій програми.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Агуров, Павел С#. Сборник рецептов / Павел Агуров. - М.: "БХВ-Петербург", 2012. - 432 с.
2. Албахари, Джозеф С# 3.0. Справочник / Джозеф Албахари, Бен Албахари. - М.: БХВ-Петербург, 2012. - 944 с.
3. Албахари, Джозеф С# 3.0. Справочник / Джозеф Албахари, Бен Албахари. - М.: БХВ-Петербург, 2013. - 944 с.
4. Альфред, В. Ахо Компиляторы. Принципы, технологии и инструментарий / Альфред В. Ахо и др. - М.: Вильямс, 2015. - 266 с.
5. Бишоп, Дж. С# в кратком изложении / Дж. Бишоп, Н. Хорспул. - М.: Бином. Лаборатория знаний, 2013. - 472 с.
6. Вагнер, Билл С# Эффективное программирование / Билл Вагнер. - М.: ЛОРИ, 2013. - 320 с.
7. Зиборов, В.В. Visual С# 2012 на примерах / В.В. Зиборов. - М.: БХВ-Петербург, 2013. - 480 с.
8. Зиборов, Виктор Visual С# 2010 на примерах / Виктор Зиборов. - М.: "БХВ-Петербург", 2011. - 432 с.
9. Ишкова, Э. А. Самоучитель С#. Начала программирования / Э.А. Ишкова. - М.: Наука и техника, 2013. - 496 с.
10. Касаткин, А. И. Профессиональное программирование на языке си. Управление ресурсами / А.И. Касаткин. - М.: Высшая школа, 2012. - 432 с.
11. Лотка, Рокфорд С# и CSLA .NET Framework. Разработка бизнес-объектов / Рокфорд Лотка. - М.: Вильямс, 2010. - 816 с.
12. Мак-Дональд, Мэтью Silverlight 5 с примерами на С# для профессионалов / Мэтью Мак-Дональд. - М.: Вильямс, 2013. - 848 с.
13. Марченко, А. Л. Основы программирования на С# 2.0 / А.Л. Марченко. - М.: Интернет-университет информационных технологий, Бином. Лаборатория знаний, 2011. - 552 с.
14. Подбельский, В. В. Язык С#. Базовый курс / В.В. Подбельский. - М.: Финансы и статистика, Инфра-М, 2011. - 384 с.
15. Прайс, Джейсон Visual С# 2.0. Полное руководство / Джейсон Прайс, Майк Гандэрлой. - М.: Век +, Корона-Век, Энтроп, 2010. - 736 с.

16. Рихтер, Джеффри CLR via C#. Программирование на платформе Microsoft .NET Framework 4.0 на языке C# / Джеффри Рихтер. - М.: Питер, 2013. - 928 с.
17. Смоленцев, Н. К. MATLAB. Программирование на Visual C#, Borland JBuilder, VBA (+ CD-ROM) / Н.К. Смоленцев. - М.: ДМК Пресс, 2011. - 456 с.
18. Троелсен, Эндрю Язык программирования C# 5.0 и платформа .NET 4.5 / Эндрю Троелсен. - М.: Вильямс, 2015. - 486 с.
19. Троелсен, Эндрю Язык программирования C# 2008 и платформа .NET 3.5 / Эндрю Троелсен. - М.: Вильямс, 2010. - 370 с.
20. Фримен, Адам ASP.NET MVC 3 Framework с примерами на C# для профессионалов / Адам Фримен , Стивен Сандерсон. - М.: Вильямс, 2011. - 672 с.
21. Campbell Parallel Programming with Microsoft® Visual C++® / Campbell. - Москва: Гостехиздат, 2011. - 784 с.
22. Альфред, В. Ахо Компиляторы. Принципы, технологии и инструментарий / Альфред В. Ахо и др. - Москва: Высшая школа, 2015. - 882 с.
23. Балена, Франческо Современная практика программирования на Microsoft Visual Basic и Visual C# / Франческо Балена , Джузеппе Димауро. - М.: Русская Редакция, 2015. - 640 с.
24. Боровский, А. C++ и Pascal в Kylix 3. Разработка интернет-приложений и СУБД / А. Боровский. - М.: БХВ-Петербург, 2015. - 544 с.
25. Давыдов, В. Visual C++. Разработка Windows-приложений с помощью MFC и API-функций / В. Давыдов. - М.: БХВ-Петербург, 2014. - 576 с.
26. Довбуш, Галина Visual C++ на примерах / Галина Довбуш , Анатолий Хомоненко. - М.: БХВ-Петербург, 2012. - 528 с.
27. Зиборов, В. MS Visual C++ 2010 в среде .NET / В. Зиборов. - М.: Питер, 2012. - 320 с.
28. Кетков, Юлий Практика программирования: Visual Basic, C++ Builder, Delphi. Самоучитель (+ дискета) / Юлий Кетков , Александр Кетков. - М.: БХВ-Петербург, 2012. - 464 с.
29. Мешков, А. Visual C++ и MFC / А. Мешков, Ю. Тихомиров. - М.: БХВ-Петербург, 2013. - 546 с.
30. Неформальное введение в C++ и Turbo Vision. - Москва: ИЛ, 2010. - 384 с.
31. Панюкова, Т. А. Языки и методы программирования. Создание простых GUI-приложений с помощью Visual C++. Учебное пособие / Т.А. Панюкова, А.В. Панюков. - Москва: Мир, 2015. - 144 с.

32. Пахомов, Б. C/C++ и MS Visual C++ 2010 для начинающих / Б. Пахомов. - М.: БХВ-Петербург, 2011. - 736 с.
33. Пахомов, Борис C/C++ и MS Visual C++ 2012 для начинающих / Борис Пахомов. - Москва: СИНТЕГ, 2015. - 518 с.
34. Пахомов, Борис C/C++ и MS Visual C++ 2012 для начинающих / Борис Пахомов. - М.: "БХВ-Петербург", 2013. - 502 с.
35. Полубенцева, М. C/C++. Процедурное программирование / М. Полубенцева. - М.: БХВ-Петербург, 2014. - 448 с.
36. Поляков, А. Методы и алгоритмы компьютерной графики в примерах на Visual C++ / А. Поляков, В. Брусенцев. - М.: БХВ-Петербург, 2011. - 560 с.
37. Понамарев, В. Программирование на C++/C# в Visual Studio .NET 2003 / В. Понамарев. - М.: БХВ-Петербург, 2015. - 917 с.
38. Роберт, С. Сикорд Безопасное программирование на C и C++ / Роберт С. Сикорд. - Москва: РГГУ, 2014. - 496 с.
39. Секунов, Н. Программирование на C++ в Linux / Н. Секунов. - М.: БХВ-Петербург, 2016. - 425 с.
40. Сидорина, Татьяна Самоучитель Microsoft Visual Studio C++ и MFC / Татьяна Сидорина. - М.: "БХВ-Петербург", 2014. - 848 с.

Додаток А

Лістинг основних функцій програми сервера

```
1      int InstallService() {
2          SC_HANDLE      hSCManager      =      OpenSCManager(NULL,      NULL,
3      SC_MANAGER_CREATE_SERVICE);
4          if(!hSCManager) {
5              addLogMessage("Error: Can't open Service Control Manager");
6              return -1;
7          }
8
9          SC_HANDLE hService = CreateService(
10             hSCManager,
11             serviceName,
12             serviceName,
13             SERVICE_ALL_ACCESS,
14             SERVICE_WIN32_OWN_PROCESS,
15             SERVICE_DEMAND_START,
16             SERVICE_ERROR_NORMAL,
17             servicePath,
18             NULL, NULL, NULL, NULL, NULL
19         );
20
21         if(!hService) {
```

```
22     int err = GetLastError();
23
24     switch(err) {
25
26         case ERROR_ACCESS_DENIED:
27
28             addLogMessage("Error: ERROR_ACCESS_DENIED");
29
30             break;
31
32         case ERROR_CIRCULAR_DEPENDENCY:
33
34             addLogMessage("Error: ERROR_CIRCULAR_DEPENDENCY");
35
36             break;
37
38         case ERROR_DUPLICATE_SERVICE_NAME:
39
40             addLogMessage("Error: ERROR_DUPLICATE_SERVICE_NAME");
41
42             break;
43
44         case ERROR_INVALID_HANDLE:
45
46             addLogMessage("Error: ERROR_INVALID_HANDLE");
47
48             break;
49
50         case ERROR_INVALID_NAME:
51
52             addLogMessage("Error: ERROR_INVALID_NAME");
53
54             break;
55
56         case ERROR_INVALID_PARAMETER:
57
58             addLogMessage("Error: ERROR_INVALID_PARAMETER");
59
60             break;
61
62         case ERROR_INVALID_SERVICE_ACCOUNT:
63
64             addLogMessage("Error: ERROR_INVALID_SERVICE_ACCOUNT");
65
66             break;
```

```
45         case ERROR_SERVICE_EXISTS:
46             addLogMessage("Error: ERROR_SERVICE_EXISTS");
47             break;
48         default:
49             addLogMessage("Error: Undefined");
50     }
51     CloseServiceHandle(hSCManager);
52     return -1;
53 }
54 CloseServiceHandle(hService);
55 CloseServiceHandle(hSCManager);
56 addLogMessage("Success install service!");
57 return 0;
58 }
59
60 Int StartService() {
61     SC_HANDLE hSCManager = OpenSCManager(NULL, NULL,
62     SC_MANAGER_CREATE_SERVICE);
63     SC_HANDLE hService = OpenService(hSCManager, serviceName,
64     SERVICE_START);
65     if(!StartService(hService, 0, NULL)) {
66         CloseServiceHandle(hSCManager);
67         addLogMessage("Error: Can't start service");
68         return -1;
```

```
69     }
70
71     CloseServiceHandle(hService);
72     CloseServiceHandle(hSCManager);
73     Return 0;
74 } Int t_main ()
75 {
76     char buff[1024];
77     printf("TCP SERVER \n");
78     if (WSAStartup(0x0202, (WSADATA *)&buff[0]))
79     {
80         printf("Error WSAStartup %d\n", WSAGetLastError());
81         return -1;
82     }
83     SOCKET mysocket;
84     if ((mysocket = socket(AF_INET, SOCK_STREAM, 0)) < 0)
85     {
86         printf("Error socket %d\n", WSAGetLastError());
87         WSACleanup();
88         return -1;
89     }
90
91     sockaddr_in local_addr;
```

```
92     local_addr.sin_family = AF_INET;
93     local_addr.sin_port = htons(MY_PORT);
94     local_addr.sin_addr.s_addr = 0;
95     if (bind(mysocket, (sockaddr *)&local_addr, sizeof(local_addr)))
96     {
97         printf("Error bind %d\n", WSAGetLastError());
98         closesocket(mysocket);
99         WSACleanup();
100        return -1;
101    }
102    if (listen(mysocket, 0x100))
103    {
104        printf("Error listen %d\n", WSAGetLastError());
105        closesocket(mysocket);
106        WSACleanup();
107        return -1;
108    }
109
110    printf("ozidanie podkluceniy...\n");
111
112    SOCKET client_socket;
113    sockaddr_in client_addr;
114    int client_addr_size = sizeof(client_addr);
```

```
115     while ((client_socket = accept(mysocket, (sockaddr *)&client_addr, \
116         &client_addr_size)))
117     {
118         nclients++;
119         HOSTENT *hst;
120         hst = gethostbyaddr((char *)&client_addr.sin_addr.s_addr, 4, AF_INET);
121
122         printf("+%s [%s] new connect!\n",
123             (hst) ? hst->h_name : "", inet_ntoa(client_addr.sin_addr));
124         PRINTNUSERS
125         DWORD thID;
126         CreateThread(NULL, NULL, ToClient, &client_socket, NULL, &thID);
127     }
128     return 0;
129 }
130
131 DWORD WINAPI ToClient(LPVOID client_socket)
132 {
133     SOCKET my_sock;
134     my_sock = ((SOCKET *)client_socket)[0];
135     char buff[BUFSIZ]; //char buff[20 * 1024];
136     #define SHELLLO "SOCKET PODKLUCHEN\r\n"
137     send(my_sock, SHELLLO, sizeof(SHELLLO), 0);
138     int bytes_recv;
```

```

138     while ((bytes_recv = recv(my_sock, &buff[0], sizeof(buff), 0)) &&
139     bytes_recv != SOCKET_ERROR)
140         /*tmp_buff = buff[0];
141         //changeWords();
142
143     send(my_sock, &buff[0], bytes_recv, 0);
144     nclients--;
145     printf("-disconnect\n"); PRINTNUSERS
146
147     closesocket(my_sock);
148     return 0;
149 int RemoveService ()
150 SC_HANDLE hSCManager = OpenSCManager(NULL, NULL,
151 SC_MANAGER_ALL_ACCESS);
152 if(!hSCManager) {
153     addLogMessage("Error: Can't open Service Control Manager");
154     return -1;
155 }
156 SC_HANDLE hService = OpenService(hSCManager, serviceName, SERVICE_STOP |
157 DELETE);
158 if(!hService) {
159     addLogMessage("Error: Can't remove service");
160     CloseServiceHandle(hSCManager);
161     return -1;

```



```
162     }  
163  
164     DeleteService(hService);  
165     CloseServiceHandle(hService);  
166     CloseServiceHandle(hSCManager);  
167     addLogMessage("Success remove service!");  
168     return 0;  
169 }
```

Додаток Б

Лістинг основних функцій програми клієнта

```
1      FileInfo f = new FileInfo("Client_Login.lg");
2      FileStream s = f.Open(FileMode.OpenOrCreate, FileAccess.Read);
3      string read = null;
4      string login = null;
5      string password = null;
6      bool z = false;
7      while ((read = s.ReadLine()) != null)
8      {
9          login = read;
10         password = s.ReadLine();
11         if ((textBox1.Text == login) & (textBox2.Text == password))
12         {
13             z = true;
14         }
15     }
16     f.Close();
17     if (z == false)
18     {
19         label3.Text = "Wrong login or password. Please check information.";
20         label3.Visible = true;
21     }
22     else
23     {
24         Form2 form2= new Form2();
```

```
25     form2.Show();
26     System.Diagnostics.Process Proc = new System.Diagnostics.Process();
27     Proc.StartInfo.FileName = "Client.exe";
28     Proc.Start();
29
30 }
31
32
33 int _tmain(int argc, _TCHAR* argv[])
34 {
35     char buff[BUFSIZ];
36     printf("TCP CLIENT\n");
37
38     if (WSAStartup(0x202, (WSADATA *)&buff[0]))
39     {
40         printf("WSAStart error %d\n", WSAGetLastError());
41         return -1;
42     }
43
44     SOCKET my_sock;
45     my_sock = socket(AF_INET, SOCK_STREAM, 0);
46     if (my_sock < 0)
47     {
48         printf("Socket() error %d\n", WSAGetLastError());
```

```
49         return -1;
50     }
51     sockaddr_in dest_addr;
52     dest_addr.sin_family = AF_INET;
53     dest_addr.sin_port = htons(PORT);
54     HOSTENT *hst;
55     if (inet_addr(SERVERADDR) != INADDR_NONE)
56         dest_addr.sin_addr.s_addr = inet_addr(SERVERADDR);
57     else
58     {
59         if (hst = gethostbyname(SERVERADDR))
60             ((unsigned long *)&dest_addr.sin_addr)[0] =
61             ((unsigned long **)hst->h_addr_list)[0][0];
62         else
63         {
64             printf("Invalid address %s\n", SERVERADDR);
65             closesocket(my_sock);
66             WSACleanup();
67             return -1;
68         }
69     }
70     if (connect(my_sock, (sockaddr *)&dest_addr, sizeof(dest_addr)))
71     {
```

```
72     printf("Connect error %d\n", WSAGetLastError());
73     ChangeWorkingType(WSAGetLastError, -1);
74     return -1;
75 }
76
77     printf("Soedinenie s %s uspeshno ustanovleno\n \
78         Type quit for quit\n\n", SERVERADDR);
79
80     int nsize;
81     while ((nsize = recv(my_sock, &buff[0], sizeof(buff) - 1, 0)) != SOCKET_ERROR)
82     {
83         buff[nsize] = 0;
84         printf("S=>C:%s", buff);
85         printf("S<=C:"); fgets(&buff[0], sizeof(buff) - 1, stdin);
86
87         if (!strcmp(&buff[0], "quit\n"))
88         {
89             printf("Exit...");
90             closesocket(my_sock);
91             WSACleanup();
92             return 0;
93         }
94         send(my_sock, &buff[0], strlen(&buff[0]), 0);
```

```
95     }
96     printf("Recv error %d\n", WSAGetLastError());
97     closesocket(my_sock);
98     WSACleanup();
99     return -1;
100
101     //return 0;
102
```

Додаток В

Комп'ютерна презентація

МАГІСТЕРСЬКА РОБОТА

НА ТЕМУ:

Дослідження та розробка програмних засобів
обміну інформацією в розподіленій системі

Виконав ст. гр. СП -16 Дм:
Керівник :

Крючков В.С.
к.т.н.доц.Щербакова М.Є.

м. Северодонецьк 2018 р.

Рисунок В.1 - Титульний лист

Актуальность

Данная тема была выбрана в связи с тем, что на ОП ИВЦ ОАО «Лисичанскуголь» возникла проблема с обменом информацией между отделами. Но уже существующие программные средства не удовлетворяют требованиям. А именно все существующие системы документооборота не очень надёжны. Это обусловлено тем, что чаще всего они являются зависимыми от сервера. И в случае его отключения или же выхода из строя система становится непригодной для эксплуатации. Есть так же и другой тип системы, а именно распределённые. В отличие от предыдущего варианта, они не нуждаются в серверной части. Но их основным минусом является то, что в случае отсутствия сегмента внутри сети, файлы которые на нём хранятся, становятся недоступными. Таким образом, каждая реализация имеет свои плюсы и минусы. И данная разработка учитывает минусы обоих вариантов, и устраняет их.

Рисунок В.2 - Актуальність

Постановка задачи

- исследовать предметную область;
- проанализировать аналогичные программные продукты;
- выбрать стратегию разработки;
- построить математическую модель;
- разработать программное средство на языках программирования C++ и C#.

Рисунок В.3 – Постановка задачі

Анализ программных средств

Проанализировав уже существующие программные продукты у были выявлены минус и составлен список целей которые необходимо выполнить для их устранения, а именно:

- Дополнительная защита документов при обмене
- Авторизация пользователей внутри сети
- Разработка удобного интерфейса для данной отрасли
- Работоспособность системы в случае частичного отказа

Рисунок В.4 – Анализ современных программных средств

График полезности системы

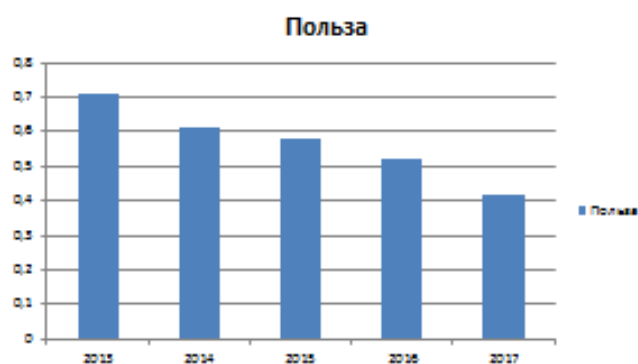


Рисунок В.5 – График полезности системы

Математическая модель

$$P = \left(\frac{T_{\text{общ}} * T_{\text{пр}}}{T_{\text{кор}}} + T_{\text{бпр}} \right) * (1 - (O_{\text{И}} + O_{\text{П}} + O_{\text{У}}))$$

$T_{\text{общ}}$ – Общее время работы программы.

$T_{\text{пр}}$ – Время работы сотрудника с программой.

$T_{\text{кор}}$ – Корректное время работы программы.

$T_{\text{бпр}}$ – Время работы сотрудника без программы.

$O_{\text{И}}$ – Процент измененной информации.

$O_{\text{П}}$ – Процент потерянной информации.

$O_{\text{У}}$ – Процент информации, которая попала в чужие руки.

Рисунок В.6 – Математична модель

Схема алгоритма работы клиента

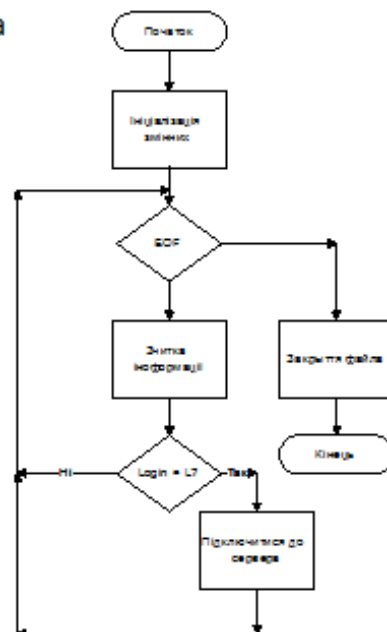


Рисунок В.7 – Схема алгоритму роботи клієнта

Выводы

- Проведено исследование в предметной области за последние 5 лет;
- Проведен анализ самых популярных программных продуктов, решающих данную проблему;
- Была выбрана стратегия разработки;
- Собрав информацию о предприятии и проблемную область, было проведено статистический анализ;
- На базе статистического анализа была построена математическая модель системы;
- Установлены основные направления разработки на базе анализа математической модели;
- Изучены основные подходы к реализации данной проблемы, так избран наиболее продуктивный;
- Создан графический интерфейс для серверной и клиентской части;
- Разработана практическая часть для реализации основных функций программы.

Рисунок В.8 – Висновки