

Міністерство освіти і науки України

Національний технічний університет України «КПІ

імені Ігоря Сікорського»

Факультет інформатики та обчислювальної техніки

Кафедра інформатики та програмної інженерії

**ЗВІТ**

лабораторної роботи №1

з курсу «Безпека програмного забезпечення»

Перевірив:

Курченко О. А.

Виконав:

Студент Гр. ІП-01

Пашковський Є. С.

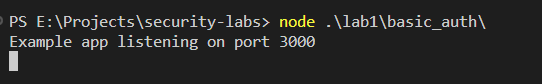
Київ 2023

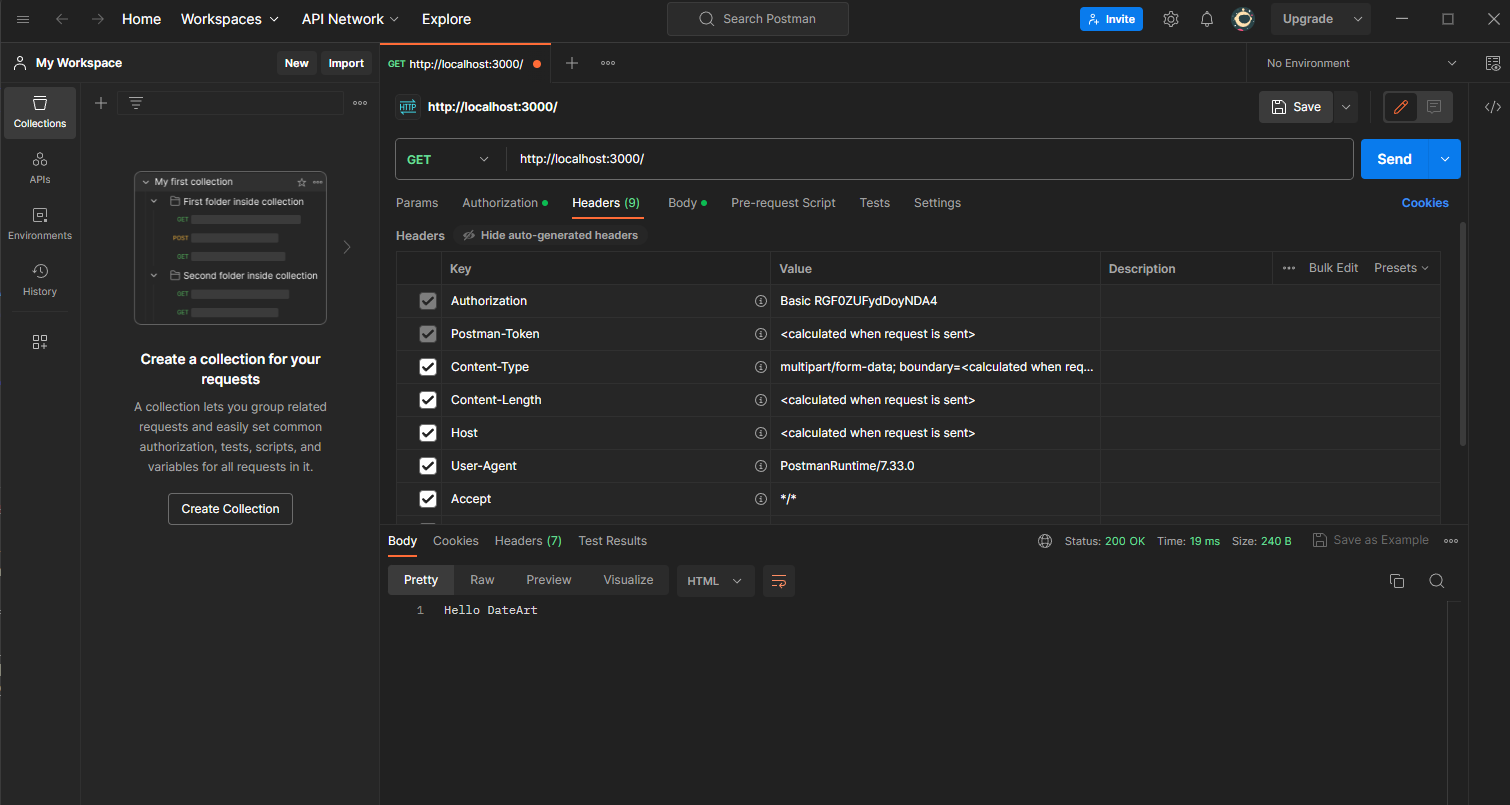
Завдання

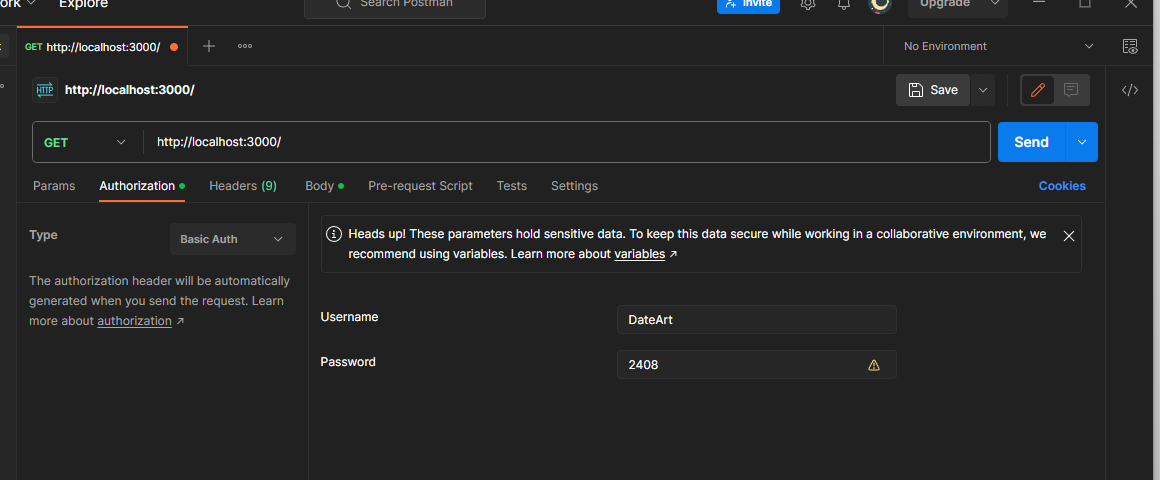
Викачати репозиторій з лекціями https://github.com/Kreolwolf1/auth\_examples. Запустити кожен з 3 аплікейшенів та зробити скріншити запитів до серверу. Для отримання додаткового балу: модифікувати token\_auth аплікейшен змінивши токен на JWT.

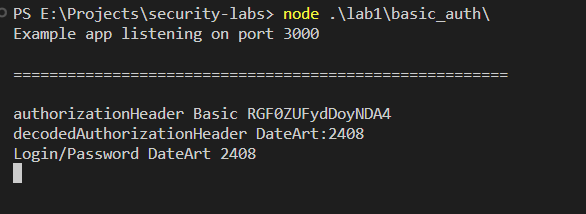
Хід роботи

1. Запуск та тестування першого додатку

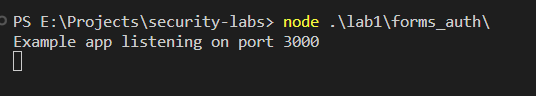


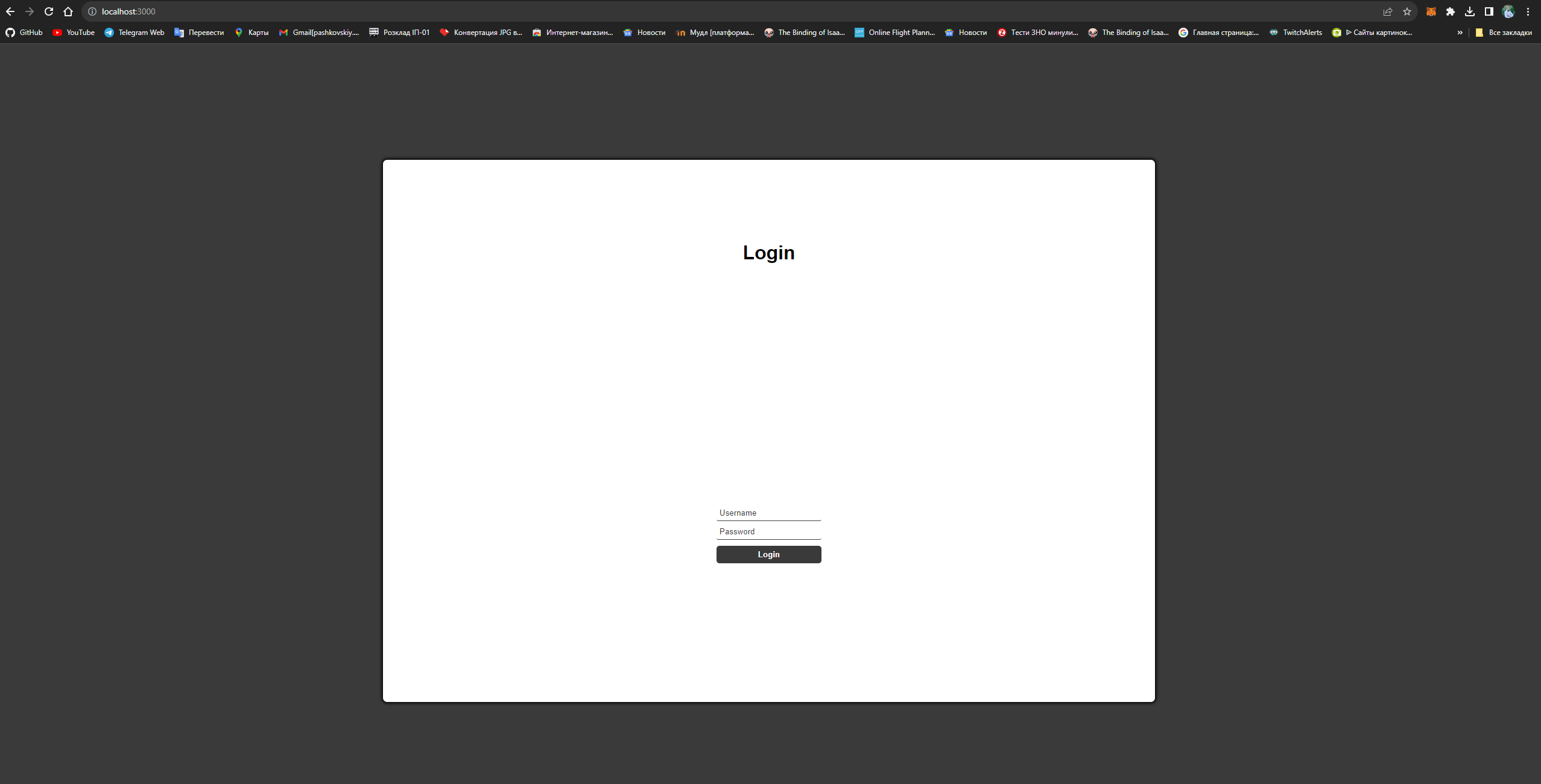


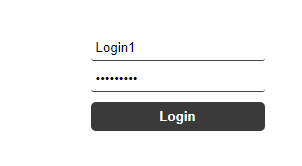


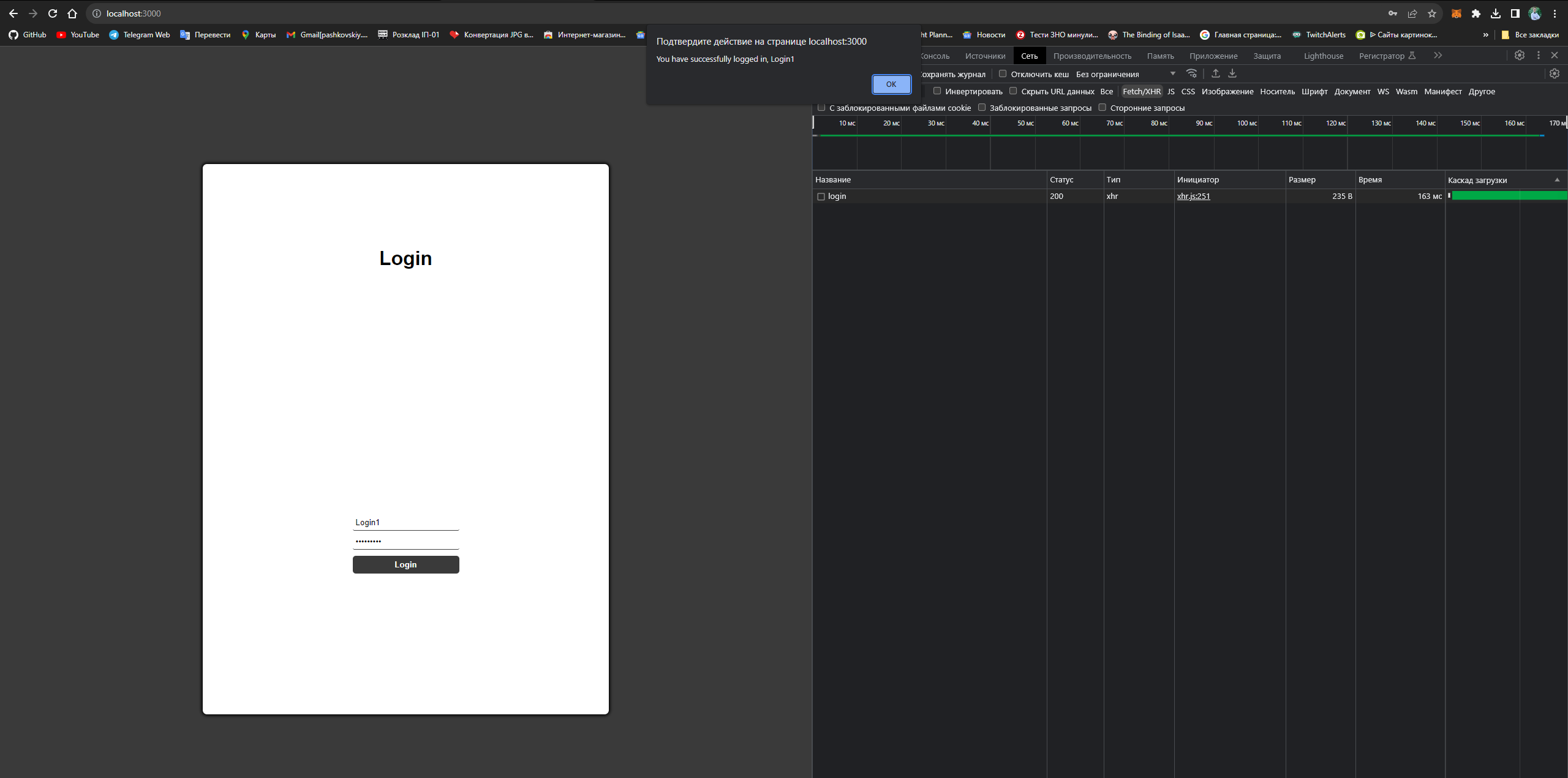


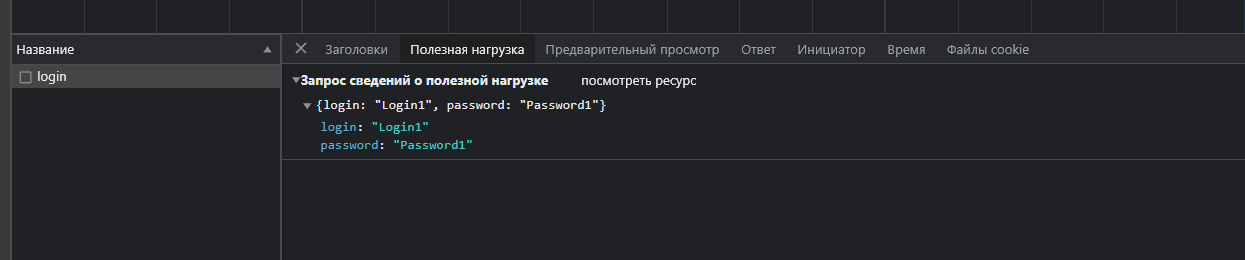
1. Запуск та тестування другого додатку

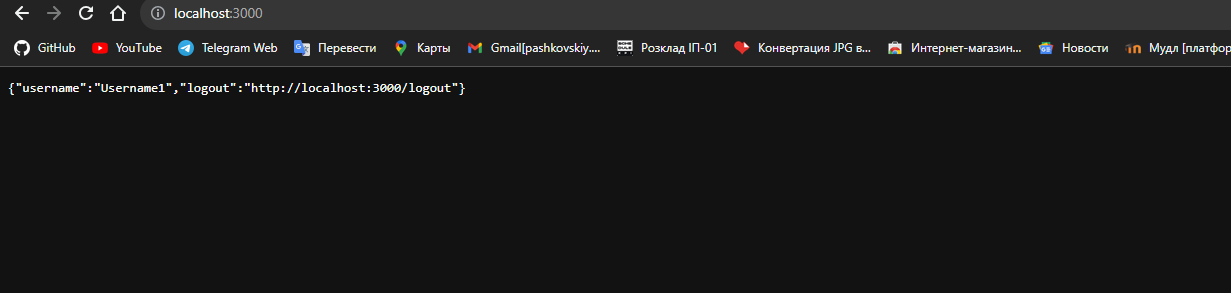


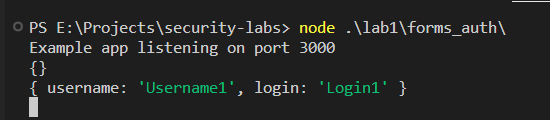




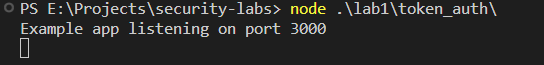


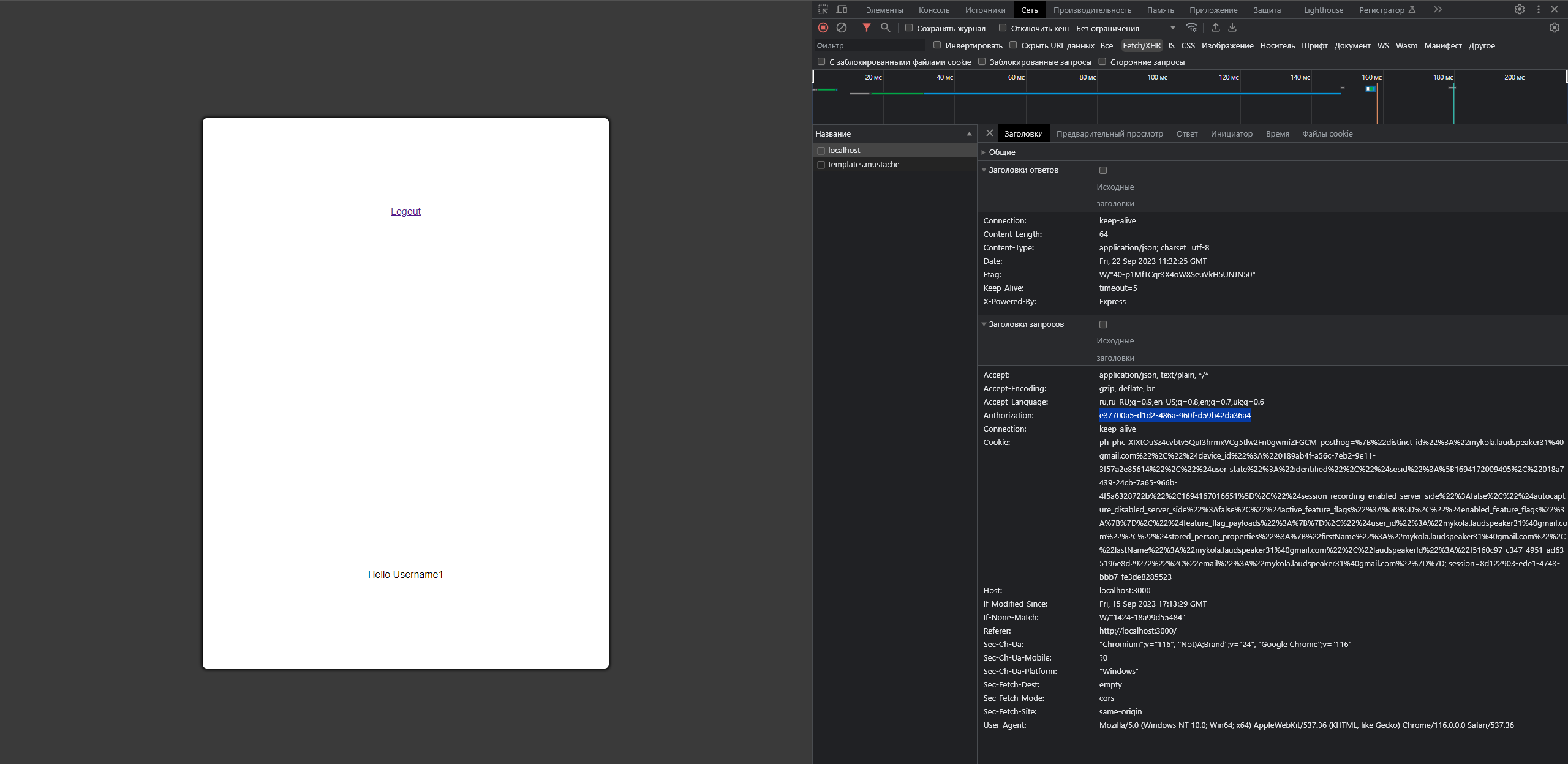






1. Запуск та тестування третього додатку





1. Авторизація по токену за допомогою json web token:

const uuid = require("uuid");

const express = require("express");

const onFinished = require("on-finished");

const bodyParser = require("body-parser");

const path = require("path");

const port = 3000;

const fs = require("fs");

const jwt = require("jsonwebtoken");

const app = express();

app.use(bodyParser.json());

app.use(bodyParser.urlencoded({ extended: true }));

// get from env

const JWT\_SECRET = "jwt\_secret";

const users = [

  {

    login: "Login",

    password: "Password",

    username: "Username",

  },

  {

    login: "Login1",

    password: "Password1",

    username: "Username1",

  },

];

app.get("/", (req, res) => {

  try {

    const token = req.get("Authorization")?.replace("Bearer ", "");

    const tokenPayload = jwt.verify(token, JWT\_SECRET);

    const user = users.find(({ login }) => login === tokenPayload.login);

    if (user) {

      return res.json({

        username: user.username,

        logout: "http://localhost:3000/logout",

      });

    }

  } catch (e) {

    console.error(e);

    res.status(401);

  }

  res.sendFile(path.join(\_\_dirname + "/index.html"));

});

app.get("/logout", (req, res) => {

  res.redirect("/");

});

app.post("/api/login", (req, res) => {

  const { login, password } = req.body;

  const user = users.find((user) => {

    if (user.login === login && user.password === password) {

      return true;

    }

    return false;

  });

  if (user) {

    const token = jwt.sign({ login }, JWT\_SECRET, {

      expiresIn: "30s",

    });

    return res.json({ token });

  }

  res.status(401).send();

});

app.listen(port, () => {

  console.log(`Example app listening on port ${port}`);

});

