

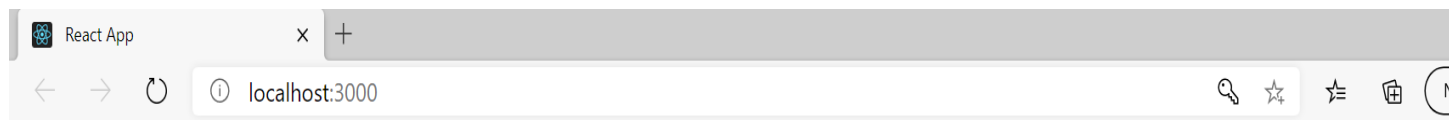
Universitatea Politehnica Timișoara
Facultatea de Automatică și Calculatoare

Verificare și Validare Software
Webserver

Student: Iacov Cristian

Capitol 1 Introducere

- Scopul acestui proiect este de a ne familiariza cu diferite tehnici de testare.În acest sens am creat un webserver folosind pachetul express.js din framework-ul nodejs și am creat o interfață grafică folosind framework-ul React.
- Serverul are posibilitatea de logare atât cu un cont de Admin cât și cu un cont de user normal.Pagina de Admin conține informații despre server: în ce stare se afla,pe ce port rulează.Și poate se schimbe starea serverului din Running în Maintenance sau chiar să oprească serverul.Un user obișnuit se poate loga și poate naviga prin paginile oferite de interfață grafica.



Admin page

Server state is RUNNING

Server is running on port 3001

Stop server Maintenance

Capitol 2 Tehnici de testare folosite

- Prima etapa de testare a fost module testing. Testarea modulelor este definită ca un tip de testare software, care verifică subprograme individuale, subrutine, clase sau proceduri într-un program. În loc să testați întregul program software simultan, testarea modulului recomandă testarea blocurilor mai mici ale programului. Testarea modulelor este în mare măsură orientată către o cutie albă. Obiectivul realizării modulului, testarea nu este de a demonstra funcționarea corectă a modulului, ci de a demonstra prezența unei erori în modul.
- Cea de a doua etapa de testare a constat într-o analiza statică a codului. Pentru analiza statică am folosit ESLint, un analizor static pentru identificarea patter-urilor problematice în javascript. În urma analizei statice am primit mesaje de eroare chiar dacă codul meu compila și aplicația mergea. Aceste erori pot fi chiar critice pentru o aplicație.

```
C:\Users\crist\Desktop\VVSPRJ\vvspriect>npx eslint server.js
Warning: React version not specified in eslint-plugin-react settings. See https://github.com/yannickcr/eslint-plugin-react#configuration .

C:\Users\crist\Desktop\VVSPRJ\vvspriect>server.js
  1:17  error  'require' is not defined            no-undef
  3:14  error  'require' is not defined            no-undef
  4:20  error  'require' is not defined            no-undef
 11:7   error  'my_users' is assigned a value but no-unused-vars
       error  never used

✖ 4 problems (4 errors, 0 warnings)

C:\Users\crist\Desktop\VVSPRJ\vvspriect>

C:\Users\crist\Desktop\VVSPRJ\vvspriect\src>npx eslint App.js

C:\Users\crist\Desktop\VVSPRJ\vvspriect\src>App.js
 22:10  error  Parsing error: Unexpected token =

✖ 1 problem (1 error, 0 warnings)
```

- O subetapa în dezvoltarea aplicației a fost și trecerea aplicației și într-un tool de Analiza Dinamica. Pentru acest lucru am folosit VisualVm

- Următoarea etapa de testare a aplicației a constat în testarea GUI și testarea web. Pentru testarea GUI am folosit un framework de testare, Jest. Acesta îți permite să scrii teste manuale și să verifici dacă elementele folosite de UI chiar funcționează. Cum framework-ul React se bazează pe componente am scris teste pentru a vedea dacă acele componente se randează și funcționează. Framework-ul păstrează într-un fișier o stare inițială, iar atunci când testul este rulat vă compară starea în care este cu cea salvată.

```

8   });
9   test('renders correctly', () => {
10     const tree = renderer.create(<Login />).toJSON();
11     expect(tree).toMatchSnapshot();
12   });

```

- Testarea Web a fost ultima etapă de validare a proiectului meu. Pentru aceasta am folosit Selenium. Selenium este un cadru portabil pentru testarea aplicațiilor web. Selenium oferă un instrument de redare pentru crearea de teste funcționale, fără a fi nevoie să înveți un limbaj de script pentru test. Am creat 4 teste folosind Selenium, unul pentru logare ca și Admin, unul pentru logare ca user normal și două pentru stările în care se afla serverul, respective Maintenance sau Stopped.

```

var webdriver = require('selenium-webdriver'),

By = webdriver.By,
until = webdriver.until;

var driver = new webdriver.Builder()
  .forBrowser('chrome')
  .build();

driver.get('http://localhost:3000/');
driver.findElement(By.id('a')).sendKeys('admin');
driver.findElement(By.id('b')).sendKeys('admin');
driver.findElement(By.id('c')).click();
driver.wait(webdriver.until.elementLocated(webdriver.By.name('stop')), 1000);
driver.findElement(By.id('stop')).click();

console.log('success');

```

Capitol 3 Concluzii

Atât cursul cât și laboratorul mi s-au părut foarte interesante ,ni s-a prezentat întreg procesul de verificare și validare în cadrul unui proiect,un lucru extrem de util.Am învățat multe lucruri noi și cel mai important am învățat unde să caut aceste lucruri.

BIBLIOGRAFIE:

<https://crossbrowstesting.com/blog/test-automation/sleeps-waits-in-selenium/>

<https://jestjs.io/docs/en/tutorial-react>

<https://visualvm.github.io/graal.html#:~:text=GraalVM%20is%20a%20universal%20virtual,interoperability%20in%20a%20shared%20runtime.>

<https://stackoverflow.com/>