# **Proovikivi.ee/placeholder TESTPLAAN**

# **Sissejuhatus**

## **Tutvustus**

Proovikivi on hariduslik koostööprogramm, mis aitab noortel kasvada vastutustundlikeks maailmakodanikeks. Programm ühendab ettevõtteid, riigiasutusi ja vabaühendusi luues proovikivisid ja koondades projekte. Proovikivideks kutsutakse väljakutseid, mille aluseks on Eesti riigi ja ÜRO kestliku arengu eesmärgid, nagu näiteks vaesuse vähendamine, keskkonnakaitse ning inimeste tervise parandamine. Proovikivid on vabalt valitavad ja nende lahendamiseks saavad huvilised algatada projekte, et probleemidega tegeleda.

Veebilehel on võimalik leida endale sobivaid projekte, kasutades otsingut ja sorteerimist, vastavalt huvipakkuvatele teemadele. Lisaks saavad kasutajad jälgida projekte ning neist osa võtta. Lisaks sellele on veebilehel ka võimalus luua uusi projekte, olles ise aktiivne panustaja hariduslikesse ja ühiskondlikesse algatustesse.

## **Planeeritud testimise liigid**

* Alfatestimine
  + Manuaaltestimine arenduse käigus
  + Jooksutatakse arendaja poolt
* Regressioonitestimine
  + Automaattestid arenduse lõpus
  + Jooksutatakse arendaja poolt
* Jõudlustestimine
* Koormustestimine
* Turvalisuse testimine

## **Testkeskkonna kirjeldus**

Testkeskkonnaks on esialgu arendaja arvuti. Hilisem tegevus lepitakse kliendiga kokku.

## **Testimise vahendid**

* Google docs - Dokumentatsioon
* Google Chrome - Veebiplatvormi brauser
* Robot Framework - Automaattestimine
* Visual Studio Code - Automaattestide kodeerimine
* Apache JMeter - Jõudlustestimine
* Apache JMeter - Koormustestimine
* Invicti - Turvalisuse testimine

## **Testide eest vastutajad**

* Manuaalne aflatestimine testimine - Kryslin Rass
* Automatiseeritud regressioontestimine- Liisi Loomets
* Jõudlus- ja koormustestimine - Karmen Klaasen
* Turvalisuse testimine - Tormi Viirg

# **Manuaalne alfatestimine**

Manuaaltestimisel kasutatakse halli kasti tehnikat.

Manuaalset testitakse järgmisi funktsionaalsusi:

1. Konto loomine korrektsete andmetega
2. Konto loomine, sisestades ebakorrektsed andmed
3. Sisselogimine korrektsete andmetega
4. Sisselogimine ebakorrektsete andmetega
5. Sisse logitud vaade - > Projektide otsimine ja sorteerimine
6. Sisse logitud vaade - > Projekti loomine kohustuslike väljade täitmisega
7. Sisse logitud vaade - > Projekti loomine jättes kohustuslikud väljad täitmata
8. Sisse logitud vaade - > Projekti loomise valikulistelt väljadelt tagasi liikumine
9. Sisse logitud vaade - > Projekti loomise katkestamine
10. Sisse logitud vaade - > Projekti lemmikuks lisamine
11. Sisse logitud vaade - > Kasutaja profiili andmete uuendamine korrektsete andmetega
12. Sisse logitud vaade - > Kasutaja profiili andmete uuendamine ebakorrektsete andmetega

# **Automatiseeritud regressioontestimine**

Automaattestimisel kasutatakse Robot Frameworki koos Seleniumi teegiga Visual Studio Code’is, mis võimaldab tõhusalt testida veebilehe funktsionaalsusi. Lisaks kasutatakse ChromeDriverit, mis integreerub Robot Frameworkiga ja võimaldab meil automatiseerida testikäike Chrome'i veebibrauseris.

Automatiseeritud testimisel testitakse järgmisi funktsionaalsusi:

1. Konto loomine korrektsete andmetega
2. Konto loomine, sisestades ebakorrektsed andmed
3. Sisselogimine korrektsete andmetega
4. Sisselogimine ebakorrektsete andmetega
5. Sisse logitud vaade - > Projektide otsimine ja sorteerimine
6. Sisse logitud vaade - > Projekti loomine
7. Sisse logitud vaade - > Projekti loomine jättes kohustuslikud väljad täitmata
8. Sisse logitud vaade - > Projekti loomise valikulistelt väljadelt tagasi liikumine
9. Sisse logitud vaade - > Projekti loomise katkestamine
10. Sisse logitud vaade - > Projekti lemmikuks lisamine
11. Sisse logitud vaade - > Kasutaja profiili andmete uuendamine korrektsete andmetega
12. Sisse logitud vaade - > Kasutaja profiili andmete uuendamine ebakorrektsete andmetega

# **Jõudlustestimine**

Jõudlustestimise käigus saab tuvastada veebilehe nõrkusi ja potentsiaalseid probleemikohti ning teha vajalikke kohandusi ja optimeerimisi.

Jõudlustesti läbiviimiseks kasutatakse Apache JMeter tööriista.

# **Koormustestimine**

Koormustestimise käigus uuritakse, kuidas veebileht reageerib suurele koormusele või liiklusele. See aitab kindlaks teha, kuidas veebileht käitub olukorras, kus paljud kasutajad üheaegselt sellele ligi pääsevad või kasutavad seda intensiivselt.

Koormustesti läbiviimiseks kasutatakse Apache JMeter tööriista.

# **Turvalisuse testimine**

Turvalisuse testimise käigus otsitakse potentsiaalseid turvaauke ja haavatavusi, et kaitsta tundlikke andmeid, täita eeskirju, vältida seisakuid ja kahjusid, suurendada usaldust ja mainet ning ennetada küberohtusi.

Turvalisuse testimiseks kasutatakse Invicti.

# **Testide läbimise kriteeriumid**

1. Manuaalsed alfatestid läbitakse 100% - kõik määratletud testjuhtumid läbitakse ning testitavad funktsioonid töötavad ettenähtud viisil.
2. Automatiseeritud regressioontestid läbitakse 100% - kõik Robot Framework-ga kirjutatud testid läbitakse ilma vigadeta ja katkestusteta.
3. Jõudlustesti läbimise kriteeriumid hetkel puuduvad.
4. Koormustesti läbimise kriteeriumid hetkel puuduvad.
5. Turvalisuse testid läbitakse 100% - levinumad turvariskid puuduvad.