

Ryhmä: TeamH

Tekijä: Henri Haverinen, [henri.s.haverinen@student.jyu.fi](mailto:henri.s.haverinen@student.jyu.fi)

Julkisuustaso: Julkinen

Päivämäärä: 10.11.2017

Versio: 1.3.0

Opintojakso: TIES546 Ohjelmistotestaus, Jyväskylän yliopisto, informaatioteknologian tiedekunta

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

# **DatabaseClient**

## **Testaussuunnitelma**

### **Järjestelmätestaus**

### **Toiminnallisuustestaus**

# Sisällysluettelo

1 Testauksen kohde.....	2
1.1 Testattava ohjelmisto.....	2
1.2 Testattavat ominaisuudet.....	2
1.3 Testausympäristö.....	3
2 Testausprojekti.....	4
2.1 Tavoitteet ja päämäärät.....	4
2.2 Testaustaso ja -tyyppi.....	4
3 Testauksen dokumentointi.....	4
3.1 Testauskerran yksilöinti- ja yhteenvetotiedot.....	5
3.2 Raportointiesimerkki.....	5
4 Testitapaukset.....	6
4.1 Testitapaus 1: Tietokantayhteyden muodostaminen.....	7
4.2 Testitapaus 2: SQL-kyselyjen tekeminen tietokantaan.....	7
4.3 Testitapaus 3: Tietokantayhteyksien tietojen tallentaminen ja lukeminen tiedostosta.....	8
4.4 Testitapaus 4: SQL-kyselyn tulosten vienti tiedostoon.....	8

## 1 Testauksen kohde

Tässä luvussa esitellään testattava ohjelmisto, ominaisuudet joita ohjelmistosta testataan sekä testausympäristö.

### 1.1 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.

### 1.2 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 testauksella testata. **Huom.** testusprojektin suunnitelmasta poiketen tätä ominaisuutta ei edellä mainitun syyn takia 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 testauksella 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.

## 1.3 Testausympäristö

Alla on taulukoituna testusympäristön vaatimukset:

Ohjelmisto / työkalu	Versio, mahdollisesti tarkenteet
Testattava ohjelmisto	DatabaseClient, versio 1.0
Käyttöjärjestelmä	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 toisella laitteistolla kuin testattava ohjelmisto)	Versio 5.7.0 tai uudempi
Git (optionaalinen)	Versio 2.7.0 tai uudempi
Mahdollisesti jokin tekstieditori	-

Asennusohjeet tarvittaville ohjelmistoille löytyvät seuraavista WWW-osoitteista:

- DatabaseClient: <https://github.com/hhaverinen/TIEA306#käyttö>.

- Apache Maven: <https://maven.apache.org/install.html>
- Java (OpenJDK): <http://openjdk.java.net/install/>
- MySQL (versio 5.7 ja myöhemmät):  
<https://dev.mysql.com/doc/refman/5.7/en/installing.html>

## 2 Testausprojekti

Tässä luvussa on esiteltynä testausprojektin tavoitteet ja päämäärät, sekä mitä testaustasoa ja -tyyppejä projektissa hyödynnetään.

### 2.1 Tavoitteet ja päämäärät

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

- Ominaisuudet ovat **virhesietoisia**. Ominaisuudet sietävät ja pystyvät toipumaan mahdollisista virhetilanteista.
- Ominaisuudet ovat **toiminnallisia**. Ominaisuudet soveltuvat tarkoitettujen toimenpiteiden suorittamiseen ja antavat oikeita ja tarkkoja tuloksia.

Tässä testausprojektissa ja testauskerrassa ei suoriteta käytettävyytestausta, mutta mikäli testauskerran suorittaja havaitsee tai kokee joidenkin ominaisuuksien kaipaavan parannuksia käytettävyyteen, on näiden huomioiden esille tuominen suotavaa. Huomioiden tekeminen ja kirjaaminen on kuitenkin täysin optionaalista.

### 2.2 Testaustaso ja -tyyppi

Tällä testauskerralla suoritetaan järjestelmätason toiminnallisuustestaus. Luvussa 1.2 on kuvattu tähän testauskertaan kuuluvat sekä pois rajatut testattavat ominaisuudet perusteluineen.

Testauksessa hyödynnetään mustalaatikkotestausta, testauskerran suorittajan ei siis tarvitse perehtyä eikä tutkia ohjelmiston lähdekoodia. Testauskerran suorittajaa suositellaan käyttävän ekvivalenssiluokkia testisyötteitä laatiessaan, mikäli kokee tämän hyödylliseksi.

## 3 Testauksen dokumentointi

Tässä luvussa on kuvattu kuinka testauskerta suoritetaan ja sen tulokset raportoidaan. Luvussa 4 on esitelty tämän testauskerran testitapaukset.

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

### 3.1 Testauskerran yksilöinti- ja yhteenvetotiedot

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 nimensä mukaisesti 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 riittävän tarkasti virheeseen johtaneet tehdyt tai tekemättä jätetyt toimenpiteet ja syötteet sekä ohjelmiston mahdollisesti antama vaste virhetilanteessa.

### 3.2 Raportointiesimerkki

Alla on esiteltyä yksittäisen testitapausten taulukkomallinen raportointiesimerkki, jota voi hyödyntää testauskerran suorituksen raportoinnissa. Testitapauksissa käytetyt syötteet ja vasteet on kirjattava raporttiin, koska testitapauksissa ei anneta testidataa vaan se on testauskerran suorittajan päätettävissä. Testitapausten yksilöivä numero vastaa luvun 4 alaluvuissa esiteltyjen testitapausten työnkulun poikkeamia. Tässä esimerkissä lukua 0 on käytetty normaalin työnkulun yksilöimiseen.

Testitapaus 1: Tietokantayhteyksien muodostaminen						
Yksilöivä numero	Työnkulku tai sen poikkeama	Syötteet	Vasteet	Odotettu vaste	Johtopäätös	Huomiot
1.0	Normaali työnkulku.	Tietokannan osoite = jdbc:mysql://localhost/testi  Käyttäjä = root  Salasana = passw0rd	Lokiin tulee tieto "yhteys muodostettu".	Yhteys muodostuu.	Ok	
1.e	Syötetty salasana on väärä.	Tietokannan osoite = jdbc:mysql://localhost/testi  Käyttäjä = root  Salasana = väärä	Lokiin tulee tieto "yhteyttä ei voitu muodostaa".	Yhteys ei muodostu.	<b>Huomio</b>	Yhteys ei muodostu, mutta virheviesti voisi olla havainnollisempi.

## 4 Testitapaukset

Tässä luvussa kuvataan testauskerran testitapaukset. Varsinaista testidataa ei ole annettu, koska testattava data riippuu käytettävien ohjelmistojen asetuksista (esimerkiksi salasanat). Testauskerran suorittajalle halutaan myös antaa vapauttaa testattavien syötteiden valitsemisessa, jotta testauskerran suorittamisesta ei tule liian mekaanista.

Työnkulku ja työnkulun poikkeamat ovat esiteltynä testitapauksissa. Työnkulun poikkeamat johtavat virhetilanteisiin, mutta ohjelmiston pitäisi pystyä toipumaan ja käsittelemään nämä tapaukset sekä mahdollisesti antamaan jonkinlainen vasta käyttäjälle. Työnkulun poikkeamat ovat numeroitu siten, että ensimmäinen numero vastaa työnkulussa olevaa askelta, ja perässä oleva kirjain erottelee samaan askeleeseen kuuluvat poikkeustapaukset. Kaikissa testitapauksissa toimijana toimii ohjelmiston käyttäjä.

Testitapauksien esitysmuoto mukailee Jyväskylän yliopiston informaatioteknologian tiedekunnan TIES546 Ohjelmistotestaus -kurssin Jukka-Pekka Santasen luentokalvoilla esitettyä esimerkkiä käyttötapauksesta johdetusta testitapauksesta (luentokalvo 30/72)

<https://koppa.jyu.fi/kurssit/216254/luento/testaustyyppien-tekniikoita-ja-testitapauksia>).

## 4.1 Testitapaus 1: Tietokantayhteyden muodostaminen

<b>Nimi</b>	Tietokantayhteyden muodostaminen
<b>Lähtötila</b>	Toimija on käynnistänyt ohjelmiston, ohjelmiston perusnäkyminen on avoinna.
<b>Työnkulku</b>	<ol style="list-style-type: none"><li>1. Toimija syöttää tietokannan osoitteen.</li><li>2. Toimija syöttää käyttäjänimen.</li><li>3. Toimija syöttää salasanan.</li><li>4. Toimija muodostaa tietokantayhteyden painamalla <i>Connect</i> -painiketta.</li></ol>
<b>Lopputila</b>	Toimija on perusnäkyssä, alhaalla olevaan tekstikenttään on tullut tieto onnistuneesta/epäonnistuneesta yhteyden muodostamisesta, <i>Active connections</i> -alasvetovalikossa on tieto mahdollisesti muodostetusta yhteydestä.

**Työnkulun poikkeamat** tulee testata seuraavilla testitapauksilla:

- 1.a Toimija syöttää väärän tietokannan osoitteen.
- 1.b Toimija jättää tyhjäksi tietokannan osoitteen.
- 1.c Toimija syöttää väärän käyttäjänimen.
- 1.d Toimija jättää tyhjäksi käyttäjänimen.
- 1.e Toimija syöttää väärän salasanan.
- 1.f Toimija jättää tyhjäksi salasanan.

## 4.2 Testitapaus 2: SQL-kyselyjen tekeminen tietokantaan

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

**Työnkulun poikkeamat** tulee testata seuraavilla testitapauksilla:

- 2.a Toimija syöttää ei validin SQL-kyselyn.
- 2.b Toimija jättää tyhjäksi SQL-kyselyn tekstikentän.

### 4.3 Testitapaus 3: Tietokantayhteyksien tietojen tallentaminen ja lukeminen tiedostosta

<b>Nimi</b>	Tietokantayhteyksien tietojen tallentaminen ja lukeminen tiedostoon
<b>Lähtötila</b>	Toimija on käynnistänyt ohjelmiston ja avannut <i>Manage Aliases</i> -näköymän.
<b>Työnkulku</b>	<ol style="list-style-type: none"><li>1. Toimija syöttää pyydettyt tiedot niille varattuihin kenttiin.</li><li>2. Toimija tallentaa tiedot painamalla <i>Add</i> -painiketta.</li><li>3. Toimija sulkee ikkunan sekä koko ohjelmiston.</li><li>4. Toimija uudelleenkäynnistää ohjelmiston.</li><li>5. Toimija avaa <i>Manage Aliases</i> -näköymän.</li></ol>
<b>Lopputila</b>	Toimija on <i>Manage Aliases</i> -näköymässä. Ikkunan ylimmässä alavetovalikossa on mahdollista valita tallennettu yhteys. Yhteyden valitsemalla yhteden tiedot tulevat niille varattuihin kenttiin.

**Työnkulun poikkeamat** tulee testata seuraavilla testitapauksilla:

- 3.a Toimija ei syötä kaikkia pyydettyjä tietoja kenttiin.
- 3.b Toimija sulkee ohjelman väärin, esimerkiksi tappamalla prosessin.

### 4.4 Testitapaus 4: SQL-kyselyn tulosten vienti tiedostoon

<b>Nimi</b>	SQL-kyselyn tulosten vienti tiedostoon
<b>Lähtötila</b>	Toimija on käynnistänyt ohjelmiston ja muodostanut tietokantayhteyden, ohjelmiston perusnäköymä on avoinna.
<b>Työnkulku</b>	<ol style="list-style-type: none"><li>1. Toimija syöttää SQL-kyselyn tekstikenttään.</li><li>2. Toimija aloittaa tiedostoon tallentamisen painamalla <i>Export query results to file</i> -painiketta (<b>huom.</b> SQL-kysely suoritetaan samalla painikkeen painalluksella).</li><li>3. Toimija valitsee kohdehakemiston ja antaa tiedostonimen avautuneessa ikkunassa ja painaa lopuksi <i>Save</i> -painiketta.</li></ol>
<b>Lopputila</b>	Toimija on perusnäköymässä, alhaalla olevaan tekstikenttään on tullut tieto onnistuneesta/epäonnistuneesta tulosten tallentamisesta, SQL-kyselyn tulokset ovat tallennettuna käyttäjän antamaan tiedostoon pilkku-eroteltuna listana.

**Työnkulun poikkeamat** tulee testata seuraavilla testitapauksilla:

- 4.a Toimija ei syötä SQL-kyselyä tekstikenttään.
- 4.b Toimija syöttää epävalidin SQL-kyselyn tekstikenttään.
- 4.c Toimija ei anna tiedostonimeä tallennettavalle tiedostolle.