

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ НАЦІОНАЛЬНИЙ
УНІВЕРСИТЕТ “ЛЬВІВСЬКА ПОЛІТЕХНІКА”**

Кафедра систем штучного інтелекту

Лабораторна робота №5
з курсу “Веб-технології та веб-дизайн”

Виконав: студент групи КН-208
Музика Михайло
Викладач:
Гасько Р.Т.

Львів - 2020

ХІД РОБОТИ

Напишемо 3 тести за допомогою *Jest-dom*. Спочатку підключимо усі необхідні файли та бібліотеки:

```
import React from "react";
import ReactDOM from "react-dom";
import { create } from "react-test-renderer";
import Footer from "../components/footer";
import request from "../components/Utils/RequestWrapper";
import ProfilePage from "../components/ProfilePage";
```

Перший тест буде перевіряти зв'язок клієнта з сервером за допомогою функції, що дістає усі замовлення, залежно від того, кому ці замовлення будуть відображатися:

```
it("server interaction by getting orders", async () => {
  const result = await request({
    method: "get",
    url: "ads/get_all/5c76f2b1-7860-4166-8179-4bf6da0ed116c/",
  });
  expect(result.data.value).toBeTruthy();
});
```

Другий тест буде перевіряти, чи коректно рендериться елемент. Для цього створюємо блок *div* та спочатку засовуємо туди *footer*, а потім його ж і дістаємо:

```
it("correct component rendering", () => {
  const div = document.createElement("div");
  ReactDOM.render(<Footer />, div);
  ReactDOM.unmountComponentAtNode(div);
});
```

Останній тест буде перевіряти, чи компонент рендериться ідеально, тобто чи збігається він з *snapshot*, який був зроблений:

```
it("userOffice renders as expected", () => {
  let elem = create(<ProfilePage />);
  expect(elem).toMatchSnapshot();
});
```

Перевіримо роботу наших тестів:

```
mihaylo@DESKTOP-HP MINGW64 /d/Універ/2 курс/2 семестр/Веб/LAB5/Restavratsia/restavratsia1/ClientApp (ui-fixes)
$ npm test
```

```
Test Suites: 1 failed, 1 total
Tests:       1 failed, 2 passed, 3 total
Snapshots:   1 passed, 1 total
Time:        9.857s, estimated 14s
Ran all test suites related to changed files.
```

ВИСНОВОК

Для тестування на цій роботі я використовував фреймворк *Jest*. Обрав я його через простоту роботи з ним, а також через можливість робити *snapshots*.