

Helpperin testaus

Maiju Airosmaa

Kalle Hiljanen

Outi Jussila

Eetu Mattila

Johanna Vehniäinen

Ohjelmiston testausdokumentaatio

Helsingin yliopisto

Heinäkuu 2016

Sisältö

1	Miten testaus on toteutettu	2
2	Testikattavuus	2
3	Liitteet	3

1 Miten testaus on toteutettu

Käytimme testauksessa Rspec:iä. Luokille on kirjoitettu yksikkötestejä ja toiminnallisuutta olemme testanneet selaintason käyttöliittymätesteillä Capybaralla. Testeissä käytettiin laajasti FactoryGirlilla tehtyjä fixtureja. Selaintason testit on tehty lähinnä mustalaatikko-toteutuksena, jossa sovelluksen backendin tuntemusta ei tarvita.

Testejä ajoimme jatkuvasti lokaalisti uusien ominaisuuksien lisäämisen jälkeen, koska pienetkin lisäykset ja muutokset vaikuttivat testeihin. Tarvittaessa korjasimme joko koodia tai testejä. Testit ajettiin myös Travis-CI-palvelimella aina, kun pushasimme muutoksia GitHub-repositorioon.

Projekti sisältää myös pienen määrän javascript-koodia sujuvoittamassa näkymien sisältämien lomakkeiden käyttöä ja virtaviivaistamassa niiden ulkoasua. Capybaran käyttöön yritettiin konfiguroida javascript-moottoria mutta käytännössä tämä osoittautui ongelmalliseksi. Javascript-koodi ei muokkaa käyttäjän syöttämää dataa, saatikka suorita validointeja eikä myöskään ole yhteydessä tietokantaan, minkä vuoksi ajankäyttöä konfigurointiin ei nähty tarpeelliseksi vaan tyydyttiin näiden ominaisuuksien perusteelliseen käsin testaamiseen.

2 Testikattavuus

Tavoitteenamme oli alusta lähtien pitää testien rivikattavuus korkealla, joten testejä on kirjoitettu sitä mukaa, kun uusia ominaisuuksia lisättiin. Alussa testejä kirjoitettiin huomattavasti laajemmin, mutta huomasimme pian, että koodin muuttuessa jouduimme jatkuvasti korjaamaan jo olemassa olevia testejä. Testejä tuli myös kirjoitettua päällekkäin eli yksikkö ja käyttöliittymätestit testasivat samoja asioita. Näiden huomioiden jälkeen vähensimme testien kirjoittamista pitäen rivikattavuuden kuitenkin korkealla. Myös sprinttien lyhyt kesto vaikutti testaamisen laajuuteen, koska aika tuntui jatkuvasti loppuvan kesken. Definition of Donin mukaan testikattavuuden tavoitteemme oli 90%. Projektin puolella välissä testikattavuus laahasi 70%:n tuntumassa, koska lisäsimme Scaffoldilla luokan Category, mutta tämän luo-

kan kontrolleria ei testattu, koska sillä ei ollut käyttöä sovelluksessa. Kategoriat luotiin seedillä ja niiden poistaminen täytyi olla mahdotonta. Jätimme kuitenkin varsinaisen sovelluksen ulkopuoliseksi ominaisuudeksi mahdollisuuden muokata kategorioita muutenkin kuin konsolin kautta. Myöhemmässä vaiheessa ignoroimme kyseessä olevan kontrollerin testikattavuuden ulkopuolelle .simplecov-tiedostossa. Samassa tiedostossa ignoressa ovat myös admin- ja info-kontrollerit. Admin- ominaisuutta emme ehtineet toteuttamaan ollenkaan ja info-kontrollerin metodeilla ei ole mitään sisältöä. Näiden tiedostojen ignoroinnin jälkeen testikattavuus pysyi jatkuvasti tavoitteessa tai hyvin lähellä sitä. Lopullinen testikattavuus oli noin 93%:a. Testikattavuuden tarkasteluun käytimme lokaalisti Simplecov:ia. Jos testit menivät palvelintasolla läpi Travisista, niin lähetettiin kattavuusraportti Coverallsiin.

3 Liitteet

Travis-CI: <https://travis-ci.org/xjoxjox/helpperi>

Coveralls: <https://coveralls.io/github/xjoxjox/helpperi>

Simplecovin kattavuusraportti:

All Files (92.96% covered at 46.45 hits/line)

23 files in total 554 relevant lines 515 lines covered and 39 lines missed

Search: <input type="text"/>						
File	% covered	Lines	Relevant Lines	Lines covered	Lines missed	Avg. Hits / Line
Q app/controllers/messages_controller.rb	72.41 %	50	29	21	8	2.0
Q app/controllers/omniauth_callbacks_controller.rb	73.33 %	31	15	11	4	1.7
Q app/controllers/conversations_controller.rb	76.92 %	22	13	10	3	1.1
Q app/models/message.rb	81.82 %	19	11	9	2	0.8
Q app/validators/file_size_validator.rb	86.49 %	68	37	32	5	1.8
Q app/models/user.rb	91.04 %	129	67	61	6	220.7
Q app/helpers/application_helper.rb	93.33 %	30	15	14	1	61.1
Q app/controllers/posts_controller.rb	94.66 %	219	131	124	7	6.6
Q app/controllers/users_controller.rb	95.74 %	86	47	45	2	4.1
Q app/validators/hetu_validator.rb	97.44 %	67	39	38	1	95.8
Q app/controllers/application_controller.rb	100.0 %	12	7	7	0	1.7
Q app/controllers/registrations_controller.rb	100.0 %	44	24	24	0	4.3
Q app/helpers/categories_helper.rb	100.0 %	2	1	1	0	1.0
Q app/helpers/devise_helper.rb	100.0 %	31	8	8	0	14.9
Q app/helpers/posts_helper.rb	100.0 %	2	1	1	0	1.0
Q app/models/candidate.rb	100.0 %	11	8	8	0	1.0
Q app/models/category.rb	100.0 %	5	4	4	0	10.8
Q app/models/conversation.rb	100.0 %	21	11	11	0	72.6
Q app/models/post.rb	100.0 %	85	54	54	0	63.4
Q app/models/post_category.rb	100.0 %	4	3	3	0	1.0
Q app/models/rating.rb	100.0 %	11	8	8	0	1.0
Q app/uploaders/image_uploader.rb	100.0 %	68	14	14	0	1.4
Q app/validators/date_in_future_validator.rb	100.0 %	11	7	7	0	75.0

Showing 1 to 23 of 23 entries