



Лабораториски вежби по Дизајн и архитектура на софтвер - Домашна работа 4

Во четвртата домашна треба да го рефакторирате кодот што го напишавте за третата домашна, да префрлите дел од функционалностите на апликацијата на АПИ/микросервиси и да ја подготвите апликацијата и сервисите за работа во контејнер. На крај ќе треба сето ова на го инсталирате во облак.

1 Рефакторирање на кодот

Рефакторирајте го кодот на апликацијата. Кодот кој ќе го предадете треба да биде структуриран според некој од софтверските шаблони кои ги учевте на предавањата. Освен тоа, тој треба да биде

- лесно разбирлив,
- да следи конзистентен стил во именувањето на променливите и методите,
- имињата на променливите и методите да бидат јасни и описни,
- да биде добро документиран (најдобро е самите имиња да служат како документација, но ако има потреба да се објаснат линии код; доколку на некое место името на методот или параметрите не се јасни и/или еднозначни, тие треба да се објаснат во коментар),
- да биде лесен за одржување,
- да може повторно да се употребува,
- да нема повторување,
- алгоритмите што ги користите да не бидат непотребно комплексни.

2 АПИ/Микросервиси

Дел од функционалностите на апликацијата треба да се претворат во микросервиси кои ќе комуницираат преку АПИ. Микросервиси се најмали компоненти на апликацијата кои се градат и функционираат независно. Вие треба да одвоите дел од функционалностите на вашата апликација и да ги



програмирајте и компајлирајте така што да можат да се извршуваат самостојно. За секој микросервис потребно е да имате посебен проект. Потоа треба да ја овозможите нивната меѓусебна комуникација, како и комуникацијата со останатиот дел на апликацијата преку АПИ. Следните туторијали даваат примери за развивање на апликации со микросервисна архитектура со Spring, но можете истите чекори да ги примените и адаптирате за технологијата која ја користите. Туторијалите можете да ги најдете на <https://spring.io/blog/2015/07/14/microservices-with-spring>
<https://www.javatpoint.com/microservices>
<https://developer.okta.com/blog/2019/05/22/java-microservices-spring-boot-spring-cloud>

3 Контејнеризација и инсталација во облак

Подготовете ја апликацијата за работа во контејнер и инсталирајте ја во облак. Контејнерите се стандардни софтверски единици кои го пакуваат кодот и неговите зависности. Спакувајте ја вашата апликација на овој начин. За подготвување на контејнерите можете да користите Docker. Повеќе за тоа како да го направите тоа можете да прочитате на <https://docs.docker.com/get-started/>. Пример за подготвување на апликација која користи микросервиси со Docker можете да најдете на <https://training.play-with-docker.com/microservice-orchestration/>. Откако ќе го направите ова, инсталирајте ја вашата апликација во облак. Апликацијата можете да ја поставите на Azure Web Apps, AWS Elastic Beanstalk, и други сервиси. Бесплатни кориснички сметки за студенти за Azure и AWS можете да креирате на следните линкови:
<https://azure.microsoft.com/en-us/free/students/>
<https://aws.amazon.com/education/awseducate/>
Имајте на ум дека ќе мора да почекате неколку денови за потврдата на вашиот статус на студент, која е потребна за да ги користите сервисите.

Оваа домашна работете ја исто како што беше опишано во текстот за претходната. Можете да продолжите да работите во истиот фолдер, кој ќе го преименувате во Домашна 3 и 4. Секој микросервис ставете го во посебен подфолдер. Притоа, **публикуваната верзија на линкот што ќе го предадете за третата домашна мора да остане непроменета** по рокот за предавање на таа домашна. За оваа цел, осигурајте се дека не работите на гранката што е поврзана со платформата што ја користите за публикување/публикуваната верзија на апликацијата, и/или дека automatic deploys се исклучени. За оваа домашна публикувајте ја апликацијата на нов линк кој исто така ќе го ставите во About делот на проектот на GitHub.

Крајниот рок за предавање на четвртата домашна работа е 28.01.2024.