

Ryhmä: TeamH

Tekijä: Henri Haverinen, henri.s.haverinen@student.jyu.fi

Julkisuustaso: Julkinen

Päivämäärä: 17.10.2017

Versio: 0.1.0

Opintojakso: TIES546 Ohjelmistotestaus, Jyväskylän yliopisto, informaatioteknologian tiedekunta

Versiohallinta: <https://github.com/hhaverinen/TIES546>

DatabaseClient

Testaussuunitelma

Järjestelmätestaus

Toiminnallisuustestaus

Sisällysluettelo

1 Testattava ohjelmisto.....	2
2 Testattavat ominaisuudet.....	2
3 Tavoitteet ja päämäärät.....	3
4 Testausympäristö.....	3
5 Testauskerran suoritus.....	4
5.1 Raportointimalli / -esimerkki.....	5
6 Testitapaukset.....	5
6.1 Testitapaus 1: Tietokantayhteyksien muodostaminen.....	5
6.2 Testitapaus 2: SQL-kyselyjen tekeminen tietokantaan.....	6
6.3 Testitapaus 3: Tietokantayhteyksien tietojen tallentaminen ja lukeminen tiedostosta.....	6
6.4 Testitapaus 4: SQL-kyselyn tulosten vienti tiedostoon.....	7

1 Testattava ohjelmisto

Testattava ohjelmisto on Henri Haverisen kehittämä prototyyppi nimeltään “DatabaseClient”. Ohjelmisto tarjoaa käyttäjälle graafisen käyttöliittymän, jonka kautta käyttäjä voi suorittaa tietokantakyselyitä tietokantoihin. Ohjelmistossa on myös mahdollisuus tallentaa tietokantayhteyksiä helpomman yhteyden muodostamisen mahdollistamiseksi. Ohjelmisto tarjoaa lisäksi mm. tietokantakyselyjen viennin tiedostoon.

2 Testattavat ominaisuudet

Testettavat ominaisuudet ja perustelut niille ovat seuraavat:

- Tietokantayhteyksien muodostaminen on edellytys ohjelmiston käyttämiselle, ja siksi sen toiminnallisuutta on testattava.
- SQL-kyselyjen tekeminen tietokantaan on ohjelmiston pääasiallinen käyttötarkoitus, ja siten sen toiminnallisuus tulee testata.
- Tietokantayhteyksien tietojen tallentaminen ja lukeminen tiedostosta on olennainen ominaisuus, jonka vuoksi sen toiminnallisuuden testaaminen on tärkeää.
- SQL-kyselyn tulosten vienti tiedostoon on usein haluttu ja tarpeellinen ominaisuus, jonka vuoksi ominaisuutta tulee testata.

Testauksesta pois rajatut ominaisuudet ja perustelut rajauksille ovat seuraavat:

- Käyttöliittymä tuottaa myös käyttäjälle näkyvää lokia, jossa kerrotaan tarkemmin onnistumis- ja virheviesteistä. Lokin toimintaa ja viestien selkeyttä ei korkean työmäärän takia tällä testauskerralla testata.

- SQL-kyselyiden lukeminen ja tallentaminen tiedostoon ei ole kriittinen ja usein tarpeellinenkaan ominaisuus, joten se on rajattu pois testattavista ominaisuuksista.
- Käyttäjän syöttämien SQL-kyselyiden ylläpitäminen sekä niiden valitseminen alasvetovalikosta ei ole myöskään käyttöä estävä ominaisuus, joten sen toiminnallisuutta ei tällä testikerralla testata.
- Tietokantataulujen näyttäminen visuaalisesti tietokantayhteyden muodostumisen jälkeen ei ole tärkeä ominaisuus. Lisäksi kyseinen tieto on käyttäjän mahdollista hakea myös SQL-kyselyllä. Näistä syistä kyseistä ominaisuutta ei testata.
- Syötekentän toiminnallisuuksia (maalatun SQL-kyselyn suorittaminen, kursorin rivillä olevan SQL-kyselyn suorittaminen) ei testata, koska niiden uupuminen ei estä ohjelmiston käyttöä.

Tällä testauskerralla toteutetaan siis järjestelmätason toiminnallisuustestaus. Edellä on kerrottu tähän testauskertaan kuuluvat sekä pois rajatut testattavat ominaisuudet perusteluineen.

3 Tavoitteet ja päämäärät

Testauskerran tavoitteena on varmistaa, että testattavien ominaisuuksien seuraavat laatuominaisuudet täyttävät mahdollisimman hyvin:

- Ominaisuudet ovat *luotettavia*. Ominaisuudet sietävät ja pystyvät toipumaan mahdollisista virhetilanteista.
- Ominaisuudet ovat *toiminnallisia*. Ominaisuudet soveltuvat tarkoitettujen toimenpiteiden suorittamiseen ja antavat oikeita ja tarkkoja tuloksia.
- (optionaalinen) Ominaisuudet ovat *käytettäviä*. Ominaisuudet ovat helppokäyttöisiä ja intuitiivisia. Tässä testausprojektissa ja testauskerrassa ei suoriteta käytettävyydestä, mutta mikäli testikerran suorittaja havaitsee tai kokee joidenkin ominaisuuksien kaipaavan parannuksia käytettävyyteen, on näiden huomioiden esille tuominen suotavaa.

4 Testausympäristö

Alla on taulukoituna testausympäristön vaatimukset:

Ohjelmisto ja versio	DatabaseClient, versio 1.0
Käyttöjärjestelmä ja versio	Ubuntu 16.04 LTS 64bit
Java (Oracle tai OpenJDK)	Versio 1.8.0 tai uudempi, mutta ei kuitenkaan versiota 1.9.0 tai uudempaa
Apache Maven	Versio 3.3.9 tai uudempi
MySQL-tietokanta (voi sijaita samalla tai	Versio 5.7.0 tai uudempi

toisella laitteistolla kuin testattava ohjelmisto	
Git (optionaalinen)	Versio 2.7.0 tai uudempi
Mahdollisesti jokin tekstieditori	-

Asennusohjeet tarvittaville ohjelmistoille löytyvät seuraavista [www-osoitteista](#):

- DatabaseClient asennusohjeet:
<https://github.com/hhaverinen/TIEA306#käyttö>.
- Apache Maven asennusohjeet: <https://maven.apache.org/install.html>
- Java (OpenJDK) asennusohjeet: <http://openjdk.java.net/install/>
- MySQL asennusohjeet (versio 5.7 ->):
<https://dev.mysql.com/doc/refman/5.7/en/installing.html>

5 Testauskerran suoritus

Testauskerran suorittajalta vaaditaan perustietämystä erilaisten ohjelmistojen asentamisesta, komentorivityöskentelystä sekä SQL-kyselykielestä. Tietämys ohjelmistotestauksesta on eduksi.

Testauskerran yksilöintitiedoissa on mainittava:

- Testattava ohjelmisto ja sen versio
- Ohjelmiston toimintaan vaaditut ohjelmistot ja niiden versiot
- Testauskerralla käytettävä testaussuunnitelma ja sen versio
- Testausympäristön kuvaus
- Testauskerran suorittaja ja ajankohta

Testauskerran yhteenvetotiedoissa on mainittava:

- Suoritettujen testitapausten kokonaismäärä
- Suorittamattomien testitapausten kokonaismäärä
- Testitapausten eri johtopäätöksien kokonaismäärät
- Testitapauksista kirjattujen havaintojen kokonaismäärä
- Suositukset ohjelmistolle tehtävistä jälkitoimenpiteistä

Testitapausten raportoinnissa käytetään seuraavia kolmea johtopäätöstä: *ok*, *virhe* ja *huomio*. Johtopäätökset *ok* ja *virhe* ovat nimensämukaisesti tarkoitettu onnistuneen ja epäonnistuneen testitapausten johtopäätöksiksi. Johtopäätös *huomio* tarkoittaa onnistuneen testitapausten

johtopäätöstä, mutta testeuskerran suorittaja on havainnut joitain mainitsemisen arvoisia puutteita tai huomioita.

Virheet ja huomiot pyritään raportoimaan mahdollisimman ytimekkäästi sekä yksiselitteisesti. Mahdollisten virheiden raportoinnissa tulee myös raportoida mahdollisimman tarkasti virheeseen johtaneet tehdyt tai tekemättä jätetyt toimenpiteet ja syötteet sekä ohjelmiston mahdollisesti antama vaste virhetilanteessa.

5.1 Raportointimalli / -esimerkki

Alla on esiteltynä yksittäisen testitapauksen taulukkomallinen raportointimalli / -esimerkki, jota voi hyödyntää testauskerran suorituksen raportoinnissa.

Nimi Tietokantayhteyksien muodostaminen

Työnkulku / työnkulun poikkeama	Syötteet	Odotettu lopputulos	Johtopäätös	Huomiot
Normaali työnkulku	Tietokannan osoite: validi Käyttäjänimi: validi Salasana: validi	Yhteys muodostuu	Ok	
Syötetty salasana on väärä	Tietokannan osoite: validi Käyttäjänimi: validi Salasana: ei validi	Yhteys ei muodostu	Ok	Yhteys ei muodostu, mutta virheviesti voisi olla havainnollistavamp i.

6 Testitapaukset

Seuraavissa alaluvuissa on esiteltynä tämän testauskerran testitapaukset. Varsinaista testidataa ei ole annettu, koska testattava data riippuu käytettävien ohjelmistojen asetuksista (esimerkiksi salasanat). Työnkulku ja työnkulun poikkeamat on esiteltynä testitapauksissa. Työnkulun poikkeamat johtavat virhetilanteisiin, mutta ohjelmiston pitäisi pystyä toipumaan ja käsittelemään nämä tapaukset sekä mahdollisesti antamaan jonkinlainen vasta käyttäjälle.

6.1 Testitapaus 1: Tietokantayhteyksien muodostaminen

Nimi Tietokantayhteyden muodostaminen

Toimija Ohjelmiston käyttäjä

Lähtötila Toimija on käynnistänyt ohjelmiston, ohjelmiston perusnäkyminen on avoinna.

Työnkulku	<ol style="list-style-type: none"> 1. Toimija syöttää tietokannan osoitteen. 2. Toimija syöttää käyttäjänimen. 3. Toimija syöttää salasanan. 4. Toimija muodostaa tietokantayhteyden painamalla “Connect” -painiketta.
Lopputila	Toimija on perusnäkyssä, alhaalla olevaan tekstikenttään on tullut tieto onnistuneesta/epäonnistuneesta yhteyden muodostamisesta, “Active connections” -alasvetovalikossa on tieto mahdollisesti muodostetusta yhteydestä.

Työnkulun poikkeamat

- Toimija syöttää väärän / jättää tyhjäksi tietokannan osoitteen (askel 1)
- Toimija syöttää väärän / jättää tyhjäksi käyttäjänimen (askel 2)
- Toimija syöttää väärän / jättää tyhjäksi salasanan (askel 3)

6.2 Testitapaus 2: SQL-kyselyjen tekeminen tietokantaan

Nimi	SQL-kyselyjen tekeminen tietokantaan
Toimija	Ohjelmiston käyttäjä
Lähtötila	Toimija on käynnistänyt ohjelmiston ja muodostanut tietokantayhteyden, ohjelmiston perusnäky on avoinna.
Työnkulku	<ol style="list-style-type: none"> 1. Toimija syöttää SQL-kyselyn tekstikenttään. 2. Toimija suorittaa SQL-kyselyn painamalla “Run query” -painiketta.
Lopputila	Toimija on perusnäkyssä, alhaalla olevaan tekstikenttään on tullut tieto onnistuneesta/epäonnistuneesta SQL-kyselystä, SQL-kyselyn tulokset näkyvät taulukossa sille varatussa tilassa.

Työnkulun poikkeamat

- Toimija syöttää ei validin SQL-kyselyn / jättää tyhjäksi SQL-kyselyn tekstikentän

6.3 Testitapaus 3: Tietokantayhteyksien tietojen tallentaminen ja lukeminen tiedostosta

Nimi	Tietokantayhteyksien tietojen tallentaminen ja lukeminen tiedostoon
Toimija	Ohjelmiston käyttäjä
Lähtötila	Toimija on käynnistänyt ohjelmiston ja avannut “Manage Aliases” -näkyvän.
Työnkulku	<ol style="list-style-type: none"> 1. Toimija syöttää pyydetyt tiedot niille varattuihin kenttiin 2. Toimija tallentaa tiedot painamalla “Add” -painiketta 3. Toimija sulkee ikkunan sekä koko ohjelmiston 4. Toimija uudelleenkäynnistää ohjelmiston

5. Toimija avaa "Manage Aliases" -näkyvän

Lopputila

Toimija on "Manage Aliases" -näkyvässä. Ikkunan ylimmässä alavetovalikossa on mahdollista valita tallennettu yhteys. Yhteyden valitsemalla yhteden tiedot tulevat niille varattuihin kenttiin.

Työnkulun poikkeamat

- Toimija ei syötä kaikkia pyydettyjä tietoja kenttiin (askel 1)
- Toimija sulkee ohjelman väärin, esimerkiksi tappamalla prosessin (askel 1)

6.4 Testitapaus 4: SQL-kyselyn tulosten vienti tiedostoon

Nimi

SQL-kyselyn tulosten vienti tiedostoon

Toimija

Ohjelmiston käyttäjä

Lähtötila

Toimija on käynnistänyt ohjelmiston ja muodostanut tietokantayhteyden, ohjelmiston perusnäkyvä on avoinna.

Työnkulku

1. Toimija syöttää SQL-kyselyn tekstikenttään.
2. Toimija aloittaa tiedostoon tallentamisen painamalla "Export query results to file" -painiketta.
3. Toimija valitsee kohdehakemiston ja antaa tiedostonimen avautuneessa ikkunassa ja painaa lopuksi "Save" -painiketta.

Lopputila

Toimija on perusnäkyvässä, alhaalla olevaan tekstikenttään on tullut tieto onnistuneesta/epäonnistuneesta tulosten tallentamisesta, SQL-kyselyn tulokset ovat tallennettuna käyttäjän antamaan tiedostoon pilkku-eroteltuna listana.

Työnkulun poikkeamat

- Toimija ei syötä SQL-kyselyä tekstikenttään (askel 1)
- Toimija syöttää epävalidin SQL-kyselyn tekstikenttään (askel 1)
- Toimija ei anna tiedostonimeä tallennettavalle tiedostolle (askel 3)