

Marko Koskinen  
014689116

## Testaussuunnitelma ja Dokumentaatio

### Testin kohde

Testin kohteena on allekirjoittaneet ohjelmoima fullstack-harjoitustyö vuodelta 2018. Lyhyesti sanottuna kyseessä on yksinkertainen opintojen suunnitteluun tarkoitettu webbisovellus. Testattava versio tuotteesta vastaa tuotantoversioita (<https://opintojenaikatauluttaja.herokuapp.com/>). Testauksessa käytetään uusinta versiota front- ja backendistä, jotka löytyvät seuraavasta github-repositorysta: (<https://github.com/markokoskinen2037/fullstack-projekti>)

### Testauksessa käytetyt lähdedokumentit

Ainoana mahdollisena lähdedokumenttina voidaan käyttää github-repositoryn readme.md tiedostoa, joka listaa sovelluksen perus-ominaisuudet – mutta ei niiden toteutusta, eikä esim. hyväksymistestien määritelmiä. Onneksi sovelluksen ainoana kehittäjänä minulla on hyvä mielikuva siitä, miten sovelluksen ominaisuuksien pitäisi toimia käytännössä.

### Testauksen tavoitteet

Tavoitteena on saada testattua mahdollisimman moni tuotteen ominaisuus vähintään hyväksyntä-testien tasolla. Koska varsinaisia hyväksymiskriteereitä ei ole, täytyy ne keksiä itse - ennen kuin testaus voidaan aloittaa.

### Käytettävät testausmenetelmät

Testauksessa tullaan käyttämään modernia end-to-end testaukseen soveltuvaa kirjastoa nimeltä Cypress. Ainakin ennen testauksen automatisointia, testit suoritetaan paikalliselta koneelta käyttäen erillistä mlabs - palvelimella pyörivää dokumenttitietokantaa. Mahdolliset havaitut viat kirjataan githubiin projektin issueiksi. Testaaja pidättää oikeuden korjata pieniä, nopeasti korjattavia ei-kriittisiä vikoja ohjelmistotestauksen aikana kirjaamatta niitä issueiksi.

### Testauksen aikataulu ja resurssit

Testauksen tulee suorittamaan allekirjoittanut. Aikaa testaukselle ja testien automatisoinnille annetaan 1 viikko, alkaen 13.5 ja loppuen 19.5. Testaajalle ei makseta palkkaa, vaan hän suorittaa testauksen vapaaehtoisesti.

## Testien priorisointi

Koska testit tehdään vain hyväksyntä-testauksen tasolla, priorisoidaan testaus siten, että loppukäyttäjälle ensisijaisen tärkeitä toiminnallisuudet, kuten kirjautuminen testataan ensin. Eli mitä enemmän loppukäyttäjä käyttää tiettyä ominaisuutta - sitä korkeampi prioriteetti kyseisellä ominaisuudella on. Koska saatavilla ei ole статистиikkaa ominaisuuksien käyttöasteesta, priorisoinnissa käytetään omaa järjeä ja asiantuntemusta.

## Testauksen lopetusehto

Testaus voidaan katsoa riittäväksi, kun jokainen *Testattavat toiminnot*- osassa määritelty toiminto on testattu vähintään yhdellä positiivisella ja yhdellä negatiivisella syötteellä. (Tai kun testaukselle asetettu 1 viikon pituinen ajanjakso päättyy.)

## Testien automatisointi

Testien ajon automatisointi on integroitu versionhallintaan viemisen yhteyteen. Kun github havaitsee uuden push operaation, se ilmoittaa siitä TravisCI:lle webhookin avulla. Travis suorittaa virtuaalikoneessa travis.yml tiedostoon määritellyt operaatiot, eli asentaa front- ja backendin, käynnistää ne, ja ajaa viimeiseksi testit.

Jos testit menevät läpi niin TravisCI buildaa softasta tuotantoversion ja vie sen automaattisesti tuotantoon Heroku-palvelimelle.

Jos testit eivät mene läpi, TravisCI ilmoittaa versionhallinnassa, että buildi epäonnistui. Tässä tapauksessa sovelluksesta ei käännetä tuotantoversiota, eikä sitä myöskään julkaista Herokuun.

Testit ajetaan komennolla **cypress run --record --key=8d251730-42e2-42ac-916b-62e79d9f1797**. Tämä komento ajaa kaikki cypress-testit, kirjaa niiden tulokset ylös, ja käyttää avaimena kyseistä merkkijonoa.

Avaimen avulla Cypress osaa selvittää mihin Cypress-testiprojektiin kyseinen testiajo kuuluu, jolloin testin tulokset ja video niiden suorituksesta lähetetään *dashboard.cypress.io*-palvelimelle. Palvelimen selainkäyttöliittymästä oikeutetut käyttäjät pääsevät tarkastelemaan kaikkia kyseisen projektin testien suorituskertoja, lukemaan niiden lokitiedostoja ja katsomaan videon testien suorituksesta. Cypress ottaa myös näyttökaappauksia, mikäli jokin testi failaa – helpottaen näin debuggaamista.

## Testattavat toiminnot (prioriteettijärjestyksessä)

- Rekisteröinti
- Sisäänkirjautuminen
- Uloskirjautuminen
- Kurssin CRUD (luominen, listaus, muokkaus ja poisto)
- Kurssien merkkaustoiminto
- Kurssin tavoitearvosanan asettaminen
- Kurssin vaikeustason asettaminen

- Automaattisen tuntityömäärä-arvion oikeellisuus

## Testijoukot

Testijoukko1: Aloitustestit	
testin nimi	odotettu lopputulos
Sivusto on pystyssä	GET operaatio osoitteeseen localhost:3000 toimii.
Sivusto sisältää oikeita elementtejä	Sivusto renderöi näytölle kirjautumis- ja rekisteröintinäkymät.
Alustetaan tietokanta testejä varten	Saatiin yhteys online-tietokantaan ja tyhjennettiin edellisten testien tiedot tietokannasta.

Testijoukko2: Rekisteröinti				
testin nimi	name	password	email	odotettu lopputulos
onnistuu oikeilla tiedoilla	testi	salasana	testi@gmail.com	Rekisteröinti onnistuu ja käyttäjä kirjataan sisään
ei onnistu jos salasana on liian lyhyt	testi	a	testi@gmail.com	Rekisteröinti epäonnistuu ja näytetään virheilmoitus: "Salasanan tulee olla ainakin 5 merkkiä pitkä!"
ei onnistu jos email on virheellinen	testi_teuvo	salasana	testiemail.gmail.com	Rekisteröinti epäonnistuu ja näytetään virheilmoitus: "Sähköpostiosoitteesi ei kelvannut."
kirjaa käyttäjän sisään jos hän yrittää rekisteröityä olemassaolevilla tunnuksilla	testi	salasana	testiemail@gmail.com	Käyttäjä kirjataan automaattisesti sisään.

Testijoukko3: Kirjautuminen			
testin nimi	name	password	odotettu lopputulos
toimii oikealla salasanalla ja käyttäjätunnuksella	testi	salasana	käyttäjä kirjataan sisään
ei toimi virheellisellä käyttäjätunnuksella	kissa	salasana	Kirjautuminen ei onnistu. Näytetään virheilmoitus: "Virheellinen käyttäjätunnus tai salasana"
ei toimi ilman käyttäjätunnusta	""	salasana	Kirjautuminen ei onnistu. Näytetään virhe: "Älä jätä mitään kenttää tyhjäksi!"
ei toimi ilman salasanaa	kissa	""	Kirjautuminen ei onnistu. Näytetään virhe: "Älä jätä mitään kenttää tyhjäksi!"
ei toimi ilman käyttäjätunnusta ja salasanaa	""	""	Kirjautuminen ei onnistu. Näytetään virhe: "Älä jätä mitään kenttää tyhjäksi!"
uloskirjautuminen toimii	NA	NA	Sivun localStorage tyhjentyy ja näytetään viesti: "Uloskirjautuminen onnistui!"

Testijoukko4: Kurssin CRUD-operaatiot				
testin nimi	name	length	credits	odotettu lopputulos
<b>(lisäys):</b>				
toimii oikeilla tiedoilla	testi	1	5	Kurssi lisättiin onnistuneesti.
ei toimi ilman nimeä	""	2	5	Kurssia ei lisätty.
ei toimi jos kurssi kestää 0 periodia	Lyhyt kurssi	0	5	Kurssia ei lisätty.
ei toimi jos kurssista saa 0 opintopistettä	Perus kurssi	1	0	Kurssia ei lisätty.
ei toimi jos kurssista saa yli 100 opintopistettä	Perus kurssi	1	100	Kurssia ei lisätty.
lisää kurssin näytettävään listaan	Tietorakenteet ja Algoritmit	2	10	Kurssi lisättiin ja nyt listassa on kaksi kurssia.
<b>(poisto):</b>				
toimii jos kirjautunut käyttäjä omistaa kurssin	NA	NA	NA	Kurssi poistuu näytettävien kurssien listasta.
<b>(muokkaus):</b>	esiehto: käyttäjä on kirjautunut sisään			
nimeä voi muuttaa	Käyttäjä muuttaa kurssin nimeksi "Uusi kurssi"			Muutos tallentuu tietokantaan.
pituutta voi muuttaa	Käyttäjä muuttaa kurssin pituudeksi 1			-  -
opintopistemäärää voi	Käyttäjä muuttaa opintopistemääräksi 1			-  -

muuttaa		
haastavuutta voi muuttaa	Käyttäjä asettaa haastavuudeksi "Vaikea".	
tavoitetta voi muuttaa	Käyttäjä muuttaa tavoitteeseen 1	-  -