ХАРКІВСЬКИЙ ПАТЕНТНО-КОМП’ЮТЕРНИЙ КОЛЕДЖ

Відділення «Інженерія програмного забезпечення»

Предметно-циклова комісія «Інженерія програмного забезпечення»

«Сервіс сумісного перегляду відео з YouTube»

Технічне завдання

КП 121

|  |  |
| --- | --- |
| Виконали: | Водянов А.А |
|  | Якименко Р.А. |
| Керівник: |  |

Харків – 2020 року

Зміст

[Вступ 3](#_Toc59451134)

[1 Підстава для розробки 4](#_Toc59451135)

[2. Призначення та розробки 4](#_Toc59451136)

[3. Вимоги до програмного виробу 5](#_Toc59451137)

[3.1 Інтерфейс користувача 5](#_Toc59451138)

[3.2 Програмний інтерфейс 7](#_Toc59451139)

[3.3 Комунікаційний протокол 7](#_Toc59451140)

[3.4 Обмеження пам`яті 7](#_Toc59451141)

[3.5 Операції 8](#_Toc59451142)

[3.6 Функції продукту 8](#_Toc59451143)

[3.7 Атрибути програмного продукту 9](#_Toc59451144)

[3.7.1 Надійність 9](#_Toc59451145)

[3.7.2 Доступність 9](#_Toc59451146)

[3.7.3 Безпека 10](#_Toc59451147)

[3.7.4 Супроводжуваність 10](#_Toc59451148)

[3.8 Вимоги бази даних 10](#_Toc59451149)

[4 Техніко-економічні показники 11](#_Toc59451150)

[5. Стадії та етапи розробки 11](#_Toc59451151)

[6. Порядок контролю і приймання системи 12](#_Toc59451152)

# Вступ

Веб-сервіс перегляду сумісного відео SweetTube – веб-сервіс, який надає можливість дивитися відео з YouTube разом з іншими людьми одночасно з різних куточках планети через інтернет.

Деякі означення:

* Блог - вебсайт, головний зміст якого є регулярно додавані записи, зображення чи мультимедіа. Для блогів характерні короткі записи тимчасової значущості.
* Блогер – людина, яка веде блог.
* Ютубер – людина, яка веде свій канал на YouTube. Зазвичай Блогер і Ютубер – синоніми.
* Плейліст – черга з якого-небуть медіа-контенту. Після закінчення одного медіа-контенту з черги запускається наступний у черзі. Наприклад: відео, музика і тд.

# 1 Підстава для розробки

Підставою для розробки є завдання на курсове проектування видане цикловою комісією «Інженерія програмного забезпечення».

Мета - зробити зручний сервіс, який забезпечить гарні умови, буде зручним для перегляду сумісного відео з іншими людьми.

# 2. Призначення розробки

Сервіс сумісного перегляду відео з YouTube дозволить переглядати та обговорювати відео з YouTube разом з іншими людьми одночасно в онлайн-кімнаті. Система буде мати можливість створити плейліст з бажаними відео для перегляду, чат для обговорення чи спілкування між учасниками онлайн-кімнати.

# 3. Вимоги до програмного виробу

### 3.1 Інтерфейс користувача

Інтерфейс повинен дотримуватися сучасних тенденцій, а саме мати темну палітру кольорів, мати якомога менше елементів на сторінках.

На рисунку 3.1 можна побачити головні елементи сторінки перегляду відео.



Рисунок 3.1 макет сторінки перегляду відео

На рисунку 3.2 приведений макет сторінки пошуку існуючих онлайн-кімнат.



Рисунок 3.4 макет пошуку кімнати на SweetTube

### 3.2 Програмний інтерфейс

Потрібно розробити алгоритм створення нових кімнат та підтримку вже існуючих.

У віртуальній кімнаті повинні бути:

* чат;
* плейліст;
* список користувачів;
* додавання відео до плейлісту;
* панель модерації для володаря кімнати;
* плеєр, який транслює відео.

Для реалізації плеєру буде використаний API YouTube для отримання відео-потоку з сервісу YouTube. З нашої сторони потрібно розробити структуру отримання даних про відео (сам відео-потік, назва відео). З плейлісту буде зчитуватися посилання на відео, після буде відбуватися запит на отримання цього відео з YouTube та відправка його всім учасникам кімнати.

### 3.3 Комунікаційний протокол

Для реалізації зв’язку між сервером та клієнтом цього веб-сервісу буде використаний протокол http, та методи запитів: post, get, put, delete.

### 3.4 Обмеження пам`яті

Веб-сервіс перегляду сумісного відео з YouTube повинен працювати на приладах від 2 ГБ ОЗУ та процесором від 2ГЦ, 2 ядра, 2 потоки.

### 3.5 Операції

Перелік операцій, які будуть доступні:

* Створити віртуальну кімнату;
* Знайти віртуальну кімнату;
* Приєднатися до кімнати;
* Додати у плейліст відео з YouTube за посиланням;
* Змінити порядок відео в черзі;
* Використовувати чат для обговорення;
* Головний в кімнаті може:
* Виганяти з кімнати інших користувачів
* Давати або забирати дозволи використовувати фунції іншим користувача: виганяти та всі інші функції окрім створення віртуальної кімнати та пошук віртуальній кімнати.

### 3.6 Функції продукту

Перелік функцій:

* створення віртуальної кімнати;
* пошук віртуальної кімнати;
* приєднання до віртуальної кімнати;
* додавання відео в плейліст;
* зміна порядку черги відео у кімнаті;
* надіслання повідомлення у чат кімнати;
* надання прав користувачам володарем кімнати.

### 3.7 Атрибути програмного продукту

### 3.7.1 Надійність

Вся інформація про кімнати, учасників кімнати, чати буде зберігатися на серверній частині у базі даних. Усі дані будуть синхронізовані з користувачами при вході у віртуальну кімнату чи при пошуку віртуальних кімнат.

Веб-застосунок повинен працювати при стабільному інтернеті та нормальній роботі приладу користувача, окрім:

* якщо сервіс YouTube не працює;
* якщо у користувача зникла мережа.

### 3.7.2 Доступність

В кімнаті є головний, який може забрати доступ до дій в цій кімнаті, а також зробити приватну кімнату(пункт 3.1.5 операції) (кімната, яка не відображається у пошуку кімнати). Запит на створення нової кімнати може оброблятися максимум 1 секунду.

### 3.7.3 Безпека

Веб-сервіс повинен забезпечувати безпеку даних, які знаходяться на сервері:

Користувачі у кімнаті будуть поділені на декілька ролей: головний та гість.

Головний має такий функціонал (пункт 3.1.5)

Гість має функціонал, який надав йому головний у кімнаті.

### 3.7.4 Супроводжуваність

Після впровадження у роботу і захист курсової роботи, будуть виправлятися баги, якщо вони будуть. Додаватися новий функціонал: додавання інших сервісів для сумісного перегляду, прослуховування і так далі.

## 3.8 Вимоги бази даних

База даних буде працювати поруч з головним сервером. Вона повинна бути завжди доступна та узгоджена (CAP-теорема CA). Зберігати в собі список кімнат, чати кімнати.

# 4. Стадії та етапи розробки

|  |  |  |
| --- | --- | --- |
| Назва стадії/етапу | Початок | Кінець |
| - Формування вимог | 03.11.2020 | 06.11.2020 |
| - Розробка концепції | 09.11.2020 | 17.11.2020 |
| * опис концепції | 09.11.2020 | 11.11.2020 |
| * фіксація очікувань | 11.11.2020 | 12.11.2020 |
| * створення WBS | 11.11.2020 | 12.11.2020 |
| * визначення ролей | 13.11.2020 | 14.11.2020 |
| * визначення ресурсів | 13.11.2020 | 14.11.2020 |
| * аналіз ризиків | 16.11.2020 | 17.11.2020 |
| - Технічне завдання | 18.11.2020 | 20.11.2020 |
| - Ескізний проект | 23.11.2020 | 27.11.2020 |
| * модель взаємодії | 23.11.2020 | 24.11.2020 |
| * прототип інтерфейсу | 26.11.2020 | 27.11.2020 |
| - Технічний проект | 30.11.2020 | 29.12.2020 |
| * розробка домашньої сторінки | 30.11.2020 | 03.12.2020 |
| * розробка сторінки пошуку кімнати | 04.12.2020 | 08.12.2020 |
| * розробка сторінки кімнати | 09.12.2020 | 15.12.2020 |
| * розробка інструментів адміністрування | 30.11.2020 | 03.12.2020 |
| * розробка бази даних | 04.12.2020 | 08.12.2020 |
| * розробка інструментів створення кімнати | 09.12.2020 | 11.12.2020 |
| * розробка інструментів синхронізації відео-потоку | 14.12.2020 | 16.12.2020 |
| * розробка чату | 23.12.2020 | 25.12.2020 |
| - Тестування | 02.02.2021 | 12.02.2021 |
| * тестування верстки | 02.02.2021 | 04.02.2021 |
| * тестування функціоналу сайту | 02.02.2021 | 04.02.2021 |
| * тестування синхронізації відео-потоку | 08.02.2021 | 10.02.2021 |
| * стрес-тестування | 10.02.2021 | 12.02.2021 |
| * глобальне тестування | 10.02.2021 | 12.02.2021 |
| - Виправлення багів | 17.02.2021 | 26.02.2021 |
| * виправлення багів front-end частини | 17.02.2021 | 13.02.2021 |
| * виправлення багів back-end частини | 17.02.2021 | 23.02.2021 |
| * тестування виправлених частин | 24.02.2021 | 26.02.2021 |
| - Робоча документація | 01.03.2021 | 31.03.2021 |
| * створення Terms of Service | 01.03.2021 | 03.03.2021 |
| * створення Private policy | 01.03.2021 | 03.03.2021 |
| * написання звітної документації | 08.03.2021 | 10.03.2021 |
| * впровадження | 03.03.2021 | 10.03.2021 |
| * захист курсового проекту | 25.03.2021 | 31.03.2021 |

# 5. Порядок контролю і приймання системи

Контроль та приймання розробки відбувається на основі перевірки контрольно – відпрацьованих прикладів. При цьому відбувається перевірка всіх функцій програми.

Термін прийому проекту: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_.