

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

**Інститут (факультет) автоматизації та комп'ютерних технологій  
Кафедра автоматизації та комп'ютерних технологій систем  
управління**

<b>«До захисту в ЕК»</b>	<b>«До захисту допущено»</b>
Директор інституту(декан факультету)	Завідувач кафедри
_____	_____
(підпис)	(підпис)
<u>Форсюк А.В.</u>	<u>Чумаченко С.В.</u>
(прізвище та ініціали)	(прізвище та ініціали)
« ____ » _____ 20__ р.	« ____ » _____ 20__ р.

**КВАЛІФІКАЦІЙНА РОБОТА  
НА ЗДОБУТТЯ ОСВІТНЬОГО СТУПЕНЯ БАКАЛАВРА**

Зі спеціальності \_\_\_\_\_ 122 “Комп'ютерні науки”  
(код та назва спеціальності)  
освітньо-професійної програми \_\_\_\_\_ Комп'ютерні програми  
на тему: створення кросплатформного чат-сервісу для спілкування  
співробітників ПрАт “Оболонь”

Виконав: здобувач 4 курсу, групи КН-4-7ск

Сьомін Денис Андрійович  
\_\_\_\_\_  
(прізвище та ініціали)

Керівник Литвинов Валерій Андроникович  
(прізвище та ініціали)

\_\_\_\_\_  
(підпис)

Консультанти Литвинов В.А.  
(прізвище та ініціали)

\_\_\_\_\_  
(підпис)

Литвинов В.А.  
(прізвище та ініціали)

\_\_\_\_\_  
(підпис)

Литвинов В.А.  
(прізвище та ініціали)

\_\_\_\_\_  
(підпис)

Рецензент Власенко Лідія Олександрівна  
(прізвище та ініціали)

\_\_\_\_\_  
(підпис)

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

Здобувач \_\_\_\_\_  
(підпис)

Київ – 2020 р.

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ХАРЧОВИХ ТЕХНОЛОГІЙ

Інститут (факультет) Факультет автоматизації і комп'ютерних систем  
Кафедра автоматизації та комп'ютерних технологій систем управління  
Освітній ступінь бакалавр

Спеціальність 122 “Комп'ютерні науки”  
(код і назва)

Освітньо-професійна програма Комп'ютерні програми  
(назва)

**ЗАТВЕРДЖУЮ**

Завідувач кафедри Чумаченко С.М.

“ ” \_\_\_\_\_ 20\_\_ року

**З А В Д А Н Н Я**

**НА КВАЛІФІКАЦІЙНУ РОБОТУ ЗДОБУВАЧА**

Сьоміна Дениса Андрійовича

(прізвище, ім'я, по батькові)

1. Тема створення кросплатформного чат-сервісу для спілкування співробітників ПрАТ “Оболонь”

Керівник роботи Литвинов Валерій Андроникович,

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

затверджені наказом закладу вищої освіти від “27” квітня 2020 року № 269-кв

2. Строк подання здобувачем роботи 30 травня 2020р.

3. Вихідні дані до роботи:

загальна діяльність підприємства “ПРАТ Оболонь”, діяльність ІТ відділу, розроблена система автоматизації спілкування співробітників

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

1) Вступ 2) Аналіз підприємства 3) Аналіз існуючих систем 4) Проектування системи 5) Розробка системи автоматизації 6) Висновки

5. Перелік графічного матеріалу

1) Структурні схеми “19” 2) Графічні зображення «23

## 6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
	Литвинов В.А.	16.01.2020	20.01.2020
	Литвинов В.А.	16.01.2020	27.04.2020
	Литвинов В.А.	16.01.2020	25.05.2020

## 7. Дата видачі завдання 27 квітня 2020

### КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів виконання кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
	Дослідження діяльності роботи “ПРАТ Оболонь”	16.01.2020 - 25.01.2020	Виконано
	Аналіз методів пошуку і використання інформації	01.02.2020 - 19.02.2020	Виконано
	Аналіз існуючого ПЗ для виконання поставлених завдань	20.02.2020 - 01.03.2020	Виконано
	Розробка алгоритму створення системи для поліпшення роботи	09.03.2020 - 15.03.2020	Виконано
	Розробка програмного продукту	25.03.2020 - 15.05.2020	Виконано
	Оформлення роботи та розробка презентації	17.05.2020 - 25.05.2020	Виконано

**Здобувач**

\_\_\_\_\_ (підпис)

**Керівник роботи**

\_\_\_\_\_ (підпис)

**Сьомін Д.А.**  
(прізвище та ініціали)

**Литвинов В.А.**  
(прізвище та ініціали)

## АНОТАЦІЯ

Анотація до дипломного проекту: «створення кросплатформного чат-сервісу для спілкування співробітників ПрАт “Оболонь”».

Дипломний проект складається з 87 сторінок, 23 ілюстрацій (рисуноків), 3 додатків, 10 літературних джерел.

Об'єктом дослідження є аналіз підприємства ПрАт “Оболонь” та створення системи комунікації співробітників.

Метою створення системи є забезпечення ПрАт “Оболонь” системою для комунікації співробітників з простим, із зручним, функціональним інтерфейсом.

Для досягнення поставленої мети використовувалися:

- Арі для створення незалежних модулів.
- Ms SQL для збереження даних.

У дипломному проекті отримані наступні результати:

1. Описано діяльність ПрАт “Оболонь”.
2. Представлено порівняльний аналіз аналогічних програм, що існують на ринку.
3. Поставлено та спроектовано задачу.
4. Розроблено конкурентний додаток.

**Ключові слова:** Desktop, Арі, С#, .NET, .NET CORE, Visual Studio, MS SQL, Entity Framework, СКБД.

## ANNOTATION

Annotation to the diploma project: "creation of a cross-platform chat service for communication of Obolon employees".

The diploma project consists of 87 pages, 23 illustrations (drawings), 3 appendices, 10 literary sources.

The object of the study is the analysis of the company OJSC "Obolon" and the creation of a communication system for employees.

The purpose of the system is to provide Obolon PJSC with a system for communication of employees with a simple, user-friendly, functional interface.

To achieve this goal were used:

- Api to create independent modules.
- Ms SQL to save data.

The following results were obtained in the diploma project:

1. The activity of Obolon PJSC is described.
2. The comparative analysis of similar programs existing in the market is presented.
3. The task is set and designed.
4. Developed a competitive application.

**Keywords:** Desktop, Api, C #, .NET, .NET CORE, Visual Studio, MS SQL, Entity Framework, DBMS.

## Зміст

Вступ.....	8
Історія виникнення чату .....	9
1. СИСТЕМНИЙ АНАЛІЗ ТОВ «ПРАТ ОБОЛОНЬ» ТА ПОСТАНОВКА ЗАДАЧІ НА ПРОЕКТУВАННЯ .....	11
Історія ПрАТ «Оболонь» .....	11
1.1. Основний технологічний процес на ПрАТ «Оболонь» .....	13
1.2. Структура управління ПрАТ “Оболонь” .....	17
1.3. Організаційна структура ПрАТ «Оболонь»:.....	18
1.4. Наявні системи автоматизації .....	22
1.5. Опис предметної області .....	24
1.6. Моделювання предметної області.....	25
1.7. Постановка задачі на автоматизацію .....	33
Постановка завдання та розробка алгоритму роботи програми .....	34
1.8. Вимоги до обладнання .....	35
1.9. Функції системи .....	36
1.10. Вхідні та вихідні дані системи .....	36
2. ОПИС КОМПЛЕКСУ ЗАДАЧ АВТОМАТИЗАЦІЇ .....	37
2.1. Обґрунтування вибору засобів розробки системи .....	37
2.1.1. Модель AS-IS і TO-BE.....	37
2.1.2. Нотація IDEF0 .....	39
2.1.3. Нотація IDEF3 .....	40
ERwin Data Modeler .....	40
СУБД .....	41
Microsoft SQL Server .....	41
ORM.....	42
.Net .....	43
2.2. Проектування бази даних .....	46
2.3. Архітектура системи .....	49
2.3.1 Мікросервісна архітектура .....	49
2.3.2 Арі .....	52
2.3.3. Проєтування системи.....	54

2.4. Техніко-економічне обґрунтування доцільності розробки .....	57
Розрахунок витрат часу для стадії “Технічний проект” (Т3).....	60
Розрахунок витрат часу для стадії “Робочий проект” (Т4).....	61
3. ОХОРОНА ПРАЦІ.....	65
3.1. Охорона праці при розробці програмного забезпечення .....	65
3.1.1. Характеристика умов праці програміста .....	65
3.1.2. Режим праці .....	66
3.1.3. Ергономічні вимоги до робочого місця .....	69
3.2. Вимоги протипожежної безпеки.....	74
3.2.1. Пожежна безпека.....	74
3.2.2. Вимоги електробезпеки .....	79
3.3. Санітарно-гігієнічні вимоги .....	79
3.3.1. Забарвлення і коефіцієнти відображення .....	79
3.3.2. Освітлення.....	80
3.3.3. Параметри мікроклімату.....	81
3.3.4. Шум і вібрація .....	83
3.4. Аналіз ризиків .....	84
3.5. Пропозиції .....	85
Висновок. ....	87

## Вступ

Найчастіше щось роблячи по роботі або відпочиваючи вдома виникає проблема потреби думки з боку, ради або просто спілкування. У таких випадках зазвичай використовують засоби накопичення інформації або засоби телекомунікацій, але не завжди це буває доступно, тому було вирішено створити програмний продукт, який буде доступний кожному, і не буде вимагати від користувача максимум можливостей.

Питання спілкування цікавили людей завжди. Для того, щоб можна було обмінюватися інформацією не тільки при особистій зустрічі, а й на великих відстанях, люди винаходили все нові і нові технічні засоби, організовували поштові системи, простягали кабелі через континенти і океани, запускали супутники зв'язку. З розвитком інформаційних технологій стали можливим ще більш глобальні комунікації. Історичним «докомп'ютерним» попередником чатів, безсумнівно, був телефон. Ні пошта, ні телеграф не дозволяли спілкуватися в реальному часі і не були доступні в домашній обстановці. Винахід і поширення телефону по планеті викликало справжню революцію в засобах і способах спілкування. Можливість поговорити зі співрозмовником на іншій стороні Землі здавалася справжнім дивом.

Програми для обміну текстовими рядками, незважаючи на простоту самої ідеї, з'явилися не відразу. Базовим для них є єдиний протокол - TCP. TCP - один з основних протоколів передачі даних Інтернету, призначений для управління передачею даних в мережах і підмережах TCP / IP. Виконує функції протоколу транспортного рівня в стекупротоколів IP. Інтерфейсом прикладного програмування для мережевих додатків TCP / IP є сокети. Сокет - назва програмного інтерфейсу для забезпечення обміну даними між процесами. Процеси при такому обміні можуть виконуватися як на одній ЕОМ, так і на різних ЕОМ, пов'язаних між собою мережею. Сокет - абстрактний об'єкт, який представляє кінцеву точку з'єднання. Сокет - це комбінація IP-адреси і порту. Кожен процес може створити слухаючий сокет (серверний сокет) і прив'язати його до якого-небудь порту операційної

системи. Хто слухає процес зазвичай знаходиться в циклі очікування, тобто прокидається при появі нового з'єднання. Для нас це означає, що сервер буде слухати будь-якої порт і після підключення клієнта, будемо виробляти з ним якісь дії.

Чат - засіб обміну повідомленнями з комп'ютерної мережі в режимі реального часу, а також програмне забезпечення, що дозволяє організовувати таке спілкування. Характерною особливістю є комунікація саме в реальному часі або близька до цього, що відрізняє чат від форумів та інших «повільних» коштів. Тобто, якщо на форумі можна написати питання і чекати, поки хто-небудь вважатиме за потрібне на нього відповісти (в той же час, можна отримати і кілька відповідей відразу від різних користувачів), то в чаті спілкування відбувається тільки з тими, хто присутній в ньому зараз, а результати обміну повідомленнями можуть і не зберігатися.

Під словом чат зазвичай розуміється групове спілкування, хоча до них можна віднести і обмін текстом «один на один» за допомогою програм миттєвого обміну повідомленнями, наприклад, XMPP, ICQ або навіть SMS.

Основна мета - створення мережевого чату - миттєвий обмін повідомленнями. Використання даного програмного продукту допоможе скоротити час виконання будь-якої роботи або просто добре провести час, спілкуючись в колі друзів.

Мережевий чат може бути використаний як для малого кола людей, тобто в домашніх умовах, так і на виробництві невеликого підприємства.

### **Історія виникнення чату**

У другій половині XX століття почали бурхливо розвиватися комп'ютери. Однак довгий час вони були великими і дуже дорогими, що перешкоджало тому, щоб витратити дорогоцінний машинний час на забави з обміном повідомленнями замість розрахунків. До того ж, до кінця 60-х років вони не були пов'язані один з одним. Предок Інтернету, мережа ARPANET, в 1969 році налічувала лише чотири пов'язаних один з одним наукових комп'ютера. Трохи пізніше, в 1971 році, була придумана електронна пошта,

яка стала надзвичайно популярна через свою зручність. Поступово з'явилися нові служби повідомлень, такі, як списки поштової розсилки, групи новин та дошки оголошень. Однак у той час мережа ARPANET ще не могла легко взаємодіяти з іншими мережами, побудованими на інших технічних стандартах, що ускладнювало її поширення. Але тим не менш, ця проблема незабаром була вирішена після переходу мереж на протокол обміну даними TCP / IP, який успішно застосовується до цих пір. Саме в 1983 році термін «Інтернет» закріпився за мережею ARPANET.

Програми для обміну текстовими рядками, незважаючи на простоту самої ідеї, з'явилися не відразу. Приблизно в 1974 році для мейнфрейма PLATO був розроблена програма Talkomatic, потенційно дозволяла спілкуватися між тисячею терміналів системи. У 1980-х з'явилася система Freelancing 'Round table. Однак по-справжньому популярним став розроблений в 1988 році протокол, названий Internet Relay Chat (IRC), що приблизно можна перекласти як ретранслюється інтернет-розмова. Десь в цей же час з'явилося і поширилося саме поняття «чат». Спілкування в IRC швидко стало популярним через простоту процесу і дружності середовища. У 1991 році під час операції «Буря в пустелі» була організована IRC-трансляція новин - повідомлення з усього світу збиралися в одному місці і в режимі реального часу передавалися в IRC. Для клієнтів IRC, написано безліч ботів, наприклад, Eggdrop, що автоматизують багато рутинних операцій. Найвідомішим з клієнтів IRC став mIRC; завдяки простій і ефективній системі команд для нього було написано безліч скриптів, які також дозволяють виконувати широкий спектр дій. Боти і mIRC-боти використовуються для різних ігор в каналах - «Мафія», «Вікторина», та інших. Розробники IRC настільки добре продумали його архітектуру, що її з тих пір практично не було потрібно змінювати. Звичайно, у нього є недоліки: короткі повідомлення, проблема з кодуваннями, неможливість подивитися історію повідомлень при підключенні. Однак він був і залишається популярним засобом для чату, хоча і в значній мірі потіснений зі своїх позицій. Зокрема, в 1998 році був придуманий схожого

призначення протокол Jabber - навіть його назву (англ. Jabber балаканина, балаканина; тарабарщина) надсилало до слова chat. Jabber містив в собі багато технічні нововведення і поступово набув широкого поширення, а також став основою багатьох сервісів. Були й інші протоколи, менш відомі, наприклад, SIP.

## **1. СИСТЕМНИЙ АНАЛІЗ ТОВ «ПРАТ ОБОЛОНЬ» ТА ПОСТАНОВКА ЗАДАЧІ НА ПРОЕКТУВАННЯ**

Компанія - незалежна торгівельна організація, яка зареєстрована компетентним органом місцевого самоврядування для задоволення соціальних або особистих потреб шляхом систематичного ведення виробництва, наукових досліджень, торгівлі та господарської діяльності в кодексі Торгівлі України або інших законах. Основне завдання компанії - задоволення потреб ринку у своїй продукції та послугах з ціллю отримання прибутку.

### **Історія ПрАТ «Оболонь»**

1974 р. В новому районі Києва Оболонь на березі річки Почайна, яка є притокою Дніпра, є місцем будівництва майбутнього пивоварного заводу Києва намиз Алювій. Місцезнаходження компанії було обрано не випадково. Систему розробили фахівці з Чехії. Вони також встановили, що прісні та кришталево чисті води горизонту Юри, з яких у цій місцевості знайдено величезні запаси, чудово відповідали всім вимогливим критеріям пивоваріння. Цей фактор набув вирішального значення, і з того часу головна українська харчова компанія використовувала воду з власних артезіанських свердловин у приміщеннях заводу для виготовлення своєї продукції.

1980 р. Офіційне відкриття київського пивоварного заводу №3, встановлене для проведення 22-х Олімпійських ігор, та видання першої партії пива. Введена в експлуатацію безалкогольна фабрика. 1982 р. Новий завод по розливу мінеральної води виробляє свою першу продукцію на київському пивоварному заводі №3.

У 1986 році на базі пивоварні № 3 у Києві було засновано Асоціацію пива та безалкогольного списку "Оболонь", до якої входили пивоварня № 1 у Києві, пивоварня № 2 у Києві (сьогодні пивоварня "Поділ") та фастів пивоварня.

1992 рік "Оболонь" отримує сертифікат про приватизацію від Державного фонду №1 нерухомості №1 в Україні, що робить його першим незалежним приватним підприємством в Україні, яке ввійшло в історію. Відтепер вся продукція цього виробника випускатиметься під брендом Obolon. Триває виробництво нового пива "Оболонь Темний".

У 1993 році приватизована компанія отримала юридичний статус як акціонерне товариство. Лінія розливу пива ємністю 0,33 літра введена в експлуатацію.

У 1994 році акціонерне товариство вперше почало випускати слабоалкогольні напої "Джин Тонік". Це відкрило ринок слабоалкогольних напоїв не лише в Україні, а й у Східній Європі. Оболонь стає власником недобудованої будівлі, що зберігається у Красилів на Хмельниччині. Сьогодні це дочірнє підприємство акціонерного товариства, яке виробляє мінеральну воду, безалкогольні напої та низький вміст алкоголю під маркою Оболонь. Перша оптово-роздрібна база "Оболонь" створена в м. Красилів. Сьогодні компанія має дев'ятнадцять баз у різних регіонах країни та реалізує бізнес-продукцію.

1997. "Оболонь" стає співвласником ВАТ "Охтирський пивоварний завод". У продажу є новий безалкогольний напій "Спорт". Вперше в Україні Оболонь отримав кредит від ЄБРР на суму 40 мільйонів доларів без гарантій уряду. Кошти були використані для фінансування стратегічних програм.

У 1998 році вона вперше отримала міжнародний сертифікат на систему управління якістю ISO-9001 версії 1994 року в галузі Оболонь, що засвідчує високий рівень усіх процесів та потужність української компанії. Випускати продукцію світового класу. В Оболоні вводяться в дію нові сучасні ферментаційні установки - циліндрично-конічні цистерни (ЦКТ).

1999 Завод "Оболонь" нагрівається сучасним паровим котлом CLAYTON з Бельгії, який був встановлений і введений в експлуатацію цього року. На головному заводі компанії були введені в дію нові заводи: завод з видалення вуглекислого газу, циліндричні та конічні цистерни для бродіння пива, пивоварня третього класу та новий завод по розливу пива. "Оболонь" стає офіційним спонсором футбольного клубу "Оболонь", який на той час грав у вищому дивізіоні професійної футбольної ліги України.

2001 рік «Оболонь» підтверджує високу організацію системи управління якістю та вперше отримує в Україні сертифікат відповідно до стандартної версії ISO 9001 2000. На заводі «Оболонь» на заводі вводяться в дію три сучасних запірних цистерни: виробництво солоду та нове німецьке виробництво Запроваджені ССТ.

У 2004 році була запущена нова лінія розливу пива потужністю 110 000 пляшок на годину. З впровадженням нової лінії розливу в завод "Оболонь" перетворився на одну з трьох найпотужніших пивоварних заводів у Європі. Був введений в дію новий кулінарний контроль та встановлені нові циліндрично-конічні цистерни.

У 2007 році «Оболонь» була визнана одним з найкращих платників податків у Києві. Розпочинається грандіозний проект ТМ «Оболонь» - Всеукраїнський чемпіонат з пивного спорту.

У 2008 році було завершено будівництво власного солодового будинку в місті Чемерівка на Хмельниччині та в проект вкладено 100 мільйонів доларів. Компанія також продовжувала модернізувати пивоварний завод «Zibert» у Фастові. Також запускається оновлена українська марка пива Magnat.

### **1.1. Основний технологічний процес на ПрАТ «Оболонь»**

Не слід забувати, що ароматне і приємне на смак пиво вариться з високоякісного солоду. Більшість українських пивоварних підприємств купують цю сировину за кордоном або у місцевих виробників. Оболонь пішла іншим шляхом і виробила власну сировину. Перший солод виробляв у 1981 році. З того часу солодовий будинок кілька разів перебудовувався. З 1985 по

1997 рр. Продуктивність виробництва солоду значно зросла встановленням трьох нових сушарок. Також був відреставрований солодовий відділ та замінено старі агітатори. Замість громіздких старих вентиляторів встановлено сучасні бельгійські пристрої. У 2001 р. Згідно з програмою збільшення потужностей заводу з виробництва солоду було зміцнено сита відділу солоду, що призвело до завантаження ящиків культури до 80 тонн кожен. Закуплено також важкі контейнери на замок. Усі ці перетворення забезпечили високу виробництво солоду та чудову якість.

Для постійного постачання ячмінного солоду та зберігання готового солоду "Оболонь" має елеватор, який піднімається по всій поверхні заводу. Ліфт можна описати як механізований склад зерна для зерна. Він складається з трьох частин - солоду, ячменю та робочих веж і є одним із великих зернових магазинів. Оболонь збільшує кількісні показники і ніколи не забуває про якість. Завдяки сумлінній роботі солодівців це виходить відмінно. Все пиво Оболонь виготовляється тільки з солоду високої якості, який пропонує приємний смак, відмінний колір, стійку і м'яку піну. Основна технологічна мета "Оболоні" - постійне поліпшення якості продукції та всіх виробничих процесів. З цієї причини компанія встановлює вдосконалені пристрої, щоб вона могла бути першим технічним пристроєм не лише в Україні, а й у Європі. До складу компанії входять дві найсучасніші пивоварні України. Це головний завод "Оболонь" у Києві та пивоварний завод "Зіберта" у Фастові на Київщині.

Київський завод має номінальну потужність 11 мільйонів штук. Пиво на рік. Це робить компанію найбільшою пивоварнею в Європі. За останні два роки система встановила унікальну систему приготування їжі, яка може варити 12 разів на день 750 гектолітрами холодного сусла. Особливістю цієї нової послідовності приготування від німецької компанії Ziemann є те, що вона економить вдвічі більше витрат на теплову енергію під час варіння. На сьогодні споживання енергії - одне з головних завдань інженерів компанії. Одним із об'єктів гордості є унікальне обладнання для виробництва пива зі старих дріжджів, яке виробляє від 500 до 600 000 Дал пива на рік. Гарантію

високої якості продукції забезпечують також чотири високопродуктивні фільтраційні установки, виготовлені на замовлення німецької компанії KHS на київському заводі. Найбільший фільтр має потужність 950 гектолітрів на годину. Система фільтру «Оболонь» не має аналогів не тільки в Україні, але і в Європі.

"Пивоварня Зібберта" у Фастові, що на Київщині, після реконструкції останніх двох років стала сучасним високотехнологічним підприємством. Сучасні потужності з виробництва пива в Німеччині встановлені в рамках великого інвестиційного проекту на суму понад 40 мільйонів євро. В результаті виробничі потужності компанії зросли в 15 разів і становлять 1,5 мільйона гектолітрів пива на рік. В результаті розширено обладнання для бродіння пива та розливні лінії в різних типах ємностей.

Сировина важлива для виробництва якісного продукту, не менше ніж обладнання. Солод - один з основних компонентів пивоваріння. Щоб отримати вибір солоду, корпорація «Оболонь» побудувала в селі солодовий будинок. Чемерівська область у Хмельницькому - один із найсучасніших підприємств на континенті. Потужність компанії становить понад 120 000 тонн солоду на рік. З цією метою завод переробляє 180 000 тонн ячменю (450 тонн на день). На заводі для солоду було впроваджено багато інноваційних рішень, що дозволяє значно покращити технологічні процеси та зменшити витрати ресурсів. Сушильний відділ відновлює гаряче повітря, мінімізує використання холоду, а мембранна система очищення стічних вод є унікальною в Європі.

Автоматизація відіграє важливу роль на всіх заводах Оболонь. Основні виробничі функції виконують машини, і лише люди контролюють процес. Це демонструє не тільки технологічну досконалість виробництва, але й відповідність найсуворішим стандартам якості та високий рівень організації всіх процесів. І завдяки найдосвідченішим технікам, багато з яких пропрацювали в цій галузі вже більше чверті століття, Оболонь має можливість впроваджувати кращі практики, створювати нові смакові якості та постійно вдосконалювати рецепти напоїв. Для цього використовуються

найсучасніші експериментальні, виробничі, радіологічні та технічні контрольні лабораторії компанії. Вони пропонують постійний аналіз якості продукції на всіх етапах виробництва та розвиток інноваційної продукції.

ПАТ "Оболонь" використовує місцевий солод для виготовлення пива. Так, ячмінь очищають, сортують і пророщують. Процес проростання відбувається в оптимальних умовах при постійному контролі технологічних параметрів та біохімічних процесів у пшениці. Ячмінь проростає до максимальної концентрації активних ферментів.

Сьогодні ПАТ "Оболонь" пропонує своїм клієнтам розливне пиво (Euro-Probe та NRW 0,5 л), ПЕТ-пляшку (1 та 2 л), нижню пляшку (0,33 л та 0,5 л) та бочка з 1) бочки з (50 і 30 літрів). Наповнення відбувається на ефективних лініях, одна з яких може заповнити до 60 000 пляшок на годину.

Усі ці складні процеси бродіння виконуються за допомогою сучасних та ефективних пристроїв, які гарантують високу якість на кожному етапі процесу бродіння. Завдяки ІТ-контролю високопрофесійні фахівці ПАТ «Оболонь» перевіряють відповідність усім вимогам та технологічним стандартам.

Компанія модернізує своє обладнання для поліпшення якості продукції та всіх виробничих процесів. Це дозволяє нам бути лідером у технічному обладнанні не лише в Україні, а й у Європі.

Київський конструкторський завод має потужність 11 мільйонів гектолітрів пива на рік, що робить його найбільшою пивоварнею в Європі. У системі є система приготування їжі від німецької компанії Ziemann, яка виробляє до 12 напоїв на день з 750 гектолітрами холодної сусла. Для забезпечення якості продукції використовуються чотири високоефективні фільтрувальні установки німецької компанії KHS. Потужність найбільшого фільтра - 950 гектолітрів на годину. Фабрика має одну з найбільших ліній для скляного наповнення в Європі потужністю 110 000 пляшок на годину, яку виробляє німецька компанія KHS. У системі також є високопродуктивні лінії для наповнення банок, ПЕТ-пляшок і барабанів.

Продовжується модернізація пивоварного заводу "Зіберт". Понад 40 мільйонів євро було вкладено у встановлення нових кухонних систем управління, бродильних баків та магістралей.

Щоб захистити навколишнє середовище, «Оболонь» інвестувала в унікальну систему сушіння пивних зерен, вироблених з пивної трави у великих кількостях (до 700 тонн на день). Після сушіння та грануляції пивні гранули є дуже корисною їжею для сільськогосподарських тварин, що може вплинути лише на великі господарства.

## **1.2. Структура управління ПрАТ «Оболонь»**

Структура Оболоні розвивалася протягом багатьох років під впливом стратегії розвитку, спрямованої на диверсифікацію виробництва, перехід на сировину з власного виробництва, інноваційний підхід, абсолютну екологічну безпеку та повну корпоративну соціальну відповідальність. Корпорація "Оболонь" об'єднує атомну електростанцію, два окремих підприємства, два філії та чотири компанії з прав людини. У компанії працює приблизно 7500 співробітників. Основні структурні підрозділи Оболоні:

Основна фабрика в Києві - ПАТ "Оболонь" (виробництво пива, безалкогольних напоїв, мінеральної води, вареників тощо).

Місцезнаходження: Оболонь ПАТ "Браур Зіберта", м. Фастів, Київська область. (Виробництво пива), Оболонь ПАТ "Красилівське", м. Красилів, Хмельницька область. (Виробництво мінеральної води, газованих та безалкогольних напоїв).

Соціальні права: ВАТ "Охтирський пивоварний завод", Охтирка, Сумська область (пиво, безалкогольні напої, виробництво солоду), ПрАТ "Бершадський пивоварний завод", Бершад, Вінницька область (безалкогольні напої, виробництво солоду), ВАТ "Дятківці", М Коломия, Івано-Франківськ (виробництво солоду), ТОВ "Оболонь Агро", м. Чемерівці, Хмельницька область (сільськогосподарське виробництво, будівництво, скотарство та свинарство), ВАТ "Рокитне склозавод", село Рокитне, Рівненська область (виробництво скляної тари).

Окремі магазини: фабрика в Олександрії, Кіровоградська область. (Низьке виробництво спирту соди до соди, перетворення ПЕТ-пляшок, виробництво бандажів), розписане в селі Чемерівці Хмельницької області (виробництво солоду).

### **1.3. Організаційна структура ПрАТ «Оболонь»:**

Організаційна структура підприємства являє собою конкретну структурну модель, яка виступає базою розподілу обов'язків працівників даної організації. Тобто, це керівництво, яке пояснює особливості та нюанси функціонування компанії, а також те, як будується весь процес роботи в ній.

Під організаційною структурою управління розуміється склад, взаємодію, підпорядкованість, а так же розподіл роботи по підрозділах і управлінським органам, між якими формуються певні відносини, пов'язані з реалізацією владних повноважень, потоків розпоряджень та інформації.

Основою для появи і функціонування того чи іншого типу організаційної структури управління на підприємстві, а також запорукою збільшення продуктивності є горизонтальний поділ праці, при якому весь обсяг роботи розбивається на компоненти.

Використання в практиці управління схеми організаційної структури дозволяє зрозуміти формальну ієрархію організації, у тому числі, структуру процесів, відповідальності та звітності.

Ці схеми застосовуються для вирішення низки завдань: визначення ролей і обов'язків всього персоналу в межах організації; встановлення ієрархічної структури повноважень, влади і процесу прийняття рішень; формування комунікаційних каналів і потоків інформації, включаючи правила підпорядкування і методи регулювання, що відносяться до методик звітності; формування механізму контролю з визначенням ступеню централізації та масштабу контролю; визначення функцій і управлінських завдань.

На рис. 1.1. приведено приклад верхнього рівня організаційної структури відділу розробки та інтеграції “ПРАТ ОБОЛОНЬ” та інших відділів з якими він найчастіше взаємодіє.

Для деталізації значення кожного працівника відділу, необхідно визначити ролі, які виокремлені конкретними посадовими обов’язками. Всі обов’язки прописуються у посадових інструкціях.

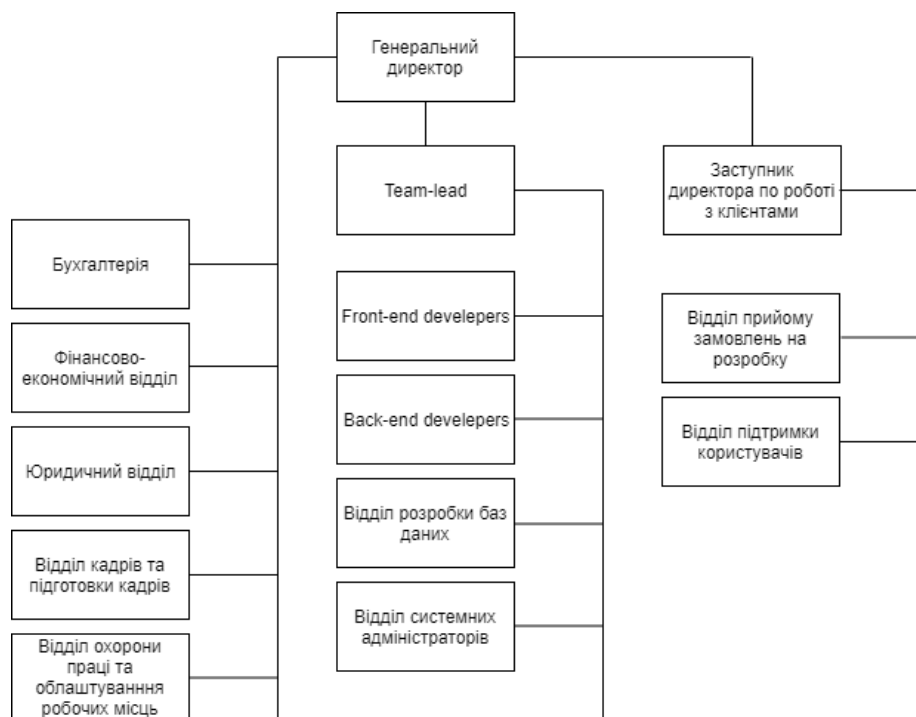


Рисунок 1.1 – Верхній рівень організаційної структури відділу розробки та інтеграції “ПРАТ ОБОЛОНЬ”

Для опису роботи відділу необхідно проаналізувати його діяльність, визначити роботи, які виконуються у відділі та створити текстовий алгоритм послідовності виконання робіт усім відділом вцілому. В описі потрібно зазначити хід виконання усіх робіт, які являються цільовими для даного відділу та зумовлюють досягнення результату діяльності усього підприємства.

Таблиця 1.1 – Завдання і функції відділу розробки ІТ-проектів

№	Функції	Задачі
1	Розробка нових проектів	<ul style="list-style-type: none"> <li>- Створення прототипів;</li> <li>- Розробка дизайну проектів;</li> <li>- Створення Back-end;</li> <li>- Розробка технічної документації до проекту;</li> <li>- Створення архітектури БД;</li> </ul>
2	Підтримка діючих проектів	<ul style="list-style-type: none"> <li>- виправлення помилок;</li> <li>- Створення нових функцій;</li> <li>- Оновлення технічної документації та своєчасне її постачання відділу підтримки користувачів;</li> </ul>
3	Тестування додатків	<ul style="list-style-type: none"> <li>- Створення unit-тестів;</li> <li>- Створення test-case;</li> <li>- Автоматизація тестування;</li> <li>- Знаходження багів;</li> <li>- Надання помилок розробникам;</li> </ul>
4	Налаштування систем компанії	<ul style="list-style-type: none"> <li>- Надання різних рівнів доступу;</li> <li>- Налаштування комп'ютерних мереж;</li> <li>- Налаштування пристроїв компанії;</li> <li>- Підключення робочого місця;</li> <li>- Підтримка серверів компанії;</li> </ul>

Кожен підрозділ організації має функціональні взаємозв'язки з іншими підрозділами – точки дотику, перетину, що представляють певні потоки (інформаційні, документальні, матеріальні, тощо). Для опису взаємозв'язків необхідно виконати опис у табличному представленні. Опис взаємозв'язків відділу розробки з іншими підрозділами та відділами аутсорс ІТ компанії приведено у таблиці 1.2.

Таблиця 1.2 – Завдання і функції відділу розробки ІТ-проектів

<b>№</b>	<b>Відділ</b>	<b>Одержання</b>	<b>Надання</b>
1	Бухгалтерія	- Відомості про опрацьовані зміни; - Інформація про діяльність компанії;	- Заробітної платні; - Звітів з бухгалтерського обліку;
2	Фінансово-економічний відділ	- Інформація щодо стану проектів; - Кількість клієнтів; - Скарг клієнтів щодо якості продукту;	- Списку замовників; - Сертифікатів на замовлений продукт; - Прогнозу на наступну розробку;
3	Юридичний відділ	- Скарг клієнтів щодо якості продукту; - Списку клієнтів; - Технічного завдання та документації проектів;	- Надання юридичного захисту продуктам та діяльності компанії;
4	Відділ кадрів та підготовки кадрів	- Список співробітників; - Список скарг і пропозицій;	- Нових кадрів; - Консультації співробітникам; - Списку кандидатів;
5	Відділ охорони праці та облаштування робочих місць	- Список скарг і пропозицій; - Стану робочих місць; - Стану апаратури;	- Надання консультацій; - Правильного розташування та облаштування робочих місць; - Лекцій по охороні праці;

Автоматизація - один із напрямків науково-технічного розвитку, який спрямований на використання технічних засобів, економічних та математичних методів та систем управління, пов'язаних із саморегулюванням, які заважають людям отримувати, перетворювати, передавати та використовувати енергію, матеріали чи інформацію. це участь або складність вжитих дій. Ось автоматизовані системи, що використовуються на заводі.

#### **1.4. Наявні системи автоматизації**

##### *Автоматизація бухгалтерського обліку*

Повна автоматизація компанії (юридичної особи) не може обійтися без створення та налаштування системи бухгалтерського обліку. Мета програми бухгалтерського обліку - збір даних бухгалтерського обліку про діяльність компанії та підготовка їх до регулюючих органів на основі зібраних звітних даних. Перевагою використання бухгалтерської програми може бути можливість надсилання звітів електронною поштою та управління документами підрядникам в електронному вигляді. У деяких випадках, наприклад, наприклад, для невеликого обсягу ділових транзакцій програма може слугувати надійним постачальником інформаційної інформації про діяльність бізнесу. Використовуючи програму обліку, ви можете отримати адміністративні дані за допомогою:

- залишки та грошові потоки
- Наявність та обіг товарів, матеріалів на складах
- борги та розрахунки з покупцями, постачальниками
- Звіт, амортизація, основні засоби (шляхом автоматизації основних засобів амортизація може бути розрахована за кілька хвилин і врахована в обліку).

##### *Автоматизація заробітної плати*

При великій кількості працівників компанії, нарахування заробітної плати вимагає багато часу і сил. Не рідкість, що через велику кількість даних, яку необхідно обчислити за короткий час, робляться помилки, які призводять

до перерахунку та штрафу за неправильно сформовані звіти. Програма обчислення заробітної плати дозволяє:

- Провести нарахування та утримання співробітників
- Розрахувати відпускні, лікарняні, відрядження
- Провести виплату зарплати співробітникам
- Підготувати об'ємні звіти

#### *Автоматизація управління людськими ресурсами*

Автоматизація управління персоналом спрямована з одного боку на збір особистої інформації про прийняття на роботу, звільнення, переведення, просування по службі та найму працівників, з іншого боку інформацію про освіту, навички, додаткові курси та навчання для виявлення найталановитіших працівників та просування ключових позицій в організації.

#### *Автоматизація виробництва*

Автоматизація виробництва пропонує компаніям безліч варіантів обліку, адміністрування та планування виробничих процесів і робочих процесів.

Збалансування виробництва за допомогою автоматизованих систем розраховує попередні виробничі витрати в режимі реального часу та фактичні виробничі витрати на кінець місяця. Розрахунок виробничих витрат дає можливість точніше визначити ціну та обсяг переговорів, для яких менеджер з продажу не повинен підтримувати прийнятну прибутковість. Враховуючи поточні витрати на виробництво, ви також можете пристосувати заробітну плату виробничих працівників до заробітної плати галузі та запобігти виходу з бізнесу кваліфікованих працівників.

Автоматизація виробництва повинна відповісти на одне з найважливіших питань: скільки і яке обладнання і скільки і які фахівці потрібні для виготовлення планової продукції. Правильне планування навантаження на ресурси підприємства є важливим моментом у підвищенні ефективності виробничої діяльності.

### *Автоматизація бюджетування*

У компанії часто є лише кілька працівників на початку бізнесу, а керівник особисто керує та контролює всі доходи та витрати компанії. Він знає, скільки може заробити його бізнес і де він витратить гроші, наприклад: зарплату, проїзд, рекламу. Однак у міру зростання бізнесу та залучення все більшої кількості людей відділи діляться на сфери діяльності. У такій ситуації керівник більше не може стежити за усім, і, звичайно, керівники відділів можуть більше знати про їх спеціалізацію та ситуацію на ринку, ніж директор. Автоматизація бюджету призначена для того, щоб підрозділи могли управляти власними коштами та вирішувати, куди вкласти гроші, щоб отримати найкращі результати від підрозділу, забезпечуючи при цьому прозору систему витрачання грошей. У межах бюджетного процесу можуть бути сформовані: плани, доходи та бюджети витрат, також розраховується необхідна кількість грошових ресурсів, яка може забезпечити фінансову стабільність, платоспроможність та ліквідність балансу. Делегуючи повноваження щодо управління коштами та відповідальність за досягнення результатів діловим підрозділам, керівництво може зосередити увагу на найважливіших стратегічних цілях свого бізнесу.

### **1.5. Опис предметної області**

Основним напрямком ІТ відділу буде якісне і своєчасне виконання завдань будь-якої складності і терміновості, допомагати компанії підвищити продуктивність бізнес-процесів і оптимізувати витрати на їх виконання і зміст власного персоналу.

Основними видами діяльності в ІТ відділу компанії буде прийняття завдань на розробку, проектування, тестування та розробка різного роду застосунків.

Основні елементи предметної області:

- керівник проекту;
- персонал;
- директор;

- план проекту;
- інформація про доступні кошти.

### **1.6. Моделювання предметної області**

Модель бізнес-процесів з управління діяльністю ІТ відділу (AS-IS).

Розроблена функціональна модель описує процес виконання проекту, який виконує менеджер з постачання товару, в тому вигляді, як вони здійснюються зараз (AS-IS). На схемі (див. рис. Б.1 у додатку б) «Організація формування завдання» зображено два рівні декомпозиції.

Функціональна модель зображує діяльність роботи керівника проекту (див. рис. Б2 у додатку Б). В функціональній моделі батьківською діаграмою є: «Управління проектами для ІТ відділу».

Вхідними стрілками до діаграми є:

- Інформація о необхідності в проекті – визначення доцільності розробки
- Інформація про терміни – строки виконання проекту

Стрілками контролю є:

- Звід правил – правила щодо виконання проектів
- План проекту - документ, який встановлює послідовність дій які потрібно виконати, щоб досягти мети проекту

Стрілками механізмів є:

- Керівник проекту

Вихідними стрілками є:

- Наказ про початок проекту
- Список проектної команди

Заключний звіт - даний звіт складається керівником або менеджером проекту до закінчення проекту.

Перший рівень декомпозиції включає в себе чотири процеса (див. рис. А.2 у додатку А). В кожному з процесів в ролі стрілок «Механізмів» виступає:

«Керівник проекту». А також керуючими факторами для ініціалізації проекту з процесів є «Звід правил», для контролю виконання проекту - «План проекту».

Ініціалізація проекту – визначається інформація про проект, який ґрунтується на оплаті та виду проекту. Вхідною стрілкою є «Інформація о необхідності в проекті», а вихідною інформацією є «Підготовка до виконання».

Організація виконання проекту – даний процес формує потреби та намагається передбачити майбутні проблеми на етапі розробки. Вхідною стрілкою є «Інформація про терміни», а результатом є «Перевірка результатів».

Контроль виконання проекту – процес контролює виконання проекту, а в якості джерела потреб використовується вхідна інформація: «Підготовка до перевірки результатів». Результатом даного процесу є «Підготовка до завершення».

Завершити проект – при завершенні проекту закриваються всі проектні завдання та виправляються всі знайдені помилки. Вихідною інформацією є «Заключний звіт».

Декомпозиція організації виконання проекту декомпозиції включає в себе чотири процеса. В кожному з процесів в ролі стрілок «Механізмів» виступає: «Керівник проекту».

Закупка необхідного обладнання – є наслідком підготовки до виконання. Контролює виконання «Керівник проекту». Вихідною інформацією є «Звіт з закупленого обладнання».

Підготовка місць для виконання – даний процес відбувається після закупки необхідного обладнання. Контролює виконання «Керівник проекту». Вихідною інформацією є «Особисті справи співробітників».

Створення списку команди - даний процес формує команду для подальшої розробки проекту. Контролює виконання «Керівник проекту». Вихідною інформацією є «Інформація про виконавців».

Розподіл завдань між співробітниками – данний процес необхідний для ефективного виконання проекту. Контролює виконання «Керівник проекту». Вихідною інформацією є «Перевірка результатів».

Проаналізувавши роботу управління проектами для ІТ відділу було виявлено ряд проблем. Основні проблеми:

- Немає інформації о витратах;
- Календарного плану;
- Указів керівництва;
- Інформація о доступних коштах;
- Звіту про строки;
- Звіту про витрати;
- Директор не приймає участі в управлінні;
- Недостатній рівень деталізації;

Побудована оптимізована модель процесів діяльності установи з проведення навчальних курсів (ТО-ВЕ) (Додаток А. рис.А.1.), яка враховує пропозиції щодо покращення результатів діяльності.

На верхньому рівні до входів було додано «Інформацію о витратах» до механізмів додано директора, що буде обирати керівника проекту. Були додані до стрілок контролю «Указ керівництва», «Календарний план», «Інформація о доступних коштах». Також до виходів було додано до виходів «Звіт про строки», «Звіт о затратах».

У першому рівні декомпозиції відбулися такі зміни. До першого блоку який має назву ініціалізація проекту було додано стрілку контролю «Указ керівництва».

Було додано блок планування проекту. До стрілок контролю належить «Указ керівництва» та «Календарний план». До механізмів належить «Керівник проекту». До входу «Підготовка до планування», до виходів «Підготовка до організації».

До організації виконання проекту було додано вхід «Інформація о витратах», до механізмів контролю «Інформація о доступних коштах». До виходів «Звіт про строки», «Звіт о витратах».

До контролю виконання проекту було додано механізм контролю «Укази керівництва».

До завершення проекту було додано механізм контролю «Укази керівництва».

Була додана декомпозиція ініціалізації проекту, що включає в себе три процеси.

Призначення керівника – вхідною інформацією є «Інформація о необхідності в проекті», до механізмів належить «Директор», вихідною інформацією є «Основні завдання».

Створення ТЗ – вхідною інформацією є «Основні завдання», до механізмів належить «Директор» та «Керівник проекту». До стрілок контролю відноситься «Звід правил», вихідною інформацією є «Документообіг».

Створення супровідної документації – вхідною інформацією є «Документообіг», до контролю належить «Керівник проекту», до стрілок контролю «Укази керівництва», до виходів «Наказ про початок проекту», «Підготовка для планування».

Планування проекту зображено в нотації IDEF3.

Організація виконання проекту складається з чотирьох блоків.

Закупка необхідного обладнання – було додано стрілку контролю «Інформація о доступних коштах».

Підготовка місць для виконання – було додано вхідну стрілку «Інформація о витратах», додано стрілки контролю «Підготовка до організації», «Інформація о доступних коштах».

Створення списку команди – додано стрілку контролю «Інформація о доступних коштах».

Розподіл завдань між співробітниками - додано стрілку контролю «Інформація о доступних коштах».

## 1.6.1 Виявлені в результаті функціонального моделювання проблеми та задачі автоматизації

Проаналізувавши функціональну модель ІТ відділу можна виявити наступні проблеми:

- при пошуку помилок у функціонуванні відділів було виявлено проблему в комунікації між відділами, навіть невелике питання займає багато часу;
- відсутність можливості вирішити питання дистанційно.

Рішенням цих проблем може стати автоматизація процесів процесів комунікації між відділами.

Основні задачі для подальшої автоматизації:

- надання співробітникам системи комунікації.
- Мережевий чат повинен забезпечувати миттєвий обмін повідомленнями, бути корисним у використанні і споживати найменшу кількість ресурсів персонального комп'ютера;
- Програмний продукт повинен мати дружній інтерфейс, що не вимагає спеціальної підготовки користувачів, повинен сприяти більш ефективному розподілу і економії часу;
- створення закритої системи для обміну повідомленнями.

## 1.5.2 Аналіз аналогічних систем

*Slack*

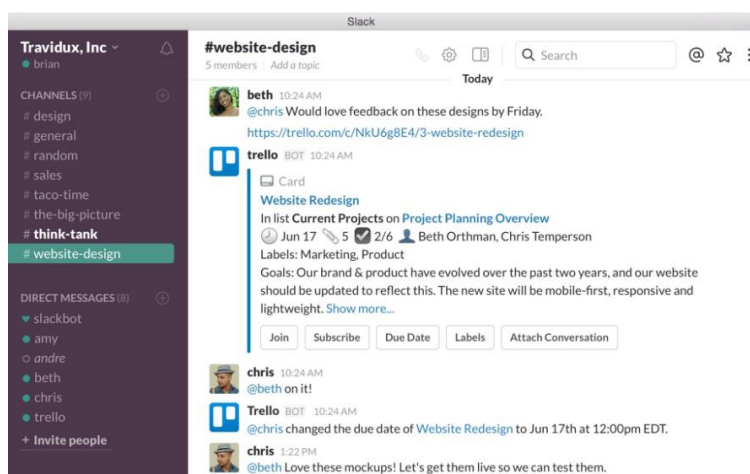


Рис 1.2 – вигляд інтерфейсу

Причиною успіху творець Slack Стюарт Баттерфілд називає роботу з клієнтами, швидкість внесення змін і побажань клієнтів. Компанії важливий кожен клієнт і кожна думка.

Розробник - Slack Technologies;

Архітектура - Microsoft Windows, macOS, iOS, Android и GNU/Linux

Мова реалізації - Electron (C++, JavaScript, etc.)

Перелік функцій:

- Для кожної теми можна створити окремий канал. Спілкування по темі продажів, розробці, фінансів буде проходити в окремому чаті;
- Обговорення окремих повідомлень;
- Робота як з комп'ютера, так і з телефону;
- Швидке додавання посилань і файлів;
- Зручний доступ до архіву повідомлень і файлів;
- Гнучка настройка повідомлень;
- Цитування і згадка учасників;

Мінусами можна назвати відсутність української мови, тому поріг входження для співробітників буде вище. Також треба пам'ятати, що Slack це просто чат. Хоч інтеграції і допомагають працювати з іншими процесами, але вони не дають повноцінно ними управляти.

*Flock*

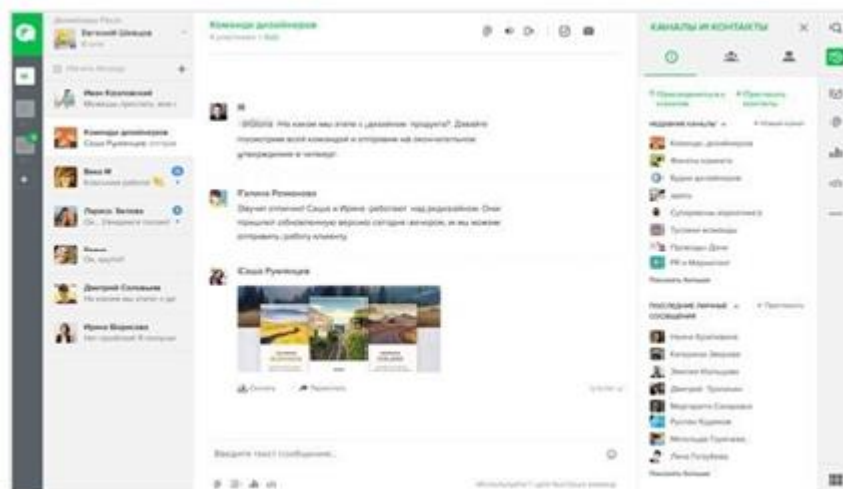


Рис 1.3 – вигляд інтерфейсу

Розробник - Flock FZ-LLC

Архітектура - iOS Android Chrome macOS Windows

Мова реалізації – C++

Ключові можливості Flock:

- Публічні канали.
- Приватні канали.
- Спілкування в режимі реального часу.
- Спільне використання файлів.
- Створення опитувань і списків завдань.

Плюси системи:

- Присутнє відправлення повідомлень з гарантованим відправленням;
- Присутні можливість створювати голосування;
- Є можливість відтворення відео з ютуба прямо в чат;

Мінуси системи:

- Є проблеми з локалізацією

*Sqwiggle*

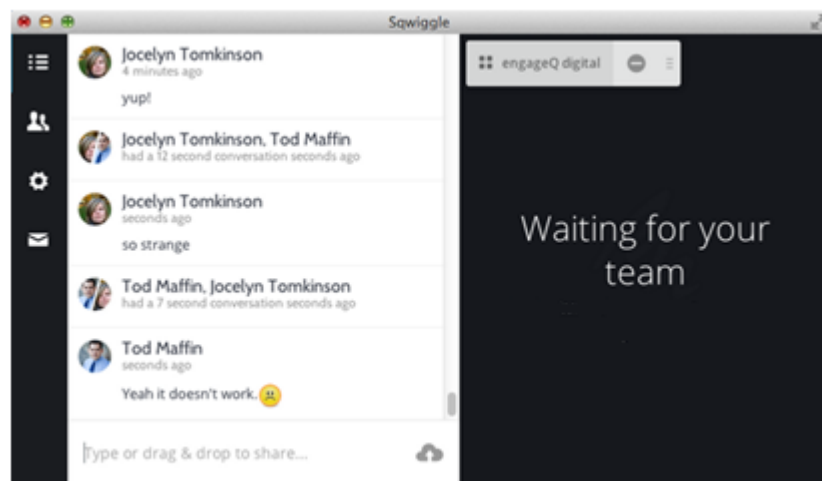


Рис 1.4 – вигляд інтерфейсу

Sqwiggle - це платформа для синхронної спільної роботи розподіленої команди. Все дійство зосереджено в одному вікні в форматі чату. Збоку виводяться коментарі і доступно відстеження завдань. Підключення активно

протягом усього дня, тому в реальному часі можна спостерігати за тим, хто на місці, а хто ні.

У Sqwiggle є кілька «робочих кімнат» для різних відділів, команд або груп. Команди не обмежуються чатами і відеоконференціями. В системі є можливість ділитися з учасниками посиланнями, фрагментами коду, знімками, роликами і статусами.

Розробник – Sqwiggle inc

Архітектура - Mac, браузерний додаток

Можливості Sqwiggle:

- Сповіщення про активність та діях.
- Запрошення електронною поштою.
- Стрічка з усіма подіями.
- SSL-з'єднання.
- Низьке завантаження каналу при бездіяльності.

Мінуси:

- Немає локалізації
- Немає версії під Windows або Linux

Таблиця 1.3 – Результати порівняння систем для автоматизації спілкування співробітників

Критерії оцінки / функції	Slack	Flock	Sqwiggle
Призначення, причина розробки	Підвищення продуктивності праці та ефективності планування.		
Локалізація	Відсутня	Наявна	Відсутня
Підтримка Windows	Наявна	Наявна	Відсутня
Підтримка Linux	Наявна	Відсутня	Відсутня

Підтримка MacOS	Відсутня	Наявна	Наявна
Web-версія	Відсутня	Наявна	Наявна
Мобільна версія	Відсутня	Наявна	Наявна
Сповіщення про дії	Наявна	Наявна	Відсутня
Відправка файлів	Наявна	Наявна	Відсутня
Відтворення відео з популярних сервісів	Відсутня	Наявна	Відсутня
Створення опитувань	Відсутня	Наявна	Відсутня
Ціна	≈10\$ за люд.	≈15\$ за люд.	≈8\$ за люд.

У зв'язку з тим що, жоден з варіантів не надає повністю необхідного функціоналу, оскільки данні зберігаються на серверах інших платформ, що є зовсім не припустимим, усі варіанти мають незамалу ціну та жоден з варіантів не надає можливості модифікації системи під свої потреби, було вирішено розробити власну систему.

### **1.7. Постановка задачі на автоматизацію**

Зараз розвиток інформаційних технологій прогресує все швидше. Було розроблено багато стандартів, протоколів та форматів для зберігання та передачі інформації між комп'ютерами. В даний час існує велика кількість програм обміну даними між комп'ютерами. Тут відіграють особливу роль поштові програми. Багато з цих програм уже настільки вдосконалені, що можуть обмінюватися файлами, голосом та відео.

Однак головною особливістю цих програм є обов'язкове підключення до сервера, що дає можливість використовувати ці програми в складних системах з великою кількістю учасників.

За допомогою сучасних засобів розробки візуального програмного забезпечення можна швидко створити власні програми електронної пошти. Ця стаття охоплює створення мікросервісного додатка для чату за допомогою середовища візуального розвитку Visual Studio.

### **Цілі розробки**

Розробити мікросервісний додаток "Чат" з підтримкою файлового обміну.

### **Завдання розробки**

Розробити мікросервісний додаток "Чат" з підтримкою файлового обміну з наступною функціональністю:

- наявність клієнт-серверної архітектури;
- обмін повідомленнями;
- передача файлів.

### **Постановка завдання та розробка алгоритму роботи програми**

Необхідно розробити клієнт-серверний додаток "Чат" з підтримкою файлового обміну. Додаток має включати клієнтську і серверні частини. Функціональність програми повинна містити можливість передачі тексту і файлів між клієнтом і сервером.

Наведемо приблизний алгоритм роботи програми. Додаток складається з серверної і клієнтської частини, в подальшому - сервер і клієнт. Сервер в робочому стані постійно знаходиться в стані готовності прийняти з'єднання від клієнта. Клієнт ініціалізує з'єднання. Після успішного з'єднання клієнт і сервер можуть надсилати один одному текст. Відправка файлів реалізована на стороні сервера, прийом файлів - на стороні клієнта. Розірвати з'єднання можуть як клієнт, так і сервер. Алгоритм роботи зображено на рис 1.5.

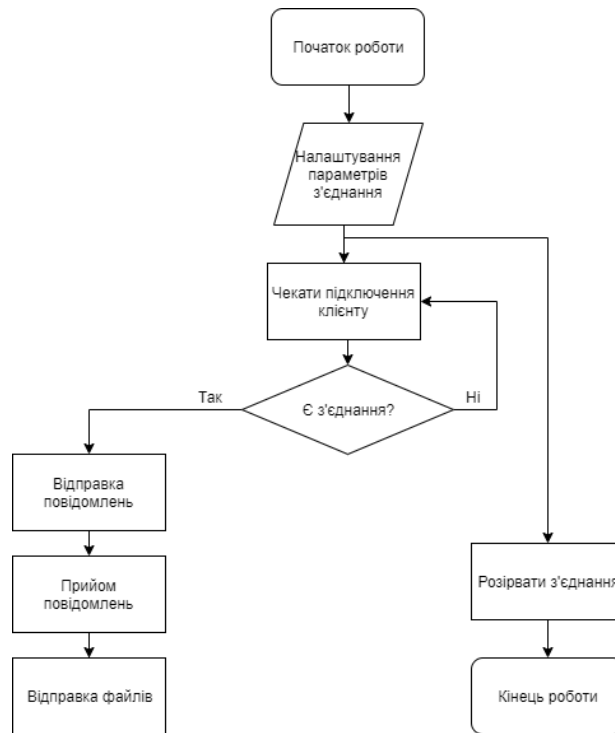


Рис 1.5 – алгоритм роботи мікросервісного чату

## 1.8. Вимоги до обладнання

### 1.8.1 Вимоги до апаратного забезпечення

Крос-платформеність - здатність програмного забезпечення працювати з двома і більше апаратними платформами і (або) операційними системами. Забезпечується завдяки використанню високорівневих мов програмування, середовищ розробки і виконання, що підтримують умовну компіляцію, компоновку і виконання коду для різних платформ.

На даний час уся серверна частина є кросплатформеною. Тобто сервери та Арі можна заhostити як на власних серверах Windows так і на Linux та MacOS.

#### Сервер:

- Процесор з тактовою частотою 2.0 GHz або більше, рекомендовано два або більше ядер, оскільки програма використовує багатопоточність;
- Оперативна пам'ять(RAM) 8 Гбайт або більше;

- Жорсткий диск(HDD): 1Гбайт вільного простору.

#### **Клієнт:**

- DirectX 9-сумісна відеокарта з об'ємом відеопам'яті не менше 1 гбайт;
- Оперативна пам'ять(RAM) 4 Гбайт або більше;
- Жорсткий диск(HDD): 500 мбайт вільного простору.
- Процесор з тактовою частотою 2.0 GHz або більше, рекомендовано два або більше ядер, оскільки програма використовує багатопоточність;

#### **1.8.2. Вимоги до програмного забезпечення**

- Операційна система клієнта: Windows (7, 10);
- Будь-яка версія серверу Windows, Linux, MacOS старше 2010 року;
- Microsoft SQL SERVER 2018 Express;

#### **1.9. Функції системи**

Основними функціями, які має виконувати система, є:

- Обмін повідомленнями між користувачами
- Обмін файлами
- Захист даних користувачів та повідомлень;

#### **1.10. Вхідні та вихідні дані системи**

Вхідними даними до системи є:

Дані користувача

Повідомлення користувачів

Файли користувачів

Виклик певних процедур та функцій

Вихідними даними до системи є:

Оброблені дані користувача

Оброблені повідомлення

Оброблені файли

Оброблені процедури та функції

## **2. ОПИС КОМПЛЕКСУ ЗАДАЧ АВТОМАТИЗАЦІЇ**

### **2.1. Обґрунтування вибору засобів розробки системи *CA ERwin Process Modeler***

ВРwin допомагає чітко зафіксувати важливі аспекти бізнес-процесу: заходи, які слід здійснити, методи їх впровадження та контролю, необхідні для цього ресурси та візуалізацію результатів цих заходів. Аналітики та дизайнери моделей можуть співвідносити корпоративні ініціативи та завдання з бізнес-вимогами, інформаційною архітектурою та процесами проектування додатків. Це створює глобальний імідж компанії: робочі процеси в невеликих відділах зі складними організаційними функціями.

CA ERwin Process Modeler (раніше відомий як AllFusion Process Modeler) - програмний продукт у галузі впровадження технології CASE.

Це дозволяє описати, проаналізувати та моделювати моделі даних - конструктор моделі метаданих. Займає одне з провідних позицій у своєму ринковому сегменті. В даний час виробляє Computer Associates. Поширюється комерційно.

Містить три стандартні методи: IDEF0 (функціональне моделювання), DFD (моделювання потоку даних) та IDEF3 (моделювання робочого процесу). Ці методи по-своєму унікальні. Кожен з них може бути виконаний окремо за допомогою ВРwin, але їх поєднання, включене в модель, дає аналітику повну картину домену замовника.

#### **2.1.1. Модель AS-IS і TO-BE**

Перш ніж намагатися вибрати існуючу або створити власну інформаційну систему, а потім впровадити її, необхідно проаналізувати, як працює система в даний час. Для цього будується функціональна модель AS-IS. Аналіз цієї функціональної моделі дозволяє зрозуміти, де знаходяться найбільш слабкі місця, в чому полягатимуть переваги нових бізнес-процесів. Деталізація бізнес-процесів дозволяє виявити недоліки. Знайдені в моделі AS-

IS недоліки можна виправити при створенні моделі TO-BE - моделі нової організації бізнес-процесів. Модель TO-BE потрібна для оцінки наслідків впровадження інформаційної системи і аналізу альтернативних шляхів виконання роботи і документування того, як система буде функціонувати в майбутньому.

Модель AS-IS - це модель «як є», тобто модель вже існуючого процесу / функції. Обстеження процесів є обов'язковою частиною будь-якого проекту створення або розвитку системи. Побудова функціональної моделі AS-IS дозволяє чітко зафіксувати якісь інформаційні об'єкти використовуються при виконанні функцій різного рівня деталізації. На основі аналізу поточних процесів інформаційної навчальної системи була створена наступна AS-IS модель, яка дозволяє виділити і систематизувати процеси, що протікають в даній системі при її функціонуванні.

2. Спочатку будується модель існуючої організації роботи - AS-IS (як є). Аналіз функціональної моделі дозволяє визначити:

- найбільш слабкі місця переваги нових бізнес-процесів
- глибину змін, яким піддається існуюча структура організації бізнесу.
- Ознаками неефективної роботи діяльності можуть бути:
- даремні, некеровані і дублюються роботи
- неефективний документообіг
- відсутність зворотних зв'язків з управління
- відсутність зворотних зв'язків по входу

Прогалини, виявлені в моделі AS-IS, можна виправити, створивши модель TO-BE (як це буде) - модель нової організації бізнес-процесів. Модель TO-BE необхідна для аналізу альтернативних методів роботи та документування того, як компанія буде вести бізнес у майбутньому.

Створюючи модель AS-IS, розробник може допустити помилку - створивши ідеалізовану модель (наприклад, модель, яка була побудована на

знаннях менеджера, а не конкретного підрядника). Цей шаблон містить неправильну та спотворену інформацію і називається TO-BE (як і належить).

Технологія проектування IP означає спочатку створити модель AS-IS, потім проаналізувати та вдосконалити бізнес-процеси, тобто створити модель TO-BE. І лише на основі моделі TO-BE створюються модель даних, прототип, а потім і кінцева версія IC.

### **2.1.2. Нотація IDEF0**

Найбільш популярне позначення моделювання бізнес-процесів засноване на методі аналізу структури SADT. Бізнес-процеси в позначенні IDEF0 представлені у вигляді прямокутника, а стрілки вказують на зв'язок з іншими процесами та зовнішнім середовищем. Особливістю оцінки є:

Можливість розділити процеси на підпроцеси і таким чином створити ієрархічні моделі бізнес-процесів.

Існує чотири типи стрілок: три типи вводу - вхід, управління та механізм (це дозволяє більш гнучко описувати логіку використання введення в процесі для подальшого аналізу) та вихід.

Сторінка процесу та пов'язана стрілка важливі для IDEF0:

- стрілка вліво - введення бізнес-процесу - інформація (документ), перетворена під час процесу;

- стрілка вправо - вихід з бізнес-процесу - перетворена інформація (документ);

- Вгорі - стрілка - Управління бізнес-процесами - Інформація або документ, який визначає, як слід виконувати бізнес-процес і як перетворюється вхід на вихід.

- Внизу, стрілка - механізм бізнес-процесу, який перетворює вхід у вихід: співробітників або обладнання. Механізм не повинен змінюватися протягом циклу процесу.

Позначення IDEF0 використовується для створення верхнього рівня моделі бізнес-процесів. На нижньому рівні може бути описаний алгоритм (сценарій) процесу з метою зміни рейтингу на процес підрахунку балів.

### **2.1.3. Нотація IDEF3**

Цей метод призначений для моделювання послідовності дій та взаємозалежності між ними всередині процесів. Моделі IDEF3 можна використовувати для деталізації функціональних блоків IDEF0 без діаграм розкладання.

Діаграми IDEF3 показують дію як прямокутник. Дії названі дієсловами або назвами дієслів. Кожній дії присвоюється унікальний ідентифікаційний номер. Усі комунікації в IDEF3 є односпрямованими та організованими зліва направо.

### **ERwin Data Modeler**

Завдяки ERwin Data Modeler менеджери проектів можуть ретельно документувати структуру бази даних, переглядати звіти про якість презентації та забезпечувати ефективне управління проектами, використовуючи спільне дизайнерське середовище AllFusion Model Manager (раніше ModelMart).

Оскільки ERwin Data Modeler підтримує роботу з базами даних на фізичному рівні, враховуючи особливості кожної бази даних, адміністратори баз даних можуть використовувати її для досягнення максимальної продуктивності інформаційної системи. Розробники, що використовують ERwin Data Modeler, можуть спочатку використовувати візуальні інструменти для опису схеми бази даних, а потім автоматично генерувати файли даних для вибраної реляційної бази даних (прямий дизайн). Тригери, що забезпечують кількість посилань на цілісність бази даних, також генеруються автоматично. ERwin Data Modeler підтримує позначення дизайну IDEF1x, IE та розмірних даних. Користувач візуально описує структуру даних.

За допомогою ERwin Data Modeler ви можете відновити логічну структуру даних у існуючих файлах баз даних. Це відомо як зворотна інженерія. З одного боку, це дозволяє передавати структуру бази даних (але не дані!) з однієї бази даних в іншу, а з іншого - досліджувати старі проекти. Це найчастіше відбувається при переході від однієї технології до іншої (з файлового сервера на клієнтський сервер) і при зміні сервера баз даних. На

основі моделі даних можна створити звіти, які значно спрощують процес документування технічного проекту.

ERwin підтримує прямий та зворотний дизайн 20 типів баз даних від різних виробників, від офісів до реляційних баз даних до спеціалізованих баз даних для створення сховищ даних.

## СУБД

Система управління базами даних (СУБД) складається з низки лінгвістичних інструментів та програмного забезпечення, за допомогою яких багато користувачів можуть створювати, керувати і ділитися базою даних.

За допомогою систем управління базами даних ви можете комбінувати та обробляти велику кількість інформації, сортувати її, сортувати за певними критеріями тощо.

Просте використання СУБД дозволяє створювати нові бази даних, не вдаючись до програмування та використовувати лише інтегровані функції. СУБД пропонує точність, повноту та послідовність даних, а також простий доступ до них.

Вибираючи базу даних, дуже важливо вибрати базу даних, яка найкраще відповідає вимогам інформаційної системи.

### Microsoft SQL Server

Найважливішими характеристиками СУБД Microsoft SQL Server є наступні:

- простота адміністрування;
- можливість підключення до Web;
- швидкодія й функціональні можливості механізму сервера СУБД;
- наявність засобів віддаленого доступу.

Ця система має чудові засоби реплікації, за допомогою яких можна синхронізувати дані з персонального комп'ютера з інформацією з бази даних і навпаки. Адже ця база даних - це сучасна і всебічна база даних, ідеальна для малого та середнього бізнесу.

Тому можна сказати, що найкращим варіантом розробки бази даних обліку врожаю сьогодні є Microsoft SQL Server, оскільки ця система автоматизації роботи на малих та середніх підприємствах була розроблена з невеликою складністю та реалізована на платформі Windows.

## ORM

Особливості Entity Framework:

- Інфраструктура формування концептуального об'єктного представлення даних за допомогою сутностей (Entities)
- Реалізація класичних задач ORM
- Абстрагування від схеми зберігання
- Гнучке співставлення

Технологія Entity Framework складається з ряду прикладних технологій ADO.NET, які дозволяють розвиватись у зв'язку з обробкою даних. Архітектори та розробники додатків, які зосереджуються на обробці даних, повинні враховувати необхідність досягнення двох абсолютно різних цілей. Ви повинні моделювати суть, з'єднання та логіку вирішених бізнес-завдань та працювати з ядрами СУБД, які використовуються для зберігання та отримання даних. Дані можуть бути розподілені по декількох системах зберігання даних, кожна з яких використовує власні протоколи. Навіть додатки даних, що працюють в одній системі зберігання, повинні відповідати вимогам системи зберігання даних вимогам до написання коду. ефективний та зручний.

У Entity Framework розробники можуть працювати з даними, представленими у вигляді певних об'єктів і властивостей, таких як клієнти та їх адреси, без необхідності посилатися на базові таблиці бази даних, в яких зберігаються ці дані. Ця можливість виникає при переході на більш високий рівень абстракції, де розробники можуть працювати з даними та використовувати менше коду для створення та управління додатками, орієнтованими на дані.

## .Net

Мова програмування C #, розроблена в Microsoft командою програмістів під керівництвом Андера Хайльсберга та Скотта Вільтамата, була спеціально розроблена для платформи .NET. C # також є веб-орієнтованою мовою програмування.

C # - це повністю об'єктно-орієнтована візуальна мова програмування, в якій програми створюються за допомогою інтегрованого середовища розробки (IDE). У цьому середовищі програміст може писати, виконувати, тестувати та налагоджувати програми, написані на C #. Швидке створення програмних додатків за допомогою IDE відоме як швидка розробка програм (RAD).

C # використовується для створення портативного коду. Всі змінні автоматично ініціалізуються середовищем і мають стандартний захист, який дозволяє уникнути небезпечних ситуацій без ініціалізації, модифікацій об'єкта або спроб здійснити недійсний перетворення типу.

Система рівномірного типу в C #: кожен тип вважається об'єктом. Об'єкти збираються в просторі імен (простори імен), за допомогою яких можна посилатися на все програмно. За допомогою C # ви можете використовувати набрані розширювані метадані, які можна приєднати до об'єкта. Архітектура проекту може визначати локальні атрибути, які призначаються всім мовним елементам - класам, інтерфейсам тощо.

Вибір платформи .NET базується на ряді переваг:

- вся платформа .NET ґрунтується на єдиній об'єктно-орієнтованій моделі;
- в склад платформи .NET входить «збиральник сміття», який звільняє ресурси, що захищає програми від втрат пам'яті і від необхідності звільняти ресурси;
- будь-яка програма, розроблена з допомогою .NET є автономною, в тому смислі, що не залежить від інших програм та від ОС;
- встановлення програми може бути проведене звичайним копіюванням файлів;

- використання безпечних типів даних, що підвищує надійність програми та сумісність;

- програма взаємодіє з єдиною моделлю обробки помилок;

- програми, написані на різних мовах можуть легко взаємодіяти;

- зручний спосіб повторного використання коду;

- вбудована підтримка автоматичної генерації XML-документації.

### *Net Core*

NET Core - це новітня універсальна платформа розробки, підтримувана Microsoft. Вона працює на різних платформах і була перероблена таким чином, щоб зробити .NET швидким, гнучким і сучасним. Це є одним з основних вкладів Microsoft. Тепер розробники можуть створювати додатки для Android, iOS, Linux, Mac і Windows за допомогою .NET, все з відкритим вихідним кодом.

### Характеристики .NET Core

#### Відкритий вихідний код

- .NET Core - це реалізація з відкритим вихідним кодом, яка використовує ліцензії MIT і Apache 2.
- .NET Core є проектом .NET Foundation і доступний на GitHub.
- Як проект з відкритим вихідним кодом, він просуває прозоріший процес розробки і сприяє активному і зацікавленій спільноті.
- Крос-платформна
- Додаток, реалізоване в .NET Core, може бути запущений, і його код може бути повторно використаний незалежно від мети вашої платформи.

В даний час він підтримує три основні операційні системи (ОС)

- Windows
- Linux
- MacOS

Операційні системи (ОС), ЦП і сценарії додатків з часом будуть розширюватися, надані Microsoft, іншими компаніями і приватними особами.

- гнучке розгортання
- Для додатків .NET Core може бути два типи розгортання:
- Фреймворк-залежне розгортання
- автономне розгортання

Під час розгортання на основі інфраструктури ваш додаток залежить від загальносистемної версії .NET Core, на якій встановлено ваші додатки та сторонні залежності.

При розгортанні в автономному режимі версія .NET Core, що використовується для створення програми, також розгорнута в залежності від програми та третіх сторін і може працювати паралельно з іншими версіями.

## WPF

Технологія WPF Windows Presentation Foundation (WPF) - графічна підсистема в .NET Framework, безпосередньо пов'язана з XAML. WPF з .NET Framework інтегрований у Windows Vista, а також може бути встановлений у Windows XP Service Pack 2 та Windows Server 2003 [14].

Це перше реальне оновлення користувальницького інтерфейсу після Windows 95. Він включає нове ядро для заміни GDI та GDI +, використовуваного на попередній платформі Windows. WPF - це об'єктно-орієнтований функціональний шар 54 високого рівня, який можна використовувати для створення двовимірних та тривимірних інтерфейсів.

## Мова розмітки XAML

XAML (скорочення розширюваної мови розмітки додатків) - мова розмітки, яка використовується для інстанціфікації об'єктів .NET. Хоча XAML - це технологія, яка може бути застосована до багатьох різних областей, її головна мета - розробити користувацькі інтерфейси WPF. Іншими словами, документи XAML визначають розташування панелей, кнопок та інших елементів керування, які складають вікна у програмі WPF. Вам навряд чи

доведеться писати код XAML вручну. Натомість використовуйте інструмент, який генерує необхідний код XAML.

## 2.2. Проектування бази даних

За допомогою CA ERwin Process Modeler було побудовано функціональну модель (див. додатки А і Б).

За допомогою CASE-засобу ERWin Data Modeler було побудовано логічну та фізичну модель (див. додатки А і Б).

На основі побудованої моделі створюємо класи моделей таблиць та в класі контексту бази даних моделі властивості з цими моделями.

```
public DbSet<User> ChatUsers;
5 references
public DbSet<UserAccountInfo> UsersAccountInfo { get; set; }
7 references
public DbSet<Chat> Chats { get; set; }
8 references
public DbSet<ChatForGroup> ChatsForGroup { get; set; }
8 references
public DbSet<ChatForTwo> ChatsForTwo { get; set; }
5 references
public DbSet<GroupsUsers> GroupsUsers { get; set; }
9 references
public DbSet<Message> Messages { get; set; }
15 references
public DbSet<MessageStatus> MessagesStatus { get; set; }
3 references
public DbSet<ForwardChatMessage> ForwardMessages { get; set; }
1 reference
public DbSet<FileDb> Files { get; set; }
```

Рис 2.1 – контекст БД

За допомогою FluentApi та dataannotations налаштовуємо зв'язки між таблицями та тригери.

Таблиця користувачів зберігає дані користувача, які необхідні для роботи системи в цілому.

Id	nvarchar(450)	<input type="checkbox"/>
UserName	nvarchar(256)	<input checked="" type="checkbox"/>
NormalizedUserName	nvarchar(256)	<input checked="" type="checkbox"/>
Email	nvarchar(256)	<input checked="" type="checkbox"/>
NormalizedEmail	nvarchar(256)	<input checked="" type="checkbox"/>
EmailConfirmed	bit	<input type="checkbox"/>
PasswordHash	nvarchar(MAX)	<input checked="" type="checkbox"/>
SecurityStamp	nvarchar(MAX)	<input checked="" type="checkbox"/>
ConcurrencyStamp	nvarchar(MAX)	<input checked="" type="checkbox"/>
PhoneNumber	nvarchar(MAX)	<input checked="" type="checkbox"/>
PhoneNumberConfirmed	bit	<input type="checkbox"/>
TwoFactorEnabled	bit	<input type="checkbox"/>
LockoutEnd	datetimeoffset(7)	<input checked="" type="checkbox"/>
LockoutEnabled	bit	<input type="checkbox"/>
AccessFailedCount	int	<input type="checkbox"/>

Рис 2.2 – таблиця користувачів

Таблиця інформації про акаунт має такий вигляд:

Id	nvarchar(450)	<input type="checkbox"/>
UserId	nvarchar(450)	<input checked="" type="checkbox"/>
FirstName	nvarchar(MAX)	<input checked="" type="checkbox"/>
LastName	nvarchar(MAX)	<input checked="" type="checkbox"/>
Status	nvarchar(MAX)	<input checked="" type="checkbox"/>
City	nvarchar(MAX)	<input checked="" type="checkbox"/>
Country	nvarchar(MAX)	<input checked="" type="checkbox"/>
Job	nvarchar(MAX)	<input checked="" type="checkbox"/>
Company	nvarchar(MAX)	<input checked="" type="checkbox"/>
Birthday	datetime2(7)	<input checked="" type="checkbox"/>
PostAddress	nvarchar(MAX)	<input checked="" type="checkbox"/>
[File]	nvarchar(450)	<input checked="" type="checkbox"/>
UserPhoto	nvarchar(MAX)	<input checked="" type="checkbox"/>
DateCreate	datetime2(7)	<input checked="" type="checkbox"/>

Рис 2.3 – додаткова інформація про користувачів

Таблиця чат містить інформацію про чати та тип чату.

Id	nvarchar(450)	<input type="checkbox"/>
ChatName	nvarchar(MAX)	<input checked="" type="checkbox"/>
CreatorId	nvarchar(450)	<input checked="" type="checkbox"/>
ChatType	int	<input type="checkbox"/>
IsDeleted	bit	<input type="checkbox"/>
Created	datetime2(7)	<input type="checkbox"/>

Рис 2.4 – таблиця чату

Таблиця чатів для двох, містить інформацію про користувачів

Id	nvarchar(450)	<input type="checkbox"/>
ChatId	nvarchar(450)	<input checked="" type="checkbox"/>
FirstUserId	nvarchar(450)	<input checked="" type="checkbox"/>
SecondUserId	nvarchar(450)	<input checked="" type="checkbox"/>

Рис 2.5 – таблиця чату для двох

Таблиця чатів для багатьох людей, містить налаштування чату

Id	nvarchar(450)	<input type="checkbox"/>
ChatId	nvarchar(450)	<input checked="" type="checkbox"/>
IsPublic	bit	<input type="checkbox"/>
CreatorMode	bit	<input type="checkbox"/>
[File]	nvarchar(450)	<input checked="" type="checkbox"/>

Рис 2.6 – таблиця чату для груп

Проміжна таблиця між користувачем та чатом в якому він знаходиться

🔑	GroupId	nvarchar(450)	<input type="checkbox"/>
🔑	UserId	nvarchar(450)	<input type="checkbox"/>
	IsRemoved	bit	<input type="checkbox"/>

Рис 2.7 – проміжна таблиця між користувачами та чатами для груп

Таблиця файлів користувача, до неї входять зображення користувачів, чатів та файли які відправляються в повідомленнях

🔑	Id	nvarchar(450)	<input type="checkbox"/>
	FilePath	nvarchar(MAX)	<input checked="" type="checkbox"/>
	IsImage	bit	<input type="checkbox"/>
	IsAudio	bit	<input type="checkbox"/>
	IsVideo	bit	<input type="checkbox"/>

Рис 2.8 – таблиця файлів користувачів

Таблиця повідомлень містить такі поля, поле контент містить текстове повідомлення, файл - файлове

🔑	Id	nvarchar(450)	<input type="checkbox"/>
	ChatId	nvarchar(450)	<input checked="" type="checkbox"/>
	UserId	nvarchar(450)	<input checked="" type="checkbox"/>
	[Content]	nvarchar(MAX)	<input checked="" type="checkbox"/>
	[File]	nvarchar(450)	<input checked="" type="checkbox"/>
	Created	datetime2(7)	<input type="checkbox"/>
	Updated	datetime2(7)	<input checked="" type="checkbox"/>

Рис 2.9 – таблиця повідомлень користувачів

Таблиця пересланих повідомлень містить посилання на записи в таблицях чат, повідомлення та користувач.

🔑	Id	nvarchar(450)	<input type="checkbox"/>
	ChatId	nvarchar(450)	<input checked="" type="checkbox"/>
	UserId	nvarchar(450)	<input checked="" type="checkbox"/>
	MessageId	nvarchar(450)	<input checked="" type="checkbox"/>
	Created	datetime2(7)	<input type="checkbox"/>

Рис 2.10 – таблиця пересланих повідомлень користувачів

Таблиця статуси повідомлень зберігає дані про його статус (прочитаний або видалений)

Id	nvarchar(450)	<input type="checkbox"/>
MessageId	nvarchar(450)	<input checked="" type="checkbox"/>
ForwardId	nvarchar(450)	<input checked="" type="checkbox"/>
UserId	nvarchar(450)	<input checked="" type="checkbox"/>
IsRead	bit	<input type="checkbox"/>
IsDeleted	bit	<input type="checkbox"/>

Рис 2.11 – таблиця статусів повідомлень користувачів

Далі необхідно створити файл міграції. В консолі Visual Studio набираємо:

Enable-Migrations – включає можливість міграції

Add-Migration – додає файли міграції БД до проекту

При першому запуску сервера БД автоматично згенерується на жорсткому диску. Прилад підключення БД до проекту.

```
// ===== Add DbContext =====
services.AddDbContext<Context>(options =>
    options.UseSqlServer(Configuration.GetConnectionString("DefaultConnection"), b => b.MigrationsAssembly("ChatDbCore"));
```

Рис 2.12 – приклад підключення БД до проекту

Вигляд схеми БД наведено в додатку.

## 2.3. Архітектура системи

### 2.3.1 Мікросервісна архітектура

Термін "архітектура мікропослуг", також відомий як мікросервіс, з'явився в середині 2010-х для опису певного архітектурного стилю розробки програмного забезпечення. Цей стиль розвитку поширився завдяки розвитку гнучких практик розвитку та DevOps. Зараз багато уваги приділяється архітектурі мікросервісів: статті, блоги, дискусії в соціальних мережах та презентації конференцій. Якщо говорити про гнучку розробку та доставку складних бізнес-додатків, цей метод розробки пропонує значні переваги.

Стиль архітектури мікросервісів - це підхід, при якому єдине додаток створюється у вигляді серії невеликих автономних, незалежних та не тісно пов'язаних між собою служб, які спілкуються між собою за допомогою простих механізмів, таких як HTTP, gRPC та AMQP. Ці послуги базуються на вимогах компанії (кожен відповідає за конкретний процес) та надаються незалежно в повністю автоматизованому середовищі. Для цих служб існує абсолютний

мінімум центрального управління. Самі сервіси можуть бути написані різними мовами та використовувати різні технології зберігання.

### *Причини використання*

Однією з причин використання мікросервісів є те, що компанії хочуть швидко розширити масштаби, щоб задовольнити мінливі потреби бізнесу і залишитися на крок попереду конкуренції. Завдяки мікросервісам розробники можуть розгортати зміни швидше, безпечніше та з більш високою якістю, тобто підтримувати швидкість розробки продукту, навіть якщо вони вже не піддаються вимірюванню. Нарешті, тісно пов'язані служби дозволяють вносити зміни з більшою частотою ітерації, тим самим мінімізуючи вплив змін на решту системи.

Однак ми не повинні забувати, що такий підхід збільшує складність всього проекту. Для моніторингу та управління нам потрібні DevOps, і між ними та розробниками повинні бути тісні стосунки та хороша взаємодія. Коли ми працюємо з мікросервісами, ми повинні надавати більше, система моніторингу ускладнюється і кількість можливих помилок різко збільшується. Тому сильна культура DevOps дуже важлива для компанії.

### *Мікросервіси vs моноліт*

- Мікросервіси проти моноліту
- Щоб ознайомитись з архітектурою мікросервісів, ви повинні порівняти цей стиль розвитку з так званим монолітним стилем.
- Моноліт будується в цілому. Усі зміни, навіть найменші, потребують реструктуризації та розгортання всієї програми. З часом стає складніше підтримувати хорошу модульну структуру. Зміни в логіці модуля, як правило, впливають на код інших модулів. Монолітні програми також можуть бути важкими для масштабування, коли різні модулі мають суперечливі потреби в ресурсах. Наприклад, модуль може реалізувати логіку обробки зображень, інтенсивно використовуючи процесор. Інший модуль може вимагати використання пам'яті. Однак, оскільки ці модулі постачаються разом, вам доведеться йти на компроміс при виборі

обладнання. Ви повинні масштабувати всю програму, навіть якщо вона потрібна лише для одного модуля цієї програми.

- На відміну від мікросервісів - кожна мікросервіс надається окремо. Тож якщо ви зміните що-небудь у будь-якій з них, ви можете розгорнути ці зміни, не торкаючись інших мікросервісів, які все ще можуть працювати.
- Як результат, веб-проекти часто розробляються як окремі сервіси, які можна розгортати та масштабувати окремо.
- Однак архітектуру мікросервісу піддавали критиці з часу її створення. Нові проблеми з їх реалізацією включають:
- Затримка в мережі: якщо в модулях виконується кілька функцій, а взаємодія відбувається локально, архітектура мікросервісу вимагає розповсюдження модулів та їх взаємодії в мережі.
- Формати повідомлень: відсутність стандартизації та необхідність узгодження форматів обміну практично для кожної пари взаємодіючих мікросервісів призводять до потенційних помилок, так і до налагодження налагодження.
- Балансування навантаження та відмовостійкість
- Складність оперативної підтримки - необхідні кваліфіковані інженери DevOps, постійне розгортання та автоматичний моніторинг. Без усього цього мікросервіси не слід використовувати.
- Тестування мікросервісу може бути виснажливим. Використовуючи моноліт, все, що нам потрібно зробити, це запустити додаток на сервері та переконатися, що ми підключені до бази даних. З мікропослугами перед початком тестів потрібно виконувати кожну послугу.
- Переваги та недоліки мікросервісів
- До переваг належать:
- Послуги працюють швидше, роблячи розробників продуктивнішими та прискорюючи розгортання

- Кожна послуга може надаватися незалежно від інших. Тож якщо ви зміните що-небудь у будь-якій з них, ви можете розгорнути ці зміни, не торкаючись інших мікросервісів, які все ще можуть працювати.
- Кожна служба може масштабуватися окремо.
- Помилка мікросервісу не підірває систему. Проблеми з мікросервісом не повинні завдати шкоди всій програмі. Швидше за все, вони не вплинуть суттєво на продуктивність додатків, особливо для великих програм.
- Усунути будь-яку довгострокову прихильність до технології. Розробляючи новий сервіс, ви можете вибрати новий стек технологій. Будь-яку службу в системі можна замінити. Його можна переписати з нуля за розумний час та бюджет, не потребуючи перебудови всієї системи.
- Мікросервіси, як правило, краще організовані, оскільки кожна мікросервіс має конкретну роботу та не працює з іншими службами. Це може полегшити розуміння, підтримку та тестування.
- Спеціалізовані сервіси також простіше конфігурувати та налаштувати для виконання різних завдань програми (наприклад, розгортання веб-клієнтів та загальнодоступного API).
- Недоліки мікропослуг:
  - Розробники стикаються з додатковою складністю створення розподіленої системи.
  - Складність розгортання. У виробництві існує також оперативна складність розгортання та управління системою, що складається з багатьох різних типів послуг.
  - Коли ви будете нову архітектуру мікросервісу, ви, ймовірно, знайдете багато багатопланових проблем, яких ви не очікували під час розробки.

### **2.3.2 Api**

Абревіатура API розшифровується як «Application Programming Interface» (інтерфейс програмування додатків, програмний інтерфейс програми). Більшість великих компаній на певному етапі розробляють API для

клієнтів або для внутрішнього використання. Щоб зрозуміти, як і яким чином API застосовується в розробці і бізнесі, спочатку потрібно розібратися, як влаштована «всесвітня павутина».

### *Всесвітня павутина й віддалені сервери*

WWW можна уявити як величезну мережу пов'язаних серверів, на яких і зберігається кожна сторінка. Звичайний ноутбук можна перетворити в сервер, здатний обслуговувати цілий сайт в мережі, а локальні сервери розробники використовують для створення сайтів перед тим, як відкрити їх для широкого кола користувачів.

Кожен раз, коли користувач відвідує якусь сторінку в мережі, він взаємодіє з API віддаленого сервера. API - це складова частина сервера, яка отримує запити і відправляє відповіді.

### *API як спосіб обслуговування клієнтів*

Застосування API: мета - сервер сайту повинен безпосередньо звертатися до сервера із запитом на створення події з зазначеними деталями, отримувати відповідь, обробляти його, і передавати відповідну інформацію в браузер, наприклад, повідомлення із запитом на підтвердження користувачеві. API визначає функціональність, яку надає програма (модуль, бібліотека), при цьому API дозволяє абстрагуватися від того, як саме ця функціональність реалізована.

Якщо програму (модуль, бібліотеку) розглядати як чорний ящик, то API - це безліч «ручок», які доступні користувачеві даного ящика, які він може крутити і смикати.

Програмні компоненти взаємодіють один з одним за допомогою API. При цьому зазвичай компоненти утворюють ієрархію - високорівневі компоненти використовують API низькорівневих компонентів, а ті, в свою чергу, використовують API ще більш низькорівневих компонентів.

За таким принципом побудовані протоколи передачі даних по Internet. Стандартний протокол Internet (мережева модель OSI) містить 7 рівнів (від фізичного рівня передачі пакетів біт до рівня протоколів програм, подібних

протоколів HTTP і IMAP). Кожен рівень користується функціональністю попереднього рівня передачі даних і, в свою чергу, надає потрібну функціональність наступного рівня.

Важливо зауважити, що поняття протоколу близьке за змістом до поняття API. І те й інше є абстракцією функціональності, тільки в першому випадку мова йде про передачу даних, а в другому - про побудову комп'ютерних додатків.

API бібліотеки функцій і класів включає в себе опис сигнатур і семантики функцій.

Application Programming Interface (API) програмний інтерфейс взаємодії між системами, що дозволяє:

- Отримувати доступ до бізнес-сервісів підприємства
- Обмінюватися інформацією між системами і додатками
- Спростити взаємодію між компаніями, партнерами, розробниками і клієнтами

### 2.3.3. Проєтування системи

При розробці системи було вирішено поділити її на декілька складових:

- Клієнт
- Сервер
- Арі

Усі складові системи будуть розглянуті через взаємодію з користувацьким інтерфейсом.

*Користувацький інтерфейс*

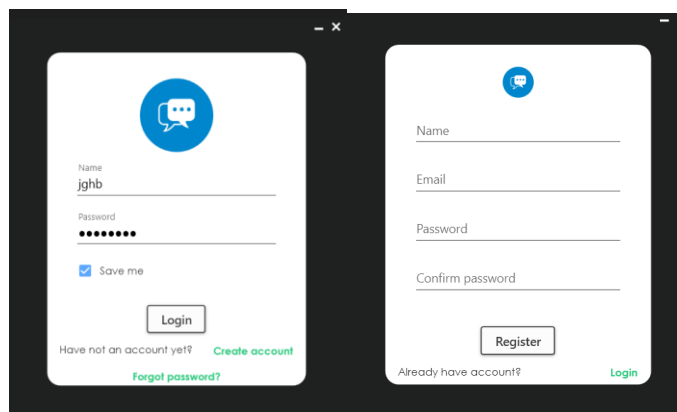


Рис 2.13 та 2.14 – форми авторизації

Перша вікно яке зустрічає користувача являє собою форми логіну та реєстрації. До полів на даних формах підключена валідація даних. При введенні коректних даних посилається запит на АРІ та проводиться авторизація або реєстрація користувача, після чого користувачу надається ключ доступу до системи.

Авторизація (англ. Authorization «дозвіл; уповноваження») - надання певній особі або групі осіб прав на виконання певних дій; а також процес перевірки (підтвердження) даних прав при спробі виконання цих дій. Часто можна почути вираз, що якийсь чоловік «авторизований» для виконання даної операції - це значить, що він має на неї право.

Обліковий запис - збережена в комп'ютерній системі сукупність даних про користувача, необхідна для його розпізнавання (аутентифікації) і надання доступу до його особистих даних і налаштувань.

Для використання облікового запису (іншими словами, для входу в систему під чийось ім'ям) зазвичай потрібне введення імені (логіну, англ. Login) і пароля. Також може вимагатися інша додаткова інформація.

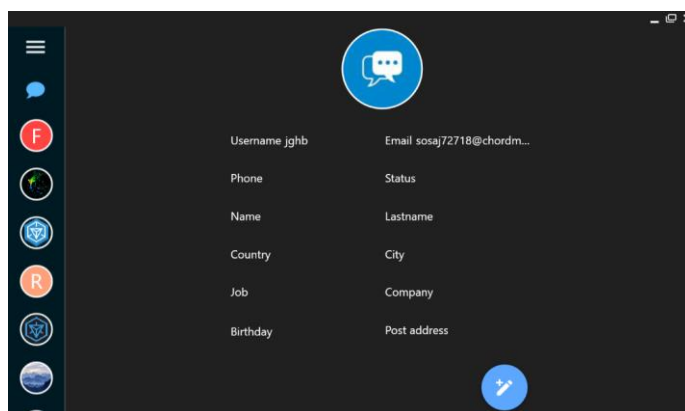


Рис 2.15 – форма користувача

Форма користувача містить інформацію про користувача та можливість редагування цих даних.

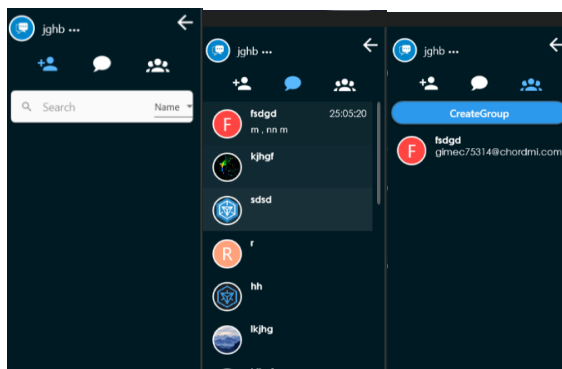


Рис 2.16 – меню

Форма меню та її вкладки. Перша - форма пошуку чатів, друга – усі доступні користувачу чати, третя – створення групи та користувачі з якими ведеться бесіда тета-тет.

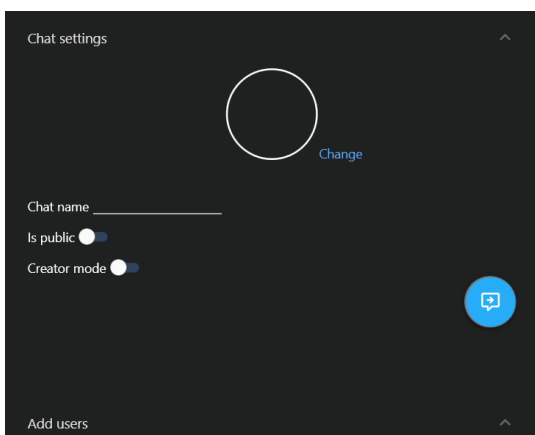


Рис 2.17 – форма створення чату для груп

Форма створення чатів, оновлення даних та додавання нових користувачів.

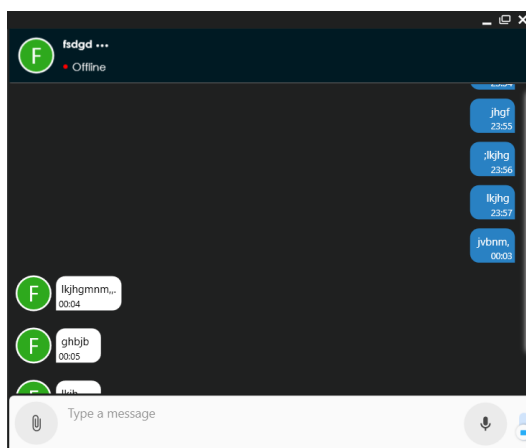


Рис 2.18 – форма чатування

Форма для перегляду та відправки повідомлень.

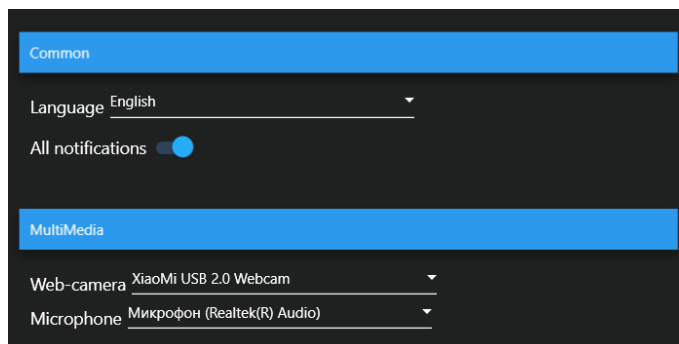


Рис 2.19 – форма налаштувань

Форма налаштувань – містить загальні налаштування які впливають на систему вцілому.

#### 2.4. Техніко-економічне обґрунтування доцільності розробки

Техніко-економічний ефект від впровадження комп’ютерної системи визначається за співвідношенням витрат на розробку системи і прибутком від її впровадження.

Витрати на розробку системи складаються з витрат на розробку програмного забезпечення –  $V_1$ , витрат на придбання і установку комп’ютера  $V_2$ , витрат на підготовку приміщення  $V_3$ , витрат на навчання персоналу  $V_4$ .

$$V_{\Sigma} = V_1 + V_2 + V_3 + V_4 \quad (4.1)$$

Автоматизована система спілкування співробітників “ПРАТ Оболонь”.

1. Ступінь новизни розроблюваних задач – «В» - використання типових проектних рішень за умови їх змін.
2. Група складності алгоритму - 2.
3. Узагальнені дані вхідної та вихідної інформації для системи комунікації співробітників спілкування співробітників “ПРАТ Оболонь” за видами вхідної та вихідної інформації табл. 4.1.

Таблиця 4.1. Узагальнені дані для вхідної та вихідної інформації системи спілкування співробітників “ПРАТ Оболонь”.

Вид інформації	Позначення	К-сть наборів даних
Змінна інформація	ЗІ	m=6
Нормативно-довідкова інформація	НДІ	n=3
Банк (база) даних	БД	p=3
Обробка в режимі реального часу	РЧ	так
Забезпечення телекомунікаційної обробки даних і управління віддаленими об’єктами	ТОУ	ні

Таблиця 4.2. Визначення часу для системи комунікації співробітників “ПРАТ Оболонь”.

Вид системи	Стадія розробки системи	
	Передпроектне дослідження	Технічне завдання
	В	В
Управління організацією праці і зарплатою.	$T_1=67$	$T_2=30$

4. Витрати часу на систему призначену для системи комунікації співробітників “ПРАТ Оболонь” табл. 4.3, а саме на розробку ескізного проекту  $T_1$  і технічного завдання  $T_2$ , будуть наступні.

Таблиця 4.3. Визначення витрат часу

№	Вид системи	Стадія розробки системи			
		Ескізний проект, $T_1$		Технічне завдання, $T_2$	
		В	Г	В	Г
1.	Управління матеріально-технічним постачанням, управління збутом продукції, управління комплектацією, управління експортними та імпортерними поставками	53	35	42	30
2.	Бухоблік, управління фінансовою діяльністю	73	57	48	35
3.	Управління організацією праці і зарплатою, управління кадрами, норми і нормативи, управління охороною праці	67	46	30	19
4.	Управління якістю продукції, управління технологічними процесами, управління стандарти зацією, управління технічною підготовкою виробництва	67	44	31	22
5.	Управління транспортними перевезеннями, управління технічним обслуговуванням виробництва, управління допоміжними службами і енергопостачанням.	70	45	43	26
6.	Управління науково-дослідною роботою (НДР) та ДКР.	67	46	24	15

5. Визначається базове значення витрат часу для стадій “Технічний проект”, “Робочий проект” і “Впровадження”. Для цього використовуються дані для системи комунікації співробітників, табл. 4.4-4.6.

Таблиця 4.4. Технічний проект

Кількість форм вхідної інформації	Кількість форм вихідної інформації				
	1	2	3-4	5-6	7-9
1	30	36	43	49	56
2	42	50	59	68	75
3	49	61	72	83	93
4	57	71	94	96	108
5	64	79	91	104	118
6	70	85	100	114	128
7	75	90	107	123	138
8	79	97	115	130	145
9	84	103	120	137	155
10	88	109	126	145	163

Таблиця 4.5. Робочий проект

Кількість форм вхідної інформації	Кількість форм вихідної інформації				
	1	2	3-4	5-6	7-9
1	44	58	76	92	110
2	59	81	103	128	150
3	71	98	125	157	182
4	82	111	143	176	208
5	90	122	159	195	231
6	97	134	174	211	251
7	104	144	186	227	269
8	110	153	197	272	286
9	116	163	208	255	302
10	123	172	219	268	318

Таблиця 4.6. Впровадження

Кількість форм вхідної інформації	Кількість форм вихідної інформації				
	1	2	3-4	5-6	7-9
1	17	22	29	37	45
2	22	31	40	51	61
3	27	38	50	62	75
4	32	44	57	71	86
5	35	48	64	80	96
6	36	53	70	87	105
7	41	58	75	94	112
8	43	62	81	100	120
9	45	66	85	106	126
10	47	68	90	112	134

Вхідними даними для визначення є:

- Кількість форм вхідної інформації  $V_1=6$
- Кількість форм вихідної інформації  $V_2=3$
- Базове значення витрат часу для стадій “Технічний проект”:  $T_{63}=85$ ;
- Базове значення витрат часу для стадій “Робочий проект”:  $T_{64}=211$ ;
- Базове значення витрат часу для стадій “Впровадження”:  $T_{65}=72$ .

Базове значення витрат часу  $T_6$  коригується за допомогою поправочних коефіцієнтів для всіх стадій розробки автоматизованої системи.

Розрахунок витрат часу для стадії “Технічний проект” ( $T_3$ )

Коефіцієнт трудомісткості робіт  $k_{II}$  визначається за ( 3 ) і з врахуванням коефіцієнтів табл. 4.7:

$$K_{II} = \frac{1.0 \cdot 6 + 0.72 \cdot 3 + 2.08 \cdot 3}{6 + 3 + 3} = 0.75$$

Таблиця 4.7. Коефіцієнти  $k_1, k_2, k_3$  для стадії “Технічний проект”

Вид використаної інформації	Ступінь новизни
	В
$k_1$ (ЗІ)	1.0
$k_2$ (НДІ)	0.72
$k_3$ (БД)	2.08

Таблиця 4.8. Коефіцієнт ступеню новизни проекту,  $k_0$  для системи спілкування співробітників “ПРАТ Оболонь”

Стадія розробки системи	Вид обробки	Ступінь новизни
		В
Технічний проект	РЧ	1.26
Робочий проект	РЧ	1.32
Впровадження	РЧ	1.21

Коефіцієнт ступеню новизни проекту,  $k_0$ , що враховує вид обробки інформації для трьох стадій розробки системи визначається з табл. 8, для системи комунікації співробітників аутсорс ІТ компанії значення в табл. 4.7.

Таблиця 4.9. Коефіцієнти  $k_1, k_2, k_3$  для стадії “Технічний проект”

Стадія розробки проекту	Вид обробки	Ступінь новизни	
		В	Г
Технічний проект	РЧ	1.26	1.1
	ТОУ	1.36	1.15
Робочий проект	РЧ	1.32	1.15
	ТОУ	1.41	1.26
Впровадження	РЧ	1.21	1.05
	ТОУ	1.26	1.1

Витрати часу для стадії “Технічний проект”  $T_3$  розраховуються за (2):

$$T_3 = T_{B3} * k_n * k_0 = 85 * 0.75 * 1.26 = 76$$

Розрахунок витрат часу для стадії “Робочий проект” ( $T_4$ )

Для визначення витрат часу на стадії “робочий проект” використовуються формулу, де  $k_n$  – коефіцієнт, що враховує вид використаної інформації і визначається за формулою:

$$k_n = \frac{1.1 * 6 + 0.58 * 3 + 0.48 * 3}{6 + 3 + 3} = 0.815$$

Таблиця 4.10. Коефіцієнти  $k_1, k_2, k_3$  для стадії “Робочий проект”

Вид використаної інформації	Група складності алгоритму	Ступінь новизни
		В
$k_1$ (ЗІ)	2	1.1
$k_2$ (НДІ)	2	0.58
$k_3$ (БД)	2	0.48

Таблиця 4.11. Коефіцієнт складності контролю вхідної та вихідної інформації  $k_c$

Складність контролю вхідної інформації	Складність контролю вихідної інформації	
	21	22
11	1.16	1.07
12	1.08	1.00

Коефіцієнт, що враховує вид обробки інформації табл. 10. Коефіцієнт складності контролю вхідної та вихідної інформації і визначається з таблиці 4.11. на стадії “Робочий проект” і “Впровадження”.

$$k_c = 1.16$$

Витрати часу T4 вимірюються в людино-днях, розраховується за:

$$T_4 = T_{БЗ} * k_{п} * k_0 * k_c = 211 * 0.815 * 1.26 * 1.16 = 251$$

Поправочні коефіцієнти мають такі ж значення, як і при обрахунку табл. 4.11 і табл. 4.8.

Для стадії визначення загальних витрат часу на “Впровадження” T5 (люд-днів) використовують формулу:

$$T_5 = T_{БЗ} * k_{п} * k_0 * k_c = 72 * 0.815 * 1.26 * 1.16 = 85$$

Таким чином, загальні витрати людської праці на проектування системи за (5.3.5) складають:

$$T_E = 67 + 30 + 76 + 211 + 85 = 469_{(люд-дн.)}$$

Для дипломного проекту кількість робочих годин складає 469 із 7-годинним робочим днем, тому на розробку проекту виділено Ф, днів:

$$\Phi = \frac{469}{7} = 67$$

Для дипломного проекту  $\Phi = 75$  днів. Тоді визначаємо кількість місяців із розрахунку 25 робочих днів.

Кількість місяців на розробку, M:

$$M = \frac{\Phi}{25} = \frac{67}{25} = 2.7$$

Отже, для виконання такого проекту потрібно таку чисельність виконавців Ч, обраховується за:

$$\mathcal{C} = \frac{T_E}{\Phi} = \frac{469}{67} = 7$$

Якщо прийняти, що оплата програміста здійснюється в розмірі 3000 грн, то оплата праці всіх виконавців, яка підраховується за формулою складе:

$$V'_1 = Ч * М * З_{П_{ПР}} = 7 * 2.7 * 3000 = 56700 \text{ грн}$$

Витрати, пов'язані з розробкою програми

1. Розрахунок річного фонду часу роботи ПК.

Дійсний річний фонд часу ПК у годинах дорівнює числу робочих годин у році для оператора, за винятком часу на технічне обслуговування і ремонт ПК.

$$T_{ПК} = 2000 - (6 * 8 + 5 * 12) = 1892 \text{ год.}$$

Оскільки під час виконання дипломного проекту студент в середньому витрачає 450 год. Машинного часу, то величина фонду часу ПК за 9 дорівнює

$$T_{ПК} = 1892 * \frac{450}{2000} = 426 \text{ год}$$

2. Поточні витрати на експлуатацію  $V_1$

Балансова вартість ПК вираховується за (15). Ц<sub>р</sub> – ринкова вартість ПК, орієнтовно складає 4000 грн., к<sub>ун</sub> – коефіцієнт, що враховує витрати на установку і налагодження ПК і дорівнює 0.12.

$$Ц_{ПК} = (1 + k_{п}) = 4000 * (1 + 0.12) = 4480 \text{ грн}$$

Амортизаційні відрахування використання ПК, ЗАМ, обчислюються за (11), норма амортизаційних відрахувань, яка для ПК дорівнює НА = 5:

$$З_{АМ} = \frac{4480}{5} = 896 \text{ грн}$$

Витрати на електроенергію, споживану ПК, визначаються за (12), де потужність ПК, Р<sub>ПК</sub> = 0.4 кВт, Фонд корисного часу роботи ПК, Т<sub>ПК</sub> = 426 год, вартість 1 кВт електроенергії для підприємств, Ц<sub>ел</sub> = 0.74 грн/кВт, коефіцієнт інтенсивного використання ПК, А = 0.9.

$$З_{ел} = Р_{ПК} * T'_{ПК} * Ц_{ел} * А = 0.4 * 426 * 0.74 * 0.9 = 113.48 \text{ грн}$$

З<sub>р</sub> – витрати на поточний ремонт і технічне обслуговування ПК визначаються як 6% від балансової вартості ПК, Ц<sub>ПК</sub>.

$$З_{р} = Ц_{ПК} * 0.06 = 4480 * 0.06 = 268.8 \text{ грн}$$

$Z_{\text{МАТ}}$  – непрямі витрати, пов’язані з експлуатацією ПК, визначаються як 5% від балансової вартості ПК  $C_{\text{ПК}}$ .

$$Z_{\text{МАТ}} = C_{\text{ПК}} * 0.05 = 4480 * 0.05 = 224 \text{ грн}$$

Таким чином, маємо:

Заробітна плата обслуговуючого персоналу (якщо роботи виконуються не на власному ПК);  $Z_{\text{ОП}} = 1680$  грн,  $Z_{\text{АМ}} = 896$  грн,  $Z_{\text{ЕЛ}} = 113.48$  грн,

Поточні витрати на експлуатацію  $V_1''$ , грн, визначаються за (10).

$$V_1'' = Z_{\text{ОП}} + Z_{\text{АМ}} + Z_{\text{Р}} + Z_{\text{МАТ}} = 1680 + 896 + 113.48 + 268.8 + 224 = 3182.28$$

Отже, загальні витрати на розробку програмного забезпечення комп’ютерної системи розраховуються за формулою 13 і складуть:

$$V_1 = V_1' + V_1'' = 56700 + 3182.28 = 59882.28$$

3. Витрати на придбання і установку ПК  $V_2$ .

Витрати на придбання і установку ПК ( $V_2$ ) визначаються:

$$V_2 = V_{\text{ПК}} = 4480 \text{ грн}$$

Якщо немає потреби в купівлі ПК, то ці витрати дорівнюють “0”.

4. Витрати на підготовку приміщення  $V_3$

Ці витрати залежать від стану приміщення, де буде встановлюватись ПК.

Так як пристосоване приміщення є, тому:

$$V_3 = 0 \text{ грн}$$

5. Витрати на навчання персоналу  $V_4$ .

В середньому навчання персоналу триватиме 1 місяць, тому можна вважати, що:

$$V_4 = 2000 \text{ грн.}$$

6. Загальна вартість розробки і впровадження системи.

Загальна вартість розробки і впровадження системи  $V_E$ , вираховується за:

$$V_E = V_1 + V_2 + V_3 + V_4 = 59882.28 + 4480 + 0 + 2000 = 66362.28 \text{ грн}$$

Оскільки норма амортизаційних витрат для комп’ютерних систем  $N_A = 5$ , то для обрахування річного економічного ефекту слід брати до розгляду величину:

$$V_p = \frac{V_2}{H} = \frac{62162.28}{5} = 12432.46 \text{ грн}$$

Річний прибуток  $P_p$  від впровадження системи буде досягнуто за рахунок підвищення якості виробів і орієнтовно складатиме 10000 грн на рік. Коефіцієнт економічної ефективності розробки вираховується за:

$$K_{\text{ЕФ}} = \frac{P_p}{V_p} = \frac{10000}{12432.46} = 0.75$$

Термін окупності розробки дорівнює і визначається за формулою:

$$T_{\text{ОК}} = \frac{1}{K_{\text{ЕФ}}} = \frac{1}{0.75} = 1.33 \text{ року}$$

Таким чином, термін окупності інформаційної системи буде 1.33 року.

### **3. ОХОРОНА ПРАЦІ**

#### **3.1. Охорона праці при розробці програмного забезпечення**

Охорона здоров'я трудящих, забезпечення безпеки умов праці, ліквідація професійних захворювань і виробничого травматизму складає одну з головних турбот людського суспільства. Звертається увага на необхідність широкого застосування прогресивних форм наукової організації праці, зведення до мінімуму ручної, малокваліфікованої праці, створення обстановки, що виключає професійні захворювання і виробничий травматизм.

На робочому місці повинні бути передбачені заходи захисту від можливого впливу небезпечних і шкідливих факторів виробництва. Рівні цих факторів не повинні перевищувати граничних значень, обумовлених правовими, технічними і санітарно-технічними нормами. Ці нормативні документи зобов'язують до створення на робочому місці умов праці, при яких вплив небезпечних і шкідливих чинників на працюючих або усунуто зовсім, або знаходиться в допустимих межах.

##### **3.1.1. Характеристика умов праці програміста**

Науково-технічний прогрес вніс серйозні зміни в умови виробничої діяльності працівників розумової праці. Їх праця стала більш інтенсивним, напруженим, які вимагають значних витрат розумової, емоційної і фізичної

енергії. Це зажадало комплексного рішення проблем ергономіки, гігієни і організації праці, регламентації режимів праці і відпочинку.

В даний час комп'ютерна техніка широко застосовується у всіх областях діяльності людини. При роботі з комп'ютером людина піддається дії ряду небезпечних і шкідливих виробничих факторів: електромагнітних полів (діапазон радіочастот: ВЧ, УВЧ і НВЧ), інфрачервоного і іонізуючого випромінювань, шуму і вібрації, статичної електрики і ін.

Робота з комп'ютером характеризується значною розумовою напругою і нервово-емоційним навантаженням операторів, високою напруженістю зорової роботи і достатньо великим навантаженням на м'язи рук при роботі з клавіатурою ЕОМ. Велике значення має раціональна конструкція і розташування елементів робочого місця, що важливо для підтримки оптимальної робочої пози людини-оператора.

У процесі роботи з комп'ютером необхідно дотримувати правильний режим праці та відпочинку. В іншому випадку у персоналу наголошуються значна напруга зорового апарату з появою скарг на незадоволеність роботою, головні болі, дратівливість, порушення сну, втому і хворобливі відчуття в очах, в попереку, в області шиї і руках.

### **3.1.2. Режим праці**

Як вже було неодноразово відзначено, при роботі з персональним комп'ютером дуже важливу роль грає дотримання правильного режиму праці і відпочинку. В іншому випадку у персоналу наголошуються значна напруга зорового апарату з появою скарг на незадоволеність роботою, головні болі, дратівливість, порушення сну, втому і хворобливі відчуття в очах, в попереку, в області шиї і руках.

Ефективність перерв підвищується при поєднанні з виробничою гімнастикою або організації спеціального приміщення для відпочинку персоналу із зручними м'якими меблями, акваріумом, зеленою зоною і т.п.

Вимоги до режимів праці і відпочинку при роботі з ЕОМ

При організації праці, що пов'язана з використанням ВДТ ЕОМ і ПЕОМ, для збереження здоров'я працюючих, запобігання професійним захворюванням і підтримки працездатності слід передбачити внутрішньо змінні регламентовані перерви для відпочинку та додаткові нетривалі перерви в періоди, що передують появі об'єктивних і суб'єктивних ознак втомлення і зниження працездатності.

При виконанні протягом дня робіт, що належать до різних видів трудової діяльності, за основну роботу з ПК слід вважати таку, що займає не менше 50% часу впродовж робочого часу мають передбачатися:

- перерви для відпочинку і вживання їжі (обідні перерви);
- перерви для відпочинку і особистих потреб (згідно з трудовими нормами);
- додаткові перерви, що вводяться для окремих професій з урахуванням особливостей трудової діяльності.

Тривалість обідньої перерви визначається чинним законодавством про працю і правилами внутрішнього трудового розпорядку підприємства.

За характером трудової діяльності виділено три професійні групи згідно з діючим класифікатором професій:

1. розробники програм (інженери-програмісти) - виконують роботу переважно з відео терміналом та документацією при необхідності і інтенсивного обміну інформацією з ЕОМ і високою частиною прийняття рішень. Робота характеризується інтенсивною розумовою творчою працею з підвищеним напруженням зору, концентрацією уваги на фоні нервово-емоційного напруження, вимушеною робочою позою, загальною гіподинамією, періодичним навантаженням на кисті верхніх кінцівок. Робота виконується в режимі діалогу з ЕОМ у вільному темпі з періодичним пошуком помилок в умовах дефіциту часу;

2. оператори електронно-обчислювальних машин - виконують роботу яка пов'язана з обліком інформації одержаної з ВДТ за попереднім запитом, або тієї, що надходить з нього, супроводжується перервами різної

тривалості, пов'язана з виконанням іншої роботи і характеризується як робота з напруженням зору, невеликими фізичними зусиллями, нервовим напруженням середнього ступеня та виконується у вільному темпі;

3. оператор комп'ютерного набору - виконує одноманітні за характером роботи з документацією та клавіатурою і нечастими нетривалими переключеннями погляду на екран дисплея, з введенням даних з високою швидкістю, робота характеризується як фізична праця з підвищеним навантаженням на кисті верхніх кінцівок на фоні загальної гіподинамії з напруженням зору (фіксація зору переважно на документи), нервово-емоційним напруженням.

Встановлюються такі внутрішньо змінні режими праці та відпочинку при роботі з ЕОМ при 8-годинній денній робочій зміні в залежності від характеру праці:

- для розробників програм із застосуванням ЕОМ, слід призначати регламентовану перерву для відпочинку тривалістю 15 хвилин через кожен годину роботи за ПК;

- для операторів із застосування ЕОМ, слід призначати регламентовані перерви для відпочинку тривалістю 15 хвилин через кожні дві години;

- для операторів комп'ютерного набору слід призначати регламентовані перерви для відпочинку тривалістю 10 хвилин після кожною години роботи за ПК.

У випадках, коли виробничі обставини не дозволяють застосувати регламентовані перерви, тривалість безперервної роботи з ПК не повинна перевищувати 4 години.

З метою зменшення негативного впливу монотонності є доцільним застосовувати чергування операцій усвідомленого тексту і числових даних (зміна змісту роботи). Наприклад, чередування вводу даних та редагування текстів.

В окремих випадках – при хронічних скаргах працюючих з ПК на зорове втомлення незважаючи на дотримання санітарно-гігієнічних вимог до режимів праці і відпочинку, а також застосування засобів локального захисту очей - допускаються індивідуальних підхід до обмеження часу робіт з ПК, зміни характеру праці, чергування з іншими видами діяльності, не пов'язаними з комп'ютером.

Для зниження нервово-емоційного напруження, втомлення зорового аналізатору, поліпшення мозкового кровообігу, подолання несприятливих наслідків гіподинамії, запобігання втомі доцільні деякі перерви використовувати для виконання комплексу спеціальних вправ та активного відпочинку.

Активний відпочинок має полягати у виконанні комплексу гімнастичних вправ, спрямованих на зняття нервового напруження, м'язове розслаблення, відновлення функцій фізіологічних систем, що порушуються протягом трудового процесу, зняття втоми очей, поліпшення мозкового кровообігу і працездатності.

За умови високого рівня напруженості робіт з ПК показане психологічне розвантаження у спеціально обладнаних приміщеннях (в кімнатах психологічного розвантаження) під час регламентованих перерв або в кінці робочого дня.

### **3.1.3. Ергономічні вимоги до робочого місця**

Проектування робочих місць, забезпечених відео терміналами, відноситься до числа найважливіших проблем ергономічного проектування в області обчислювальної техніки.

Робоче місце і взаємне розташування всіх його елементів повинне відповідати антропометричним, фізичним і психологічним вимогам. Велике значення має також характер роботи. Зокрема, при організації робочого місця програміста повинні бути дотримані наступні основні умови: оптимальне розміщення устаткування, що входить до складу

робочого місця і достатній робочий простір, що дозволяє здійснювати всі необхідні рухи і переміщення.

Ергономічними аспектами проектування відео термінальних робочих місць, зокрема, є: висота робочої поверхні, розміри простору для ніг, вимоги до розташування документів на робочому місці (наявність і розміри підставки для документів, можливість різного розміщення документів, відстань від очей користувача до екрану, документа, клавіатури і т.д.), характеристики робочого крісла, вимоги до поверхні робочого столу, можливість регулювання елементів робочого місця.

Головними елементами робочого місця програміста є стіл і крісло. Основним робочим положенням є положення сидячи.

Робоча поза сидячи викликає мінімальне стомлення програміста. Рациональне планування робочого місця передбачає чіткий порядок і сталість розміщення предметів, засобів праці і документації. Те, що потрібно для виконання робіт частіше, розташоване в зоні легкої досяжності робочого простору.

Моторне поле - простір робочого місця, в якому можуть здійснюватися рухові дії людини.

Максимальна зона досяжності рук - це частина моторного поля робочого місця, обмеженого дугами, що описуються максимально витягнутими руками при русі їх в плечовому суглобі.

Оптимальна зона - частина моторного поля робочого місця, обмеженого дугами, описуваними передпліччями при русі в ліктьових суглобах з опорою в точці ліктя і з відносно нерухомим плечем.

На малюнку 3.1 показані зони досяжності рук у горизонтальній площині.

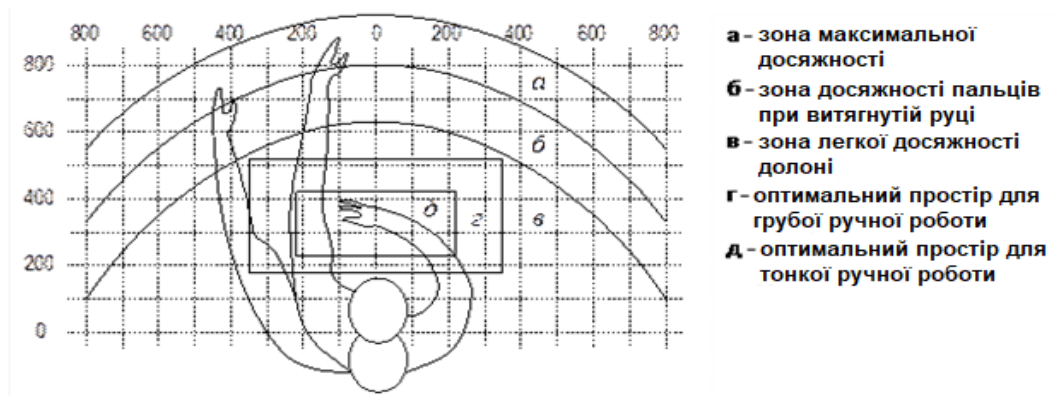


Рис. 3.1 Зони досяжності рук у горизонтальній площині

Оптимальне розміщення предметів праці і документації в зонах досяжності:

- дисплей розміщується в зоні 2 (в центрі);
- системний блок розміщується в передбаченій ніші столу;
- клавіатура - в зоні 5;
- «миша» - в зоні 6 справа;
- сканер в зоні 1 (зліва);
- принтер знаходиться в зоні 3 (праворуч);
- документація необхідна при роботі - в зоні легкої досяжності долоні - 4.

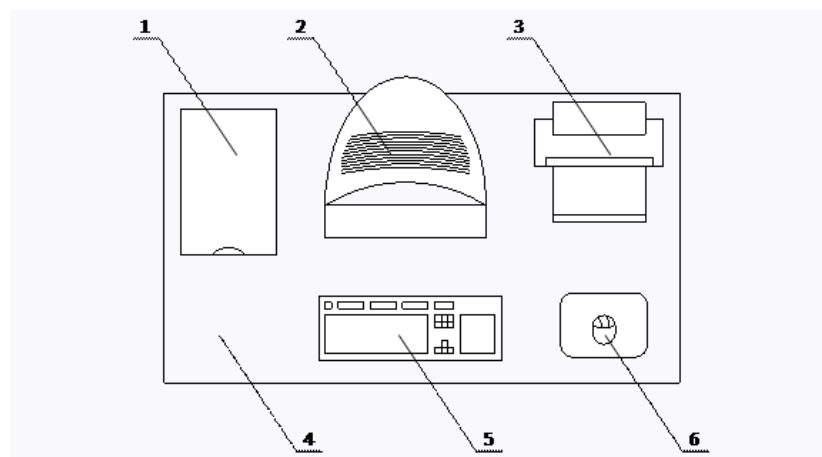


Рис. 3.2 Розміщення основних і периферійних складових ПК

На малюнку 3.2 показаний приклад розміщення основних і периферійних складових ПК на робочому столі програміста.

1 - сканер, 2 - монітор, 3 - принтер, 4 - поверхня робочого столу,  
5 - клавіатура, 6 - маніпулятор типу «миша».

Для комфортної роботи стіл повинен задовольняти наступним умовам:

- висота столу повинна бути вибрана з урахуванням можливості сидіти вільно, в зручній позі, при необхідності спираючись на підлокітники;

- нижня частина столу повинна бути сконструйована так, щоб програміст міг зручно сидіти, ні змушений підбирати ноги;

- поверхня стола повинна мати властивості, що виключають появу відблисків в полі зору програміста;

- конструкція столу повинна передбачати наявність висувних ящиків (не менше 3 для зберігання документації, лістингів, канцелярського приладдя).

- висота робочої поверхні рекомендується в межах 680-760 мм. Висота поверхні, на яку встановлюється клавіатура, повинна бути біля 650 мм.

Велике значення надається характеристикам робочого крісла. Так, рекомендована висота сидіння над рівнем підлоги знаходиться в межах 420-550 мм. Поверхня сидіння м'яка, передній край закруглений, а кут нахилу спинки - регульований.

Необхідно передбачати при проектуванні можливість різного розміщення документів: збоку від відео терміналу, між монітором і клавіатурою і т.п. Крім того, у випадках, коли відео термінал має низьку якість зображення, наприклад помітні мелькання, відстань від очей до екрана роблять більше (близько 700 мм), ніж відстань від ока до документа (300-450 мм). Взагалі при високій якості зображення на відео терміналі відстань від очей користувача до екрану, документа і клавіатури може бути рівним.

Положення екрану визначається:

- відстанню зчитування (0,6 ... 0,7 м);
- кутом зчитування, напрямком погляду на  $20^\circ$  нижче горизонталі

до центру екрану, причому екран перпендикулярний цьому напрямку.

Повинна також передбачатися можливість регулювання екрану:

- по висоті +3 см;
- по нахилу від  $-10^\circ$  до  $+20^\circ$  щодо вертикалі;
- в лівому і правому напрямках.

Велике значення також надається правильній робочій позі користувача. При незручній робочій позі можуть з'явитися болі в м'язах, суглобах і сухожиллях. Вимоги до робочої пози користувача відео термінала наступні:

- голова не повинна бути нахилена більш ніж на  $20^\circ$ ,
- плечі повинні бути розслаблені,
- лікті - під кутом  $80^\circ$  ...  $100^\circ$ ,
- передпліччя і кисті рук - в горизонтальному положенні.

Причина неправильної пози користувачів обумовлена наступними факторами: немає гарної підставки для документів, клавіатура знаходиться дуже високо, а документи - низько, нікуди покласти руки і кисті, недостатній простір для ніг.

З метою подолання зазначених недоліків даються загальні рекомендації: краще пересувна клавіатура; повинні бути передбачені спеціальні пристосування для регулювання висоти столу, клавіатури і екрану, а також підставка для рук.

Істотне значення для продуктивної і якісної роботи на комп'ютері мають розміри знаків, густину їх розміщення, контраст і співвідношення яскравості символів і фону екрану. Якщо відстань від очей оператора до екрану дисплея складає 60 ... 80 см, то висота знаку повинна бути не менше 3 мм, оптимальне співвідношення ширини і висоти знаку складає 3: 4, а відстань між знаками - 15 ... 20% їх висоти. Співвідношення яскравості фону екрану і символів - від 1: 2 до 1:15.

Під час користування комп'ютером медики радять встановлювати монітор на відстані 50-60 см від очей. Фахівці також вважають, що верхня частина дисплея повинна бути на рівні очей або трохи нижче. Коли людина дивиться прямо перед собою, його очі відкриваються ширше, ніж коли він дивиться вниз. За рахунок цього площа огляду значно збільшується, викликаючи обезводнення очей. До того ж якщо екран встановлений високо, а очі широко відкриті, порушується функція моргання. Це означає, що очі не закриваються повністю, не омиваються слізною рідиною, не отримують достатнього зволоження, що приводить до їх швидкої стомлюваності.

Створення сприятливих умов праці і правильне естетичне оформлення робочих місць на виробництві має велике значення як для полегшення праці, так і для підвищення його привабливості, позитивно впливає на продуктивність праці.

## **3.2. Вимоги протипожежної безпеки**

### **3.2.1. Пожежна безпека**

Пожежі завдають величезний матеріальний збиток і в ряді випадків супроводжуються загибеллю людей. Тому захист від пожеж є найважливішим обов'язком кожного члена суспільства і проводиться в загальнодержавному масштабі.

Пожежна безпека може бути забезпечена заходами пожежної профілактики і активного пожежного захисту. Пожежна профілактика включає комплекс заходів, спрямованих на попередження пожежі або зменшення його наслідків.

Пожежа - це горіння поза спеціальним вогнищем, яке не контролюється і може привести до масового ураження і загибелі людей, а також до нанесення екологічного, матеріальної та іншої шкоди.

Горіння - це хімічна реакція окислення, що супроводжується виділенням теплоти і світла. Для виникнення горіння потрібна наявність трьох чинників: горючої речовини, окиснювача і джерела загорання.

Окиснювачами можуть бути кисень, хлор, фтор, бром, йод, окис азоту та інші.

Процес виникнення горіння поділяється на кілька видів.

Спалах - швидке згоряння горючої суміші, що не супроводжується утворенням стислих газів.

Займання - виникнення горіння під впливом джерела запалювання.

Самозаймання - явище різкого збільшення швидкості екзотермічних реакцій, що приводить до виникнення горіння речовини при відсутності джерела запалювання. Розрізняють декілька видів самозаймання:

- Хімічне - від впливу на горючі речовини кисню, повітря, води або взаємодії речовин;

- Мікробіологічне - відбувається при певній вологості і температури в рослинних продуктах (самозаймання зерна);

- Теплове - внаслідок довгострокового впливу незначних джерел тепла.

Вибух - надзвичайно швидке (вибухове) перетворення, що супроводжується виділенням енергії з утворенням стислих газів.

Основними показниками пожежної небезпеки є температура самозаймання та концентраційні межі запалення.

Горючими називаються речовини, здатні самостійно горіти після вилучення джерела загоряння.

За ступенем горючості речовини поділяються на: горючі (згорає), важко горючі (вогнестійкі) і негорючі (вогнетривкі).

До пальних відносяться такі речовини, які при запаленні стороннім джерелом продовжують горіти і після його видалення.

До важко горючих відносяться такі речовини, які не здатні поширювати полум'я і світять лише в місці впливу джерела запалювання.

Негорючими є речовини, незаймисті навіть при дії досить потужних джерел запалювання (імпульсів).

Вода є хорошим вогнегасним засобом, що володіє наступними перевагами: охолоджуючу дію, розбавлення горючої суміші паром (при випаровуванні води її об'єм збільшується в 1700 разів), механічний вплив на полум'я, доступність і низька вартість, хімічна нейтральність.

Гасіння пожеж водою виробляють установками водяного пожежогасіння, пожежними автомашинами і водяними стволами. Для подачі води в ці установки використовують водопроводи.

До установок водяного пожежогасіння відносять спринклерні і дренчерні установки.

Спринклерна установка являє собою розгалужену систему труб, заповнену водою і обладнану спринклерними головками. Вода з системи під тиском виходить з отвору головки і зрошує конструкції приміщення та обладнання.

Дренчерні установки представляють собою систему трубопроводів, на яких розташовані спеціальні головки-дренчери з відкритими вихідними отворами діаметром 8, 10 і 12, 7 мм лопатевого або розеточного типу, розраховані на зрошення до 12 м<sup>2</sup> площі статі.

Дренчерні установки можуть бути ручної і автоматичної дії. Після приведення в дію вода заповнює систему і виливається через отвори в дренчерних голівках.

Пар застосовують в умовах обмеженого повітрообміну, а також у закритих приміщеннях з найбільш небезпечними технологічними процесами. Гасіння пожежі паром здійснюється за рахунок ізоляції поверхні горіння від навколишнього середовища. При гасінні необхідно створити концентрацію пара приблизно 35%.

Піну застосовують для гасіння твердих та рідких речовин, які не вступають у взаємодію з водою. Вогнегасячий ефект при цьому досягається за рахунок ізоляції поверхні горючої речовини від навколишнього повітря. Вогнегасні властивості піни визначаються її кратністю - відношення об'єму

піни до об'єму її рідкої фази, стійкістю, дисперсністю, в'язкістю. Залежно від способу отримання піни ділять на хімічні та повітряно-механічні.

Стаціонарні установки призначені для гасіння пожеж в початковій стадії їх виникнення без участі людини. Поділяються на водяні, пінні, газові, порошкові, парові. Можуть бути автоматичними і ручними з дистанційним управлінням.

Вогнегасники - пристрої для гасіння пожеж вогнегасячою речовиною, яке він випускає після приведення його в дію, використовується для ліквідації невеликих пожеж. В вогнегасниках використовують хімічну або повітряну механічну піну, діоксид вуглецю (рідкому стані), аерозолі та порошки до складу яких входить бром. Поділяються:

по рухливості:

- ручні до 10 літрів
- пересувні
- стаціонарні

по вогнегасячому складу:

- рідинні;
- вуглекислотні;
- хімічно-пінні;
- повітряно-пінні;
- хладонові;
- порошкові (рис. 5.3);
- комбіновані;

Вогнегасники маркуються буквами (вид вогнегасника за розрядом) і цифровий (обсяг).



Рис. 3.3. Порошковий вогнегасник

Ручний пожежний інструмент - це інструмент для розкриття і розібрання конструкцій і проведення аварійно-рятувальних робіт при гасінні пожежі. До них відносяться: гаки, ломи, сокири, відра, лопати, ножиці для різання металу. Інструмент розміщується на видному та доступному місці на стендах і щитах.

До систем сигналізації пред'являються такі технічні вимоги: вони повинні мати мінімальну інерційність спрацювання, забезпечувати задану достовірність інформації, відсутність помилкового спрацювання; бути надійними в роботі при всіх умовах експлуатації, забезпечувати автономне включення сигналу тривоги.

Основними елементами пожежної сигналізації є:

- датчики пожежної сигналізації, які розміщуються в найбільш пожежо і вибухонебезпечних місцях;
- електронно-підсилювальний блок, який забезпечує дистанційний контроль за станом датчиків;
- виконавчий блок, за допомогою якого включається перший рубіж протипожежної системи та блок сигналізації.

Датчики - найбільш важливий елемент системи сигналізації, який в основному визначає можливості і характеристики системи в цілому. Залежно

від фізичної суті, закладеної в основу роботи датчика, системи підрозділяються на: теплові, іонізаційні, радіаційні, тощо. Теплові системи

реагують на підвищення температури або стінок конструкції, або навколишнього середовища, іонізаційні і радіаційні спрацьовують при наявності вогню, принцип їх роботи заснований на тому, що під впливом високої температури іонізуються продукти горіння, а також приблизно 20% всієї енергії - випромінювання.

### **3.2.2. Вимоги електробезпеки**

Є неприпустимими:

- наявність переплутаних кабелів при з'єднання частин ПК;
- експлуатація кабелів та проводів з пошкодженою або такою, що втратила захисні властивості за час експлуатації, ізоляцією;
- залишення під напругою кабелів та проводів з неізольованими провідниками;
- застосування саморобних подовжувачів, які не відповідають вимогам ПУЕ до переносних електропроводок;
- користування пошкодженими розетками, розгалужувальними та з'єднувальними коробками, вимикачами та іншими електровиробами, а також лампами, скло яких має сліди затемнення або випинання.

### **3.3. Санітарно-гігієнічні вимоги**

#### **3.3.1. Забарвлення і коефіцієнти відображення**

Забарвлення приміщень і меблів повинна сприяти створенню сприятливих умов для зорового сприйняття, гарного настрою.

Джерела світла, такі як світильники і вікна, які дають віддзеркалення від поверхні екрану, значно погіршують точність знаків і тягнуть за собою перешкоди фізіологічного характеру, які можуть виразитися в значній напрузі, особливо при тривалій роботі. Відображення, включаючи відображення від вторинних джерел світла, повинне бути зведено до мінімуму. Для захисту від надмірної яскравості вікон можуть бути застосовані штори і екрани.

У приміщеннях, де знаходиться комп'ютер, необхідно забезпечити наступні величини коефіцієнта віддзеркалення: для стелі: 60 ... 70%, для стін: 40 ... 50%, для підлоги: близько 30%. Для інших поверхонь і робочих меблів: 30 ... 40%.

### **3.3.2. Освітлення**

Правильно спроектоване і виконане виробниче освітлення покращує умови зорової роботи, знижує стомлюваність, сприяє підвищенню продуктивності праці, благотворно впливає на виробниче середовище, надаючи позитивну психологічну дію на працюючого, підвищує безпеку праці і знижує травматизм.

Недостатність освітлення призводить до напруги зору, послаблює увагу, приводить до настання передчасної стомленості. Надмірно яскраве освітлення викликає осліплення, роздратування і різь в очах. Неправильний напрямок світла на робочому місці може створювати різкі тіні, відблиски, дезорієнтувати працюючого. Всі ці причини можуть призвести до нещасного випадку або профзахворювань, тому настільки важливий правильний розрахунок освітленості.

Існує три види освітлення - природне, штучне і поєднане (природне і штучне).

Природне освітлення - освітлення приміщень денним світлом, що потрапляє через світлові прорізи в зовнішніх огорожувальних конструкціях приміщень. Природне освітлення характеризується тим, що змінюється в широких межах залежно від часу дня, пори року, характеру області і ряду інших чинників.

Штучне освітлення застосовується при роботі в темний час доби і вдень, коли не вдається забезпечити нормовані значення коефіцієнта природного освітлення (похмура погода, короткий світловий день). Освітлення, при якому недостатнє за нормами природне освітлення доповнюється штучним, називається змішаним освітленням.

Штучне освітлення підрозділяється на робоче, аварійне, евакуаційне, охоронне. Робоче освітлення, у свою чергу, може бути загальним або комбінованим. Загальне - освітлення, при якому світильники розміщуються у верхній зоні приміщення рівномірно або стосовно до розташування обладнання. Комбіноване - освітлення, при якому до загального додається місцеве освітлення.

В приміщенні обчислювальних центрів необхідно застосувати систему комбінованого освітлення.

При виконанні робіт категорії високої зорової точності (найменший розмір об'єкта розрізнення 0,3 ... 0,5 мм) величина коефіцієнта природного освітлення (КПО) повинна бути не нижче 1,5%, а при зоровій роботі середньої точності (найменший розмір об'єкта розрізнення 0,5 ... 1,0 мм) КПО повинен бути не нижче 1,0%. Як джерела штучного освітлення звичайно використовуються люмінесцентні лампи типу ЛЛ, або ДРЛ, які попарно об'єднуються в світильники, які повинні розташовуватися рівномірно над робочими поверхнями .

Вимоги до освітленості в приміщеннях, де встановлені комп'ютери, наступні: при виконанні зорових робіт високої точності загальна освітленість повинна складати 300 лк, а комбінована - 750 лк; аналогічні вимоги при виконанні робіт середньої точності - 200 і 300 лк відповідно.

Крім того все поле зору повинне бути освітлено достатньо рівномірно - ця основна гігієнічна вимога. Іншими словами, ступінь освітлення приміщення і яскравість екрану комп'ютера повинні бути приблизно однаковими, тому що яскраве світло в районі периферійного зору значно збільшує напруженість очей і, як наслідок, призводить до їх швидкої стомлюваності.

### **3.3.3. Параметри мікроклімату**

Параметри мікроклімату можуть змінюватися в широких межах, в той час як необхідною умовою життєдіяльності людини є підтримка

постійності температури тіла завдяки терморегуляції, тобто здатності організму регулювати віддачу тепла в навколишнє середовище. Принцип нормування мікроклімату - створення оптимальних умов для теплообміну тіла людини з навколишнім середовищем.

Обчислювальна техніка є джерелом істотних тепловиділень, що може привести до підвищення температури і зниження відносної вологості в приміщенні. У приміщеннях, де встановлені комп'ютери, повинні дотримуватися певні параметри мікроклімату. У санітарних нормах встановлені величини параметрів мікроклімату, що створюють комфортні умови. Ці норми встановлюються в залежності від пори року, характеру трудового процесу і характеру виробничого приміщення .

Для забезпечення комфортних умов використовуються як організаційні методи (раціональна організація проведення робіт залежно від пори року і доби, чергування праці і відпочинку), так і технічні засоби (вентиляція, кондиціонування повітря, опалювальна система).

### 3.3.4. Шум і вібрація

Шум погіршує умови праці здійснюючи шкідливу дію на організм людини. Працюючі в умовах тривалої шумової дії випробовують дратівливість, головні болі, запаморочення, зниження пам'яті, підвищену стомлюваність, зниження апетиту, біль у вухах і т. Д. Такі порушення в роботі ряду органів і систем організму людини можуть викликати негативні зміни в емоційному стані людини аж до стресових. Під впливом шуму знижується концентрація уваги, порушуються фізіологічні функції, з'являється втома у зв'язку з підвищеними енергетичними витратами і нервово-психічним напруженням, погіршується мовна комутація. Все це знижує працездатність людини і його продуктивність, якість і безпеку праці. Тривала дія інтенсивного шуму вище 80 дБА на слух людини приводить до його часткової або повної втрати.

Рівень шуму на робочому місці математиків-програмістів і операторів відеоматеріалів не повинен перевищувати 50 дБА, а в залах обробки інформації на обчислювальних машинах - 65 дБА. Для зниження рівня шуму стіни і стеля приміщень, де встановлені комп'ютери, можуть бути облицьовані звукопоглинальними матеріалами. Рівень вібрації в приміщеннях обчислювальних центрів може бути понижений шляхом встановлення устаткування на спеціальні віброізолятори.

Максимальний рівень рентгенівського випромінювання на робочому місці оператора комп'ютера звичайно не перевищує 10 мкбер/ч, а інтенсивність ультрафіолетового і інфрачервоного випромінювань від екрану монітора лежить в межах 10 ... 100 МВт / м<sup>2</sup>.

Для зниження впливу цих видів випромінювання рекомендується застосовувати монітори із зниженим рівнем випромінювання (MPR-II, TCO-92, TCO-99), встановлювати захисні екрани, а також дотримуватися регламентовані режими праці та відпочинку.

### 3.4. Аналіз ризиків

Трудова діяльність людини, що здійснюється в виробничих умовах, є однією з основних форм діяльності. Беручи участь у виробничому процесі, працівники піддаються переважно техногенним факторам, явищ і процесів, тобто небезпекам, безпосередньо пов'язаним з природою технологічних процесів, обладнання, технічних пристроїв.

За складом і властивостями виробничі небезпечні і шкідливі фактори підрозділяються на фізичні, хімічні, біологічні та психофізіологічні.

Фізичні небезпечні і шкідливі фактори - це рухомі машини і механізми; незахищені рухливі елементи виробничого обладнання; пересуваються вироби; підвищена напруга в електричних мережах; підвищений рівень статичної електрики; підвищений рівень електромагнітного, рентгенівського, лазерного та ультрафіолетового випромінювання; підвищений рівень вібрації і шуму; недостатнє освітлення; несприятливі метеорологічні умови та ін.

Хімічні небезпечні та шкідливі фактори являють собою різні хімічні елементи та їх сполуки, що володіють загально токсичними, дратівливими, мутагенними і канцерогенними властивостями. Найбільшу канцерогенність мають миш'як, нікель, кадмій, хлорфенол, три- і тетрахлоретилен, вінілхлорид, бензапірен і інші смолисті летючі речовини.

До біологічних небезпечних і шкідливих факторів відносяться мікро- і макроорганізми (бактерії, віруси, гриби, рослини, тварини). Їх вплив на людину може призводити до травм і інфекційних захворювань.

Психофізіологічні чинники включають в себе фізичні перевантаження (Статичні, динамічні, гіподинамічні), а також нервово-психічні перевантаження, до яких відносяться розумове перенапруження, монотонність праці, емоційні перевантаження, перенапруження органів чуття і ін. Виникнення нервово-психічних перевантажень обумовлено зміною характеру сучасного виробництва, підвищенням складності трудової діяльності і істотною зміною функції і ролі людини (людського

фактора) в процесі праці, особливостями якого стає інтенсифікація психічної діяльності людини.

### **3.5. Пропозиції**

Робочі місця повинні розміщуватися на відстані не менше 1 м від стіни з вікнами, так, щоб природне освітлення падало збоку, бажано зліва.

Конструкція робочого місця повинна забезпечувати підтримання оптимальної робочої пози (при роботі сидячи): ступні ніг - на підлозі або підставці для ніг; стегна - в горизонтальній площині; передпліччя - вертикально; лікті - під кутом 70-90° до вертикальної площини; зап'ястя зігнуті під кутом не більше 200° відносно горизонтальної площини; нахил голови - 15-200° відносно вертикальної площини.

Конструкція крісла користувача повинна створювати умови для підтримки корпусу в фізіологічно раціональному положенні зі збереженням природних вигинів хребта, а також повинна забезпечувати зниження статичного напруження м'язів шийно-плечової області спини і не повинна ускладнювати робочих рухів.

Крісло користувача має включати такі основні елементи: сидіння, спинку, підлокітники, а також додатковий елемент - підставку для ніг.

Робоча сидіння користувача має бути підйомно-поворотним: таким, що регулюється по висоті, куту нахилу сидіння і спинки, за відстанню спинки до переднього краю сидіння, висотою підлокітників.

Покриття сидіння, спинки, підлокітників має бути виготовлено з вологовідштовхувальним, не електрешуючим, повітряопроникного, м'якого та нековзкого матеріалу.

Найкращим рішенням для освітлення стануть світлодіодні (рис.5.4) і галогенні лампи: вони містять у собі практично все, що потрібно для комфортної роботи.



Рис. 3.4. Світлодіодна лампа

Дисплей повинен бути обладнаний поворотним майданчиком, що дозволяє переміщати його в горизонтальній і вертикальній площинах в межах  $+300$  або  $-300$  (вправо-вліво), змінювати кут нахилу екрану до  $10-150$ .

Монітор повинен бути типу РК або плазмовим. РК і плазмові монітори створюють менше навантаження на зір, ніж ЕПТ-монітори, багато в чому завдяки відсутності мерехтіння і відблисків, а також завдяки тому, що не дають електромагнітного випромінювання.

## **Висновок**

На ринку програмного забезпечення існує велика кількість подібних програмних продуктів, які так само допомагають вирішувати поставлені проблеми. Однак подібні програми часто вимагають від користувача максимум знань ПК, мають зовелику ціну, не мають необхідного функціоналу, не мають відповідного захисту даних і споживають велику кількість ресурсів комп'ютера, тому власне і було вирішено написати дану програму.

Даний програмний продукт дає можливість, маючи мінімальні знання для роботи на комп'ютері вільно користуватися програмою і проводити будь-які підстроювання.

Даний програмний продукт може бути використаний як в домашніх умовах, так і в умовах виробництва, а також може застосовуватися в ролі розважальної програми.

Шляхи подальшого вдосконалення системи:

- Створення системи адміністрування.
- Додавання медіа – контенту (смайлики, новини, соціальні мережі тощо).
- Можливість інтеграції з різними системами контролю версій.
- Інтеграція з наявними системами на підприємстві.
- Створення детальної технічної документації та керівництва користувача.

## Список використаних джерел

1. [2]. ДСанПіН 3.3.2.007-98 «Державні санітарні правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин» - 12ст.
2. [3]. НПАОП 0.00-1.28-10 " Правила охорони праці під час експлуатації електронно-обчислювальних машин" – 8 ст.
3. Маклаков С.В. ВРwin и ERwin. CASE-средства разработки информационных систем. - М.: ДИАЛОГ-МИФИ, 1999 — 256 с.
4. М'якшило О. М. Моделювання баз даних засобами CASE-технології ERWin: Конспект лекцій з дисципліни «Структурне моделювання систем» для студ. спец. 6.080400 напряму «Комп'ютерні науки» всіх форм навчання. – К.: НУХТ, 2008. – 60с.
5. Бідюк П.І., Коршевніук Л.О. Проектування комп'ютерних інформаційних систем підтримки прийняття рішень: Навчальний посібник. — Київ: ННК «ІПСУ» НТУУ «КПІ», 2010. —340с.
6. Іцік Б. Г. Microsoft SQL Server 2008. Основы T-SQL / Б. Г. Іцік., 2009. – 430 с. – (БХВ-Петербург).
7. Маккі А. Введение в NET 4.0 и Visual Studio 2010 для профессионалов / Алекс Макки., 2012. – 416 с.
8. Ріхтер Д. CLR via C#. Программирование на платформе Microsoft .NET Framework 4.5 на языке C# / Д. Рихтер., 2017. – 896 с. – (4-е изд.).
9. Хейлсберг А. Язык программирования C#. Классика Computers Science. / А. Хейлсберг, М. Торгерсен, С. Вилтамут., 2011. – 784 с. – ("Питер").
10. Методичні вказівки до виконання кваліфікаційної бакалаврської роботи для студентів за напрямом підготовки 6.050101 «Комп'ютерні науки» денної та заочної форм навчання / Уклад.: В.В. Самсонов, Л.Ю. Маноха, Т.М. Горлова, Л.Г. Загоровська, О.М. М'якшило, О.А Хлобистова. – К.: НУХТ, 2011. – 15с.

## ДОДАТОК А. Функціональна модель as-is

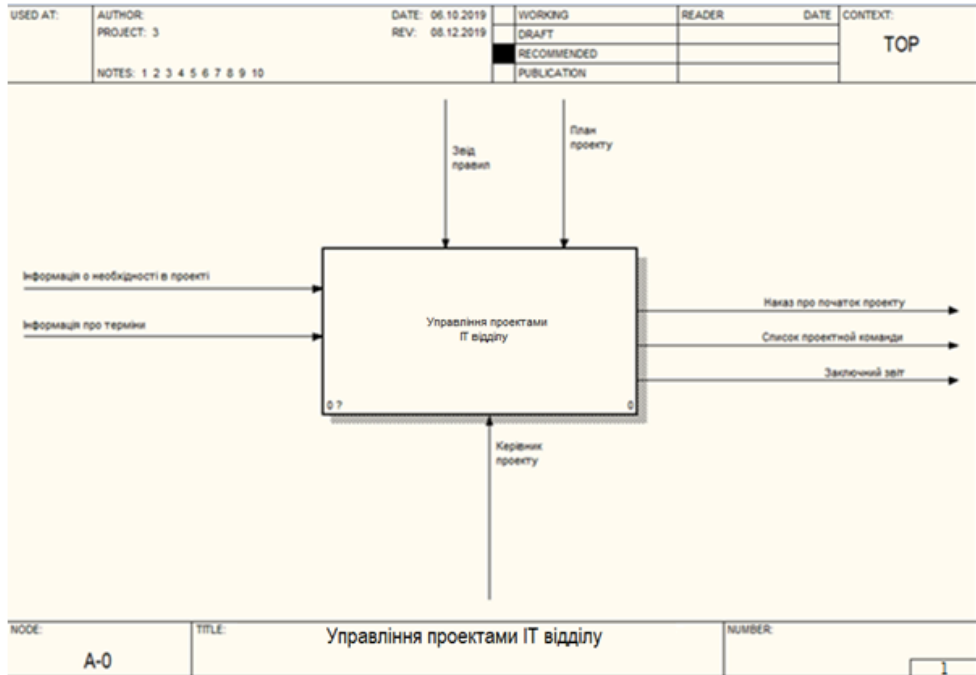


Рисунок А.1 – Функціональна модель AS-IS

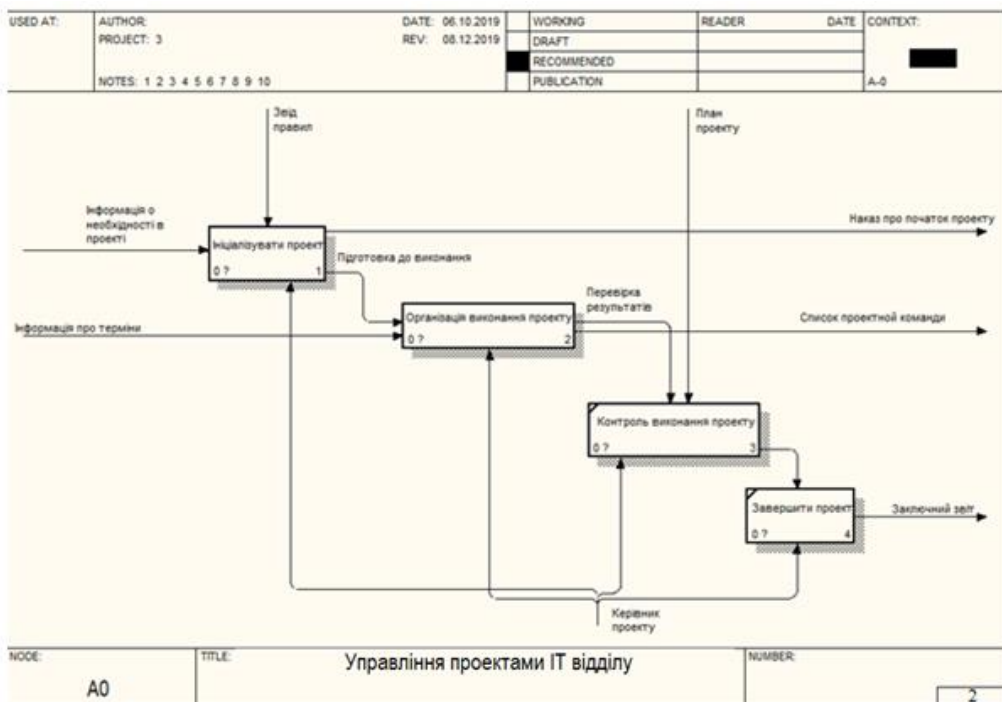


Рисунок А.2 – Діаграма А0 моделі AS-IS

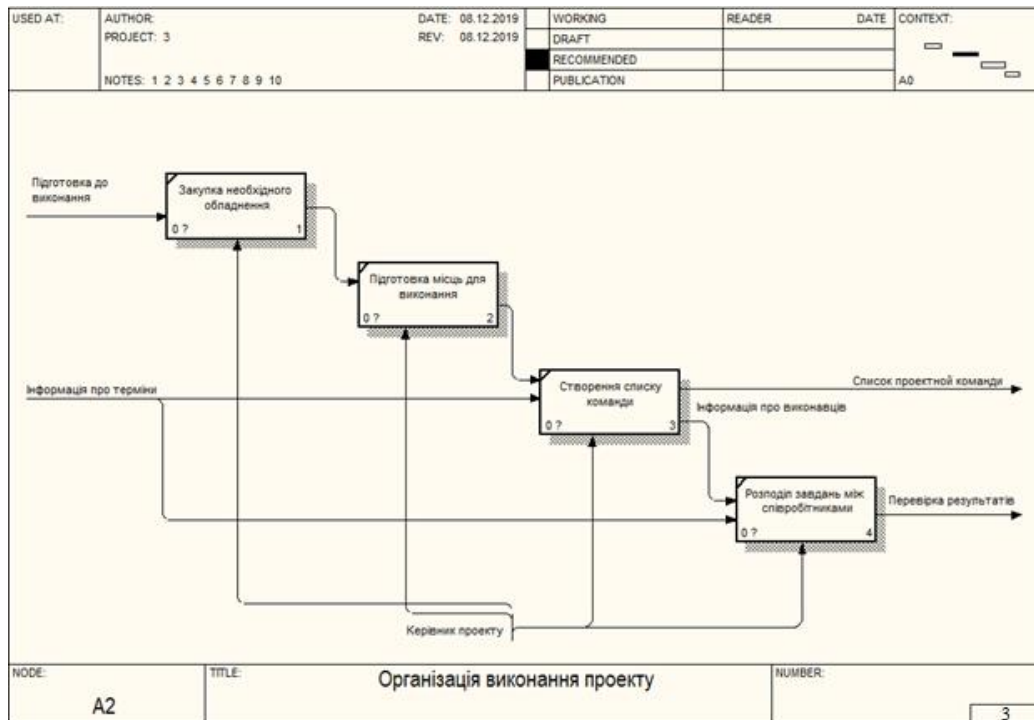


Рисунок А.3 – Деталізація блоку “Організація проекту”

Функціональна модель to-be

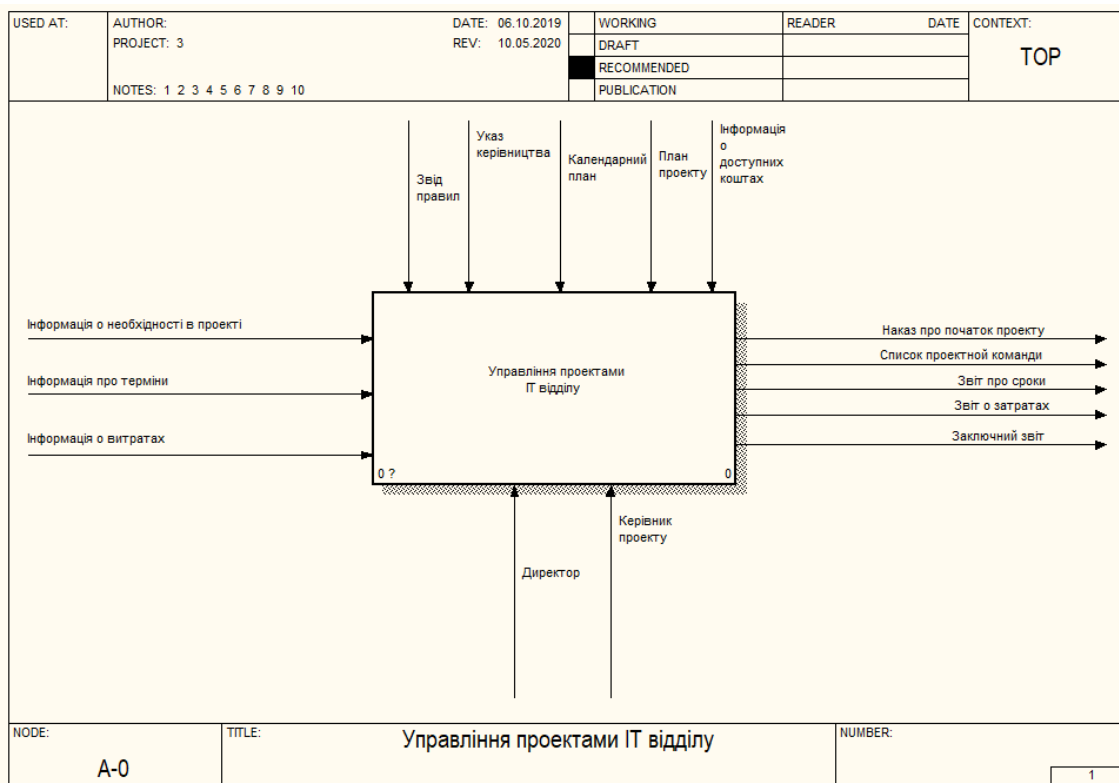


Рисунок А.4 – Функціональна модель TO-BE

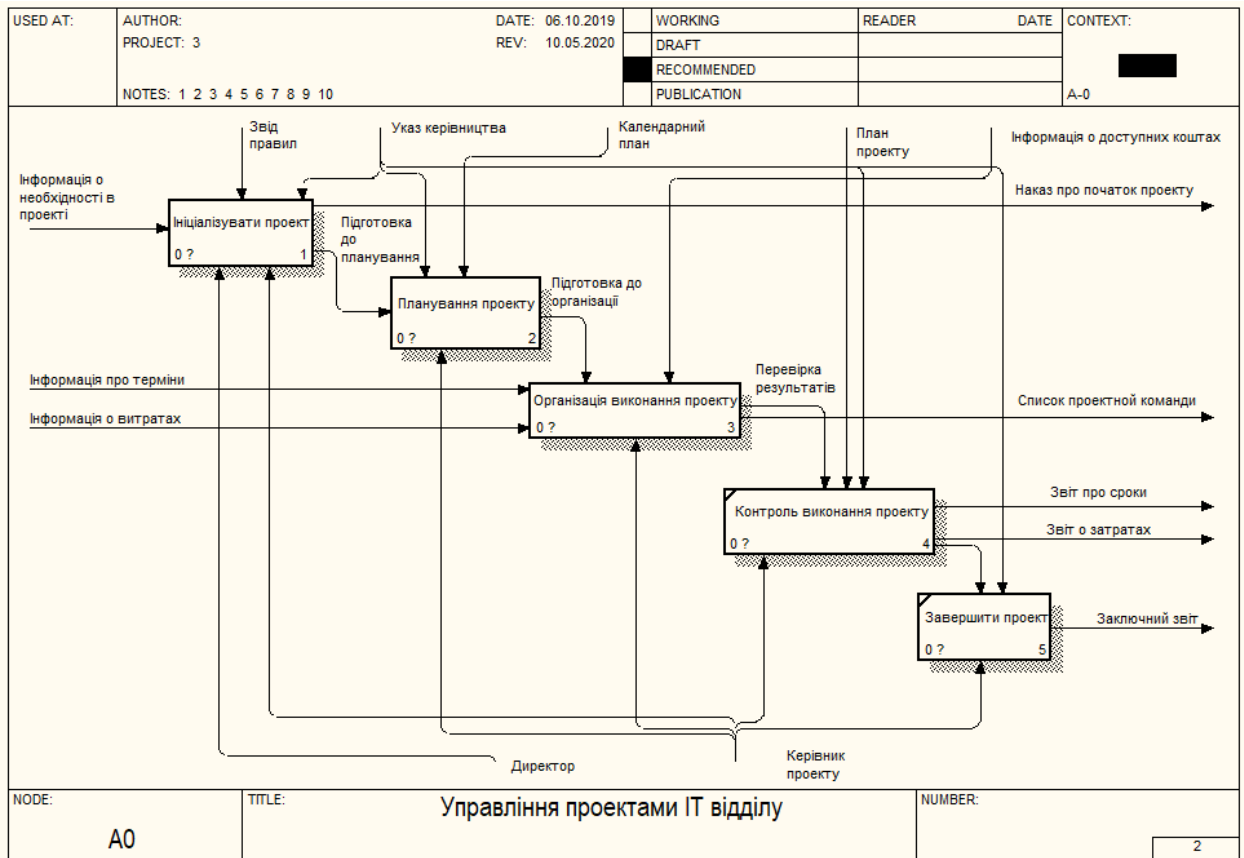


Рисунок А.5 – Декомпозиція діаграми А1 моделі TO-BE

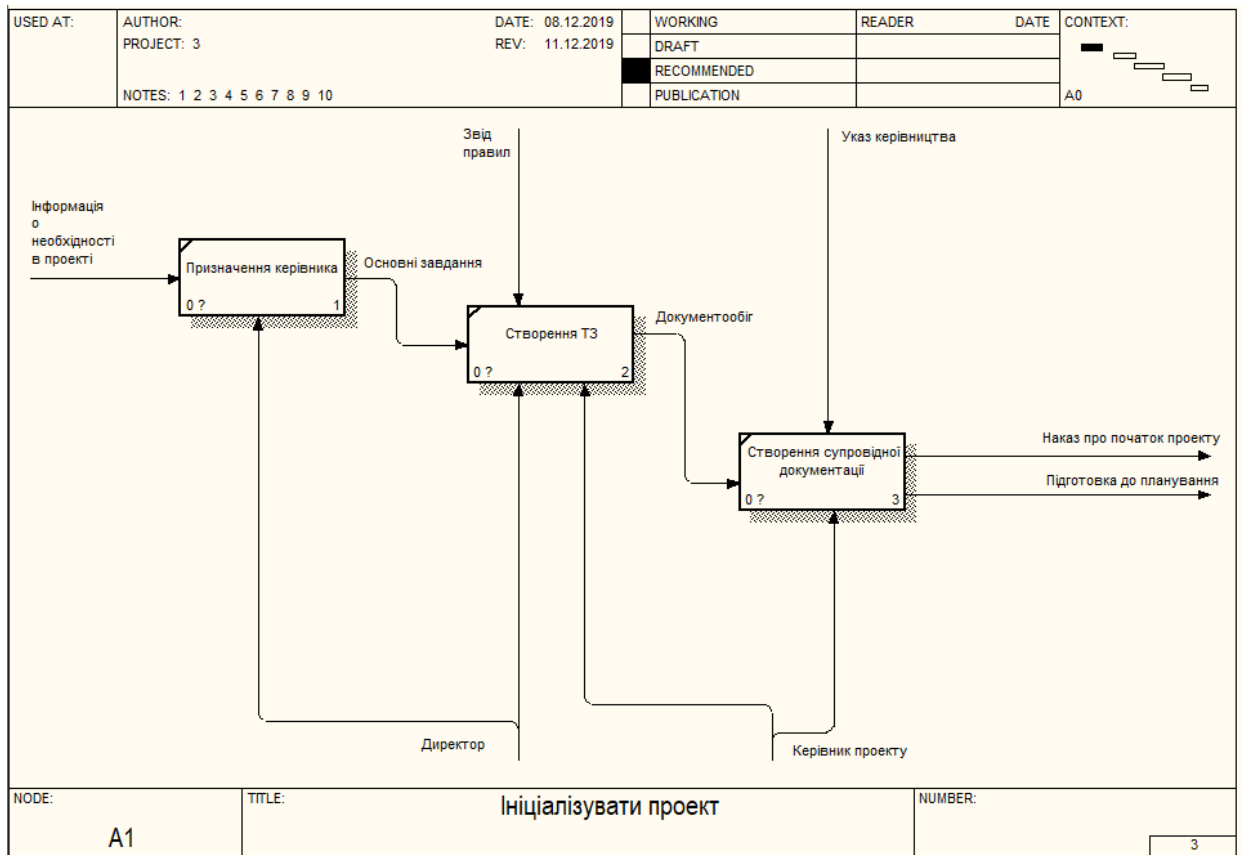


Рисунок А.6 – Деталізація блоку “Призначення керівника”

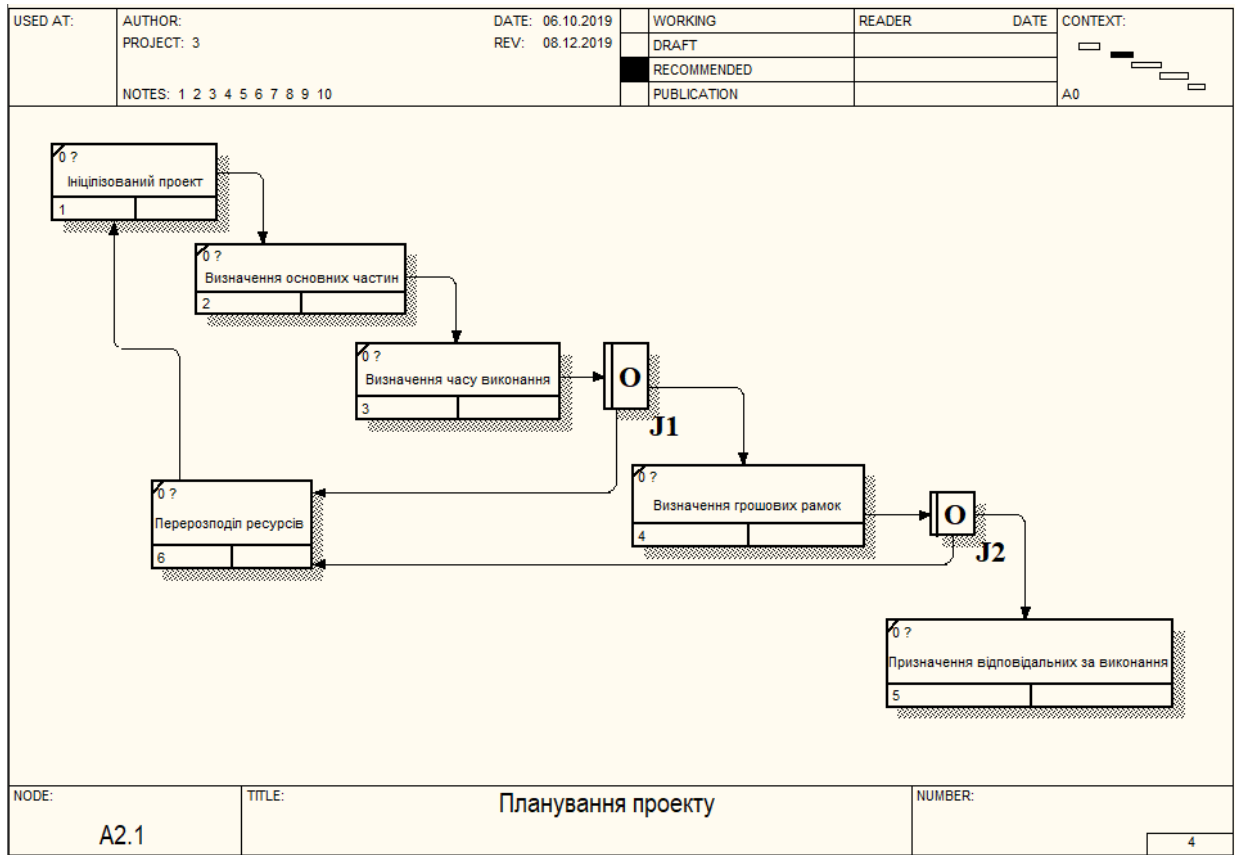


Рисунок А.7 – Деталізація блоку Планування проекту моделі TO-VE в нотації IDEF3

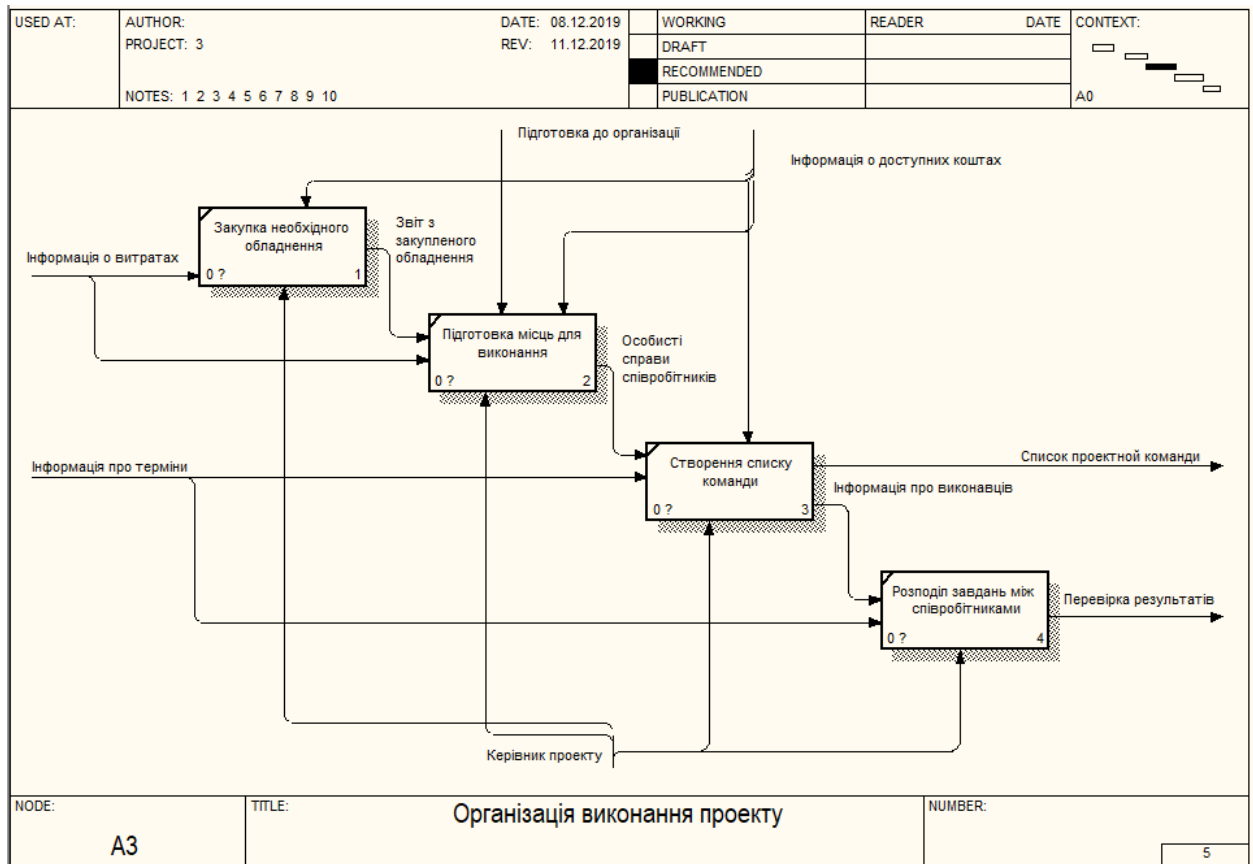


Рисунок А.8 – Деталізація блоку “Організація виконання проекту”

## ДОДАТОК Б. Функціональна модель чату

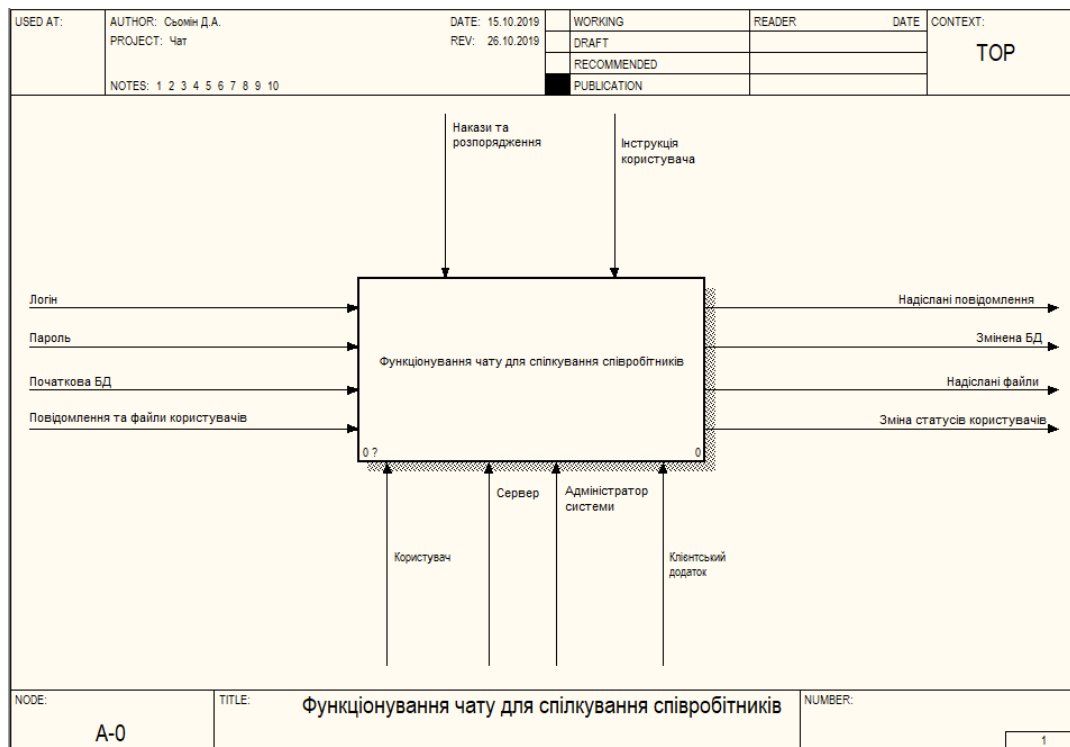


Рисунок Б.1 – Функціональна модель

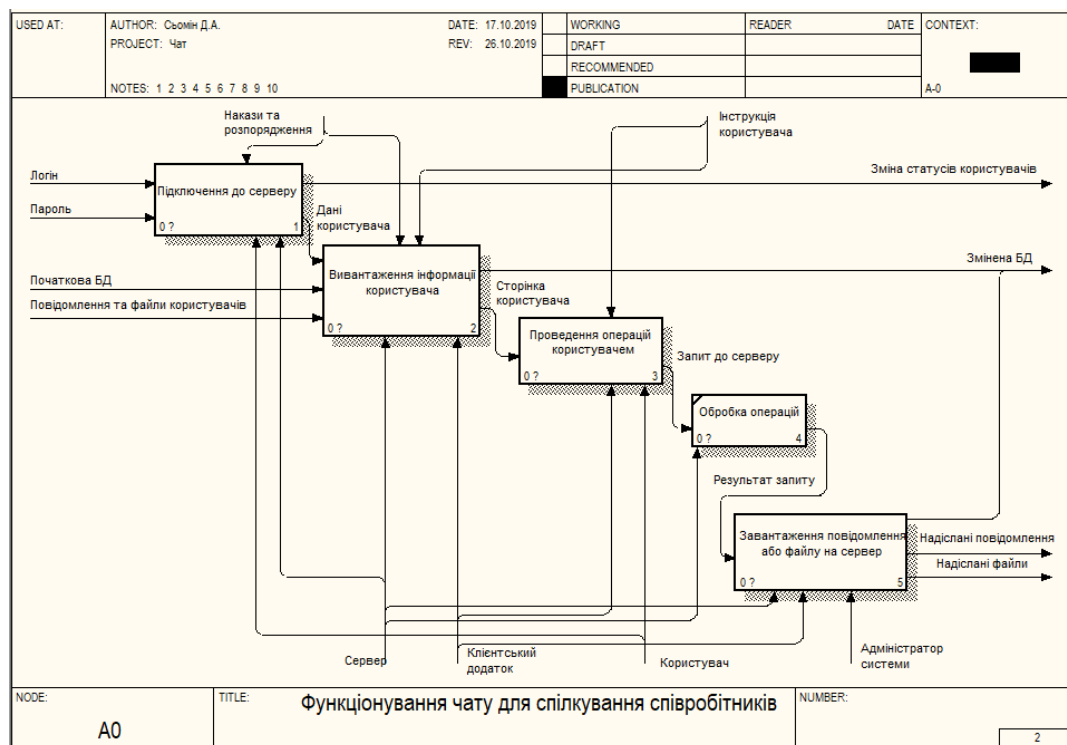


Рисунок Б.2 – Декомпозиція діаграми Б1

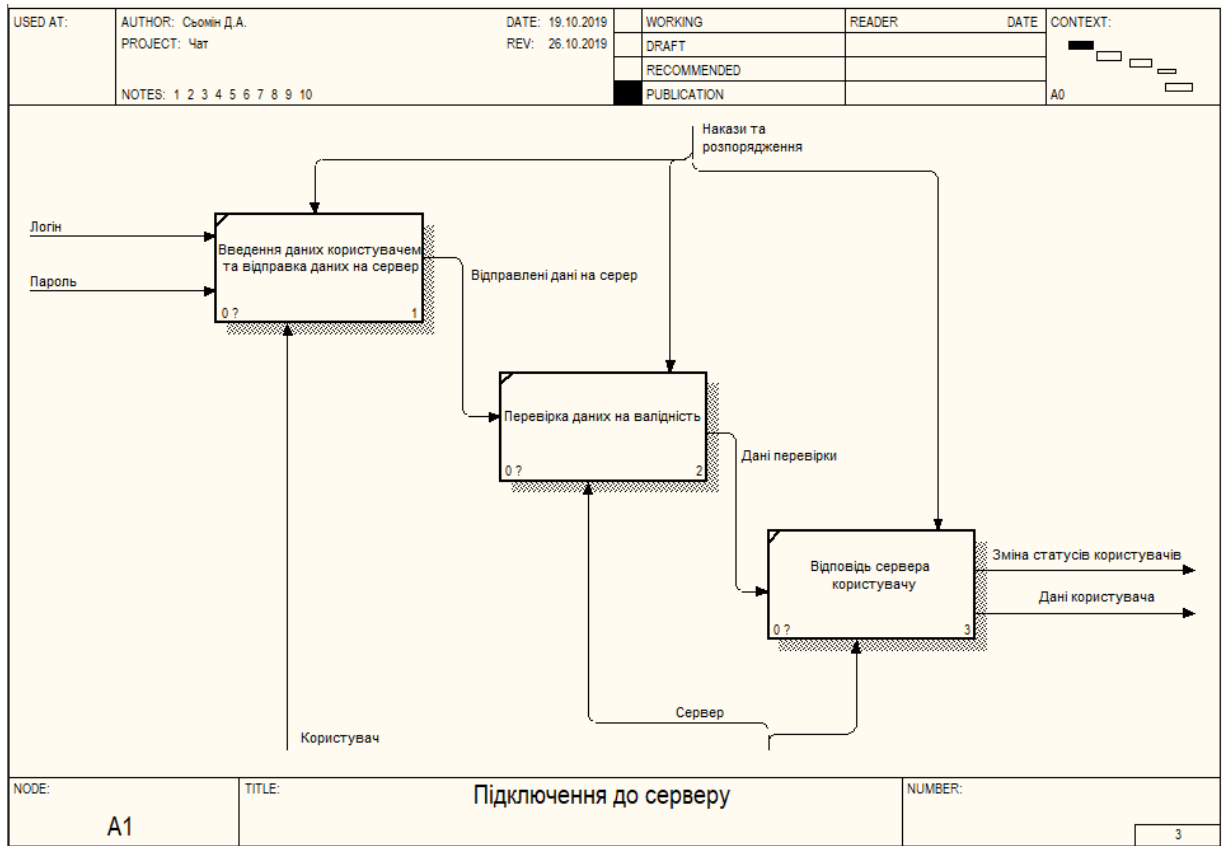


Рисунок Б.3 – Деталізація блоку “Підключення до серверу”

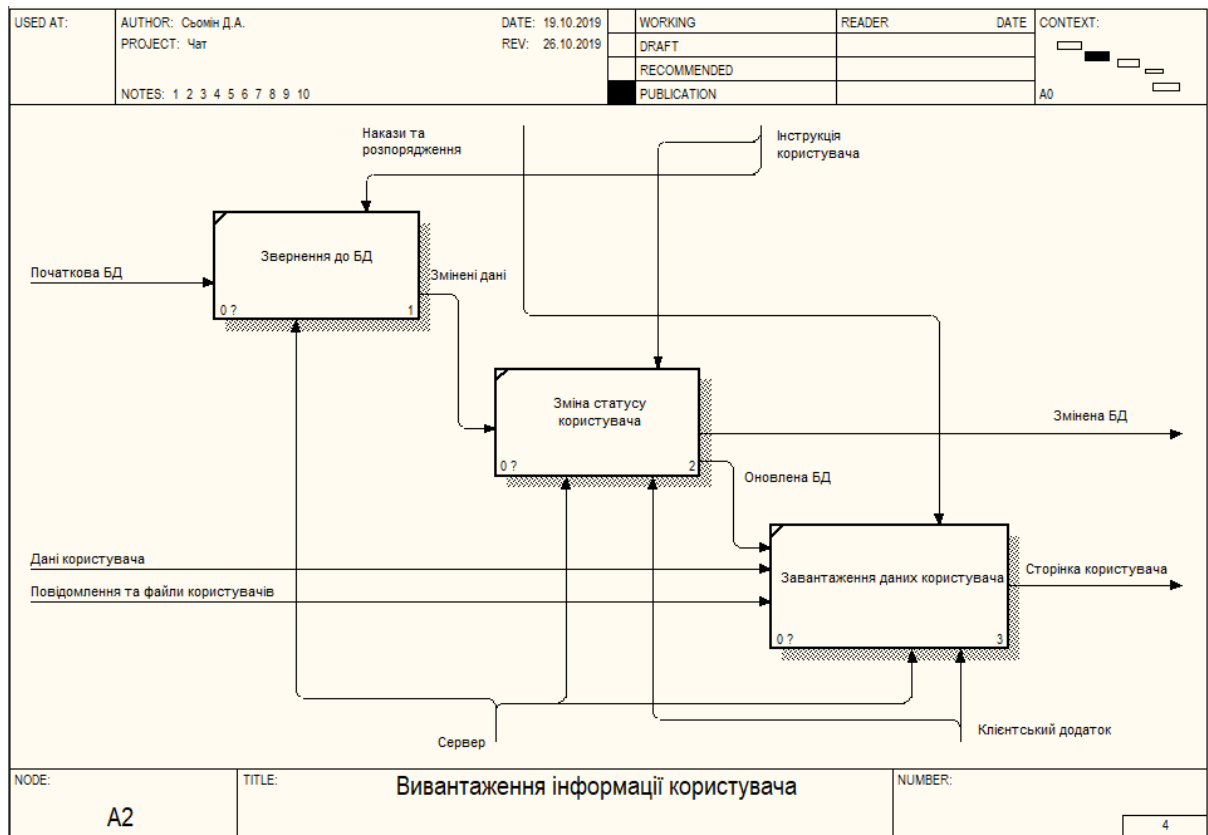


Рисунок Б.4 – Деталізація блоку “Вивантаження інформації користувача”

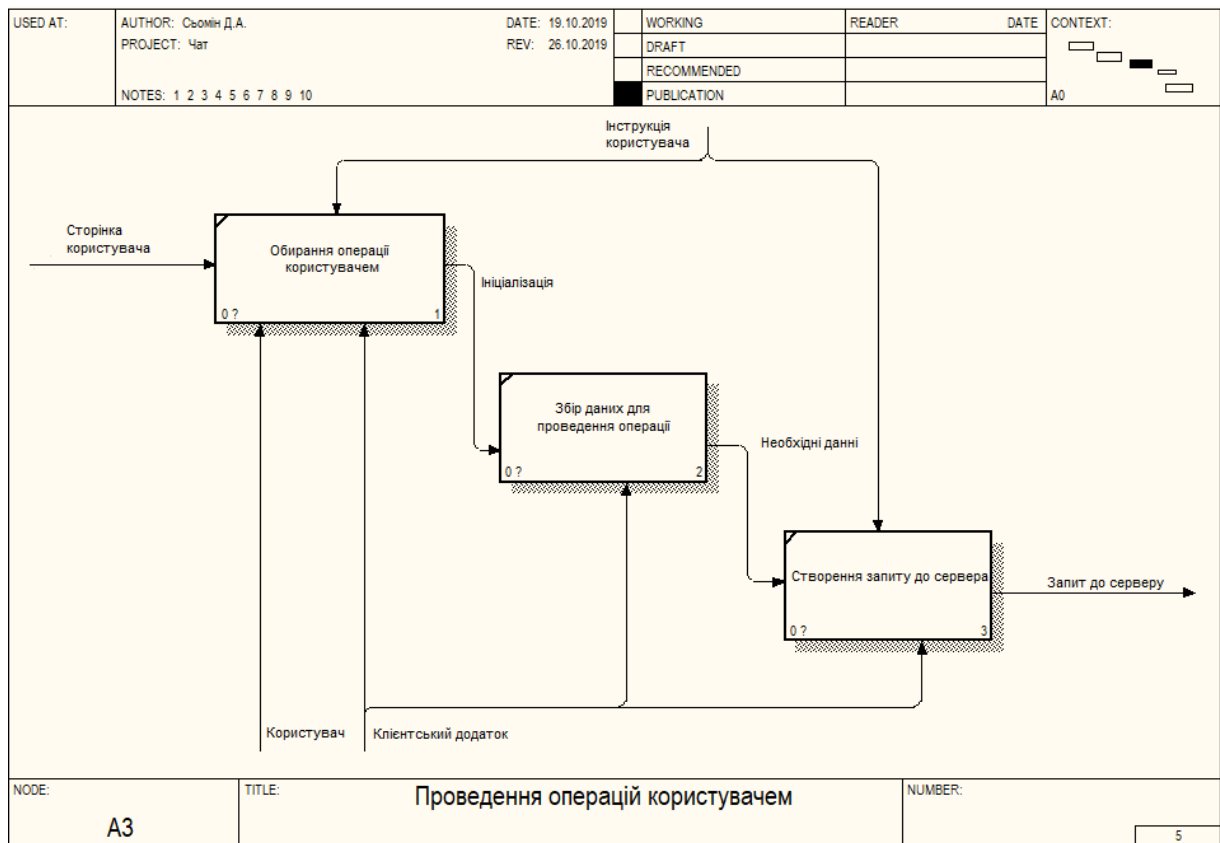


Рисунок Б.5 – Деталізація блоку “Проведення операцій користувачем”

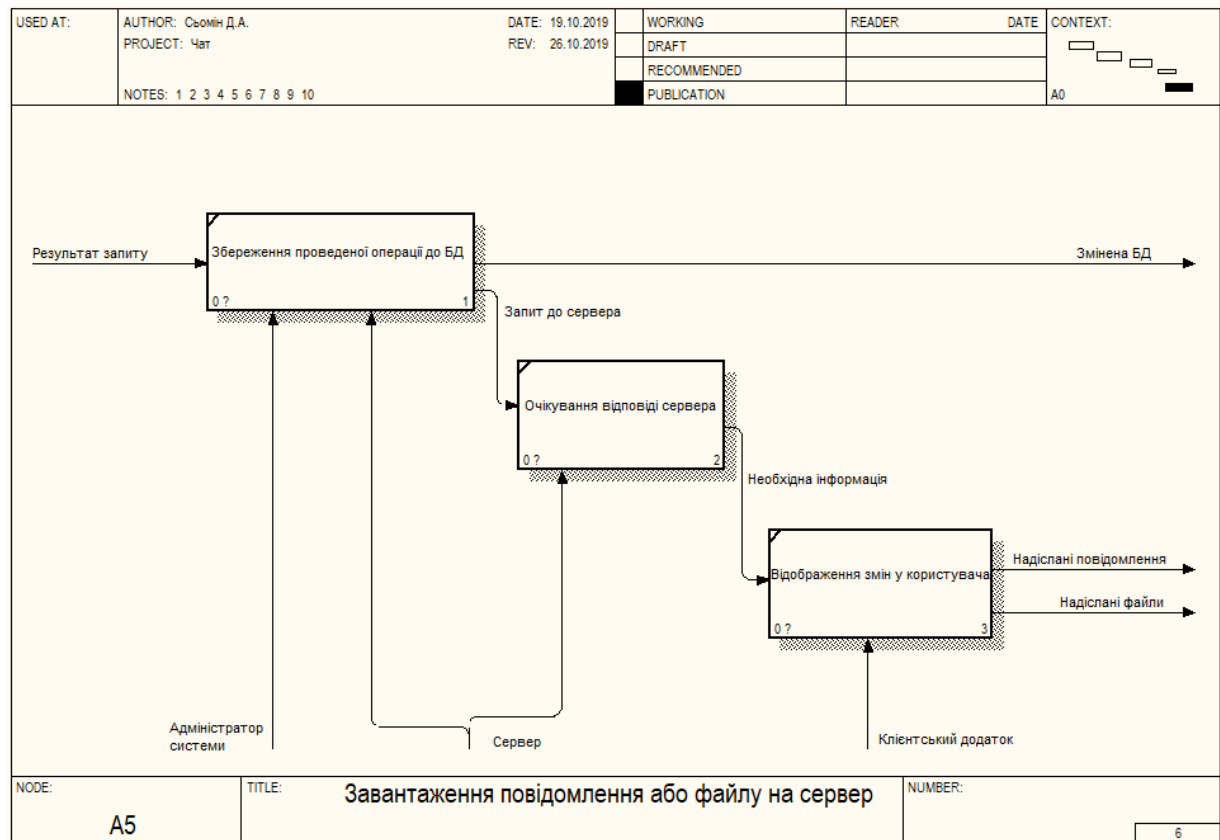


Рисунок Б.6 – Деталізація блоку “Завантаження повідомлення або файлу на сервер”

## ДОДАТОК С. Моделі БД

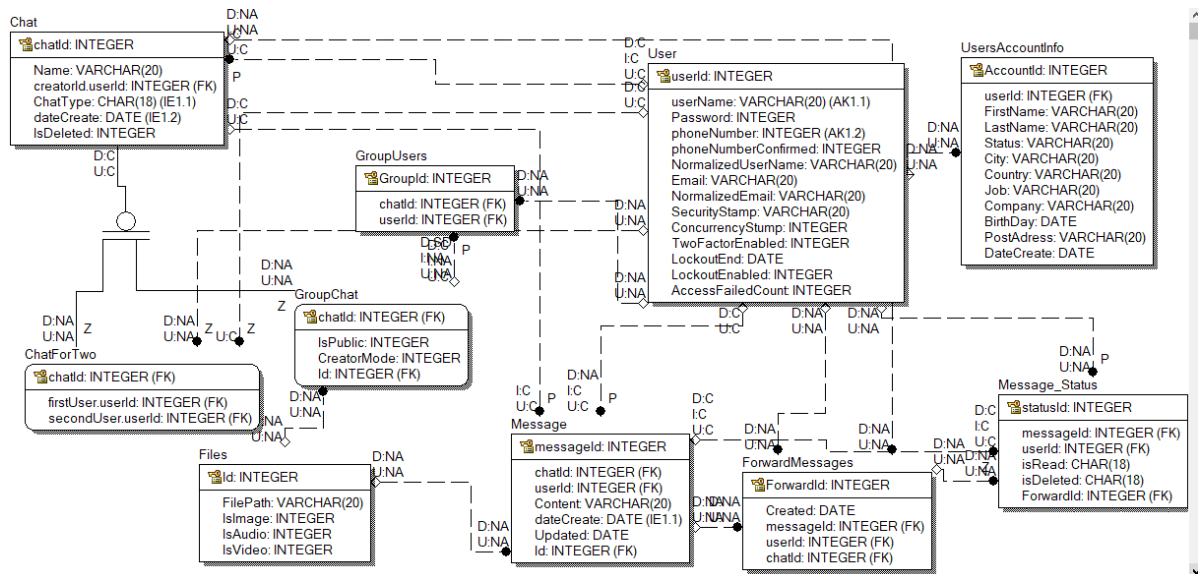


Рисунок С.1 – Логічна модель

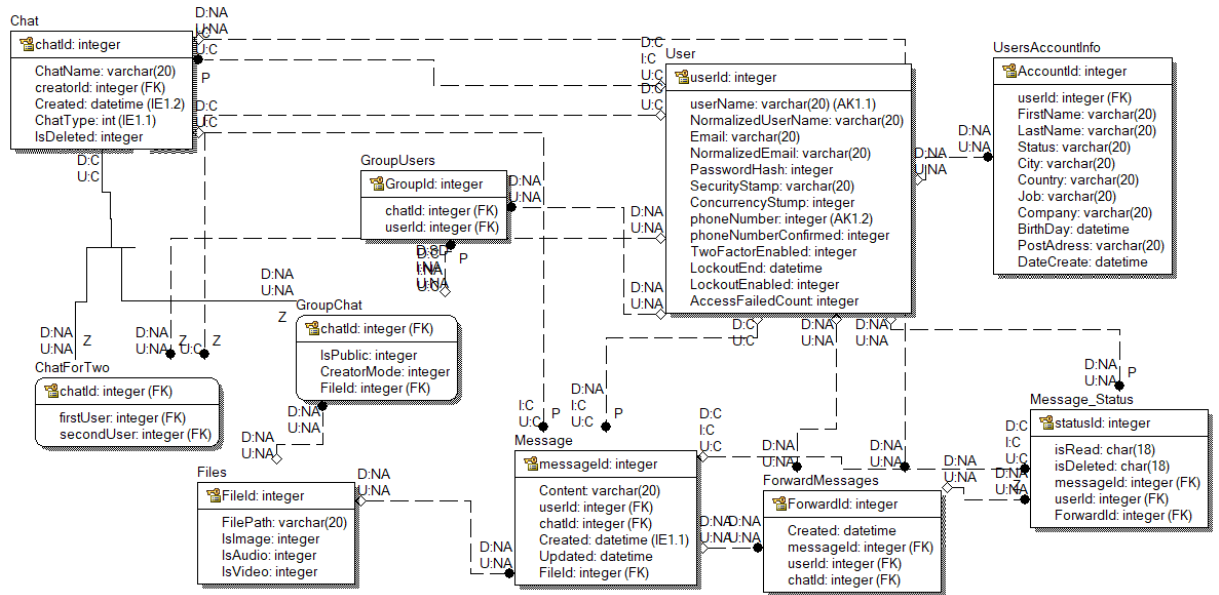


Рисунок С.2 – Фізична модель

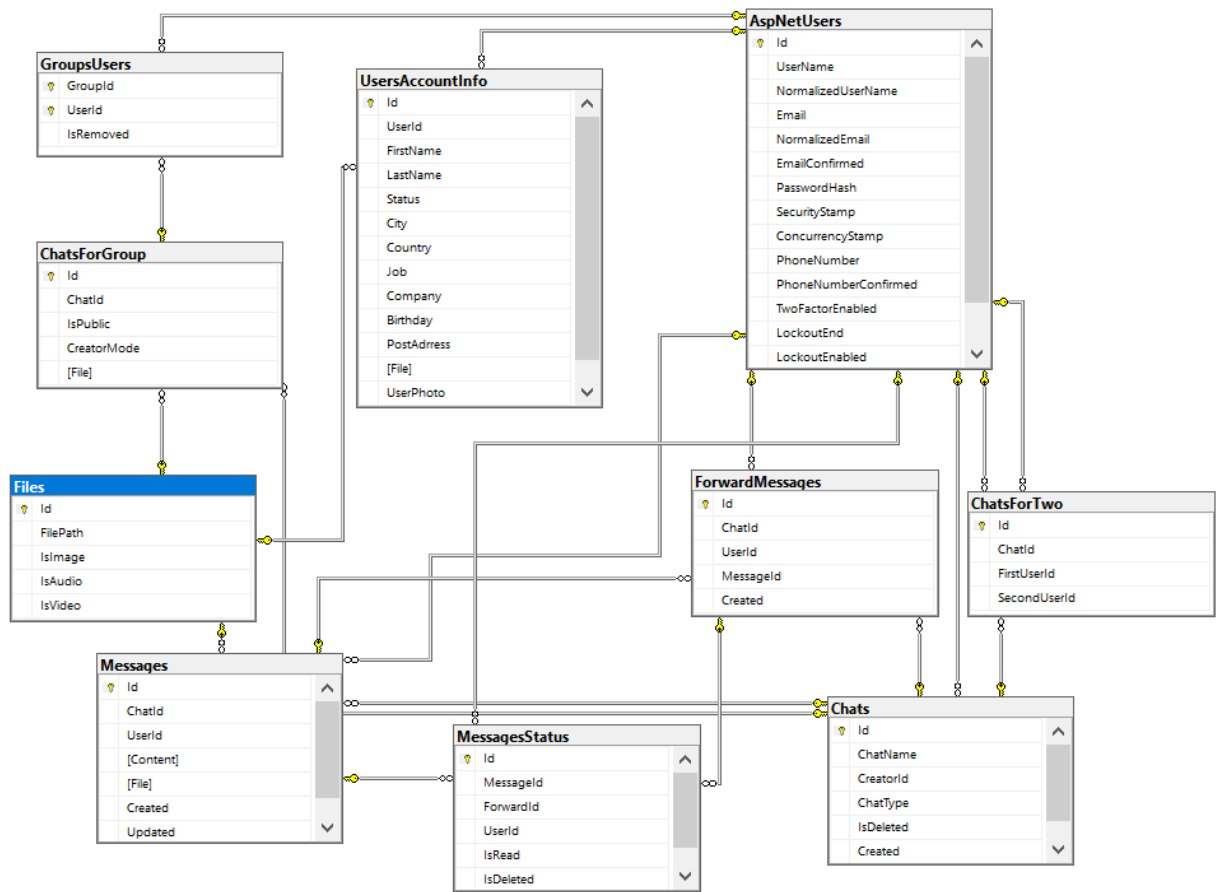


Рисунок С.3 – Схема БД