

QA insener

Testija ülesanne on leida tarkvaras vigu ja kirjutada korrektselt testskipte. Kursus sobib neile, kes alustab nullist ja kellel puudub IT-alane kogemus.

## Kursusest

### Eesmärk:

kursuse lõpuks oskab õppija välja töötada teksti dokumentatsiooni, rakendada erinevat tüüpi testimist ning kasutada kaasaegseid tööriistu veebi- ja mobiilirakenduste, API-de ja andmebaaside testimiseks.

### Õpitulemused

Pärast kursuse läbimist õpilane:

tunneb testide kavandamise põhimõtteid ja võtteid;

kasutab erinevaid testimisvahendeid;

mõistab tarkvara arendusprotsessi ja testimise rolli selles protsessis;

teab, kuidas koostada testi dokumentatsiooni(checklist / test case / bug report / testimisplaan / testiaruanne);

teab, kuidas planeerida testimise protsessi;

mõistab veebi- ja mobiilirakenduste testimise põhimõtteid;

mõistab klient-server arhitektuuri, API põhimõtteid;

teab, kuidas kirjutada SQL päringuid ja oskab töötada andmebaasiga;

automatiseerib API testimiskripte Postmaniga.

### Plaan:

Kursuse kestus on 12 nädalat.

### Sihtrühm, kursusele pääsemise tingimused

Kellele sobib?

Kursus sobib IT-valdkonnas töötavatele ja tööle hakkamist planeerivatele inimestele, kes soovivad omandada Eestis ja välismaal töötamiseks vajalikud algteadmised ja -oskused.

Õppekavarühm:

153135 Tarkvara ja rakenduste arendus ning analüüs

### Treeningu alustamise tingimused:

personaalarvuti või sülearvuti kasutamise võimalus treeningu ajal.

Minimaalne osalejate arv grupis kursuse läbiviimiseks on 5 inimest.  
Tunniplaan võib muutuda, sellest me alati anname eelnevalt teada.

## Täpsem kirjeldus

### Õppimise protsess:

klassiruumis (veebikoosolekud) ja praktilised tööd, mida õpilane peab sooritama kursustevälistel päevadel. Tavaliselt toimub koolitus tööpäeviti.

Kursuse kestus on 12 nädalat. Tunnid toimuvad veebis.

Kursus algab vähemalt 5-liikmelise grupi olemasolul.

### Pärast kursuse läbimist õpilane:

tunneb testide kavandamise põhimõtteid ja võtteid;

kasutab erinevaid testimisvahendeid;

mõistab tarkvara arendusprotsessi ja testimise rolli selles protsessis;

teab, kuidas koostada testi dokumentatsiooni(checklist / test case / bug report / testimisplaan / testiaruanne );

teab, kuidas planeerida testimise protsessi;

mõistab veebi- ja mobiilirakenduste testimise põhimõtteid;

mõistab klient-server arhitektuuri, API põhimõtteid;

teab, kuidas kirjutada SQL päringuid ja oskab töötada andmebaasiga;

automatiseerib API testimiskripte Postmaniga.

Kursuse õppekava on saadaval [sellelt lingilt](#).

Koolitusprogramm loodi kutsestandardi Tarkvaraarendaja, tase 6, pädevus B.3.5 alusel, võttes arvesse Rahvusvahelise Tarkvaratestimise Kvalifikatsiooninõukogu (ISTQB) baastaseme sertifitseerimise nõudeid ja kaasaegseid suundumusi tarkvara testimise valdkonnas.

## Kursuse maht ja struktuur, vahendid

**Maht:** 96 akadeemilist tundi, sealhulgas auditoorset tööd 48 tundi (veebis), 48 tundi praktilist tööd (2 ak.tunni kohta 24 ülesannet).

Praktilised ülesanded viiakse läbi Google Documents

keskkonnas ja neid kontrollib õpetaja. Iga töö kohta saadetakse üksikasjalik tagasiside.

### **Harjutused sisaldavad:**

ülesanded testide kavandamise põhimõtete ja tehnikate kohta, erinevate testimisvahendite valdamine; tarkvara arendusprotsessi ülesanded ja testimise roll selles protsessis; ülesanded testimise dokumentatsiooni koostamiseks (checklist / test case / bug report / testimisplaan / testiaruanne ); testimise planeerimisprotsessi valdamine; ülesanded veebi- ja mobiilirakenduste testimiseks; klient-server arhitektuuri valdamine, API põhimõtete valdamine; SQL päringute ja andmebaasidega töötamise ülesanded; ülesanded API testskriptide automatiseerimiseks Postmani abil.

Täiendav iseseisev töö sisaldab teemakohaste lisamaterjalide lugemist ja õppevideote vaatamist.

### **Õppimiskeskond:**

Koolitus toimub veebis, platvormil Zoom, Google Documents.

### **Õppematerjalid**

Õpetajad valmistavad ette kõik vajalikud materjalid ja märkmed, samuti jagavad mugavaid testide dokumentatsiooni malle.

Kõik need materjalid on kursuse üliõpilaste edasises töös suureks kasuks.

Samuti antakse soovitusi täiendõppeks: raamatud, artiklid, blogid, tasuta videod jne.

### **Õppemeetodid:**

Auditoorne (veebi)töö 48 ak.h: loengud, näidete arutelu.

Praktiline töö 48 akadeemilist tundi sisaldab:

ülesannete täitmist, tarkvara testimise läbiviimist, testi dokumentatsiooni koostamist.

Kursuse lõpus toimub läbitud materjali koondamiseks veebipõhine test ning antakse nõuandeid lisakirjanduse, videoklippide ja artiklite lugemiseks.

Arutelu peetakse kinnises grupis Telegramis.

## **Hindamise ja lõpetamise tingimused**

### **Hindamine**

Kursuse edukaks läbimiseks on vaja osaleda vähemalt 80% akadeemilistes ja praktilistes tundides ning täita kõik kodused ülesanded. Õpiväljundite saavutamist hinnatakse praktilise töö ja lõputesti kaudu.

**Tunnistus** väljastatakse õpilasele, kes on osalenud vähemalt 70% õppetundidest, täitnud vähemalt 90% praktilistest ülesannetest ja sooritanud edukalt lõputesti.

**Tõend** koolitusel osalemise või koolituse läbimise kohta väljastatakse õpilasele, kui õpitulemusi ei saavutatud, kuid õppija osales koolitusel.

**Tõend** väljastatakse vastavalt osaletud õppetundide arvule, kui õpilane osales vähemalt pooltes tundides.

## Üldinfo

QA Engineer kursuse eesmärk on anda õpilastele oskused ja kogemused, mis on vajalikud Junior QA insenerina alustamiseks. Programm on mõeldud algajatele, kes soovivad alustada oma karjääri IT-valdkonnas.

**Kursuse maht:** 24 õppetundi 2 akadeemilist tundi = 48 akadeemilist tundi.

**Toetus:** kogenud õpetaja toetab teid ja vastab kõikidele teie küsimustele.

**Kursuse portfoolio:** 8 praktilist projekti, mida potentsiaalsele tööandjale tutvustada saab.

### Moodul 1. Testimise alused

1. Mis on testimine? Testija ülesanded. Ülevaade programmi nõuetest. Testi dokumentatsioon: check list , bug report
  2. Katsedokumentatsioon: defekti tõsidus ja prioriteetsus. Keskkond ja versioon veaaruanne atribuutidena. Testjuhtumid ja nende olekud. Katsekomplekt, katsearuanne.
  3. Testimise liikide klassifikatsioon. Testimistasemed.
  4. Testi disain.
- Osa 1. Tarkvaranõuete analüüs. Nõuete lagunemine. Nõuete visualiseerimine MindMapiga. Testi kavandamise põhimõtted.
5. Testi disain. 2. osa. Testide kavandamise tehnikad – ekvivalentklassid ja piirväärtused. Otsuste tabel.

### Moodul 2. Veebirakenduste testimine

6. Veebirakenduste testimine. Klient-server arhitektuur. URL. HTTP. DevToolsiga töötamise põhimõtted.
7. Kahte tüüpi kasutajaliidest – GUI ja CLI. GUI elementide testimise printsiibid. Veebivormid. Valideerimine esi- ja taustaprogrammis. Vigade otsimine veebirakendusest DevToolsi abil.
8. Platvormide ja brauseriülene testimine. Adaptiivne ja responsiivne veebirakenduste disain. DevToolsi piirangud. Mis on küpsised, local storage ja cash.
9. Materjali vahetest ja kinnitamine. Tüüpilise testülesande sõelumise praktika.

## **Moodul 3. Mobiilirakenduste ja API testimine**

10. Sissejuhatus mobiilitestimisse. Mobiilirakenduste liigid. Mobiilseadme seaded. Miks me vajame emulaatoreid, simulaatoreid ja mobiilseid farme? Töötamine Android Studios.

11. Ülevaade veebirakenduste arhitektuursetest lahendustest: monoliit, teenus, mikroteenus. API põhimõte. API testimise põhimõtted, mis põhinevad REST arhitektuuristiilil. Sissejuhatus JSON-vormingusse. Töö Postmanis. Curl utiliit. API dokumentatsiooni ülevaade Swaggeri näitel. API testi disainifunktsioonid.

12. SOAP-protokollil põhinev API. XML-failide struktuur ja elemendid. XSD skeem ja selle rakendus.

## **Moodul 4. Andmebaasi ja SQL-i põhialused. Linuxi konsool**

13. Andmebaasi põhimõtted ja SQL-i tutvustus. DBMS-i tüübid. SQL-päringute koostamise praktika simulaatoris.

14. SQL osa 2. Sorteerimine ja loogilised operaatorid. Sisseehitatud funktsioonid. SQL-päringute koostamise praktika simulaatoris.

15. SQL osa 3. Uute kirjade koostamine. Andmete muutmine ning kustutamine. Alampäringud ja tabelite ühendamise JOIN-iga. SQL päringute koostamise praktika simulaatoril.

16. Operatsioonisüsteemid. Failisüsteem. Konsooli alused.

Käskude ülesehitus: võtmed ja argumendid.

17. Konsooliga töötamine.

Faili tee: absoluutne ja suhteline. Failide kopeerimine, teisaldamine ja kustutamine. VIM-i redaktor. grep ja ping utiliidid.

18. Tarkvara elutsükkel. Katseplaan ja testimisstrateegia. Testi mõõdikud.

## **Moodul 5. Simulaator ja testimise automatiseerimise alused**

19. Töö simulaatoriga. Frontend (UI) testimine.

20. Töö simulaatoriga. Taustatestimine (REST API).

21. Töö simulaatoriga. Integratsioonitestid ja töö andmebaasiga.

22. Töö simulaatoriga. Mobiilirakenduse testimine.

23. Testimise automatiseerimise põhimõtted kasutades Postmani simulaatori REST API näitel. Sissejuhatus CI/CD-sse.

24. Teeside retsenseerimine ja arutelu. Materjali kordamine ja kinnistamine.  
Treeningu tulemuste kokkuvõte.