

Задача за Automation QA: Пишување на тестови за апликацијата „Едноставен систем за инвентар на производи“

1. Цел

Изработка на **Unit**, **e2e** и **Integration** тестови на back-end(Spring Boot-Java) и **Unit** тестови на front-end(Angular 19.2.X-TypeScript) на проект кој е едноставен систем за инвентар и производи.

2. Опис на апликацијата

Се работи за основна веб-апликација каде што корисниците можат да ги прават следниве операции:

- Додавање, прегледување, ажурирање и бришење производи во инвентарот.
- Прегледување на деталите за производот во едноставен кориснички интерфејс.
- Зачувување на податоците за производот во релациона база на податоци (SQL Server PostgreSQL).

3. Стартување на проектите

На двата проекта имате детално објаснување во README фајл на секој од нив, за како да ги стартувате и дополнително како да се поврзете на База на податоци(PostgreSQL). Доколку имате проблеми околу стартувањето на проектите контактирајте не.

Откако ќе ги стартувате проектите, со цел проверка дали е се во ред, додадете неколку производи, релоадирајте и проверете дали истите ги има на интерфејс и во база. Доколку е се во ред продолжете на нареден чекор.

4. Имплементација на тестови на Backend

- На патека InventoryManagementBackend-main\src\test\java\org\example\inventorymanagementbackend се дефинирани 3 packages: unit, integration и e2e. Пишувајте ги тестовите во зависност од типот во соодветните папки.
- Сите потребни зависности за интеграција на сите видови на тестови веќе се снимени во проектот(**jUnit, Mockito, Test Containers, Selenium**). Така што може да почнете директно со пишување на тестовите. Или доколку сакате да пишувате тестови со други зависности, истите можете да ги снимите во pom.xml фајлот и потоа да ги користите истите во соодветните папки.

1. Unit тестови

- Напишете најмалку 3 поедноставни или 2 посложени unit тестови од методите во апликацијата со користење на алатките **jUnit Jupiter**, и **Mockito**(или снимете и искористете друга алатка/и)

2. Integration тестови

- Напишете најмалку 2 integration тестови за REST API-њата од проектот во соодветниот package со користење на **Test Containers**(или снимете и искористете друга алатка/и).
- **НАПОМЕНА: За да ги пишувате тестовите со Test Containers потребно е на Вашата локална машина да имате снимено и**

стартувано Docker Desktop. Процедурата за инсталација и старт на Docker Desktop е релативно едноставна

3. е2е тестови

- Напишете најмалку 1 посложен или 2 поедноставни е2е тестови во Селениум(или снимете и искористете друга алатка).

5. Имплементација на тестови на Frontend

- Тестовите на Angular проектот треба да ги пишувате со комбинација од две библиотеки: **Jasmine** и **Karma**. Истите се веќе снимени на проектот, така што можете веднаш да почнете со пишување на истите.
- Тип на тестови на front-end е **само Unit**
- Концептот на пишување на тестови на Angular проект е отворање на специфичен фајл за секоја компонента. Веќе е креиран еден специфичен фајл `product-list.component.spec.ts` за соодветната компонента. Pattern-от е ист за било која компонента.
- Потребно е да напишете најмалку 3 теста за 3 методи

*****НАПОМЕНА 1:** Доколку не сте успеале да исполните некои од овие барања не значи дека по автоматизам нема да бидете примени. Сфатете го „најмалку“ како „добро би било“ и не се обесхрабрувајте, пратете ни го решението дури и доколку не е целосно.



*****НАПОМЕНА 2:** Доколку пишувате тестови со други зависимости задолжително заедно со source кодот напоменете што се користеле и додаде во `rom.xml` и доколку сметате дека е потребно додадете и кратки инструкции за стартување на истите.

*****ДОБРО БИ БИЛО:** Доколку заедно со одговорот приклучите и видео фајл со извршување на тестовите.

6. Опционално (Бонус поени)

Ќе се издвоите ако:

- Користите соодветен јасен pattern (i.e. AAA)
- Детектирате и додадете тестови за edge-cases

7. Упатство за прикачување на кодот

- Поставете го кодот на Ваш public GitHub Repository.
- Доколку сметате дека има потреба Вклучете README фајл со:
 - Чекори за стартување на тестовите(ова само доколку имате некои нови зависимости или специфични работи кои треба да се подесат)

- Доколку сте направиле нешто плус од наведеното споменете го истото во README

GitHub link-от од hero-то пратете го на contact@interway.com.mk .

6. Бидете креативни

Издвојте се од другите. Збогатете го Вашето решение надвор од конкретните задачи и споменете го истото во README од решението. Вашата самоиницијатива и желба нема да остане незабележана.

Со среќа! 😊