



Приватний заклад вищої освіти
Одеський технологічний університет «ШАГ»
Кафедра інформаційних технологій та фундаментальної підготовки

випускна кваліфікаційна робота бакалавра

«ІНФОРМАЦІЙНА СИСТЕМА ДЛЯ ОБМІНУ МЕДІАКОНТЕНТОМ»

на здобуття бакалаврського рівня вищої освіти
зі спеціальності «122 Комп'ютерні науки»

Виконавці проекту

№	П.І.Б.	Група
1	Бондаренко М. О.	КН-П-192
2	Андрухів В. Р.	КН-А-191
3	Панкратов. М. А	КН-Д-192
4	Тарасюк.А.С	КН-П-191
5	Кохановский І. О.	КН-П-191
6	Сапригін А. Д.	КН-П-191

Автор звіту

_____ Тарасюк Анна Сергіївна

АНОТАЦІЯ

УДК 004.9

К-191

Тарасюк А.С. Соціальна мережа для обміну медиа-контентом: Автореферат випускної кваліфікаційної роботи на здобуття бакалаврського рівня вищої освіти зі спеціальності «122 Комп'ютерні науки». - Одеса: ОТУШ, 2023 - 13 с.

Робота присвячена дослідженням, розвиненню та впровадженню інформаційної системи, головним призначенням якої є взаємодія між користувачами. Додаткові функції системи включають реєстрацію, авторизацію та автентифікацію, можливість створити новий пароль.

Актуальність роботи визначається великою кількістю людей, які хочуть вести активний спосіб життя в інтернет-просторі без витрат на покупку преміум привілеїв. Програма націлена на людей, які хочуть: ділитися новими враженнями, фото, відео, стежити за активністю інших користувачів, коментувати чиюсь думку, знайомитися з новими людьми, брати участь у різних спільнотах. Також вона буде корисна тим хто вимушено переміщений і підтримує контакти переважно через такі системи. Особливу увагу привертає простота використання програми без зайвого. Все що потрібно користувачеві, це: аутентифікувати свою особистість, змінити свій пароль якщо забув, створити пост, видалити пост, завантажити на пристрій медіаконтент, знайти користувачів (групи), лайкати, зберігати, коментувати та пересилати пости.

Об'єктом дослідження у рамках даної роботи виступає засіб комунікації. У якості задач дослідження було встановлене наступне:

- провести огляд інформаційних систем, що враховують відомості про те, які з інструментів найчастіше використовують в соціальних мережах, встановити характерні риси та напрями удосконалення.
- визначити програмні засоби (інтерфейси API), які дозволяють досягти максимальної швидкості, комфорту та безпеки користувачу.
- скласти програмне забезпечення, реалізувавши у ньому результати, досягнуті у попередніх задачах, провести випробування за різних умов та з різних пристроїв.
- впровадити у систему програмні засоби захисту даних для покращення показників інформаційної безпеки створеної системи.

Методи дослідження, що використані для досягнення поставлених задач, утворені суперпозицією методів системного аналізу, моделювання та когнітивної методології.

ЗМІСТ

ВСТУП _____	3
Технічне завдання до проекту _____	4
РОЗДІЛ 1. Загальна характеристика системи _____	6
РОЗДІЛ 2. Реалізація клієнтської частини _____	7
РОЗДІЛ 3. Результати випробувань _____	10
ВИСНОВКИ _____	14
ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ _____	13

ВСТУП

Розвиток і поширення інформаційних технологій супроводжується перенесенням великої кількості можливостей у цифровий простір. Стратегія розвитку інформаційного суспільства в Україні визначає, що «комунікаційні технології є необхідним інструментом соціально-економічного прогресу, одним з основних чинників інноваційного розвитку економіки» .

На сьогоднішній день кожна друга людина підтримує контакти з іншими за допомогою мобільного зв'язку, але у ситуації де мобільний оператор, послугами якого користувалася людина, не може надавати свої послуги, треба мати альтернативні засоби підтримання зв'язку. Завдяки соціальним мережам людина може дізнатися багато про іншу людину навіть не телефонуючи їй або знаходячись у країні де потрібний мобільний оператор не працює. Також, соціальна мережа може бути в ролі рекламного майданчика, завдяки чому за неї не обов'язково потрібно платити. Завдяки рекламним можливостям людина може просунути своє ремесло, бізнес і так далі, про яке може дізнатися більша кількість людей.

Однак, якщо давати людям можливість ділитися своєю особистою інформацією це потребує покращення інформаційної безпеки захисту персональних даних користувача. Конфіденційність може зумовлюватися перевіркою особистості, а саме використанням можливості реєстрації, авторизації та аутентифікації як мінімум. Це актуалізує додаткові дослідження і розробки в зазначеній області.

Як предмет даної роботи була обрана досить поширена система обміну графічним контентом у функціональність якої входить розміщення постів різного роду, які приходять від одних користувачів та обробляються іншими, головною метою яких є донести до читачів ідею самого посту завдяки опису та графічному контенту.

Також функції переважної більшості подібних систем повинні передбачати можливість додавати історії - короткі відео формати, додавати місце створення публікації, обмеження, що накладаються на групу користувачів.

Детальне технічне завдання до проекту наводиться далі. У рамках даного звіту розглядається створення бази даних. Звіт складається з трьох розділів у яких описується загальна характеристика системи, деталі створення бази та результати її тестування. Висновки відображають результати, що досягнуті у рамках даної частини проекту, загальний висновок є сукупністю звітів усіх учасників проекту.

ТЕХНІЧНЕ ЗАВДАННЯ ДО ПРОЕКТУ

Загальний опис.

Програмне забезпечення (ПЗ) необхідно для надання можливості ділитися інформацією з іншими людьми та реагувати на контент, який був доданий.

Призначення:

Програмне забезпечення орієнтовно на людей, які ведуть активне соціальне життя в інтернет просторі.

Вимоги до функціоналу:

Реєстрація користувачів:

Користувач ПЗ має змогу зареєструватись як за допомогою авторизаційних даних: емейлу, логіну, паролю, дати народження.

При реєстрації обов'язково зазначається емейл адреса користувача. Далі вказується: логін (унікальний ідентифікатор користувача), пароль, день народження завдяки якому усі підписники будуть повідомлені. Останній етап реєстрації - створення аватарки. Користувач може пропустити цей крок. На сторінці реєстрації користувач може обрати: чи буде відображено його емейл, чи будуть приходити листи новин від додатку на пошту клієнта. Також клієнт матимете можливість на етапі створення емейл адреси вказати, чи буде запам'ятовувати його авторизаційні дані, додаток у майбутньому.

Основне робоче місце користувача ПЗ клієнтської частини повинне забезпечити усі необхідні засоби для:

- Авторизації, зокрема реєстрації та аутентифікації
- Зміни паролю у разі, якщо користувач забув
- Розміщення постів
- Перегляду посів
- Дати оцінку, додати в збереженні, відправити клієнту, залишити коментар
- Перегляду коментарів на пости
- Відповідей на коментарі інших користувачів
- Завантаження медіа-контент поста
- Копіювання посилання на пост
- Знаходження користувачів додатку
- Фільтрування постів
- Переходу на сторінку користувача
- Підписки, відписки від користувача
- Перегляду інформації користувача
- Зміни особистої інформації (нік, email, дата народження, тощо.)

Архітектура:

ПЗ має бути спроектоване за клієнт-серверною архітектурою з розподіленими вузлами: Backend - серверне програмне забезпечення, що забезпечує збереження та оброблення поточних даних, взаємодію з базою даних. Надає програмний інтерфейс взаємодії (API) з іншими вузлами.

Web App - додаток, доступний з веб-простору (сайт), що забезпечує візуальний інтерфейс користувачів усіх категорій.

Mobile App - додаток, що встановлюється на мобільних пристроях (смартфон, планшет).

Усі модулі ПЗ повинні мати засоби

- розгортання на новому пристрої (інсталяції)

Технології

Зберігання вихідних кодів ПЗ повинно здійснюватись за допомогою системи контролю версій (вибір конкретної системи узгоджується групою проекту).

Структура файлів та каталогів повинна відбивати архітектуру ПЗ, на зразок:

- main
 - Backend
 - Web
 - Mobile

Для кожного з модулів рекомендується (з можливістю часткової зміни) наступні технології створення:

Backend - Java, MySQL

Web - HTML, CSS, Javascript, Typescript, JS фреймворки, зокрема React JS, Next JS

Mobile - Java або Kotlin

Вихідний код повинен оформлятися у відповідності до загальноприйнятих стандартів, на зразок стандартів GNU <http://www.gnu.org/prep/standards/>, у тому числі з додержанням вимог до коментування та документування коду.

РОЗДІЛ 1

ЗАГАЛЬНА ХАРАКТЕРИСТИКА СИСТЕМИ

У відповідності до технічного завдання (ТЗ) система була спроектована за клієнт-серверною розподіленою архітектурою. У складі системи було умовно відокремлено наступні архітектурні частини:

- «Mockup»
- «Backend»
- «Frontend»

Частина «Mockup» являє собою набір моделей та описів поведінки користувача. Модель подає інтерфейси взаємодії користувачів різного типу з інформаційною системою, траєкторії рухів та переходів між різними інтерфейсами. При створенні інтерфейсів застосовано принципи розумної навігації, включно із підходами UI/UX, висновки колористики та ергономіки мобільних пристроїв. Частина розміщена за адресою <https://www.figma.com/file/iaMsWLpWsp6rpYfFM8WW4Z/ClickShot-graduate-work?type=design&node-id=0-1&mode=design>

Частина «Backend» реалізована як централізований вузол збереження та оброблення даних, що створюються та модифікуються протягом життєвого циклу інформаційної системи. Дана частина включає у себе базу даних проекту та програмний інтерфейс (API) доступу до основних її функцій, зокрема, керування контентом, авторизації, узгодження профілів користувачів різного типу та/або різних пристроїв.

Частина «Backend» виконана на базі програмного забезпечення Java, СУБД «MySQL».

Клієнтська частина системи «Front End», призначена для надання користувачеві доступу до функцій системи через засоби візуального інтерфейсу. Для обміну даними між «Backend», «Frontend», використовується бібліотека «Retrofit».

Вихідні коди та ресурси додатку розміщені на репозиторії <https://github.com/drDrake6/ClickshotBackend.git>. Подальший звіт присвячений детальному опису цієї частини.

З метою покращення безпеки, звертаючи увагу на той факт, що мобільні пристрої досить часто підключаються до незахищених мереж передавання даних, інформаційний обмін між частинами «Backend», «Frontend» захищений стандартом шифрування PBCS, SHA1. Це убезпечує роботу системи у відкритих мережах, а також спрощує задачі автентифікації та доступу до контенту з обмеженим доступом без наявності ключів розшифрування даних, навіть за умови дискредитації інших модулів системи.









РОЗДІЛ 2

РЕАЛІЗАЦІЯ БАЗИ ДАНИХ

Для нашого проекту ми обрали реляційну базу даних MySQL, розміщували базу даних на платформі PlanetScale.

Загалом ми маємо базу даних, в якій знаходяться 8 таблиць. А саме:

1. Comments - таблиця для коментарів
2. Followers - таблиця для підписників
3. LikedBy - таблиця для лайків
4. Posts - таблиця для публікацій
5. SavedBy - таблиця для збережених публікацій іншими користувачами
6. TaggedPeople - таблиця для відмічених на публікації користувачів
7. Users - таблиця, де зберігаються самі користувачі
8. Responses - таблиця з відповідями на коментарі

 Comments	128.0 KB	▼
 Followers	144.0 KB	▼
 LikedBy	144.0 KB	▼
 Posts	128.0 KB	▼
 Responses	144.0 KB	▼
 SavedBy	144.0 KB	▼
 TaggedPeople	144.0 KB	▼
 Users	160.0 KB	▼

Поперше ми зробили таблицю Users.


```
Users 160.0 KB ^
CREATE TABLE `Users` (
  `id` varchar(50) NOT NULL,
  `login` varchar(32) NOT NULL,
  `password` char(40) NOT NULL COMMENT 'SHA-160 hash',
  `name` tinytext NOT NULL,
  `surname` tinytext NOT NULL,
  `salt` char(40),
  `phone` varchar(40),
  `email` varchar(40),
  `email_code` tinytext,
  `email_attempt` int DEFAULT '0',
  `avatar` varchar(40),
  `birthday` date NOT NULL,
  `bio` varchar(200),
  `isPremium` tinyint(1) NOT NULL DEFAULT '0',
  `language` varchar(40) NOT NULL DEFAULT 'eng',
  `token` varchar(36),
  `role` char(1) DEFAULT 'u',
  `delete` datetime,
  PRIMARY KEY (`id`),
  UNIQUE KEY `Users_pk` (`login`),
  UNIQUE KEY `Users_pk2` (`email`),
  UNIQUE KEY `Users_pk3` (`token`)
) ENGINE InnoDB,
CHARSET utf8mb3;
```

Звернемо увагу на те, що ця таблиця має усі необхідні атрибути для налаштування профілю користувача, його реєстрації та подальшого існування. Також зауважу, що таблиця має атрибут “role” створений аби відокремити звичайного користувача від адміністратора.

Далі ми ознайомимося з таблицею Posts.

```
Posts 128.0 KB ^
CREATE TABLE `Posts` (
  `id` varchar(50) NOT NULL,
  `postponePublication` datetime,
  `author` varchar(50),
  `description` varchar(1000),
  `mediaUrl` varchar(50),
  `metadata` varchar(50),
  `addDate` datetime,
  `banned` timestamp NOT NULL DEFAULT current_timestamp(),
  `delete` datetime,
  PRIMARY KEY (`id`),
  KEY `user_idx` (`author`)
) ENGINE InnoDB,
CHARSET utf8mb4,
COLLATE utf8mb4_0900_ai_ci;
```

Ця таблиця має звичні для нас атрибути , варто звернути увагу, що дана таблиця використовує складовий первинний ключ - це ідентифікатор , що складається з двох або більше атрибутів.

Розглянемо таблиці LikedBy, TaggedPeople, SavedBy.

LikedBy

144.0 KB

```
CREATE TABLE `LikedBy` (  
  `postId` varchar(50) NOT NULL,  
  `userId` varchar(50) NOT NULL,  
  `date` timestamp NOT NULL DEFAULT current_timestamp(),  
  PRIMARY KEY (`postId`, `userId`),  
  KEY `post_idx` (`postId`),  
  KEY `user_idx` (`userId`)  
) ENGINE InnoDB,  
  CHARSET utf8mb4,  
  COLLATE utf8mb4_0900_ai_ci;
```

SavedBy

144.0 KB

```
CREATE TABLE `SavedBy` (  
  `postId` varchar(50) NOT NULL,  
  `userId` varchar(50) NOT NULL,  
  `date` timestamp NOT NULL DEFAULT current_timestamp(),  
  PRIMARY KEY (`postId`, `userId`),  
  KEY `post_idx` (`postId`),  
  KEY `user_idx` (`userId`)  
) ENGINE InnoDB,  
  CHARSET utf8mb4,  
  COLLATE utf8mb4_0900_ai_ci;
```

TaggedPeople

144.0 KB

```
CREATE TABLE `TaggedPeople` (  
  `postId` varchar(50) NOT NULL,  
  `userId` varchar(50) NOT NULL,  
  PRIMARY KEY (`postId`, `userId`),  
  KEY `post_idx` (`postId`),  
  KEY `user_idx` (`userId`)  
) ENGINE InnoDB,  
  CHARSET utf8mb4,  
  COLLATE utf8mb4_0900_ai_ci;
```

Ці таблиці схожі між собою, та так само як попередня використовують складовий первинний ключ користувача, що або вподобав публікацію, або відмітив іншого користувача, або взагалі зберіг публікацію.

Далі розглянемо таблицю Followers.

```
Followers 144.0 KB ^
CREATE TABLE `Followers` (
  `followerId` varchar(50) NOT NULL,
  `userId` varchar(50) NOT NULL,
  `date` timestamp NOT NULL DEFAULT current_timestamp(),
  PRIMARY KEY (`followerId`, `userId`),
  KEY `follower_idx` (`followerId`),
  KEY `user_idx` (`userId`)
) ENGINE InnoDB,
CHARSET utf8mb4,
COLLATE utf8mb4_0900_ai_ci;
```

Тут ми зберігаємо id користувача, та його підписника, та дату підписки.

Далі в нас таблиця Comments.

```
Comments 128.0 KB ^
CREATE TABLE `Comments` (
  `id` varchar(50) NOT NULL,
  `date` datetime NOT NULL,
  `author` varchar(32) NOT NULL,
  `content` varchar(120) NOT NULL,
  `postId` varchar(36) NOT NULL,
  `delete` datetime,
  PRIMARY KEY (`id`),
  KEY `user_idx` (`author`)
) ENGINE InnoDB,
CHARSET utf8mb4,
COLLATE utf8mb4_0900_ai_ci;
```

Серед атрибутів має id поста, зберігає автора коментаря, сам коментар, дату та інше.

Таблиця Responses.

```
Responses
CREATE TABLE `Responses` (
  `id_comment` varchar(50) NOT NULL,
  `id_response` varchar(50) NOT NULL,
  PRIMARY KEY (`id_response`, `id_comment`),
  KEY `comment_idx` (`id_comment`),
  KEY `response_idx` (`id_response`)
) ENGINE InnoDB,
CHARSET utf8mb4,
COLLATE utf8mb4_0900_ai_ci;
```

Зберігає id коментаря, та відповіді на нього.

З повним кодом проекту можна ознайомитись у репозиторії <https://github.com/drDrake6/ClickshotBackend>

РОЗДІЛ 3

РЕЗУЛЬТАТИ ВИПРОБУВАНЬ

Результатом випробування бази даних є прийом та обробка запитів.

```
> use instabox;
Completed in 2.41 ms

> show tables;

8 rows in (4.03 ms)
Tables_in_instabox

Comments
Followers
LikedBy
Posts
Responses
SavedBy
TaggedPeople
Users
```

```
> select * from Users;

3 rows in (5.12 ms)
Copy query Collapse rows
```

id	login	password	name	surname	salt	phone	email	email_code	email_attempt	avatar
1cc24bbe-b51f-430f-a9e9-9b912e92568e	Suslik	815a596a93e9ad5cc6118b4e64fbc7123471dcd	Suslik		af2d466893a759c807573104484c53fbbc442d2	NO_PHONE	slauik@gmail.com	0		538bcb2d4093-ad658f1f8
6d110bed-2250-4b78-b9bd-bca3aeb25f38	Andry	0a23360ae07953f02df669f3cab4ac7d71d2e34	Mafiyoznic	Saprygin	ce44c90abb0556612aae969849e133af2335dd9e	NO_PHONE	coffee1.2002@gmail.com	0		e6b46ca4af5-90b2686c85f
df430020-124e-4fc8-93dc-dea01e9a9aac	Ivan	37736449a8308083e4271c064db2bc3879a575e2	Ivan		f0d80b94365c2f8b2445b3d5cdd041fbec85fbf	NO_PHONE	deriglazov189@gmail.com	0		57cc429a11ee-b856242ac12

Тобто за гарним результатом варто вважати коректну роботу бази даних на додавання, редагування, видалення та пошук записів у таблицях.

```
> select * from Users where name = 'Suslik';

1 row in (4.77 ms)
```

id	login	password	name	surname	salt	phone	email
1cc24bbe-b51f-430f-a9e9-9b912e92568e	Suslik	815a596a93e9ad5cc6118b4e64fbc7123471dcd	Suslik		af2d466893a759c807573104484c53fbbc442d2	NO_PHONE	slauik@gmail.com

ВИСНОВКИ

За допомогою аналізу інших систем ми розробили додаток за допомогою якого користувачі зможуть ділитися різноманітними медіафайлами. Для того щоб проект зберігав швидкість роботи було обрано реляційну базу даних.

У результаті отримуємо базу даних, що складається з восьми таблиць, що забезпечують впевнену роботу нашого застосунку. Маємо змогу додавати, редагувати та видаляти контент. Всі вимоги, щодо функціоналу бази були виконані.

У майбутньому планується додати певний функціонал, котрий має контролювати спроби входу до додатку, та інший функціонал аби користувач мав більше можливостей у налаштуванні свого простору.

У підсумку, нашою командою був створений застосунок, що надає змогу поширювати не лише фото та відео контент, а й інше. Ми надаємо можливість користувачу максимальну змогу для творчості, розвитку, роботи.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Сільвія Ботрос, Джеремі Тінлі. MySQL по максимуму 4-те видання.2022 рік.
2. Люк Веллінг, Лора Томсон. MySQL. Учебное пособие.
3. Віктор Гольцман. MySQL 5.0. Бібліотека програміста
4. Sufyan bin Uzayr. Mastering MySQL for Web. A beginner's guide.