

# КРОСПЛАТФОРМЕННИЙ ВЕБ-ЗАСТОСУНОК ДЛЯ РЕТРАНСЛЯЦІЇ НАВЧАЛЬНИХ МАТЕРІАЛІВ В УМОВАХ КАРАНТИННИХ ОБМЕЖЕНЬ

Вінницький національний технічний університет

## Анотація

*Робота присвячена розробці веб-застосунку для трансляції навчальних матеріалів, який зможе взаємодіяти з програмним забезпеченням напряму, без використання сторонніх сервісів. В даній статті описано структуру та призначення основних частин додатку, реалізованого мовою JavaScript з використанням бібліотеки React для використання реактивності у розробці.*

**Ключові слова:** веб-застосунок, стрімінг, ретрансляція, архітектура, тестування, програмне забезпечення, React.

## Abstract

*A web application for broadcasting educational materials has been proposed, which will be able to interact with the software directly, without the use of third-party services. The structure of the application was built, and a web application in JavaScript was implemented using the React library to use reactivity in development.*

**Keywords:** web-application, streaming, structure, testing, software, React.

## Вступ

В умовах пандемії COVID-19 все більше уваги приділяється засобам навчання школярів та студентів шляхом використання мережі Інтернет. Досить популярними стають різні системи для дистанційного навчання, освітні платформи для проходження он-лайн-курсів, системи Е-навчання, курси для підвищення кваліфікації, тощо, але такі системи більше орієнтовані на дорослих, ніж на школярів та студентів. Також особливий інтерес викликають сервіси з можливістю трансляції (або стрімінгу) відеоматеріалів, наприклад, такі популярні сервіси як YouTube, Twitch та інші. Але такі сервіси більше орієнтовані на розважальний, ніж на навчальний контент. При цьому вони мають обмежений функціонал або потребують використання сторонніх проміжних сервісів для ретрансляції. Як видно, сервісів, мета яких полягає у наданні можливості переглянути виключно навчальні матеріали для студентів та школярів не так багато, тому в сучасних умовах карантинних обмежень актуальною є розробка сервісу для ретрансляції відеоматеріалів саме з навчальним контентом, що дасть можливість педагогічним працівникам записувати матеріал та використовувати його для показу студентам у прямому ефірі.

## Результати дослідження

Пропонуємий веб-застосунок для ретрансляції навчальних матеріалів, складається з серверної та клієнтської частин. Серверна частина додатку відповідає за налаштування відеоплеєру та взаємодією з програмним забезпеченням для трансляції [1]. Клієнтська частина – частина видима користувачеві і являє собою інтерфейс для ретрансляції, реалізований комбінуванням декількох модулів, таких як генератор ключів захисту, інформаційна панель та кнопка для початку роботи веб-застосунку.

Для роботи розроблюваного додатку необхідні лише комп'ютер та вихід в мережу Інтернет. Веб-застосунок не потребує потужної системи і його навантаження на існуючу систему мінімальне.

Детальну структуру веб-застосунку наведено в таблиці 1.

У теці /client представлена візуальна частина веб-застосунку, реалізована за допомогою бібліотеки React. Користувач напряму взаємодіє з інтерфейсом, що реалізовується у функції.

`export default class Navbar extends React.Component {}` – дана функція є основною у клієнтській частині веб-застосунку і слугує програмною частиною інтерфейсу. Користувач, при взаємодії з застосунком, буде користуватись інтерфейсом або панеллю навігації. За навігацією користувач напряму підключається до серверу застосунку, щоб в подальшому взаємодіяти з стороннім програмним забезпеченням.

Таблиця 1 – Структура веб-застосунку для ретрансляції навчальних матеріалів

Файл/директорія	Призначення
/client	Клієнтська частина веб-застосунку на React
/server	Серверна частина веб-застосунку з використанням MongoDB
/public	Тека зі збереженими форматами стилів, шрифтів
package.json	Файл зі скриптами виконання файлів та підвантаженими залежностями
Webpack.config.js	Файл для збирання веб-застосунку

Тека /server слугує для зберігання основної частини веб-застосунку, а саме взаємодії з сервером. Відео, що в подальшому буде транслюватись, користується сервером та відправляє останньому запит, щоб отримати у відповідь певну адресу, яку додаток використовуватиме для підключення до програмного забезпечення.

Основним файлом у даній теці є app.js, в якому підсумована вся функціональність серверної частини і саме тут формується запит до бази даних.

Наведений нижче метод app.use відповідає за динамічне створення місця в базі даних, шляхом виконання методу MongoStore.create. Далі формується спеціальна адреса, яку отримає клієнтська частина. Ця адреса є захищеною та має термін життя 14 днів. Це термін дії ключа захисту, що формується в процесі налаштування взаємозв'язку веб-застосунку та програмного забезпечення для створення трансляцій.

```
app.use(session({
  store: MongoStore.create({
    mongoUrl: 'mongodb://127.0.0.1/nodeStream',
    ttl: 14 * 24 * 60 * 60 // = 14 days. Default
  }),
  secret: config.server.secret,
  maxAge : Date().now + (60 * 1000 * 30),
  resave : true,
  saveUninitialized : false,
}));
```

У теці /public збереженні шаблони стилів та шрифтів, що застосовуються по всьому додатку. Виділення таким чином стилів та шрифтів є дуже корисним під час розробки, код стає чистіший та легше редагується.

У файлі package.json зберігаються основні скрипти виконання файлів та залежності які використовуються під час розробки.

Система для ретрансляції являє собою веб-застосунок який напряму взаємодіє з ретрансляційним програмним забезпеченням OBS Studio або Xplit. При відповідному налаштуванні є можливість під'єднатися до даного програмного забезпечення. Після налаштування користувач може запустити трансляцію та отримати візуальний результат на основній сторінці веб-застосунку.

Для створення скриптів виконання файлів використовується код даного файлу.

```
"scripts": {
  "test": "echo \"Error: no test specified\" && exit 1",
  "watch": "./node_modules/.bin/webpack",
  "start": "./node_modules/.bin/supervisor -w server ./server/app.js"
}
```

Наприклад стрічки “test” відповідає певний скрипт, який тестує додаток, та описує помилку, якщо така виникає.

У файлі Webpack.config.js зберігаються загальні налаштування для збирання самого додатку. Зазвичай React сам модифікує даний файл та задає базові налаштування для збирання. Проте, розробник може скористатись даним файлом, щоб змінити деякі параметри і створити унікальний файл для збирання.

Для програмної реалізації інформаційної технології було обрано мову програмування JavaScript та середовище VScode. Також було використано бібліотеку React та засіб Gulp для збирання додатку. Було здійснено програмну реалізацію веб-застосунку для ретрансляції навчальних матеріалів, шляхом створення методу взаємодії веб-застосунку з програмним забезпеченням для створення онлайн-трансляцій напряду.

### **Висновки**

Отже, запропоновано веб-застосунок для трансляції навчальних матеріалів, який зможе взаємодіяти з програмним забезпеченням напряду, без використання сторонніх сервісів. Побудовано структуру додатку, та реалізовано веб-застосунок мовою JavaScript з використанням бібліотеки React для використання реактивності у розробці.

Даний веб-застосунок є перспективним рішенням, завдяки інноваційному методу ретрансляції навчальних матеріалів шляхом взаємодії з ретрансляційним інструментом OBS. В подальшому даний веб-застосунок може бути покращений для збереження трансляцій навчальних матеріалів.

### **СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ**

1. Dean Alan Hume. Progressive Web Apps. Manning, Manning Publications, 2017. — 200 p.
2. Gulp або Grunt. [Електронний ресурс] – Режим доступу до ресурсу: <https://stfalcon.com/ru/blog/post/grunt-vs-gulp>
3. Gerald D. Everett, Raymond McLeod Jr. Software Testing: Testing Across the Entire Software Development Life Cycle. – NJ.: Wiley-IEEE Computer Society Pr. – 2007. – 280 p. ISBN: 978-0471793717.

**Коваленко Максим Максимович** — студент групи ІКІ-20м, факультет інформаційних технологій та комп'ютерної інженерії, Вінницький національний технічний університет, Вінниця, email:

**Войцеховська Олена Валеріївна** — кандидат технічних наук, доцент кафедри обчислювальної техніки, Вінницький національний технічний університет, Вінниця

**Kovalenko Maxim M.**, — student of group ІКІ-20m, Faculty of Information Technologies and Computer Engineering, Vinnytsia National Technical University, Vinnytsia, e-mail: justspook42@gmail.com

**Voytsekhovska Olena V.** — PhD, Assistant Professor of the Computer Techniques Department, Vinnytsia National Technical University, Vinnytsia.