

Проектна документација по предметот Софтверски квалитет и тестирање

Автоматизирано тестирање на Spring Boot Web апликација

Студент: Ангела Цветковска 221001

Линк до GitHub репозиториум: <https://github.com/angi45/SKIT>

## Вовед

Во рамки на предметот „Софтверски квалитет и тестирање“ изработен е проект чија цел е да демонстрира спроведување на автоматизирано тестирање на веб апликација развиена со Spring Boot. Во проектот се опфатени различни техники за тестирање, покривајќи повеќе слоеви на апликацијата: од сервисниот и репозиториумскиот слој, преку веб контролерите, па сè до тестирање на корисничкиот интерфејс и сигурносните аспекти.

Апликацијата претставува систем за управување со јадења (*Dish*) и готвачи (*Chef*). Корисниците можат да:

- Гледаат листа на јадења и готвачи (USER & ADMIN)
- Креираат, уредуваат и бришат јадења и готвачи (само ADMIN)
- Да се најават преку login страница

Функционалностите се обезбедени преку: Spring Boot, Thymeleaf за UI и Spring Security за role-based пристап

Улогите се:

| Улога | Овластувања          |
|-------|----------------------|
| USER  | само преглед         |
| ADMIN | целосен CRUD пристап |

Ова овозможува да се тестира реална и сигурна веб апликација со повеќеслојни сценарија.

Во проектот се реализирани следниве типови тестирање:

| ТИП НА ТЕСТ       | ШТО СЕ ТЕСТИРА                              | АЛАТКА                    |
|-------------------|---|---------------------------|
| UNIT TESTS        | Сервис логика, исклучоци                    | Mockito                   |
| REPOSITORY TESTS  | Запис, читање и бришење од база             | Testcontainers            |
| WEB MVC TESTS     | Controllers: статуси, пренасочувања, view-и | Spring MVC Test (MockMvc) |
| INTEGRATION TESTS | End-to-end backend flow                     | MockMvc + реална база     |
| UI TESTS          | Логирање, CRUD, UI елементи                 | Selenium WebDriver        |
| SECURITY TESTS    | Role-based забрана/дозвола                  | Spring Security Test      |

На овој начин се покриени сите ризични точки на системот.

## Користени алатки и технологии

| ТЕХНОЛОГИЈА / АЛАТКА             | УЛОГА ВО ПРОЕКТОТ   |
|----------------------------------|---|
| <b>SPRING MVC TEST FRAMEWORK</b> | Тестирање на контролери без стартување сервер                 |
| <b>SELENIUM WEBDRIVER</b>        | UI E2E тестирање со вистински browser                         |
| <b>TESTCONTAINERS</b>            | Интеграционо тестирање со реална PostgreSQL база во контејнер |
| <b>MOCKITO</b>                   | Тестови со mock објекти                                       |
| <b>SPRING SECURITY TEST</b>      | Тестирање на пристап  |

## Реализирани тест сценарија

### Backend:

- Креирање, уредување и бришење Chef/Dish преку контролер
- Забрана на CRUD операции за корисник со улога USER
- Проверка дека промените се запишани во база

### UI :

- Најава на ADMIN корисник
- Отворање форма за додавање јадење
- Креирање ново јадење преку UI чекори
- Проверка дека јадењето е прикажано на листата
- USER не гледа Delete/Edit копчиња → role-based visibility

## Постигнати резултати

Сите тестови успешно поминуваат.

Со тоа е потврдено дека:

- Апликацијата е функционална
- UI корисничката интеракција работи како што треба
- PostgreSQL базата е правилно поврзана со backend логиката
- Системот е безбеден и не дозволува недозволени операции

Ова покажува високо ниво на стабилност и квалитет на софтверот.

## Заклучок

Со изработката на проектот е демонстрирано дека софтверското тестирање има критична улога во развојот на сигурни и стабилни системи.

Во проектот се применети:

- повеќе типови тестови
- тестирање на повеќе слоеви
- напредни алатки и технологии
- реални сценарија со автентикација и UI интеракција