

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»
Факультет інформатики та обчислювальної техніки
Кафедра Обчислювальної Техніки

Лабораторна робота №6
з дисципліни "Безпека програмного забезпечення"
Тема: "Засвоювання базових навичок роботи з OAuth2 протоколом"

Виконав:

студент групи ІП-93

Домінський В.О.

Київ 2023

Зміст

Завдання:	3
Виконання:	3
Висновок:	7
Посилання:	7

Завдання:

1. Розширити Лабораторну роботу 4, змінивши логін сторінку на стандартну від SSO провайдера, для цього, треба зробити редірект на API_DOMAIN

<https://kpi.eu.auth0.com/authorize>

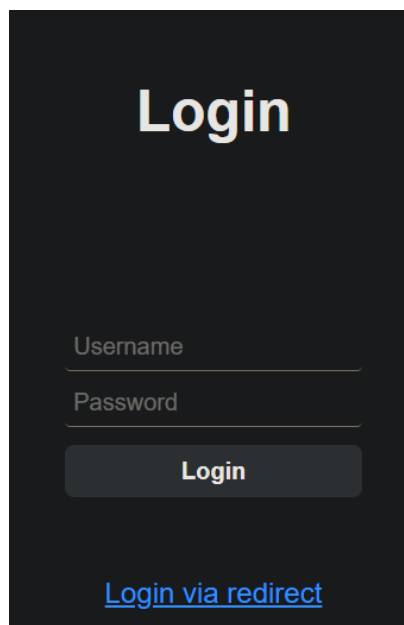
та додатково додати параметри Вашого аплікейшена

client_id, redirect_uri, response_type=code, response_mode=query

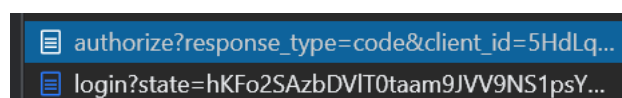
https://kpi.eu.auth0.com/authorize?client_id=JlvCO5c2IBHlAe2patn6l6q5H35qxti0&redirect_uri=http%3A%2F%2Flocalhost%3A3000&response_type=code&response_mode=query


Виконання:

Для початку було додано нову кнопку, натискаючи на яку, Вас переносить на сторінку авторизації через сторонні сервіси:




При переході у вкладці Network можемо побачити ось такі запити:





Welcome

Log in to dev-wyvlhkwn475iv1wn to continue to BPZ-Advanced.




[Forgot password?](#)

Continue


Don't have an account? [Sign up](#)

OR

 Continue with Google

Тут є 2 опції:


1. Увійти за допомогою логіну та паролю, які вже є в системі



Welcome

Log in to dev-wyvlhkwn475iv1wn to continue to BPZ-Advanced.

carolanne.gerlach81@gmail.com




[Forgot password?](#)

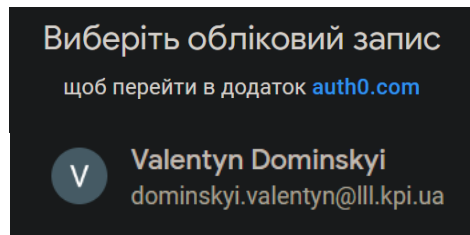
Continue

Don't have an account? [Sign up](#)

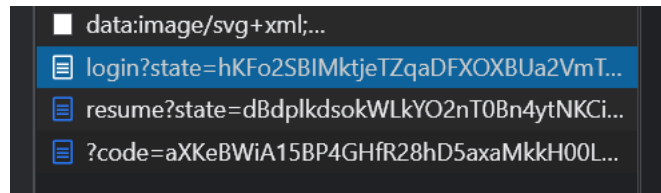
OR

 Continue with Google

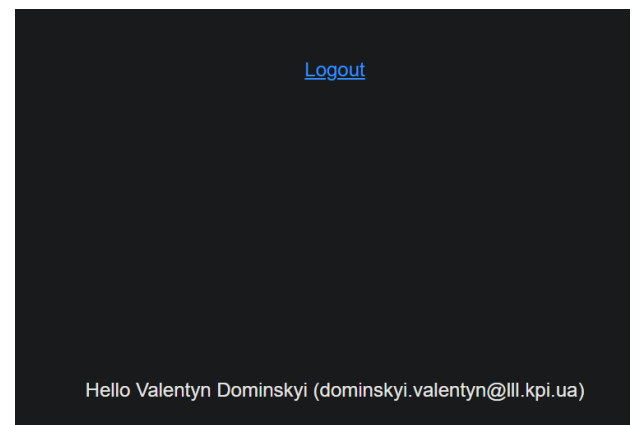
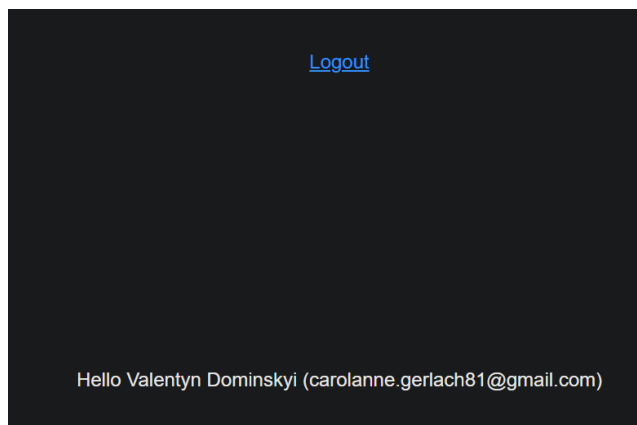
2. Увійти за допомогою акаунту Google



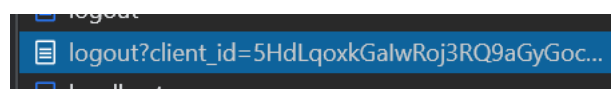
Обидва шляхи приведуть до одного й того ж результату, а саме – входу в систему:



Name	Connection	Logins	Latest Login
Valentyn Dominskyi dominskyi.valentyn@lil.kpi.ua	google-oauth2	1	a few seconds ...
Valentyn Dominskyi carolanne.gerlach81@gmail.com	Username-Password-Authenti...	16	a minute ago



При натисканні на Logout Нас переносить на сторінку авторизації з автоматичним виходом з акаунту.



Перейдемо до програмної частини. З'явилося 2 додаткових функції для логіну та логауту:

```
app.get('/logout', (req, res) => {
  res.redirect(`https://${config.domain}/v2/logout?client_id=${config.clientId}&returnTo=http://localhost:3000/`);
});

app.get('/login', (req, res) => {
  res.redirect(`https://${config.domain}/authorize?response_type=code&client_id=${config.clientId}&redirect_uri=http://localhost:3000/&scope=offline_access&audience=${config.audience}`);
});
```

Щоб вони працювали треба перейти на сайт auth0 та в налаштуваннях додати сторінки для редіректів:

Application URIs

Application Login URI

`https://myapp.org/login`

In some scenarios, Auth0 will need to redirect your user to this URI. This URI needs to point to a route in your tenant's `/authorize` endpoint.

Allowed Callback URLs

`http://localhost:3000/`

After the user authenticates we will call your application. You can specify multiple valid URLs by comma separating them. You can use different environments like QA or test (`https://`) otherwise the callback URL must be a valid URL. You can use custom URI schemes for native applications. You can use [Organizational URIs](#).

Allowed Logout URLs

`http://localhost:3000/`

Додав до `app.use` оновлення `user token` на фронтенді:

```
app.get('/', async (req, res) => {
  if (req.query.code) {
    const userTokenData = await getTokenByCode(req.query.code);
    return res.send(`
      <script>
        sessionStorage.setItem('session', '${JSON.stringify({token: userTokenData.access_token})}');
        document.location = '/';
      </script>
    `);
  }
});
```

Спочатку ловимо редірект від провайдера auth0, встановлюємо токен в `SessionStorage` та переходимо на основний url. Маючи токен нас пускає в систему.

Код функції для отримання токєну:

```
const codeTokenOptions = (code) => ({
  method: 'POST',
  url: `https://${config.domain}/oauth/token`,
  headers: {
    'content-type': 'application/x-www-form-urlencoded',
  },
  data: new URLSearchParams( {
    grant_type: 'authorization_code',
    client_id: config.clientId,
    client_secret: config.clientSecret,
    code,
    redirect_uri: 'http://localhost:3000',
  })
});

const getTokenByCode = async (code) => {
  try {
    const body = await axios.request(codeTokenOptions(code));
    return body.data;
  } catch (error) {
    console.log(error);
    return null;
  }
};
```

Висновок:

Під час виконання роботи Я дізнався про Authorization Code Flow, додав SSO provider та навчився налаштовувати перенаправлення користувачів, як в кодї, так і на сайті Auth0

Посилання:

- [Проект на GitHub](#)