

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ  
Кафедра комп'ютеризованих систем управління**

ДОПУСТИТИ ДО ЗАХИСТУ

Завідувач кафедри

Литвиненко О.Є.

“\_\_” \_\_\_\_\_ 2022 р.

**ДИПЛОМНИЙ ПРОЄКТ  
(ПОЯСНЮВАЛЬНА ЗАПИСКА)**

**ЗДОБУВАЧА ОСВІТНЬОГО СТУПЕНЯ  
“БАКАЛАВР”**

**Тема:** “Програмний модуль організації спілкування між користувачами соціальних мереж”

**Виконавець:** Клопенко Роман Юрійович

**Керівник:** доцент, Марченко Надія Борисівна

**Нормоконтролер:** Тупота Євгеній  
Вікторович

**Київ 2022**

# НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ

Факультет кібербезпеки, комп'ютерної та програмної інженерії

Кафедра комп'ютеризованих систем управління

Спеціальність 126 «Інформаційні системи та технології»

(шифр, найменування)

Освітньо-професійна програма «Інформаційні системи та технології»

Форма навчання денна

ЗАТВЕРДЖУЮ

Завідувач кафедри

Литвиненко О. Є.

«\_\_»\_\_\_\_\_2022 р.

## ЗАВДАННЯ

### на виконання дипломної роботи (проєкту)

Клопенка Романа Юрійовича

(прізвище, ім'я, по батькові випускника в родовому відмінку)

1. Тема дипломної роботи (проєкту): «Програмний модуль організації спілкування між користувачами соціальних мереж»

затверджена наказом ректора від «15» лютого 2022 р. № 251/ст

2. Термін виконання роботи (проєкту): з 16.05.2022 р. по 19.06.2022 р.

3. Вихідні дані до роботи (проєкту): Програмний модуль організації спілкування між користувачами соціальних мереж

4. Зміст пояснювальної записки:

1. Розгляд відомих аналогів та необхідності їх створення.

2. Класифікація рішень до програмного модулю організації спілкування

3. Створення прототипу програмного модулю.

4. Тестування прототипу програмного модулю.

5. Перелік обов'язкового графічного (ілюстративного) матеріалу:

1. Діаграми варіантів використання
2. Технічна структура модулів
3. Діаграма класів модуля.
4. Регулювання запитів тестування

## 6. Календарний план-графік

№ пор.	Завдання	Термін виконання	Відмітка про виконання
1.	Створення та узгодження графіку роботи дипломного проектування Написання 1 розділу, обговорення з керівником	16.05.22 – 18.05.22	
2.	Завчасний друк 1 розділу та додаткових сторінок (чернетки) - титульної, завдання, графік, реферат, зміст, вступ, 1-й нормо-контроль.	18.05.22 – 23.05.22	
3.	Написання 2 розділу, обговорення з керівником	23.05.22 – 27.05.22	
4.	Написання 3 розділу, обговорення з керівником	27.05.22 – 30.05.22	
5.	Написання 4 розділу, обговорення з керівником	31.05.22 – 03.06.22	
6.	Закінчення редагування та друк пояснювальної записки	03.06.22 – 04.06.22	
7.	Проходження нормо-контролю, перевірка на антиплагіат, перепліт пояснювальної записки.	04.06.22 – 06.06.22	
8.	Створення тексту доповіді. Представлення графічного матеріалу для презентації	07.06.22 – 10.06.22	
9.	Отримання відгуку керівника, рецензії.	11.06.22 – 12.06.22	
10.	Підготовка матеріалів для передачі секретарю ДЕК в папці	18.06.22	

7. Дата видачі завдання: “15” лютого 2022 р.

Керівник дипломної роботи (проєкту) \_\_\_\_\_ Марченко Надія Борисівна  
(підпис керівника) (П.І.Б.)

Завдання прийняв до виконання \_\_\_\_\_ Клопенко Роман Юрійович  
(підпис випускника) (П.І.Б.)

## РЕФЕРАТ

Пояснювальна записка до дипломної роботи «програмний модуль організації спілкування між користувачами соціальних мереж»: 47 с., 27 рис., 13 таблиць.

**ПРОГРАМНИЙ МОДУЛЬ, СОЦІАЛЬНА МЕРЕЖА, КОРИСТУВАЧ**

**Об’єкт розробки** – програмний модуль організації спілкування між користувачами соціальних мереж

**Мета роботи** – створення програмного модулю організації спілкування між користувачами соціальних мереж

## ЗМІСТ

РЕФЕРАТ.....	3
ВСТУП.....	5
РОЗДІЛ 1.....	7
1.1 Види та функції соціальних мереж .....	7
1.2 Опис предметної області .....	13
1.3 Специфікація вимог до системи.....	16
Висновки до розділу 1.....	23
РОЗДІЛ 2.....	24
2.1 Розроблення архітектури програмної системи .....	24
2.2 Планування системи бази даних.....	28
Висновки до розділу 2.....	31
РОЗДІЛ 3.....	32
3.1 Програмна реалізація модуля .....	32
3.2 Програмна реалізація бази даних.....	39

Висновки до розділу 3 .....	42
РОЗДІЛ 4.....	43
4.1 Тестування.....	43
4.2 Формування програмного продукту .....	48
Висновки до розділу 4.....	51
ВИСНОВКИ .....	52
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ .....	53

## ВСТУП

Напевно кожному з нас відомі такі сайти, як «Instagram», «Telegram», «Facebook». В народному сегменті їх називають просто – «соціальні мережі». В деяких людей вони навіть стали платформою бізнесу, маркетингу, дозвілля та стали призначені для багатьох інших корисних цілей. Кількість людей в таких мережах рахується на сьогодні мільйонами, кількість даних важить не один ексабайт, а число серверів становить не одну тисячу.

Всі ці ресурси, вже як було сказано, займають велику роль в певних людей. Для когось вони стали об'єктом самовираження, хтось займається бізнесом в «Instagram», хтось спілкується зі своєю дівчиною в «Telegram», а дехто вирішує бізнес-питання через «Facebook». Соціальні мережі в наш час мають досить великий спектр можливостей - ви можете зареєструвати власний профіль та вести його (змінювати фото, додавати статус профілю, вказати місце навчання та ін.), можливість додати своє красиве фото та показати йому всьому світу, та переглядати фото інших користувачів, додавати та слухати музику, додавати відео, та багато іншого.

Так як в наш час великого розвитку набула технологія Web, а з 2004 інтернет «вибухнув» технологією Web 2.0, з'являється чимале число схожих проектів та структур. В наш час також набуває популярності технологія Web 3.0 ( ті ж самі NFT), проте це ще дуже «сирий» проект і глобальні корпорації мають слабке уявлення про його розвиток. Про що йде мова, якщо щоквартальний дохід компанії МЕТА (вони володіють такими соціальними мережами як «Facebook», з 2012 року – «Instagram», та сервісом «Messenger») складає в середньому більше 20 мільярдів доларів. Проте, навіть в корпорацій – гігантів не все так добре з соціальними мережами. Проблеми в деяких мережах видно неозброєним оком – наприклад проблеми функціональності мереж (недостатня варіативність користувача в змінах свого профілю, невелика кількість «вільних» мереж, та платні сервіси) або навіть проблематика взаємодії мережі з користувачем. На сьогодні актуальною є мета створення засобу спілкування між користувачами соціальних мереж з новою технологією. Завданням роботи є створення мережі, яку користувачам буде зручно використовувати та мережі, яка зможе конкурувати вже з існуючою кількістю мереж.

Однією з основних умов проектування є ціль на відкрите програмне забезпечення. Програмування було проведено на мові PHP з використанням відкритого фреймворку під назвою Symphony, якою володіє компанія Sensio Labs. У функції СУБД було обране ядро MySQL.

На наш час проект знаходиться на стадії допрацювання та закритого тестування. Тестування буде здійснюватися за допомогою різних людей з різних категорій та з залученням сторонніх експертів. За певними результатами, що були отримані на даний час, була здійснена порівняльна характеристика.



## РОЗДІЛ 1

### АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ПІДТРИМКИ ПРОЦЕСІВ ОРГАНІЗАЦІЇ СПІЛКУВАННЯ МІЖ КОРИСТУВАЧАМИ СОЦІАЛЬНИХ МЕРЕЖ

#### 1.1 Види та функції соціальних мереж

Соціальна мережа – це інтернет сайт, площадка або онлайн-послуги, що призначені для організації, відображення та взаємодії користувачів. Закритими соціальними мережами називаються такі мережі, які відповідають певним умовам або мають певну сталу кількість людей. Однією з характеристик оцінки соціальних мереж є наявність «друзів» та «груп».

Відмінними рисами соціальних мереж є:

- Формування свого особистого профілю (він може бути доступний або напівдоступний). В профілі, як правило, вказуються якісь особисті реальні дані – ім'я, дата народження, особисті фото, місце навчання та багато різної інформації (вибір в музиці, жанри фільмів, які цікаві вам, хобі та ін.)
- Дозвіл користувачеві мережі надвеликого користування та обміну інформації з іншими користувачами (обмін фото, відео, повідомлень,

створення на своїй сторінці текстових вибірок (у вигляді своїх думок, спостережень, новин і т. д.), розвиток своїх співтовариств та ін.)

- Здатність створити особистий список інших користувачів із своїми відносинами («Друзі», «Робота», «Однокласники» та ін.)

Дехто вважає соціальними мережами такі платформи як Tumblr, SilverStripe, Medium та багато інших подібних. Це є помилкою, тому що в таких платформах доволі невелика кількість функцій.

З процесом удосконалення технології Web 2.0 соціальні мережі істотно поширилися насамперед в веб-сервісах та форумах. Із цього й почався розвиток ідеї соціальних мереж.

Починаючи з 1995 американський форум Classmates.com зробив відчутний «стрибок» у сфері соціальних мереж. На сьогодні російські «Однокласники» є самим прямим його аналогом. Форум набрав все більше і більше нових учасників і був доволі продуктивним, тому не дивно що за деякий час з'явилась велика кількість його аналогів. Але офіційною датою «вибуху» соціальних мереж прийнято вважати 2003-2004 роки, коли було створено такі відомі соціальні мережі, як Facebook, Tagged, Mixi.

І якщо Tagget було створено виключно як ідею для нових знайомств та платформою для запрошення на побачення, то японська мережа Mixi та американський Facebook створили ідею «самовираження в інтернеті». За Пірамідою Маслоу потреба в самовираженні – це найвища потреба людини. Людині не настільки важлива потреба в повазі до себе, як потреба в повазі від інших.

Соціальні мережі стали невід'ємною частиною нашого життя. Всі ми хочемо поспілкуватися з друзями, оцінити нове фото подруги чи друга, або написати коментар до статті про технології.

Мережа інтернету настільки неосяжна, що людському розуму дуже важко сприймати такі речі, в соціальних мережах «живуть» мільйони людей. І серед цих мільйонів людей необхідно знайти «своїх». Людей, яким буде цікаво те, чим ви займаєтесь і що пропонуєте світу. Зарекомендувати себе як людину, яка стала профі в тій чи іншій галузі і спонукати їх обрати саме ваш продукт і перейти на саме ваш сайт. Наприклад, розглянемо детальніше термін «Спільнота».

Ваші читачі і фоловери будуть вам довіряти в майбутньому. Ви провели певну кількість семінарів, написали не одну кількість статей і за весь цей час (зазвичай, як показує практика це займає від пів року до року) ви набираєте свою аудиторію, яка вас слухає та довіряє вам. І після того, як досягається довіра з читачами ми отримуємо можливість рекомендувати ті чи інші товари їм, ці люди будуть купувати ці товари, оскільки ми себе перед цим чудово зарекомендували.

Проте, на цьому все не закінчується. Якщо ми зарекомендували себе, необхідно не підірвати довіру людей, які на нас поклались і нас послухали. І, навіть не дивлячись на те, що припускається можливість того, що ми не є експертами в тій чи іншій області, ми за певний час можемо такими стати. І ні в якому разі не можна недооцінювати себе і свою цінність.

Для цього необхідно створити зворотний зв'язок – спілкування з користувачами нашого сайту. Це можна зробити за допомогою регулярних опитувань, дослухатись до рекомендацій людей в коментарях та ін. Після такого знаєш, що людям потрібно і що вони хочуть бачити на твоєму сайті і знаєш, як створити мотивацію для нових користувачів і подавати щоразу щось цікаве і нове.

Якраз для вище описаних цілей нам і підходять соціальні мережі. Питання про те чому саме, має досить просту відповідь – соціальні мережі за своїми характеристиками гнучкі, перспективні, вони мають певний ряд виділення аудиторії (це можливість чітко визначити «хто є хто»), вони безкоштовні і в соціальних мережах можливо спілкуватися буквально з ким забажаєш – відомою зіркою, своїм другом, цікавими та видатними людьми. Звісно, ніхто не гарантує, що зірка Голівуду знайде час в своєму графіку, щоб відповісти студенту НАУ. Але якщо таке все ж вдалося, у вас з'явиться унікальна можливість – наприклад, взяти інтерв'ю і опублікувати його на своїй сторінці, або у якійсь своїй роботі посилатися на джерела відомого науковця.

Такі маніпуляції та методи проявляють себе не відразу. Такий метод можна назвати довготривалим. Але в той же час, не дивлячись на такий мінус, в результатах такий метод є дуже надійним і його можна використовувати в роботі.

Все, що необхідно для цього – зайти один раз на день в соціальну мережу, додати декілька друзів або написали коментарі до їх постів і все, для чого довго там затримуватися. І якщо такі дії проводити, ну скажімо декілька місяців, результат буде вражаючим – в оптимальному варіанті це тисячі друзів і дискусії в коментарях, в яких вас, як людину, яка зарекомендувала себе, вже знають та чекають. Ви можете поширити статтю або будь-які свої думки в фейсбук і здивуєтеся результату – це будуть тисячі читачів та критиків.

Тобто, соціальні мережі в наш час стали щось на кшталт інтернет-сховища. Тут кожен може бути ким забажає, виражати своє «Я», яке заховано глибоко всередині та ділитися своїми працями (нехай це буде фото, відео, стаття чи просто особиста думка) з багатомільйонною аудиторією людей, що так само, як і ми існують в соціальних мережах.

Якщо ми заговорили про кількість людей в соціальних мережах, варто виділити найбільші. Мережі Facebook, Instagram, Tagged, My Space популярні

в Північній Америці. Звісно, деякі з них мають велику кількість користувачів і у нас, проте створені вони були саме там.

Англомовними мережами-гігантами на сьогодні є:

- Facebook (є український інтерфейс, рис. 1.1);
- Twitter (є український інтерфейс, рис. 1.2);
- LinkedIn (є український інтерфейс, рис. 1.3);
- MySpace (є український інтерфейс, рис. 1.4);
- Instagram (є український інтерфейс, рис. 1.5).

Російськомовні соціальні мережі:

- ВКонтакте
- Однокласники
- Telegram (є український інтерфейс).

Інші мережі:

- Mixi (Японія);
- Nexora (Канада);
- Bebo (Великобританія);
- Friendster, Multiply, Orkut, Xiaonei і Cyworld (Азія).

За кількістю зареєстрованих користувачів лідирують:

- Facebook ( 2 000 000 000 )
- MySpace ( 25 000 000 )
- Twitter ( 1 300 000 000 )
- Instagram ( 30 000 000 )
- Telegram ( 500 000 000 )

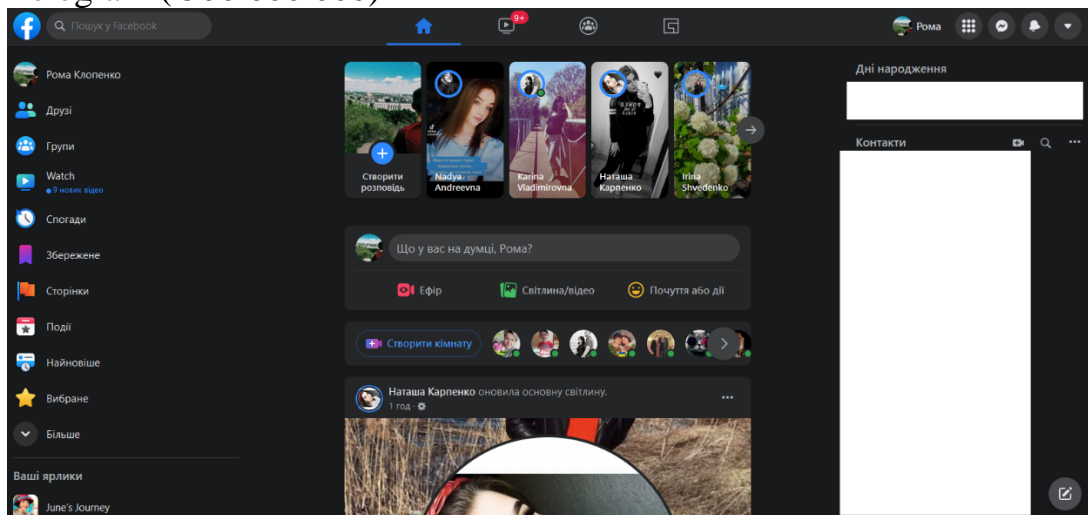


Рис. 1.1 Facebook

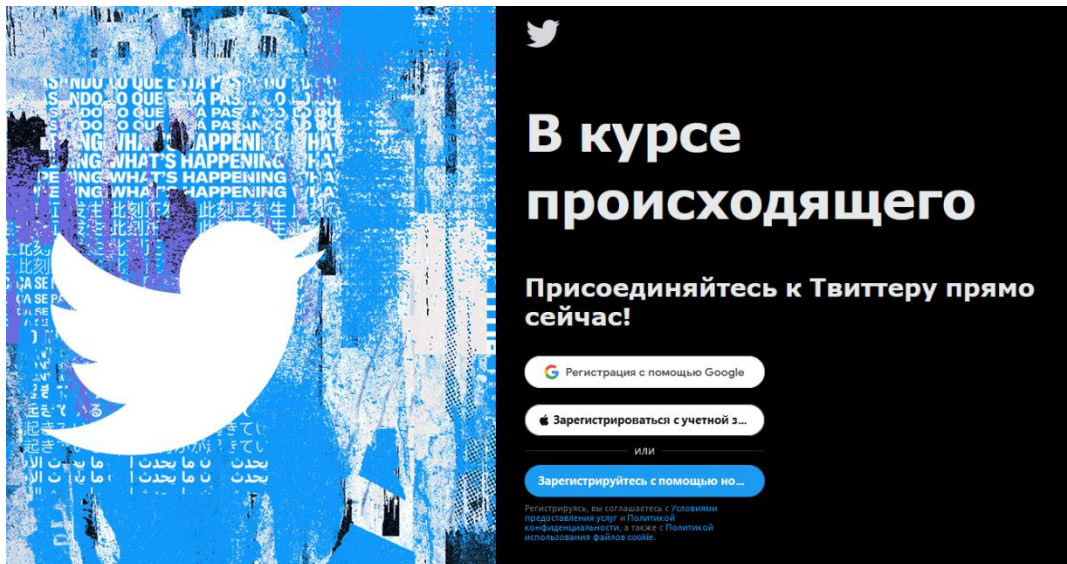


Рис. 1.2 Twitter



Обзор Люди LinkedIn Learning Вакансии Присоединиться Войти

Добро пожаловать в сообщество специалистов!

- Поиск новой работы >
- Найдите знакомого >
- Получите новый навык >



Рис. 1.3 LinkedIn

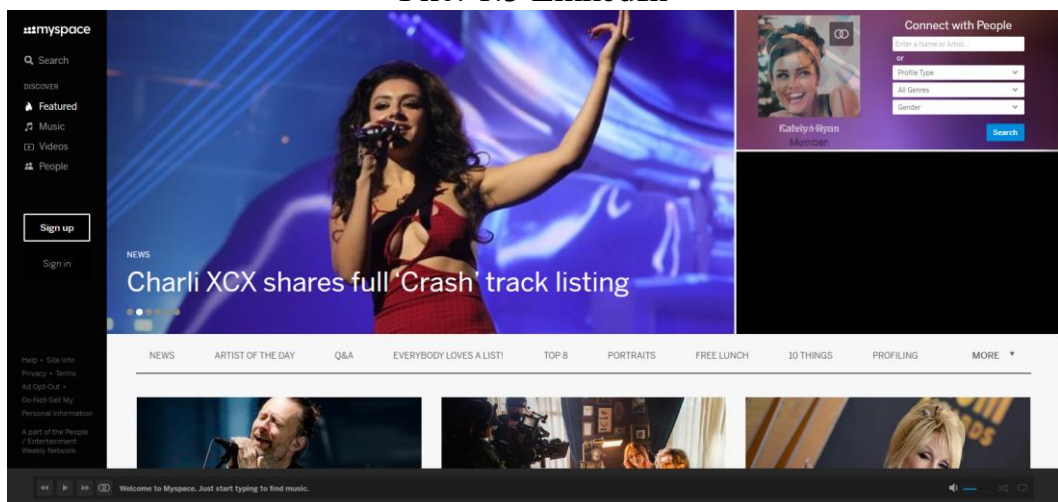


Рис. 1.4 MySpace

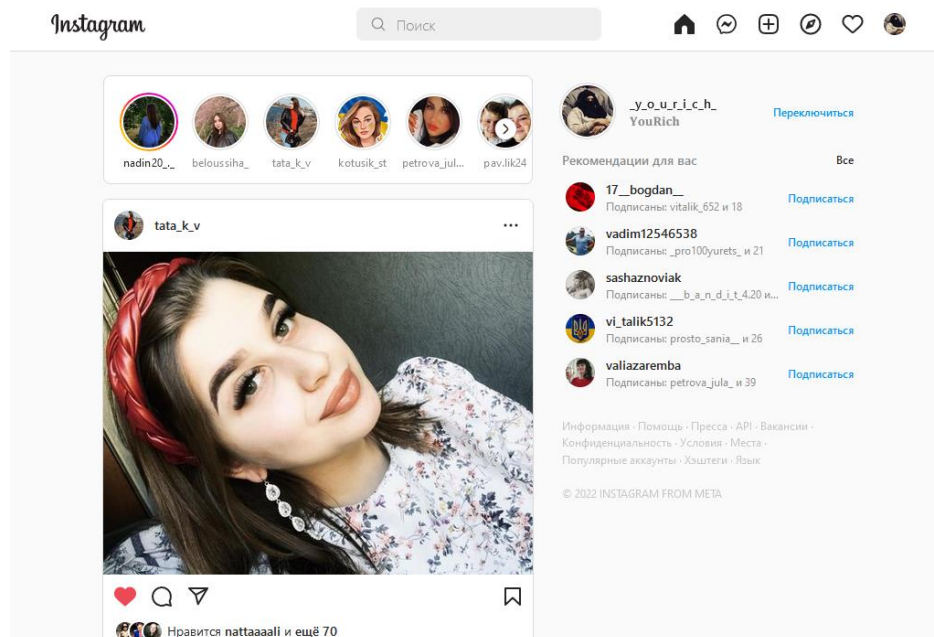


Рис. 1.5 Instagram

Web 2.0 має певну кількість типів соціальних ресурсів:

- Соціальні закладки надають можливість користувачеві створити і використовувати список популярних веб-сайтів або будь яких закладок. Ще однією особливою можливістю таких сайтів є пошук користувачів із спільними інтересами.
- Соціальні каталоги дозволяють користувачам використовувати бази знань наукових статей, проте на перший погляд вони дуже схожі на соціальні закладки.
- Соціальні бібліотеки – існує певна кількість додатків, що дозволяють своїм користувачам залишити посилання на свої наукові праці, книги, музику, статті, які в подальшому майбутньому будуть доступні іншим. Зазвичай, в таких бібліотеках передбачені списки рекомендацій, оцінки якості, рейтингу та ін.
- Соціальні мережі вебмайстрів можуть використовуватися для того, щоб автори мали змогу залишати посилання на свої пости, обговорювати між собою та голосувати за різні цікаві анонси.
- Сервісні соціальні мережі дозволяють об'єднуватись користувачам, які мають спільні інтереси, займаються однією справою або мають спільні інтереси.
- Соціальні медіасховища – це сервіси, які можна використовувати для спільного збереження будь-яких медіафайлів. В основному, вони класифікуються за типом файлів, які розташовані на сервері.
- Спеціалізовані соціальні мережі – призначені для об'єднання людей за будь-яким певним критерієм – вік, стать, світогляд і т. д.
- Професійні соціальні мережі – в таких мережах зазвичай користувачі обмінюються досвідом, інформацією, обговорюють професійні теми та структурують ділові зв'язки.
- Корпоративні соціальні мережі призначені для вирішення задач

організації та регуляції роботи компанії.

- Геосоціальні мережі призначені для налагодження суспільного спілкування в залежності від позиції користувача. До того ж використовуються абсолютно різні способи визначення геолокації, що дозволяє визначити поточне місце людини і співставити його позицію в просторі з іншими користувачами, що знаходяться поблизу.
- Комерційні соціальні мережі призначені для допомоги клієнтам та персоналу компанії. В залежності від компанії, вони призначені для зворотного зв'язку між клієнтами і доповіді відгуків.

## 1.2 Опис предметної області

Не дивлячись на те, що в наш час соціальні мережі мають безліч можливостей і мають дуже велике значення для інтернет-об'єднання, вони також мають певний ряд істотних недоліків, які дають користувачеві певні труднощі з комунікацією з іншими:

- У користувача є суворо визначений список ознак, які можливо змінити (ім'я, прізвище, улюблені фільми та ін.), і створити на сторінці щось нове і «своє» неможливо;
- Недостатнє з'єднання послуг в системі;
- Відсутня стандартизація, кожна мережа має свої окремі протоколи для обміну повідомленнями, доступу до плейлистів і т. д.;
- Занадто невелика функціональність спільнот, відсутність виділення спільноті окремого імені домену;
- Погано розвинена система міток (теги), її можливості затосовані лише частково;
- Неможливість додавати контакти до груп з самого початку.
- (у більшості ресурсів) незручний інтерфейс користувача, ухилення від сучасних нових web-технологій, в яких більше розвинена швидкодія та зручність інтерфейсу для користувачів.

Така кількість недоліків і вивели в світ ідею соціальної мережі нового покоління.

Обговоримо пропоновані новинки для соціальних мереж, які будуть зроблені в даному завданні. Соціальні мережі, як вже було до цього сказано, мають доволі низький рівень розвитку стандартизації. В нашій соціальній мережі за допомогою сучасних протоколів, таких як наприклад ІМАР, ця проблема буде вирішена.

Мережа електронної пошти – це одна з найстаріших мереж інтернету, яка призначена для обміну повідомлень. Для взаємодії користувачів в інтернеті за допомогою e-mail було створено декілька стандартних протоколів. Останнім часом найбільш популярним був

протокол POP. Його ідеєю є отримання повідомлень від сервера та обробка їх в offline-режимі. Оскільки інтерактивні системи не стоять на місці і розвиваються, звертаючи увагу на клієнт-серверний підхід, функціональність POP протоколу виявилась недостатня, і лідером став новий протокол IMAP, в якому розмістилось все краще за всі роки розвитку електронної пошти. Протокол IMAP може отримувати заголовки повідомлень, шукати листи за певними ознаками, і створювати будь-які інші дії з поштовими скриньками так, наче вони є локальними. Для роботи з таким протоколом на сьогодні було написано сотні програм для десятків платформ. З цього випливає, що де б користувач не знаходився, він завжди матиме доступ до своїх особистих повідомлень. Тому що, на сьогодні протокол IMAP є в кожному телефоні і їх можна буде отримати декількома натисканнями на екран.

Для забезпечення задач електронної пошти в соціальну мережу рекомендується ввести поштовий клієнт. Враховуючи ще й те, що відправити повідомлення в e-mail в системі має бути так само просто й зручно, як і в звичних соціальних мережах. Взірцем для прикладу на сьогодні є електронна пошта Gmail компанії Google. За приклад буде взято і реалізовано схожу по зручності для користувача і функціональності мережу.

Коли користувач реєструється в соціальній мережі йому автоматично надається електронна пошта на IMAP-серверах. Проте, якщо в людини вже є електронна скринька, або він не хоче заводити нову, то йому буде запропоновано ввести свій електронний ящик який зареєстровано в будь-якого провайдера.

Щоб виконати такі дії, користувачеві буде достатньо ввести свій пароль від своєї пошти, наприклад на ukr.net – і він буде працювати з нашою поштою на нашій системі.

Також варто приділити увагу розширеному редагуванню особистого профілю. Поруч з загальновідкритими місцями «ім'я», «стать», «дата народження» та інші, користувач може ввести свої особисті поля. На додаток, стандартні розділи будуть включені автоматично, а для того, щоб додати свої, користувачу достатньо просто натиснути на вибрану кнопку.

Поряд з обміном особистими повідомленнями також можемо ввести використання web-чатів в соціальній мережі. Такий обмін можна створити на основі протоколу XMPP (Jabber). При бажанні будь який користувач зможе читати свої миттєві повідомлення з використанням будь-якого Jabber-клієнта, який створений буквально для будь-якої платформи.



Відчутною проблемою соціальних мереж є відсутність системи групування «друзів». Наприклад, «однокласники», «кращі друзі», «спіробітники» і т.д. Залежно від того, як багато часу ви проводите в соціальних мережах, кількість друзів буде змінюватися.

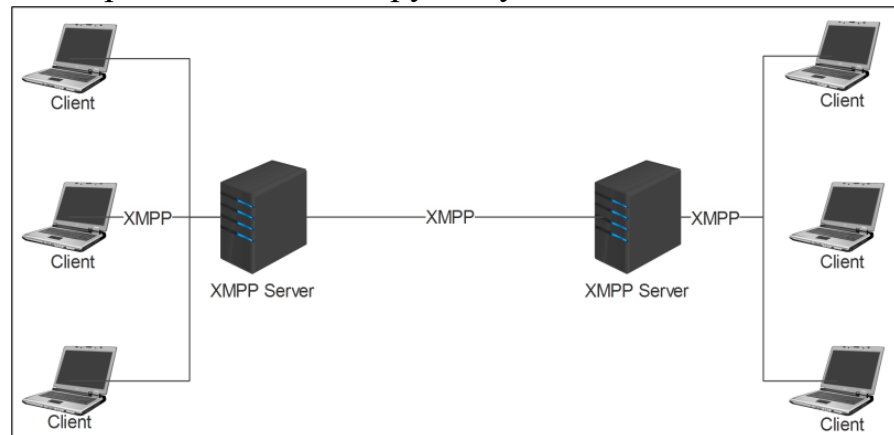


Рис. 1.6 Використання XMPP

Після певного періоду часу (зазвичай 1 чи 2 роки) користування сервісом, список друзів досягає 200-300, а то й більше і в такій кількості дуже важко знайти потрібну людину. Групування користувачів створили трішки пізніше, але не всі почали використовувати його – групувати більше ніж 200 чоловік в списку контактів є не тим, чим хочеться займатись. Виходячи з цього, користувачеві при додаванні друга відразу пропонується додати його в якийсь певний список друзів.

В наш час спільноти являють собою сторінку з схожим одне на одну оформленням і доволі невеликою кількістю функцій (блог, пости, галерея, контакти і т. д.), проте в деяких груп кількість повноти цієї сторінки цілком здатна конкурувати з повноцінними web-сервісами. Наприклад, спільноти груп, телеканалів, фірм та інших великих колективів. Запропоновано дати користувачу (адміністратору) можливість кардинально змінити групу – самому обрати її дизайн, функції, які будуть доступні звичайним користувачам, тобто буквально дозволити адміністратору створити свій сайт на основі ресурсів соціальної спільноти. Як варіант, також необхідно надати ресурсам, які будуть задіяні доменне ім'я або навіть надати користувачу можливість присвоювати доменне ім'я своїй адресі групи.

Мітки в наш час або не використовуються взагалі, або задіяні в дуже малих кількостях. У соціальній мережі сучасного зразку пропонується ввести можливість помічати абсолютно будь-яку інформацію, від листа повідомлення до музики. І на додаток, вони будуть пропонуватися самою системою за принципом найбільш вживаних слів, тобто коли користувач буде шукати мітку, в назві якої є слово «Окуляри», йому будуть доступні його мітки у вкладці «Сонячні

окуляри», обговорення в особистих повідомленнях окулярів для покращення зору або будь-які окуляри, які були помічені мітками. Вибрати, що саме він хоче, користувач зможе за допомогою натискання мишки по відповідній вкладці.

### 1.3 Специфікація вимог до системи

Специфікація вимог до програмної системи – це визначення деякого певного програмного продукту, проекту, або програми, що реалізують деякі рішення в деяких середовищах. Іншими словами – це абсолютний опис рішень системи, що розробляється.

Як правило, специфікація містить в собі наступне:

- Глосарій проекту
- Опис варіантів використання.

Запишемо список основних назв та понять в колі розробки програмної системи для організації спілкування між користувачами соціальних мереж. Інша кажучи, глосарій. Глосарій – це перелік понять в певній області знань і їх визначення. Всі ці назви і визначення до них наведено в таблиці 1.1

Таблиця 1.1

#### Глосарій

Назва	Опис назви
1. Головні поняття та класи області завдання	
Соціальна мережа	Веб-сервіс, або будь-яка інша служба веб, призначена для створення особистих профілів, списку людей, з якими користувач має взаємовідношення, та мати можливість дивитись на списки інших користувачів
Поштова скринька	Відома послуга в веб, що дозволяє створювати комунікацію між користувачами, обмінюватись повідомленнями різного змісту.
Чат	Інтернет засіб для швидкого обміну повідомленнями між користувачами в режимі онлайн
Програмний продукт	Програмне забезпечення готове до видачі замовнику (користувачу, покупцеві)
Програмне забезпечення	Певна кількість програм, що призначені для обробки інформації і документації, призначеного для експлуатації цих програм
База даних	Набір даних, що зв'язані між собою і логічно організовані, що призначені для задоволення інформаційної необхідності користувача.

Бізнес-процес	Будь-які задачі, що виконує вхідний продукт, змінює його цінність та творює вихідний продукт для зовнішнього і внутрішнього використання
2. Користувачі системи	
Адміністратор проекту	Особа, яка регулює управління програми на технічних рівнях.
Користувач	Особа, яка безпосередньо управляє БД
Власник профілю	Особа, що регулює відділ кадрів та займається зв'язком між всіма структурними відділами
3. Вхідна та вихідна документація	
База даних	Набір даних, що зв'язані між собою і логічно організовані, що призначені для задоволення інформаційної необхідності користувача.
Список контактів	Документ, який містить в собі інформацію про контактні дані користувача

Таблиці 1.2-1.12 містять в собі інформацію про варіанти використання для реалізації функцій системи. Діаграма варіантів використання наведена на рисунку 1.7

Таблиця 1.2

Версія застосування «Додати плагін до спільноти»

Середовище використання	Керування спільнотою
Задіяні люди	Адміністратор
Причина	Користувач зареєстрований та здійснив вхід в профіль
Тригер	Натискання на кнопку «Додати плагін»
План дій	<ol style="list-style-type: none"> <li>1. Додавання інформації в поля</li> <li>2. Аналіз правильності введених даних</li> <li>3. Натискання на кнопку «Зберегти»</li> </ol>
Результат	Опис наявної інформації про плагіни

Таблиця 1.3

Версія застосування «Редагувати зовнішній вигляд спільноти»

Середовище використання	Керування спільнотою
Задіяні люди	Адміністратор

Причина	Користувач зареєстрований та здійснив вхід в профіль
Тригер	Натискання на кнопку «Змінити»
План дій	<ol style="list-style-type: none"> <li>1. Додавання інформації в поля</li> <li>2. Аналіз правильності введених даних</li> <li>3. Натискання на кнопку «Зберегти»</li> </ol>
Результат	Опис наявної інформації про спільноту

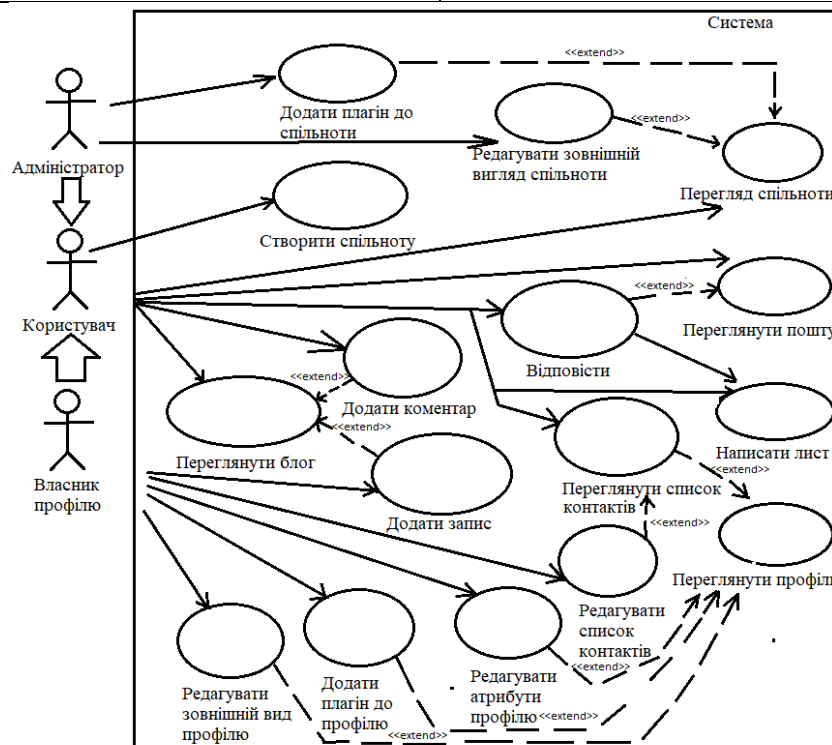


Рис. 1.7 Діаграми варіантів використання

Таблиця 1.4

Версія застосування «Створити спільноту»

Середовище використання	Керування спільнотою
Задіяні люди	Користувач
Причина	Користувач зареєстрований та здійснив вхід в профіль
Тригер	Натискання на кнопку «Створити»
План дій	<ol style="list-style-type: none"> <li>1. Встановлення характеристик</li> <li>2. Редагування спільноти</li> <li>3. Натискання на кнопку «Зберегти»</li> </ol>

Результат	Перегляд списку спільнот
-----------	--------------------------

Таблиця 1.5

## Версія застосування «Перегляд спільноти»

Середовище використання	Керування інформацією
Задіяні люди	Користувач
Причина	-
Тригер	Відкриття вкладки «Спільноти»
План дій	-
Результат	Показ спільнот за ознаками

Таблиця 1.6

## Версія застосування «Перегляд пошти»

Середовище використання	Керування інформацією
Задіяні люди	Користувач
Причина	Користувач зареєстрований та здійснив вхід в профіль
Тригер	Натискання на кнопку «Переглянути пошту»
План дій	-
Результат	Перегляд листів

Таблиця 1.7

## Версія застосування «Перегляд блогу»

Середовище використання	Керування інформацією
Задіяні люди	Користувач
Причина	Користувач зареєстрований та здійснив вхід в профіль
Тригер	Натискання на кнопку «Переглянути блог»
План дій	-
Результат	Перегляд блогу

Таблиця 1.8

## Версія застосування «Редагувати список контактів»

Середовище використання	Керування контактами
Задіяні люди	Власник профілю
Причина	Користувач зареєстрований та здійснив вхід в профіль
Тригер	Перегляд вкладки «Редагування списку контактів»

План дій	<ol style="list-style-type: none"> <li>1. Вибір контакту</li> <li>2. Редагування полів інформації</li> <li>3. Перевірка даних</li> <li>4. Натискання на кнопку «Зберегти»</li> </ol>
Результат	Перегляд інформації

Таблиця 1.9

## Версія застосування «Додати плагін до профілю»

Середовище використання	Керування профілем
Задіяні люди	Власник профілю
Причина	Користувач зареєстрований та здійснив вхід в профіль
Тригер	Вхід в розділ «Профіль»
План дій	<ol style="list-style-type: none"> <li>1. Редагування модулю</li> <li>2. Створення параметрів</li> <li>3. Перевірка інформації</li> <li>4. Натискання на кнопку «Зберегти»</li> </ol>
Результат	Перегляд списку модулів

Таблиця 1.10

## Версія застосування «Редагувати атрибути профілю»

Середовище використання	Керування профілем
Задіяні люди	Власник профілю
Причина	Користувач зареєстрований та здійснив вхід в профіль
Тригер	Вхід в розділ «Профіль»
План дій	<ol style="list-style-type: none"> <li>1. Підбір потрібної ознаки</li> <li>2. Натискання кнопки «Змінити»</li> <li>3. Затвердження змін</li> </ol>
Результат	Перегляд профілю

Таблиця 1.11

## Версія застосування «Редагування зовнішнього виду профілю»

Середовище використання	Керування профілем
Задіяні люди	Власник профілю
Причина	Користувач зареєстрований та здійснив вхід в профіль
Тригер	Вхід в розділ «Профіль»
План дій	<ol style="list-style-type: none"> <li>1. Обирання теми профілю</li> </ol>

	2. Натискання кнопки «Змінити» 3. Редагування полів інформації 4. Перевірка інформації 5. Натискання на кнопку «Зберегти»
Результат	Перегляд нового вигляду профілю

Таблиця 1.12 являє собою специфікації вимог функціонування системи

Таблиця 1.12

### Специфікація вимог функціонування

Номер вимоги	Назва вимоги	Особливості вимоги		
		Важливість	Складність	Контакт
1	Додати плагін до спільноти	Рекомендовано	Середня	Адміністратор
2	Редагувати зовнішній вигляд спільноти	Пріорітетно	Висока	Адміністратор
3	Створити спільноту	Пріорітетно	Висока	Адміністратор
4	Перегляд спільноти	Пріорітетно	Висока	Адміністратор
5	Перегляд пошти	Пріорітетно	Висока	Адміністратор
6	Перегляд блогу	Рекомендовано	Середня	Адміністратор
7	Редагувати список контактів	Пріорітетно	Висока	Адміністратор
8	Додати плагін до профілю	Пріорітетно	Висока	Адміністратор
9	Редагувати атрибути профілю	Пріорітетно	Висока	Адміністратор
10	Редагування зовнішнього виду профілю	Раціонально	Висока	Адміністратор

Нефункціональні вимоги зображено в таблиці 1.13.

Приведемо декілька прикладів нефункціональних вимог, що грають велику роль в завданні:

1. Застосовність:

- Кількість годин, що необхідна для навчання нових і досвідчених спеціалістів
- Умови зручності і відповідності стандарту графічного інтерфейсу користувача
- 2. Надійність
  - Якомога більша кількість часу безвідмовної роботи
  - Пропускна здатність каналу 100 Mb/s
  - Віддалений доступ
  - Відкритість – 5%
- 3. Робочі характеристики:
  - швидкість завантаження інтернет-ресурсу: 0,1 – 1 с;
  - число передач даних: 100 / 1 с;
- 4. Обмеження завдання:
  - Операційна система Windows;
  - MySQL;
  - Symfony компанії Sensio Labs;
- 5. Інтерфейси:
  - Інтернет інтерфейс – інтерфейс користувача.

Таблиця 1.13

Специфікація нефункціональних вимог

Номер вимоги	Назва вимоги	Особливості вимоги		
		Важливість	Складність	Контакт
Застосовність				



1.1	Кількість годин, що необхідна для навчання нових і досвідчених спеціалістів	Рекомендовано	Низька	Адміністратор
1.2	Головні умови використання нової системи в співвідношенні до дистем, які відомі користувачам	Раціонально	Низька	Адміністратор
1.3	Умови зручності і відповідності стандарту графічного інтерфейсу користувача	Рекомендовано	Низька	Адміністратор
<b>Надійність</b>				
2.1	Відкритість	Пріоритетно	Середня	Адміністратор
2.2	Час, який в середньому система працює безвідмовно	Рекомендовано	Середня	Адміністратор
2.3	Чіткість	Пріоритетно	Середня	Адміністратор
<b>Робочі характеристики</b>				
3.1	Необхідна кількість ресурсів	Рекомендовано	Середня	Адміністратор
4.1	Умови до технології програмування	Рекомендовано	Середня	Адміністратор

#### Висновки до розділу 1

Було створено та опрацьовано опис предметної діяльності, її напрями. Створено певну кількість функцій, які будуть включені до бізнес-процесу, на базі яких створено схема управління таким процесом. Проаналізовано сучасні соціальні мережі та проведено аналіз специфікацій до програмної системи.

## **РОЗДІЛ 2**

### **ПРОЕКТУВАННЯ СИСТЕМИ ОРГАНІЗАЦІЇ СПІЛКУВАННЯ МІЖ КОРИСТУВАЧАМИ СОЦІАЛЬНИХ МЕРЕЖ**

#### **2.1 Розроблення архітектури програмної системи**

Для вирішення даного завдання було обрано архітектуру клієнт-сервер. Така архітектура найбільш комфортно підходить для рішення таких завдань, оскільки нам необхідно надати доступ багатьом користувачам. Головною особливістю архітектури є те, що ми матимемо можливість віддаленого доступу.

Рішення задачі, яке побудовано на такій архітектурі, має складатися з 3 частин: взаємодія з базою даних, представлення даних користувачу, та бізнес логіка завдання, що виконує обробку запитів користувача та видає йому бажану інформацію. Всі дії з даними (їх представлення та збереження) виконуються на самому сервері, а показ даних і їх зміна відбувається на стороні клієнта.

Статичний погляд на таку архітектуру зображено на діаграмі на рисунку 2.1.

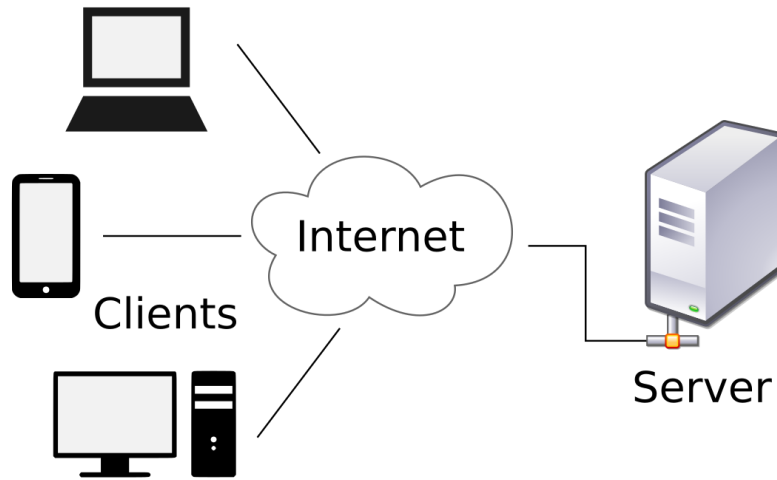


Рис. 2.1 Реалізація клієнт-серверної архітектури

На даній діаграмі клієнтською моделлю виступає користувач, який створений на основі PHP. В той же час користувач надсилає на сервер запити в залежності від того, з якими завданнями працює працівник відділу кадрів. Створення взаємодії регулюється за допомогою SQL-запитів.

Всі завдання бізнес-логіки виконуються на абсолютно іншому рівні. Будь-яка обробка потреб клієнта і відправка йому результату здійснена на даному рівні. На цьому ж рівні визначається, які потреби необхідно відправити в базу даних. Останній рівень – це сама база, яка отримує запити від SQL і відповідно, повертає дані назад.

На рисунку 2.2 зображено технічну структуру модулів.

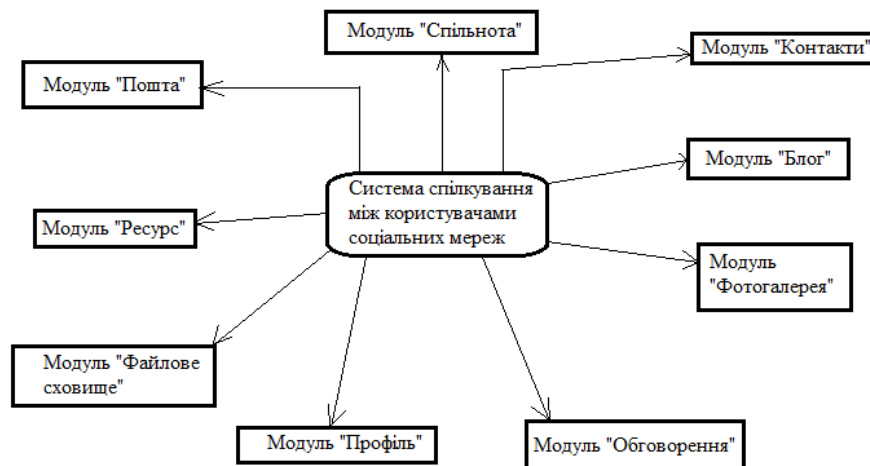


Рис. 2.2 Технічна структура модулів.

На рисунку 2.3 зображено вигляд розміщення відносно особливостей використання.

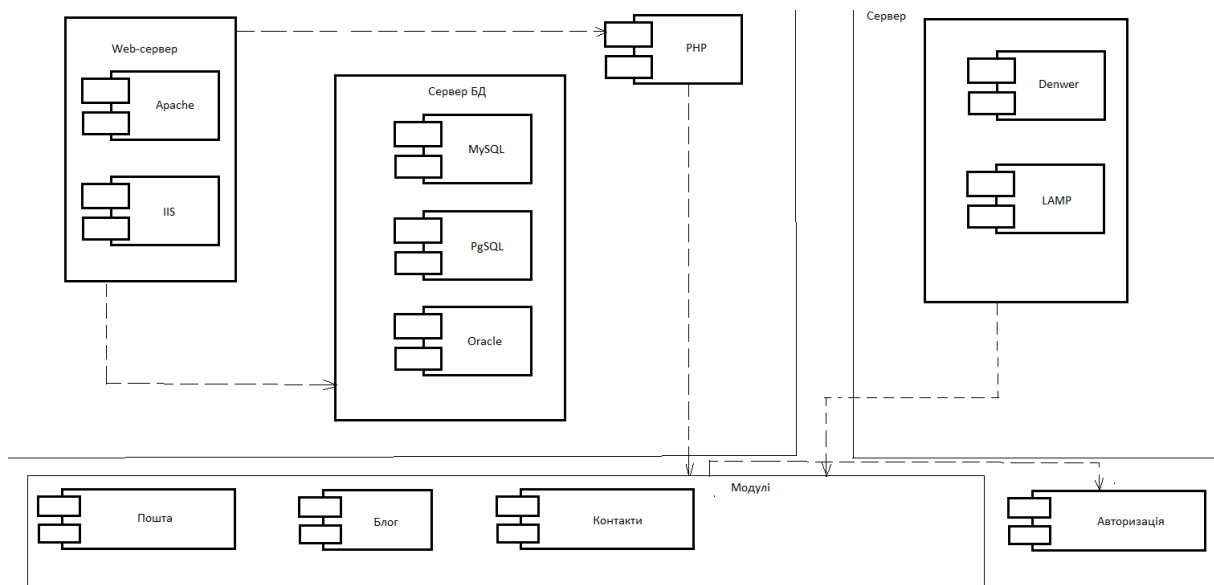


Рис. 2.3 Вигляд розміщення

На рисунку 2.4 зображено послідовність дій при авторизації користувача в соціальній мережі.

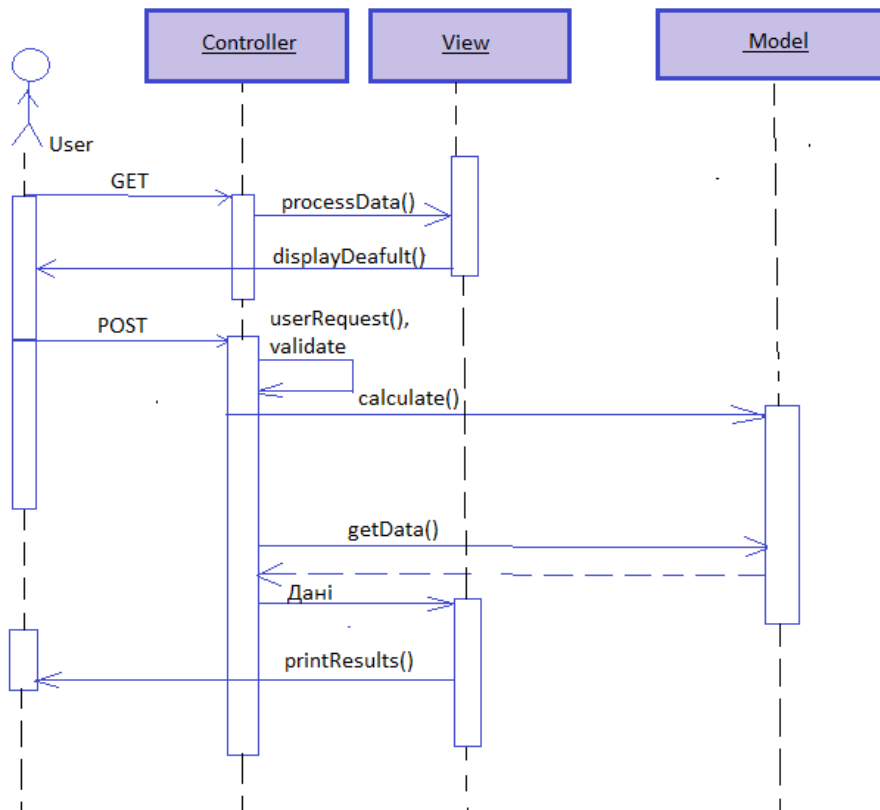


Рис. 2.4 Дії користувача при вході в систему

В процесі свого використання система здатна перебувати в деяких станах – авторизація, введення, оновлення, виведення, створення та збереження. Всі із перелічених станів зображена на рисунку 2.5.

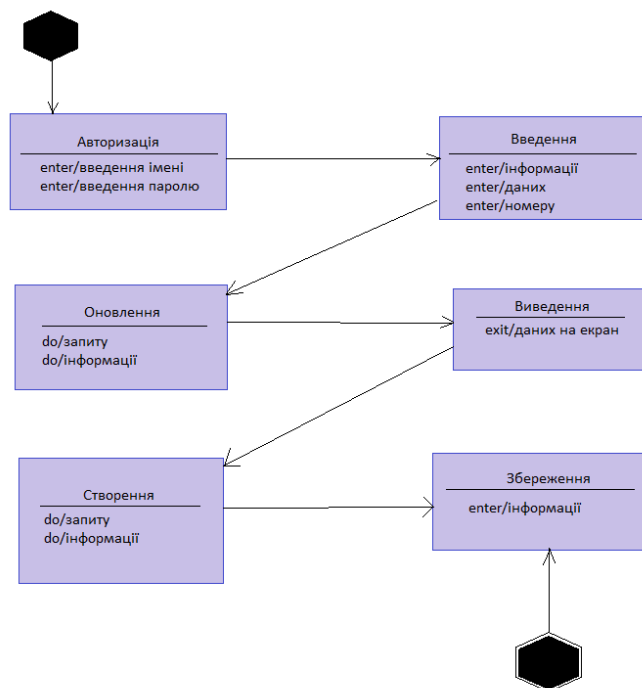


Рис. 2.5 Порядок станів процесу авторизації

Операція авторизації клієнта проходить за декілька кроків (рисунок 2.6) – ввід логіну, паролю та їх перевірку на правильність. Якщо ж щось із цього

було введено невірно, то система переключається знову на перший крок – ввід логіну і паролю.

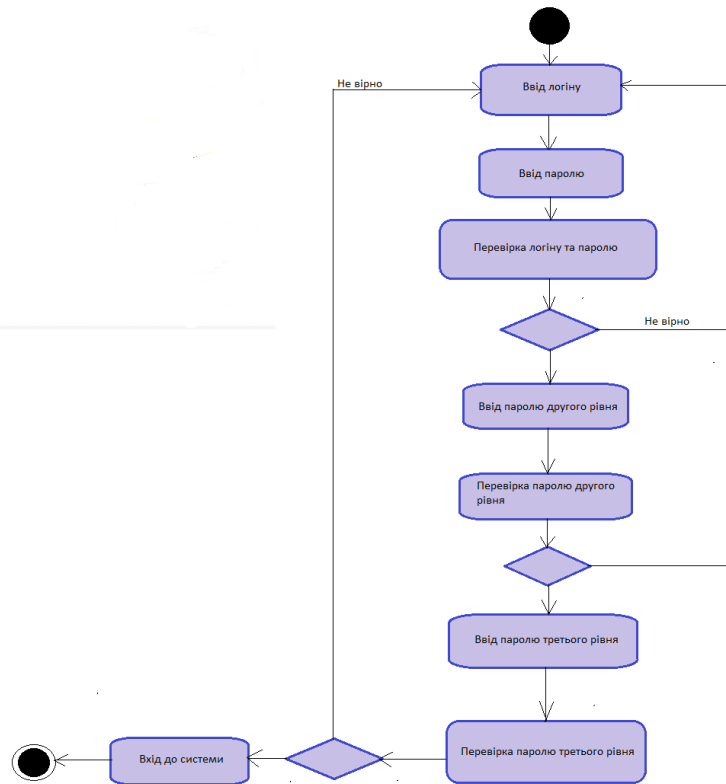


Рис. 2.6 Виконання операції «Авторизація»

## 2.2 Планування системи бази даних

У соціальних мережах містяться два види користувачів – адміністратор (вони також можуть відповідати за модерацію) і просто користувачі. І якщо між другими, в основному, немає ніяких відмінностей і вони не володіють великою кількістю переваг та можливостями то в Адміністратора в цьому плані спектр можливостей на порядок вище. Він також може мати профіль, який для інших користувачів не буде нічим відрізнитись від їх профілів. Проте, якщо адміністратор знаходиться в самій системі, то повноважень в нього на порядок більше.

Соціальна мережа частіше за все має в собі певну кількість таблиць, наприклад:

- Країна
- Банери
- Спільноти
- Місто
- Чат
- Новини
- Користувачі та ін.

На основній сторінці користувач бачитиме новини. Також він зможе бачити новини і посилання на них, які є в інтернеті. Такі новини

зможуть переглянути не тільки авторизовані користувачі. Також на основну сторінку можна розмістити погоду, курси валют і зареєструвати власний аккаунт для нових користувачів. Для цього їм необхідно буде натиснути на кнопку «Зареєструватись», після чого користувач побачить вікно користувацької угоди, з якою для продовження реєстрації він повинен погодитись. Після того, як користувач прочитає угоду, він має обрати кнопку «Згоден» і після чого на його особисту пошту буде відправлено e-mail з листом на підтвердження електронної пошти. Після того, як користувач набере код, він стане повноцінним користувачем соціальної мережі.

По закінченню реєстрації користувачу надаються можливості користування мережею. Також йому буде привласнено його особистий id. Id – це так звана «папка» користувача, в якій будуть розміщені фото, посилання на відео, музика та інші дані. Наступною сторінкою є профіль користувача. В ньому збережено всі дані про нього, такі як місце роботи, адреса, коментарі користувача і т.д. Якщо користувач не бажає встановлювати фото профілю, то соціальна мережа встановить таку сама. Внизу після фото будуть кнопки для взаємодії з ним – це «Видалити», «Змінити», «Обрати нове фото» та ін.

На сторінці редагування профілю в користувача є можливість змінити, наприклад ключове слово, або навіть видалити свій профіль, якщо є таке бажання. Сторінка «Друзі» буде показувати всіх інших користувачів, яких додав користувач і їхні фото профілю. Відео на веб-серверах використовуватись не буде. Користувач може в режимі реального часу дивитися відео з YouTube, всього лише при умові, що у вкладці «Редагування» він введе свій логін на сайті YouTube.

На вкладці «Пошта» користувач матиме можливість відправити листа на будь-який e-mail. Сторінка «Коментар» містить в собі всі коментарі користувачів у відкритому доступі. На додаток, це дозволяє переглядати всі коментарі і не дивитись кожен профіль інших користувачів окремо.

Сторінка «Чат» дає можливість користувачам спілкуватися в режимі реального часу. Є тільки певна умова для користування такою сторінкою – користувач повинен бути зареєстрований. Пункт сторінки «Вийти» дозволяє користувачу завершити роботу. В той же час за допомогою ключового слова і своєї адреси користувач може поновити вхід і працювати далі.

Основним елементом даної мережі є ресурс. Ресурс – це так звана вкладка з інформацією, яка може бути або особистим профілем користувача або головною сторінкою спільноти. Всі деталі сторінки є

інтерактивними, це означає що вони підлягають зміні як того захоче користувач, проте він має мати необхідні дозволи на такі зміни.

Наступним кроком є введення значення Віджета. Це певна частина ресурсу, яка може містити в собі певну інформацію. Взагалі віджет існує як область, яку можна змінювати – перемістити по сторінці, змінити колір та ін. Вони можуть бути або вбудовані – перелік друзів, задач, так і підключені окремо – блог, фото та ін.

Кожен віджет має зв'язок з якимось місцем і для кожного такого блоку має існувати «повна» вкладка в частині ресурсу, на якій буде показана видима частина віджета.

Уже було сказано, що система має можливість з'єднання додатковою функціональністю. Реалізація такої здатності можлива за допомогою списку плагінів – це свого роду модулі, які підключаються до ресурсу. Будь-який плагін має в собі віджет, який демонструється на основній сторінці і показує її при відповідному натисканні на плагін.

З цього випливає, що кожна частина ресурсу містить в собі певну кількість віджетів, що містяться на екрані в будь-якому порядку і певного числа вкладок, які створено для переходу на інші практичні елементи ресурсу. Список демонструється на вкладці плагінів і віджетів, зареєстрований користувач може здійснити вхід в свій обліковий запис та змінити їх на вкладці налаштувань.

Не можливо недооцінювати важливість поштового клієнту. Він показаний як інтерфейс для відправки і перегляду email-повідомлень.

Виходячи з основних архітектурних принципів, головною таблицею в реляційній базі даних є таблиця ресурсів. У зв'язку з нею існують таблиці плагінів, що містять інформацію блогу. Також зв'язки з таблицею ресурсів мають таблиці профілів та спільнот. Словники атрибутів пов'язані з таблицею профілів та із словником атрибутів. Контакти мають в собі інформацію про знаходження користувача в списку контактів різних користувачів, зв'язуючи профілі самі з собою. На рисунку 2.7 представлено приклад для діаграми бази даних «Сутність-зв'язок».

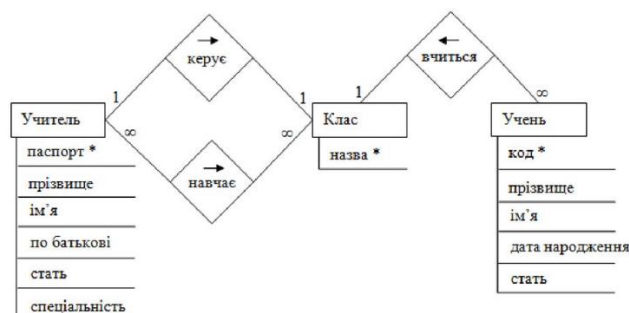


Рис. 2.7 Приклад моделі «Сутність- зв'язок»



## Висновки до розділу 2

У ході роботи розділу 2 було розроблено архітектуру програмної мережі, що в свою чергу дозволить краще розуміти основні частини функціоналу мережі. Було написано та описано структурну схему, яка містить в собі такі компоненти - рівень клієнта, рівень бізнес-логіки та рівень даних. Було проведено відображення структури схеми і основних елементів – модулів, які призначені для роботи з даними. Створено основні елементи бази даних та описано їх зв'язки.

## РОЗДІЛ 3

### ПРОГРАМНА РЕАЛІЗАЦІЯ СИСТЕМИ

#### 3.1 Програмна реалізація модуля

Ми вже згадували до цього, що проект розробляється на основі програми Symfony (рисунок 3.1), в якій він поділяється на певну кількість плагінів, контролери які інспірують виконання позначених задач. У ході створення роботи було створено модулі, які є в складі частини ядра системи - : «пошта», «ресурс», «профіль», «спільнота», «контакти», і додано до вище сказаних кілька прикладних модулів: «блог», «фотогалерея», «файлове сховище», «особистий блог» і «обговорення».

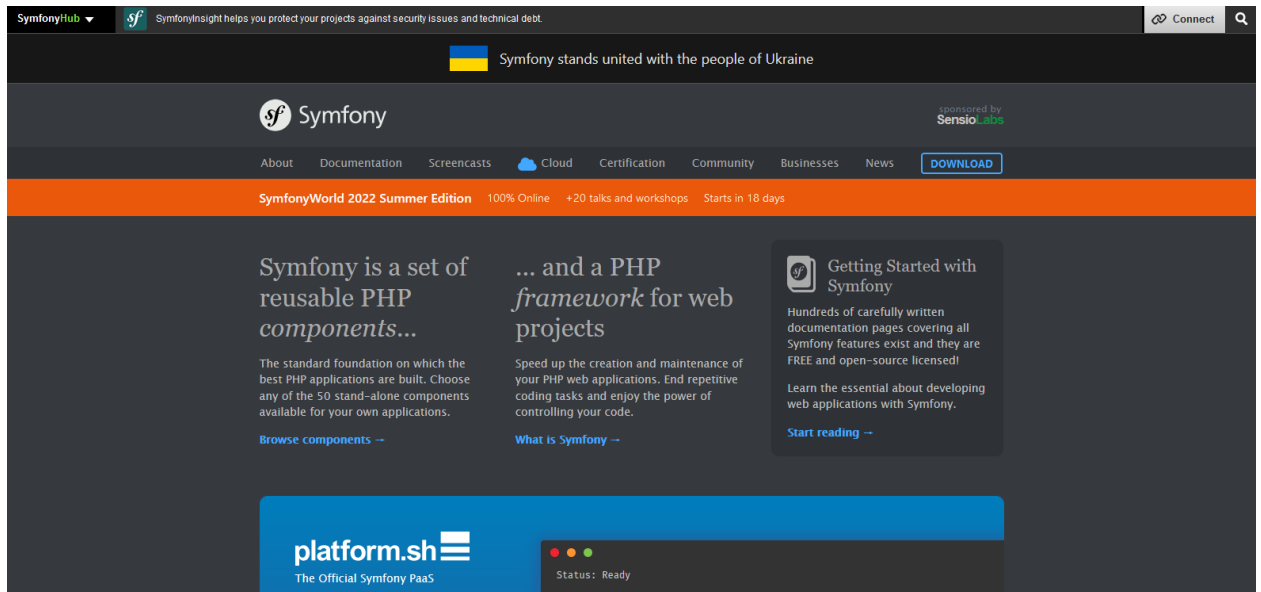


Рис. 3.1 SymfonyHub

Платформа Symfony містить в собі велику кількість службових класів для роботи з запитами, їх фільтрами, що виникають перед ініціацією рішень контролера. Не має великої необхідності описувати їх всі на діаграмі класів. Можливо, необхідно виділити, що контролер для запуску формується методом класу `sfRouting`, після чого запускається рішення `execute` даного контролера, який вже за своїм завданням ініціює необхідну дію. На рисунку 3.2 наведено основні контролери і їх класи, дії і методи, що виконують додаткові логічні дії області завдання.

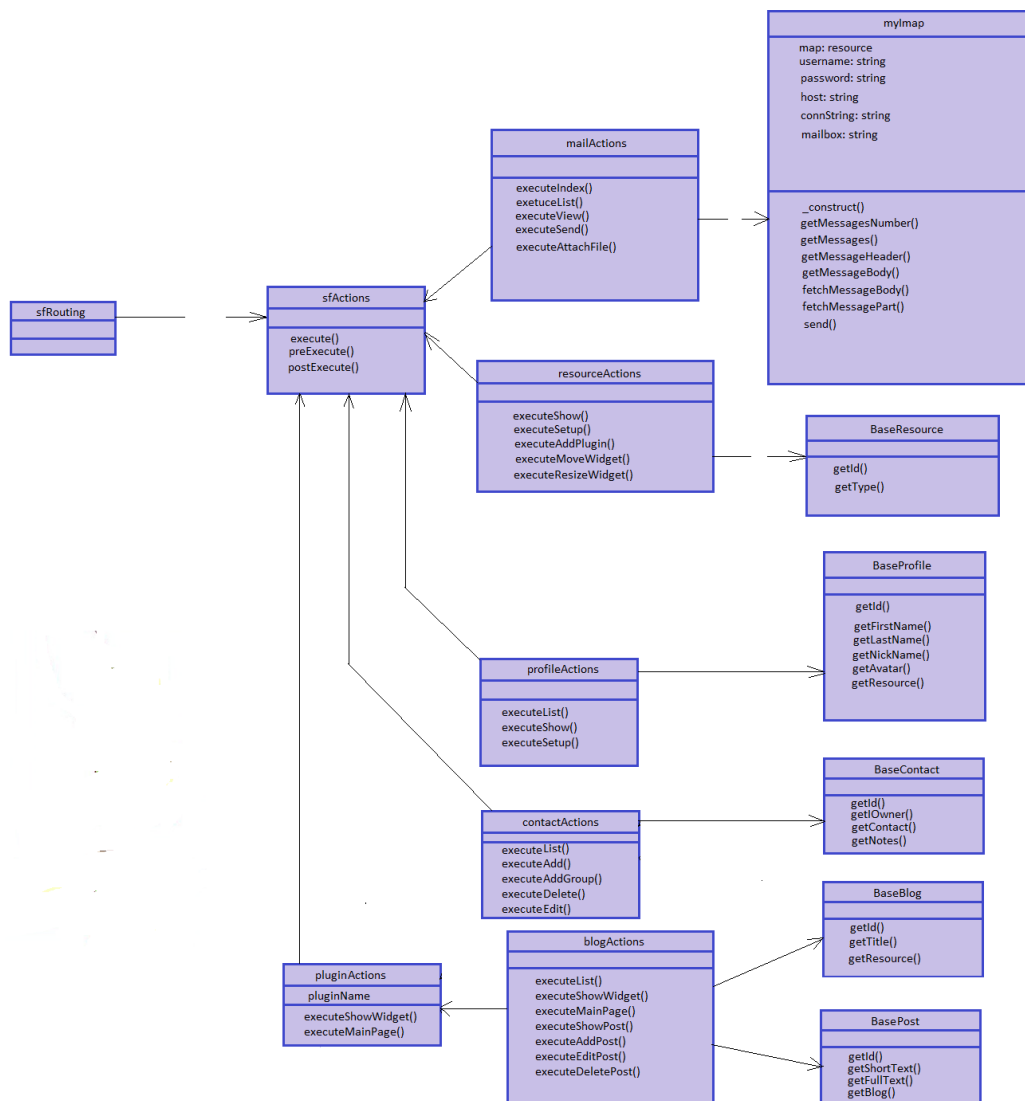


Рис. 3.2 Діаграма класів модуля

Під час обзору модуля буде описано певну кількість його функціональності в загальному, і також способи відповідних класів-контролерів, які є діями. Якщо при користуванні бізнес-модулем класів буде задіяно сторонні класи, вони також будуть описані.

Модуль «Пошта» призначений для реалізації функцій e-mail клієнту, що реалізується за допомогою протоколу IMAP. Оскільки всі повідомлення надсилаються між користувачами за допомогою даного протоколу, даний модуль є основним для комунікації між користувачами.

Категорія mailActions модуля містить в собі наступні рішення-дії:

- Index – приймає від IMAP сервера перелік заголовків повідомлень в вкладці і виводить їх певну кількість на екран. Також поточна дія виводить на екран список вкладок («Вхідні», «Надіслані», і т.д.) і навігацію кожної сторінки знизу вікна повідомлень. Ця дія влючає 1 параметр, який необхідно виконати, наприклад - ім'я папки, з якої необхідно прочитати листа.

- List – дія, що виконується за допомогою асинхронного умови. Отримує дана дія від IMAP-сервера перелік заголовків листів в конкретній папці на певній сторінці і віддає їх назад за допомогою протоколу на JavaScript. Протокол завантажує перелік в задану область на екрані. Створений для завантажень листів при умові, що користувач обрав певну сторінку або вкладку. Отримує два параметри : ім'я папки на сервері і номер сторінки.
- AttachItem – призначений для збереження обраного користувачем файлу і наступної його відправки поштовою вкладкою. Здійснюється після натискання на вікно «Прикріпити файл» під час зміни листа і вибору файлу для завантаження.
- Send – за умови, якщо дія викликана за допомогою методу GET – відкриває вікно редагування повідомлення - текстові рамки для вписування адресата, теми повідомлення, тексту повідомлення, кнопок «Прикріпити файл» і «Надіслати лист». Якщо буде натиснено на кнопку «Прикріпити файл», відкриється вкладка з можливістю обрати бажаний файл.

В якості підвищення комфорту користування всі дії із завантаженнями файлів в лист відбуваються без рестарту сторінки. Це стало можливим за допомогою змінного створення елемента IFRAME і його вказування в атрибуті target за сприйняття форми самої назви елемента. Під час вибору користувачем необхідного файлу для надсилання активується рішення submit в даний елемент, і після чого JavaScript здійснює перевірку всередині вибраного файлу. Після отримання значення success та назви файлу, який обрав користувач, файл зараховано як завантажений і JavaScript пише текст з назвою файлу в особисту ділянку на екрані.

Якщо ж дія викликана за допомогою методу POST і за умови, що всі дані, що були отримані від користувача введено вірно (адресати, теми, тексти листа і завантажених файлів та ін.), тоді відбувається створення багат шарового листа і здійснюється надсилання за допомогою протоколу SMTP.

- DeleteAttach – здійснює видалення файлу для вкладки та отримує на вхід ім'я файлу. Для того, щоб виконувати операції, що пов'язані з отриманням листів, до протоколу IMAP додається додатковий клас myImap. Він містить в собі наступні методи:
  - \_\_construct (\$username, \$password, \$mailbox, \$host) – конструктор класів, що призначений для з'єднання з вибраним поштовим сервером, користуючись введени раніше ім'я користувача та пароль.

- `GetMessagesNumber()` – приймає серверну кількість повідомлень і повертає кількість повідомлень з поштової скриньки.
- `GetMessages ($page, $perPage)` – відкликає заголовки листів на вибраній вкладці конкретної поштової скриньки.
- `GetMessageHeader ($messageNum)` – повертає заголовок повідомлення з обраним номером.
- `GetMessageBody ($messageNum)` – повертає основу повідомлення з обраним номером.
- `FetchMessagePart ($part, $partNum, $messageNum, $message)` – рекурсивний метод, аналізує склад рекурсивного листа.
- `FetchMessageBody ($messageNum)` – ініціалізує старт методу `FetchMessagePart` від певної частини листа верхнього рівня. Повертає об'єкт `myMessage`.
- `Send ($to, $subject, $text, $files, $uploadDir)` – надсилає повідомлення з обраною темою, вмістом та завантаженими додатковими файлами вибраному адресату.

Всі методи класу `myImap` користуються бібліотекою `php_imap` для здійснення правильної роботи своїх функцій.

Модуль «Ресурс» здійснює надійність відображення ресурсів – особистих сторінок і спільнот, а також відповідальний за результати та інтерактивні можливості віджетів. Робота контролера `resourceActions`:

- `Show` – створює вивід головної вкладки ресурсу. Отримує з бази інформацію про віджети і показує їх на екрані. До цього ж, виводить сторінки для зміни між плагінами.
- `Setup` – дістає і показує налаштування ресурсу: підключені плагіни, віджети, іншу інформацію про дизайн і місткість ресурсу.
- `AddPlugin` – додає новий плагін до функціоналу ресурсу. Включає в себе параметр з номером ресурсу.
- `MoveWidget` – реєструє в базі даних зміни, що стосуються зміни положення віджету по монітору. Основним параметром є інформація про координати та ідентифікація віджета.
- `ResizeWidget` – реєструє зміну даних про розмір віджета. Містить в собі наступні параметри – оновлений розмір віджета і ідентифікація віджета.

Модуль «Профіль» використовується для виводу наданого користувачу профілю та містить способи його редагування. Робота контролера profileActions:

- Show – надсилає запит користувача на задачу Show модуля Resource аналогічного профілю ресурсу.
- Setup – демонструє налаштування, які призначена для певного профілю користувача - ім'я, прізвище, по батькові, псевдонім, аватар, у образі текстових сторінок, а також пункт «Зберегти».
- Save – займається збереженням зміненої інформації та налаштувань в базу даних.

Модуль «Спільнота» вводиться для того, щоб мати можливість виведення головної сторінки спільноти.

- Show – змінює запит користувача на дію Show модуля Resource для аналогічного ресурсу.

Модуль «Контакти» призначений для користування списками контактів користувача. Він може додавати, видаляти учасників з кількості контактів, а також давати їм псевдоніми, під якими їх буде показано в мережі. Модуль може виконувати такі рішення:

- Index – надає поточному користувачу список контактів з бази даних в вибраній групі і демонструє їх на екрані. Також є доступ до вкладки редагування, в якій можливо замінити ім'я вибраного редагованого користувача на місці для вводу. Слідом за цими діями, відбувається збереження і після натискання на відповідну цьому кнопку завантажується дія Edit даного модуля. Виконання команди List отримує на вхід певні параметри. Перший з них – сам ідентифікатор користувача, для якого потрібно вивести на екран список друзів, або можна не виводити нічого, якщо за потребу визначено необхідність отримання кількості користувачів для користувача, який здійснив вхід в систему. Другий – ідентифікація групи, якій необхідно вивести цей список.
- Edit – рішення, що виникає за допомогою технології AJAX при діями з кнопкою «Зберегти» при редагуванні друзів. Отримує на вхід ідентифікатор користувача та інформацію, яка потребує збереження. Відбувається збереження даних, і якщо все проходить успішно, тоді вона зберігає нову інформацію про контакти в базі даних і повертає значення success.
- Delete – здійснює видалення користувача зі списку контактів. Отримує на вхід ідентифікатор користувача. Викликається за допомогою тієї ж самої технології.

- Add – відповідає за додавання контакту в відповідний список контактів. На вході отримує відповідний ідентифікатор користувача, якого необхідно додати та ідентифікатор групи, до якої буде доданий контакт.
- AddGroup – створює нову групу в списку груп користувача та зберігає її інформацію в базі даних. Виконується за допомогою AJAX-запиту. Отримує параметр ім'я групи.

Модуль «Блог» призначений для створення на сторінці ресурсу особистого блогу, на якому адміністратор може написати повідомлення, а користувачі залишити свій відгук про його повідомлення в коментарях. Аспекти модулю:

- Index – надає інформацію з бази даних і демонструє на екран певну кількість повідомлень в блозі обраного ресурсу. Отримує параметр ідентифікатор ресурсу.
- Post – розміщує повідомлення в блозі: створює його збереження в базу даних. Отримує параметри заголовку і самого тексту адміністратора.
- Show – дістає з бази і демонструє повний текст окремого поста в блозі. Може виводити коментарі до посту та місце для додавання нових коментарів.
- Comment – займається збереженням коментарів в базу даних. Сприймає 2 параметри – ідентифікатор поста і текст коментаря.

Модуль «Обговорення» було створено для реалізації співтовариства та профілів користувачів – кожен може створити та залишити коментар до коментаря іншого учасника. Дії модуля:

- Index – призначений для отримання з бази даних список тем для поточного обговорення. Приймає параметр ідентифікатора ресурсу.
- Post – відповідає за збереження повідомлення в темі. Отримує параметр теми і її текст, здійснює аналіз правильності даних та поміщає повідомлення в записи бази даних.
- Edit – здійснює редагування повідомлення на самому форумі, якщо користувач, що залишив це повідомлення, є поточним користувачем.
- List – отримує з бази повний перелік повідомлень для поточної теми. Отримує на вхід тільки ідентифікатор теми.

Модуль «Фотогалерея» існує як прикладний модуль для того, щоб переглядати, зберігати, та редагувати галерею, що зв'язана з деяким ресурсом. Модуль виконує наступні дії:

- Index – демонструє користувачу список альбомів для вибраного ресурсу. Приймає параметр ідентифікатора ресурсу.

- Album – дістає список фотографій для обраного користувачем альбому. Параметром слугує ідентифікатор цього альбому.
- Show – показує на сторінці певну фотографію з альбому. Параметром є ідентифікатор фотографії.
- Add – відповідальний за додавання фотографій в альбом. Параметром є ідентифікатор альбому.

Модуль «Файлове сховище» надає користувачу можливість використання файлового сховища на сторінці ресурсу. Дії модуля включають в себе:

- Index – демонструє певну кількість файлів, що задані певними критеріями. Параметрами є: ідентифікатор ресурсу, критерії пошуку файлів (тип, розмір, дата додавання, і т.д.).
- Upload – дає змогу завантажити файл в сховище.
- Audio - ініціює дію Index з критерієм пошуку по аудіо-файлах.
- Video - ініціює дію Index з критерієм пошуку по відео-файлах.
- Pictures - ініціює дію Index з критерієм пошуку по файлах зображень.

Для збільшення функцій системи для початку варто зрозуміти основні принципи побудови і розташування головних системних компонентів. Оскільки додаток буде на базі MVC основні компоненти будуть розділені на 3 ланки - Модель, Вид і Контролер.

Основним «прошарком» Моделі є система класів, що призначені для обробки даних. Такі класи розміщені в каталозі `lib/model` основної директорії додатку. В каталозі `model` існує свій підкаталог `om`, в якому містяться файли `BaseModelNamePeer.php` і `BaseModelName.php`, в яких `ModelName` – це назва моделі (частіше за все схожа на назву таблиці в базі даних). Такі класи відповідають за обробку даних і роботу з самими даними. Для прикладу, використовуючи методи класу `BaseProfilePeer` програміст здатен витягти інформацію з бази даних, які на свій лад, існують як частини класу `BaseProfile`. Названі класи є теоретичними і згенеровані за допомогою ORM `Propel`. У діях із системою використовуються наслідувані від цих класів класи `ModelNamePeer` і `ModelName`, що розміщені на рівень вище в каталозі `model`. Такі класи програміст здатен доповнити власними методиками, як варіант, дописати `ProfilePeer.php` незвичні методи «діставання» інформації про профілі користувачів з бази даних, а в `Profile.php` – способи роботи з певними варіаціями профілів. Виходячи з таких пунктів, програміст може збільшувати прошарок моделі.

Шар вигляду існує як певна кількість шаблонів на мові `Smarty`. Вони всі знаходяться в каталозі додатків - `apps/frontend`. Шаблони поділяються на загальні шаблони та шаблони модулів.



Загальні шаблони розміщено в каталозі `templates` та експлуатуються великою кількістю модулів, для прикладу можна розглянути `layout.tpl`. Такий шаблон на звичайній мові є граничною рамкою сайту, в якому міститься логотип, меню, і багато інших можливостей представлення сайту. Він містить в собі результати праці модуля. Також можливо виділити шаблон `pager.tpl`, всередині якого існує зовнішній вигляд кодної сторінки навігації на сайті.

З основними шаблонами поруч можна приділити увагу і модульним шаблонам. Їх розміщено в каталозі `modules/module/template` і призначені вони для демонстрації дій роботи контролерів.

Шар контролерів продемонстрований безліччю класів з назвами як `ModuleActions`, де `Module` - назва модуля. Такі класи зазвичай містяться в файлах типу `actions.class.php` в підкаталогах `actions` списку окремих модулів. Завданнями таких класів є рішення контролера, що ініціюється після отримання запиту обробки клієнту класом-маршрутизатором. Для більшої кількості функцій контролера потрібно додати або змінити необхідний метод в деякий клас. Для прикладу можна обрати дію `Index` до модуля `Profile`. Для збільшення ефективності необхідно дописати метод `indexSuccess` до класу `ProfileActions`, що розміщений в `apps/frontend/modules/profile/actions/actions.class.php`. Більше інформації по темі методів, що доступні для реалізації більш розширеного спектру функцій для контролерів, їх видів та моделей, за бажанням можна побачити в документації на офіційному сайті платформи `Symfony`. Перелік основних модулів системи зображено в додатку А.

### 3.2 Програмна реалізація бази даних

`MySQL` – це незалежна комунікативна система управління базами даних. Така база має можливість пошуку, прийняття, сортування і збереження даних. Сервер `MySQL` керує можливістю читання даних та надає дозволи на роботу з ним великої кількості користувачів одночасно, гарантує швидкий доступ до інформації та видає дозволи на доступ тим користувачам, що мають на це відповідні права. Він застосовує мову структурованих запитів (`SQL`), яка у всьому світі вважається стандартною мовою запитів до бази даних.

Такий сервер постійно має бути ввімкнений на комп'ютері. Програми клієнту надсилають серверу `MySQL` `SQL`-запити за допомогою роботи сокетів (вони ж мережеві засоби), сервер здійснює роботу з ними і записує собі результат. Інакше кажучи скрипт показує, яка інформація йому необхідна від сервера баз даних. Після чого цей сервер відправляє відповідь користувачу.

Взагалі, система MySQL була створена компанією «ТсХ» для збільшення швидкості роботи масивних баз даних. Така система керування базами даних (СКБД) з відкритим кодом за ідею була винайдена в протидію платним системам. Спочатку MySQL була практично аналогом mSQL, проте починаючи з 1999 року вона на порядок випередила її за популярністю. Це призвело до того, що останню версію 3.8 mSQL випустили 9 червня 2006.

MySQL виникла за ідею застосувати mSQL до власних задач компанії: рішенням, для яких використовувалися ISAM — підпрограми нижчого рівня. Результатом став новий інтерфейс SQL, проте API- інтерфейс було збережено в нащадок від mSQL.

З плином часу MySQL ставала дедалі більшою і в наш час вона є однією з найбільш практичних систем керування базами даних. В першу чергу її практичне використання найбільше проявляється в створенні динамічних веб-сторінок, тому що має за плечима велику кількість підтримки у вигляді різноманітних мов програмування.

Станом на квітень 2022 найновішою версією MySQL є версія 8.0.29.0 Community Server. Також існує комерційний варіант Server Enterprise. В наявності спільного вони мають код програми і містять в собі наступні можливості:

- Крос-платформна підтримка.
- Можливість зберігання процедур та функцій
- Курсори
- Тригери
- Представлення оновлень
- Системний словник (схема інформації)
- SSL підтримка
- Кешування запитів.
- Внутрішні запити SELECT.
- Реплікаційна підтримка
- Систематично повна підтримка Unicode (UTF-8 і UCS2).
- Таблична сегментація

MySQL існує як великий багатопоточний сервер баз даних, для якого характерним є висока швидкість, стійкість та легкість користування.

Таку систему зазвичай рекомендують для невеликих та середніх додатків. Отриманий код з такою системи може бути компільований на дуже великій кількості платформ. Найбільш місткі і практичні можливості сервера проявляються в UNIX-системах, в яких існує можливість багатопотокового користування, що значно підвищує продуктивність системи в загальному випадку. При некомерційному користуванні MySQL є безкоштовним.

### Практичні переваги сервера MySQL:

- Легко встановити та використовувати
- Існує підтримка великої кількості користувачів, які одночасно працюють з базою даних.
- Кількість рядків в таблиці гранична 50 мільйонам.
- Швидкодія виконання команд
- Наявність відносно легкої, проте надійної системи захисту.

Мінусами системи MySQL залишається, як і було до цього, не реалізована підтримка транзакцій, альтернативою яких пропонується користуватися LOCK/UNLOCK TABLE, і на додаток відсутність підтримки зовнішніх (foreign) ключів, тригерів, збережених процедур та представлень (VIEW).

Проте, не дивлячись на ці мінуси, при розробці невеликих і середніх систем, які призначені для користування в межах робочих груп такі мінуси є практично невідчутними.

Текстовий режим роботи з БД на перший погляд виглядає як прописування команд у командний рядок, а рішення вибірок повертаються у виді певних таблиць, поля в яких розміщені один на одному, якщо дані не поміщаються на екрані.

На рисунку 3.3 зображено рішення відношення profile в середовищі MySQL та його відповідну DDL демонстрацію.

#	Имя	Тип	Сравнение	Атрибуты	Null	По умолчанию	Дополнительно
<input type="checkbox"/>	1 ID	int(11)			Нет	Нет	
<input type="checkbox"/>	2 user_id	int(11)			Да	NULL	
<input type="checkbox"/>	3 first_name	varchar(255) utf8_general_ci			Да	NULL	
<input type="checkbox"/>	4 last_name	varchar(255) utf8_general_ci			Да	NULL	
<input type="checkbox"/>	5 middle_name	varchar(255) utf8_general_ci			Да	NULL	
<input type="checkbox"/>	6 nick_name	varchar(255) utf8_general_ci			Да	NULL	
<input type="checkbox"/>	7 avatar	varchar(255) utf8_general_ci			Да	NULL	
<input type="checkbox"/>	8 resource_id	int(11)			Да	NULL	

Рис. 3.3. Реалізація відношення profile в середовищі MySQL

```
CREATE TABLE IF NOT EXISTS `mydb`.`profile` (  
  `ID` INT NOT NULL ,  
  `user_id` INT NULL ,  
  `first_name` VARCHAR(255) NULL ,  
  `last_name` VARCHAR(255) NULL ,  
  `middle_name` VARCHAR(255) NULL ,  
  `nick_name` VARCHAR(255) NULL ,  
  `avatar` VARCHAR(255) NULL ,  
  `resource_id` INT NULL , PRIMARY KEY (`ID`)) ENGINE = InnoDB
```

### Висновки до розділу 3

У цьому розділі було обґрунтовано технологію, обрану мову програмування та створено програмну систему. Обговорено можливості розробки бази даних та зроблено програмну реалізацію бази даних програмної системи за допомогою механізму збережених процедур.

## РОЗДІЛ 4

### ТЕСТУВАННЯ ТА ДОСЛІДНА ЕКСПЛУАТАЦІЯ

#### 4.1 Тестування

Для того, щоб об'єктивно оцінити якість проведеної роботи соціальної мережі, було зроблено її порівняння з провідними аналогами в наш час в загальному об'єктиві зручності взаємодії з користувачем. Для того, щоб простіше сформулювати такі великі думки, в сучасних книгах застосовується термін «Usability».

Виходячи із міжнародних стандартів ISO-9241, головними ознаками цього критерію є:

- Ефективність – надійність та цілісність, за допомогою яких користувачі виконують установлені задачі.
- Продуктивність – певна кількість засобів, яка необхідна користувачу, щоб грамотно і швидко виконати поставлену задачу.
- Задоволеність користувача – комфорт і можливості використання.

Аналогами для порівняння було обрано такі відомі соціальні мережі: Facebook і MySpace. Минулі способи оцінювання дали основу для того, щоб вважати, що дана соціальна мережа може цілком достатньо конкурувати з існуючим мережами. Велика кількість користувачів зауважили нові гнучкі особливості і також достатньо впевнено говорили про ефективність. Майже всі користувачі приділили увагу зручності інтерфейсу, яка належним чином відповідає критеріям користувачів.

Але поруч з плюсами, було виявлено і ряд мінусів – основним звісно є новизна проекту і відносно невелика налагодженість соціальної мережі. Над цими недоліками при певних маніпуляціях можливо покращити систему.

Тестування інтернет-орієнтованих систем – це вагома частина життя проекту, після якої можливо віддавати готовий проект замовнику без помилок, з достатньою комфортністю для читання, що в свою чергу означає легкість, зручність та надійність. Проведення тестування – це різниця між реальними результатами та бажаними на початку, інакше кажучи – це завдання для пошуку помилок та багів.

Головними правилами тестування інтернет-сайтів є дії, під час яких буде продемонстровано користувачеві, в якій мірі проект буде зручним та надійним, як легко буде виконувати пошук інформації, чи навпаки. Чи буде користувачеві просто сприйняти інтерфейс і чи дійсно весь функціонал сайту працює так, як було поставлено в ТЗ.

Процес тестування може бути виконано за допомогою багатьох методів, проте важливо не забувати про безпосередньо процес тестування і політику тестування. Від цих рішень і залежить порядок дій тестувальника.

В наш час, експерти з тестування веб-сайтів ввели певні категорії і способи тестування:

- Функціональне тестування
- Тестування комфорту користування (Usability)
- Тестування продуктивності
- Тестування інтерфейсу користувача
- Тестування безпеки

Тестування системи було проведено на домашньому сервері Apache, який міститься в збірці Denwer. На додаток, тестування також було проведено на браузерах Google Chrome та Mozilla Firefox на операційній системі Windows 10.

Для того, щоб почати перевірку, необхідно встановити збірку Denwer, всередині якої міститься локальний сервер Apache, інтерпретатор PHP5, та систему що взаємодіє з базами даних phpMyAdmin. Після чого необхідно ініціювати запуск локального сервера та відкрити браузер для перевірки роботи сайту. Якщо сайт не було запущено, тоді необхідно перевірити наскільки вірно прописана сама адреса сайту, та перевірити на самому сервері наявність файлу index.php.

По закінченню тестування системи вказаними браузерами інформація і інтерфейс відображались коректно та без нарікань. Основні модулі сайту було зображено вірно та перебували в робочому стані.

Розробники-початківці доволі рідко проводять тестування навантажень сайтів та веб-додатків. І тоді настає момент, коли при великому онлайні сайт в самий відповідальний час починає підводити.

Тестування було проведено за допомогою програми Apache JMeter. Дана програма є дуже сильним інструментом наднавантаження тестування, який має можливість виробництва величезної кількості запитів в один і той самий час за допомогою паралельній роботі на 2 і більше комп'ютерах. Програма має підтримку плагінів ,які допомагають в рішеннях розширення функціоналу. Завантажити програму можливо з офіційного сайту, а для того, щоб запустити її, необхідно здійснити її розпакування та запустити з директиви `\bin\jmeter`.

Після запуску програми в лівому куті бачимо 2 пункти – Test Plan і WorkBench. Для початку розширимо Test plan за допомогою Thread Group, що в свою чергу дасть нам можливість взаємодіяти з потоками. Щоб здійснити такі дії, нам необхідно натиснути правою клавішею на Test plan – Add – Threads (Users) – Thread group.

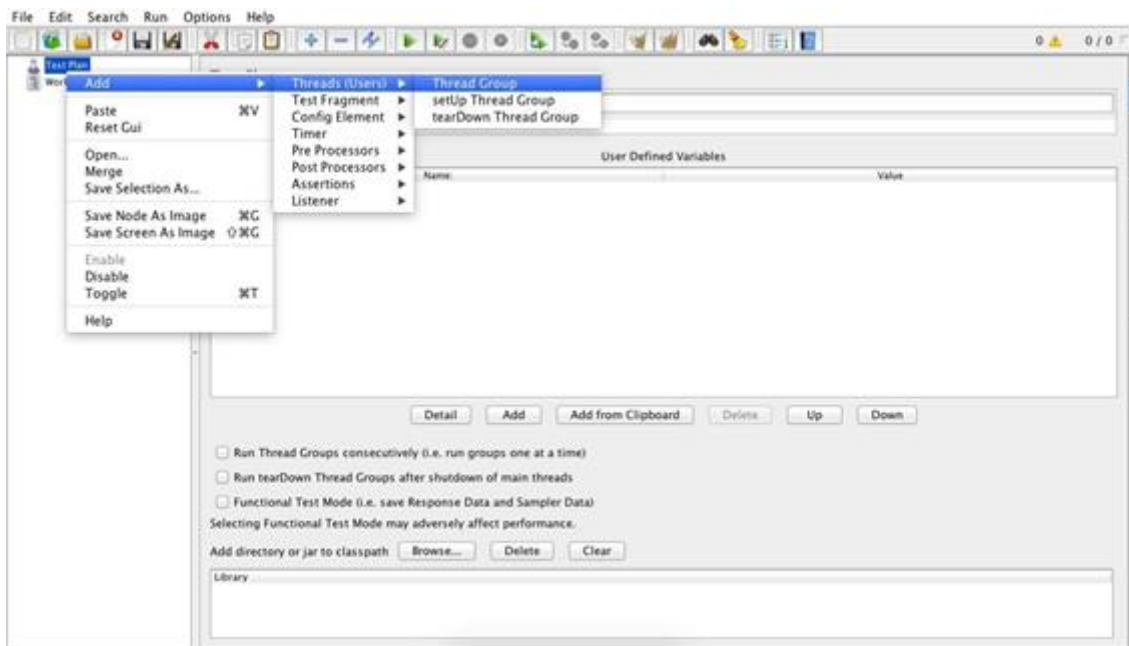


Рис. 4.1 Розробка тесту в JMeter

Також необхідно створити HTTP Request Defaults, що призначений для регулювання запитів тестування. Це можливо зробити за допомогою натискання правою кнопкою миші по Thread Group і обравши Add – Config Element – HTTP Request Defaults.

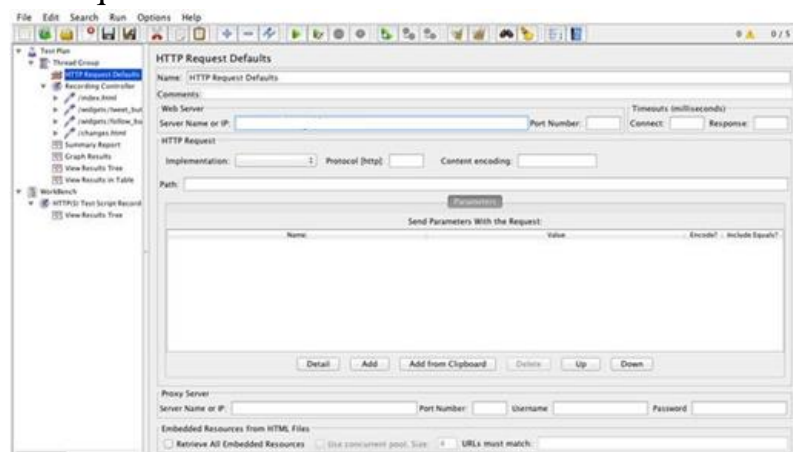


Рис. 4.2 Регулювання запитів тестування

Тепер нас нічого не зупиняє перед тим, як почати запис тестування. Щоб здійснити його, нам необхідно перейти на адресу сайту і виконати на ньому якісь звичайні дії. Даний сайт користується кешуванням інформації для входу в акаунт, тому нам не потрібно користуватися плагіном JMeter-а для цього.

Recording Controller зчитав та зберіг нашу активність і все майже виконано для того, щоб розпочати тестування навантаження.

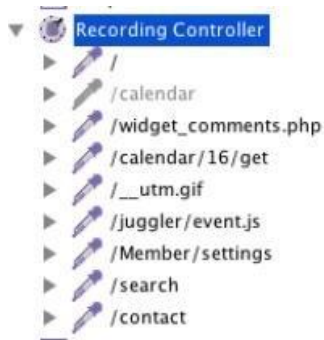


Рис.4.3 Налаштування Recording Controller

На додаток ми розширимо аспекти для перегляду результатів Summary Report (Thread Group – Add – Listener – Summary Report) і Graph Results (Thread Group – Add – Listener – Graph Results).

Наступним кроком ми будемо взаємодіяти з Thread Group. А саме нам необхідно вибрати кількість потоків (користувачів) – 5, час приєднання нового користувача пропишемо значення 10 (секунд), а в кількість циклів нехай матиме значення 50.

За деякий час до запуску необхідно виконати певні дії:

- Ctrl-S для збереження;
- Ctrl-E для видалення попередніх результатів;
- Ctrl-R для запуску.

Через певний час тестування буде завершено і ми зможемо переглянути результати. Для цього необхідно відкрити вкладку Graph Results та проаналізувати отримані результати.

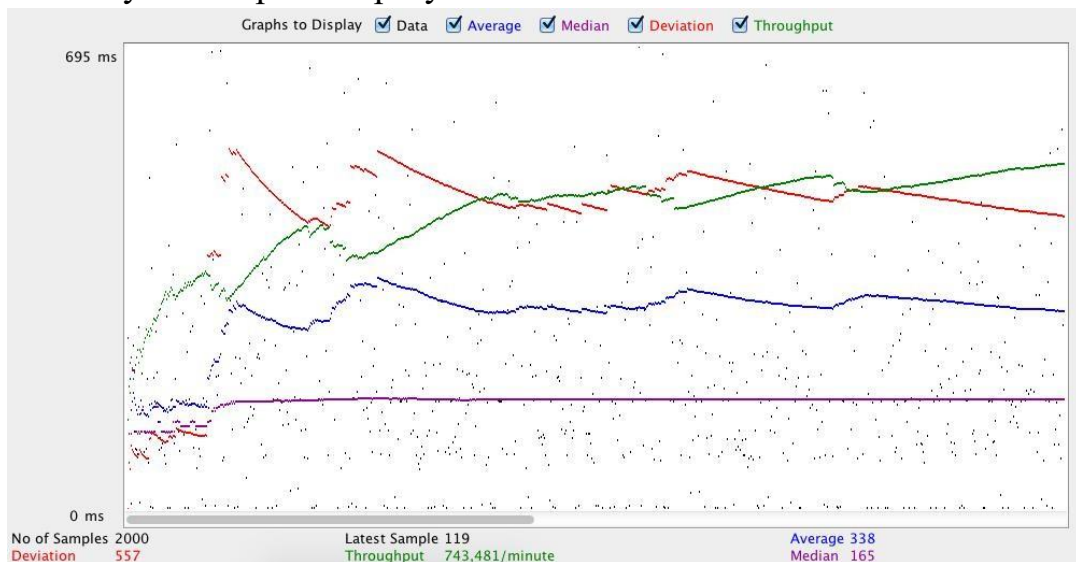


Рис. 4.4 Результати тестування

Значення надані в мілісекундах:

- Data – час відповіді особисто окремої одиниці даних, інакше кажучи кожного url, щ обув перевірений.
- Average – середній час відгуку, об’єктивний графік змін навантаження.
- Median – значення медіани (створено для статистики).



- Deviation – похибка, стандартне відхилення.
- Throughput – пропускна здатність запитів, що було виконано.

Для подальших рішень нам необхідні тільки значення Average і Throughput, які продемонструють навантаження на веб-сервер і пропускну можливість запитів. Дивлячись на графік вище, бачимо, що час відповіді приблизно 400мс і не збільшується. Це означає що сервер нормально витримує навантаження в 5 віртуальних користувачів. Але що вийде, якщо їх буде 50?

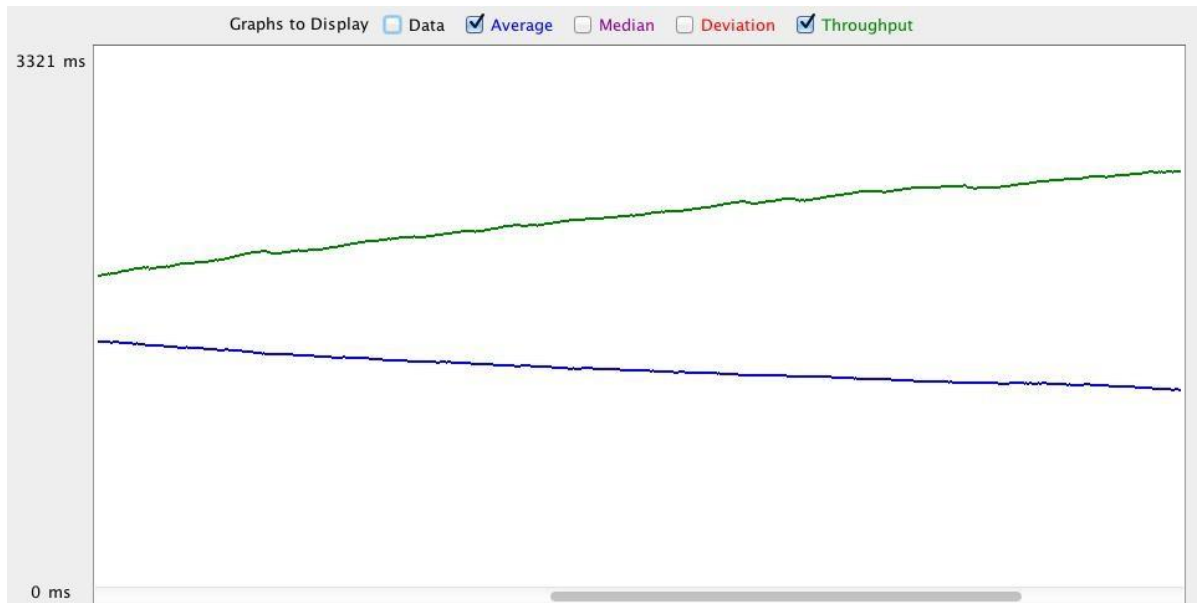


Рис. 4.5 Результати тестування

Час відгуку збільшується в декілька раз. Summary Report може продемонструвати статистику по кожному індивідуальному url.

Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	KB/sec	Avg. Bytes
/	250	365	80	3358	563,98	0,00%	1,6/sec	5,22	3422,8
/widget_comm...	250	108	60	3314	207,53	0,00%	1,6/sec	9,75	6394,4
/calendar/16/...	250	715	170	4559	822,59	0,00%	1,6/sec	7,11	4666,7
/__utm.gif	250	5	2	47	6,34	0,00%	1,6/sec	0,63	411,0
/juggler/event.js	250	173	0	788	62,62	1,20%	1,6/sec	0,45	295,7
/Member/setti...	250	688	164	4462	760,30	0,00%	1,6/sec	7,10	4664,0
/search	250	346	77	4112	502,32	0,00%	1,6/sec	2,76	1813,3
/contact	250	306	69	2683	399,52	0,00%	1,6/sec	3,30	2166,8
TOTAL	2000	338	0	4559	557,81	0,15%	12,4/sec	36,05	2979,3

Рис. 4.6 Результати тестування

У вкладці Average ми можемо помітити середній час відгуку, з чого з'являється логічне припущення, що чим більше часу, тим більше навантаження на вибраній url. Доволі в багатьох випадках існує ймовірність важких сторінок, які в свою чергу і «загружають» систему, і якщо здійснити їх оптимізацію, цілому продуктивність збільшиться в декілька разів. View Result in Table продемонструє результати у вигляді таблиці, в яких вказано час, і його статус (успішно / неуспішно).

View Results in Table

Name: View Results in Table

Comments:

Write results to file / Read from file

Filename:  Browse... Log/Display Only:  Errors  Successes

Sample #	Start Time	Thread Name	Label	Sample Time(ms)	Status	Bytes	Latency
115	15:52:56.007	Thread Group 1-8	/calendar/16/get	3388	▲	4660	199
116	15:52:59.293	Thread Group 1-34	/widget_comment...	126	▲	6511	125
117	15:52:59.399	Thread Group 1-8	/_utm.gif	26	▲	411	26
118	15:52:59.052	Thread Group 1-26	/contact	395	▲	2160	388
119	15:52:55.125	Thread Group 1-3	/Member/settings	4322	▲	4642	3158
120	15:52:54.654	Thread Group 1-6	/Member/settings	4798	▲	4643	2326
121	15:52:56.772	Thread Group 1-24	/	2680	▲	3459	2664
122	15:52:56.914	Thread Group 1-21	/Member/settings	2554	▲	4649	110
123	15:52:57.372	Thread Group 1-27	/	2131	▲	3439	2130
124	15:52:57.574	Thread Group 1-28	/	1935	▲	3408	1932
125	15:52:57.977	Thread Group 1-30	/	1535	▲	3413	1532
126	15:52:59.179	Thread Group 1-36	/	334	▲	3406	325
127	15:52:59.314	Thread Group 1-31	/juggler/event.js	202	▲	276	202
128	15:52:59.389	Thread Group 1-22	/widget_comment...	143	▲	6510	142
129	15:52:59.394	Thread Group 1-23	/widget_comment...	139	▲	6511	139
130	15:52:58.874	Thread Group 1-1	/Member/settings	674	▲	4674	338
131	15:52:59.459	Thread Group 1-24	/widget_comment...	106	▲	6511	106
132	15:52:59.507	Thread Group 1-27	/widget_comment...	109	▲	6511	109
133	15:52:55.568	Thread Group 1-18	/	4050	▲	3407	4049
134	15:52:59.428	Thread Group 1-8	/juggler/event.js	200	▲	276	200
135	15:52:59.523	Thread Group 1-28	/widget_comment...	111	▲	6512	110
136	15:52:59.522	Thread Group 1-30	/widget_comment...	112	▲	6512	112
137	15:52:59.523	Thread Group 1-36	/widget_comment...	117	▲	6511	116
138	15:52:56.386	Thread Group 1-12	/calendar/16/get	3290	▲	4655	2958
139	15:52:59.679	Thread Group 1-12	/_utm.gif	7	▲	411	7
140	15:52:59.630	Thread Group 1-18	/widget_comment...	121	▲	6512	119
141	15:52:53.966	Thread Group 1-7	/calendar/16/get	5819	▲	4650	1042
142	15:52:59.789	Thread Group 1-7	/_utm.gif	8	▲	411	8

Scroll automatically?  Child samples? No of Samples 705 Latest Sample 7374 Average 1022 Deviation 1481

Рис. 4.7 Результати тестування

На цьому етапі тестування можна вважати завершеним. В процесі було перевірено систему на навантаження та отримали хороші результати.

#### 4.2 Формування програмного продукту

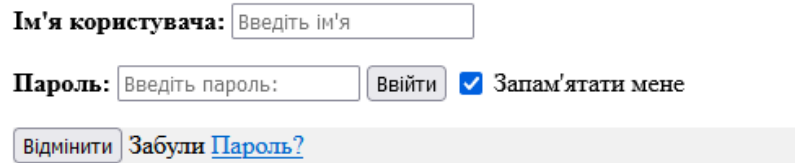
Для формування програмного продукту система потребує надійного хостингу у провайдера високої якості: фізична інформація сайту завантажуються за допомогою ftp на сервер, а за потреби відбувається розробка початкової бази даних, і на додаток відбуваються необхідні налаштування і після всіх цих дій сайт готовий приймати користувачів з усієї мережі Інтернет. Зазвичай, настройка сайту на хост відбувається безпосередньо розробником, при умові, що замовник надає доступ до ftp та бази даних, або до панелі управління самим хостингом.

Умови хостингу:

- Apache версії мінімум 3.0 на базі Unix/Linux
- В повній мірі підтримка PHP версії щонайменше 5.0, варто виділити відправку поштових листів та відсутність save-mode
- Можливість редагувати кодування (відсутність чітко встановленого кодування)
- В переважній більшості випадків - база даних MySQL версії як мінімум 5.5
- Якщо необхідно запустити деякі скрипти з періодичністю – необхідна наявність Cron (інструмент Unix системи)

#### 4.3 Керівництво користувача

Щоб отримати дозвіл на користування функціями соціальної мережі , потрібно авторизуватись в системі. Після того, як було введено дані, необхідно натиснути на кнопку «Вхід».



Ім'я користувача:

Пароль:    Запам'ятати мене

[Забули Пароль?](#)

Рис. 4.8 Вхід в систему

Потім, по завершенню авторизації, в правому верхньому куті екрану буде показано зменшена версія нашого фото, а також окреме меню.

- Для того, щоб зареєструватись в системі потрібно буде натиснути на вкладку «Реєстрація» , яка знаходиться під вікнами введення логіну та пароллю. На екрані буде показано вікно реєстрації. Тут вводиться власне ім'я, прізвище, псевдонім, адресу електронної пошти (якщо є) і текст, що показано на малюнку знизу сторінки. Далі просто підтверджуємо свої дані, натискаючи на кнопку «Реєстрація». Результатом таких дій буде відкритий вхід в систему, буде присвоєно власну електронну пошту , а якщо при реєстрації було введено власний e-mail, тоді його буде автоматично додано до пошти.

Щоб переглянути профіль користувача достатньо натиснути на його зображення або ім'я. Після цього вам буде продемонстровано фотографію користувача, лінк для відправки йому повідомлення та можливість додати його до списку своїх друзів, також буде показана особиста інформація , список друзів цього користувача та ін. На додаток, доступні вкладки елементів профілю, такі як фото-архів, блог. Щоб отримати доступ до них, всього лише потрібно буде натиснути на них.

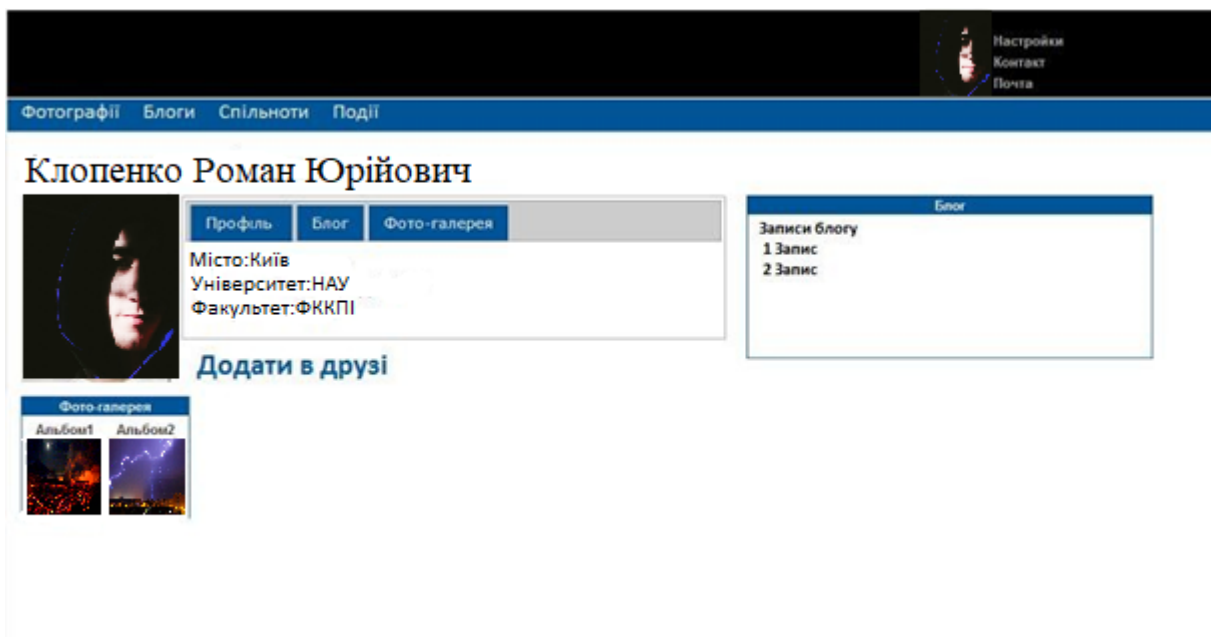


Рис. 4.9 Інформація профілю людини

Проте, якщо ви власник профілю, вам буде доступно більше можливостей для його зміни. Для переходу в такий режим достатньо натиснути на кнопку «Редагування», яка знаходиться під фото вашого профілю. Після цього зійніть всі потрібні вам зміни з віджетами, та збережіть зміни за допомогою кнопки «Зберегти».

Для редагування особистої інформації натисніть на впливаючу іконку редагування біля інформації, яку ви хотіли б змінити. Після цього текст буде змінено на вікно з його редагуванням. Редагуйте текст, та натисніть на іконку зберегти, що міститься поруч з полем.



Рис. 4.10 Редагування віджетів

Якщо вам необхідно додати новий атрибут в особисту інформацію, необхідно відкрити іконку «Плюс» біля нижньої частини списку

атрибутів. Буде продемонстровано поле для введення назви атрибута. Потім, після початку введення, вам уде запропоновано перелік атрибутів, які були введені раніше іншими користувачами. Для прикладу, якщо ви матимете бажання ввести атрибут «Адреса», то при перших трьох буквах «Адр» вам буде показано підказку «Адреса».

Для того, щоб відправити лист, натисніть на посилання «Надіслати повідомлення», що розміщено зверху над папками. Вам буде показано форму для введення теми листе, адресата, тексту листа, кнопки «Прикріпити файл» і «Надіслати». Потім, коли ви ввели інформацію в форму для відправки, натисніть «Відправити» знизу. Після цього, лист буде надіслано адресату.

#### Висновки до розділу 4

Було проведено опис операції тестування, його результатів, написані тест-вимоги до програмного забезпечення, а також виявлено недоліки. Вирішено питання установки та налаштування ПЗ на сервері а додано вимоги, за якими необхідно користуватися системою, також створено інструкцію для користувачів.

## **ВИСНОВКИ**

В підсумку праці над проектом, була створена дійсна версія web-сайту організації спілкування між користувачами соціальної мережі нового покоління. Створений базовий функціонал і також частково включено плагіни файлового сховища, обговорень, фото-галерей та блогу.

Модуль було створено на основі відкритого програмного забезпечення і відкритий для можливого розширення.

В процесі створення інформаційної системи дипломної роботи було використано найбільш поширений в наш час з професійних HTML-редакторів Nomesite редактор.

Він призначений для людей, які працюють з web-сторінками вручну, тобто створений для тих, хто знає HTML. В такому разі, ви маєте можливість повністю контролювати HTML код, також існує можливість створити зручною власну сторінку в одному з стандартних браузерів.

Створена система складається з двох технічних модулів: модуль організації структури, що створений на мові програмування PHP і модуль розробки і демонстрації структури соціальної мережі.

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Internet of Things for Industry and Human Applications. Dependability and Security of Internet of Things: practicum / V.V. Sklyar, V. V. Yatskiy, N. G. Yatskiev; Ministry of education and science of Ukraine, National aerospace university "Kharkiv aviation institute". – Харків: Юстон, 2019. – 98 р.
2. Сайт Вікіпедія [https://uk.wikipedia.org/wiki/Соціальна\\_мережа](https://uk.wikipedia.org/wiki/Соціальна_мережа)
3. Васильєв О. Програмування мовою PHP. – Дніпро: Ліра-К, 2022. – 368 с.
4. Lincoln Stein, Doug MacEachern. Writing Apache Modules with Perl and C (1st ed.), O'Reilly & Associates, 1999. – 723с.
5. Мельник Р. Програмування веб-застосувань (фронт-енд та бек-енд). – Львів: Львівська політехніка, 2018. – 248 с.
6. Цеслів О.В. WEB-програмування: навч. посібник / О.В. Цеслів; М-во освіти і науки, молоді та спорту України, Нац. техн. ун-т України “Київ. політехн. ін-т”. – Київ : НТУУ “КПІ”, 2011. – 296, с.
7. Andrew Ford. Apache Pocket Reference (1st ed.), O'Reilly & Associates, 2000. – 193с.
8. Кулик А. В. Теорія економічного аналізу: навч. посіб. / А.В. Кулик. – К.: ДП «Вид. дім «Персонал», 2018. – 452 с. – Бібліогр.: с. 444–451. .
9. Захист веб-сервісів: Лабораторний практикум [Електронний ресурс]: навч. осіб. для студ. спеціальності 123 «Комп’ютерна інженерія», спеціалізації «Комп’ютерні системи та компоненти» / І. А. Терейковський, Л. О. Терейковська, К. О. Радченко, С. О. Гнатюк; КПІ ім. Ігоря Сікорського. – Електронні текстові дані (1 файл: 2,72 Мбайт). – Київ: КПІ ім. Ігоря Сікорського, 2018. – 68 с.
10. Гайдаржи В., Ізварін І. Бази даних в інформаційних системах. – К.: Університет «Україна», 2018. – 418 с.

11. Lars Eilebrecht. Apache Web – Server (3rd ed.), MITP – Verlag GmbH, 2010. – 169с.
12. Інформаційні матеріали по PHP [Електронний ресурс] – Режим доступа: <http://www.master-web.com.ua/book/16> – Назва з екрану.
13. Інженерія програмного забезпечення: навчальний посібник / Д. П. Кучеров, Є. Б. Артамонов; МОН України, Національний авіаційний університет. – Київ: НАУ, 2017. – 388 с.
14. PHP: Hypertext Preprocessor [Електронний ресурс] – Режим доступа: <http://www.php.net/> – Назва з екрану.
15. Ерік Фрімен, Елізабет Робсон. Head First. Програмування на JavaScript. – Харків: Фабула, 2022. – 672 с.