

Автор:  
студентка 41ППЗ групи  
**Назаренко Євеліна Юріївна**  
Науковий керівник:  
кандидат педагогічних наук,  
доцент, доцент кафедри комп'ютерної  
та програмної інженерії  
Стецик Сергій Павлович

## РОЗРОБЛЕННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ ДЛЯ РЕАЛІЗАЦІЇ ПРОЕКТІВ

**Анотація.** Метою дослідження є розроблення та створення вебзастосунку для автоматизації управління проектами, який забезпечує ефективне планування, розподіл завдань та моніторинг їх виконання. Завданнями дослідження є аналіз наявних методологій управління (Kanban), проектування архітектури системи на основі сучасного технологічного стеку та реалізація механізмів безпечної автентифікації. Об'єктом дослідження є процес управління проектами в організаціях, а предметом дослідження є методи та засоби розроблення вебзастосунків для автоматизації цих процесів. У дослідженні використано системний підхід та методи об'єктно-орієнтованого проектування. Результатом дослідження є повнофункціональна інформаційна система, що поєднує зручність інтерфейсу з високою продуктивністю серверної частини [1].

**Ключові слова:** інформаційна система, управління проектами, REST API, Node.js, React, JWT-автентифікація, Kanban, Sequelize, SQLite, Docker.

**Вступ.** Актуальність нашого дослідження визначена потребою у створенні власної інформаційної системи, яка поєднує гнучкість налаштування та сучасний технологічний стек. Більшість наявних рішень мають або надмірну складність, або високу вартість. Об'єктом дослідження є процес управління проектами в організаціях, а предметом - методи розроблення вебзастосунків для їх автоматизації. Мета роботи. Проектування та реалізація системи на основі Node.js, Express.js та React, яка забезпечує повний цикл роботи з проектами.

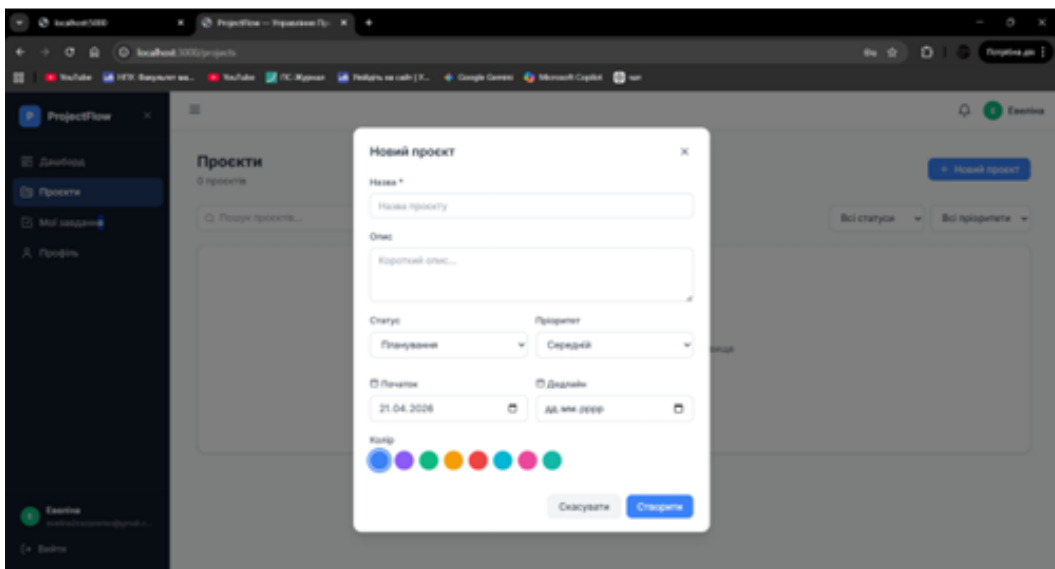
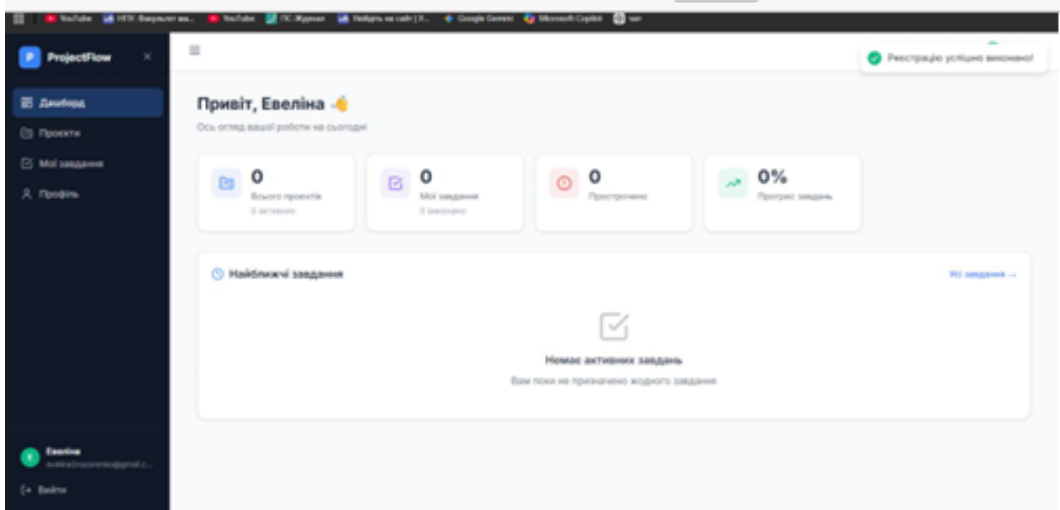
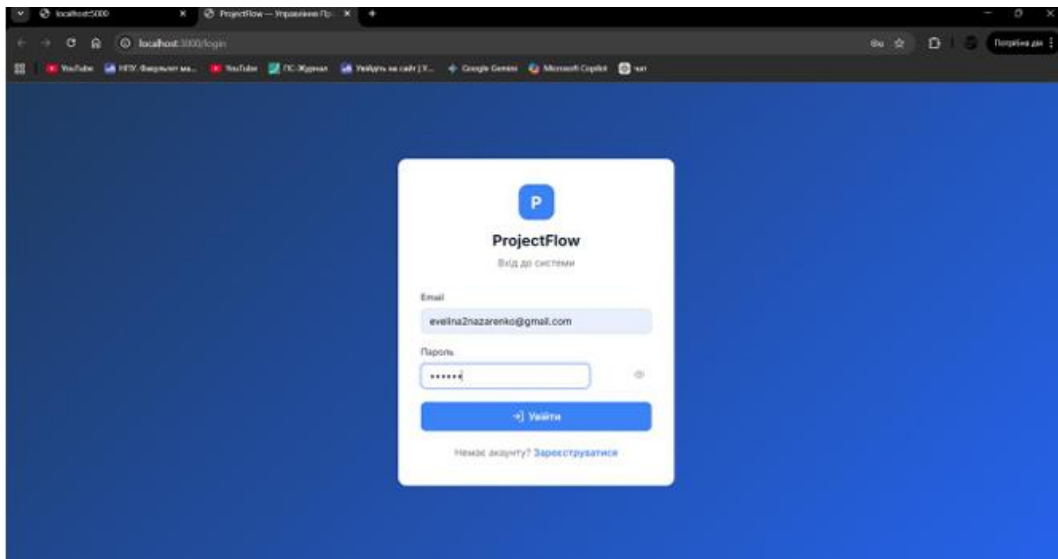
### **Основна частина.**

1. Аналіз та обґрунтування технологій. У процесі дослідження було проаналізовано методології Agile та Kanban. Обґрунтовано використання Node.js для серверної частини та React для клієнтської, що забезпечує високу продуктивність та масштабованість застосунку [2-4].

2. Проектування та структура системи. Архітектура системи побудована на принципах REST API, що дозволяє відокремити бізнес-логіку від інтерфейсу.

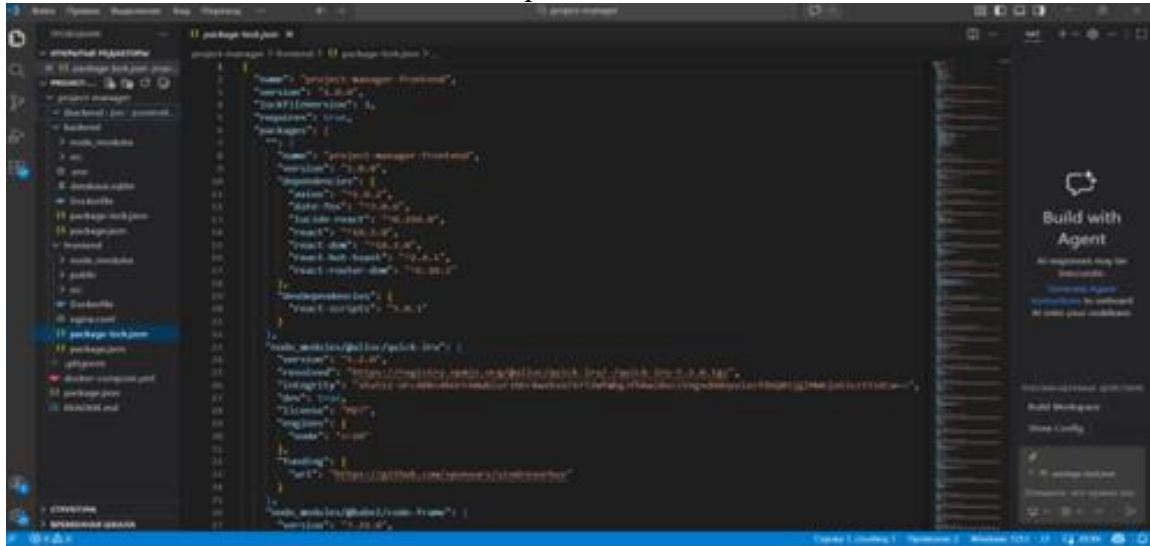
- Структура бази даних: За допомогою ORM Sequelize реалізовано реляційну схему в СУБД SQLite. Схема включає таблиці Users (збереження хешованих паролів та ролей), Projects (метадані проектів) та Tasks (завдання з прив'язкою до статусів та виконавців).

- Інтерфейс: Створено прототипи на основі Kanban-дошки, де завдання розподілені за колонками: «Дашборд», «Проекти», «Мої завдання» та «Профіль».



3. Реалізація та тестування. Серверна частина реалізована на Node.js, клієнтська – на React. Безпеку забезпечено через JWT-автентифікацію.

- Файлова структура: Організація проєкту передбачає чіткий поділ на backend та frontend модулі. Використання Dockerfile та docker-compose.yml забезпечує швидке розгортання системи в ізольованих контейнерах.



- Результати тестування: Проведено функціональне тестування модулів та аналіз продуктивності. Тести показали, що час обробки запиту на отримання списку завдань не перевищує 150 мс. Система сповіщень коректно ідентифікує завдання з наближенням дедлайну [1].

**Висновки.** Розроблена система відповідає сучасним вимогам до інструментів управління проєктами. Практична цінність полягає у створенні відкритого та продуктивного рішення, яке може бути використане командами для оптимізації робочих процесів.

#### Список використаних джерел

1. Канбан: принципи, переваги та практичне застосування [Електронний ресурс]. – Режим доступу: <https://laba.ua/blog/1529-kanban-principyu-i-polza>
2. Що таке REST API і як він працює? [Електронний ресурс]. – Режим доступу: <https://dan-it.com.ua/uk/blog/shho-take-rest-api-i-yak-vin-praczuuye/>
3. React: Tutorial: Intro to React [Електронний ресурс]. – Режим доступу: <https://uk.legacy.reactjs.org/tutorial/tutorial.html>
4. About Node.js® [Електронний <https://nodejs.org/uk/about>]