

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

До захисту допускається

Завідувач кафедри

_____ Скарга-Бандурова І.С.

«_____» _____ 20 р.

ДИПЛОМНИЙ ПРОЕКТ (РОБОТА) БАКАЛАВРА

ПОЯСНЮВАЛЬНА ЗАПИСКА

НА ТЕМУ:

Комплексна тема. Система технічної підтримки клієнтів за допомогою
соціальних мереж: БД і серверна частина

Освітньо-кваліфікаційний рівень “бакалавр”

Напрямок підготовки 6.050102 – “комп’ютерна інженерія”

Керівник проекту:

(підпис)

Скарга-Бандурова І.С.

(ініціали, прізвище)

Консультант з охорони праці:

(підпис)

Критська Я.О.

(ініціали, прізвище)

Студент:

(підпис)

Лунякін Р.О.

(ініціали, прізвище)

Група:

КІ-15Д

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
СХІДНОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМЕНІ ВОЛОДИМИРА ДАЛЯ

Факультет Інформаційних технологій та електроніки
Кафедра Комп'ютерних наук та інженерії
Освітньо-кваліфікаційний
рівень бакалавр
Напрямок підготовки 6.050102 – “комп'ютерна інженерія”
(шифр і назва)
Спеціальність _____
(шифр і назва)

ЗАТВЕРДЖУЮ:

Завідувач кафедри _____
_____ І.С. Скарга-Бандурова
« _____ » _____ 20____ р.

**З А В Д А Н Н Я
НА ДИПЛОМНИЙ ПРОЕКТ (РОБОТУ) БАКАЛАВРА**

Лунякіну Руслану Олександровичу

(прізвище, ім'я, по батькові)

1. Тема роботи Комплексна тема. Система технічної підтримки клієнтів за допомогою соціальних мереж: БД і серверна частина

керівник проекту
(роботи)

Скарга-Бандурова І.С., д.т.н., проф.

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

затверджені наказом вищого навчального закладу від "15 " 05 2019 р. № _____

2. Термін подання студентом роботи 18.06.2019

3. Вихідні дані до
роботи матеріали переддипломної практики

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) Аналіз систем технічної підтримки та формулювання технічного завдання; проектування системи; розробка БД; охорона праці та безпека в надзвичайних ситуаціях

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

6. Консультанти розділів проекту (роботи)

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Охорона праці та безпека в надзвичайних ситуаціях	ст. викл. Критська Я.О.		

7. Дата видачі завдання 15.05.2019

Керівник

Завдання прийняв до виконання

_____ (підпис)

_____ (підпис)

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проекту (роботи)	Строк виконання етапів проекту (роботи)	Примітка
1	Ознайомлення з предметною галуззю	03.05 – 15.05	
2	Аналіз задачі та постановка проблеми	16.05 – 18.05	
3	Розгляд існуючих рішень щодо вирішення задачі	19.05 – 25.05	
4	Розробка бази даних	26.05 – 03.06	
5	Розробка програмного коду і тестування	04.06 – 10.06	
6	Розробка розділу «Охорона праці та безпека в надзвичайних ситуаціях»	04.06 – 08.06	
7	Оформлення пояснювальної записки	09.06 – 11.06	

Студент

_____ (підпис)

Лунякін Р.О.

_____ (прізвище та ініціали)

Керівник

_____ (підпис)

Скарга-Бандурова І.С.

_____ (прізвище та ініціали)

РЕФЕРАТ

Дипломна робота бакалавра складається з вступу, чотирьох розділів, висновків, списку використаних джерел та додатків.

Об'єкт дослідження: побудова бази даних та серверної частини системи для автоматизації системи технічної підтримки через соціальні мережі.

Мета: розробка системи для збору повідомлень з соціальних мереж і месенджерів та автоматизованої передачі даних для подальшої обробки операторами підтримки.

В процесі розробки коротко розглядаються та порівнюються існуючі рішення, описується архітектура та взаємодія із системою.

Робота містить 74 сторінки, 7 рисунків, 2 додатки, 27 джерел.

Ключові слова: веб-додаток, технічна підтримка, соціальна мережа, месенджер, чат-бот.

ЗМІСТ

ВСТУП	7
Перелік умовних позначок та скорочень.....	9
1 АНАЛІЗ СИСТЕМ ТЕХНІЧНОЇ ПІДТРИМКИ ТА ФОРМУЛЮВАННЯ ТЕХНІЧНОГО ЗАВДАННЯ.....	10
1.1 Огляд предметної області	10
1.2 Огляд існуючих рішень	11
1.2.1 Kayako Resolve	11
1.2.2 Open Ticket Request System.....	12
1.2.3 osTicket.....	13
1.2.4 Zendesk	13
1.3 Формулювання завдання роботи.....	14
1.3.1 Назва та призначення розробки	14
1.3.2 Вимоги до функціональних характеристик	15
1.3.3 Вимоги до програмного забезпечення.....	16
1.3.4 Вимоги до апаратного забезпечення.....	16
Висновки до розділу 1	16
2 ПРОЕКТУВАННЯ СИСТЕМИ.....	17
2.1 Архітектура системи.....	17
2.1.1 Компоненти системи	17
2.1.2 ASP.NET MVC Framework.....	20
2.2 Основні дані системи.....	22
2.2.1 Проектування бази даних.....	23
2.2.2 Інтерфейс роботи з базою даних	27
2.3 Взаємодія веб-клієнта з сервером	29
2.3.1 Архітектурний стиль REST	30

	5
2.3.2	Технологія Long Polling 31
2.3.3	Робота системи аутентифікації..... 32
2.4	Середовище розробки..... 33
2.4.1	Visual Studio..... 33
2.4.2	JetBrains Rider..... 35
2.4.3	Eclipse..... 36
2.5	Бази даних..... 37
2.5.1	MySQL 38
2.5.2	Microsoft SQL Server..... 38
2.5.3	PostgreSQL..... 39
	Висновки до розділу 2 40
3	МЕТОДИ ДЛЯ ВЗАЄМОДІЇ З СЕРВЕРОМ..... 41
3.1	Головна сторінка додатку 41
3.2	Авторизація та реєстрація 41
3.3	Сторінка створення компанії 43
3.4	Сторінка компанії 44
3.5	Сторінка з діалогом..... 45
3.6	Сторінка управління компанії 46
	Висновки до розділу 3 50
4	ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ 51
4.1	Загальні питання з охорони праці 51
4.2	Аналіз стану та умов праці 51
4.2.1	Вимоги до приміщення 52
4.2.2	Вимоги до організації робочого місця..... 52
4.2.3	Навантаження та напруженість процесу праці..... 54
4.3	Виробнича санітарія 55

	6
4.3.1 Загальні заходи безпеки	55
4.3.2 Електробезпека.....	56
4.3.3 Мікроклімат.....	58
4.3.4 Освітлення	58
4.3.5 Рекомендації щодо пожежної безпеки	61
Висновок до розділу 4	63
ВИСНОВКИ.....	64
СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ.....	65
ДОДАТОК А ПРЕЗЕНТАЦІЯ	68
ДОДАТОК Б ЛІСТИНГ ФУНКЦІЇ ДОДАННЯ НОВОГО БОТУ “ВКОНТАКТЕ”	73

ВСТУП

На сьогоднішній день широку популярність набули веб-додатки, які виконують найрізноманітніші задачі, як ресурсозатратні обчислення так і операції, що майже не потребують ресурсів комп'ютера. За такої умови від користувачу потрібен лише браузер та доступ до мережі Internet.

Різноманітність операційних систем вимусила розробників зміщувати свій пріоритет на веб-додатки, таке рішення має безліч переваг. Розробнику потрібно лише вибрати платформу, на якій буде розміщена серверна частина програми і потурбуватися про невеликі відмінності у відображенні інформації в різних браузерах. Це дозволяє швидко та ефективно розробляти програми, без проблем вести підтримку та впроваджувати нові функції.

Одна з головних характеристик інтернет-додатків є те, що вони поширюються у вигляді сервісу, а не товару, тобто використовують бізнес-модель Software as a service (SaaS). Така модель не має проблем з неліцензійним поширенням програмного забезпечення (ПЗ), дозволяє радикально скоротити витрати на розгортання системи в організації, відпадає потреба встановлення спеціалізованого ПЗ на робочі місця користувачів, доступ до ПЗ здійснюється через звичайний браузер.

Рано чи пізно компанії, що ведуть технічну підтримку у соціальних мережах, месенджерах, займаються продажами чи покупками або пропонують там свій продукт, стикаються з необхідністю збору та обробки повідомлень в одному місці.

Розробивши систему, де підприємець буде реєструвати своє підприємство та надасть доступ працівникам, прив'яже потрібні йому соціальні мережі або месенджери, після сервіс почне збирати повідомлення в базу та даних структуровано відображати працівнику на веб-сторінці підприємства, можливо скоротити час відповіді і покращити цим якість підтримки або кількість продаж.

“ВКонтакте” (міжнародна назва: “VK”) – соціальна мережа доступна на більш ніж 90 мовах, друга за розміром у Європі, найбільш популярна в Росії, Білорусі, Казахстані та Узбекистані. В соціальній мережі велика кількість Internet магазинів та спільнот компаній, які можуть вести технічну через повідомлення групи. Має API для роботи з повідомленнями, що сприяє створенню чат-ботів.

“Telegram” – месенджер доступний на більшості платформах, дозволяє обмінюватись текстовими повідомленнями та медіа файлами, є можливість створення чат-бота. За допомогою “Telegram” можна вести технічну підтримку продукту у зручному місці для користувача.

Таким чином, метою даного дипломного проекту є розробка системи технічної підтримки організацій у соціальних мережах для збору повідомлень, автоматизованої їх передачі до співробітника, подальшої обробки, та відправлення зворотної відповіді.

Перелік умовних позначок та скорочень

ACID	Atomicity, Consistency, Isolation, Durability
API	Application Programming Interface
ASP	Active Server Pages
ER	Entity-Relationship
HTML	Hypertext Markup Language
HTTP	Hypertext Transport Protocol
IDE	Integrated Development Environment
LINQ	Language Integrated Query
MVC	Model-View-Controller
REST	Representational state transfer
URL	Uniform Resource Locator
XML	Extensible Markup Language

1 АНАЛІЗ СИСТЕМ ТЕХНІЧНОЇ ПІДТРИМКИ ТА ФОРМУЛЮВАННЯ ТЕХНІЧНОГО ЗАВДАННЯ

1.1 Огляд предметної області

З розвитком нових технологій та онлайн-бізнесу сфера обслуговування клієнтів вийшла на якісно новий рівень. Зокрема, це стосується задоволення запитів споживачів. Тепер, якщо у користувача з'являється проблема, він може залишити запит з описом виниклого питання і проханням про допомогу в її вирішенні.

Проте, процес відправлення запитів в службу підтримки не такий простий, як може здатися на перший погляд. Він включає заповнення форми, відправку e-mail з підтвердженнями, аналіз і вирішення проблеми та інше.

При продажі товарів та пропонування свого продукту або послуг у мережі Internet також можлива робота через соціальні мережі та месенджери, це дозволяє зацікавити клієнта під час його відпочинку, без прямого пошуку товару. Тому є важливим швидко відповісти на запитання клієнта, або обробити його заявку на покупку.

Коли підприємець використовує соціальні мережі або месенджери для роботи з клієнтами. Користувач має можливість задати питання там де йому зручно. Такий підхід може покращити якість підтримки. Але при цьому підприємець змушений слідкувати за повідомленнями на різних веб-сторінках, що збільшує час відповіді, який є важливим для кожного клієнта.

Робота з соціальними мережами вимагає розробки чат-бота, який буде підключатися до application programming interface (API) і відправляти повідомлення на яку соціальна мережа або месенджер буде відправляти нові повідомлення, при такому підході отримати старі повідомлення не є можливим, для вирішення цієї проблеми потрібне створення бази даних, в якій буде зберігатися весь діалог між працівниками підтримки та клієнтом.

1.2 Огляд існуючих рішень

Розглянемо найбільш відомі рішення з тих, що представлені на ринку в даній області, а саме Kayako Resolve, Cerberus HelpDesk, osTicket, Live Agent.

1.2.1 Kayako Resolve

Багатофункціональна система підтримки користувачів роботі з заявками Kayako є повним набором інструментів, включаючи програмне забезпечення для чату в реальному часі. Kayako здатний вести підтримку по електронній пошті і приймати заявки, через чат, дзвінки або центр самообслуговування, в ньому зберігається вся історія підтримка клієнтів.

Програмне забезпечення служби підтримки Kayako постачається з потужною функціональністю, що полегшує роботу команд обслуговування клієнтів, щоб розпочати керування запитами та бесідами по будь-якому каналу, краще підтримувати клієнтів і залишатися особистими в міру зростання.

Переваги:

- підтримка чату в реальному часі;
- перегляд статистики;
- налаштування робочих процесів;
- хрестовий відділ підтримки.

Недоліки:

- займає великий об'єм пам'яті;
- складний інтерфейс;
- відсутність російського мови в інтерфейсі;
- відсутність підтримки соціальних мереж.

1.2.2 Open Ticket Request System

Open Ticket Request System – безкоштовна helpdesk система, якісно написана на мові perl з підтримкою безлічі СУБД. OTRS використовують такі гіганти, як Яндекс, Mail.RU, REG.RU, Mikrotik, RU-CENTER і багато інших.

Функції:

- взаємодія с багатьма БД;
- створення звітів;
- зручний пошук;
- можливість роботи за допомогою мобільного.

Переваги:

- систему можливо встановити як на windows так і на linux системи;
- можливість інтеграції з LDAP;
- можливість створювати відділи (черги звернень);
- особистий кабінет для користувачів;
- гнучка настройка прав користувачів;
- «Безмежний» пошук;
- російська локалізація та ще 34 мови;
- підтримка ITIL, так само є ITSM версія;
- «Особистий кабінет» для клієнтів, де видно всю історія звернень;
- гнучка система звітів і багато іншого;
- існує дуже зручна документація та російська спільнотою.

Недоліки:

- відсутність підтримки російської або української мов;
- висока вартість;
- заплутаність інтерфейсу;
- відсутність підтримки соціальних мереж.

1.2.3 osTicket

osTicket – це організація технічної підтримки користувачів в режимі реального часу, включаючи відстеження потоків електронної пошти, вхід на поштовий сервер по протоколу POP3, необмежену кількість адрес email, методи уникнення зациклення автовідповідачів, та ін.

Переваги:

- наявність безкоштовної версії;
- установка на Windows, Mac OS X і Linux;
- гнучка настройка практичних всіх основних функцій;
- міжнародний статус продукту – підтримується безліч мов;
- клієнту не потрібно проходити процес реєстрації для створення заявки.

Недоліки:

- запит вважається закритим відразу після відповіді співробітника-без можливості для користувача що-небудь уточнити або прокоментувати;
- клієнт може бачити тільки власні заявки, немає можливості вивчити чужі запити зі схожою темою;
- працювати з системою можна тільки при наявності розробника, який буде її підтримувати і додавати функціонал під потреби компанії;
- відсутність підтримки соціальних мереж.

1.2.4 Zendesk

Zendesk – це веб-сервіс з низкою додатків для підтримки клієнтів і відвідувачів. Він допомагає співробітникам, менеджерам і адміністраторам допомагати клієнтам на ходу. Мобільні додатки допомагають швидко переглядати зведені дані запитів – обсяг і швидкість дій, реагування, оцінку якості підтримки і т. д. Zendesk дозволяє переглядати профіль клієнта з заявками, і працювати з додатком жестами.

Переваги:

- дозволяє працювати з загальним ящиком Gmail;
- дозволяє створювати і приєднується з сеансів технічної підтримки;
- присутні мобільні додатки;
- працює завантаження і виведення файлів, прикріплених до звернень.

Недоліки:

- дуже повільне завантаження;
- поганий UI;
- досить часто відповіді не відправляються і доводиться натискати кілька разів і перезавантажувати браузер, викликає купу незручностей;
- відсутність підтримки соціальних мереж.

По результатам проведеного аналізу можна зробити висновок, про необхідність розробки системи, оскільки жодна з розглянутих систем не відповідає у повній мірі вимогам впровадження для роботи з соціальними мережами та месенджерами.

1.3 Формулювання завдання роботи

1.3.1 Назва та призначення розробки

Web-система технічної підтримки клієнтів за допомогою соціальних мереж.

Забезпечити роботу серверної частини сервісу технічної підтримки клієнтів у соціальних мережах та месенджерах на будь-яких пристроях з Web-браузером.

Систему має бути поділено на 3 модулі: модуль Web-додатку, серверна частина та модуль баз даних.

1.3.2 Вимоги до функціональних характеристик

Необхідно реалізувати функції серверної частини додатку та базу даних, забезпечити інтерфейси для взаємодії веб-клієнту з серверною частиною.

Реалізувати систему реєстрації та авторизації користувачів за допомогою e-mail та паролю для безпечної роботи та конфіденційності діалогів.

Система має повністю забезпечувати роботу з технічної підтримки через соціальну мережу VK та месенджер Telegram, для цього потрібно розробити API для отримання повідомлень від них та відправлення відповіді назад. Реалізувати можливість створення сторінки підприємства де можна підключити спільноту соціальної мережі VK, бота у месенджері Telegram та додати співробітників для роботи.

Необхідно забезпечити комунікацію між співробітником та клієнтом, для миттєвої передачі повідомлень на веб-сторінку підприємства зі списком діалогів.

Усі дані підприємств, облікових записів та діалоги системи повинні зберігатися в структурованому вигляді під управлінням реляційної СУБД.

Систему має бути розроблено таким чином щоб уможливити подальше розширення функцій та спростити внесення змін у існуючі функції.

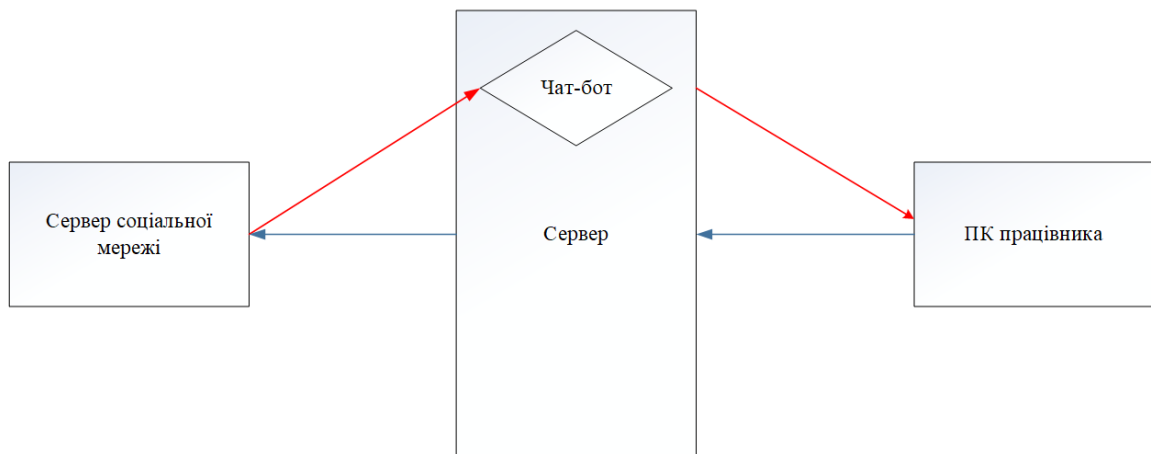


Рисунок 1.1 – Схема роботи додатку

1.3.3 Вимоги до програмного забезпечення

Серверна частина та база даних повинні бути розраховані для роботи на Internet Information Services 7.5 і вище під керуванням операційної системи Windows Server 2012 і вище. База даних повинна розміщуватись на Microsoft SQL Server 2016.

1.3.4 Вимоги до апаратного забезпечення

Мінімальні системні вимоги для запуску програмного забезпечення на Web-сервері:

- процесор – тактова частота якого не менше 2ГГц та кількість ядер не менше ніж 2;
- оперативна пам'ять – об'єм якої не менше 1028 МБ;
- дискова пам'ять – об'єм якої не менше 20ГБ.

Висновки до розділу 1

У розділі було розглянуто предметну область та порівняно декілька існуючих рішень систем технічної підтримки. Крім того було сформовано призначення розробки системи, функціональні характеристики, що розроблюється, вимоги до програмного та апаратного забезпечення.

2 ПРОЕКТУВАННЯ СИСТЕМИ

2.1 Архітектура системи

2.1.1 Компоненти системи

Система складається з трьох основних компонентів зображених на рис.2.1 – серверу додатку, веб-клієнта та серверу баз даних. Серверна частина додатку відіграє центральну роль у роботі системи, вона поєднує у собі сервер автентифікації та авторизації, чат ботів, сервер ресурсів.



Рисунок 2.1 – Схема розгортання системи

Сервер додатку відповідає за обробку інформації та передачу її між клієнтами та сервером бази даних і містить підсистему автентифікації, підсистему керування користувачами та організаціями, підсистему керування діалогами та підсистему чат-ботів. Основною є підсистема керування користувачами та організаціями, яка відповідає за створення, зміну і збереження таких даних, як обліковий запис користувача, профілі організації та її учасників. Підсистема автентифікації залежить від підсистеми керування користувачами та відповідає за їх ідентифікацію і видачу повноважень на роботу з системою. Підсистема діалогів відповідальна створення та керування діалогами. За прийом повідомлень від соціальних мереж та месенджерів відповідальні чат-боти.

Основна задача серверу додатку відправлення веб-представлення користувачу та забезпечення взаємодії сторінки з підсистемами.

Данні про користувачів та організації та діалоги зберігають на сервері баз даних.

2.1.1.1 Створення облікового запису користувача

Для створення нового облікового запису користувачу треба перейти на сторінку реєстрації та заповнити форму. У формі вказується ім'я та прізвище, адресу електронної скриньки користувача, яка буде використовуватись для подальшого входу в систему, пароль та його повторення. На сервері форма обробляється, у базі створюється запис про користувача, та присвоюються роль користувача.

Якщо створення облікового запису пройшло успішно користувача переадресує на сторінку де можна почати роботу.

2.1.1.2 Створення організації та запрошення користувачів

Для створення організації потрібно перейти на робочу сторінку, якщо ви не створили організації на ній буде розміщено посилання на сторінку з формою створення організації.

Після відправлення форми на сервер, створюється запис нової організації у базу, користувачеві, що створив організацію, додається роль керівника. Після створення користувача перенаправляє на сторінку з управлінням організації.

Щоб приєднатися до організації, що вже існує, її керівник повинен відправити форму з електронною адресою співробітника, сервер додасть у запис користувача в базі даних ідентифікатор організації.

2.1.1.3 Додання чат-боту до організації

Для підключення чат-боту до організації необхідно перейти на сторінку управління організації та заповнити форму для боту соціальної мережі “Вконтакте” або месенджеру “Telegram”.

Форма підключення боту “Вконтакте” містить токен авторизації, ід групи до якої приєднується бот, назва боту. Після відправлення форми сервер відправляє запит з токеном авторизації до API соціальної мережі для підключення, у базу даних вносить дані про підключення та ід належності організації.

Форма підключення боту “Telegram” містить токен авторизації, назва боту. Після відправлення форми сервер відправляє запит з токеном авторизації до API месенджеру для підключення, у базу даних вносить дані про підключення та ідентифікаційний номер належності організації.

2.1.1.4 Взаємодія системи з соціальною мережею чи месенджером

Для автоматизованої передачі повідомлень від соціальної мережі або месенджера використовується схема відображена на Рис. 2.2.

Після відправки повідомлення на сервер від соціальної мережі чи месенджеру, повідомлення отримує чат-бот, що визначає відправника та за допомогою інтерфейсу роботи з базою даних надсилає повідомлення до серверу бази даних, для подальшого використання.

Веб-клієнт відправляє запити на сервер для отримання нових повідомлень та очікує. Сервер додатку відправляє запит до бази даних та перевіряє чи є не прочитані повідомлення.

Після появи нового повідомлення сервер складає відповідь веб-клієнту, та відсилає йому повідомлення. Веб-клієнт отримує відповідь та відображує в представленні, після робить новий запит та очікує відповіді.

Якщо користувач відправляє відповідь на повідомлення, сервер звертається до чат-боту, який формує повідомлення у відповідь до соціальної мережі або месенджера.

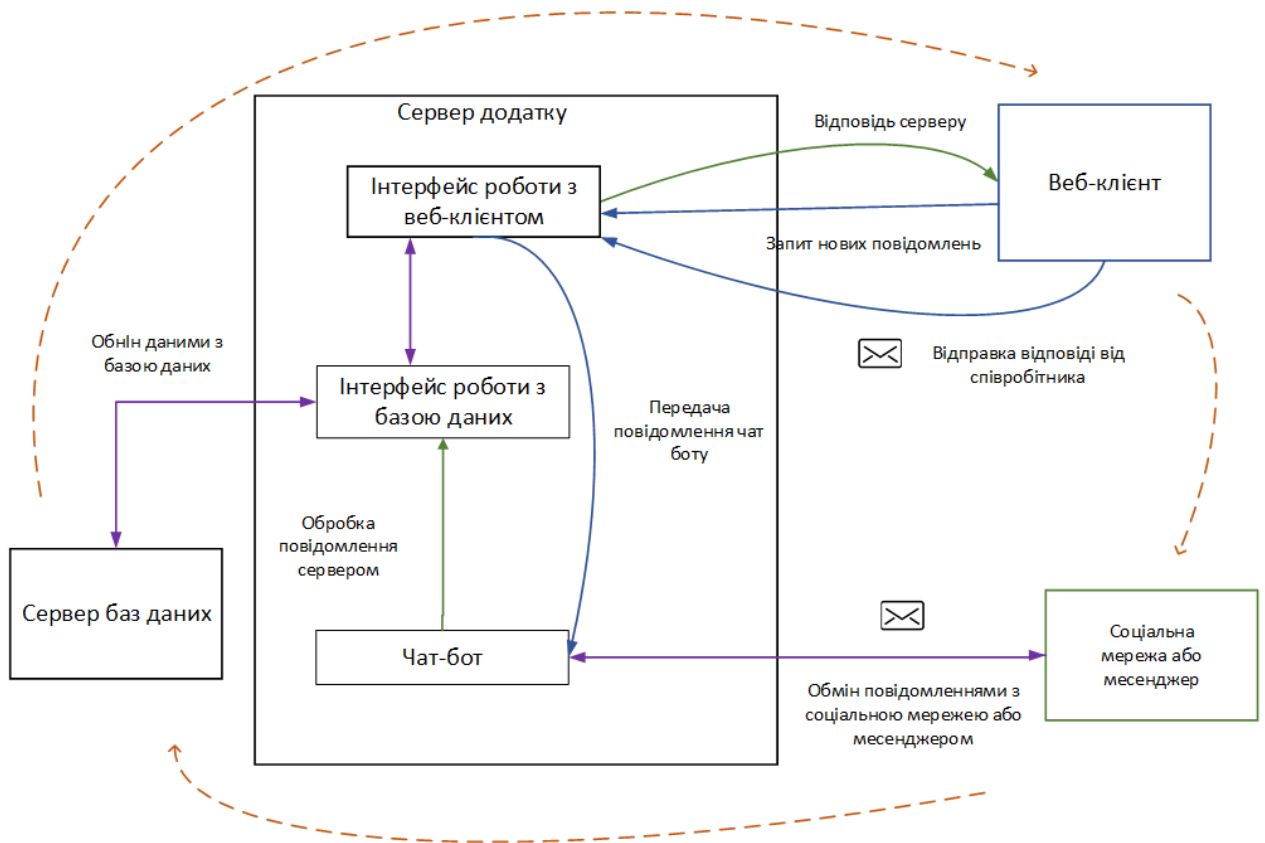


Рисунок 2.2 – Схема циклу обміну повідомлення

2.1.2 ASP.NET MVC Framework

Для реалізації системи обрано шаблон проектування Model-view-controller та фреймворк, що реалізує цей шаблон ASP.NET MVC Framework.

Платформа ASP.NET MVC являє собою фреймворк для створення сайтів і веб-додатків за допомогою реалізації патерну MVC.

Концепція шаблону MVC передбачає поділ програми на три компоненти:

1. Контролер (controller) представляє клас, що забезпечує зв'язок між користувачем і системою, поданням і сховищем даних. Він отримує вводяться користувачем дані і обробляє їх. І в залежності від результатів

обробки відправляє користувачу певний висновок, наприклад, у вигляді подання.

4. Представлення (view) - це власне візуальна частина або користувацький інтерфейс програми. Як правило, html-сторінка, яку користувач бачить, зайшовши на сайт.

3. Модель (model) представляє клас, що описує логіку використовуваних даних.

Загальна схему взаємодії цих компонентів зображена на Рис. 2.3. У цій схемі модель є незалежним компонентом будь - які зміни контролера або подання не зачіпають модель. Контролер і подання є відносно незалежними компонентами, і нерідко їх можна змінювати незалежно один від одного.

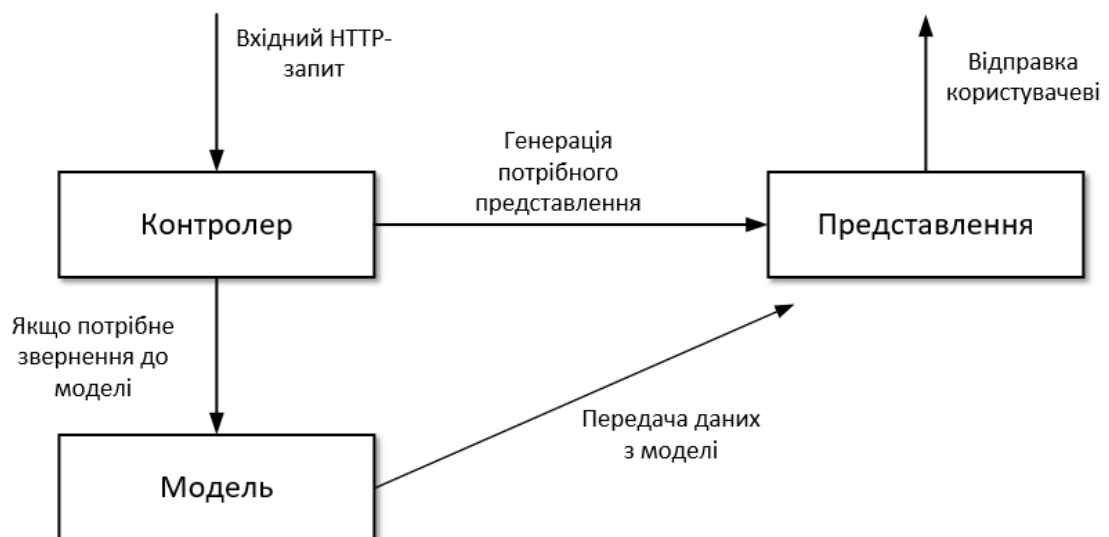


Рисунок 2.3 – Схема роботи шаблону Model-view-controller

Завдяки цьому реалізується концепція поділ відповідальності, у зв'язку з чим легше побудувати роботу над окремими компонентами. Крім того, внаслідок цього додаток має кращу тестованість. І якщо нам, припустимо, важлива візуальна частина, то ми можемо тестувати уявлення незалежно від контролера. Або ми можемо зосередитися на бекенді і тестувати контролер.

Конкретні реалізації і визначення даного шаблону можуть відрізнятися, але в силу своєї гнучкості та простоти він став дуже популярним останнім часом, особливо в сфері веб-розробки.

При HTTP запиті контролер визначає метод до якого звертається клієнт, прикладами таких методів можуть бути:

- а) запит представлення;
- б) виконання обчислень;
- в) занесення даних у базу даних.

Контролер, якщо потрібне внесення або вивід даних, звертається до моделі. Згенероване контролером представлення та данні передані з моделі відправляються користувачу.

2.2 Основні дані системи

Мета моделювання даних полягає в забезпеченні розробника інформаційної системи концептуальна схема бази даних у вигляді однієї або декількох моделей окремих частин моделей, які без проблем можуть бути відображені в будь-яку систему баз даних.

Логічна модель бази даних – версія концептуальної моделі, яка може бути реалізована конкретною БД.

Логічна модель відображає логічні зв'язки між атрибутами об'єктів незалежно від їх змісту і середовища зберігання. Як правило, при розробці логічних структур використовуються в основному евристичні методи проектування з формальною оцінкою якості прийнятих рішень.

Для логічного проектування реляційних БД застосовуються наступні методики:

1. Метод моделювання “сутність-зв'язок” дає абстрактну модель предметної області, використовуючи наступні основні поняття: сутності, взаємозв'язку між сутностями і атрибути для представлення властивостей сутностей і взаємозв'язків.

2. Метод багатовимірного моделювання дає абстрактну модель предметної області, використовуючи наступні основні поняття: показники або показники, факти і вимірювання .

3. Методи моделювання тимчасових даних дають абстрактну модель фрагмента предметної області, що представляє часові ряди даних, і використовують наступні основні поняття: тимчасові мітки , часовий ряд , дата, діапазон дат, класи.

4. Метод моделювання “звід даних” дає абстрактну модель фрагмента предметної області, базуючись на математичних принципах нормалізації відносин і використовує такі основні поняття: сутності-концентратори, що зв'язують сутності, сутності-сателіти.

2.2.1 Проектування бази даних

Для проектування бази даних у проекті використана модель «сутність-зв'язок» (англ. "Entity-Relationship model"), або ER-модель є найбільш відомим представником способів графічного представлення моделей даних.

Основні переваги ER-моделей:

- наочність;
- моделі дозволяють проектувати бази даних з великою кількістю атрибутів та сутностей;
- а більшості системах автоматизованого проектування баз даних реалізовані ER-моделі .

Основні елементи ER-моделей:

- об'єкти (сутності);
- атрибути об'єктів;
- зв'язку між об'єктами.
- сутність — об'єкт предметної області, що має атрибути.

Зв'язок між сутностями характеризується:

- типом зв'язку (1:1, 1:N, N:M);

– класом приналежності. Клас може бути обов'язковим і необов'язковим. Якщо кожен примірник сутності бере участь у зв'язку, то клас приналежності – обов'язковий, інакше – необов'язковий.

Для початку потрібно визначити основні об'єкти системи, данні яких необхідно зберігати. Для проєктованої системи такими об'єктами є:

Інформація про користувача, обліковий запис та приналежність до компанії;

- дані про бота “Вконтакте”;
- дані про бота “Telegram”;
- інформація про окремий діалог;
- інформація про окреме повідомлення.
- наступним етапом є визначення атрибутів для кожного об'єкту.

Інформація про користувача, обліковий запис та приналежність до компанії – в цьому об'єкті мають бути зазначені такі дані:

- ім'я;
- прізвище;
- адреса електронної поштової скриньки;
- пароль;
- назва компанії;
- роль користувача.

Дані про бота “Вконтакте” – в цьому об'єкті мають бути зазначені такі дані:

- назва боту;
- токен авторизації;
- код підтвердження;
- пароль захисту від підробки запитів;
- ідентифікаційний номер групи до якої підключається бот;
- ідентифікаційний номер серверу у групі.

Дані про бота “Telegram” – в цьому об'єкті мають бути зазначені такі дані:

- назва боту;
- токен авторизації.
- інформація про окремий діалог – цей об’єкт зберігає такі дані:
- ім’я співрозмовника;
- прізвище;
- нікнейм;
- ідентифікаційний номер співрозмовник у соціальній мережі або месенджері.

Інформація про окреме повідомлення – в цьому об’єкті мають бути зазначені такі дані:

- повне ім’я співрозмовника;
- текст повідомлення;
- час та дата відправлення повідомлення.

Після визначення атрибутів, розділено об’єкти на сутності:

- акаунт;
- роль;
- компанія;
- бот “Вконтакте”;
- бот “Telegram”;
- діалог;
- повідомлення.

Наступним етапом, для завершення проектування, є визначення типу зв’язків між сутностями.

Зв’язок – це асоціація, встановлена між кількома сутностями.

Між двома сутностями, наприклад, А і В можливі три види зв’язків.

Перший тип – зв’язок Один-до-Одного (1:1): у кожен момент часу кожному представнику сутності А відповідає ні одного чи один представник сутності В.

Другий тип – зв'язок Один-до-Багатьох (1:M): одному представнику сутності А відповідають ні одного, один або декілька представників сутності В.

Третій тип – зв'язок Багато-до-Багатьох (M:N): ні одному, одному або декільком представникам сутності А відповідають ні одного, один або декілька представників сутності В.

Після визначення типів отримали такі зв'язки:

– зв'язок Бот “Вконтакте” – Компанія відноситься до другого типу зв'язку M:1. Первинний ключ сутності Компанія повинен бути в сутності Бот “Вконтакте”;

– зв'язок Бот “Telegram” – Компанія відноситься до другого типу зв'язку M:1. Первинний ключ сутності Компанія повинен бути в сутності Бот “Telegram”;

– зв'язок Діалог – Компанія відноситься до другого типу зв'язку M:1. Первинний ключ сутності Компанія повинен бути в сутності Діалог;

– зв'язок Повідомлення – Діалог відноситься до другого типу зв'язку M:1. Первинний ключ сутності Діалог повинен бути в сутності Повідомлення;

– зв'язок Діалог – Бот “Вконтакте” відноситься до другого типу зв'язку M:1. Первинний ключ сутності Бот “Вконтакте” повинен бути в сутності Діалог.

– зв'язок Діалог – Бот “Telegram” відноситься до другого типу зв'язку M:1. Первинний ключ сутності Бот “Telegram” повинен бути в сутності Діалог;

– зв'язок Повідомлення – Діалог відноситься до другого типу зв'язку M:1. Первинний ключ сутності Діалог повинен бути в сутності Повідомлення;

– зв'язок Акаунт – Компанія відноситься до другого типу зв'язку M:1. Первинний ключ сутності Компанія повинен бути в сутності Акаунт;

– зв'язок Акаунт – Роль відноситься до третього типу зв'язку М:М. Необхідно ввести додаткову сутність, яка буде містити первинний ключ обох сутностей.

Після проектування можна візуально зобразити схему фізичну схему бази даних Рис. 2.4.

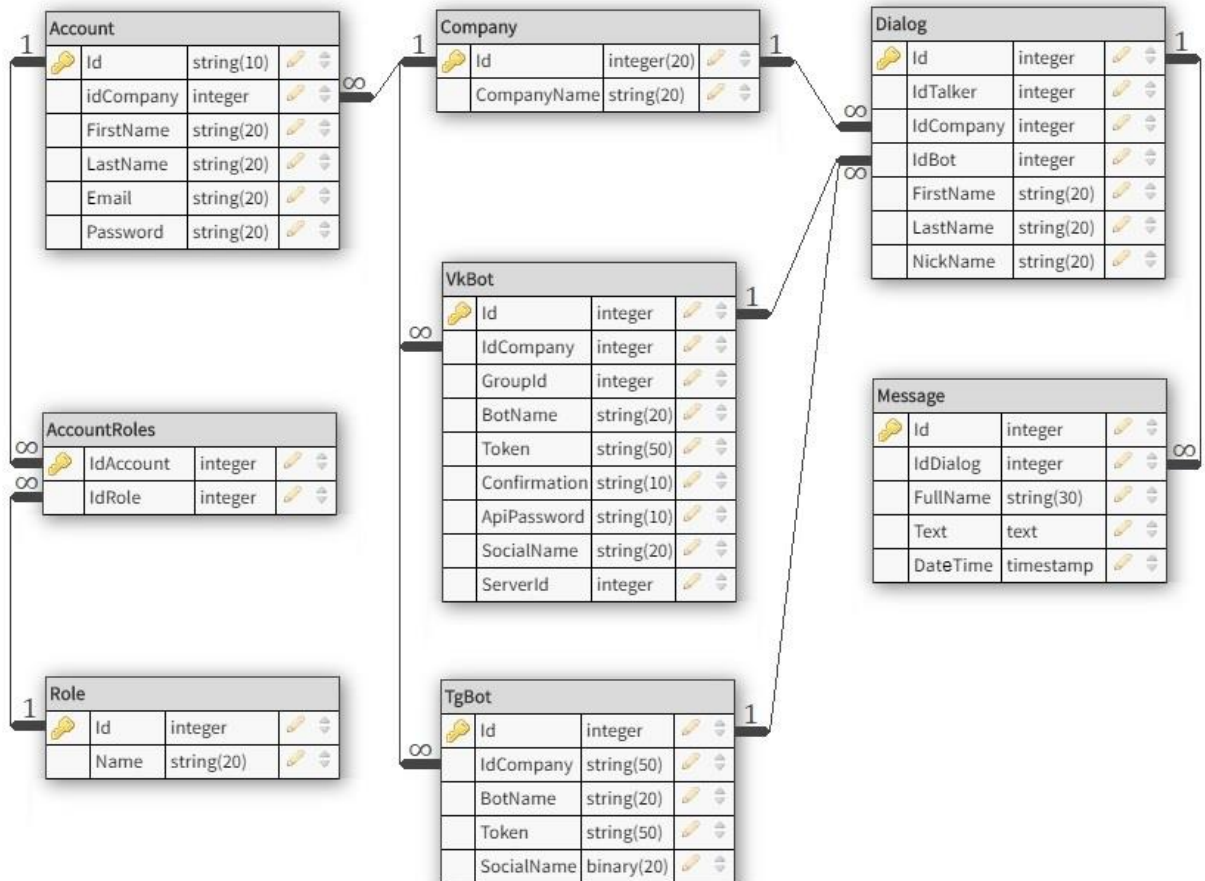


Рисунок 2.4 – Схема бази даних системи

2.2.2 Інтерфейс роботи з базою даних

Для роботи з базою даних використовується технологія Entity Framework, що спрощує створення, внесення та запит даних за допомогою технології LINQ.

Entity Framework представляє спеціальну об'єктно-орієнтовану технологію на базі фреймворка .NET для роботи с даними. Entity Framework представляє собою більш високий рівень абстракції, який дозволяє

абстрагуватися від самої бази даних і працювати з даними незалежно від типу сховища.

Центральною концепцією Entity Framework є поняття сутності або entity. Сутність представляє набір даних, асоційованих з визначеним об'єктом. Тому дана технологія передбачає роботу не з таблицями, а з об'єктами та їх наборами.

Будь-яка сутність, як і будь-який об'єкт із реального миру, володіє рядом властивостей. Наприклад, якщо сутність описує людину, то ми можемо виділити такі властивості, як ім'я, прізвище, ріст, вік, вага. Властивості необов'язково представляють прості дані типу цілого числа, але й можуть представляти більш комплексні структури даних. У кожній сутності може бути одне або декілька властивостей, які будуть відрізняти цю сутність від інших та будуть унікально визначати цю сутність. При цьому сутності можуть бути зв'язані асоціативним зв'язком один-до-багатьох, один-до-одного та багато-до-багатьох, подібно тому, як у реальній базі даних проходить зв'язок через зовнішні ключі.

Відмінною рисою Entity Framework є використання запитів LINQ для вибору даних з бази даних. За допомогою LINQ ми можемо не тільки виймати окремі рядки, які зберігають об'єкти із бази даних, але й отримувати об'єкти, зв'язані різними асоціативними зв'язками.

Іншими ключовим поняттям є Entity Data Model. Ця модель зіставляє класи сутностей з реальними таблицями у базі даних. Entity Data Model складається із трьох рівнів: концептуального, рівень сховища та рівень зіставлення.

На концептуальному рівні відбувається визначення класів сутностей, використовуваних у додатку.

Рівень сховища визначає таблиці, стовпці, відношення між таблицями та типи даних, з якими зіставляється використовувана база даних.

Рівень зіставлення служить посередником між попередніми двома, визначаючи зіставлення між властивостями класу сутності та стовпцями таблиць.

Таким чином, через класи, визначені у додатку, виконується взаємодія з таблицями із бази даних.

Entity Framework передбачає три можливі способи взаємодії з базою даних:

- Database first: Entity Framework створює набір класів, які відображають модель конкретної бази даних.
- Model first: Спочатку розробник створює модель бази даних, по якій Entity Framework створює реальну базу даних на сервері.
- Code first: Розробник створює класи моделі даних, які будуть зберігатися у базі даних, а потім Entity Framework по цій моделі генерує базу даних та її таблиці.

2.3 Взаємодія веб-клієнта з сервером

Для взаємодії веб-клієнта з сервером використовуються HTTP запити з архітектурним стилем REST.

HTTP (англ. HyperText Transfer Protocol – протокол передачі гіпертексту) – протокол прикладного рівня передачі даних спочатку у вигляді гіпертекстових документів у форматі “HTML”, в даний момент використовується для передачі довільних даних. Основою HTTP є технологія «клієнт-сервер», тобто передбачається існування:

- споживачів (клієнтів), які ініціюють з'єднання і посилають запит;
- постачальників (серверів), які очікують з'єднання для одержання запиту, виробляють необхідні дії і повертають назад повідомлення з результатом.

Для того щоб на веб-представленні завжди були відображені найновіші повідомлення використана технологія Long Polling.

2.3.1 Архітектурний стиль REST

REST (скорочення від англ. Representational State Transfer – "передача стану представлення") – архітектурний стиль для взаємодії компонентів додатка, що розподілено в мережі. REST являє собою узгоджений набір правил, що враховуються при проектуванні розподіленої системи. У певних випадках, таких як інтернет-магазини, пошукові системи, інші системи, засновані на даних, це призводить до підвищення продуктивності і спрощення архітектури. В широкому сенсі компоненти в REST взаємодіють зразок взаємодії клієнтів і серверів у Всесвітній павутині.

В мережі Інтернет виклик віддаленої процедури може являти собою звичайний HTTP-запит, а необхідні дані передаються як параметри запиту.

Для веб-служб, побудованих з урахуванням REST, застосовують термін RESTful.

Властивості архітектури, які залежать від обмежень, накладених на REST-системи:

- продуктивність-взаємодія компонентів системи може бути домінуючим фактором продуктивності та ефективності мережі з точки зору користувача;

- масштабованість для забезпечення великої кількості компонентів і взаємодій компонентів;

Вплив архітектури REST на масштабованість:

- простота стандартизованого інтерфейсу;
- відкритість компонентів до можливих змін для задоволення мінливих потреб;

- прозорість зв'язків між компонентами системи для сервісних служб;
- переносимість компонентів системи шляхом переміщення програмного коду разом з даними;

- надійність є стійкістю до відмов на рівні системи при наявності відмов окремих компонентів, з'єднань або даних.

Переваги додатку з використанням REST:

- надійність (за рахунок відсутності необхідності зберігати інформацію про стан клієнта, яка може бути втрачена);
- продуктивність
- масштабованість;
- прозорість системи взаємодії;
- простота інтерфейсів;
- портативність компонентів;
- легкість внесення змін;
- здатність еволюціонувати, пристосовуючись до нових вимог.

2.3.2 Технологія Long Polling

HTTP Long Polling — технологія і спосіб взаємодії між клієнтом і сервером. Long Polling передбачає встановлення постійного з'єднання між клієнтом і сервером.

Клієнт-серверна модель в стандартному вигляді має на увазі, що ініціатором будь-якого обміну даними є клієнт. Сервер завжди знаходиться в мережі і працює, віддаючи контент тільки тоді, коли до нього звертається клієнт, схема роботи зображена на Рис. 2.5.

Така організація не дуже добре підходить для ряду додатків. Наприклад, месенджерів і соціальних мереж. В будь-якій соціальній мережі є стрічка новин або повідомлення, повідомлення в них повинні з'являтися не тоді коли приходить запит від клієнта, а тоді коли на сервер надходять нові дані.

Реалізувати це дозволяє HTTP Long Poling. Клієнт відправляє запит на сервер та очікує відповідь. Сервер відправляє відповідь лише коли з'являються нові дані, після прийому відповіді клієнт відправляє запит знову та очікує.

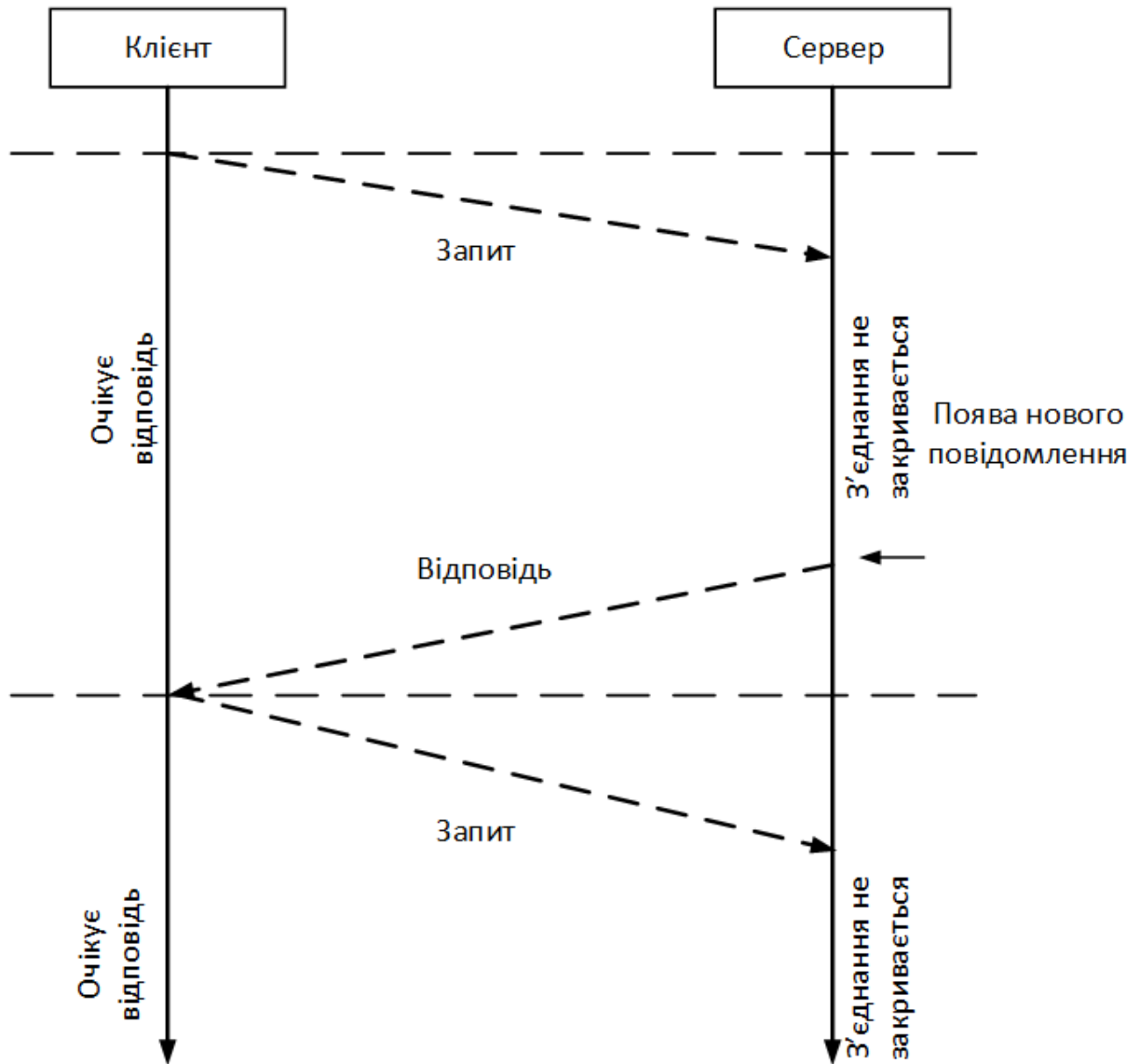


Рисунок 2.5 – Схема роботи HTTP Long Polling

2.3.3 Робота системи аутентифікації

В якості системи аутентифікації, для додатку, було використано ASP.NET Identity на базі фреймворку ASP.NET MVC 5.

ASP.NET Identity представляє вбудовану в ASP.NET систему аутентифікації та авторизації. Дана система дозволяє користувачам створювати облікові записи, аутентифіцироваться, управляти обліковими записами або використовувати для входу на сайт облікові записи зовнішніх провайдерів, таких як Facebook, Google, Microsoft, Twitter та інших.

Ключовими об'єктами в AspNet Identity є користувачі і ролі. Вся функціональність по створенню, видаленню користувачів, взаємодії зі сховищем користувачів зберігається в класі в бібліотеці Microsoft.AspNet.Identity.Core. AspNet Identity дозволяє швидко та ефективно налаштувати систему аутентифікацію у проекті. Для цього при створенні ASP.NET MVC 5 проекту достатньо додати систему аутентифікації, У проекті будуть автоматично створені класи, які реалізують структуру, для всіх необхідних функцій:

- створення таблиць для збереження записів про акаунт та ролі користувачів;
- забезпечення функцій реєстрації та авторизації;
- методи для управління акаунтом.

2.4 Середовище розробки

Середовище розробки – це програма або набір програм, середовище для створення, зміни, тестування та обслуговування програмного забезпечення. Такі програми є інтегрованими середовищами програмування характеризуються тим, що надають функції редагування та компіляції вихідного коду, створення ресурсів програми створення баз даних, компонентів та інших.

В якості середовища розробки системи розглянуто три популярних варіанта:

2.4.1 Visual Studio

Visual Studio – включає в себе повний набір інструментів розробки для створення додатків ASP.NET, веб-служби XML, настільні додатки і мобільні додатки. Visual Basic, Visual C++, Visual C # і Visual J # всі використовують ту ж інтегровану середу розробки (IDE), яка дозволяє їм спільно використовувати інструменти і спрощує створення рішень з використанням

декількох мов програмування. Ці мови дозволяють використовувати функції .Net Framework, яка забезпечує доступ до ключових технологій, що спрощують розробку веб-додатків ASP і XML Web Services з допомогою Visual Web Developer.

Переваги:

а) Офіційний. Так як і мова, і середовище розробки створені в Microsoft, логічно припустити, що нічого більш функціонального ви не знайдете в усьому Інтернеті. В деяких випадках без Visual Studio не обійтися – наприклад, при використанні технологій UWP і WPF;

б) Безкоштовний. Версії “Community edition” для рядового користувача буде достатньо, є можливість підключати плагіни ;

в) Функціональний. У Visual Studio велика кількість якісних плагінів. З їх допомогою можна розширити функціональність програми і підключити інші мови.

г) Підтримує платформи .NET. Visual Studio має широкі можливості по розробці додатків під Windows, в тому числі в .NET-сегменті;

д) Хмарні сховища. Зареєструйтеся в співтоваристві Visual Studio-і отримаєте доступ до хмарного сховища, де зможете розташовувати файли проектів.

Недоліки:

а) Помилки у роботі при переходах з пробної версії. При переході на платну версію можуть губитися налаштування і порушуватися робота корпоративного сервера.

б) Складність. Самостійно освоїти Visual Studio складна в освоєнні – велика кількість доступних функцій, прихованих в підрозділах меню.

2.4.2 JetBrains Rider

JetBrains Rider — платформна інтегроване середовище розробки програмного забезпечення для платформи .NET, що розробляється компанією JetBrains. Підтримуються мови програмування C#, VB.NET і F#.

Переваги:

а) Раніше був плагіном для Visual Studio, що підвищував ефективність її роботи, під назвою ReSharper. Тепер на його основі випущене середовище розробки.

б) Підтримка повного циклу. Rider дає можливість організувати весь цикл створення ПЗ: від ідеї до підтримки.

в) Функціональність. Rider дозволяє підключити MSBuild і XBuild, працювати з CLI-проектами і організувати налагодження додатків .NET and Mono. Велика кількість опцій для швидкого створення коду покращує продуктивність.

г) Multiple runtime. Підтримка декількох запущених програм.

д) Багатоплатформеність. JetBrains Rider працює з Windows, Linux і MacOS.

е) Контроль версій. Вбудований інструмент дозволяє безпосередньо організувати роботу з системами управління версіями.

Недоліки:

а) Молодість. Частина функціональності ще в розробці тому не всі стартові недоліки виправлені.

б) Вартість. Найдешевша версія Project Rider обійдеться в 139 доларів за перший рік використання. Є пробна версія та спеціальні пропозиції для студентів і непрофільних організацій.

2.4.3 Eclipse

Eclipse - одна з популярних багатомовних середовищ. Орієнтована переважно на розробку Java-додатків, але може використовуватися і для розробки на C#.

Переваги:

- а) Велика кількість плагінів.
- б) Активна спільнота. Допомагає швидше освоїти середовище розробки, випускає нові плагіни.
- в) Швидкий компілятор і відладчик. Перший працює на порядок швидше, ніж у конкурентів, другий – показує потоки, перетину, дозволяє гнучко управляти ходом налагодження.
- г) Кастомізація. Завдяки плагінам і налаштувань можна повністю персоналізувати Eclipse.
- д) Безкоштовність. Це проект з відкритим кодом і є абсолютно безкоштовним.
- е) Висока функціональність. Завдяки розробникам і членам спільноти за допомогою Eclipse можна провести будь-який C # - продукт по повному циклу розробки.

Недоліки:

- а) Складність. Як і будь-який функціональний продукт, Eclipse може здатися новачкові складним.
- б) Немає гарантій надійності. Так як плагіни створюються спільнотою, за їх якість відповідає тільки розробник. Крім того, самі творці Eclipse з кожною новою версією погіршують продукт.

В якості середовища розробки було обрано Visual Studio, на Рис. 2.6 зображено знімок екрану з відкритим проектом у програмі, ця IDE має важливі переваги над іншими середовищами з підтримкою C#, такі як:

- а) Програма як і мова C# є продуктом компанії Microsoft;

б) Широка спільнота та велика кількість плагінів спрощуючих розробку.

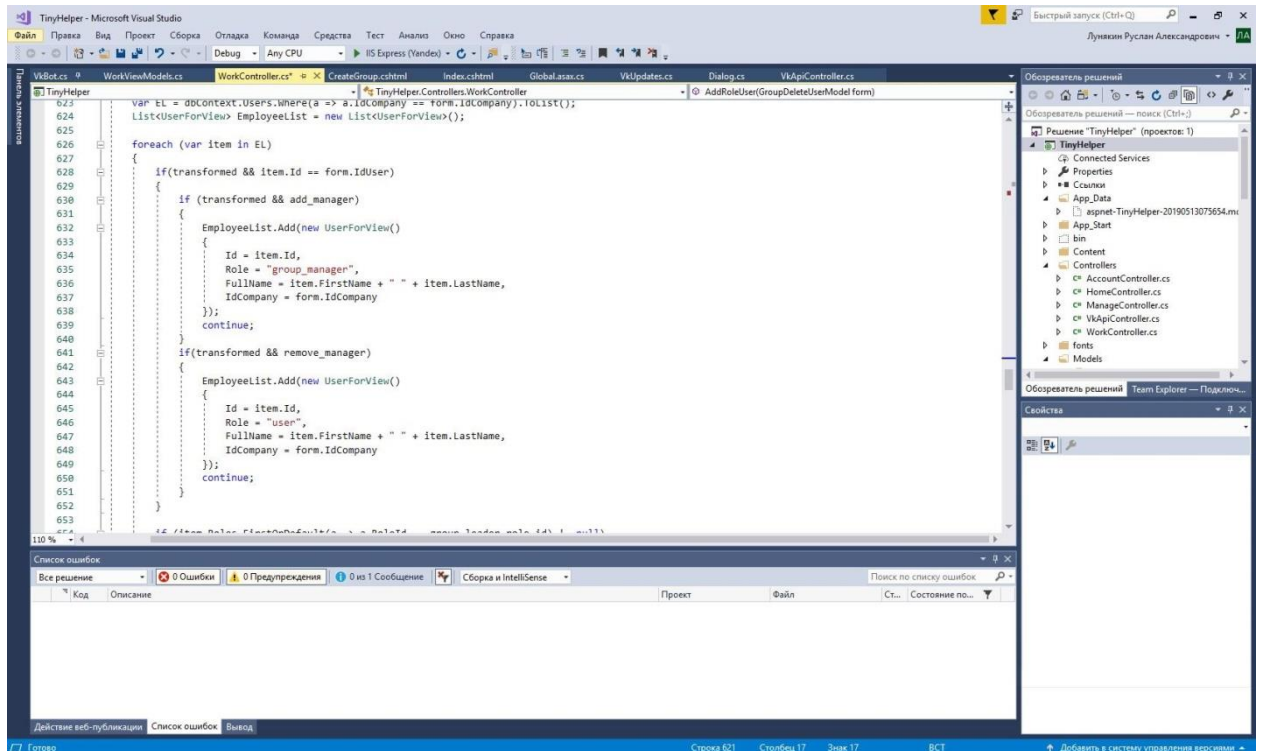


Рисунок 2.6 – Знімок екрану з відкритим проектом у Visual Studio

2.5 Бази даних

Для зберігання великої кількості даних необхідно мати систему, яка надає можливість швидкого внесення та отримання даних. СУБД - Система управління базами даних є набором даних і програм, що надають доступ до цим даним. Основні характеристики СКБД

- а) контроль за надмірністю даних;
- б) несуперечливість даних;
- в) підтримка цілісності бази даних;
- г) цілісність описується за допомогою обмежень;
- д) незалежність прикладних програм від даних;
- е) спільне використання даних;
- ж) підвищений рівень безпеки;

В якості СКБД розглянуто три популярних варіанта:

2.5.1 MySQL

MySQL — це найпопулярніша з усіх великих серверних БД. Проста в освоєнні, в мережі про неї можна знайти велику кількість інформації. Хоча MySQL і не намагається повністю реалізувати SQL-стандарти, вона пропонує широкий функціонал. Додатки спілкуються з базою даних через процес-демон.

Переваги:

- а) Простота: MySQL легко встановлюється. Існує багато сторонніх інструментів, включаючи Візуальні, що полегшують початок роботи з БД;
- б) Багато функцій: MySQL підтримує більшу частину функціоналу SQL;
- в) Безпека: в MySQL вбудовано багато функцій безпеки;
- г) Швидкість: нехтування деякими стандартами дозволяє MySQL працювати продуктивніше, місцями зрізуючи на поворотах.

Недоліки:

- а) Відомі обмеження: за визначенням, MySQL не може зробити все, що завгодно, і в ній присутні певні обмеження функціональності;
- б) Питання надійності: деякі операції реалізовані менш надійно, ніж в інших реляційна СКБД;
- в) Застій у розробці: хоча MySQL і є продуктом з відкритим кодом, робота над нею сильно загальмована.

2.5.2 Microsoft SQL Server

Microsoft SQL Server – система керування базами даних від компанії Microsoft. Використовується як для невеликих сховищ даних користувача, так і для сховища даних великих корпорацій. Крім підтримки баз даних цей продукт володіє цілим комплексом різноманітних корисних технологій, таких як підтримку бізнес аналітики та нових засобів розробки.

Певаги:

- а) ефективний інструмент для аналітики великих обсягів інформації;
- б) Поліпшені функціональні можливості і тісна інтеграція з різноманітними засобами розробки: Microsoft Office System, Visual Studio;
- в) Відмінні показники безпеки, масштабованості і надійності;
- г) Тісна інтеграція з хмарними системами;
- д) Стабільність роботи системи.

Недоліки:

- а) Відсутній контроль ресурсів. Один користувач з запитом, що потребує велику кількість ресурсів, що сповільнює роботу інших користувачів.
- б) Складний інтерфейс для роботи.

2.5.3 PostgreSQL

PostgreSQL - це найбільш просунута реляційна СКБД, що орієнтується в першу чергу на повну відповідність стандартам і розширюваність. PostgreSQL, або Postgres, намагається повністю відповідати SQL-стандартам.

PostgreSQL відрізняється від інших СКБД тим, що має об'єктно-орієнтованим функціоналом, у тому числі повною підтримкою концепту ACID.

Будучи заснованим на потужній технології Postgres відмінно справляється з одночасною обробкою декількох завдань. Підтримка конкурентності реалізована з використанням MVCC, що також забезпечує сумісність з ACID.

Хоча ця СКБД не так популярна, як MySQL, існує багато сторонніх інструментів і бібліотек для полегшення роботи з PostgreSQL.

Переваги:

- а) Повна SQL-сумісність;
- б) Спільнота: PostgreSQL підтримується досвідченим співтовариством 24/7;

в) Підтримка сторонніми організаціями: незважаючи на дуже просунуті функції, PostgreSQL використовується в багатьох інструментах, пов'язаних з СКБД;

г) Об'єктно-орієнтованість: PostgreSQL - не тільки реляційна, але і об'єктно-орієнтована СКБД.

Недоліки:

а) Продуктивність: у простих операціях читання PostgreSQL може поступатися своїм суперникам;

б) Популярність: через свою складність інструмент не дуже популярний;

в) Хостинг: через перерахованих вище факторів проблематично знайти відповідного провайдера.

В якості СКБД було обрано Microsoft SQL Server, ця система має тісну інтеграцію з обраною IDE Visual Studio.

Висновки до розділу 2

У розділі було розглянуто архітектуру системи, описані основні компоненти системи. Описано основні дані системи, та розглянуто проектування бази даних. Описано основні моменти взаємодії веб-клієнта з системою. Коротко описано основні фреймворки, що забезпечують роботу системи. Розглянуто декілька середовищ розробки та реляційні системи керування базами даних.

3 МЕТОДИ ДЛЯ ВЗАЄМОДІЇ З СЕРВЕРОМ

Для звернення до серверу використовуються HTTP запити, для отримання веб-представлення використовуються GET запити, для відправки форм POST. Методи, що приймають POST запити захищені, для виконання запиту необхідно вказати токен, що служить захистом підробки запитів. Токен захисту відправляється з веб-представленням, на якому знаходиться форми відправлення даних.

3.1 Головна сторінка додатку

Цей ресурс дозволяє отримати головну сторінку додатку з інформацією про сервіс.

Формат запиту

```
GET /Home/Index HTTP/1.1
Host: example.com
```

Відповідь

У відповідь сервер повертає веб-представлення головної сторінки.

3.2 Авторизація та реєстрація

Для реєстрації користувача необхідно відправити запит на ресурс, що поверне веб-представлення з формою з полями для реєстрації.

Формат запиту

```
GET /Account/Register HTTP/1.1
Host: example.com
```

Відповідь

У відповідь сервер повертає веб-представлення сторінки з формою для реєстрації.

Для реєстрації необхідно відправити запит з даними про користувача:

- а) Адреса електронної пошти;
- б) Ім'я;
- в) Прізвище;
- г) Пароль.

Для роботи потрібен токен захисту підробки запитів.

Формат запиту

```
POST /Account/Register HTTP/1.1
Host: example.com
Content-Type: application/x-www-form-urlencoded\r\n
Content-Length: 242\r\n
\r\n
__RequestVerificationToken=CzRv2w2E7qF9hL-
RmNLQiltUrBnPeREhDbvU8ir_bmGxGmghTZUfFVT1JmW43Tpyyg-
X8VFkgx2mnHjqGUmGvqLYEdlK7RSzJELSqC2pUooFb2S8oPbDoFvCCW79QqEacG7GcbnxDJv68kn
Hkprq1Q2
&FirstName=NewName
&LastName=NewName
&Email="example@ex.ex"
&Password=NewPass
```

Відповідь:

У відповідь сервер при успішному створенні нового облікового запису перенаправляє користувача на головну сторінку додатку.

Для авторизації користувача, необхідно перейти на сторінку з полями для входу в обліковий запис.

Формат запиту

```
GET /Account/Login HTTP/1.1
Host: example.com
```

Відповідь

У відповідь сервер повертає веб-представлення сторінки з формою для авторизації.

Для авторизації необхідно відправити запит з електронною адресою, та паролем користувача. Для роботи потрібен токен захисту підробки запитів.

Формат запиту

```
POST /Account/Login HTTP/1.1
Host: example.com
Content-Type: application/x-www-form-urlencoded\r\n
Content-Length: 242\r\n
\r\n
__RequestVerificationToken=CzRv2w2E7qF9hL-
RmNLQiltUrBnPeREhDbvU8ir_bmGxGmghTZUfFVT1JmW43Tpyyg-
X8VFkgx2mnHjqGUmGvqLYEdlK7RSzJELsqC2pUooFb2S8oPbDoFvCCW79QqEacG7GcbnxDJv68kn
Hkprq1Q2
&Email="example@ex.ex"
&Password=NewPass
```

Відповідь:

У відповідь сервер при успішній авторизації перенаправляє користувача на головну сторінку додатку.

3.3 Сторінка створення компанії

Цей ресурс дозволяє отримати сторінку створення нової компанії. Створити компанію може лише зареєстрований та авторизований користувач.

Формат запиту

```
GET /Work/CreateGroup HTTP/1.1
Host: example.com
```

Відповідь:

У відповідь сервер повертає веб-представлення сторінки з формою створення компанії.

Для створення компанії необхідно відправити запит на ресурс з назвою компанії. У базу вноситься інформація про нову компанію, користувачеві видається права лідера. Для роботи потрібен токен захисту підробки запитів.

Формат запиту

```
POST /Work/CreateGroup HTTP/1.1
Host: example.com
```

```

Content-Type: application/x-www-form-urlencoded\r\n
Content-Length: 242\r\n
\r\n
__RequestVerificationToken=CzRv2w2E7qF9hL-
RmNLQiltUrBnPeREhDbvU8ir_bmGxGmghTZUfFVT1JmW43Tpyyg-
X8VFkgx2mnHjqGUmgGvqLYEdlK7RSzJELSqC2pUooFb2S8oPbDoFvCCW79QqEacG7GcbnxDJv68kn
Hkpq1Q2
&CompanyName=NewCompany
&Id=27ba0e1b-0c6a-41a9-8283-a02ead315a20

```

Відповідь:

У відповідь при успішному створенні компанії сервер перенаправляє користувача на сторінку компанії.

3.4 Сторінка компанії

Цей ресурс дозволяє отримати головну сторінці компанії де розміщено список діалогів. Доступ до ресурсу є тільки у зареєстрованих та авторизованих користувачів, при наявності причетності до одної з компаній.

Формат запити

```

GET /Work/Index HTTP/1.1
Host: example.com

```

Відповідь:

У відповідь сервер повертає веб-представлення сторінки зі списком активних діалогів.

Для переходу до обраного діалогу необхідно за відправити запит до ресурсу з ідентифікаційним номером діалогу.

Формат запити

```

GET /Work/Dialog/55 HTTP/1.1
Host: example.com

```

Відповідь:

Сервер повертає веб-представлення з обраним діалогом, на ньому знаходиться список повідомлень та форма відправлення нового повідомлення.

Для оновлення списку діалогів та отримання повідомлень необхідно відправити запит з часовою міткою останнього повідомлення на веб-представленні. Цей ресурс працює з технологією Long Poling та відправляє відповідь при появі нового повідомлення, або через 10 секунд відповідь, що нових повідомлень немає. Для роботи потрібен токен захисту подробиць запитів.

Формат запиту

```
POST /Work/GetNewDialogs HTTP/1.1
Host: example.com
Content-Type: application/x-www-form-urlencoded\r\n
Content-Length: 242\r\n
\r\n
__RequestVerificationToken=CzRv2w2E7qF9hL-
RmNLQiltUrBnPeREhDbvU8ir_bmGxGmghTZUfFVT1JmW43Tpyyg-
X8VFkgx2mnHjqGUmGvqLYEdlK7RSzJELSqc2pUooFb2S8oPbDoFvCCW79QqEacG7GcbnxDJv68kn
Hkprq1Q2
&Time=17:04:43
```

Відповідь:

У відповідь сервер поверне часткове веб-представлення з оновленим списком діалогів.

3.5 Сторінка з діалогом

Для оновлення списку повідомлень у діалозі необхідно відправити запит з часовою міткою останнього повідомлення на веб-представленні. Цей ресурс працює з технологією Long Poling та відправляє відповідь при появі нового повідомлення, або через 10 секунд відповідь, що нових повідомлень немає. Для роботи потрібен токен захисту подробиць запитів.

Формат запиту

```
POST /Work/GetNewDialogMessage HTTP/1.1
Host: example.com
Content-Type: application/x-www-form-urlencoded\r\n
Content-Length: 242\r\n
\r\n
__RequestVerificationToken=CzRv2w2E7qF9hL-
RmNLQiltUrBnPeREhDbvU8ir_bmGxGmghTZUfFVT1JmW43Tpyyg-
X8VFkgx2mnHjqGUmGvqLYEdlK7RSzJELSqC2pUooFb2S8oPbDoFvCCW79QqEacG7GcbnxDJv68kn
Hkprq1Q2
&Time=17:04:43
```

Відповідь:

У відповідь сервер поверне часткове веб-представлення з оновленим списком повідомлень у діалозі.

Для відправлення нового повідомлення до співрозмовника необхідно відправити запит з даними про користувача, співрозмовника та текст повідомлення. Для роботи потрібен токен захисту підробки запитів.

Формат запиту

```
POST /Work/SendMessage HTTP/1.1
Host: example.com
Content-Type: application/x-www-form-
urlencoded\r\n
Content-Length: 242\r\n
\r\n
__RequestVerificationToken=CzRv2w2E7qF9hL-
RmNLQiltUrBnPeREhDbvU8ir_bmGxGmghTZUfFVT1JmW43Tpyyg-
X8VFkgx2mnHjqGUmGvqLYEdlK7RSzJELSqC2pUooFb2S8oPbDoFvCCW79QqEacG7GcbnxDJv68kn
Hkprq1Q2
&DialogId=235
&Id=27ba0e1b-0c6a-41a9-8283-a02ead315a20
&Text="An example of the message"
```

3.6 Сторінка управління компанії

При запиті цього ресурсу сервер поверне сторінку на якій ведеться налаштування ботів, та управління співробітниками.

Формат запиту

```
GET /Work/GroupManage HTTP/1.1
Host: example.com
```

Відповідь:

У відповідь сервер поверне веб-представлення з списком ботів, форми додання ботів “Вконтакті” та “Telegram”, списком співробітників та формою додання співробітника.

Для додання нового бота необхідно за відправити запит до ресурсу з даними про нового бота. Права на додання новго боту має лише лідер компанії та менеджер. Для роботи потрібен токен захисту підробки запитів.

Формат запиту додання нового бота “Вконтакті”

```
POST /Work/AddVkBot HTTP/1.1
Host: example.com
Content-Type: application/x-www-form-urlencoded\r\n
Content-Length: 242\r\n
\r\n
__RequestVerificationToken=CzRv2w2E7qF9hL-
RmNLQiltUrBnPeREhDbvU8ir_bmGxGmghTZUfFVT1JmW43Tpyyg-
X8VFkgx2mnHjqGUmGvqLYEdlK7RSzJELSqC2pUooFb2S8oPbDoFvCCW79QqEacG7GcbnxDJv68kn
Hkpq1Q2
&CompanyId=5
&Token=ongnsdfghFDdhFSfSDlhbhfjHSdDfshg54NFFsdfjdfhzbKZNfsdjf3DNfdfsfa
&BotName="NewVkBot"
&GroupId="234523524"
```

Формат запиту додання нового бота “Telegram”

```
POST /Work/AddTgBot HTTP/1.1
Host: example.com
Content-Type: application/x-www-form-
urlencoded\r\n
Content-Length: 242\r\n
\r\n
__RequestVerificationToken=CzRv2w2E7qF9hL-
RmNLQiltUrBnPeREhDbvU8ir_bmGxGmghTZUfFVT1JmW43Tpyyg-
X8VFkgx2mnHjqGUmGvqLYEdlK7RSzJELSqC2pUooFb2S8oPbDoFvCCW79QqEacG7GcbnxDJv68kn
Hkpq1Q2
```



```
&CompanyId=5
&Token=ongnsdfghFDdhFSfSDlhbhfjHSdDfshg54NFFsdfjdfhzbKZNfsdjf3DNfdsfa
&BotName="NewTgBot"
```

Відповідь:

У відповідь сервер поверне часткове веб-представлення з оновленим списком ботів.

Для видалення бота необхідно за відправити запит до ресурсу з даними про бота, права на видалення боту має лише лідер компанії та менеджер. Для роботи потрібен токен захисту підробки запитів.

Формат запиту

```
POST /Work/DeleteBot HTTP/1.1
Host: example.com
Content-Type: application/x-www-form-urlencoded\r\n
Content-Length: 242\r\n
\r\n
__RequestVerificationToken=CzRv2w2E7qF9hL-
RmNLQiltUrBnPeREhDbvU8ir_bmGxGmghTZUfFVT1JmW43Tpyyg-
X8VFkgx2mnHjqGUmgGvqLYEdlK7RSzJELSqC2pUooFb2S8oPbDoFvCCW79QqEacG7GcbnxDJv68kn
Hkpq1Q2
&CompanyId=5
&BotId="52"
```

Відповідь:

У відповідь сервер поверне часткове веб-представлення з оновленим списком ботів.

Для додання співробітника до компанії необхідно відправити запит до ресурсу з електронною адресою, що вказував співробітник при реєстрації. Додати співробітника може лише лідер компанії та менеджер. Для роботи потрібен токен захисту підробки запитів.

Формат запиту

```
POST /Work/AddUser HTTP/1.1
Host: example.com
Content-Type: application/x-www-form-urlencoded\r\n
Content-Length: 242\r\n
```

```

\r\n
__RequestVerificationToken=CzRv2w2E7qF9hL-
RmNLQiltUrBnPeREhDbvU8ir_bmGxGmghTZUfFVT1JmW43Tpyyg-
X8VFkgx2mnHjqGUmGvqLYEdlK7RSzJELsqC2pUooFb2S8oPbDoFvCCW79QqEacG7GcbnxDJv68kn
Hkpq1Q2
&Email="example@ex.ex"

```

Відповідь:

У відповідь сервер поверне часткове веб-представлення з оновленим списком користувачів.

Для видачі та зняття співробітнику прав менеджера необхідно відправити запит за даними про співробітника. Можливість видачі та зняття прав менеджера є лише у лідера. Для роботи потрібен токен захисту підробки запитів.

Формат запиту

```

POST /Work/AddRoleUser HTTP/1.1
Host: example.com
Content-Type: application/x-www-form-urlencoded\r\n
Content-Length: 242\r\n
\r\n
__RequestVerificationToken=CzRv2w2E7qF9hL-
RmNLQiltUrBnPeREhDbvU8ir_bmGxGmghTZUfFVT1JmW43Tpyyg-
X8VFkgx2mnHjqGUmGvqLYEdlK7RSzJELsqC2pUooFb2S8oPbDoFvCCW79QqEacG7GcbnxDJv68kn
Hkpq1Q2
&Id=27ba0e1b-0c6a-41a9-8283-a02ead315a20

```

Відповідь:

У відповідь сервер поверне часткове веб-представлення з оновленим списком користувачів.

Для видалення співробітника з компанії необхідно відправити запит до ресурсу з інформацією про співробітника, видалити співробітника може лише лідер. Для роботи потрібен токен захисту підробки запитів.

Формат запиту

```

POST /Work/DeleteUser HTTP/1.1
Host: example.com

```

```
Content-Type: application/x-www-form-urlencoded\r\n
Content-Length: 242\r\n
\r\n
__RequestVerificationToken=CzRv2w2E7qF9hL-
RmNLQiltUrBnPeREhDbvU8ir_bmGxGmghTZUfFVT1JmW43Tpyyg-
X8VFkgx2mnHjqGUmGvqLYEdlK7RSzJELSqC2pUooFb2S8oPbDoFvCCW79QqEacG7GcbnxDJv68kn
Hkprq1Q2
&Id=27ba0e1b-0c6a-41a9-8283-a02ead315a20
```

Відповідь:

У відповідь сервер поверне часткове веб-представлення з оновленим списком користувачів.

Висновки до розділу 3

У розділі було сформовано та описано методи для взаємодії з системою, задокументовано формат запитів, описано відповідні дії сервера на запит.

Описано функції взаємодії з веб-представленнями. Описані функції взаємодії с системою авторизації, роботу з методами створення та управління підприємства, описані методи обміну повідомленнями.

4 ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ

В даному розділі було розглянуто загальні питання з охорони праці. Були проаналізовані умови праці, вимоги до приміщення та організації. Розглянуті заходи, які дозволяють забезпечити гігієну праці та виробничу санітарію. Також були розглянуті рекомендації щодо пожежної безпеки, електробезпеки, мікроклімату та освітлення.

4.1 Загальні питання з охорони праці

Згідно з законом “Про охорону праці” [22] охорона праці це – система правових, соціально-економічних, організаційно-технічних, санітарно-гігієнічних і лікувально-профілактичних заходів та засобів, спрямованих на збереження життя, здоров'я і працездатності людини у процесі трудової діяльності.

При роботі з обчислювальною технікою змінюються фізичні і хімічні фактори навколишнього середовища: виникає статична електрика, електромагнітне випромінювання, змінюється температура і вологість, рівень вміст кисню і озону в повітрі. Забезпечення цих умов покладається на власника або уповноважений ним орган (далі роботодавець). Умови праці на робочому місці, безпека технологічних процесів, машин, механізмів, устаткування та інших засобів виробництва, стан засобів колективного та індивідуального захисту, що використовуються працівником, а також санітарно-побутові умови повинні відповідати вимогам нормативних актів про охорону праці.

4.2 Аналіз стану та умов праці

Для роботи над створенням мобільного додатку “Cash Eater” достатньо однієї людини для якої надано робоче місце ці стаціонарним комп'ютером.

4.2.1 Вимоги до приміщення

Геометричні розміри приміщення зазначені в табл. 4.1.

Таблиця 4.1 – Розміри приміщення

Найменування	Значення
Довжина, м	5
Ширина, м	4
Висота, м	3
Площа, м ²	20
Об'єм, м ³	60

Згідно з ДСН 3.3.6.042-99 “Санітарні норми мікроклімату виробничих приміщень” [23] розмір площі для одного робочого місця оператора персонального комп'ютера має бути не менше 6 кв. м, а об'єм — не менше 20 куб. м. Отже, дане приміщення цілком відповідає зазначеним нормам.

Також для дотримання визначеного рівня мікроклімату в будівлі встановлено систему опалення та кондиціонування.

Для забезпечення потрібного рівного освітленості кімната має вікно та систему загального рівномірного освітлення, що встановлена на стелі. Для дотримання вимог пожежної безпеки встановлено порошковий вогнегасник та систему автоматичної пожежної сигналізації.

4.2.2 Вимоги до організації робочого місця

При порівнянні відповідності характеристик робочого місця нормативним основні вимоги до організації робочого місця за ДСанПіН 3.3.2.007-98 «Правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин» [24] і відповідними

фактичними значеннями для робочого місця, констатуємо повну відповідність в таблиці 4.2.

Таблиця 4.2 - Характеристики робочого місця

Найменування параметра	Фактичне значення	Нормативне значення
Висота робочої поверхні, мм	700	680 ÷ 800
Висота простору для ніг, мм	680	не менше 600
Ширина простору для ніг, мм	550	не менше 500
Глибина простору для ніг, мм	700	не менше 650
Висота поверхні сидіння, мм	500	400 ÷ 500
Ширина сидіння, мм	400	не менше 400
Глибина сидіння, мм	400	не менше 400
Висота поверхні спинки, мм	600	не менше 300
Ширина опорної поверхні спинки, мм	400	не менше 380
Радіус кривини спинки в горизонтальній площині, мм	400	400
Відстань від очей до екрану дисплея, мм	800	700 ÷ 800

Приміщення кабінету знаходиться у вчасній одноповерховій будівлі і має об'єм 54 м³, площу — 18 м².

Температура в приміщенні протягом року коливається у межах 18–24°C, відносна вологість — близько 50%. Система вентилявання приміщення — природна неорганізована, а опалення — централізоване.

Розміщення вікон забезпечує природне освітлення з коефіцієнтом природного освітлення не менше 1,5%, а загальне штучне освітлення, яке здійснюється за допомогою однієї люмінесцентної лампи, забезпечує рівень освітленості не менше 200 Лк

За ступенем пожежної безпеки приміщення належить до категорії В

4.2.3 Навантаження та напруженість процесу праці

Під час виконання робіт використовують ПК та периферійні пристрої (лазерні та струменеві), що призводить до навантаження на окремі системи організму. Такі перекося у напруженні різних систем організму, що трапляються під час роботи з ПК, зокрема, значна напруженість зорового аналізатора і довготривале малорухоме положення перед екраном, не тільки не зменшують загального напруження, а навпаки, призводять до його посилення і появи стресових реакцій.

Найбільшому ризику виникнення різноманітних порушень піддаються: органи зору, м'язово-скелетна система, нервово-психічна діяльність, репродуктивна функція у жінок.

Тобто наявні психофізіологічні небезпечні та шкідливі фактори:

а) фізичного перевантаження:

- статичного;
- динамічного;

б) нервово-психічного перевантаження:

- розумового перенапруження;
- монотонності праці;
- перенапруження аналізаторів;
- емоційних перевантажень.

Роботу за дипломним проектом визнано, таку, що займає 50% часу робочого дня та за восьмигодинної робочої зміни рекомендовано встановити додаткові регламентовані перерви тривалістю 15 хв через кожен годину роботи.

4.3 Виробнича санітарія

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

4.3.1 Загальні заходи безпеки

Відповідно до санітарно-гігієнічних нормативів та правил експлуатації обладнання приклад деяких заходів безпеки:

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

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

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

- постійно стежити за справним станом електромережі, розподільних щитків, вимикачів, штепсельних розеток, лампових патронів, а також мережевих кабелів живлення, за допомогою яких електроприлади під'єднують до електромережі;

- постійно стежити за справністю ізоляції електромережі та мережевих кабелів, не допускаючи їхньої експлуатації з пошкодженою ізоляцією;
- не тягнути за мережевий кабель, щоб витягти вилку з розетки;
- не закривати меблями, різноманітним інвентарем вимикачі, штепсельні розетки;
- не підключа4ти одночасно декілька потужних електропристроїв до однієї розетки, що може викликати надмірне нагрівання провідників, руйнування їхньої ізоляції, розплавлення і загоряння полімерних матеріалів;
- не залишати включені електроприлади без нагляду;
- не допускати потрапляння всередину електроприладів крізь вентиляційні отвори рідин або металевих предметів, а також не закривати їх та підтримувати в належній чистоті, щоб уникнути перегрівання та займання приладу;
- не ставити на електроприлади матеріали, які можуть під дією теплоти, що виділяється, загорітися (канцелярські товари, сувенірну продукцію тощо).

4.3.2 Електробезпека

Основним небезпечним фактором при роботі з ЕОМ є безпека ураження людини електричним струмом, яка посилюється тим, що органи чуття людини не можуть на відстані виявити наявність електричної напруги на обладнанні.

Проходячи через тіло людини, електричний струм чинить на нього складний вплив, що є сукупністю термічної (нагрів тканин і біологічних середовищ), електролітичної (розкладання крові і плазми) і біологічної (роздратування і збудження нервових волокон та інших органів тканин організму) дій.

Тяжкість ураження людини електричним струмом залежить від цілого ряду чинників:

- 1) значення сили струму;
- 2) електричного опору тіла людини і тривалості протікання через нього струму;
- 3) типу і частоти струму;
- 4) індивідуальних властивостей людини і навколишнього середовища.

Приміщення для ЕОМ відноситься до приміщень без підвищеної небезпеки, тобто в приміщення, в яких відсутні умови, що створюють підвищену або особливу небезпеку. Небезпека ураження електричним струмом існує всюди, де використовуються електроустановки, тому приміщення без підвищеної небезпеки не можна назвати безпечними.

Електробезпека забезпечується:

- 1) відповідною конструкцією електроустановок;
- 2) застосуванням технічних способів і засобів захисту;
- 3) організаційними і технічними заходами.

Конструкція електроустановок відповідає умовам їх експлуатації та забезпечує захист персоналу від дотику до струмоведучих частин.

Основними технічними способами і засобами захисту від ураження електричним струмом, що використовуються окремо або в поєднанні один з одним, є:

- 1) захисне заземлення;
- 2) занулення;
- 3) вирівнювання потенціалів;
- 4) мале напруга;
- 5) електричне поділ мереж;
- 6) захисне відключення;
- 7) ізоляція струмоведучих частин;
- 8) компенсація струмів замикання на землю;
- 9) захисні пристрої;

- 10) попереджувальна сигналізація, блокування, знаки безпеки;
- 11) ізолюючі захисні та запобіжні пристосування.

4.3.3 Мікроклімат

Мікроклімат робочих приміщень – це клімат внутрішнього середовища цих приміщень, що визначається діючої на організм людини з'єднанням температури, вологості, швидкості переміщення повітря. В даному приміщенні проводяться роботи, що виконуються сидячи і не потребують динамічного фізичного напруження, то для нього відповідає категорія робіт Іа. Отже оптимальні значення для температури, відносної вологості й рухливості повітря для зазначеного робочого місця відповідають ДСН 3.3.6.042-99 «Санітарні норми мікроклімату виробничих приміщень» [23] і наведені в таблиці 4.3.

Таблиця 4.3 – Норми мікроклімату робочої зони об'єкту

Період року	Температура, °С	Відносна вологість, %	Швидкість вітру, м/с, не більше
Холодний	21 - 23	60 – 40	0.1
Теплий	22 - 24	60 - 40	0.2

4.3.4 Освітлення

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

Збільшення освітленості сприяє поліпшенню працездатності навіть в тих випадках, коли процес праці практично не залежить від зорового сприйняття. При поганому освітленні людина швидко втомлюється, працює менш продуктивна, виникає потенційна небезпека помилкових дій і нещасних випадків.

У проекті, що розробляється, передбачається використовувати суміщене освітлення. У світлий час доби використовуватиметься природне освітлення приміщення через віконні отвори, в решту часу використовуватиметься штучне освітлення.

Штучне освітлення в робочому приміщенні передбачається здійснювати з використанням люмінесцентних джерел світла у світильниках загального освітлення, оскільки люмінесцентні лампи мають високу потужність (80 Вт), тривалий термін служби (до 10000 годин), спектральний складом випромінюваного світла, близький до сонячного. При експлуатації ПК виконується зорова робота IV в розряді точності (середня точність). При цьому нормована освітленість на робочому місці (Ен) рівна 200 лк. Джерелом природного освітлення є сонячне світло. У приміщенні, де розташовані ЕОМ передбачається природне бічне освітлення, рівень якого відповідає ДБН В.2.5-28:2018 Природне і штучне освітлення [25]

Регулярно повинен проводитися контроль освітленості, який підтверджує, що рівень освітленості задовольняє [25]

Розрахунок освітлення.

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

$$S_b = \left(\frac{1}{5} \div \frac{1}{10} \right) \cdot S_n \quad (4.9)$$

де S_b – площа віконних прорізів, м²;

S_n – площа підлоги, м².

$$S_n = a \times b = 3 \times 6 = 20 \text{ м}^2 ,$$

$$S = 1/8 \times 20 = 2.5 \text{ м}^2 .$$

Приймаємо 1 вікно площею $S = 2.5 \text{ м}^2$. Світильники загального освітлення розташовуються над робочими поверхнями в рівномірно-прямокутному порядку. Для організації освітлення в темний час доби

передбачається обладнати приміщення, довжина якого складає 4м, ширина 4м, світильниками ЛПО2П, оснащеними лампами типа ЛБ (дві по 80Вт) з світловим потоком 5400лм кожна. Розрахунок штучного освітлення виробляється по коефіцієнтах використання світлового потоку, яким визначається потік, необхідний для створення заданої освітленості при загальному рівномірному освітленні.

Розрахунок кількості світильників n визначається по формулі (4.10):

$$n = \frac{E \times S \times Z \times K}{F \times U \times M} \quad (4.10)$$

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

S – освітлювана площа, м²; $S = 20$ м²;

Z – поправочний коефіцієнт світильника ($Z = 1.15$ для ламп розжарювання та ДРЛ;

$Z = 1,1$ для люмінесцентних ламп) приймаємо рівним 1.1;

K – коефіцієнт запасу, що враховує зниження освітленості в процесі експлуатації – 1.5;

U – коефіцієнт використання, залежний від типу світильника, показника індексу приміщення і т.п. – 0.575;

M – число люмінесцентних ламп в світильнику – 2;

F – світловий потік лампи – 5400лм (для ЛБ-80). Підставивши числові значення у формулу (4.10), отримуємо:

$$n = \frac{300 \times 20 \times 1.1 \times 1.5}{5400 \times 0.575 \times 2} \approx 1,59$$

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

4.3.5 Рекомендації щодо пожежної безпеки

Пожежна безпека при застосуванні ПК забезпечується:

- системою запобігання пожежі,
- системою протипожежного захисту,
- організаційно-технічними заходами.

Згідно ДСТУ Б В.1.1-36:2016 [26] таке приміщення, площею 25 м², відноситься до категорії "В" (пожежонебезпечної) та для протипожежного захисту в ньому проектом передбачено устаткування автоматичною пожежною сигналізацією із застосуванням датчиків-сповіщувачів РІД-1 (сповіщувач димовий ізоляційний) в кількості 1 шт., і застосуванням первинних засобів пожежогасіння. Відповідно до норм первинних засобів пожежогасіння пропонується використовувати:

- ручний вуглекислий вогнегасник ОУ-5 в кількості 1 шт;
- ковдра 1 м², кошму 2×1,5 м² або азбестове полотно 2×2 м² в кількості 1 шт.

Виникнення пожежі можливе, якщо на об'єкті є горючі речовини, окислювач і джерела запалювання. Вірогідність пожежної небезпеки приймається значною, якщо ймовірна взаємодія цих трьох чинників. Горючими компонентами є: будівельні матеріали для акустичної і естетичної обробки приміщень, перегородки, підлоги, двері, ізоляція силових, сигнальних кабелів і т.д.

Горючими матеріалами в приміщенні, де розташовані ПК, є:

- поліамід – матеріал корпусу мікросхем, горюча речовина, температура самозаймання 420 °С,
- полівінілхлорид – ізоляційний матеріал, горюча речовина, температура запалювання 335 °С, температура самозаймання 530 °С,
- склотекстоліт ДЦ – матеріал друкарських плат, важкогорючий матеріал, показник горючості 1.74, не схильний до температурного самозаймання,

- пластикат кабельний №.489 – матеріал ізоляції кабелів, горючий матеріал, показник горючості більше 2.1,
- деревина – будівельний і обробний матеріал, з якого виготовлені меблі, горючий матеріал, показник горючості більше 2.1, температура запалювання 255 °С, температура самозаймання 399 °С.

Для відводу теплоти від ПК діє система кондиціонування. Тому кисень, як окиснювач процесів горіння, є в будь-якій точці приміщень ВЦ.

Простори усередині приміщень в межах, яких можуть утворюватися або знаходиться пожежонебезпечні речовини і матеріали відповідно до [26] відносяться до пожежонебезпечної зони класу П-Па. Це обумовлено тим, що в приміщенні знаходяться тверді горючі та важко займисті речовини та матеріали. Приміщенню, у якому розташоване робоче місце, присвоюється II ступень вогнестійкості.

Потенційними джерелами запалювання можуть бути:

- іскри і дуги короткого замикання;
- електрична іскра при замиканні і розмиканні ланцюгів;
- перегріву від тривалого перевантаження,
- відкритий вогонь і продукти горіння,
- наявність речовин, нагрітих вище за температуру самозаймання,
- розрядна статична електрика.

Причинами можливого загоряння і пожежі можуть бути:

- несправність електроустановки;
- конструктивні недоліки устаткування;
- коротке замикання в електричних мережах;
- запалювання горючих матеріалів, що знаходяться в безпосередній близькості від електроустановки.

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

хлористий водень, фосген, синильна кислота, аміак, фенол, ацетон, стирол. (ГОСТ 12.1.044-89) [27].

Для захисту персоналу від дії небезпечних і шкідливих чинників пожежі проектом передбачається застосування промислового протигаза, що фільтрує, з коробкою марки «В» із сірою відміткою забарвлення – захист від неорганічних газів (хлор, фтор, бром, сірководень, сірковуглець, хлорциан, галогени), а цей фільтр не захистить від СО (тобто від чадного газу).

Можливе також відповідне застосування фільтрувальної коробки з маркуванням «СО» із фіолетовим забарвленням на фільтрі означає, що він захищає від Чадного газу. Або фільтру для протигазу з літерним маркуванням «SX» із фіолетовим забарвленням захистить від спец речовин таких як (зарин, зоман та фосген).

Висновок до розділу 4

У розділі "Охорона праці" виконаний аналіз потенційних небезпек при роботі із засобами обчислювальної техніки. Приведені рекомендації щодо організації робочого місця, електробезпеки та пожежної безпеки. Наведені розміри приміщення та значення температури, вологості й рухливості повітря, необхідна кількість і потужність ламп та інші параметри, значення яких впливає на умови праці, рекомендації з охорони праці, техніки безпеки при роботі на комп'ютері.

ВИСНОВКИ

Метою даної роботи було створення Web-системи технічної підтримки клієнтів за допомогою соціальних мереж.

У першому розділі було розглянуто предметну область, було сформовано призначення розробки системи, функціональні характеристики, вимоги до програмного та апаратного забезпечення.

У другому розділі було розглянуто архітектуру системи, описані основні компоненти системи, основні дані системи та розглянуто проектування бази даних. Описано основні моменти взаємодії веб-клієнта з системою. Коротко описано основні фреймворки, використані у проекті та розглянуто середовища розробки, реляційні системи керування базами даних.

У третьому розділі було сформовано та описано методи для взаємодії з системою, задокументовано формат запитів, описано відповідні дії сервера на запит.

Таким чином у результаті проекту було створено серверну частину системи технічної підтримки через соціальні мережі.

У четвертому розділі було визначено параметри і характеристики приміщення для роботи над проектом, заходи, які потрібно зробити для того, щоб дане приміщення відповідало необхідним нормам і було комфортним і безпечним для роботи. Приведені рекомендації щодо організації робочого місця, електробезпеки та пожежної безпеки. Проаналізовано основні навантаження та напруженості у процесу праці над проектом.

На даному етапі система підтримує лише підтримку через соціальну мережу “Вконтакте” та месенджер “Telegram”. Покращення роботи реалізованих функцій та розширення списку підтримуваних месенджерів буде проводитись після запуску системи в експлуатацію.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Software as a service [Електронний ресурс] // Вікіпедія: вільна енциклопедія. URL: https://uk.wikipedia.org/wiki/Програмне_забезпечення_як_послуга (дата звернення: 4.06.2019).
2. Веб-додаток [Електронний ресурс] // Вікіпедія: вільна енциклопедія. URL: <https://uk.wikipedia.org/wiki/Веб-застосунок> (дата звернення: 3.06.2019).
3. Соціальна мережа [Електронний ресурс] // Вікіпедія: вільна енциклопедія. URL: https://uk.wikipedia.org/wiki/Соціальна_мережа (дата звернення: 1.06.2019).
4. Месенджер [Електронний ресурс] // Вікіпедія: вільна енциклопедія. URL: https://uk.wikipedia.org/wiki/Миттєві_повідомлення (дата звернення: 2.06.2019).
5. API [Електронний ресурс] // Вікіпедія: вільна енциклопедія. URL: <https://ru.wikipedia.org/wiki/API> (дата звернення: 2.06.2019).
6. Open Ticket Request System [Електронний ресурс] // Вікіпедія: вільна енциклопедія. URL: <https://otrs.com> (дата звернення: 4.06.2019).
7. osTicket [Електронний ресурс] // Вікіпедія: вільна енциклопедія. URL: <https://osticket.com> (дата звернення: 5.06.2019).
8. Zendesk [Електронний ресурс] // Вікіпедія: вільна енциклопедія. URL: <https://www.zendesk.com> (дата звернення: 3.06.2019).
9. СУБД [Електронний ресурс] // Вікіпедія: вільна енциклопедія. URL: https://uk.wikipedia.org/wiki/Система_керування_базами_даних (дата звернення: 2.06.2019).
10. ASP.NET MVC 5 [Електронний ресурс] // Metanit.com: сайт про програмування. URL: <https://metanit.com/sharp/mvc5/> (дата звернення: 1.06.2019).

11. ER-модель [Електронний ресурс] // Вікіпедія: вільна енциклопедія.
URL: https://uk.wikipedia.org/wiki/Модель_«сутність_—_зв%27язок»
(дата звернення: 3.06.2019).
12. Entity Framework [Електронний ресурс] // Metanit.com: сайт про програмування. URL:
<https://metanit.com/sharp/entityframework/1.1.php> (дата звернення: 6.06.2019).
13. REST [Електронний ресурс] // Хабр: корпоративний блог. URL:
<https://habr.com/ru/post/38730/> (дата звернення: 6.06.2019).
14. Long Polling [Електронний ресурс] // Вікіпедія: вільна енциклопедія.
URL: <https://server-gu.ru/http-long-polling/> (дата звернення: 4.06.2019).
15. ASP.NET Identity [Електронний ресурс] // Metanit.com: сайт про програмування. URL: <https://metanit.com/sharp/mvc5/12.1.php> (дата звернення: 3.06.2019).
16. Visual Studio [Електронний ресурс] // Вікіпедія: вільна енциклопедія. URL:
https://ru.wikipedia.org/wiki/Microsoft_Visual_Studio (дата звернення: 5.06.2019).
17. JetBrains Rider [Електронний ресурс] // JetBrains :ровідний світовий виробник професійних засобів розробки. URL:
<https://jetbrains.ru/products/rider/> (дата звернення: 3.06.2019).
18. Eclipse [Електронний ресурс] // Вікіпедія: вільна енциклопедія.
URL: [https://ru.wikipedia.org/wiki/Eclipse_\(среда_разработки\)](https://ru.wikipedia.org/wiki/Eclipse_(среда_разработки)) (дата звернення: 5.06.2019).
19. MySQL [Електронний ресурс] // Хабр: корпоративний блог. URL:
<https://habr.com/ru/hub/mysql/> (дата звернення: 2.06.2019).
20. Microsoft SQL Server [Електронний ресурс] // Вікіпедія: вільна енциклопедія. URL:

- https://ru.wikipedia.org/wiki/Microsoft_SQL_Server (дата звернення: 1.06.2019).
21. PostgreSQL [Електронний ресурс] // PostgreSQL: The world's most advanced open source database. URL: <https://www.postgresql.org/about/> (дата звернення: 1.06.2019).
22. Закон України «Про охорону праці». [Електронний ресурс] // Законодавство України. URL: <https://zakon.rada.gov.ua/laws/show/2694-12> (дата звернення 10.06.2019)
23. Санітарні норми мікроклімату виробничих приміщень ДСН 3.3.6.042-99. // Законодавство України. URL: [Електронний ресурс] <https://zakon.rada.gov.ua/rada/show/va042282-99> (дата звернення 10.06.2019)
24. Державні санітарні правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин ДСанПН 3.3.2.007-98. // Законодавство України. URL: [Електронний ресурс] <https://zakon.rada.gov.ua/rada/show/v0007282-98> (дата звернення 10.06.2019)
25. Природне і штучне освітлення ДБН В.2.5-28:2018. [Електронний ресурс] https://dbn.co.ua/load/normativy/dbn/dbn_v_2_5_28/1-1-0-1188 (дата звернення 10.06.2019)
26. Визначення категорій приміщень, будинків, установок за вибухопожежною та пожежною небезпекою ДСТУ Б В.1.1-36:2016 2018. . // Державні будівельні норми. URL: [Електронний ресурс] https://dbn.co.ua/load/normativy/dstu/dstu_b_v_1_1_36/5-1-0-1759 (дата звернення 10.06.2019)
27. ГОСТ 12.1.044-89:2016 2018. . // Кодекс. URL: [Електронний ресурс] <http://docs.cntd.ru/document/gost-12-1-044-89> (дата звернення 10.06.2019)

ДОДАТОК А ПРЕЗЕНТАЦІЯ

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
СХІДНОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМ. В. ДАЛЯ

Комплексна тема.
Система технічної підтримки клієнтів за
допомогою соціальних мереж:
БД і серверна частина

Студент групи КІ-15д
Лунякін Р.О.

Керівник проекту
Скарга-Бандурова І.С.

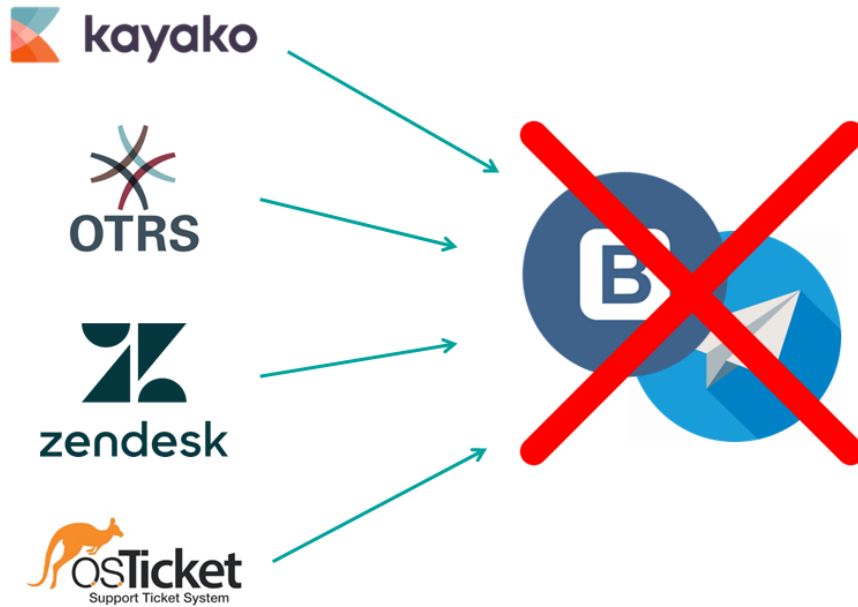
Мета роботи



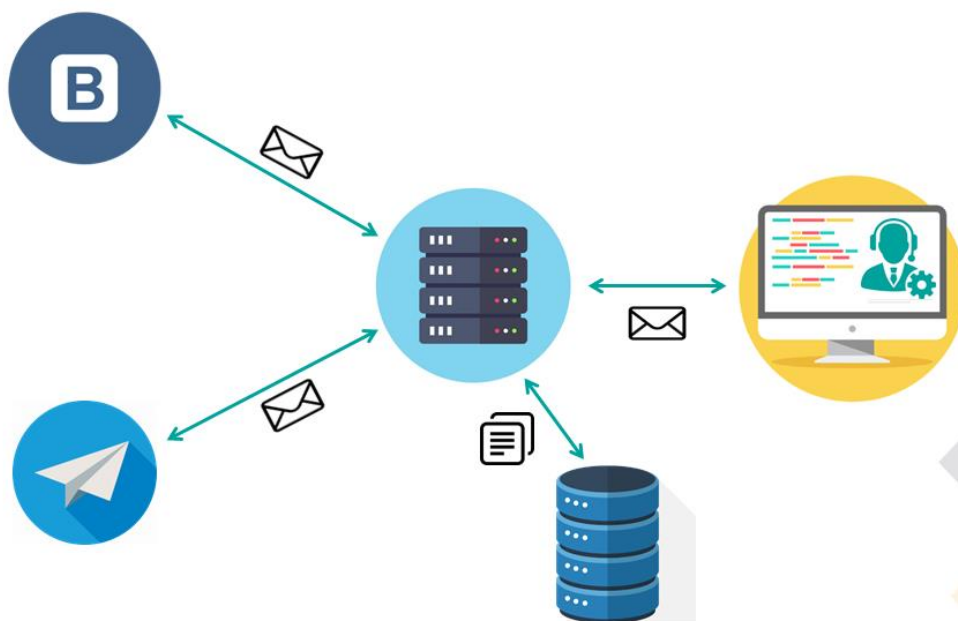
Розробка системи для технічної підтримки
через соціальні мережі



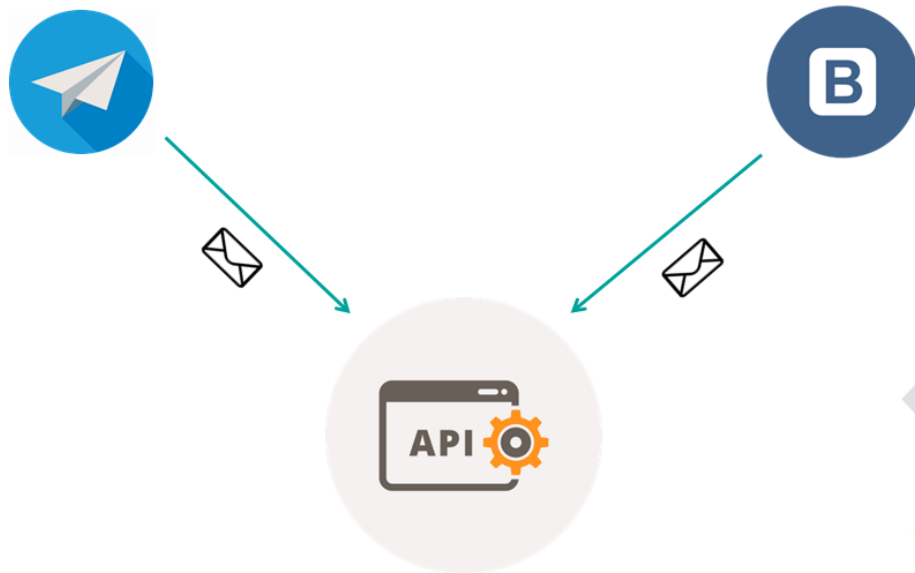
Існуючі рішення



Архітектура системи



Робота чат-ботів



Взаємодія з веб-клієнтом



Додання нового боту



HTTP



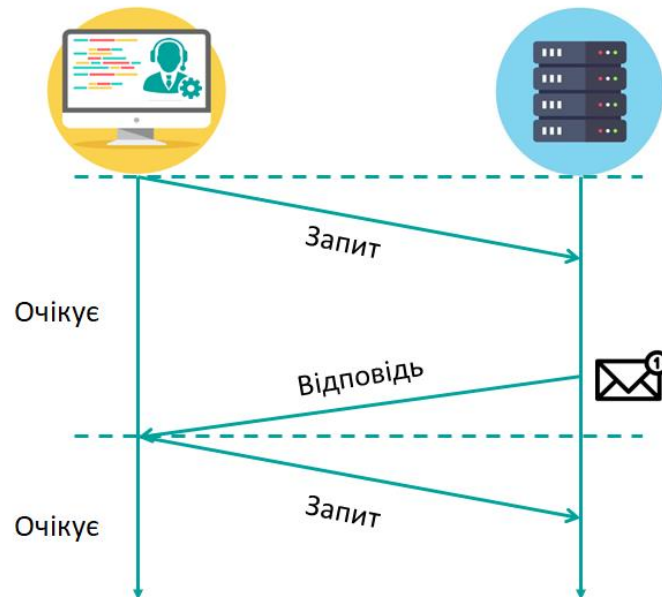
POST /Work/AddVkBot HTTP/1.1

CompanyId = 26
Token = "NewToken"
BotName = "NewVkBot"
GroupId = "234523524"

Технології

- ✓ Обмін повідомленнями з веб-клієнтом: **Http Long Polling**
- ✓ Система автентифікації: **Asp.NET Identity**
- ✓ Взаємодія з БД: **Entity framework**

Long Poilling



Майбутнє проекту

- Оптимізація роботи серверу
- Розширення списку соціальних мереж та месенджерів

ДОДАТОК Б ЛІСТИНГ ФУНКЦІЇ ДОДАННЯ НОВОГО БОТУ “ВКОНТАКТЕ”

```
[HttpPost]
[ValidateAntiForgeryToken]
public ActionResult AddNewVkBot(CreateVkBotModel form) {
    if (dbContext.VkBots.FirstOrDefault(a =>a.Token == form.Token) == null)
    {
        var company = dbContext.Companies.FirstOrDefault(a =>a.Id ==
form.IdCompany);

        var vkApi = new VkApi();

        vkApi.Authorize(new ApiAuthParams {
            AccessToken = form.Token
        });
        var server_id = vkApi.Groups.AddCallbackServer(form.GroupId,
"http://www.tinyhelper.somee.com/api/vkapi/update", company.CompanyName,
"secretKey8841");
        dbContext.VkBots.Add(new VkBot() {
            Token = form.Token,
            GroupId = form.GroupId,
            IdCompany = form.IdCompany,
            socialName = SocialName.Vkontakte,
            BotName = form.BotName,
            ApiPassword = "secretKey8841",
            ServerId = server_id
        });
        dbContext.SaveChanges();
        bool setconfig = vkApi.Groups.SetCallbackSettings(new
CallbackServerParams() {
            GroupId = form.GroupId,
            ServerId = server_id,
            CallbackSettings = new CallbackSettings() {
                MessageNew = true
            }
        });
        var listVk = dbContext.VkBots.Where(a =>a.IdCompany ==
form.IdCompany).ToList();
        var listTg = dbContext.TgBots.Where(a =>a.IdCompany ==
form.IdCompany).ToList();
```

```

List < BotPartial > allBot = new List < BotPartial > ();
foreach(var item in listVk) {
    allBot.Add(item);
}
foreach(var item in listTg) {
    allBot.Add(item);
}
ViewBag.Company = company;
ViewBag.allBot = allBot;

var currentUserId = User.Identity.GetUserId();
var currentUser = dbContext.Users.FirstOrDefault(a =>a.Id ==
currentUserId);

var group_leader_role_id = dbContext.Roles.FirstOrDefault(a =>a.Name == "group_leader").Id;
var group_manager_role_id = dbContext.Roles.FirstOrDefault(a =>a.Name == "group_manager").Id;

if (currentUser.Roles.FirstOrDefault(a =>a.RoleId ==
group_leader_role_id) != null) {
    ViewBag.RoleUserLeader = true;
    ViewBag.RoleUserManager = false;
    ViewBag.RoleUser = false;
}
else if (currentUser.Roles.FirstOrDefault(a =>a.RoleId ==
group_manager_role_id) != null) {
    ViewBag.RoleUserLeader = false;
    ViewBag.RoleUserManager = true;
    ViewBag.RoleUser = false;
}
else {
    ViewBag.RoleUserLeader = false;
    ViewBag.RoleUserManager = false;
    ViewBag.RoleUser = true;
}
return PartialView("BotList", new BotListModel() {
    list = allBot
});
}

return HttpNotFound();
}

```