

МІНІСТЕРСТВО ОСВІТИ І НАУКИ
НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ
Факультет кібербезпеки, комп'ютерної та програмної інженерії
Кафедра комп'ютерних інформаційних технологій

ДОПУСТИТИ ДО ЗАХИСТУ
Завідувач кафедри

_____ (Савченко А.С.)
« _____ » _____ 2021 р.

ДИПЛОМНИЙ ПРОЕКТ
(ПОЯСНЮВАЛЬНА ЗАПИСКА)
ВИПУСКНИКА ОСВІТНЬОГО СТУПЕНЯ
“БАКАЛАВР”

Тема: _____ „Web-портал для соціальної мережі”

Виконавець: _____ *Корнієнко Валентин Дмитрович*

Керівник: к.т.н., доцент _____ *Зіатдінов Юрій Кашафович*

Нормоконтролер: ст. викл. _____ *Шевченко А. Т.*

Київ 2021

КАЛЕНДАРНИЙ ПЛАН-ГРАФІК

№ п/п	Етапи виконання дипломної роботи	Термін виконання етапів	Примітка
1.	Аналіз предметної області за темою дипломного проекту.	13.05.2021-15.05.2021	
2.	Пошук, систематизація та обробка інформації по темі проекту.	15.05.2021-17.05.2021	
3.	Консультація з дипломним керівником, узгодження зібраних матеріалів.	18.05.2021	
4.	Дослідження актуальності соціальних інтернет-ресурсів в сучасному житті.	19.05.2021-21.05.2021	
5.	Обрання стратегії розробки веб-застосунку, засобів та методів для його реалізації.	22.05.2021	
6.	Розробка веб-порталу.	22.05.2021-28.05.2021	
7.	Оформлення висновків та пояснювальної записки.	29.05.2021-30.05.2021	
8.	Погодження з дипломним керівником, отримання рецензії від нього.	02.06.2021-05.06.2021	
9.	Створення презентації по проекту та доповіді для його захисту.	05.06.2021-14.06.2021	

Студент

(Корнієнко В.Д.)

Керівник дипломної роботи

(Зіатдінов Ю.К.)

РЕФЕРАТ

Пояснювальна записка до дипломного проекту «Web-портал для соціальної мережі» містить: 59с., 18 рисунків, 9 таблиць, 6 літературних джерел.

Об'єкт дослідження – web-портал для соціальної мережі.

Предмет дослідження – основні принципи створення web-порталу для соціальної мережі.

Мета роботи – розробка і створення web-порталу для соціальної мережі на із застосуванням актуальних технологій.

Методи дослідження – аналіз наукової літератури для визначення основних характеристик веб-ресурсів; аналіз програмного комплексу для реалізації програмного рішення; наукові: метод проектування для розробки концепції веб-порталу.

У результаті були зроблені належні кроки для створення і розробки пакету програмного забезпечення, яке цілком вирішує поставлену задачу. При реалізації були використані сучасні та актуальні інструменти, технології та ресурси.

ІНТЕРНЕТ-РЕСУРС, WEB-ПОРТАЛ, ВЕБ-САЙТ, РОЗРОБКА ІНТЕРНЕТ-РЕСУРСУ, ПРОГРАМНА РОЗРОБКА.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ ТА СКОРОЧЕНЬ.....	6
ВСТУП.....	7
РОЗДІЛ 1. ТЕОРЕТИЧНІ ЗАСАДИ РОЗРОБКИ WEB – ПОРТАЛУ СОЦІАЛЬНОЇ МЕРЕЖІ.....	8
1.1. Структурні характеристики та основи організації соціальної мережі	8
1.2. Дослідження проблеми.....	9
1.3. Дослідження популярності.....	10
1.4. Дослідження існуючих конкурентів.....	13
1.5. Область застосування проекту.....	14
РОЗДІЛ 2. СИСТЕМНІ ВИМОГИ.....	16
2.1. Огляд літератури	16
2.2. Функціональні специфікації	17
2.3. Нефункціональні специфікації	25
2.4. Специфікація програмних інструментів	25
РОЗДІЛ 3. ОСОБЛИВОСТІ ПРОЕКТУВАННЯ	29
3.1. Модульна конструкція	29
3.2. Проектування системи.....	30
РОЗДІЛ 4. ДЕТАЛІ РЕАЛІЗАЦІЇ	44
4.1. Приклад коду	44
4.2. Скріншоти	55
ВИСНОВКИ.....	58
СПИСОК БІБЛІОГРАФІЧНИХ ПОСИЛАНЬ ВИКОРИСТАНИХ ДЖЕРЕЛ.....	59

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ ТА СКОРОЧЕНЬ

World Wide Web – всесвітня мережа

WEB-портал – мережевий портал (ресурс)

URL – уніфікований показник ресурсу, посилання

JQuery – бібліотека JavaScript, створена для спрощення обробки подій.

JavaScript (JS) — об'єктно-орієнтована прототипна мова програмування

MySQL – вільна реляційна система управління базами даних

PHP – мова реалізації веб-сторінок з динамічним вмістом

ВСТУП

На сьогоднішній день інтернет займає величезне місце у нашому житті. Його використання стало звичайною справою, хоча зовсім нещодавно доступ до мережевого зв'язку був величезною рідкістю. Інтернет «пронизує» усі сфери сучасного життя: від навчальних та професійних потреб, до розважальних та соціальних. Він відкрив нові можливості для спілкування з людьми, які знаходяться на великій відстані, а також здійснив шалений ривок для розвитку науки та техніки, допоміг при навчанні та роботі. Одним з найбільших досягнень, які стали можливими завдяки інтернету, можна сміливо вважати соціальні мережі.

Темою дипломного проекту не просто так був обраний саме WEB-портал соціальної мережі. Саме онлайн-спілкування на даний момент є основним способом комунікації більшості людей світу. Ми ділимося думками, відео, фотографіями, музикою та іншим саме завдяки соціальним мережам. Вони здобули шаленої популярності у більшості населення світу. І саме тому, питання актуальності розробки WEB-порталу соціальної мережі навіть і не стоїть.

Завдання проекту полягає у тому, щоб залучити якомога більше людей до сучасного, зручного та приємного засобу онлайн спілкування, завдяки створеному WEB-порталу соціальної мережі.

Об'єктом дослідження є WEB-портал соціальної мережі, з усіма функціями притаманними даному сайту, а саме: модулями Управління профілем та Організацією друзів.

Методом дослідження виступає системний підхід до розробки та WEB-порталу соціальної мережі для багатокористувацького використання.

РОЗДІЛ 1

ТЕОРЕТИЧНІ ЗАСАДИ РОЗРОБКИ WEB – ПОРТАЛУ СОЦІАЛЬНОЇ МЕРЕЖІ

1.1. Структурні характеристики та основи організації соціальної мережі

Соціальні мережі - це найпопулярніший спосіб спілкування 21 століття. Соціальні мережі - це об'єднання людей у певні групи, наприклад, невеликі сільські громади чи спільноти за інтересами. Хоча соціальні зв'язки можливі особисто, на робочих місцях, в університетах чи середніх школах, найпопулярнішим способом зв'язку залишається Інтернет. Це пояснюється тим, що на відміну від більшості середніх шкіл, коледжів чи робочих місць, Інтернет наповнений мільйонами людей, які прагнуть познайомитися з іншими людьми. Без цієї онлайн комунікації нині неможливо уявити сучасне життя.

Соціальна мережа - це відображення та вимірювання взаємозв'язків та потоків між людьми, групами, організаціями, комп'ютерами, URL-адресами та іншими пов'язаними сутностями інформації та знань. Вузлами в мережі є люди та групи, тоді як посилання вказують на взаємозв'язки або потоки між вузлами. Соціальна мережа забезпечує як візуальний, так і математичний аналіз людських стосунків.

Модуль УПРАВЛІННЯ ПРОФІЛЕМ підтримує профіль користувача, такі функції як: ім'я, лайки, дизлайки, хобі, статус, тощо.

Модуль «ОРГАНІЗАЦІЯ ДРУЗІВ» веде список друзів, обробляє запит у друзі та надсилає запит іншому користувачеві.

Кафедра КІТ (47)				НАУ 21 34 83 000 ПЗ			
Розробив	Корнієнко В.Д.			Теоретичні засади розробки web – порталу соціальної мережі	Літера	Аркуш	Аркушів
Керівник	Зіатдінов Ю.К..					8	8
Консультант					412 122		
Н-контролер	Шевченко О.П.						
Зав. каф.	Савченко А.С.						

Профілі та списки друзів - дві ключові особливості на сайтах соціальних мереж. Третя - це функція публічного коментування ("Відгуки", "Коментарі", "Стіна"). Ця функція дозволяє людям коментувати профілі своїх друзів. Ці коментарі відображаються на стіні та є видимими для всіх, хто має доступ до цього профілю.

1.2. Дослідження проблеми

Ми визначаємо сайти соціальних мереж як веб-сервіси, які дозволяють людям створювати публічний або приватний образ в рамках обмеженої системи, складати список інших користувачів, з якими вони підтримують зв'язок, переглядати і переміщатися за списком своїх, та створених іншими людьми зв'язків в рамках системи. Характер і вигляд цих зв'язків можуть варіюватися від сайту до сайту.

З моменту своєї появи сайтів соціальних мереж, таких як MySpace, Facebook і Cyworld, було залучено мільйони користувачів, багато з яких включили ці сайти в своє повсякденне життя. На момент написання існують тисячі сайтів з різними технологічними можливостями, що підтримують широкий спектр інтересів і практик. Хоча їх ключові технологічні особливості досить постійні, культури, що виникають навколо сайтів соціальних мереж - різноманітні. Більшість сайтів підтримують вже існуючі соціальні мережі, але інші допомагають незнайомим людям спілкуватися на основі спільних інтересів, політичних поглядів або діяльності. Деякі сайти розраховані на різноманітну аудиторію, інші привертають людей на основі спільної мови або загальної расової, сексуальної, релігійної чи національної ідентичності. Сайти також різняться за рівнем впровадження нових інформаційних і комунікаційних інструментів, таких як мобільний зв'язок, ведення блогів, обмін фотографіями і відео.

Сайти соціальних мереж - це не тільки можливість спілкуватися або взаємодіяти з іншими людьми по всьому світу, а й один з ефективних способів просування бізнесу. Багато ділових людей в наші дні ведуть бізнес в Інтернеті і використовують ці соціальні мережі, щоб відповідати на запити клієнтів. Це не просто сайт соціальних мереж, який використовується для спілкування з друзями, а величезне джерело інформації про наше повсякденне життя.

WEB-портал соціальної мережі - це онлайн-сервіс, платформа або сайт, який спрямований на сприяння створенню соціальних мереж або соціальних відносин між людьми, які, наприклад, мають спільні інтереси, діяльність, походження або зв'язок в реальному житті. Сервіс соціальної мережі складається з представлення кожного користувача (зазвичай це профіль), його соціальних зв'язків і різних додаткових функцій. Більшість сервісів соціальних мереж засновані на веб-технологіях і надають користувачам засоби для взаємодії через Інтернет, такі як електронна пошта і миттєвий обмін повідомленнями.

Послуги онлайн-спільнот іноді розглядаються як послуги соціальних мереж, хоча в більш широкому сенсі послуги соціальних мереж зазвичай означають послуги, орієнтовані на індивідуума, в той час як послуги онлайн-спільнот орієнтовані на групу. Сайти соціальних мереж дозволяють користувачам обмінюватися ідеями, діями, подіями та інтересами в рамках своїх індивідуальних мереж.

1.3. Дослідження популярності

Станом на травень 2013 року майже три чверті (72%) дорослих користувачів, що знаходяться онлайн, користуються соціальними мережами, в порівнянні з 67% в кінці 2012 року. Коли вперше почали питати про сайти соціальних мереж в лютому 2005 року, лише 8% дорослих користувачів Інтернету відповіли, що користуються сайтами соціальних мереж.

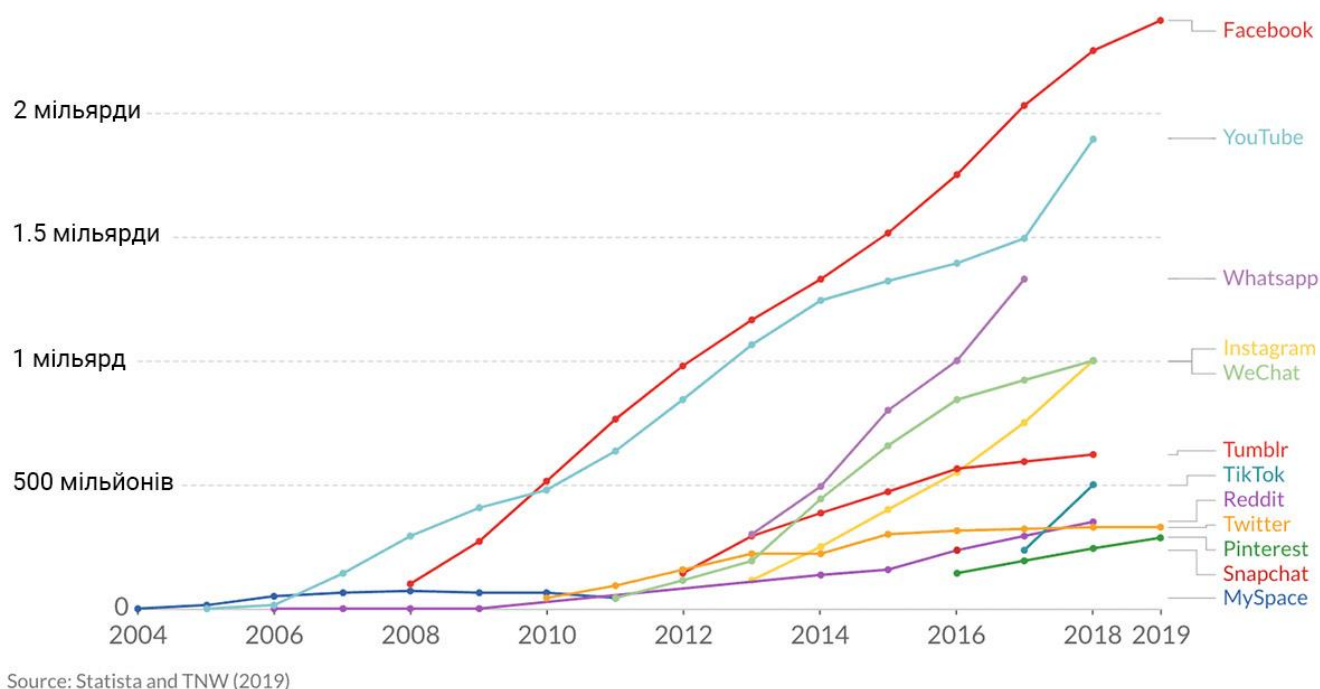


Рис. 1.1. Число людей, які користуються соціальними мережами

Сьогодні використання сайтів соціальних мереж є одним з основних видів діяльності для користувачів Інтернету з найрізноманітніших демографічних груп. Особливо активними користувачами є молоді люди, але соціальні мережі продовжують набирати популярність і серед людей старшого віку. Шість з десяти інтернет-користувачів у віці 50-64 років є користувачами соціальних мереж, як і 43% користувачів у віці 65 років і старше. Хоча літні люди рідше, ніж інші вікові групи, користуються соціальними мережами, за останні чотири роки частка користувачів у віці 65 років і старше збільшилася в три рази (з 13% навесні 2009 року до 43% в даний час).

Основними типами соціальних мереж є ті, які містять місця категорій (наприклад, колишній шкільний рік або однокласники), засоби для зв'язку з друзями (зазвичай зі сторінками самоопису) і систему рекомендацій, пов'язану з активністю користувача. Популярні методи в даний час об'єднані у таких соціальних мережах, як Facebook, Instagram, YouTube, LinkedIn, Pinterest, Tumblr і Twitter, широко використовуваних у всьому світі; Nexopia в Канаді; Badoo,

ВКонтакте, Однокласники (Росія), Delphi (також званий Delphi Forums), Draugiem.lv (в основному в Латвії), Hi5 (Європа), Nyves (в основному в Нідерландах), iWiW (в основному в Угорщині), Nasza-Klasa, Soup (в основному в Польщі), Glocals в Швейцарії, Skyrock, The Sphere, StudiVZ (в основному в Німеччині), Tagged, Tuenti (в основному в Іспанії) і XING в деяких частинах Європи; Hi5 і Orkut в Південній і Центральній Америці; Mxit в Африці; Cyworld, Mixi, Orkut, renren, weibo і Wretch в Азії і на островах Тихого океану.

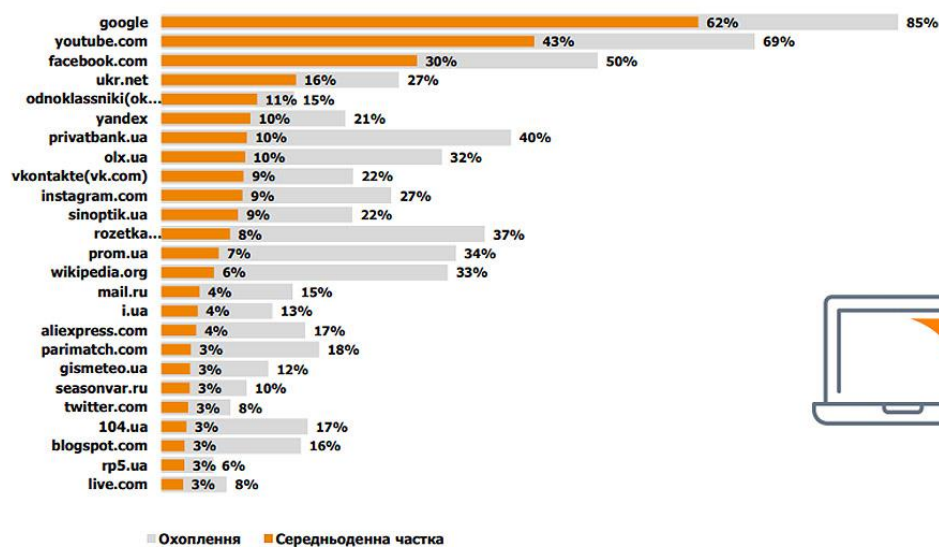


Рис. 1.2. Топ сайтів за відвідуваністю на території України

Багато з цих найперших спільнот фокусувалися на об'єднанні людей для спілкування один з одним через чати, а також заохочували користувачів ділитися особистою інформацією та ідеями через персональні веб-сторінки, надаючи прості у використанні інструменти публікації і безкоштовний або недорогий засіб для цього. Деякі спільноти - наприклад, Classmates.com - використовували інший підхід, просто пропонуючи людям зв'язуватися один з одним через адреси електронної пошти.

В кінці 1990-х років профілі користувачів стали центральною особливістю сайтів соціальних мереж, дозволяючи користувачам складати списки "друзів" і

шукати інших користувачів зі схожими інтересами. До кінця 1990-х років були розроблені нові методи створення соціальних мереж, і багато сайтів почали розробляти більш просунуті функції, що дозволяють користувачам знаходити друзів і управляти ними. Нове покоління сайтів соціальних мереж почало процвітати з появою SixDegrees.com в 1997 році, потім Makeoutclub у 2000 році, HubCulture і Friendster в 2002 році і незабаром стали частиною інтернет-мейнстріму. За Friendster пішли MySpace і LinkedIn роком пізніше, і, врешті-решт, Bebo. Friendster став дуже популярний на тихоокеанських островах. Orkut став першою соціальною мережею в Бразилії і швидко розвивався в Індії (Madhavan, 2007). Про швидке зростання популярності соціальних мереж свідчить той факт, що до 2005 року, за повідомленнями, MySpace отримував більше переглядів сторінок, ніж Google. Facebook, запущений в 2004 році, на початку 2009 року став найбільшим сайтом соціальних мереж в світі. Facebook був вперше представлений (в 2004 році) як Гарвардська соціальна мережа (Cassidy, 2006).

1.4. Дослідження існуючих конкурентів

Відповідно до наведеного вище визначення, перший відомий сайт соціальної мережі з'явився в 1997 році. SixDegrees.com дозволяв користувачам створювати профілі, складати списки друзів і, починаючи з 1998 року, переглядати списки друзів. Звичайно, кожна з цих функцій в тій чи іншій формі існувала і до SixDegrees. Списки друзів існували на більшості великих сайтів знайомств і на багатьох сайтах спільнот. Списки друзів AIM і ICQ підтримували списки друзів, хоча ці друзі не було видно іншим.

SixDegrees був першим, хто об'єднав ці функції. SixDegrees рекламував себе як інструмент, що допомагає людям спілкуватися і відправляти повідомлення іншим людям. Хоча SixDegrees привернув мільйони користувачів, він не зміг стати стійким бізнесом, і в 2000 році сервіс заклався. Озираючись

назад, його засновник вважає, що SixDegrees просто випередив свій час (А. Вайнрайх у особистому спілкуванні, 11 липня 2007 року). Хоча люди вже були знайомі з Інтернетом, у більшості з них не було широких мереж друзів, які були б онлайн. Перші користувачі сайту Classmates.com пояснювали, що після прийняття запитів на додавання в друзі мало що можна було зробити, і більшість користувачів не були зацікавлені в знайомстві з незнайомцями.

Classmates.com дозволив людям приєднатися до своєї середньої школи або коледжу і шукати в мережі інших людей, які також приєдналися до них, але користувачі не могли створювати друзів або складати списки друзів тільки за роком випуску.

З 1997 по 2001 рік ряд веб порталів почали підтримувати різні комбінації імен і публічні списки друзів. AsianAvenue, BlackPlanet і MiGente дозволяли користувачам створювати особисті, професійні зв'язки та знайомства - користувачі могли вказувати друзів в своїх особистих профілях, не запрошуючи схвалення на ці зв'язки (О. Walt, особисте спілкування, 16 серпня 2007 року). Аналогічно, незабаром після свого запуску в 1999 році LiveJournal вказав односпрямовані зв'язки на сторінках користувачів. Творець LiveJournal підозрює, що він створив цих друзів за зразком списків друзів в системі миттєвого обміну повідомленнями. Корейський сайт віртуальних світів Cyworld був створений в 1999 році і додав функцію списку друзів в 2001 році, незалежно від цих інших сайтів.

1.5. Область застосування проекту

Система WEB-порталу соціальної мережі надає користувачам низку функціональних можливостей, таких як:

- Можливість реєструвати різні типи профілів, такі як соціальний, особистий, загальний, професійний.

- Дозволяє користувачам відправляти своїм друзям повідомлення, зображення та файли даних. Користувач може вести теку даних, надісланих іншим користувачам.
- Надає користувачеві можливість завантажувати фотографії, щоб користувач міг вести власний альбом.
- Надає користувачеві можливість приєднатися до спільнот відповідно до його уподобань.
- Дозволяє користувачеві вести і оновлювати список друзів.
- Дає можливість користувачеві відправити запрошення іншій людині і додати її в свій список друзів на майбутнє.

РОЗДІЛ 2 СИСТЕМНІ ВИМОГИ

2.1. Огляд літератури

Соціальні мережі на базі Інтернету дозволяють об'єднати людей, які поділяють інтереси і діяльність, через політичні, економічні та географічні кордони. За допомогою електронної пошти і миттєвого обміну повідомленнями створюються мережеві спільноти, в яких заохочується економіка дарування та взаємний альтруїзм. Інформація підходить для економіки дарування, тому що інформація є неконкурентним товаром і може бути подарована практично без витрат.

Facebook та інші соціальні мережі все частіше стають об'єктом наукових досліджень. Вчені в багатьох областях почали досліджувати вплив сайтів соціальних мереж, вивчаючи, як такі сайти можуть впливати на питання ідентичності, конфіденційності, соціального капіталу, молодіжної культури та освіти.

Кілька веб-сайтів починають використовувати можливості моделі соціальних мереж для благодійності. Такі моделі надають кошти для об'єднання розрізаних галузей і невеликих організацій, що не мають ресурсів для охоплення ширшої аудиторії зацікавлених користувачів. Соціальні мережі надають людям інший спосіб цифрового спілкування. Ці гіпертекстові спільноти дозволяють обмінюватися інформацією та ідеями - стара концепція, вміщена в цифровий простір.

Кафедра КІТ (47)				НАУ 21 34 83 000 ПЗ			
Розробив	Корнієнко В.Д.			Системні вимоги	Літера	Аркуш	Аркушів
Керівник	Зіатдінов Ю.К.					16	13
Консультант					412 122		
Н-контролер	Шевченко О.П.						
Зав. каф.	Савченко А.С.						

2.2. Функціональні специфікації

Об'єкт серверу

Клас Server виступає в якості обгортки для всіх серверних функцій нашого сайту соціальної мережі. По суті, він є сполучною ланкою між всією інформацією, такою як акаунти, дані акаунтів, сторінки, замітки і т.д., і нашою базою даних. Коли будь-який інший об'єкт моделі, наприклад, сторінка, витягується з сервера, створюється тимчасова копія. Якщо ця тимчасова копія будь-яким чином змінюється, нова версія повинна бути відправлена на сервер, щоб оновити постійну копію. Причина створення локальних копій полягає в тому, що вся необхідна інформація для об'єкта передається в одному простому у використанні пакеті. Потім оновлення бази даних можна виконати за один раз, відправивши назад цей єдиний об'єкт. Немає необхідності у використанні декількох функцій або функції, що приймає велику кількість параметрів.

Об'єкт облікового запису

Кожен користувач, який хоче використовувати сайт, повинен створити обліковий запис. Це головний клас, який використовують всі інші об'єкти для визначення того, що робить користувач і коли він це зробив. Інформація облікового запису має чотири цілі: зберігати інформацію для входу в систему, зберігати інформацію про друзів, зберігати інформацію про профіль і зберігати інформацію про параметри конфіденційності. При такій великій кількості інформації, яку необхідно відстежувати, клас Account був би дуже великим і незручним для роботи. Тому для полегшення навантаження клас Account був розбитий на три різних класи. Є клас власне облікового запису, який відстежує інформацію про вхід в систему і інформацію про друзів. Він також містить два інших класи. Професійна інформація була передана класу Account Details, а настройки конфіденційності - класу Privacy Settings. Єдиний раз, коли клас Account потребує оновлення, це коли користувач змінює своє ім'я користувача

та/або пароль. Всі інші налаштування обробляються класами Account Details і Privacy Settings.

Об'єкт Account Details

Об'єкт Account Details - це допоміжний клас, створюваний кожен раз, коли створюється новий об'єкт Account. Об'єкт містить всю інформацію, яка відображається в профілі користувача. На сайті користувач може редагувати цю інформацію, змінюючи свій обліковий запис. В цілому, цей клас не має іншого призначення, крім як бути допоміжним класом для свого об'єкта Account.

Об'єкт налаштувань конфіденційності

Об'єкт Privacy Settings - це інший допоміжний клас, створюваний кожен раз, коли створюється новий об'єкт Account. Цей об'єкт містить всі налаштування конфіденційності, які є у користувача, наприклад, хто може переглядати його медіа або призначені для користувача сторінки. Цей клас викликається кожен раз, коли користувач заходить на сторінку або контент, створений іншим користувачем. Однак він не взаємодіє безпосередньо з іншими класами моделі, тільки з їх поданням.

Об'єкт діалогу, об'єкт події, об'єкт посилання, об'єкт замітки і об'єкт сторінки

Ці об'єкти містять унікальну інформацію для певного типу дій, виконуваних користувачем. Всі вони містять посилання на обліковий запис, до якого вони належать. Кожен раз, коли один з цих об'єктів створюється на сервері, запис про його створення додається в базу даних стрічки новин.

Об'єкт повідомлення

Об'єкт Message створюється, коли користувач складає нове повідомлення для відправки другу. Після того, як підтверджується, що друг знаходиться в базі даних, об'єкт Message додає інформацію про нього в базу даних. Коли користувач перевіряє свою поштову скриньку, список повідомлень, які були відправлені користувачеві, буде показаний в порядку убутання часу їх отримання.

Об'єкт Wall Post

Об'єкт Wall Post працює дуже схоже на об'єкт Message. Єдина відмінність полягає в тому, що об'єкти Wall Post доступні для перегляду всім друзям користувача. Ще одна відмінність полягає в тому, що об'єкт Wall Post може містити коментарі. Якщо користувач додає повідомлення на стіну до свого облікового запису, його статус зміниться на нове повідомлення на стіні.

Друзі

Найважливіша функція - можливість додавати і видаляти друзів. На нашому порталі соціальної мережі додавання друзів - досить простий процес. Користувачі можуть ввести ім'я друга в рядок пошуку у верхній частині домашньої сторінки.

Наприклад, користувач А може шукати користувача Б в рядку пошуку. Після клацання на профілі користувача Б користувач А побачить кнопку з написом Відправити запит на дружбу. Натиснувши на неї, користувач А відправить повідомлення користувачу Б про те, що користувач А хоче з ним подружитися. Пропозицію дружби тепер буде видно в списку друзів користувача Б, де він зможе або прийняти його, або проігнорувати, залишивши його там на невизначений час. Якщо користувач Б прийме запит, користувач А буде доданий до списку друзів користувача Б, і навпаки. У дружби є свої переваги. Наприклад, тільки друзі можуть спілкуватися один з одним. Крім того, друзі можуть переглядати будь-яку частину профілю, яка позначений як "тільки для друзів". Нарешті, для перегляду своїх друзів і вхідних запитів на додавання в друзі користувач просто натискає на вкладку "Друзі", в результаті чого з'являється відповідна сторінка.

Створення облікового запису

Коли користувач заходить на сайт вперше, він повинен створити обліковий запис, перш ніж зможе користуватись усіма можливостями сайту. Процес створення облікового запису складається з трьох частин. У першому розділі міститься інформація для входу в систему, яка необхідна користувачеві для

входу. Сюди входять електронна пошта, пароль і підтвердження пароля. Мета підтвердження пароля полягає в тому, щоб переконатися, що користувач випадково не помилився при створенні пароля. У другому розділі міститься інформація про те, хто ви такий, наприклад, ім'я, місцезнаходження та стать. Більшість цих полів необов'язкові, за винятком імені і статі. Це була б не дуже хороша соціальна мережа, якби всі користувалися анонімними іменами. Як тільки користувач натискає кнопку Створити акаунт - новий акаунт, дані облікового запису та налаштування конфіденційності додаються на сервер, і користувач повертається на сторінку входу в систему.

Параметри конфіденційності

Конфіденційність - дуже важлива характеристика для деяких людей, і соціальні мережі не є винятком. Наша система надає три рівня конфіденційності: відкритий, тільки для друзів і закритий. Інформація з відкритим рівнем конфіденційності є публічною і може бути переглянута будь-яким користувачем. На відміну від цього, інформація з закритим рівнем конфіденційності є повністю приватною і може бути переглянута тільки власником облікового запису. Інформація з рівнем конфіденційності "Тільки для друзів" досить зрозуміла. За замовчуванням при створенні нового облікового запису всі рівні конфіденційності встановлюються відкритими. Щоб змінити рівень конфіденційності, користувач може перейти на вкладку "Параметри" у верхньому рядку меню. Поточні функції з рівнями конфіденційності - це обмін медіафайлів, стіна акаунта, сторінки користувача, замітки і загальний рівень конфіденційності.

Реалізація конфіденційності - досить простий процес. Коли користувач вносить зміни в рівень конфіденційності, об'єкт Налаштування конфіденційності в обліковому записі оновлюється з новими налаштуваннями конфіденційності. Ця робоча копія потім передається в метод оновлення сервера, який запитує базу даних, щоб встановити запис для цього об'єкта налаштувань конфіденційності на нові значення. Коли користувач відвідує іншу сторінку, власник цієї сторінки

витагується з бази даних. У функцію `can view page ()` передається власник профілю, профіль глядача і тип сторінки. У функції об'єкт налаштувань конфіденційності витагується з профілю глядача, а тип сторінки визначає, яка настройка конфіденційності витагується з об'єкта. Якщо параметр відкритий, функція автоматично повертає `true`. Якщо настройка тільки для друзів, функція повертає `true`, тільки якщо глядач профілю є одним друзів власника сторінки або якщо глядач профілю і власник сторінки - одна і та ж людина. Нарешті, якщо параметр закритий, функція повертає `true`, тільки якщо глядач профілю є власником сторінки. Коли функція поверне відповідь, сторінка виконає одну з таких дій. Якщо функція повертає `true`, то глядачеві дозволено переглядати цю сторінку, і вона завантажиться нормально. Якщо функція поверне `false`, то глядач не повинен переглядати цю сторінку, і його буде перенаправлено на сторінку помилки.

Чат

Одним з переваг наявності друзів є можливість спілкуватися з ними в чаті. Коли друг знаходиться онлайн, він з'являється на вкладці чату як доступний для спілкування. Щоб почати розмову, потрібно просто натиснути на ім'я друга, що призведе до початку нового сеансу чату. Маленька смужка внизу зміниться з відключеного чату на чат з ім'ям друга. Крім того, друг, який приймає чат, отримає повідомлення про те, що почався новий сеанс чату. Після цього двоє друзів можуть спілкуватися один з одним до тих пір, поки один або обидва не вийдуть із чату.

Функція чату трохи складніше інших, оскільки для її роботи потрібно поєднання JavaScript і JQuery. Коли користувач натискає на ім'я друга, щоб почати спілкування, запит на чат відправляється в базу даних. На кожній сторінці є функція JQuery, яка раз в секунду запитує сервер на предмет нових запитів чату. Це повинно бути, щоб клієнт міг запросити сервер виконати перевірку і повернути будь-яку необхідну інформацію. Якби це був виключно PHP, сторінка ніколи б не завантажилася повністю, тому що вона не може завантажитися, поки

сервер не виконає всю попередню обробку. JavaScript-частина JQuery дозволяє клієнту попросити сервер виконати додаткову обробку після завантаження сторінки. Коли користувач відправляє повідомлення, викликається інша функція JQuery, щоб попередити сервер про необхідність оновити розмову новим повідомленням. У той же час виконується ще одна функція JQuery, яка раз в секунду запитує сервер, чи була оновлена сесія чату. В цілому, клієнт запитує у сервера всі існуючі зміни.

Обмін повідомленнями

На відміну від чату, користувачі можуть відправити повідомлення будь-якому іншому користувачеві. Для того щоб користувач А відправив повідомлення користувачу Б, він просто переходить на вкладку "Центр повідомлень" і натискає кнопку "Створити повідомлення". Потім він заповнить форму, вказавши електронну пошту одержувача, заголовок повідомлення і його зміст. Після цього нове повідомлення з'явиться в папці вхідних повідомлень користувача Б в центрі повідомлень.

Обмін повідомленнями здійснюється шляхом зберігання записів про повідомленні на сервері. Спочатку створюється об'єкт повідомлення, в який заносяться дані облікових записів "Кому" і "Від кого", тіло повідомлення і дата відправки. Тіло повідомлення витягується з форми, заповненої користувачем, обліковий запис "від" вилучають із сесії користувача, а обліковий запис одержувача витягується з адреси електронної пошти одержувача. Адреса одержувача перевіряється, щоб переконатися, що відправник не намагався відправити повідомлення неіснуючому одержувачу. Нарешті, нове повідомлення зберігається на сервері. Отримання повідомлень, відправлених користувачеві, здійснюється простим методом запиту сервера на наявність повідомлень, які були відправлені користувачеві. Це робиться шляхом звірки ідентифікатора одержувача кожного повідомлення з ідентифікатором облікового запису користувача. Всі збіги повертаються в папку вхідних повідомлень користувача.

Події

Крім відправки повідомлень своїм друзям, сайти соціальних мереж є відмінним місцем для оповіщення своїх друзів про важливі майбутні події. Наприклад, якщо користувач влаштовує вечірку з нагоди дня народження, він може створити подію і запросити друзів, яких він хоче побачити, або просто зробити її загальнодоступною, відкритою для всіх бажаючих. Процес створення події простий: потрібно заповнити спеціальну форму.

Після заповнення форми користувач потрапляє на стандартну сторінку реєстрації. Потім він може переглянути нову подію, перейшовши на вкладку Події і натиснувши на нову подію. Якщо користувач хоче відредагувати інформацію про подію, він може натиснути на посилання "Редагувати" поруч з подією, де з'явиться форма, аналогічна формі створення події, з усіма полями, заповненими поточною інформацією про подію.

Збереження події - досить простий процес. Після відправки даних форми створюється новий об'єкт Event, в якому зберігається обліковий запис, що створив подію, а також вся інформація, відправлена разом з формою. Потім ця нова подія відправляється на сервер, який створює запит до бази даних і зберігає поля об'єкта події в еквівалентні поля таблиці бази даних подій. Витяг подій також є досить простим процесом. Викликається запит MySQL для вилучення вмісту події в масив, і цей масив використовується для створення нової робочої копії події в об'єкті Event.

Завантаження медіафайлів

У міру використання облікового запису в соціальній мережі люди захочуть мати можливість завантажувати кумедні або цікаві зображення, відео, музику, тощо, щоб поділитися ними зі своїми друзями. Розділ завантаження медіа буде розташований на вкладці медіа, де користувачі зможуть вказати файли для завантаження, а також надати короткий опис завантажуваним файлам. Щоб запобігти завантаженню потенційно шкідливих файлів, таких як виконувані файли, підтримуються тільки певні розширення файлів. Ці дозволені

розширення охоплюють популярні розширення зображень, такі як jpg, png, gif та bitmap, розширення відео mp4 і avi, розширення аудіо mp3, wma і wav, а також деякі розширення документів, включаючи doc, txt, rtf і pdf.

Внутрішня робота цього процесу завантаження напрочуд проста, оскільки HTTP-сервер автоматично запитує файл у клієнта. Після того як користувач натискає кнопку "Завантажити", файл відправляється на сервер і зберігається у тимчасовому місці. Через PHP вся інформація про ці тимчасові файли може бути доступна через змінну \$ FILES. Тимчасове місце розташування зберігається в цій змінній і може бути використане для отримання імені файлу і його розширення. Яке потім перевіряється по масиву дозволених розширень. Якщо розширення є в списку, то файл переміщається в постійне місце в папці мультимедіа у вкладеній папці облікового запису користувача. Якщо розширення немає в списку, воно залишається в тимчасовій папці, звідки буде автоматично видалено сервером після завершення роботи php-скрипта.

Інші можливості

Всі інші функції, такі як сторінки стіни і замітки, досить схожі. Їх основна мета - додати додаткові місця для розміщення додаткового контенту. Всі вони створюються аналогічними методами і зберігаються однаково, але використовуються для різних цілей.

1. Стіна

Стіна - це місце, де власник аккаунта може висловити свої поточні почуття або думки за допомогою статусу стіни. Інші користувачі можуть додати свої думки, створюючи повідомлення на стіні, які відображаються під статусом. По суті, це спосіб для багатьох користувачів публічно спілкуватися або просто залишати свої думки один про одного.

2. Нотатки

Нотатки дуже схожі на повідомлення на стіні за одним великим виключенням. На відміну від повідомлень на стіні, які можуть бути створені будь-яким користувачем, замітки може створювати тільки власник облікового

запису. Їх основне призначення - створювати нагадування для власника аккаунта або оголошення для інших користувачів.

2.3. Нефункціональні специфікації

Важливі, але не функціональні риси, яким відповідає розробка:

- Безпечний доступ до конфіденційних даних по логіну користувача і паролю. Цей додаток безпечний для всіх користувачів, так як якщо користувач вийде з будь-якої сесії, то ніхто не зможе отримати доступ до його профілю, не знаючи його конфіденційного пароля.
- Доступність 24 на 7.
- Краща конструкція компонентів для досягнення найкращої продуктивності в час пікових навантажень.
- Використовувана тут база даних є міцною, надійною і швидкою. Тому користувачам не доведеться чекати відгуку дуже довго.
- Доступ до цього додатка можливий з платформ будь-якого типу.
- У базі даних немає надмірності, тому вона не займає зайвого місця в пам'яті.
- Ім'я користувача і пароль висилаються користувачам електронною поштою після реєстрації.
- Також передбачена система відновлення пароля в разі його забування.

2.4. Специфікація програмних інструментів

MySQL

MySQL є популярним вибором бази даних для використання в веб-додатках і є центральним компонентом широко використовуваного стека програмного забезпечення для веб-додатків з відкритим вихідним кодом LAMP

(і інших стеків 'AMP'). LAMP - це аббревіатура від "Linux, Apache, MySQL, Perl / PHP / Python". Вільні програмні проекти з відкритим вихідним кодом, яким потрібна повнофункціональна система управління базами даних, часто використовують MySQL.

MySQL може бути зібраний і встановлений вручну з вихідного коду, але це може бути дуже довго та виснажливо, тому найчастіше він встановлюється з бінарного пакета, якщо не потрібні спеціальні налаштування. У більшості дистрибутивів Linux система управління пакетами може завантажити і встановити MySQL з мінімальними зусиллями, хоча часто потрібне додаткове налаштування параметрів безпеки і оптимізації.

Хоча MySQL створювався у якості дешевої альтернативи більш потужним власним базам даних, він поступово розвивався, щоб підтримувати і більш масштабні потреби. Він як і раніше найчастіше використовується в невеликих і середніх односерверних розгортаннях, або як компонент веб-додатку на базі LAMP, або як окремий сервер баз даних. Привабливість MySQL багато в чому обумовлена його відносною простотою і легкістю використання, що забезпечується екосистемою інструментів з відкритим вихідним кодом, таких як phpMyAdmin. В середньому діапазоні MySQL можна масштабувати, розгорнувши його на більш потужному обладнанні, наприклад, на багатопроцесорному сервері з гігабайтами пам'яті.

Мова: PHP

PHP - це мова сценаріїв призначена для веб-середовища. Її основне застосування - реалізація веб-сторінок з динамічним вмістом. Останнім часом PHP отримала досить широку популярність і є однією з лідерів руху за програмне забезпечення з відкритим вихідним кодом. Популярність PHP обумовлена її синтаксисом, схожим на C, і її простотою. В даний час PHP ділиться на дві основні версії: PHP 4 і PHP 5, хоча PHP 4 є застарілою версією і більше не розробляється і не поставляється з виправленнями критичних помилок. PHP 6 знаходиться в стадії розробки.

PHP дозволяє статичній веб-сторінці стати динамічною. "PHP" - це аббревіатура, яка розшифровується як "PHP: передпроцесор гіпертексту". Слово "передпроцесор" означає, що PHP вносить зміни до створення HTML-сторінки. Це дозволяє розробникам створювати потужні програми, які можуть публікувати блог, дистанційно керувати обладнанням або запускати потужні веб-сайти, такі як Wikipedia або Wikibooks. Зрозуміло, для створення подібних додатків необхідна база даних, наприклад MySQL.

Код PHP інтерпретується веб-сервером з процесорним модулем PHP, який генерує результуючу веб-сторінку: Команди PHP можуть бути вбудовані безпосередньо в вихідний документ HTML, а не викликати зовнішній файл для обробки даних.

Спочатку мова PHP була реалізована у вигляді інтерпретатора, і це як і раніше найпопулярніша реалізація. Було розроблено кілька компіляторів, які відокремлюють мову PHP від інтерпретатора. Переваги компіляції включають кращу швидкість виконання, статичний аналіз і поліпшену сумісність з кодом, написаним на інших мовах.

До складу PHP входять безкоштовні бібліотеки з відкритим вихідним кодом. PHP - це система, орієнтована на роботу з Інтернетом, з вбудованими модулями для доступу до серверів File Transfer Protocol (FTP), та багатьма базами даних.

Back End: PHP MyAdmin

PHPMyAdmin - це безкоштовний програмний інструмент, написаний на PHP, призначений для адміністрування MySQL через Інтернет. PhpMyAdmin підтримує широкий спектр операцій з MySQL, Maria DB і Drizzle. Часто використовувані операції (управління базами даних, таблицями, стовпцями, відносинами, індексами, користувачами, дозволами і т.д.) можуть бути виконані через призначений для користувача інтерфейс, при цьому у вас залишається можливість безпосередньо здійснювати будь-SQL-запит.

Функції, що надаються програмою, включають:

1. Веб-інтерфейс
2. Управління базою даних MySQL
3. Імпорт даних з CSV і SQL
4. Експорт даних в різні формати: CSV, SQL, XML, PDF (через бібліотеку TCPDF), ISO / IEC 26300 - Open Document Text and Spreadsheet, Word, Excel, LaTeX та інші.
5. Адміністрування декількох серверів
6. Створення графічних зображень макета бази даних в форматі PDF
7. Створення складних запитів за допомогою Query-by-Example (QBE)
8. Глобальний пошук в базі даних або її підмножині

РОЗДІЛ 3 ОСОБЛИВОСТІ ПРОЕКТУВАННЯ

3.1. Модульна конструкція

Додаток складається з наступних основних модулів:

Модуль «Реєстрація»

Цей модуль надає функціональні можливості для тих людей, які хочуть створити акаунт. Користувач може розміщувати свої думки з особистими і професійними цілями. Він також може оновлювати профіль так часто, як це необхідно. Користувач також може переглядати профілі друзів.

Модуль профілю

Цей модуль надає функціональні можливості, пов'язані з профілем користувача. Зареєстровані користувачі можуть бачити свої дані і, якщо вони хочуть змінити будь-яку інформацію, вони можуть відредагувати її.

Модуль адміністратора

Цей модуль надає функціональні можливості, пов'язані з адмініструванням. Адміністратор управляє всім додатком, адмініструє профілі всіх зареєстрованих користувачів і відповідає за їх діяльність.

Кафедра КІТ (47)				НАУ 21 34 83 000 ПЗ			
Розробив	Корнієнко В.Д.			Особливості проектування	Літера	Аркуш	Аркушів
Керівник	Зіатдінов Ю.К.					29	15
Консультант					412 122		
Н-контролер	Шевченко О.П.						
Зав. каф.	Савченко А.С.						

3.2. Проектування системи

3.2.1. Діаграми потоків даних

Діаграми потоків даних моделюють потік даних в інформаційну систему, через неї і з неї:

- показують процеси, які змінюють або перетворюють дані
- показують переміщення даних між процесами
- представляють систему як мережу процесів, які перетворюють дані, що протікають між ними

Екран користувача показує, що побачить користувач сайту. Після успішного входу в систему користувачеві будуть запропоновані різні посилання (наприклад, пошук користувачів, пошук спільнот, перегляд пошти і т.д.), і він зможе вибрати потрібні варіанти або повернутися до себе додому.

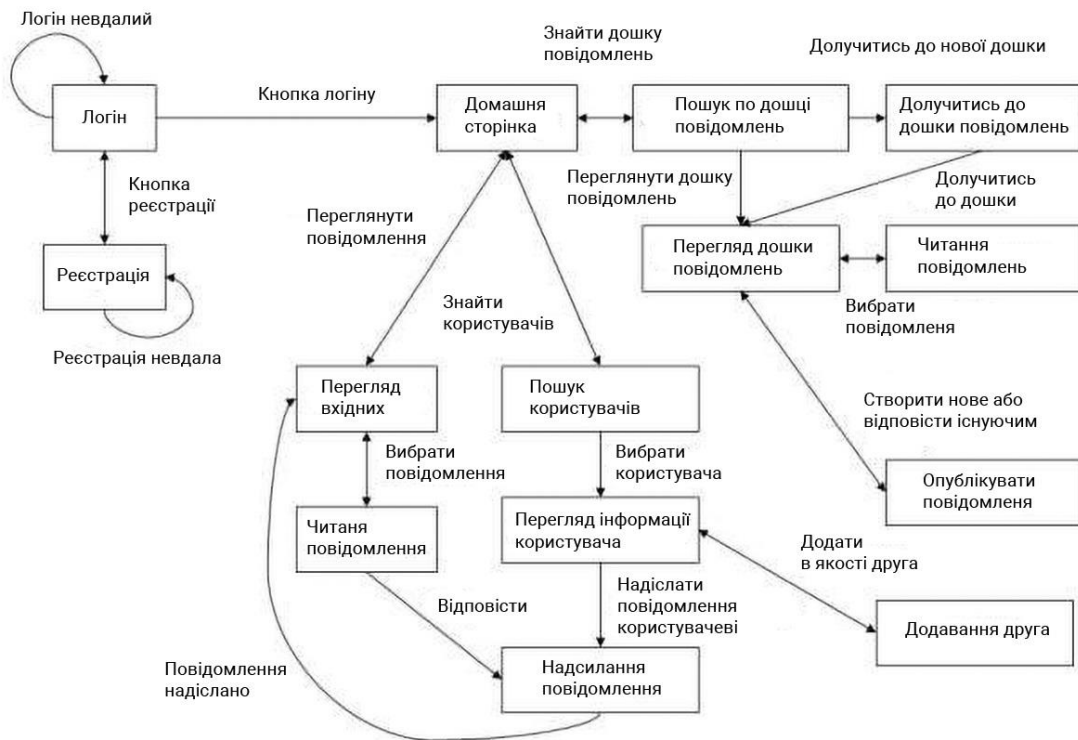


Рис. 3.1. Потік клієнтських екранів

3.2.2 Діаграми варіантів використання.

Діаграма варіантів використання - це графічне зображення взаємодій між елементами системи. Варіант використання - це методологія, яка використовується в системному аналізі для визначення, уточнення і організації вимог до системи. В даному контексті термін "система" відноситься до того, що розробляється або експлуатується, наприклад, веб-сайт з продажу і обслуговування товарів поштою. Діаграми варіантів використання використовуються в UML (Unified Modeling Language), стандартній нотації для моделювання об'єктів і систем реального світу.

Цілі системи можуть включати в себе планування загальних вимог, перевірку апаратного забезпечення, тестування і налагодження розроблюваного програмного продукту, створення довідкової інформації в Інтернеті або виконання завдання, орієнтованого на обслуговування споживачів. Наприклад, сценарії використання в середовищі продажу товарів включають замовлення товару, оновлення каталогу, обробку платежів і роботу з клієнтами.

Діаграма варіантів використання містить чотири компоненти:

- Область, яка визначає, що саме описує система по відношенню до навколишнього світу.
- Дійові особи, зазвичай залучені в роботу системи, що визначаються відповідно до їх ролей.
- Варіанти використання, які представляють собою конкретні ролі, виконувані учасниками всередині і навколо системи.
- Відносини між дійовими особами і варіантами використання.

Домашня сторінка

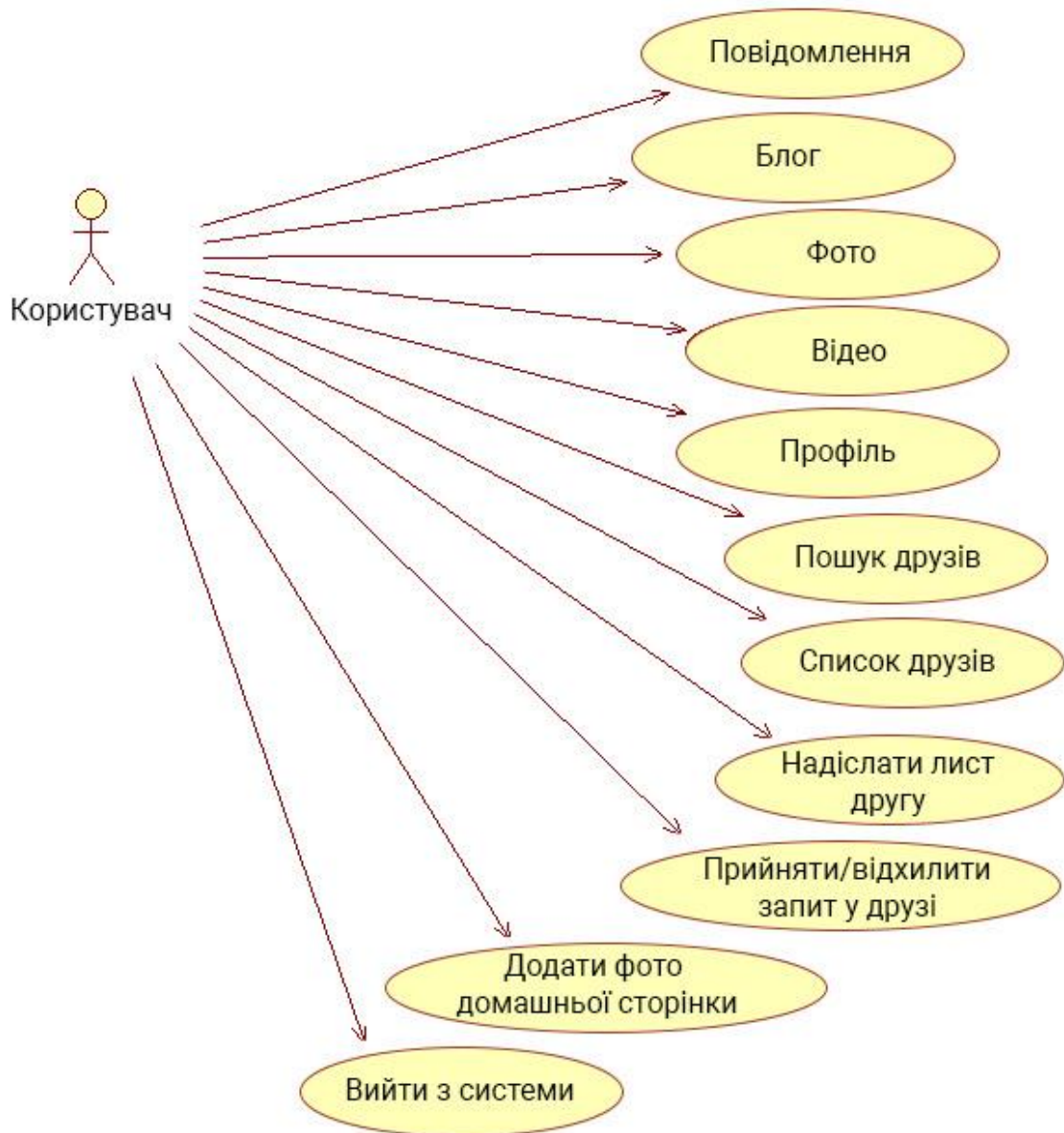


Рис. 3.2. Домашня сторінка

Вхід в систему та реєстрація

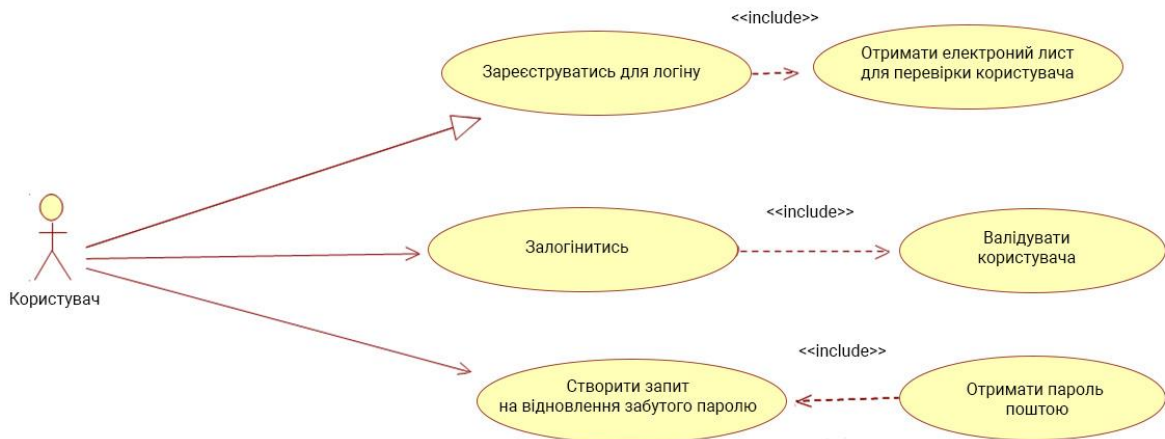


Рис. 3.3. Вхід в систему та реєстрація

Сторінка стіни

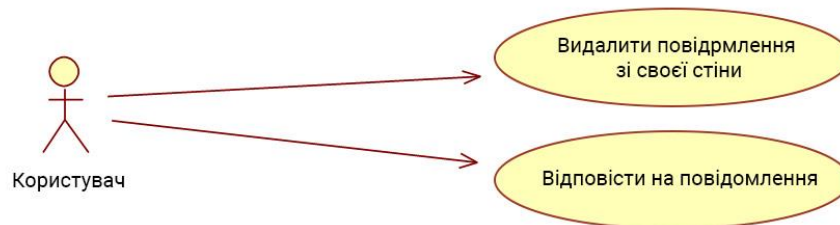


Рис. 3.4. Сторінка стіни

Сторінка блогу

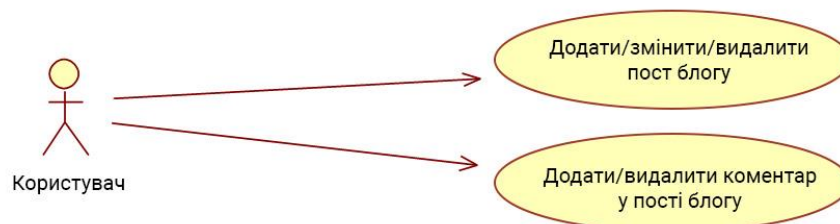


Рис. 3.5. Сторінка блогу

Сторінка профілю

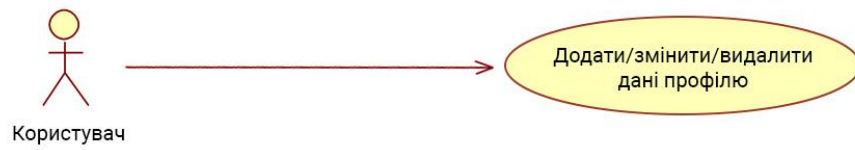


Рис. 3.6. Сторінка профілю

Сторінка пошуку друзів

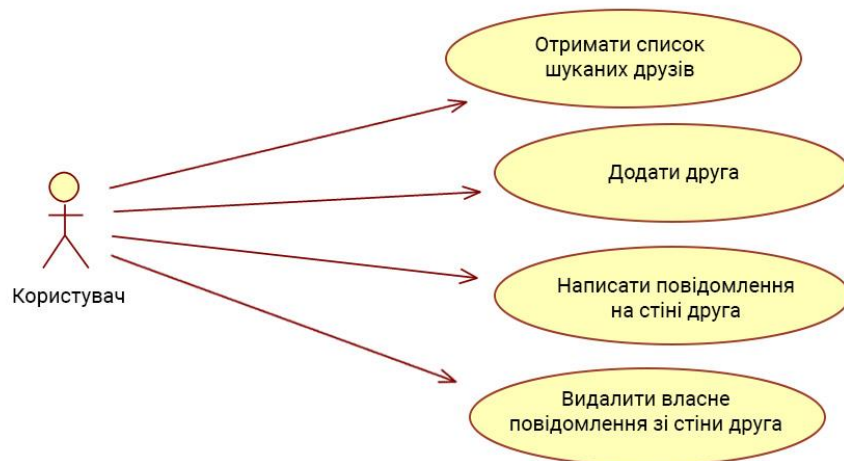


Рис. 3.7. Сторінка пошуку друзів

Сторінка списку друзів



Рис. 3.8. Сторінка списку друзів

Сторінка фото

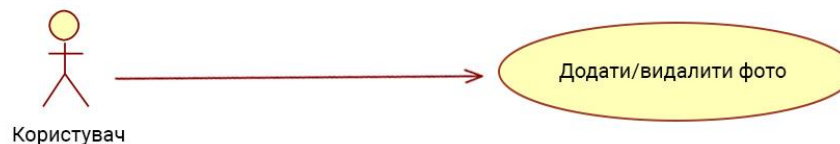


Рис. 3.9. Сторінка фото

3.2.3 Контекстні діаграми

Діаграма потоку даних найвищого рівня - це контекстна діаграма.

- Контекстна діаграма показує взаємодію системи з її оточенням в термінах потоків даних.
- Контекстна діаграма визначає межі системи (область застосування).
- Показані тільки ті потоки даних, які покидають систему, і потоки даних, які приходять ззовні системи.

DFD рівня 0

DFD рівня 0, також звана фундаментальною моделлю системи або контекстною діаграмою, представляє весь елемент програмного забезпечення у вигляді єдиного міхура з вхідними і вихідними даними, позначеними вхідними та вихідними стрілками відповідно.

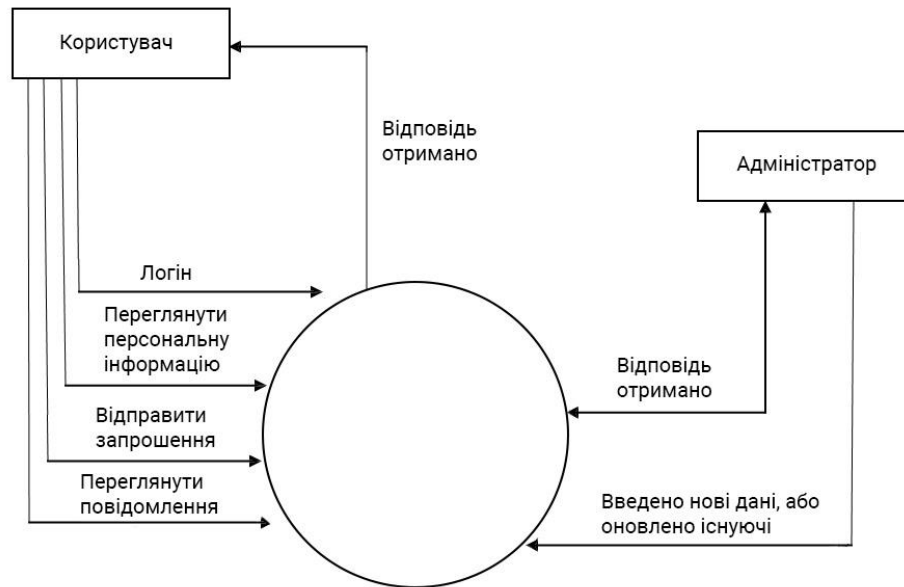


Рис. 3.10. Діаграма DFD нульового рівня

DFD першого рівня

Цей рівень DFD забезпечує більш детальну структуру. Він забезпечує детальне уявлення вимог і потоку даних від одного об'єкту до іншого.

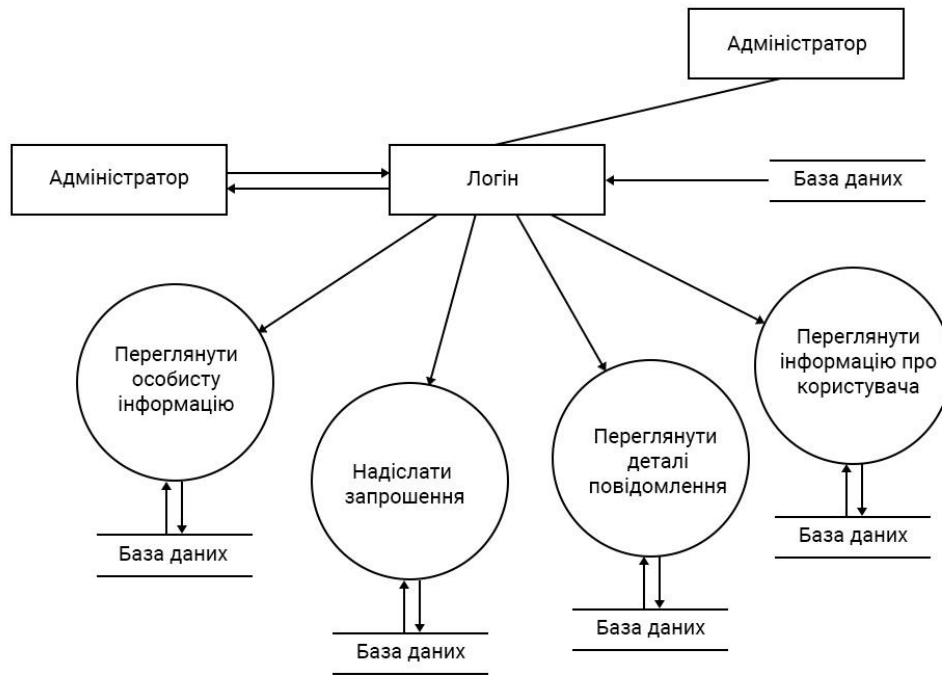


Рис. 3.11 Діаграма DFD першого рівня

3.2.4. Діаграма відношень сутностей

Модель відносин між сутностями - це високорівнева модель даних. Вона заснована на уявленні про реальний світ, та складається з набору базових об'єктів, званих сутностями, і відносин між цими об'єктами. Вона була розроблена для полегшення проектування баз даних, дозволяючи специфікувати схему підприємства, яка являє собою загальну логічну структуру бази даних.

Сутність - це об'єкт, який має своє існування в реальному світі. Він включає в себе всі ті "речі", про які збираються дані. Сутність може бути матеріальним об'єктом, таким як студент, місце або деталь. Вона також може бути нематеріальною, як, наприклад, подія, посада або рахунок клієнта. Наприклад, якщо ми говоримо, що клієнт купує товар, то це означає, що клієнт і товар є сутностями.

На діаграмі сутності зображуються у вигляді прямокутників.

Набір сутностей - це безліч сутностей одного типу, що володіють однаковими властивостями, або атрибутами. Наприклад, набір всіх осіб, які є клієнтами даного банку, можна визначити як набір сутностей "клієнт".

Атрибути - це одиниці, які описують характеристики або властивості сутностей. У базі даних сутності представлені таблицями, а атрибути - стовпцями. Наприклад, сутність клієнт може мати безліч атрибутів, таких як код, ім'я та адреси. Аналогічно, сутність "товар" може мати такі атрибути, як код і ціна. Вони малюються в формі еліпсів разом з прямокутниками сутностей.

Діаграма відносин сутностей системи переписки відображена на наступному рисунку:

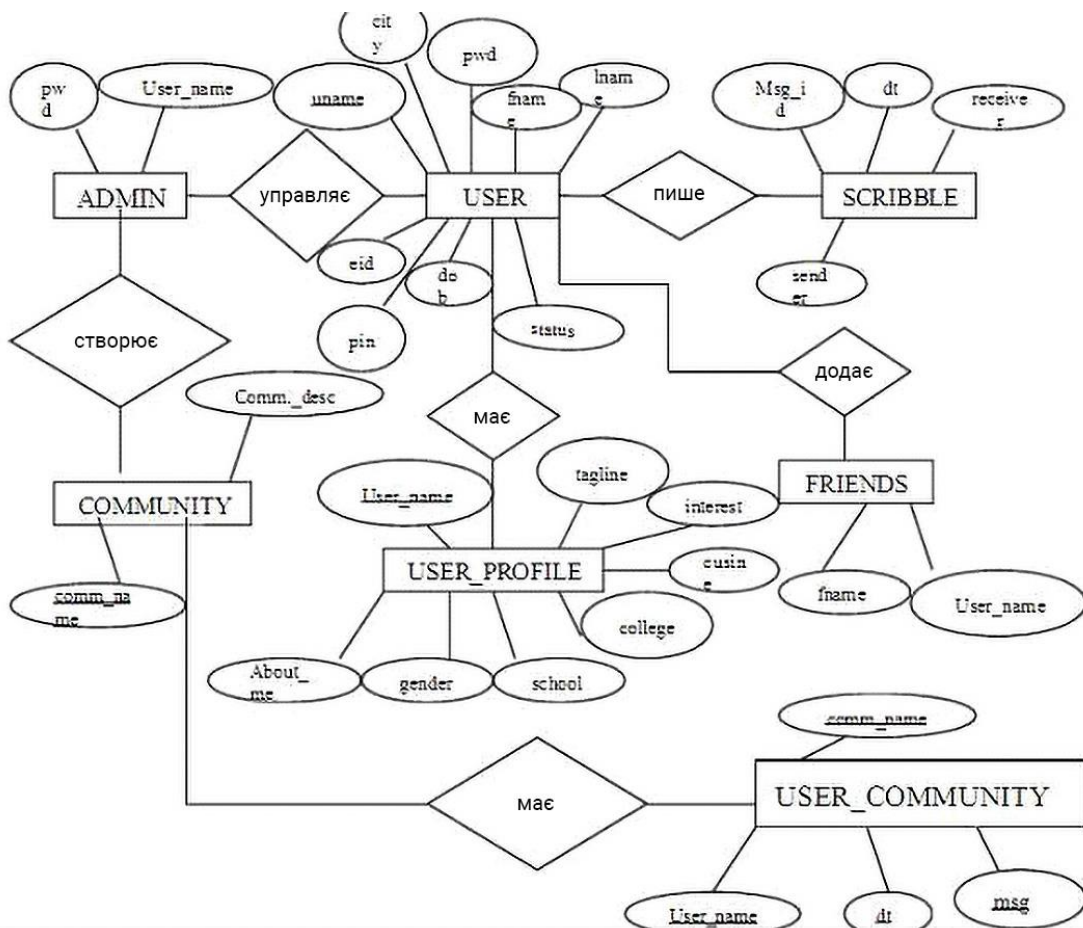


Рис. 3.12. Діаграма відносин сутностей системи

3.2.5 База даних

База даних використовується для зберігання відповідної інформації про людей. База даних - це набір рядків і стовпців, в якому рядки позначають кортеж, а стовпці - область таблиці.

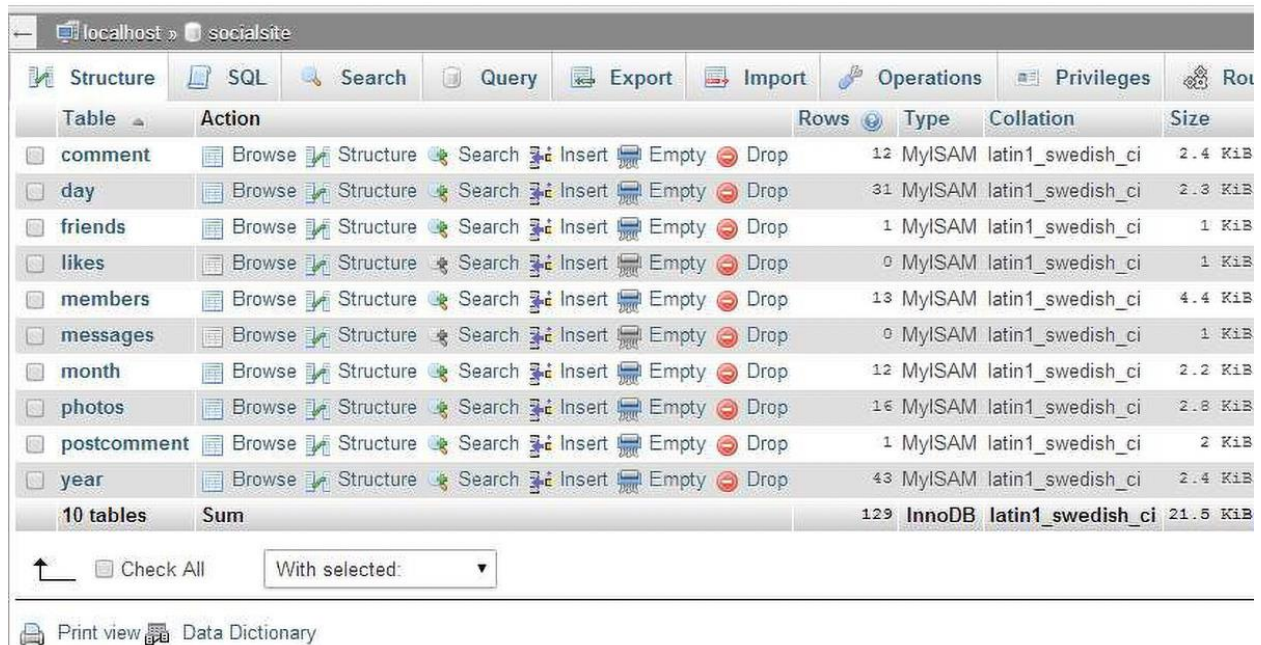


Table	Action	Rows	Type	Collation	Size
comment	Browse Structure Search Insert Empty Drop	12	MyISAM	latin1_swedish_ci	2.4 KiB
day	Browse Structure Search Insert Empty Drop	31	MyISAM	latin1_swedish_ci	2.3 KiB
friends	Browse Structure Search Insert Empty Drop	1	MyISAM	latin1_swedish_ci	1 KiB
likes	Browse Structure Search Insert Empty Drop	0	MyISAM	latin1_swedish_ci	1 KiB
members	Browse Structure Search Insert Empty Drop	13	MyISAM	latin1_swedish_ci	4.4 KiB
messages	Browse Structure Search Insert Empty Drop	0	MyISAM	latin1_swedish_ci	1 KiB
month	Browse Structure Search Insert Empty Drop	12	MyISAM	latin1_swedish_ci	2.2 KiB
photos	Browse Structure Search Insert Empty Drop	16	MyISAM	latin1_swedish_ci	2.8 KiB
postcomment	Browse Structure Search Insert Empty Drop	1	MyISAM	latin1_swedish_ci	2 KiB
year	Browse Structure Search Insert Empty Drop	43	MyISAM	latin1_swedish_ci	2.4 KiB
10 tables	Sum	123	InnoDB	latin1_swedish_ci	21.5 KiB

Рис. 3.13. Список таблиц бази даних

Проектування бази даних - це процес створення докладної моделі даних бази даних. Ця логічна модель даних містить всі необхідні логічні і фізичні параметри проектування та параметри фізичного зберігання даних. Необхідно створити проект на мові визначення даних, який потім може бути використаний для створення бази даних. Повністю атрибутована модель даних містить докладні атрибути для кожної сутності. Термін "проектування бази даних" може використовуватися для опису різних частин проектування всієї системи баз даних. В основному, і найбільш правильно, його можна розглядати як логічне проектування відносин базових структур даних, використовуваних для зберігання даних. У реляційній моделі це класи та іменовані відносини. Однак

термін "проектування бази даних" можна також використовувати для позначення загального процесу проектування не тільки базової структури даних, але і форм і запитів, які використовуються як частина загальної програми бази даних в системі керування базою даних (СКБД).

3.2.6. Словник даних

Таблиця 3.1

Коментар

Ключ	Колонка	Тип	Null	Початкове значення
PRIMARY	comment_id	int(11)	-	-
	comment	text	-	-
	date_created	varchar(50)	-	-
	member_id	varchar(30)	-	-

Таблиця 3.2

День

Ключ	Колонка	Тип	Null	Початкове значення
PRIMARY	day_id	int(11)	-	-
	day	Int(2)	-	-

Таблиця 3.3

Друзі

Ключ	Колонка	Тип	Null	Початкове значення
PRIMARY	member_id	int(11)	-	-
	datetime	datetime	-	-
	status	varchar(11)	-	-
	friends_with	int(11)	-	-

Таблиця 3.4

Лайки

Ключ	Колонка	Тип	Null	Початкове значення
PRIMARY	like_id	int(11)	-	-
	remarks	datetime	-	-
	remarksby	varchar(11)	-	-

Таблиця 3.5

Учасники

Ключ	Колонка	Тип	Null	Початкове значення
PRIMARY	member_id	int(11)	-	-
	username	varchar(11)	-	-
	password	varchar(80)	-	-
	firstname	varchar(30)	-	-
	lastname	varchar(30)	-	-
	address	varchar(200)	-	-
	contactno	varchar(14)	-	-
	url	varchar(100)	-	-
	birthdate	varchar(20)	-	-
	gender	varchar(6)	-	-
	dateadded	varchar(45)	-	-
	profimage	varchar(200)	-	-
	curcity	varchar(50)	-	-
	hometown	varchar(50)	+	NULL
	interested	varchar(30)	+	NULL
	language	varchar(30)	+	NULL
	college	varchar(100)	+	NULL
	school	varchar(200)	+	NULL
	arts	text	+	NULL
	aboutme	text	+	NULL

Продовження таблиці 3.5

	month	varchar(20)	+	NULL
	day	varchar(2)	+	NULL
	year	varchar(4)	+	NULL
	stats	varchar(30)	+	NULL

Таблиця 3.6

Повідомлення

Ключ	Колонка	Тип	Null	Початкове значення
PRIMARY	message_id	int(11)	-	-
	receiver	varchar(40)	-	-
	recipient	varchar(40)	-	-
	datetime	datetime	-	-
	content	varchar(100)	-	-
	status	varchar(6)	-	-

Таблиця 3.7

Місяць

Ключ	Колонка	Тип	Null	Початкове значення
PRIMARY	month_id	int(11)	-	-
	month	varchar(11)	-	-

Таблиця 3.8

Фото

Колонка	Тип	Null	Початкове значення
photo_id	int(11)	-	-
location	varchar(200)	-	-
member_id	int(11)	-	-

Коментар посту

Колонка	Тип	Null	Початкове значення
comment_id	int(11)	-	-
content	text	-	-
commentedby	varchar(30)	-	-
pic	varchar(100)	-	-
id	int(40)	-	-
date_created	varchar(50)	-	-

РОЗДІЛ 4

ДЕТАЛІ РЕАЛІЗАЦІЇ

4.1. Приклад коду

4.1.1. Homepage

```
if ( ! defined( 'ABSPATH' ) )
exit;

class WP_MyNet {
public function    construct() {

define( 'MY_NET_VERSION', '1.7.3' );
define( 'MY_NET_PLUGIN_DIR', untrailingslashit(
plugin_dir_path(    FILE    ) ) );
define( 'MY_NET_PLUGIN_URL', untrailingslashit( plugins_url(
basename( plugin_dir_path(    FILE    ) ), basename(    FILE    ) ) ) );

include( 'wp-my_net-functions.php' );
include( 'wp-my_net-template.php' );
include( 'includes/class-wp-my_net-post-types.php' );
include( 'includes/class-wp-my_net-ajax.php' );
include( 'includes/class-wp-my_net-shortcodes.php' );
include( 'includes/class-wp-my_net-api.php' );
include( 'includes/class-wp-my_net-forms.php' );
include( 'includes/class-wp-my_net-geocode.php' );

if ( is_admin() )
include( 'includes/admin/class-wp-my_net-admin.php' );
```

Кафедра КІТ (47)

НАУ 21 34 83 000 ПЗ

Розробив	Корнієнко В.Д.			Літера			Аркуш	44	Аркушів	15
Керівник	Зіатдінов Ю.К..			Деталі реалізації						
Консультант										
Н-контролер	Шевченко О.П.									
Зав. каф.	Савченко А.С.									
412 122										

```

$this->forms      = new WP_MyNet_Forms();
$this->post_types = new WP_MyNet_Post_Types();

register_activation_hook( basename( dirname( FILE ) ) . '/' . basename( FILE ), array(
$this->post_types, 'register_post_types' ), 10 );

register_activation_hook( basename( dirname(

FILE

) ) . '/' .
basename(

FILE

), create_function( "", "include_once( 'includes/class-wp-my-net-install.php' );" ), 10 );
register_activation_hook( basename( dirname( FILE ) ) . '/' . basename( FILE ),
'flush_rewrite_rules', 15 );
), 10 );

add_action( 'plugins_loaded', array( $this, 'load_plugin_textdomain' ) );
add_action( 'switch_theme', array( $this->post_types, 'register_post_types'
add_action( 'switch_theme', 'flush_rewrite_rules', 15 );
add_action( 'widgets_init', create_function( "", "include_once(
'includes/class-wp-my_net-widgets.php' );" ) );
add_action( 'wp_enqueue_scripts', array( $this, 'frontend_scripts' ) );
add_action( 'admin_init', array( $this, 'updater' ) );
}

public function updater() {
if ( version_compare( MY_NET_VERSION, get_option(
'WP_MyNet_version' ), '>' ) )
include_once( 'includes/class-wp-my_net-install.php' );
}

public function load_plugin_textdomain() {
load_plugin_textdomain( 'wp-my_net', false, dirname(
plugin_basename( FILE ) ) . '/languages/' );
}

public function frontend_scripts() {
wp_register_script( 'wp-my_net-ajax-filters', MY_NET_PLUGIN_URL . '/assets/js/ajax-filters.min.js',
array( 'jquery' ), MY_NET_VERSION, true );
wp_register_script( 'wp-my_net-job-dashboard', MY_NET_PLUGIN_URL . '/assets/js/job-dashboard.min.js',
array( 'jquery' ), MY_NET_VERSION, true );

```

```

wp_register_script( 'wp-my_net-job-application', MY_NET_PLUGIN_URL . '/assets/js/job-
application.min.js', array( 'jquery' ), MY_NET_VERSION, true );
wp_register_script( 'wp-my_net-job-submission', MY_NET_PLUGIN_URL . '/assets/js/job-
submission.min.js', array( 'jquery' ), MY_NET_VERSION, true );

wp_localize_script( 'wp-my_net-ajax-filters',
'MY_NET_ajax_filters', array(
'ajax_url' => admin_url('admin-ajax.php')
) )

```

4.1.2. Admin

```

<?php //encoding: utf-8
require_once( MYNET_DIR . '/admin-settings.php' );
require_once( MYNET_DIR . '/admin-frontend-settings.php' );
require_once( MYNET_DIR . '/admin-jobs-settings.php' );
require_once( MYNET_DIR . '/admin-jobs.php' );
require_once( MYNET_DIR . '/admin-application-form.php' );
require_once( MYNET_DIR . '/admin-applications.php' );
require_once( MYNET_DIR . '/admin-emails.php' );
require_once( MYNET_DIR . '/admin-interviews.php' );
require_once( MYNET_DIR . '/admin-comments.php' );

function MYNET_admin_setup() {
$options = get_option( 'MYNET_options' );

$pages = array();

add_menu_page(
( 'My Net', 'MYNET' ), __( 'My Net', 'MYNET' ),
'publish_posts', 'MYNET-conf', 'MYNET_conf' );
$pages[] = add_submenu_page( 'MYNET-conf', ( 'My Net', 'MYNET' ), (
'Settings', 'MYNET' ), 'manage_options', 'MYNET-conf', 'MYNET_conf' );
$pages[] = add_submenu_page( 'MYNET-conf', ( 'My Net', 'MYNET' ), (
'Add Job', 'MYNET' ), 'publish_posts', 'MYNET-add-job', 'MYNET_add_job' );
$pages[] = add_submenu_page( 'MYNET-conf', ( 'My Net', 'MYNET' ), (
'Jobs', 'MYNET' ), 'publish_posts', 'MYNET-list-jobs', 'MYNET_list_jobs' );
$pages[] = add_submenu_page( 'MYNET-conf', ( 'My Net', 'MYNET' ), (
'Applications', 'MYNET' ), 'read_private_pages', 'MYNET-list-applications',
'MYNET_list_applications' );
$pages[] = add_submenu_page( 'MYNET-conf', ( 'My Net', 'MYNET' ), (
'Emails', 'MYNET' ), 'read_private_pages', 'MYNET-list-emails', 'MYNET_list_emails' );

```

```

if( $options['interviews'] )
$pages[] = add_submenu_page( 'MYNET-conf',      ( 'My Net', 'MYNET'
),      ( 'Interviews', 'MYNET' ), 'read_private_pages', 'MYNET-interviews',
'MYNET_interviews' );

foreach( $pages as $page ) {
add_action(      "admin_print_styles-$page",      'MYNET_admin_print_styles'      );      add_action(
"admin_print_scripts-$page",      'MYNET_admin_print_scripts'      );      add_action(      "admin_head-$page",
'MYNET_admin_header' );
}

function MYNET_admin_print_styles() {
global $wp_styles;

wp_enqueue_style( 'MYNET-admin', MYNET_URL . '/css/admin.css', false, MYNET_VERSION, 'all' );
wp_enqueue_style( 'MYNET-admin-ie7', MYNET_URL . '/css/admin-ie7.css', false, MYNET_VERSION, 'all' );
wp_enqueue_style( 'MYNET-admin-print', MYNET_URL . '/css/admin-print.css', false, MYNET_VERSION,
'print' );
wp_enqueue_style( 'MYNET-admin-print-ie7', MYNET_URL . '/css/admin-print- ie7.css', false,
MYNET_VERSION, 'print' );
wp_enqueue_style( 'dashboard' );

$wp_styles->add_data( 'MYNET-admin-print-ie7', 'conditional', 'lte IE 7' );
$wp_styles->add_data( 'MYNET-admin-ie7', 'conditional', 'lte IE 7' );
}

function MYNET_admin_print_scripts() {
wp_enqueue_script( 'MYNET-admin', MYNET_URL . '/js/admin.js', false, MYNET_VERSION );
wp_deregister_script( 'jquery-ui-datepicker' );
wp_enqueue_script( 'jquery-ui' );
wp_enqueue_script( 'jquery-ui-datepicker', MYNET_URL . '/js/jquery-ui- datepicker.js', false,
MYNET_VERSION );
wp_enqueue_script( 'dashboard' );
}

function MYNET_admin_header() {
global $wp_version;
$options = get_option( 'MYNET_options' );

$textareas = array();
foreach( $options['job_fields'] as $id => $field ) {
if( 'textarea' == $field['type'] ) {
$textareas[] = $id;
}
}
}

```

```

?>
<script type="text/javascript">
//
addLoadEvent(function() {
jQuery(".datepicker").datepicker({

MYNET_URL ?&gt;/images/calendar.gif',

dateFormat: 'yy-mm-dd', changeMonth: true, changeYear: true, gotoCurrent: true, showOn: 'button',
buttonImage: '&lt;?php echo
});

buttonImageOnly: true
jQuery(".column-cb &gt; *").click(function() { jQuery(".check-column &gt;
*").attr('checked', jQuery(this).is(':checked')) } );
jQuery("div.star-holder img").click(function() {
var cssclass = jQuery(this).parent().attr("class"); var count = cssclass.replace("star star", "");
jQuery(this).parent().parent().find('input[name="MYNET-
rating"]').attr("value", count);
jQuery(this).parent().parent().find("div.star-rating").css("width", (count *
19) + "px");

var data = jQuery(this).parent().parent().find('input[name="callbackid"]');
var func = jQuery(this).parent().parent().find('input[name="callbackfunction"]');
var callback;
if( data.length &gt; 0 ) {
callback = {
action: func[0].value, appid: data[0].value, rating: count
};
}
});

jQuery.post( ajaxurl, callback );

jQuery("div.star-holder img").mouseenter(function() {
var cssclass = jQuery(this).parent().attr("class"); var count = cssclass.replace("star star", "");
jQuery(this).parent().parent().find("div.star-rating").css("width", (count * 19) + "px");
});

jQuery("div.star-holder img").mouseleave(function() {
var count = jQuery(this).parent().parent().find('input[name="MYNET- rating"]').attr("value");
jQuery(this).parent().parent().find("div.star-rating").css("width", (count *
19) + "px");
});
}
</pre>
</div>
<div data-bbox="905 921 940 939" data-label="Page-Footer">48</div>
```



```

<?php
}
?>
});

function MYNET_reset_rating( id, func ) {

jQuery( "#MYNET-rating-" + id ).attr("value", 0);
jQuery( "#MYNET-star-rating-" + id ).css("width", "0px");

if( "filter" != id ) {
callback = {
};
action: func, appid: id, rating: 0
}
}
//]]>

jQuery.post( ajaxurl, callback );
</script>

```

```

<?php
}

function MYNET_print_settings_tabs() {
$tabs = array(
'admin' =>      ( 'Admin Settings', 'MYNET' ),
'display' =>   ( 'Display Settings', 'MYNET'),
'appform' =>   ( 'App. Form Settings', 'MYNET'),
'jobform' =>   ( 'Job Form Settings', 'MYNET'),
);
?>

```

```

<?php

if( ! array_key_exists( 'tab', $_REQUEST ) )
$_REQUEST['tab'] = 'admin';

<h2 class="MYNET-tabs">

foreach( $tabs as $key => $title ) {
$class = '';
if( $key != $_REQUEST['tab'] )
$class = ' menu-tab-inactive';
echo "<a href='" . admin_url( "admin.php?page=MYNET-

```

```

conf&tab=$key" ) . "' class='menu-tabs$class'>$title</a>";
}
}

```

4.1.3. Registration

```

define( 'RPR_VERSION', '3.9.9' );
define( 'RPR_ACTIVATION_REQUIRED', '3.9.6' );

if ( !class_exists( 'Register_Plus_Redux' ) ) {
class Register_Plus_Redux {
private /*.array[string]mixed*/ $options;

public /*.void*/ function    construct() {
register_activation_hook(    FILE    , array( $this, 'rpr_activation' )
);

register_deactivation_hook( __FILE
'Register_Plus_Redux', 'rpr_uninstall' ) );

, array('rpr_uninstall' ) );

register_uninstall_hook(    FILE    , array( 'Register_Plus_Redux',

add_action( 'init', array( $this, 'rpr_i18n_init' ), 10, 1 );

if ( !is_multisite() ) {
add_filter( 'pre_user_login', array( $this,
'rpr_filter_pre_user_login_swp' ), 10, 1 ); }
add_action( 'admin_enqueue_scripts', array( $this,
'rpr_admin_enqueue_scripts' ), 10, 1 );
add_action( 'show_user_profile', array( $this,
'rpr_show_custom_fields' ), 10, 1 );
add_action( 'edit_user_profile', array( $this,
'rpr_show_custom_fields' ), 10, 1 );
add_action( 'profile_update', array( $this, 'rpr_save_custom_fields'
), 10, 1 );

add_action( 'admin_footer-profile.php', array( $this,
'rpr_admin_footer' ), 10, 0 );
add_action( 'admin_footer-user-edit.php', array( $this,
'rpr_admin_footer' ), 10, 0 );
}

```

```

public /*.void.*/ function rpr_activation() {
global $wp_roles;
add_role( 'rpr_unverified', 'Unverified' );
update_option( 'register_plus_redux_last_activated', RPR_ACTIVATION_REQUIRED );
}

public static /*.void.*/ function rpr_uninstall() {
global $wp_roles;
remove_role( 'rpr_unverified' );
delete_option( 'register_plus_redux_last_activated' );
}

public static /*.mixed.*/ function default_options( $option = '' )
{

$blogname = stripslashes( wp_specialchars_decode( get_option(
'blogname' ), ENT_QUOTES ) );

$options = array(
'verify_user_email' => is_multisite() ? '1' : '0',

'message_verify_user_email' => is_multisite() ?
( "<h2>%user_login% - ваш новий логін</h2>\n<p>Але, перш ніж ви зможете використовувати ваш акаунт,
<strong>ви мусите активувати його</strong></p>\n<p>Перевірте вхідні повідомлення у пошті, вказаній
при реєстрації<strong>%user_email%</strong> та натисніть на лінк у листі для
підтвердження.</p>\n<p>Якщо ви не підтвердите ваш акаунт протягом двох днів, вам доведеться
реєструватись знову.</p>", 'register- plus-redux' ) :
( 'Будь ласка перевірте ваш акаунт шляхом натискання на лінк у листі.', 'register-plus-redux' ),
'verify_user_admin' => '0',
'message_verify_user_admin' => ( 'Ваш акаунт буде переглянуто адміністратором і, як тільки його
буде погоджено, ви отримаєте доступ до сайту.', 'register-plus- redux' ),
'delete_unverified_users_after' => is_multisite() ? 0 : 7,
'autologin_user' => '0',

Agreement',

'username_is_email' => '0',
'double_check_email' => '0',
'user_set_password' => '0',
'min_password_length' => 6,
'disable_password_confirmation' => '0',
'show_password_meter' => '0',
'message_empty_password' => 'Сила пароля',
'message_short_password' => 'Надто короткий пароль',

```

```

'message_bad_password' => 'Поганий пароль',
'message_good_password' => 'Хороший пароль',
'message_strong_password' => 'Сильний пароль',
'message_mismatch_password' => 'Паролі не співпадають',
'enable_invitation_code' => '0',
'require_invitation_code' => '0',
'invitation_code_case_sensitive' => '0',
'invitation_code_unique' => '0',
'enable_invitation_tracking_widget' => '0',
'show_disclaimer' => '0',
'message_disclaimer_title' => 'Увага',
'require_disclaimer_agree' => '1',
'message_disclaimer_agree' => 'Прийняти',
'show_license' => '0',
'message_license_title' => 'Ліцензійна угода',
'require_license_agree' => '1',
'message_license_agree' => 'Прийняти ліцензійну угоду',

background-color:#FFFFE0;',

'show_privacy_policy' => '0',
'message_privacy_policy_title' => 'Політика конфіденційності',
'require_privacy_policy_agree' => '1',
'message_privacy_policy_agree' => 'Прийняти політику конфіденційності',
'default_css' => '1',
'required_fields_style' => 'border:solid 1px #E6DB55;
'required_fields_asterisk' => '0',
'starting_tabindex' => 0,

/*
'datepicker_firstdayofweek' => 6,
'datepicker_dateformat' => 'дд/мм/рррр',
'datepicker_startdate' => '',
'datepicker_calyear' => '',
'datepicker_calmonth' => '',
*/

'disable_user_message_registered' => '0',
'disable_user_message_created' => '0',
'custom_user_message' => '0',
'user_message_from_email' => get_option( 'admin_email' ),
'user_message_from_name' => $blogname,
'user_message_subject' => '[' . $blogname . ']' . ('Ваші дані для входу', 'register-plus-redux'
),
'user_message_body' => "Ім'я користувача:

```

```

%user_login%\nPassword: %user_password%\n\n%site_url%\n",
'send_user_message_in_html' => '0',
'user_message_newLine_as_br' => '0',
'custom_verification_message' => '0',
'verification_message_from_email' => get_option(
'admin_email' ),

'verification_message_from_name' => $blogname,
'verification_message_subject' => '[' . $blogname . ']' . (
'Підтвердіть свій акаунт', 'register-plus-redux' ),
'verification_message_body' => "Verification URL:
%verification_url%\nБудь ласка скористайтесь посиланням вище, для того щоб підтвердити ваш акаунт\n",
'send_verification_message_in_html' => '0',
'verification_message_newLine_as_br' => '0',

'disable_admin_message_registered' => '0',
'disable_admin_message_created' => '0',
'admin_message_when_verified' => '0',
'custom_admin_message' => '0',

'admin_message_from_email' => get_option( 'admin_email' ),
'admin_message_from_name' => $blogname,
'admin_message_subject' => '[' . $blogname . ']' . ( 'Новий користувач зареєстрований', 'register-
plus-redux' ),
'admin_message_body' => "New user registered on your site %blogname%\n\nUsername: %user_login%\nE-
mail: %user_email%\n",
'send_admin_message_in_html' => '0',
'admin_message_newLine_as_br' => '0'
);

if ( !empty( $option ) ) {
if ( array_key_exists( $option, $options ) ) {
return $options[$option];
}
else {
}
}
return $options;
}

$options ) {

public /*.bool.* / function rpr_update_options( /*.array[string]mixed.* /

if ( empty( $options ) && empty( $this->options ) ) return FALSE;

```

```

if ( !empty( $options ) ) {
update_option( 'register_plus_redux_options', $options );
$this->options = $options;

>options );
}
else {
}

update_option( 'register_plus_redux_options', $this->return TRUE;
}

private /*.void.* / function rpr_load_options( $force_refresh = FALSE ) {
if ( empty( $this->options ) || $force_refresh === TRUE ) {
$this->options = get_option( 'register_plus_redux_options'
);
}
if ( empty( $this->options ) ) {
$this->rpr_update_options( Register_Plus_Redux::default_options() );
}
}

public /*.mixed.* / function rpr_get_option( /*.string.* / $option ) {
if ( empty( $option ) ) return NULL;
$this->rpr_load_options( FALSE );
if ( array_key_exists( $option, $this->options ) ) {
return $this->options[$option];
}
}

```

4.2. Скріншоти

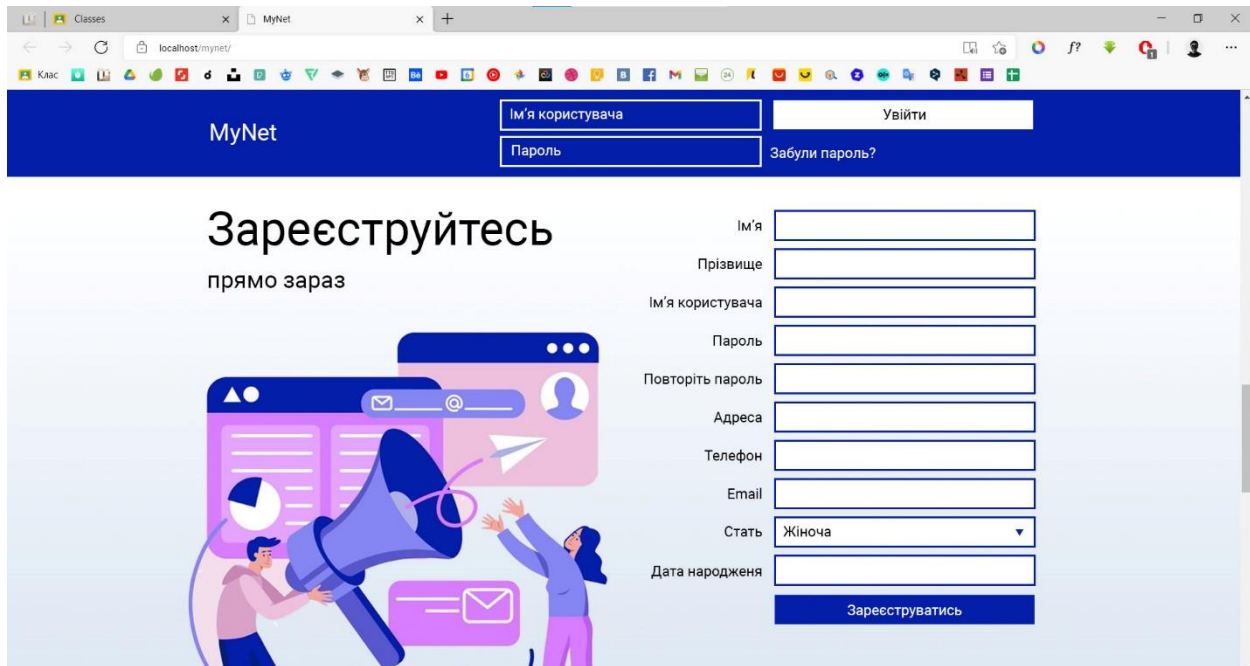


Рис. 4.1. Сторінка входу

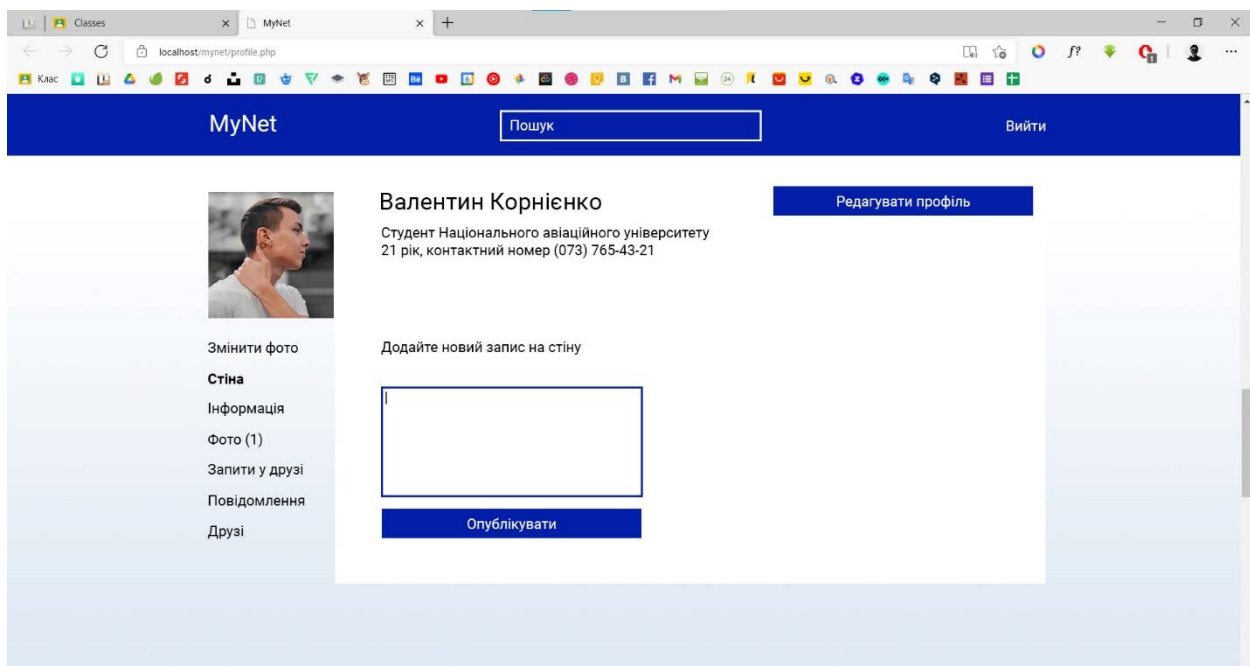


Рис. 4.2. Головна сторінка

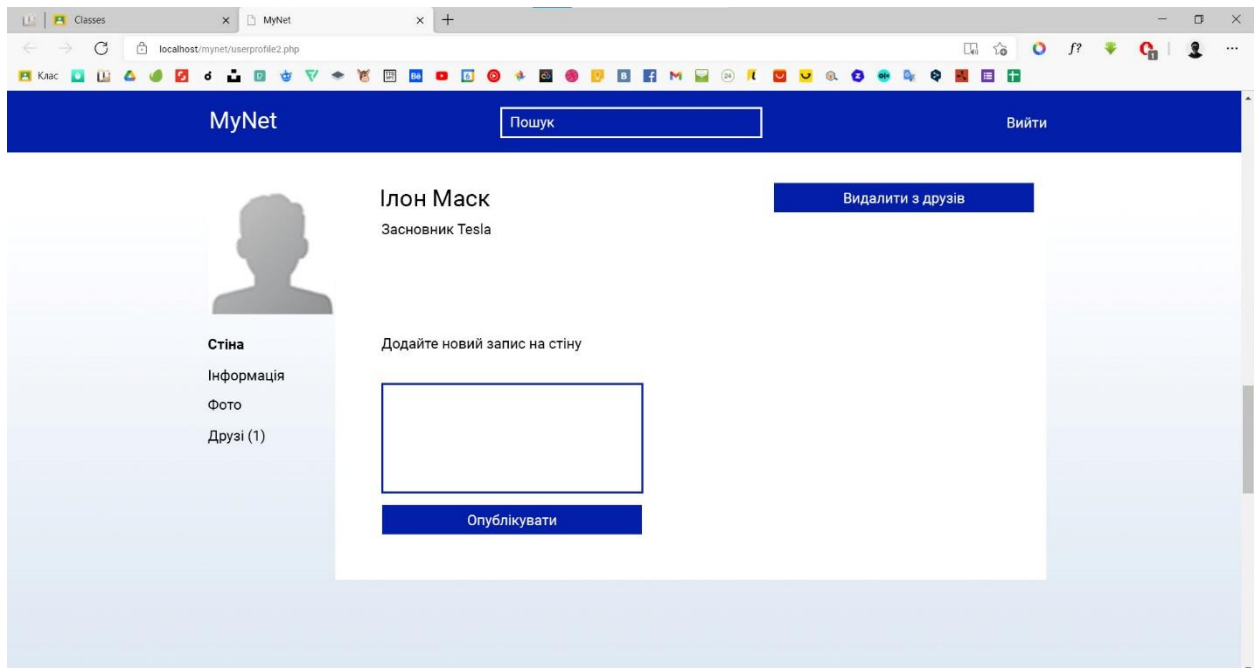


Рис. 4.3. Сторінка друга

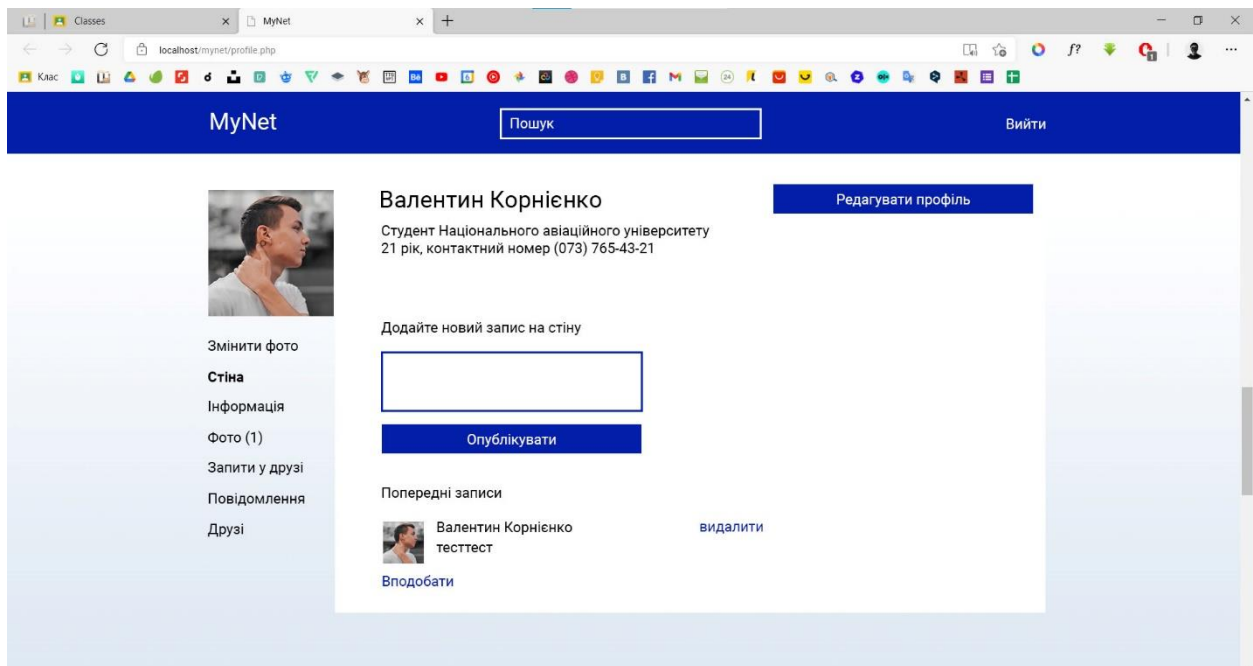


Рис. 4.4. Створення посту

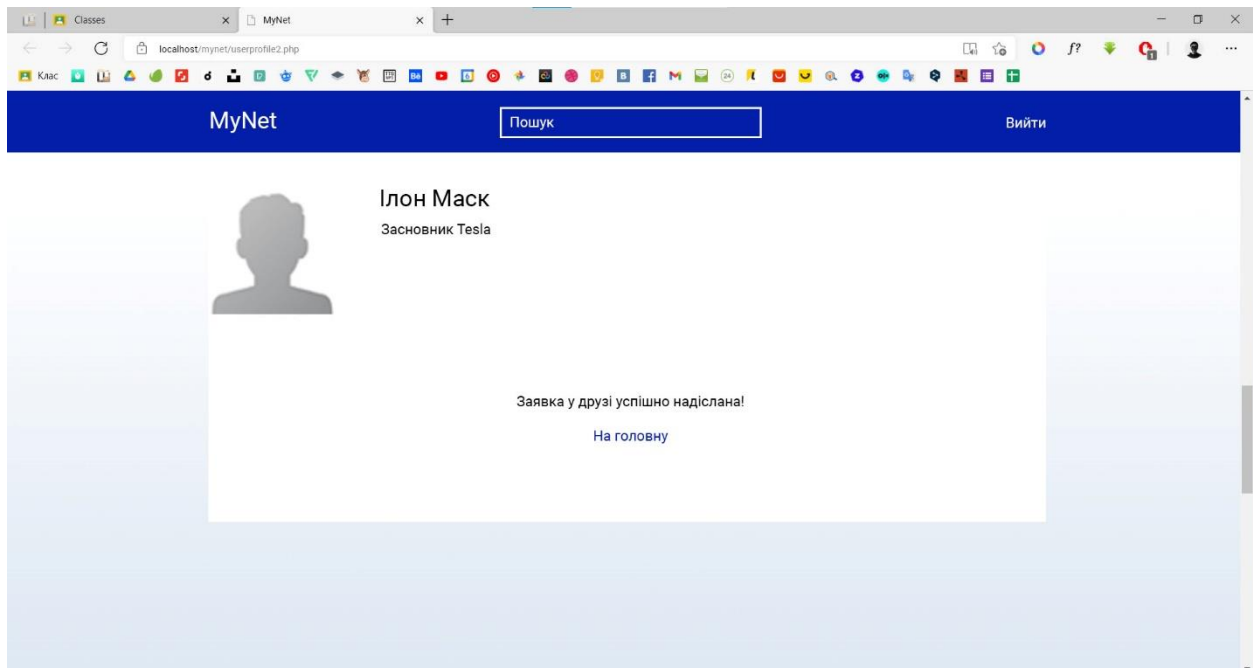


Рис. 4.5. Запит у друзі

ВИСНОВКИ

При розробці системи веб-порталу соціальної мережі були зроблені належні кроки для створення і розробки пакету програмного забезпечення, яке цілком вирішує поставлену задачу. При реалізації були використані сучасні та актуальні інструменти, технології та ресурси.

Особливу увагу було приділено зручності та зрозумілості веб-порталу для користувача. Оскільки лише в такому випадку можна сподіватися, що система буде належним чином задовольняти його потреби.

У першому розділі було сформульовано характеристику задачі, її мету вирішення, було поставлено цілі, які необхідно було досягнути та проведено аналіз існуючих тенденцій та продуктів-аналогів.

У другому розділі було розглянуто технічні особливості системи, проведено облік необхідних технологічних та нетехнологічних функцій, що мають бути присутні. Були проаналізовані методи, технології інструментарій для розробки.

У третьому розділі були наведені особливості проектування системи, діаграми варіантів використання, потоків даних, таблиці бази даних.

У четвертому розділі були наведені результати проведеної роботи, продемонстровані критичні елементи коду та візуальне представлення готового сайту.

СПИСОК БІБЛІОГРАФІЧНИХ ПОСИЛАНЬ ВИКОРИСТАНИХ ДЖЕРЕЛ

- 1 Що таке вебсайт? – [Електронний ресурс] Режим доступу: <http://www.freeservers.com/WebHosting101/WhatIsAWebsite.html> (дата звернення 20.05.2021) – Назва з екрана.
- 2 How does the web work? – [Електронний ресурс] Режим доступу: <http://www.bbc.co.uk/webwise/guides/how-the-web-works> (дата звернення 20.05.2021) – Назва з екрана.
- 3 Use of Social Networking Sites for Recruiting and Selecting in the Hiring Process? – [Електронний ресурс] Режим доступу: <http://www.bbc.co.uk/webwise/guides/how-the-web-works> (дата звернення 21.05.2021) – Назва з екрана.
- 4 Upender Singh Popularity of SNSs among Indian Non-IT Students http://ijeit.com/vol%202/Issue%207/IJEIT1412201301_32.pdf (дата звернення 21.05.2021) – Назва з екрана.
- 5 Найпопулярніші соціальні мережі в Україні та країнах світу у 2020 – [Електронний ресурс] Режим доступу: <https://uaspectr.com/2020/06/23/najpopulyarnishi-sotsialni-merezhi-v-ukrayini-ta-krayinah-svitu-2020/> (дата звернення 21.05.2021) – Назва з екрана.
- 6 Hypertext – [Електронний ресурс] Режим доступу: <https://www.britannica.com/technology/hypertext> (дата звернення 21.05.2021) – Назва з екрана.