

Tahiti — tähtitieteellisten havaintojen tietokanta

Tomi Hänninen
Juho Muhonen
Ismo Puustinen
Kai Pääsky
Pekka Simola
Nuutti Varis

Helsinki 12.5.2003

Testausdokumentti

HELSINGIN YLIOPISTO

Tietojenkäsittelytieteen laitos

Tiedekunta/Osasto — Fakultet/Sektion — Faculty		Laitos — Institution — Department	
Matemaattis-luonnontieteellinen		Tietojenkäsittelytieteen laitos	
Tekijä — Författare — Author			
Tomi Hänninen Juho Muhonen Ismo Puustinen Kai Pääsky Pekka Simola Nuutti Varis			
Työn nimi — Arbetets titel — Title			
Tahiti — tähtitieteellisten havaintojen tietokanta			
Oppiaine — Läroämne — Subject			
Tietojenkäsittelytiede			
Työn laji — Arbetets art — Level		Aika — Datum — Month and year	Sivumäärä — Sidoantal — Number of pages
Testausdokumentti		12.5.2003	164 sivua
Tiivistelmä — Referat — Abstract			
<p>Testit on nyt käyty läpi.</p> <p>Uusin versio tästä dokumentista tulee aikanaan saataville Tahiti-ryhmän kotisivuille osoitteeseen http://www.cs.helsinki.fi/group/tahiti/ .</p> <p>Versiohistoria:</p> <ol style="list-style-type: none"> 1. Versio 1.0 (katselmoitava versio) 2. Versio 1.1 (korjattu versio) 			
Avainsanat — Nyckelord — Keywords			
Tahiti, fotometria, Standard Asteroid Photometric Catalogue			
Säilytyspaikka — Förvaringsställe — Where deposited			
Muita tietoja — övriga uppgifter — Additional information			
Versio 1.1			

Sisältö

1 Johdanto	1
2 Yleistä testauksesta	1
3 Moduulitestaus	3
3.1 AsteroidDataBean	4
3.2 AsteroidQueryBean	6
3.3 CustomElementSubmitBean	10
3.4 EventHistoryDataBean	11
3.5 LightcurveDataBean	11
3.6 LightcurveQueryBean	15
3.7 LoginBean	17
3.8 LoginDataBean	17
3.9 SiteConfigurationSubmitDataBean	20
3.10 UserSearchDataBean	22
3.11 AtlasImporter	22
3.12 DBControl	22
3.12.1 Julkiset metodit	22
3.12.2 Yksityiset metodit	27
3.13 LightcurveHandler	27
3.13.1 Yksityiset metodit	27
3.13.2 Normaalit metodit	33
3.14 LogHandler	48
3.15 MailHandler	51
3.15.1 Julkiset metodit	51
3.15.2 Yksityiset metodit	53
3.15.3 Aliluokka MailerThread	54
3.16 SystemHandler	54
3.16.1 Kirjastometodit	54
3.16.2 Julkiset metodit	56
3.17 TahitiLibrary	58
3.18 TrajectoryHandler	61
3.19 UserHandler	74

3.19.1	Yksityiset metodit	74
3.19.2	Normaalit metodit	77
4	Integraatiotestaus	89
4.1	getAsteroids	89
4.2	getLightcurves	100
4.3	getTrajectory	105
4.4	makeDataFile	106
4.5	makeRawData	111
4.6	registrationRequest	112
4.7	login	117
4.8	insertLightcurve	118
4.9	insertLightcurveAtlas	132
4.10	changePassword	133
4.11	getUsers	134
4.12	setUserData	135
4.13	resetPassword	136
4.14	changeLightcurve	137
4.15	logoff	146
4.16	removeUser	147
4.17	getSettings	147
4.18	changeSettings	148
4.19	getLogEntries	150
4.20	deleteLightcurve	153
4.21	restoreLightcurve	154
4.22	addTrajectory	155
4.23	spamRecommender	156
4.24	spamUser	157
5	Validointitestaus	158
5.1	Valokäyrien haku	158
5.2	Valokäyrien syöttäminen järjestelmään	158
5.3	Valokäyrien tulostaminen	159
5.4	Valokäyrien muuttaminen	159
5.5	Omien tietojen muuttaminen	159

5.6	Salasanan nollaaminen	159
5.7	Käyttäjien tietojen hakeminen	160
5.8	Käyttäjien lisääminen ja poistaminen järjestelmästä	160
5.9	Asetusten muuttaminen	161
5.10	Järjestelmän tapahtumien selaaminen	161
5.11	Kirjautuminen	162
5.12	Käyttäjätietojen muuttaminen	162
5.13	Rataelementtien lisääminen	162
5.14	Atlas-syötin	162
5.15	Rekisteröintipyyntöä täyttämisen	163
5.16	Käyttäjien hyväksyminen tai hylkääminen	163
5.17	Valokäyrän poistaminen	163

Liitteet

1 Atlasimporter 1

2 Atlasimporter 2

1 Johdanto

Tämä on Tahiti-projektin tuottaman ohjelmiston testaamaus dokumentti. Dokumentti koostuu testaussuunnitelmasta, jota on päivitetty sisältämään myös tehtyjen testien tulokset sekä käytetyt aineistot.

Toisessa luvussa kuvataan yleinen testausstrategia, sekä perustellaan joitain tehtyjä päätöksiä. Luvussa kolme käsitellään moduulitestausta, luvussa neljä integraatiotestausta ja viimeisessä, viidennessä, luvussa validointitestausta.

2 Yleistä testauksesta

Testauksen tavoitteena on saada poistettua tuotettavasta ohjelmasta mahdollisimman monta virheitä, jotta toimitettava ohjelma olisi mahdollisimman täydellinen. Testaukselle on varattu projektisuunnitelman mukaan aikaa 3 viikkoa, jona aikana projektiryhmällä on myös muita tehtäviä, kuten esimerkiksi käyttöohjeen tekoa.

Testauksesta pyritään tekemään mahdollisimman kattava käytössä olevien resurssien valossa. Moduulitestauksessa jokainen ohjelmiston metodi testataan lausekattavasti tietyin varauksin. Näihin poikkeamiin lausekattavuudesta palataan moduulitestauksen yhteydessä. Kaikki tehdyt testit dokumentoidaan vähintään siten, että jokainen testauksessa käytetty testiluokka on saatavissa.

Toisessa testivaiheessa, integraatiotestauksessa, testataan järjestelmän eri komponenttien yhteistoimintaa. Integraatiotestausta suoritetaan vain Tahiti-Apille, ja Apin toiminnot toteuttaville käsitelijäluokille. Integraatiotestauksen päätavoitteena onkin varmistaa, että Tahiti-Apin eri luokat ja metodit toimivat oikein yhteen. Samalla testausaineistoa pyritään jakamaan ekvivalenssiluokkiin, jotta mahdollisimman moni virhe huomattaisiin. Testiaineiston jakoon palataan integraatiotestauksen yhteydessä.

Testauksen viimeisessä, kolmannessa, vaiheessa tarkistetaan että järjestelmä toimii määrittelyn mukaisesti. Tätä viimeistä testausvaihetta kutsutaan validointitestaukseksi. Validointitestauksen yhteydessä testataan siis järjestelmää kokonaisuutena, ja tarkistetaan että järjestelmä toimii oletetulla tavalla. Myös järjestelmätestauksessa testausaineisto jaetaan ekvivalenssiluokkiin mahdollisten virhetilanteiden löytämiseksi.

Testauksesta siis ensimmäinen vaihe, moduulitestaus suoritetaan ns. White-box-testauksena, jossa testiaineisto muodostetaan ohjelmakoodin mukaisesti. Kaksi seuraava vaihetta, integraatiotestaus sekä järjestelmätestaus suoritetaan nk. Black-box testauksena, jossa järjestelmälle annetaan erilaisia syötteitä, ja tarkastellaan saatuja tuloksia. Black-box -testauksessa testitapausten suunnittelu perustuu järjestelmän oletettuun toimintaan. Käyttöliittymästä on testattu pelkästään niisanotut Bean-luokat, joiden avulla käyttöliittymä siirtää tietoa apin ja itsensä välillä.

Moduulitestaus toteutetaan itsenäisesti luokkien toteuttajien toimesta, eikä moduulitestausta varten pystytetä erillistä testausympäristöä. Myös integraatiotestaus suoritetaan itsenäisesti, joten siihen ei erillistä testausympäristöä tarvita. Järjestelmätestaus sen sijaan suoritetaan yhteisessä testausympäristössä, joka ryhmälle perustetaan koneelle db.cs.helsinki.fi. Testauksessa käytetään Java versiota 1.4.1_01. TahitiLibraryyn testeissä olevat pienet heitot johtuvat siitä, että järjestel-

mään asetetut iteraativakiot eivät ole täsmälleen samat kuin millä vertailuaineisto on tuotettu.

Modulitestauksessa käytettiin seuraavia apuluokkia, jotka löytyvät testatun luokan pakkaushakemiston alahakemistosta test. Esimerkiksi pakkauksen `fi.helsinki.cs.group.tahiti.api` modulitestauksessa käytetyt luokat löytyvät CVS:än hakemistosta `src/fi/helsinki/cs/group/tahiti/api/test`.

- `LogTester` – luokkaa käytettiin `LogHandler`-luokan testaamiseen.
- `SystemTester` – luokkaa käytettiin `SystemHandler`-luokan testaamiseen.
- `UserHandlerTester` –luokkaa käytettiin `UserHandler`-luokan testaamiseen.
- `LcHandlerTester` –luokkaa käytettiin `LightcurveHandler`-luokan testaamiseen.
- `LcHandlerPrivateTester` –luokkaa käytettiin `LightcurveHandler`in yksityisen `parseDataPoints`-metodin testaamiseen, joka käyttää `HandlerDataPoint`-sisäluokkaa.
- Muunneltu `DBCControl`-luokka, jonka `update`-metodit eivät tee muutoksia tietokantaan saakka, vaan tulostavat annetut SQL-lauseet ruudulle.
- `DBCControlT` –luokkaa käytettiin `DBCControl`in testaamiseen.
- `MailHandlerT` –luokkaa käytettiin `MailHandler`in testaamiseen
- `AtlasImporterT` –luokkaa käytettiin `AtlasImporter`in testaamiseen.
- Käyttöliittymän bean-luokat ovat testattu erikseen. Jokaisen käyttöliittymän beanin testiluokka löytyy paketista `fi.helsinki.cs.group.tahiti.ui.Test`, jossa jokaisella bean-luokalla on testiluokka nimettynä tyyliin `BeanLuokanNimiTest`.

Integraatiotestausta varten luotiin pakkaus `fi.helsinki.cs.group.tahiti.integration`, johon integraatiotestauksen toteuttavat luokat laitettiin. Testauksessa käytettiin seuraavia luokkia:

- `Tester.java`
 1. `registrationRequest`
 2. `logoff`
 3. `getTrajectory`
 4. `changeLightcurve`
 5. `getLogEntries`
 6. `spamRecommender`
- `JuhonTester.java`
 1. `getAsteroids`
 2. `InsertLightCurve`
 3. `makeDataFile`
 4. `changePassword`

- 5. getSettings
- 6. restoreLightcurve
- KainTester.java
 - 1. resetPassword
 - 2. getLightcurves
 - 3. insetLightcurveAtlas
 - 4. changeSettings
 - 5. makeRawData
 - 6. addTrajectory

Erillisiä integraatiotestiluokkia:

- SetUserDataTester
- GetUsersTester
- LoginTester
- RemoveUserTester
- DeleteLightcurveTester
- SpamUserTester

Lisäksi integraatiotestauksessa käytettiin lievästi muunneltuja käyttöliittymän Bean-luokkia tiedon kuljettamiseen TahitiApille. Luokkiin oli lisätty `print()`-metodi, jolla kontainerin sisältö saadaan helposti ruudulle.

Järjestelmätestausta ei suoritettu kiirrellisen aikautaulun vuoksi mihinkään tiettyyn versioon, vaan testaus suoritettiin jatkuvasti ohjelmiston uusimpaan versioon, joka cvs-puusta saatiin. Tällä tavoin korjaukset virheisiin saatiin myös testattua mahdollisimman nopeasti, ja järjestelmän oikea toiminta pystyttiin validoimaan.

3 Moduulitestausta

Moduulitestausta suoritettiin järjestelmässä ns. ”White-Box”-testauksena, lausekattavasti. Testauksessa ei kuitenkaan pyritty täydelliseen lausekattavuuteen, vaan lausekattavuutta pidetään vain eräänlaisena ohjenuorana. Käytännössä tämä tarkoittaa sitä, että metodit testataan lausekattavasti, mutta muiden metodien tuottamiin virheisiin ei tarvitse varautua.

Rajallisten resurssien vuoksi moduulitestausta on jokaisen luokan kehittäjän vastuulla, sillä metodien rakenne on oletettavasti toteuttajille kaikista selvän. Muussa tapauksessa testaajat olisivat joutuneet käyttämään liikaa aikaa testattavaan luokkaan tutustumiseen.

3.1 AsteroidDataBean

AsteroidDataBean-luokan metodit testataan erillisellä testiluokalla. Beanit voidaan testata tavallisin luokkin, koska ne ovat tavallisia javan luokkia. Luokan sisältö talletetaan muuttujaan, jonka arvo voi olla null. Tällöin luokalla ei ole sisältöä.

Koko luokan testaus on tehty JUnit-luokassa: *fi.helsinki.cs.group.tahiti.ui.Test.AsteroidDataBeanTest*.

Koska luokan metodit testataan JUnit-kehikkoa käyttäen, testattavan metodin oletettavia arvoja ei ole lueteltu. JUnit-kehikon assert-metodeilla voidaan kyseisen metodin paluuarvoa verrata sopivaan arvoon turvallisesti. Virhetilanteen sattuessa testiluokka heittää käyttäjälle AssertionExceptionin ja keskeyttää luokan ajon. Tästä johtuen testituloksia ei ole metodien kohdalla listattu.

- *returnData()* Perustapauksena voidaan pitää AsteroidDataBean ilmentymää, jolla on joitain asteroideja. Tällöin metodi palauttaa kaikki kyseisen bean-luokan asteroidit taulukossa. Jos beanilla ei ole mitään sisältöä, metodi palauttaa null-viitteen.

Perustapaukselle luodaan testiaineistoksi viisi ApiAsteroid-luokan ilmentymää, joiden tiedot ovat seuraavat:

- Asteroidin nimi: "Asteroid 1-5"
- Asteroidin tunnus: "ADesignation 1-5"
- Asteroidin numero: 1-5
- Asteroidin id: 0-4

Tyhjän AsteroidDataBeanin testausta varten luodaan uusi ilmentymä kyseisestä beanista konstruktorissa.

- *returnData(long asteroidId)* Perustapauksena voidaan olettaa, että bean-luokan sisällöstä löytyy kysieisellä asteroidId-parametrilla ilmentymä, jolloin kyseinen ilmentymä palautetaan. Metodin perustapauksen testauksessa käytetään *returnData()*-metodin perustapauksen testausaineistoa.

Tämän lisäksi on testattava tapaukset:

1. *AsteroidId:tä ei löydy beanista* Tällöin metodi palauttaa null viitteen. Testaustilanteen voi järjestää helposti luomalla asteroidId:n, jota järjestelmästä ei löydy, esimerkiksi negatiivisen arvon.

Testiaineistona käytetään kahta erillistä arvoa id-numeroksi: -1 sekä Javan vakiota Integer.MAX_VALUE.

2. *Bean-instanssilla ei ole sisältöä* Tällöin metodi palauttaa null viitteen. Tapaus voidaan järjestää luomalla uusi ilmentymä AsteroidDataBean-luokasta ja kutsumalla tätä metodia jollakin parametrilla.

Testiaineistona käytetään tyhjää AsteroidDataBean-ilmentymää, joka luodaan testiluokan konstruktorissa.

- *getAsteroidName(String aid)* Metodin on tarkoitus palauttaa parametrin aid ilmoittaman asteroidin määritelty nimi. Nimi voi olla kolmea eri muotoa: <numero> <nimi> (<tunnus>),

<numero> <nimi>, (<tunnus>). Parametrissa aid joudutaan hakemaan asteroidin numero käyttämällä Long.parseLong()-metodia.

Perustapauksena on epätyhjä ilmentymä AsteroidDataBean-luokasta sekä parametrin aid avulla sisällöstä voidaan löytää asteroidi, jolla on sekä nimi että tunnus. Tällöin metodi palauttaa yllä olevista vaihtoehdoista ensimmäisen.

Testiaineistona perustapauksessa on konstruktorissa luotu AsteroidDataBean, jolla on viisi ApiAsteroid-ilmentymää. Testissä käydään läpi koko testiaineisto, ts. asteroidien id numerot nollassa neljään. Testiaineiston kuvaus on ilmoitettu returnData()-metodin yhteydessä.

Lisäksi on testattava seuraavat tapaukset:

1. *Parametri aid on null-viite.* Parametrin aid ollessa null-viite, metodi palauttaa käyttäjälle tyhjän merkkijonon "". Tilanne voidaan helposti luoda kutsumalla metodia null-viitteellä.

Testiaineistona on perustapauksen aineisto.

2. *Bean-instanssilla ei ole sisältöä.* Kuten yllä, jos bean-instanssilla ei ole sisältöä, metodi palauttaa käyttäjälle tyhjän merkkijonon "". Tilanne voidaan luoda tekemällä uusi instanssi AsteroidDataBeanista alustamatta sitä millään tavoin.

Testiaineistona on konstruktorissa luotu tyhjä AsteroidDataBean-instanssi.

3. *Parametrin aid avulla löytyy asteroidi, jolla on vain nimi.* Metodin tulisi palauttaa merkkijono, joka on muotoa <numero> <nimi>. Tilanne voidaan luoda tekemällä yksi testiasteroidi, jolle ei aseteta tunnusta.

Metodia varten luodaan uusi AsteroidDataBean-ilmentymä, jolle annetaan parametrina yksielementtinen ApiAsteroid-taulukko, jonka ainoan instanssin tiedot ovat muotoa:

- Asteroidin nimi: "Asteroid 1"
- Asteroidin numero: 1
- Asteroidin tunnus: tyhjä
- Asteroidin id: 0

4. *Parametrin aid avulla löytyy asteroidi, jolla on vain tunnus.* Metodin tulisi palauttaa merkkijono, joka on muotoa (<tunnus>). Tilanne voidaan järjestää kuten yllä, mutta nimen sijaan asetetaan pelkkä tunnus.

Metodia varten luodaan uusi AsteroidDataBean-ilmentymä, jolle annetaan parametrina yksielementtinen ApiAsteroid-taulukko, jonka ainoan instanssin tiedot ovat muotoa:

- Asteroidin nimi: tyhjä
- Asteroidin numero: tyhjä
- Asteroidin tunnus: "ADesignation 1"
- Asteroidin id: 0

5. *Annettu parametri aid ei ole muunnettavissa kokonaisluvuksi.* Tällöin metodi palauttaa tyhjän merkkijonon "". Tilanne muodostetaan kutsumalla metodia vaikkapa parametrillä "kissa".

Testiaineistona käytetään perustapauksen aineistoa.

6. *Annetulla parametrilla aid ei löydetä asteroidia.* Metodi palauttaa tällöin tyhjän merkkijonon. Tilanne voidaan luoda antamalla parametriksi aid negatiivinen kokonaisluku. Testiaineistona käytetään perustapauksen aineistoa. Parametriksi aid valitaan kaksi erillistä arvoa: -1 sekä Javan staattinen muuttuja Integer.MAX_VALUE.

3.2 AsteroidQueryBean

Luokka testataan erillisellä testiluokalla. Alla beaneille tyypillisten get- sekä set-metodien testaus on jätetty, koska ne ovat lausekattavasti testattuja jokaisella kutsukerralla, riippumatta syötteistä tai palautteesta. Muutamalla get metodilla on kuitenkin poikkeus tähän, getDetector():lla, getAbsolutePhotometry():llä sekä getLighttimeCorrected():lla, joista on testitapaukset alla. Tilannetta, jolloin mikään luokan kentistä olisi null-viite ei tule, koska konstruktori kutsuu luokan reset()-metodia, joka alustaa kaikki kentät ei-null-arvoiksi.

Koko luokan testaus on tehty JUnit-luokassa: *fi.helsinki.cs.group.tahiti.ui.Test.AsteroidQueryBeanTest*.

- *getDetector()* Perustilanteessa syötteenä setDetector()-metodille on taulukko, jossa ei ole sanaa "Other". Tällöin metodi palauttaa kyseisen taulukon takaisin muuttumattomana.

Testitapausta varten luodaan seuraava AsteroidQueryBean-luokan ilmentymä. Alla olevat tiedot ovat esitetty muodossa Kentän nimi: kentän syöte. Jos kentälle on esitetty useampia syötteitä pilkulla erotettuna, tämä tarkoittaa, että kyseiset syötteet ovat annettu kentälle taulukkona.

- Detector: "Photoelectric", "CCD"
- DetectorOther: "SomeOtherDetector"
- Seuraaviin kenttiin asetetaan arvoksi "-1": AsteroidNumber, MinimumLightcurves, MinimumDataPoints, Minimum/Maximum PhaseAngle, Minimum/Maximum DistanceFromSun, Minimum/Maximum DistanceFromEarth.
- Minimum/Maximum Latitude: "-91"
- MinimumLongitude: "Pujut"
- MaximumLongitude: "Jukran"

Testiaineistona perustapauksessa käytetään yllä luotua ilmentymää sekä merkkijonotaulukkoa "Photoelectric", "CCD".

Ainoa poikkeava tapaus on:

1. *Detector-tilaukossa on elementtinä "Other"* Tällöin metodi korvaa taulukon "Other" elementin luokan detectorOther kentän arvolla. Tilanne voidaan järjestää antamalla setDetector()-metodille parametriksi merkkijonotaulukko, joista yksi jäsen on "Other" ja antamalla setDetectorOther()-metodilla "Other" elementin korvaava teksti.

Käytetään *getDetector()*-metodin perustapauksessa luotua AsteroidQueryBean-luokan ilmentymää ja testiaineistona metodille merkkijonotaulukkoa "Photoelectric", "CCD", "Other".

- *getAbsolutePhotometry()* Perustilanteessa *getAbsolutePhotometry()*-metodi palauttaa merkkijonon "false" kyseisen kentän ollessa beanissa tyhjä merkkijono "". Jos kyseinen kenttä ei ole tyhjä merkkijono, metodi palauttaa kentän arvon.

Testiaineistona käytetään perustestissä luotua *AsteroidQueryBean*-luokan ilmentymää.

Testiaineistona käytetään *getDetector()*-metodin perustestin yhteydessä luotua *AsteroidQueryBean*-luokan ilmentymää.

- *getLighttimeCorrected()* Perustilanteessa *getLighttimeCorrected()*-metodi palauttaa merkkijonon "false" kyseisen kentän ollessa beanissa tyhjä merkkijono "". Jos kyseinen kenttä ei ole tyhjä merkkijono, metodi palauttaa kentän arvon.

Testiaineistona käytetään *getDetector()*-metodin perustestin yhteydessä luotua *AsteroidQueryBean*-luokan ilmentymää.

Lisätestin suorittamiseksi perustestin testiaineistoa muutetaan antamalla *setLighttimeCorrected*-metodille arvo "true".

- *getValueArray(String field)* Metodien tarkoitus on palauttaa parametrin määrittelemän tekstimuotoisen kentän arvo. Perustilanteessa parametri *field* on merkkijono "detector", jolloin metodi palauttaa kentän "detector" arvot. Huomioitavaa on, että metodi ei käytä tähän yllä olevaa *getDetector()*-metodia.

Testiaineistoksi luodaan uusi *AsteroidQueryBean*-luokan ilmentymä, jolle annetaan seuraavat syötteen:

- *Detector*: "Photoelectric", "CCD", "Other"
- *DetectorOther*: "OtherDetector"
- *AbsolutePhotometry*: "true"
- *LighttimeCorrected*: "true"
- *Filters*: "1", "3", "4"

Muita metodien testitapauksia ovat:

1. *Parametri field on eri kuin "detector"*. Parametrin "field" ollessa mitä tahansa muuta kuin null tai "detector", metodi palauttaa *filters* merkkijonotaulun. Testitapaus voidaan luoda helposti luomalla *AsteroidQueryBean* olio ja antamalla *getValueArray()*-metodille parametriksi "kissa".

Testiaineistona käytetään yllä luotua *AsteroidQueryBean*-luokan ilmentymää.

2. *Parametri field on null-viite*. Parametrin ollessa null-viite metodi palauttaa tyhjän merkkijonotaulukon.

Testiaineistona käytetään yllä luotua *AsteroidQueryBean*-luokan ilmentymää. Huomioitavaa, että testitapauksessa tyhjä merkkijonotaulukko todetaan vertaamalla merkkijonotaulukon pituutta nollaan.

- *validate()* *Validate*-metodi tarkastaa muuttujissa olevan tiedon oikeellisuuden. *Validointi*-metodi validoi tiettyjen arvojen oikeellisuuden. Tekstin yksinkertaistamiseksi alla on esitetty

testiaineisto, jolla validointimetodi palauttaa epätosiarvon, kysely on siis väärä. Lausekattavuus saavutetaan luomalla syöte, jossa kaikki allaolevat luvut eivät ole lukuja, ts. kokonaisluku on merkkijono ”abc”. Erillistä testitapausta validaation oikeellisuudelle ei tarvita, koska lausekattavuuteen vaaditaan vain erilaisten väärin syötteen tarkastelu.

1. Asteroidin numero ei ole positiivinen kokonaisluku.
2. Valokäyrien minimimäärä ei ole positiivinen kokonaisluku.
3. Valokäyrän havaintopisteiden määrä ei ole nolla tai positiivinen kokonaisluku.
4. Minimivaihekulma ei ole desimaaliluku väliltä 0-180.
5. Maksimivaihekulma ei ole desimaaliluku väliltä 0-180.
6. Minimimatka auringosta ei ole nolla tai positiivinen desimaaliluku.
7. Maksimimatka auringosta ei ole nolla tai positiivinen desimaaliluku.
8. Minimimatka maasta ei ole nolla tai positiivinen desimaaliluku.
9. Maksimimatka maasta ei ole nolla tai positiivinen desimaaliluku.
10. Minimilattitudi ei ole desimaaliluku väliltä 0-360.
11. Maksimilattitudi ei ole desimaaliluku väliltä 0-360.
12. Minimilongitudi ei ole desimaaliluku.
13. Maksimilongitudi ei ole desimaaliluku.
14. Havaintovälineeksi ei ole valittu ”Other” ja Otherille on annettu arvo.

Testiaineistona käytetään kahta eri AsteroidQueryBean-luokan ilmentymää. Ensimmäinen testi tehdään *getDetector()*-metodin yhteydessä tehdyllä ilmentymällä, toista testiä varten luodaan uusi AsteroidQueryBean-ilmentymä, jolle annetaan seuraavat syötteen:

- Detector: ”Photoelectric”, ”CCD”
 - DetectorOther: ”SomeOtherDetector”
 - Seuraaviin kenttiin asetetaan arvoksi ”a”: AsteroidNumber, MinimumLightcurves, MinimumDataPoints, Minimum/Maximum PhaseAngle, Minimum/Maximum DistanceFromSun, Minimum/Maximum DistanceFromEarth, Minimum/Maximum Latitude, Minimum/Maximum Longitude.
- *returnErrorMsg(String field)* Perusoletuksena parametriksi field valitaan jokin beania vastaavan lomakkeen kentän nimi jonka arvo ei ole läpäissyt validointitarkistusta. Metodi palauttaa tällöin tekstimuotoisen virheilmoituksen kyseisestä kentästä.

Testiaineistona käytetään *getDetector()*-metodin yhteydessä tehtyä AsteroidQueryBean-ilmentymää, jonka *validate()*-metodia kutsutaan, aiheuttaen virheilmoituksia.

Lausekattavuus saavutetaan seuraavilla lisätesteillä:

1. *Parametri field on null-viite.* Parametrin ollessa null-viite metodi palauttaa null-viitteen. Testi on helppo suorittaa kutsumalla metodia parametrina null-viite. Testiaineistona käytetään perustapauksessa yllä luotua aineistoa.

2. Parametriä *field* ei löydetä lomakkeen kentistä. Parametrin ollessa kenttä, jota lomakkeessa ei ole, metodi palauttaa null-viitteen.

Testiaineistona käytetään perustapauksessa yllä luotua aineistoa.

- *setApiError(int errornum)* Perusoletuksena parametri *errornum* on positiivinen kokonaisluku tai nolla, jolloin apin virhe liitetään muiden validoinnin yhteydessä tapahtuneiden virheiden joukkoon ja metodi palauttaa tosiarvon. Parametrin ollessa negatiivinen virhetietorakenteseen lisätään ilmoitus virheellisestä apin virheestä ja metodi palauttaa epätosiarvon..

Kaikkien *setApiError*-metodin testitapauksien testiaineistona toimii *getDetector()*-metodin yhteydessä esitelty *AsteroidQueryBean*-luokan ilmentymä.

AsteroidSubmitDataBean-luokka testataan erillisellä testiluokalla.

Testiluokkana toimii *fi.helsinki.cs.group.tahiti.ui.Test.AsteroidSubmitDataBeanTest*.

Validate-metodin testausta varten luokasta luodaan kaksi ilmentymää, jotka ladataan erilaisella datalla. Toisen on tarkoitus mennä läpi validoinnista ja toisen ei. Kunnossa olevan beanin sisältö:

```
\begin{itemize}
  \item Annetut syötteet:
    \begin{description}
      \item[Anomaly] ''1''
      \item[Au] ''2''
      \item[Designation] ''3''
      \item[Eccentricity] ''4''
      \item[Ref] ''6''
      \item[Perihelion] ''7''
      \item[Longitude] ''8''
    \item[Epoch] ''0''
      \item[Inclination] ''5''
      \item[Data] ''1 2 3 4 5 6 7 8 9''
    \end{description}
  \end{itemize}
```

Toimimattoman beanin sisältö:

```
\begin{itemize}
  \item Annetut syötteet:
    \begin{description}
      \item[Anomaly] ''1''
      \item[Au] ''2''
      \item[Designation] ''3''
      \item[Eccentricity] ''4''
      \item[Ref] ''6''
      \item[Perihelion] ''6''
      \item[Longitude] ''r''
    \end{description}
  \end{itemize}
```

```

\item[Epoch] ''0''
        \item[Inclination] ''5''
        \item[Data] ''''
\end{description}
\end{itemize}

```

Validate palautti odotetusti true-arvon ensimmäisen beanin tapauksessa ja false-arvon toisen beanin tapauksessa.

GenerateColumns-metodin testausta varten luotiin sarja lukuja, jotka vastasivat jsp-sivulta saatavaa sarakesarjaa, ja kokeiltiin, millainen saraketaulukko apia varten niistä muodostuu. Käsien laskettiin, että sarja 1,2,3,4,5,6,7,8,0 tuottaa tulokseksi sarjan 8,14,9,16,10,11,15,13,12. Näin myös tapahtui.

3.3 CustomElementSubmitBean

Luokka testataan erillisellä testiluokalla. Kuten yleensä, beanissa olevien get- sekä set-metodien testaus on jätetty, koska ne ovat lausekattavia jokaisella kutsukerralla. Sama voidaan sanoa konstruktorista sekä reset-metodista. Returnerrormsg-metodi on testattu AsteroidQueryBean-luokan testauksen yhteydessä, joten sen testiä ei tässä ole myöskään tehty.

Koko luokan testaus on tehty JUnit-luokassa: *fi.helsinki.cs.group.tahiti.ui.Test.CustomElementSubmitBeanTest*.

- *validate()* Kuten AsteroidQueryBean-metodin yhteydessä, validate-metodi tarkastaa käyttäjän antamien kenttien arvojen oikeellisuuden. Lausekattavuus saavutetaan testaamalla luokka niin, että kaikkiin kenttiin annetaan virheellistä numerotietoa. Sen lisäksi lausekattavuuteen vaaditaan, että kentät testataan myöskin arvoilla, jotka eivät ole numeroita. Seuraavassa on listattu oikeellisuusvaatimukset:
 1. Kentän epoch arvon täytyy olla positiivinen desimaaliluku.
 2. Kentän longitude tulee olla desimaaliluku väliltä 0-360.
 3. Kentän axis tulee olla desimaaliluku.
 4. Kentän anomaly tulee olla desimaaliluku väliltä 0-360.
 5. Kentän argument tulee olla desimaaliluku väliltä 0-360.
 6. Kentän eccentricity arvo tulee olla desimaaliluku väliltä 0-1.
 7. Kentän inclination arvo tulee olla desimaaliluku väliltä 0-360.

Testiaineistoksi luodaan kaksi erillistä CustomElementSubmitBean-luokan ilmentymää seuraavin syöttein:

- Seuraaviin kenttiin asetetaan arvoksi -1": Inclination, Eccentricity, Argument, Anomaly, Longitude.
- Axis: "foo"
- Epoch "0"

Ei numeroarvotestiä varten yllä luotua ilmentymää muutetaan seuraavin syöttein:

- Kaikille yllälistatuille kentille asetetaan arvoksi "a".

3.4 EventHistoryDataBean

EventHistoryDataBean sisältää pelkästään get- ja set-metodin sekä reset-metodin, joten sen testaaminen on sivuutettu triviaalina.

3.5 LightcurveDataBean

LightcurveDataBean-luokassa säilytetään tietyn asteroidin kaikkia järjestelmän asteroidihaun yhteydessä löytämiä valokäyrätietoja. Tämän lisäksi luokassa on kaikki asteroidiin sekä maahan liittyvät ratatiedot. Luokkaan liitetään myös kaikki käyttäjän valokäyrille itse luomat ratatiedot. Metodi testataan erillisellä testiluokalla. Alla on testattu lausekattavasti vain metodit, joissa on haarautuvia polkuja. Yksisuuntaiset polut voidaan todeta lausekattaviksi kutsumalla niitä kerran.

Beanin testaus on tehty luokassa *fi.helsinki.cs.group.tahiti.ui.Test.LightcurveDataBeanTest*.

- *findLightcurve(long lcid)* Perustestissä metodille annetaan arvoksi jokin valokäyrän numerotunniste, joka löytyy tästä luokan ilmentymästä. Tällöin metodi palauttaa kyseisen valokäyrän sijainnin taulukossa. Testi voidaan suorittaa luomalla instanssi tästä luokasta niin, että haluttu valokäyrän numerotunniste löytyy.

Testiaineistona perustapauksessa käytetään seuraavia LightcurveDataBean-luokan ilmentymää, jolle annetaan parametreinä seuraavanlaiset valokäyrä sekä ratatiedot:

- Luodaan viisi kappaletta maan erilaisia ratatietoja taulukkoon, joiden syötteinä ovat seuraavat arvot:
 - * Ratatiedon id: 1 - 5
 - * Päiväys (epoch): 05-01-2003 - 05-05-2003
- Luodaan myöskin viisi kappaletta asteroidin erilaisia ratatietoja taulukkoon, joiden syötteinä ovat seuraavat arvot:
 - * Ratatiedon id: 5 - 1
 - * Päiväys (epoch): 06-01-2003 - 06-05-2003
- Tämän lisäksi luodaan viisi kappaletta ApiLightcurve-luokan instanssia, joille annetaan seuraavat arvot:
 - * Valokäyrän id: 0 - 4
 - * Valokäyrän asteroididatatieid id: 1 - 5
 - * Valokäyrän maan ratatiedon id: 1, 0, 3, 0, 5

Lisäksi metodi on testattava sellaisella valokäyrän tunnisteella, jota ei löydy järjestelmästä. Tällöin metodi palauttaa -1 ja on lausekattavasti testattu. Kyseinen testi voidaan luoda helposti testaamalla metodia negatiivisella arvolla.

Testiaineistona käytetään perustestissä luotua aineistoa.

- *insertCustomElement(ApiTrajectory ce, long lcId, boolean earthTrajectory)* Perustestinä metodi saa maan rataelementtitiedon olemassaolevalle valokäyrälle. Tällöin metodi palauttaa tosiarvon. Perustilanne voidaan simuloida luomalla uusi ratatietoluokan instanssi, antamalla tämän luokan sisällä olevien valokäyrien joukosta jonkin tunnus ja antamalla viimeiseksi parametriksi true.

Perustestiä varten luodaan ApiTrajectory-luokan ilmentymä, joka kuvastaa käyttäjän antamaa ratatietoa, jolla on ratatiedon id:ksi annettu luku 1024. Tämän lisäksi testiaineistona käytetään *findLightcurve()*-metodin perustapauksen yhteydessä luotua LightcurveDataBean-luokan ilmentymää.

Lausekattavuuden saavuttamiseksi metodi on lisäksi testattava seuraavilla tapauksilla:

1. Halutaan lisätä validille valokäyrälle asteroidin rataelementti. Metodi palauttaa tosiarvon. Testi voidaan luoda kuten yllä antamalla viimeiseksi boolean-parametriksi false. Testiaineistona käytetään kyseisen metodin perustapauksessa luotua testiaineistoa.
2. Haluttua valokäyrää ei löydy järjestelmästä. Metodi palauttaa epätosiarvon. Testi voidaan luoda antamalla valokäyrän tunnisteeksi negatiivinen luku. Testiaineistona käytetään kyseisen metodin perustapauksessa luotua testiaineistoa.

- *changeTrajectoryInfo(long lcId, boolean earthTrajectory, long newTrajectoryType)* Perustestissä metodille annetaan arvoksi jokin ilmentymässä olevan valokäyrän tunnus jolle halutaan vaihtaa maan ratatiedoksi jokin viimeisen parametrin määräämä ratatietotunnus. Tällöin metodi palauttaa tosiarvon. Testi voidaan tehdä luomalla instanssi tästä oliosta ja antamalla metodille parametriksi jotkin oikeelliset valokäyrä- sekä ratatietotunnisteet.

Testiaineistona käytetään *findLightcurve()*-metodin perustapauksen yhteydessä luotua LightcurveDataBean-luokan ilmentymää. Testin jälkeen luokan tila palautetaan testiä edeltävään tilaan.

Lausekattavuus syntyy seuraavin lisätestein:

1. *Valokäyrää ei löydy luokan ilmentymästä.* Metodi palauttaa epätosi. Testi voidaan luoda antamalla valokäyrän tunnukseksi negatiivinen kokonaisluku. Testiaineistona käytetään metodin perustapauksessa käytettävää aineistoa. Tilaa ei tarvitse muuttaa takaisin testiä edeltävään tilaan.
 2. *Valokäyrälle halutaan muuttaa asteroidin rataelementtityyppiä.* Testi palauttaa tosiarvon. Testin suoritus käy kuten perustapauksessa, parametrin earthTrajectory arvoksi asetetaan vain false. Testiaineistona käytetään metodin perustapauksessa käytettävää aineistoa. Testin jälkeen tila palautetaan testiä edeltäneeseen tilaan.
- *getTrajectory(long lcId, boolean earthTraj)* Perustapauksessa metodille annetaan jokin luokan ilmentymässä olevan valokäyrän tunnus jonka maan ratatiedon kutsuja haluaa. Valokäyrälle on määritelty ilmentymässä maan ratatietotyyppiä 0 ("Uppsala"). Metodi palauttaa kutsujalle null-viitteen. Testi voidaan tehdä luomalla uusi instanssi luokasta jolle on annettu oikeelliset valokäyrä- sekä ratatiedot.

Testiaineistona käytetään *findLightcurve()*-metodin yhteydessä luotua LightcurveDataBean-luokan ilmentymää. Kyseisen luokan tilaa muutetaan vaihtamalla ensimmäisen valokäyrän maan ratatiedon tyyppiä ”Uppsala” (luku 0). Testin jälkeen tila palautetaan testiä edeltävään tilaan.

Lausekattavuus varmistetaan testaamalla seuraavat lisäehdot:

1. *Valokäyrää ei löydy ilmentymästä.* Metodi palauttaa null-viitteen. Tapaus voidaan testata antamalla metodilla negatiivinen kokonaisluku parametriksi *lcid*.
Testiaineistona käytetään *findLightcurve()*-metodin yhteydessä luotua LightcurveDataBean-luokan ilmentymää.
2. *Kutsuja haluaa maan ratatiedon valokäyrälle, jonka ratatietotyyppi on ”Custom” (-1).* Metodi palauttaa null-viitteen. Testi voidaan toteuttaa antamalla kyseiselle valokäyrälle maan ratatiedoksi jokin omavalintainen ratatieto *insertCustomElement*-metodin avulla. Testiaineistoksi luodaan uusi *ApiTrajectory*-luokan ilmentymä, joka annetaan *findLightcurve()*-metodin yhteydessä luodulle LightcurveDataBean-luokan ilmentymälle maan *ra*-taelementiksi. Testin jälkeen valokäyrän ratatiedon tyyppi palautetaan testiä edeltävään tilaan.
3. *Kutsuja haluaa maan ratatiedon valokäyrälle, jonka ratatietotyyppi on positiivinen kokonaisluku.* Käyttäjä haluaa tällöin jonkin järjestelmässä maalle olevan ratatiedon. Metodi palauttaa kyseisen ratatiedon. Testi voidaan suorittaa antamalla järjestelmälle ratatietoinformaatiota niin, että kyseisen valokäyrän *getEarthTrajectoryID()* palauttaa kyseisen ratatietoinformaation tunnuksen.
Testiaineistona käytetään *findLightcurve()*-metodin yhteydessä luotua LightcurveDataBean-luokan ilmentymää.
4. *Kutsuja haluaa asteroidin ratatiedon valokäyrälle, jonka ratatietotyyppi on ”Uppsala” (0).* Metodi palauttaa null-viitteen. Testi voidaan toteuttaa kuten perustapaus.
Testiaineistona käytetään *findLightcurve()*-metodin yhteydessä luotua LightcurveDataBean-luokan ilmentymää. Luokan tilaa muutetaan niin, että ensimmäiselle valokäyrälle järjestelmässä muutetaan asteroidin ratatyyppi ”Uppsala”-arvoksi. Testin jälkeen tila palautetaan ennalleen.
5. *Kutsuja haluaa asteroidin ratatiedon valokäyrälle, jonka ratatietotyyppi on ”Custom” (-1).* Metodi palauttaa null-viitteen. Testi voidaan toteuttaa antamalla kyseiselle valokäyrälle asteroidin ratatiedoksi jokin omavalintainen ratatieto *insertCustomElement*-metodin avulla.
Testiaineistona käytetään *findLightcurve()*-metodin yhteydessä luotua LightcurveDataBean-luokan ilmentymää. Tämän lisäksi käytetään lisätestin kaksi yhteydessä luotua *ApiTrajectory*-luokan ilmentymää, joka annetaan LightcurveDataBean-luokalle. Testin jälkeen valokäyrän ratatiedon tyyppi palautetaan testiä edeltävään tilaan.
6. *Kutsuja haluaa asteroidin ratatiedon valokäyrälle, jonka ratatietotyyppi on positiivinen kokonaisluku.* Käyttäjä haluaa tällöin jonkin järjestelmässä asteroidille olevan ratatiedon. Metodi palauttaa kyseisen ratatiedon. Testi voidaan suorittaa antamalla järjestelmälle ratatietoinformaatiota niin, että kyseisen valokäyrän *getDefaultTrajectoryID()* palauttaa kyseisen ratatietoinformaation tunnuksen.

Testiaineistona käytetään *findLightcurve()*-metodin yhteydessä luotua *LightcurveDataBean*-luokan ilmentymää.

- *getTrajectoryOptions(boolean earthTrajectories, int pos)* Peruskutsulla halutaan maan ratatietotekstit luokan ilmentymästä löytyvälle ensimmäiselle valokäyrälle. Testi palauttaa merkkijonon, joka on muotoa "Uppsala, <ratatietopäiväys>, <ratatietopäiväys>, Custom". Testi voidaan järjestää luomalla instanssi luokasta jolle annetaan oikeaa valokäyräinformaatiota, ja jonka datapisteille on määritelty "Uppsala"-arvot.

Testiaineistona käytetään *findLightcurve()*-metodin yhteydessä luotua *LightcurveDataBean*-luokan ilmentymää.

Lausekattavuus saavutetaan seuraavin lisätestein:

1. *Parametri pos on virheellinen.* Kutsujalle palautetaan tyhjä merkkijono. Testi voidaan helposti toteuttaa kutsumalla metodia negatiivisella pos-parametrilla.

Testiaineistona käytetään *findLightcurve()*-metodin yhteydessä luotua *LightcurveDataBean*-luokan ilmentymää.

2. *Kutsuja haluaa maan ratatietotekstit valokäyrälle jolla on "Uppsala"-arvot.* Metodi palauttaa merkkijonon, joka on samaa muotoa kuin perustestitapauksessa. Testi järjestää kuten perustestitapaus.

Testiaineistona käytetään *findLightcurve()*-metodin yhteydessä luotua *LightcurveDataBean*-luokan ilmentymää.

- *getTrajectoryOptionValues(boolean earthTrajectories, int lcid)* Metodien testitapaukset vastaavat täsmällisesti *getTrajectoryOptions*-metodin testitapauksia. Tekstin "Uppsala" tilalla on nyt luku nolla, "Custom"-tekstin tilalla on luku -1 ja normaalien ratatietojen luvut ovat kyseisen ratatiedon tunnus (1..n).
- *getTrajectoryType(long lcid, boolean earthTrajectory)* Perustestissä halutaan tietää järjestelmästä löytyvän valokäyrän maan ratatietotyyppi. Ratatietotyyppi voi olla kolmea eri tyyppiä (-1, 0, 1..n). Metodi palauttaa 0. Testi voidaan järjestää luomalla instanssi luokasta ja antamalla sille oikeellista valokäyrätietoa niin, että kyseisellä valokäyrällä on "Uppsala"-arvot. Tämän jälkeen kyseistä metodia kutsutaan.

Testiaineistona käytetään *findLightcurve()*-metodin yhteydessä luotua *LightcurveDataBean*-luokan ilmentymää.

Tämän lisäksi lausekattavuuden takia metodi testataan seuraavin lisätestein:

1. *Valokäyrää ei löydy järjestelmästä.* Metodi palauttaa nollan. Testi voidaan järjestää kutsumalla metodia negatiivisella lcid arvolla.

Testiaineistona käytetään *findLightcurve()*-metodin yhteydessä luotua *LightcurveDataBean*-luokan ilmentymää.

2. *Halutaan asteroidin ratatietotyyppi.* Metodi palauttaa halutun valokäyrän asteroidin tämän hetkisen ratatietotyyppin. Testi voidaan järjestää kuten perustesti.

Testiaineistona käytetään *findLightcurve()*-metodin yhteydessä luotua *LightcurveDataBean*-luokan ilmentymää.

- *getLightcurve(long lcid)* Perustilanteessa metodia kutsutaan valokäyrän tunnuksella, joka löytyy luokan instanssista. Metodi palauttaa kyseisen valokäyrän instanssin.

Testiaineistona käytetään *findLightcurve()*-metodin yhteydessä luotua *LightcurveDataBean*-luokan ilmentymää.

Lausekattavuus todetaan testaamalla metodi valokäyrätunnuksella, jota ei ole olemassa, esimerkiksi negatiivisella arvolla. Tällöin metodi palauttaa null-viitteen.

Testiaineistona käytetään *findLightcurve()*-metodin yhteydessä luotua *LightcurveDataBean*-luokan ilmentymää.

3.6 LightcurveQueryBean

Luokka testataan erillisellä testiluokalla. Alla beaneille tyypillisten *get*- sekä *set*-metodien testaus on jätetty, koska ne ovat lausekattavasti testattuja jokaisella kutsukerralla, riippumatta syötteistä tai palautteesta.

Beanin testaus on tehty luokassa *fi.helsinki.cs.group.tahiti.ui.Test.LightcurveQueryBeanTest*.

- *isSelected(long lc)* Perustapauksessa valokäyrä on valittuna. Metodi palauttaa tällöin tosiarvon. Testi voidaan järjestää antamalla metodille *setLcid()* metodin avulla taulukko, jossa on haluttu *lc*-parametrin arvo.

Perustestiä varten luodaan ilmentymä *LightcurveQueryBean*-luokasta, jolle annetaan seuraavat syötteet:

- Kenttä *Lcid*: "1", "2", "10", "399"

Lisäksi metodi testataan *lc* arvolla, jota ei ole valittu. Testi voidaan järjestää valitsemalla *lc:n* arvoksi jokin arvo, jota *setLcid()* metodin avulla ei ole määritelty.

Testiaineistona käytetään perustestissä määriteltyä *LightcurveQueryBean*-luokan ilmentymää.

LightcurveSubmitBean-luokka testataan erillisellä testiluokalla.

Testiluokkana toimii *fi.helsinki.cs.group.tahiti.ui.Test.LightcurveSubmitBeanTest*.

Validate-metodin testausta varten luokasta luodaan kaksi ilmentymää, jotka ladataan erilaisella datalla. Toisen on tarkoitus mennä läpi validoinnista ja toisen ei. Kunnossa olevan beanin sisältö:

```
\begin{itemize}
  \item Annetut syötteet:
    \begin{description}
      \item[AsteroidNumber] ''323''
      \item[ObservingSite] ''paikka''
      \item[Observer] ''Ismo''
      \item[ZeroTime] ''0''
      \item[UnitOfTime] ''0''
      \item[AbsolutePhotometry] ''false''
    \end{description}
  \end{itemize}
```

```

        \item[ZeroMagnitude] ''0''
        \item[LightTimeCorrected] ''false''
        \item[Data] ''Tot''
        \item[Detector] ''CCD''
        \item[Time] ''0''
        \item[U] ''1''
        \item[B] ''2''
        \item[V] ''3''
        \item[R] ''4''
        \item[I] ''5''
        \item[Unfiltered] ''6''
        \item[Error] ''7''
    \end{description}
\end{itemize}

```

Toimimattoman beanin sisältö:

```

\begin{itemize}
    \item Annetut syötteet:
        \begin{description}
            \item[AsteroidNumber] ''''
            \item[ObservingSite] ''''
            \item[Observer] ''''
            \item[ZeroTime] ''0''
            \item[UnitOfTime] ''0''
            \item[AbsolutePhotometry] ''false''
            \item[ZeroMagnitude] ''0''
            \item[LightTimeCorrected] null
            \item[Data] ''''
            \item[Detector] null
            \item[Time] ''0''
            \item[U] ''1''
            \item[B] ''2''
            \item[V] ''3''
            \item[R] ''4''
            \item[I] ''6''
            \item[Unfiltered] ''6''
            \item[Error] ''7''
        \end{description}
    \end{itemize}

```

Validate palautti odotetusti true-arvon ensimmäisen beanin tapauksessa ja false-arvon toisen beanin tapauksessa.

GenerateColumns-metodin testausta varten luotiin sarja lukuja, jotka vastasivat jsp-sivulta saa-

tavaa sarakesarjaa, ja kokeiltiin, millainen saraketaulukko apia varten niistä muodostuu. Käsien laskettiin, että sarja 0,1,2,3,4,5,6,7 tuottaa tulokseksi sarjan 0,2,3,4,5,6,7,1. Näin myös tapahtui.

3.7 LoginBean

LoginBean testataan erillisellä testiluokalla. `getErrorMessage`-, `get`- sekä `set`- ja `setApiError`-metodit ovat testattu esimerkiksi `AsteroidQueryBean`-luokan yhteydessä, ja koska tämän luokan vastaavat metodit ovat identtisiä näiden kanssa, niiden moduulitestausta lausekattavasti ei ole alla tehty.

- `validate()` Seuraavilla kenttien arvoilla metodi tulee testauksi lausekattavasti:

1. Kentän `loginName` pituus on alle neljä merkkiä.
2. Kentän `loginPw` pituus on alle neljä merkkiä.

Testi palauttaa epätosiarvon. Testi voidaan järjestää luomalla uusi `LoginBean` instanssi ja kutsumalla sen jälkeen kyseistä metodia.

3.8 LoginDataBean

`LoginDataBean`-luokka testataan erillisellä testiluokalla.

Testiluokkana toimii `fi.helsinki.cs.group.tahiti.ui.Test.LoginDataBeanTest`.

- `getName()` Perustapauksessa `LoginDataBean`-ilmentymällä on kentässä `loginInfo` viite `User`-luokan ilmentymään. Tällöin metodi palauttaa kyseisen `User`-luokan ilmentymän `getName()` metodin palauttaman arvon. Testi voidaan luoda luomalla uusi ilmentymä `User`-luokasta ja annetaan kyseinen ilmentymä konstruktorissa `LoginDataBean`-luokalle.

Perustestin testiaineistoksi luodaan uusi `LoginDataBean`-luokan ilmentymä, jolle annetaan seuraavat syötteen:

- `ApiUser`-luokan ilmentymä, jolle annetaan seuraavat syötteen:
 - * Nimi: "Test"
 - * Käyttäjät: 1
 - * Käyttäjätaso: 1

Ainoa lisätestetapaus lausekattavuuden saavuttamiseksi on tapaus, jossa `LoginDataBean`-luokan ilmentymällä ei ole `User`-luokan ilmentymää. Tällöin metodi palauttaa tyhjän merkijonon. Testitapaus voidaan luoda luomalla uusi ilmentymä `LoginDataBean`-luokasta sen peruskonstruktorilla.

Testiaineistoksi luodaan uusi (tyhjä) `LoginDataBean`-luokan ilmentymä.

- `getUserID()` Perustapauksessa `LoginDataBean`-ilmentymällä on kentässä `loginInfo` viite `User`-luokan ilmentymään. Tällöin metodi palauttaa kyseisen `User`-luokan ilmentymän `getUserID()` metodin palauttaman arvon. Testi voidaan luoda luomalla uusi ilmentymä `User`-luokasta ja annetaan kyseinen ilmentymä konstruktorissa `LoginDataBean`-luokalle.

Testiaineistona toimii *getName()*-metodin perustestissä esitelty *LoginDataBean*-luokan ilmentymä.

Ainoa lisätestitapaus lausekattavuuden saavuttamiseksi on tapaus, jossa *LoginDataBean*-luokan ilmentymällä ei ole *User*-luokan ilmentymää. Tällöin metodi palauttaa arvon -1. Testitapaus voidaan luoda luomalla uusi ilmentymä *LoginDataBean*-luokasta sen peruskonstruktorilla.

Testiaineistoksi luodaan uusi (tyhjä) *LoginDataBean*-luokan ilmentymä.

- *getLevel()* Metodi voidaan testata kuten *getUserID*-metodi. Perustapauksessa palautetaan *User*-ilmentymän *getLevel*-metodin arvo. Samanlaisessa lisätestitapauksessa kuin yllä palautetaan nolla.

Testiaineistona toimii *getName()*-metodin perustestissä esitelty *LoginDataBean*-luokan ilmentymä.

Lisätestitapauksen testiaineistona on tyhjä *LoginDataBean*-luokan ilmentymä.

- *loggedIn()* Perustestissä *LoginDataBean*-luokan ilmentymällä on kentässä *loginInfo* viite *User*-luokan ilmentymään ja kyseisen *User*-luokan ilmentymän *getLevel*-metodi palauttaa positiivisen kokonaisluvun. Tällöin metodi palauttaa tosiarvon.

Testiaineistona toimii *getName()*-metodin perustestissä esitelty *LoginDataBean*-luokan ilmentymä.

Ainoa lisätestitapaus lausekattavuuden saavuttamiseksi syntyy kun *User*-luokan ilmentymää ei *LoginDataBean*-luokan ilmentymällä ole. Tällöin metodi palauttaa arvon *false*.

Lisätestitapauksen testiaineistona on tyhjä *LoginDataBean*-luokan ilmentymä.

LogQueryBean-luokka testataan erillisellä testiluokalla.

Testiluokkana toimii *fi.helsinki.cs.group.tahiti.ui.Test.LogQueryBeanTest*.

Validate-metodin testausta varten luokasta luodaan neljä ilmentymää, jotka ladataan erilaisella datalla. Ensimmäisen on tarkoitus mennä läpi validoinnista ja muiden ei. Kunnossa olevan beanin sisältö:

```
\begin{itemize}
  \item Annetut syötteet:
    \begin{description}
      \item[TimeStart] ''23.3.2003''
      \item[TimeEnd] ''2.4.2003''
    \end{description}
\end{itemize}
```

Ensimmäisen toimimattoman beanin sisältö:

```
\begin{itemize}
  \item Annetut syötteet:
    \begin{description}
```

```

        \item[TimeStart] ''21.4.2003''
        \item[TimeEnd] ''21.3.2003''
    \end{description}
\end{itemize}

```

Toisen toimimattoman beanin sisältö:

```

\begin{itemize}
    \item Annetut syötteet:
        \begin{description}
            \item[TimeStart] ''Toot''
            \item[TimeEnd] null
        \end{description}
\end{itemize}

```

Kolmannen toimimattoman beanin sisältö:

```

\begin{itemize}
    \item Annetut syötteet:
        \begin{description}
            \item[TimeStart] ''21.9.2007''
            \item[TimeEnd] ''21.10.2009''
        \end{description}
\end{itemize}

```

Validate palautti odotetusti true-arvon ensimmäisen beanin tapauksessa ja false-arvon muiden beanien tapauksessa.

MailSubmitBean-luokka testataan erillisellä testiluokalla.

Testiluokkana toimii *fi.helsinki.cs.group.tahiti.ui.Test.MailSubmitBeanTest*.

Validate-metodin testausta varten luokasta luodaan kaksi ilmentymää, jotka ladataan erilaisella datalla. Toisen on tarkoitus mennä läpi validoinnista ja toisen ei. Kunnossa olevan beanin sisältö:

```

\begin{itemize}
    \item Annetut syötteet:
        \begin{description}
            \item[ActionType] ''0''
            \item[UserID] ''21''
            \item[UserName] ''Tauno''
        \end{description}
\end{itemize}

```

Toimimattoman beanin sisältö:

```

\begin{itemize}

```



```

\item Annetut syötteet:
  \begin{description}
    \item[ActionType] '''
    \item[UserID] null
    \item[UserName] '''
  \end{description}
\end{itemize}

```

Validate palautti odotetusti true-arvon ensimmäisen beanin tapauksessa ja false-arvon toisen beanin tapauksessa.

3.9 SiteConfigurationSubmitDataBean

Alla beaneille tyypillisten get- sekä set-metodien testaus on jätetty, koska ne ovat lausekattavasti testattuja jokaisella kutsukerralla, riippumatta syötteistä tai palautteesta. Myös reset-metodi tulee lausekattavasti testatuksi jokaisella kutsumiskerralla.

validate() Validate-metodin tarkoitus on palauttaa virhearvo, jos joku beanin muuttujista puuttuu tai on tyypiltään virheellinen. Tosiarvon palauttamiseen vaaditaan seuraavat ehdot:

adminEMail täytyy olla epätyhjä merkkijono, joka on vähintään kuusi merkkiä pitkä.

mailServer täytyy olla epätyhjä merkkijono, joka on vähintään kolme merkkiä pitkä.

serverPort täytyy olla epätyhjä numeromuotoinen merkkijono, joka on välillä 1 – 65534.

passwordMessage täytyy olla epätyhjä merkkijono, joka on vähintään kolme merkkiä pitkä.

approveMessage täytyy olla epätyhjä merkkijono, joka on vähintään kolme merkkiä pitkä.

referenceMessage täytyy olla epätyhjä merkkijono, joka on vähintään kolme merkkiä pitkä.

passwordMessageHeader täytyy olla epätyhjä merkkijono, joka on vähintään kolme merkkiä pitkä.

approveMessageHeader täytyy olla epätyhjä merkkijono, joka on vähintään kolme merkkiä pitkä.

referenceMessageHeader täytyy olla epätyhjä merkkijono, joka on vähintään kolme merkkiä pitkä.

Testaus suoritetaan ajamalla metodi oikealla syöttellä ja syötteellä, jossa muuttujilla on arvonaan null-arvo tai tyhjä merkkijono. Tuloksien pitäisi olla true ja false.

UserInfoDataBean-luokka testataan erillisellä testiluokalla.

Testiluokkana toimii *fi.helsinki.cs.group.tahiti.ui.Test.UserInfoDataBeanTest*.

Validate-metodin testausta varten luokasta luodaan kaksi ilmentymää, jotka ladataan erilaisella datalla. Toisen on tarkoitus mennä läpi validoinnista ja toisen ei. Kunnossa olevan beanin sisältö:

```

\begin{itemize}
  \item Annetut syötteet:
    \begin{description}
      \item[RecommenderEMail] ''reference@reference.ref''
    \end{description}

```

```

        \item[EMail] ''testi@testi.tes''
        \item[Recommender] ''Heikki Taunonen''
        \item[UserName] ''tauno''
        \item[Name] ''Tauno Heikkinen''
    \end{description}
\end{itemize}

```

Toimimattoman beanin sisältö:

```

\begin{itemize}
    \item Annetut syötteet:
        \begin{description}
            \item[RecommenderEMail] ''ref''
            \item[EMail] ''tsti''
            \item[Recommender] ''''
            \item[UserName] ''tau''
            \item[Name] ''Tau''
        \end{description}
\end{itemize}

```

Validate palautti odotetusti true-arvon ensimmäisen beanin tapauksessa ja false-arvon toisen beanin tapauksessa.

UserSearchBean-luokka testataan erillisellä testiluokalla.

Testiluokkana toimii *fi.helsinki.cs.group.tahiti.ui.Test.UserSearchBeanTest*.

Validate-metodin testausta varten luokasta luodaan kaksi ilmentymää, jotka ladataan erilaisella datalla. Toisen on tarkoitus mennä läpi validoinnista ja toisen ei. Kunnossa olevan beanin sisältö:

```

\begin{itemize}
    \item Annetut syötteet:
        \begin{description}
            \item[UserName] ''Tauno''
        \end{description}
\end{itemize}

```

Toimimattoman beanin sisältö:

```

\begin{itemize}
    \item Annetut syötteet:
        \begin{description}
            \item[UserName] null
        \end{description}
\end{itemize}

```

Validate palautti odotetusti true-arvon ensimmäisen beanin tapauksessa ja false-arvon toisen beanin tapauksessa.

3.10 UserSearchDataBean

EventHistoryDataBean sisältää pelkästään get- ja set-metodin, joten sen testaaminen on sivuutettu triviaalina.

3.11 AtlasImporter

AtlasImporteria ei testata sen erikoisluontoisuuden vuoksi metodi kerrallaan, vaan testaus suoritetaan kyseistä ohjelmaa suorittamalla. Käytössä on oikeastaan AtlasImporterin testiluokka, joka ei kutsu atlasimporter-pakagen ulkopuolisia luokkia, vaan esimerkiksi tulostaa tietokantaan oikeasti syötettävät arvot näytölle. Testaus voidaan jakaa kahteen osaan. Ensiksi testataan ohjelmalle annettavat argumentit, ja sen jälkeen testataan testausta varten rakennettu atlas-tiedosto. Tällaisella testauksella päästään ohjelman olennaiset osat koskevaan lausekattavuuteen. Ensimmäisessä vaiheessa testataan tapaukset: väärä määrä argumentteja ja atlashakemisto tai tiedosto, jota ei ole olemassa. Jälkimmäisessä vaiheessa AtlasImporterin pitäisi tulostaa ruudulle testi-atlas-tiedostossa annetut tiedot tai antaa virheilmoitus, jos pakollisia kenttiä ei ole täytetty. Testi-atlas-tiedosto on tämän dokumentin liitteenä.

Testi eteni seuraavasti: Väärällä määrällä argumentteja AtlasImporter palauttaa: Usage: AtlasImporter [username] [password] [atlasfile/directory] [rejected lightcurve files directory]

Testiaineisto palautti odotetut tulokset, jotka on esitetty myöskin liitteessä.

3.12 DBControl

3.12.1 Julkiset metodit

- *makeUpdate(String update[])* Metodi saadaan testattua testaus-informaatiota sisältäviä yksittäisiä lauseita lukuun ottamatta lausekattavasti testaamalla tapaus, jossa tietokantayhteys on olemassa ja syötteenä annetut SQL-lauseet ovat toimivia; ja tapaus, jossa tietokantayhteyttä ei ole olemassa tai SQL-lauseista joku on virheellinen.

– Luokan muuttujien tila

conn tietokantayhteys on olemassa

– Annetut syötteet

update[] insert into Asteroid values (111,'Testi','Krypton',12345), insert into Asteroid values (112,'Testi','Uraanus',54321)

– Metodi palautti:

2

Tietokantataulussa Asteroid: 111, Testi, Krypton, 12345 ja 112, Testi, Uraanus, 54321

– Luokan muuttujien tila

conn tietokantayhteyttä ei ole olemassa

– Annetut syötteet

update[] insert into Asteroid values (111,'Testi','Krypton',12345), insert into Asteroid values (112,'Testi','Uraanus',54321)

– Metodi palautti:

0

Virhe ERR_NO_CONNECTION_OR_ERRONEOUS_SQL.

- *makeUpdate(String sql[], int expectedRowCount[], String log[])* Metodi saadaan testattua lausekattavasti ulkoisten seikkojen aiheuttamia virhetilanteita (tietokantayhteys katkeaa kesken suorituksen) lukuun ottamatta käymällä läpi seuraavat tapaukset:

– *sql*- ja *expectedRowCount*-taulukko eivät ole saman kokoisia Metodi palauttaa false ja virhekentässä arvo TahitiLibrary.ERR_WRONG_ARRAY_SIZES.

– virheetön suoritus: *sql*- ja *expectedRowCount*-taulukot ovat saman kokoisia, tietokantayhteys on olemassa ja SQL-lauseet ovat virheettömiä, *expectedRowCount*in arvot vastaavat päivitettyjen rivien määrää sekä *log* syöte ei ole null Metodi palauttaa true.

– Tietokantayhteyttä ei ole olemassa tai syötteenä annettu SQL on virheellistä Metodi palauttaa false ja virhekentässä arvo Tahiti.Library.ERR_NO_CONNECTION_OR_ERRONEOUS_SQL.

– *expectedRowCount*it eivät vastaa päivitettyjen rivien määrää Metodi palauttaa false ja virhekentässä arvo TahitiLibrary.ERR_ROW_COUNT_FAILED.

Testit:

– Luokan muuttujien tila

conn tietokantayhteys on olemassa

– Annetut syötteet

sql[] insert into Asteroid values (113,'Testi','Mars',8888), insert into Asteroid values (114,'Testi','Sputnik',777)

expectedRowCount[] 1, 1, 1

log[] null

– Metodi palautti:

false

Virhe ERR_WRONG_ARRAY_SIZES

– Luokan muuttujien tila

conn tietokantayhteys on olemassa

– Annetut syötteet

sql[] insert into Asteroid values (117,'Testi','Kuukuu',8888), insert into Asteroid values (118,'Testi','Apollo 13',777)

expectedRowCount[] 1, 1

log[] insert into Asteroid values (123,'Testi','Loki1',8888), insert into Asteroid values (456,'Testi','Loki2',777)

– Metodi palautti:

true

Tietokantatauluun asteroid: 117, Testi, Kuukuu, 8888; 118, Testi, Apollo 13, 777; 123, Testi, Loki1, 8888; 456, Testi, Loki2, 777

– Luokan muuttujien tila

conn tietokantayhteyttä ei ole olemassa

– Annetut syötteet

sql[] insert into Asteroid values (117,'Testi','Kuukuu',8888), insert into Asteroid values (118,'Testi','Apollo 13',777)

expectedRowCount[] 1, 1

log[] insert into Asteroid values (123,'Testi','Loki1',8888), insert into Asteroid values (456,'Testi','Loki2',777)

– Metodi palautti:

false

Virhe ERR_NO_CONNECTION_OR_ERRONEOUS_SQL

– Luokan muuttujien tila

conn tietokantayhteys on olemassa

– Annetut syötteet

sql[] insert into Asteroid values (115,'Testi','Pluto',8888), insert into Asteroid values (116,'Testi','MikkiHiiri',777)

expectedRowCount[] 3, 1

log[] null

– Metodi palautti:

false

Virhe ERR_ROW_COUNT_FAILED

- *makeQuery(String query)* Metodi muodostuu yhdestä try-catch-osiosta, ja näin ollen testattavia tapauksia on kaksi. Ensimmäisessä tapauksessa tietokantayhteys on olemassa ja syötteenä annetaan toimiva SQL-lause. Jälkimmäisessä tapauksessa tietokantayhteyttä ei ole tai SQL-lause on virheellinen.

– Luokan muuttujien tila

conn tietokantayhteys on olemassa

– Annetut syötteet

query select * from Asteroid;

– Tietokantaan kohdistuvat oletukset:

asteroidnumber 0, 1, 2

designation Earth, , 1998 SF36

name Earth, Ida, (tyhjä)

number 0, 234, 25143

– Metodi palautti:

ResultSet ResultSetistä oli saatavissa esimerkiksi sarakkeen 3 arvot: Earth, Ida, Null

– Luokan muuttujien tila

conn tietokantayhteyttä ei ole olemassa

– Annetut syötteen

query select * from Asteroid;

– Tietokantaan kohdistuvat oletukset:

asteroidnumber 0, 1, 2

designation Earth, , 1998 SF36

name Earth, Ida, (tyhjä)

number 0, 234, 25143

– Metodi palautti:

ResultSet null

Virhe ERR_NO_CONNECTION_OR_ERRONEOUS_SQL

- *changeLevel(int level)* Lausekattavuuden saavuttamiseksi metodi testataan tapauksessa, jossa kaikki sujuu niin kuin piti; ja tapauksissa, joissa yhteys katkeaa tai käyttäjätasoksi annetaan epäkelvollinen arvo.

– Luokan muuttujien tila

conn tietokantayhteys on olemassa

– Annetut syötteen

level USER_ADMIN

– Metodi palautti:

true

– Luokan muuttujien tila

conn tietokantayhteys on olemassa

– Annetut syötteen

level 54

– Metodi palautti:

false

virhe ERR_WRONG_PARAM_VALUE;

– Luokan muuttujien tila

conn tietokantayhteyttä ei ole olemassa, eikä sitä saada luotua

- Annetut syötteet

level USER_ADMIN

- Metodi palautti:

false

Virhe ERR_NO_CONNECTION

- *resetLevel()* Metodi testataan lausekattavasti testaamalla tapaus, jolloin yhteys saadaan muodostettua; ja tapaus, jolloin sitä ei saada muodostettua.

Toimii pääpiirteissään kuten `changeLevel`. Testattiin lausekattavasti.

- *isOk()* Metodi testataan tavallisessa tapauksessa ja tapauksessa, jossa yhteysmuuttujan (`conn`) arvo on null. Jälkimmäinen tapaus voidaan testata silloin, kun yhteyttä ei ole muodostettuna. Testaamatta jää ulkoisista syistä johtuva `catch`-osio.

- Luokan muuttujien tila

conn tietokantayhteys on olemassa

- Metodi palautti:

true

- Luokan muuttujien tila

conn tietokantayhteyttä ei ole olemassa

- Metodi palautti:

false

Virhe ERR_NO_CONNECTION

- *createConnection(int level)* Metodi muodostuu kahdesta `try-catch`-osiosta. Lausekattava testaaminen vaatii näin ollen myös virhetilanteiden (`exception`) testaamista. Testikierroksia tulee kolme: tavallinen suoritus; suoritus, jolloin tietokanta-ajuria ei löydy; ja suoritus, jolloin yhteyttä ei saada syntymään muiden syiden kuin tietokanta-ajurin puuttumisen takia. Ensimmäinen virhetapaus saadaan testattua poistamalla tietokanta-ajuri `CLASSPATH`-ympäristömuuttujasta. Jälkimmäinen virhetapaus voidaan testata siten, että tietokantayhteyttä yritetään käynnistää väärästä osoitteesta, väärällä portilla tai väärillä käyttäjätunnuksilla. Yleisesti ottaen koko metodin testaus tapahtuu luokan ilmentymän luonnin yhteydessä. Konstruktori kutsuu `createConnectionia`.

- Metodi palautti:

true

- Tietokanta-ajuria ei löydy

- Metodi palautti:

false

Virhe ERR_NO_DB_DRIVER_FOUND

- Luokan muuttujien tila
dbServer Virheellinen osoite
- Metodi palautti:
false
Virhe ERR_NO_CONNECTION

- *getErrorMessages()* Metodi suoritetaan aina lausekattavasti.
Testattiin lausekattavasti.
- *finalize()* Metodi suoritetaan luokan ilmentymän automaattisen tuhoamisen yhteydessä. Metodi kutsuu ainoastaan `closeConnection`-metodia, joka tulee testattua tässä samassa yhteydessä.
Testattiin lausekattavasti.

3.12.2 Yksityiset metodit

- *closeConnection()* Metodi suoritetaan aina lausekattavasti catch-osiossa olevaa yhtä testausinformaatiota antavaa koodiriviä lukuun ottamatta. Tällainen vastaava rivi tulee testattua eräissä muissa metodeissa ja sen toiminta ei ole oleellista metodin toiminnan kannalta, joten rivi jää testauksen ulkopuolelle.
Testattiin lausekattavasti.

3.13 LightcurveHandler

3.13.1 Yksityiset metodit

Yksityisten metodien testaamista varten metodeille tehdään julkiset wrapper-metodit, joiden kautta yksityisiä metodeja pystytään käyttämään myös testausluokasta. `HandlerDataPoint`-apuluokka testataan kopioimalla koko luokka omaan luokkaansa, jossa testaus suoritetaan.

- *getLightcurveByID*
Lausekattavan testauksen perustapauksena on tilanne, jossa annetulla valokäyrätunnuksella oleva havainto löytyy järjestelmästä. Koska tätä metodia käytetään vain, mikäli valokäyrä on muuttunut kesken hakua, ei tilannetta, jossa valokäyrää ei järjestelmästä löydy, pitäisi esiintyä kertaakaan.
 1. Annetut syötteen:
 - LightcurveID** 1
 2. Tietokantaan kohdistuvat oletukset:
 - Valokäyrän ID 1 löytyy tietokannasta. Valokäyrällä on 5 havaintopistettä.
 3. Metodi palautti:
 - HandlerDataPoint**-taulukon jossa 5 alkoita.

Havaintopiste 1 Time: Fri Feb 02 03:32:42 EET 2001

Magnitude: 18.344

Havaintopiste 2 Time: Fri Feb 02 03:42:47 EET 2001

Magnitude: 18.371

Havaintopiste 3 Time: Fri Feb 02 03:51:25 EET 2001

Magnitude: 18.158

Havaintopiste 4 Time: Fri Feb 02 04:10:08 EET 2001

Magnitude: 17.949

Havaintopiste 5 Time: Fri Feb 02 04:25:59 EET 2001

Magnitude: 17.833

1. Annetut syötteet:

LightcurveID -312422

2. Tietokantaan kohdistuvat oletukset:

Valokäyrän ID -312422 ei ole olemassa

3. Metodi palautti:

null-viitteen

- *parseLightcurveDataPoint*

Testauksen perustapauksena voidaan pitää aineistoa, josta löytyy sarakkeet aika, error sekä jokin filtti. Toinen keskeinen testaustapaus on Atlas-syöttimen käyttöön liittyvä, sillä tällöin joitain lauseita ei suoriteta laisinkaan. Vaikkakaan lausekattavassa testauksessa ei normaalisti tarvitsisikaan tällaisia tilanteita testata erikseen, on ero tässä koettu niin suureksi, että yksi perustestitapaus, jossa tiedot ovat oikeita, suoritetaan myös Atlas-syötin simuloituna. Seuraavassa on lueteltu perustapauksessa käytetyt syötteet:

1. Annetut syötteet:

columns {"0", "1", "2"}

data "253 -0.52353 352\n276 0.5435 323\n773 0 7000\n"

asteroidID 2

zeroTime 100

zeroMagnitude 10

dataTimeDivider 1

atlasImported false

2. Tietokantaan kohdistuvat oletukset:

Asteroidin #2 tulee löytyä tietokannasta.

3. Metodi palautti:

HandlerDataPoint-taulukon , jossa 3 alkioita. Alkioissa seuraavat tiedot:

Datapoint 0, aika 353

Datapoint 0, magnitude 362

Datapoint 0, virhe -0.52353

Datapoint 1, aika 276

Datapoint 1, magnitude 333

Datapoint 1, virhe 0.5435

Datapoint 2, aika 873

Datapoint 2, magnitude 7010

Datapoint 2, virhe 0

Havaintopisteiden koordinaatit sisältävät informaatiota.

Vastaava perustapaus tilanteessa, jossa kyseessä on Atlas-syötetty valokäyrä. Yksinkertaisuuden vuoksi tässä ilmoitettu vain muutokset edelliseen testitapaukseen.

1. Annetut syötteen:

atlasImported true

2. Tietokantaan kohdistuvat oletukset:

Asteroidin #2 tulee löytyä tietokannasta.

3. Metodi palautti:

HandlerDataPoint-taulukon , jossa 3 alkioita. Alkioissa seuraavat tiedot:

Datapoint 0, aika 353

Datapoint 0, magnitude 362

Datapoint 0, virhe -0.52353

Datapoint 1, aika 276

Datapoint 1, magnitude 333

Datapoint 1, virhe 0.5435

Datapoint 2, aika 873

Datapoint 2, magnitude 7010

Datapoint 2, virhe 0

Havaintopisteiden koordinaatit eivät sisällä informaatiota.

Perustilanteen lisäksi testataan metodin toimintaa seuraavissa tilanteissa:

- Null-viitteet

Lausekattavuuden saamiseksi metodia tulee testata myös tilanteissa, jokin annetuista olio-syötteistä (data, columnString) on null-viittaus. Tällöin metodin suorituksen pitäisi keskeytä virheeseen, ja metodi palauttaa null-viitteen.

1. Annetut syötteen:

Tapaus 2 data null

Tapaus 3 columnString null

2. Metodi palautti:

null-viitteen kaikissa tapauksissa.

Virheen TahitiLibrary.ERR_MANDANTORY_DATA_MISSING

- Käytetyt sarakkeet

Koska metodin toiminta eroaa normaalista hiukan, riippuen siitä sisältääkö annettu data error-saraketta, testaan metodin toimintaa molemmissa tilanteissa. Lisäksi lausekattavuuden saavuttamiseksi tulee testata tilanne, jossa time-sarake, tai varsinainen havaintotieto puuttuu. Mikäli jompikumpi sarakkeista puuttuu, tulee metodin suorituksen keskeyttää virheeseen, ja metodi palauttaa null-viitteen.

1. Annetut syötteet:

```
columns {"0", "2"}
data "253 352\n276 323\n773 7000\n"
```

2. Metodi palautti:

```
Datapoint 0, aika 353
Datapoint 0, magnitude 364
Datapoint 1, aika 376
Datapoint 1, magnitude 333
Datapoint 2, aika 873
Datapoint 2, magnitude 7010
```

1. Annetut syötteet:

```
columns {"0", "1", "2"}
data "253 352\n276 323\n773 7000\n"
```

2. Metodi palautti:

```
null-viitteen
Virhe TahitiLibrary.ERR_NUMBERFORMAT
```

1. Annetut syötteet:

```
Tapaus1, Ei dataa: columns "0", "1"
Tapaus2, Ei aikaa: columns "1", "2"
```

2. Metodi palautti:

```
null-viitteen
Virhe, Tapaus 1 TahitiLibrary.ERR_NO_DATAPOINT_DATA
Virhe, Tapaus 2 TahitiLibrary.ERR_NO_EPOCH_IN_DATA
```

- Rataelementtien haun epäonnistuminen

Metodin testausta varten pitää myös järjestää tilanne, jossa käsiteltävälle asteroidille ei löydy oletusratatietoja. Tämä onnistuu helpoiten käyttämällä sellaista asteroidinumeroa, jota järjestelmästä ei löydy. Mikäli sopivaa rataelementtiä ei löydy, keskeytyyn metodin suoritus virheeseen. Tällöin metodi palauttaa null-viitteen.

1. Annetut syötteet:

AsteroidID -2412

2. Metodi palautti:

null-viitteen kaikissa tapauksissa.

Virheen TahitiLibrary.ERR_FETCHING_TRAJECTORY

- Vääränlaista dataa

Lausekattavuuden saavuttamiseksi metodia pitää myös testata tilanteissa, joissa kaikkia annettuja sarakkeita ei ole täytetty, tai niihin on laitettu vääränmuotoista tietoa, kuten tekstiä. Riittää testata tilanteet, joissa kaikissa sarakkeissa jokin edellämainituista ehdoista täyttyy, sillä tyhjät sarakkeet ja vääränmuotoinen data tarkistetaan samalla kerralla. Myös näissä tapauksissa metodin suoritus päättyy virheeseen ja metodi palauttaa null-viitteen.

1. Annetut syötteen:

columns {"432", "1"}

2. Metodi palautti:

null-viitteen

Virhe TahitiLibrary.ERR_DATAFORMAT

1. Annetut syötteen:

columns {"0", "2", "3"}

data "1 4324\n2 523\n43 325"

2. Metodi palautti:

null-viitteen

Virhe TahitiLibrary.ERR_NUMBERFORMAT

1. Annetut syötteen:

columns {"0", "2", "3"}

data "1 32 43\n3 35"

2. Metodi palautti:

null-viitteen

Virhe TahitiLibrary.ERR_NUMBERFORMAT

- *getSearchWhere*

Koska *getSearchWhere*-apumetodi on hyvin suoraviivainen, ei sen testaukseenkaan tarvita mitään erityisen monimutkaisia syötteitä. Lausekattava testiaineisto saadaan aikaan yksinkertaisesti tilanteessa, jossa kaikki annettavan query-kontainerin kentät ovat täytettyinä. Tämän lisäksi on syytä kuitenkin testata myös tilanne, jossa ainakin jotkin annettavista kentistä ovat tyhjiä merkkijonoja, ja jotkin null-viittauksia. Näissä tapauksissa metodi vain jättää kyseiset kentät huomiotta SQL-where lauseketta muodostettaessa.

1. Annetut syötteen:

AsteroidName asteroidinNimi

AsteroidDesignation AsteroidDesignationtekst
AsteroidNumber 123
ObservationTimeStart 01.01.2002
ObservationTimeEnd 01.01.2003
Observer Arska
Location Park-pub
MinimumLightcurves 1
MinimumPhaseAngle -90
MaximumPhaseAngle 90
MinimumDistanceFromSun 100
MaximumDistanceFromSun 1000
MinimumDistanceFromEarth 500
MaximumDistanceFromEarth 5000
MinimumLongitude -1
MaximumLongitude 5
MinimumLatitude -6
MaximumLatitude 7
Detector CDD, Koff-III, ja semmonenkin vielä
AbsolutePhotometry true
Information Tärkeää infoa
Filters 1, 2, 3
MinimumDataPoints 3
LighttimeCorrected true

2. Metodi palautti:

SQL-lauseen (AsteroidName like 'AND ObservationTime >= to_timestamp('2452275.4166666665', 'J') AND ObservationTime <= to_timestamp('2452640.4166666665', 'J') AND (Filter = 1 OR Filter = 2 OR Filter = 3) AND Longitude >= -1 AND Longitude <= 5 AND Latitude >= -6 AND Latitude <= 7 AND PhaseAngle >= -90 AND PhaseAngle <= 90 AND DistanceFromSun >= 100 AND DistanceFromSun <= 1000 AND DistanceFromEarth >= 500 AND DistanceFromEarth <= 5000 AND Observers like '

1. Annetut syötteet:

Detector <tyhjä taulukko>
LighttimeCorrected null
AbsolutePhotometry <tyhjä String>

2. Metodi palautti:

SQL-lauseen (AsteroidName like 'to_timestamp('2452275.4166666665', 'J') AND ObservationTime <= to_timestamp('2452640.4166666665', 'J') AND (Filter = 1 OR Filter = 2 OR Filter = 3) AND Longitude >= -1 AND Longitude <= 5 AND

Latitude >= -6 AND Latitude <= 7 AND PhaseAngle >= -90 AND PhaseAngle <= 90 AND DistanceFromSun >= 100 AND DistanceFromSun <= 1000 AND DistanceFromEarth >= 500 AND DistanceFromEarth <= 5000 AND Observers like
,

- *HandlerDataPoint*-apuluokka

Yksityisen *HandlerDataPoint*-apuluokan metodien testaus on hyvin suoraviivaista. Metodien *addRow* testiaineistoon tarvitaan lisäys oikeilla tiedoilla, sekä lisäykset, joissa jokin lisätävistä tiedoista puuttuu. Lisäksi tarvitaan testata tilanne, jossa annettu havainto sisältää virhetiedon ja tilanne, jossa virhetietoa ei ole, eli virhearvona annetaan `Double.NaN`. Myös tilanne, jossa täyteen *HandlerDataPoint*-olioon yritetään lisätä riviä. Muiden, tiedon noutamiseen käytettävien metodien testaus on helppoa; tarvitsee vain testata metodien toiminta oikealla syötteen, sekä sellaiselle rivinumerolla, jota on joko alle nollan, tai suurempi kuin *HandlerDataPoint*-olion sisältämä rivimäärä.

HandlerDataPoint-luokan toimivuus on testattu *parseLightcurveDataPoint*-metodin yhteydessä. Tämän vuoksi apuluokan metodeita ei testata erikseen.

3.13.2 Normaalit metodit

SQL-Operaatioita suorittavien metodien testauksessa *insert*, *update* ja *delete*-operaatiot keskeytetään *DBControlista* tehdystä muunnelmassa ja muodostetut SQL-lausekkeet tulostetaan ruudulle. Normaalit SQL-kyselyt suoritetaan normaalisti ja tieto haetaan kannasta.

- *getAsteroids*

Koska varsinaisen työn *getAsteroids*-metodia käytettäessä tekee yksityinen *getSearchWhere*-metodi, on *getAsteroids* metodi modulitestausta hyvin yksinkertaista. Testauksessa täytyy koikeilla antaa *minLightcurves*-muuttujalle ei numeerinen arvo, sekä tyhjää syötettä täytyy testata erikseen. Mikäli *minimumLightcurves* on tyhjä, pitäisi metodin korvata annettu arvo luvulla 1.

1. Annetut syötteen:

MinimumLightcurves <tyhjä String>

2. Metodi palautti:

ApiAsteroid-taulukon , jossa 2 alkiota. Yksinkertaisuuden vuoksi näiden alkoiden sisältöä ei ole listattu tähän.

1. Annetut syötteen:

MinimumLightcurves 5439034

2. Tietokantaan kohdistuvat oletukset:

Yhdestäkään asteroidista ei ole 5439034 havaintoa.

3. Metodi palautti:

ApiAsteroid-taulukon , jossa 0 alkiota.

- *getLightcurves*

Myös `getLightcurves` metodi käyttää yksityistä `getSearchWhere`-metodia avukseen, mutta metodi on muuten `getAsteroidsia` monimutkaisempi. Metodin toiminta on kuitenkin hyvin suoraviivaista, joten testitapausten määrä ei kuitenkaan ole kauhean suuri. Minimum lightcurves -muuttujalle tarvitaan vastaavanlainen testaus kuin `getAsteroids`-metodissakin.

1. Annetut syötteen:

MinimumLightcurves <tyhjä String>

2. Metodi palautti:

ApiLightcurve-taulukon , jossa 3 alkiota. Taulukossa olevat valokäyrät sisältävät seuraavan määrän havaintopisteitä:

Valokäyrä 1 5

Valokäyrä 2 3

Valokäyrä 3 3

Yksinkertaisuuden vuoksi näiden alkoiden sisältöä ei ole listattu tähän.

1. Annetut syötteen:

MinimumLightcurves 5439034

2. Tietokantaan kohdistuvat oletukset:

Yhdestäkään asteroidista ei ole 5439034 havaintoa.

3. Metodi palautti:

ApiLightcurve-taulukon , jossa 0 alkiota.

Normaalien tilanteiden lisäksi täytyy myös testata tilannetta, jossa valokäyrän tiedot muuttuvat otsikko- ja datapistehakujen välissä. Tällöin suoritetaan haku yksityisen `getLightcurveByID`-metodin avulla. Tämä saavutetaan laittamalla metodiin rivi, joka pyytää käyttäjältä rivivaihtoa jatkaakseen. Tällä välin kannassa olevan havainnon versionumero käydään päivittämässä.

Tämä tilanne testattiin, mutta koska testaaminen vaati keskeyttävän käskyn lisäämistä `LightcurveHandler`-luokkaan, ei testitapausta voida samalla tavalla dokumentoida, kuin muita tapauksia. Metodi toimi testissä oletetulla tavalla.

- *insertLightcurves*

Valokäyrien lisäämisen testauksen perustapaukseksi otetaan sellaisen havainnon lisäys, jossa jokainen tietokenttä sisältää jonkinlaista informaatiota. Tässä ensimmäisessä testitapauksessa vain Atlas-tiedostojen siirtämiseen tarkoitettujen kenttien tulee olla null-vitteitä. Toinen tarvittava perustapaus on Atlas-tyyppisen valokäyrän tallentaminen järjestelmään. Myös Atlas-tapauksessa kaikkien kenttien tulee olla täytettyinä validilla datalla.

1. Annetut syötteen:

LightcurveID null

AsteroidNumber 25143

AsteroidName <tyhjä String>

AsteroidDesignation null
ObservingSite Kallen paari
Observer Masa
ZeroTime 1000
UnitOfTime 1
AbsolutePhotometry true
ZeroMagnitude 100
LightTimeCorrected true
PhotometricSystem tyhjä pullo
Detector Paha silmä
TimeStandard Ai etta aikastaandaardi? huh!
Reference Medical science, 3/457
Information Tis is veri impootant
Version 234
Columns {"0", "2", "2"}
Data "2 63643.345 52.23\n3 25234 12.324\n4 42.4234 97653.4"
atlasEarth null
atlasAsteroid null
validTime -1
atlasAspect null

2. Tietokantaan kohdistuvat oletukset:

Asteroidi 25143 löytyy järjestelmästä.

3. Metodi palautti:

SQL-lauseet INSERT INTO Observation (ObservationNumber, Information, Location, Observers, Reference, Submitter, Target) VALUES (40, 'Tis is veri impootant', 'Kallen paari', 'Masa', 'Medical science, 3/457', 1,2)
 INSERT INTO Lightcurve (ObservationNumber, absolutePhotometry, LightTimeCorrected, PhotometricSystem, Detector, TimeStandard, Filter, DefaultTrajectory, DefaultEarthTrajectory,PhaseAngle, DistanceFromSun, DistanceFromEarth, Longitude, Latitude) VALUES (40, TRUE ,TRUE ,'tyhjä pullo', 'Paha silmä', 'Ai etta aikastaandaardi? huh!', 2, (default_trajectory(2,to_timestamp('1000.0833333334886', 'J'))), (default_trajectory(0,to_timestamp('1000.0833333334886', 'J'))), 42.86947076017127, 0.9932140939925469, 1.150775737440672, 63.71084953145576, 1.2738486494099663)
 INSERT INTO DataPoint (Lightcurve, PointNumber, MeasuringTime, Magnitude, AsteroidX, AsteroidY, AsteroidZ, EarthX, EarthY, EarthZ) VALUES (40,0, to_timestamp('1000.0833333334886', 'J'), 63743.345, -0.41521157274485765, 1.393825521212935, 0.025583072717663927, -0.9247658235431878, 0.3623291433377973, 1.525179234836779E-7)
 INSERT INTO DataPoint (Lightcurve, PointNumber, MeasuringTime, Magnitude, AsteroidX, AsteroidY, AsteroidZ, EarthX, EarthY, EarthZ) VALUES (40,1,


```
to_timestamp('1000.125', 'J'), 25334.0, -0.41569051692056924, 1.3935280290981056,
0.0255937942385085, -0.925039066431075, 0.36166124434783875, 1.5189972952178664E-
7)
```

```
INSERT INTO DataPoint (Lightcurve, PointNumber, MeasuringTime, Magnitu-
de, AsteroidX, AsteroidY, AsteroidZ, EarthX, EarthY, EarthZ) VALUES (40,2,
to_timestamp('1000.1666666665114', 'J'), 142.42340000000002, -0.4161693916890852,
1.3932303042920355, 0.025604511485839036, -0.9253118242980993, 0.3609931557396154,
1.512814559233872E-7)
```

```
INSERT INTO Observation (ObservationNumber, Information, Location, Obser-
vers, Reference, Submitter, Target) VALUES (41, 'Tis is veri impootant', 'Kallen
paari', 'Masa', 'Medical science, 3/457', 1,2)
```

```
INSERT INTO Lightcurve (ObservationNumber, absolutePhotometry, LightTimeCor-
rected, PhotometricSystem, Detector, TimeStandard, Filter, DefaultTrajectory, De-
faultEarthTrajectory,PhaseAngle, DistanceFromSun, DistanceFromEarth, Longitu-
de, Latitude) VALUES (41, TRUE ,TRUE ,'tyhjä pullo', 'Paha silmä', 'Ai etta ai-
kastaandaardi? huh!', 2, (default_trajectory(2,to_timestamp('1000.0833333334886',
'J'))), (default_trajectory(0,to_timestamp('1000.0833333334886', 'J'))), 42.86947076017127,
0.9932140939925469, 1.150775737440672, 63.71084953145576, 1.2738486494099663)
```

```
INSERT INTO DataPoint (Lightcurve, PointNumber, MeasuringTime, Magnitu-
de, AsteroidX, AsteroidY, AsteroidZ, EarthX, EarthY, EarthZ) VALUES (41,0,
to_timestamp('1000.0833333334886', 'J'), 152.23, -0.41521157274485765, 1.393825521212935,
0.025583072717663927, -0.9247658235431878, 0.3623291433377973, 1.525179234836779E-
7)
```

```
INSERT INTO DataPoint (Lightcurve, PointNumber, MeasuringTime, Magnitu-
de, AsteroidX, AsteroidY, AsteroidZ, EarthX, EarthY, EarthZ) VALUES (41,1,
to_timestamp('1000.125', 'J'), 112.324, -0.41569051692056924, 1.3935280290981056,
0.0255937942385085, -0.925039066431075, 0.36166124434783875, 1.5189972952178664E-
7)
```

```
INSERT INTO DataPoint (Lightcurve, PointNumber, MeasuringTime, Magnitu-
de, AsteroidX, AsteroidY, AsteroidZ, EarthX, EarthY, EarthZ) VALUES (41,2,
to_timestamp('1000.1666666665114', 'J'), 97753.4, -0.4161693916890852, 1.3932303042920355,
0.025604511485839036, -0.9253118242980993, 0.3609931557396154, 1.512814559233872E-
7)
```

1. Annetut syötteet:

```
LightcurveID null
AsteroidNumber 25143
AsteroidName <tyhjä String>
AsteroidDesignation null
ObservingSite Kallen paari
Observer Masa
ZeroTime 1000
UnitOfTime 1
```

```

AbsolutePhotometry true
ZeroMagnitude 100
LightTimeCorrected true
PhotometricSystem tyhjä pullo
Detector Paha silmä
TimeStandard Ai etta aikastaandaardi? huh!
Reference Medical science, 3/457
Information Tis is veri impootant
Version 234
Columns {"0", "2", "2"}
Data "2 63643.345 52.23\n3 25234 12.324\n4 42.4234 97653.4"
atlasEarth {-0.297911, -0.969713, -0.000015, 0.016165, -0.005116, 0.000000 }
atlasAsteroid {-1.330945, -2.674771, -0.058888, 0.008685, -0.004387, 0.000028}
validTime 2445854.6
atlasAspect {2.9882, 1.9944, 4.80, 238.80, -1.70}

```

2. Tietokantaan kohdistuvat oletukset:

Asteroidi 25143 löytyy järjestelmästä.

3. Metodi palautti:

```

SQL-lauseet INSERT INTO Observation (ObservationNumber, Information, Location, Observers, Reference, Submitter, Target) VALUES (42, 'Tis is veri impootant', 'Kallen paari', 'Masa', 'Medical science, 3/457', 1,2)
INSERT INTO Lightcurve (ObservationNumber, absolutePhotometry, LightTimeCorrected, PhotometricSystem, Detector, TimeStandard, Filter, PhaseAngle, DistanceFromSun, DistanceFromEarth, Longitude, Latitude) VALUES (42, TRUE, TRUE, 'tyhjä pullo', 'Paha silmä', 'Ai etta aikastaandaardi? huh!', 2, 4.8, 2.9882, 1.9944, 238.8, -1.7)
INSERT INTO DataPoint (Lightcurve, PointNumber, MeasuringTime, Magnitude, AsteroidX, AsteroidY, AsteroidZ, EarthX, EarthY, EarthZ) VALUES (42,0, to_timestamp('1000.0833333334886', 'J'), 63743.345, -21234.892422250003, 10722.901993616668, -68.51481446666666, -39521.37117291666, 12506.905994266666, -1.5E-5)
INSERT INTO DataPoint (Lightcurve, PointNumber, MeasuringTime, Magnitude, AsteroidX, AsteroidY, AsteroidZ, EarthX, EarthY, EarthZ) VALUES (42,1, to_timestamp('1000.125', 'J'), 25334.0, -21234.892422250003, 10722.901993616668, -68.51481446666666, -39521.37117291666, 12506.905994266666, -1.5E-5)
INSERT INTO DataPoint (Lightcurve, PointNumber, MeasuringTime, Magnitude, AsteroidX, AsteroidY, AsteroidZ, EarthX, EarthY, EarthZ) VALUES (42,2, to_timestamp('1000.1666666665114', 'J'), 142.42340000000002, -21234.892422250003, 10722.901993616668, -68.51481446666666, -39521.37117291666, 12506.905994266666, -1.5E-5)
INSERT INTO Observation (ObservationNumber, Information, Location, Observers, Reference, Submitter, Target) VALUES (43, 'Tis is veri impootant', 'Kallen paari', 'Masa', 'Medical science, 3/457', 1,2)

```

```
INSERT INTO Lightcurve (ObservationNumber, absolutePhotometry, LightTimeCorrected, PhotometricSystem, Detector, TimeStandard, Filter, PhaseAngle, DistanceFromSun, DistanceFromEarth, Longitude, Latitude) VALUES (43, TRUE, TRUE, 'tyhjä pullo', 'Paha silmä', 'Ai etta aikastaandaardi? huh!', 2, 4.8, 2.9882, 1.9944, 238.8, -1.7)
```

```
INSERT INTO DataPoint (Lightcurve, PointNumber, MeasuringTime, Magnitude, AsteroidX, AsteroidY, AsteroidZ, EarthX, EarthY, EarthZ) VALUES (43, 0, to_timestamp('1000.0833333334886', 'J'), 152.23, -21234.892060375, 10722.901810825, -68.5148133, -39521.370499375, 12506.9057811, -1.5E-5)
```

```
INSERT INTO DataPoint (Lightcurve, PointNumber, MeasuringTime, Magnitude, AsteroidX, AsteroidY, AsteroidZ, EarthX, EarthY, EarthZ) VALUES (43, 1, to_timestamp('1000.125', 'J'), 112.324, -21234.892060375, 10722.901810825, -68.5148133, -39521.370499375, 12506.9057811, -1.5E-5)
```

```
INSERT INTO DataPoint (Lightcurve, PointNumber, MeasuringTime, Magnitude, AsteroidX, AsteroidY, AsteroidZ, EarthX, EarthY, EarthZ) VALUES (43, 2, to_timestamp('1000.1666666665114', 'J'), 97753.4, -21234.892060375, 10722.901810825, -68.5148133, -39521.370499375, 12506.9057811, -1.5E-5)
```

Näiden perustapausten lisäksi myös seuraavanlaiset poikkeamat tulee testata:

- Pakollisten tietojen puuttuminen

Koska jokaiseen havaintoon liittyy pakollisia tietoja, joiden on oltava täytettyinä, täytyy metodin toimintaa kokeilla tilanteissa, joissa järjestelmällisesti jokin pakollinen tieto puuttuu. Lisäksi joidenkin tiedoille (UnitOfTime tulee löytyä TahitiLibrarystä) tehtävien tarkistusten toimivuus tulee testata. Näissä tilanteissa suorituksen pitäisi keskeytyä virheeseen, ja metodin palauttaa false.

1. Annetut syötteen:

Tapaus 1, ObservingSite <tyhjä String>

Tapaus 2, Observer <tyhjä String>

Tapaus 3, AbsolutePhotometry null

Tapaus 4, LightTimeCorrected null

2. Metodi palautti:

false

Virhe TahitiLibrary.ERR_MANDANTORY_DATA_MISSING

1. Annetut syötteen:

Tapaus 1, UnitOfTime 42432

Tapaus 2, ZeroTime ei oo numero

Tapaus 3, ZeroMagnitude pip pip piip

2. Metodi palautti:

false

Virhe TahitiLibrary.ERR_NUMBERFORMAT

- Kohteen puuttuminen järjestelmästä

Jokaisella järjestelmään syötettävällä valokäyrällä tulee olla kohde, joka on talletettu järjestelmään. Lausekattavuuden aikaansaamiseksi metodia testattaessa tulee myös testata tilanne, jossa kohteena olevaa asteroidia ei löydy järjestelmän tietokannasta. Lisäksi tarvitaan myös testitapaus, jossa kohteena on asteroidi, jonka sisäinen tunnus on 0, eli maa. Tällaisia havaintoja ei tietokantaan haluta, ja metodin suoritus päättyy tällöin virheeseen.

1. Annetut syötteet:

Tapaus 1, AsteroidNumber 23553453

Tapaus 2, AsteroidNumber 0

2. Metodi palautti:

false

Virhe TahitiLibrary.ERR_asteroidid

- Ongelmat datapisteiden kanssa

Lausekattavuuden saavuttamiseksi metodia tulee testata myös sellaisilla syötteillä, joista datapistetietoja ei pystytä muodostamaan. Tällaisissa tilanteissa yksityinen `parseLightcurveDataPoint`-metodi palauttaa null, joten tällainen tilanne pitää myös pystyä simuloimaan. Mikäli näin tapahtuu, keskeytyy metodin suoritus.

1. Annetut syötteet:

data <tyhjä String>

2. Metodi palautti:

false

Virhe TahitiLibrary.ERR_NUMBERFORMAT

- Vääränlaiset Atlas-tiedot

Metodin toimintaa pitää myös testata tilanteessa, jossa Atlas-siirron yhteydessä annettavat paikkatiedot ovat vääränpituisia. Tällöin suorituksen tulisi keskeytyä.

1. Annetut syötteet:

atlasAspect new double[1]

2. Tietokantaan kohdistuvat oletukset:

Asteroidi 25143 löytyy järjestelmästä.

3. Metodi palautti:

false

Virhe TahitiLibrary.ERR_DATAFORMAT

- *changeLightcurve*

Kuten `insertLightcurve`-metodissa, myös muutoksen perustapauksena pidetään tapausta, jossa kaikki tietokentät on täytettyinä, ja sisältävät jotain muuta tietoa, kuin tietokannassa ennen muutosta on. (Myös datapisteiden tietojen tulee muuttua) Näihin kaikkiin tietoihin

sisältyy myös oletamus, että myös valokäyräpisteiden havaintoajat muuttuvat siinä määrin, että myös datapisteisiin liittyvät koordinaatit muuttuvat, jolloin myös havaintoon liittyvät paikkatieto-ominaisuudet (PhaseAngle jne) vaihtuvat. Lisäksi lausekattavuus vaatii, että tekstikenttien AbsolutePhotometry ja LightTimeCorrected arvoina on "true". Koska versio-numeroilla on merkitystä vasta, kun muodostettuja SQL-lauseita käytetään aidosti kantaa vasten, ei niinhin testauksessa tarvitse kiinnittää huomiota. Perustapaus testattiin seuraavalla syötteellä:

1. Annetut syötteen:

```

LightcurveID 2
AsteroidNumber 25143
AsteroidName <tyhjä String>
AsteroidDesignation null
ObservingSite Kallen paari
Observer Masa
ZeroTime 1000
UnitOfTime 1
AbsolutePhotometry true
ZeroMagnitude 100
LightTimeCorrected true
PhotometricSystem tyhjä pullo
Detector Paha silmä
TimeStandard Ai etta aikastaandaardi? huh!
Reference Medical science, 3/457
Information Tis is veri impootant
Version 234
Columns {"0", "2"}
Data "2 63643.345\n3 25234\n4 42.4234"
atlasEarth null
atlasAsteroid null
validTime -1
atlasAspect null

```

2. Tietokantaan kohdistuvat oletukset:

Valokäyrä #2 löytyy järjestelmästä, ja sen kohteena on jokin muu kuin asteroidi 25143.
Lisäksi uuden kohdeasteroidin tulee löytyä tietokannasta.

3. Metodi palautti:

```

SQL-lauseet UPDATE Observation SET Version = 1, Information = 'Tis is veri impootant', Location = 'Kallen paari', Observers = 'Masa', Reference = 'Medical science, 3/457', Target = 2 WHERE ObservationNumber = 2 AND Version = 0
UPDATE Lightcurve SET DefaultTrajectory = default_trajectory(2,to_timestamp('1000.0833333334'J)), DefaultEarthTrajectory = default_trajectory(0, to_timestamp('1000.0833333334886',

```

```
'J') ), AbsolutePhotometry = true , PhotometricSystem = 'tyhjä pullo' , Detector
= 'Paha silmä' , TimeStandard = 'Ai etta aikastaandaardi? huh!' , Filter = 2 ,
PhaseAngle = 42.86947076017127 , DistanceFromSun = 0.9932140939925469 , Dis-
tanceFromEarth = 1.150775737440672 , Longitude = 63.71084953145576 , Latitude
= 1.2738486494099663 WHERE ObservationNumber = 2
UPDATE DataPoint SET MeasuringTime = to_timestamp('1000.0833333334886',
'J') , Magnitude = 63743.345 , Error = NULL , AsteroidX = -0.41521157274485765,
AsteroidY = 1.393825521212935, AsteroidZ = 0.025583072717663927, EarthX = -
0.9247658235431878, EarthY = 0.3623291433377973, EarthZ = 1.525179234836779E-
7 WHERE Lightcurve = 2 AND PointNumber = 0
UPDATE DataPoint SET MeasuringTime = to_timestamp('1000.125', 'J') , Magni-
tude = 25334.0 , Error = NULL , AsteroidX = -0.41569051692056924, AsteroidY =
1.3935280290981056, AsteroidZ = 0.0255937942385085, EarthX = -0.925039066431075,
EarthY = 0.36166124434783875, EarthZ = 1.5189972952178664E-7 WHERE Lightcur-
ve = 2 AND PointNumber = 1
UPDATE DataPoint SET MeasuringTime = to_timestamp('1000.1666666665114',
'J') , Magnitude = 142.42340000000002 , Error = NULL , AsteroidX = -0.4161693916890852,
AsteroidY = 1.3932303042920355, AsteroidZ = 0.025604511485839036, EarthX = -
0.9253118242980993, EarthY = 0.3609931557396154, EarthZ = 1.512814559233872E-
7 WHERE Lightcurve = 2 AND PointNumber = 2
```

Perustapauksen lisäksi lausekattavuuden saavuttamiseksi metodin toimintaan pitää testata seuraavissa tapauksissa:

- Pakollisten kenttien puuttuminen

Kuten lisäyksessäkin, valokäyrän muutoksessa tulee kaikki pakollisten kenttien olla täytettyinä. Lausekattavuuden saavuttamiseksi pitää metodin toimintaa kokeilla tilanteissa, joissa mikä tahansa pakollinen kenttä on joko tyhjä, tai null-viittaus. Tällöin metodin suoritus keskeytyy virheeseen ja metodi palauttaa false. Lisäksi metodin toimintaa testattiin tilanteessa, jossa jotkin kentät sisälsivät vääränmuotoista dataa.

1. Annetut syötteen:

Tapaus 1, ObservingSite <tyhjä String>

Tapaus 2, Observer <tyhjä String>

Tapaus 3, AbsolutePhotometry null

Tapaus 4, LightTimeCorrected null

2. Metodi palautti:

false

Virhe TahitiLibrary.ERR_MANDANTORY_DATA_MISSING

1. Annetut syötteen:

Tapaus 1, UnitOfTime 42432

Tapaus 2, ZeroTime ei oo numero

Tapaus 3, ZeroMagnitude pip pip piip

2. Metodi palautti:

false

Virhe TahitiLibrary.ERR_NUMBERFORMAT

- Muutoksen kohteena olevaa valokäyrää ei löydy järjestelmästä

Metodin toimintaa pitää kokeilla myös tilanteessa, jossa muutoksen kohteen, valokäyrän sisäistätunnusnumeroa ei löydy järjestelmästä. Myös tässä tilanteessa suoritus keskeytyy, ja palautetaan false.

1. Annetut syötteen:

LightcurveID -23423

2. Tietokantaan kohdistuvat oletukset:

Valokäyrää -23423 ei löydy tietokannasta.

3. Metodi palautti:

false

Virhe TahitiLibrary.ERR_LIGHTCURVEID

- Atlas-siirretty valokäyrä

Metodia pitää myös kokeilla tilanteessa, jossa muutoksen kohteena on Atlas-siirtimellä järjestelmään syötetty valokäyrä. Tällöin muuntamisen pitäisi olla mahdotonta. Samaa suoritushaaraan päästään, mikäli muutettavaa valokäyrää ei löydy järjestelmästä.

1. Annetut syötteen:

LightcurveID 3

2. Tietokantaan kohdistuvat oletukset:

Valokäyrä 3 on syötetty järjestelmään Atlas-syöttimen kautta.

3. Metodi palautti:

false

Virhe TahitiLibrary.ERR_LIGHTCURVEID

- Kohteen haku

Lausekattavuuden saavuttamiseksi myös kaikki asteroidiin liittyvät tietokentät tulee olla parametrinä annettavassa containerissa täytettyinä. Tämän lisäksi tulee testata tilanteet, joissa kohdetta ei löydy järjestelmästä, sekä tilanne, jossa havainnon kohteena on virheellisesti maa. Molemmissa tilanteissa metodin suoritus keskeytyy.

1. Annetut syötteen:

Tapaus 1, AsteroidNumber 23553453

Tapaus 2, AsteroidNumber 0

2. Tietokantaan kohdistuvat oletukset:

Asteroidia 23553453 ei löydy järjestelmästä. Asteroidi 0 tulee olla maa.

3. Metodi palautti:

false

Virhe TahitiLibrary.ERR_ASTEROIDID

- Valokäyrätietojen käsittely

Koska valokäyrämuutoksessa muutos voi koskea vain yhtä valokäyrää kerrallaan, on yksityisen `parseLightcurveDataPoints`-metodin palauttaman `HandlerDataPoint`-taulukon pituuden oltava tasan yksi. Metodien toimintaa pitää testata erikseen virheellisillä havaintoarvotiedoilla, jolloin `parseLightcurveDataPoints`-metodi palauttaa arvon null, sekä tilanteessa, jossa purettavat havaintotiedot sisältävät enemmän, kuin yhden havainnon tietoja. Tällaisessa tilanteessa metodin suoritus keskeytetään.

1. Annetut syötteen:

Tapaus 1, Columns {"0", "3", "2"}

Tapaus 1, Data "2 2352 2342\n345 325 252\n34 234 234\n43 42 432"

Tapaus 2, Columns {"0", "3"}

Tapaus 2, Data 2

2. Metodi palautti:

false

Tapaus 1, Virhe TahitiLibrary.ERR_MODIFY_DATAPOINT_COUNT

Tapaus 2, Virhe TahitiLibrary.ERR_NUMBERFORMAT

- Vapaaehtoisten arvojen poistaminen kannasta

Lausekattavuuden saavuttamiseksi metodia pitää kokeilla myös tilanteessa, jossa muutettavasta valokäyrästä on kaikki vapaaehtoiset kentät (Information, Reference, PhotometricSystem, Detector, timeStandard) ovat tyhjiä, mutta tietokannan vastaavat kentät sisältävät tietoa. Tällöin metodin tulisi muodostaa SQL-lause, joka poistaa näiden kenttien sisällön tietokannasta.

1. Annetut syötteen:

LightcurveID 2

PhotometricSystem null

Detector null

TimeStandard <tyhjä String>

Reference null

Information <tyhjä String>

2. Tietokantaan kohdistuvat oletukset:

Lightcurven #2 kaikki vapaaehtoiset kentät sisältävät informaatiota.

3. Metodi palautti:


```

SQL-lauseet UPDATE Observation SET Version = 1, Information = null, Reference
= null WHERE ObservationNumber = 2 AND Version = 0
UPDATE Lightcurve SET DefaultTrajectory = default_trajectory(2,to_timestamp('1000.0833333333334
'J')), DefaultEarthTrajectory = default_trajectory(0, to_timestamp('1000.08333333334886',
'J') ), PhotometricSystem = null , Detector = null , TimeStandard = null , Pha-
seAngle = 42.86947076017127 , DistanceFromSun = 0.9932140939925469 , Dis-
tanceFromEarth = 1.150775737440672 , Longitude = 63.71084953145576 , Latitude
= 1.2738486494099663 WHERE ObservationNumber = 2
UPDATE DataPoint SET MeasuringTime = to_timestamp('1000.08333333334886',
'J') , Magnitude = 63743.345 , Error = 52.23 , AsteroidX = -0.41521157274485765,
AsteroidY = 1.393825521212935, AsteroidZ = 0.025583072717663927, EarthX = -
0.9247658235431878, EarthY = 0.3623291433377973, EarthZ = 1.525179234836779E-
7 WHERE Lightcurve = 2 AND PointNumber = 0
UPDATE DataPoint SET MeasuringTime = to_timestamp('1000.125', 'J') , Magni-
tude = 25334.0 , Error = 12.324 , AsteroidX = -0.41569051692056924, AsteroidY =
1.3935280290981056, AsteroidZ = 0.0255937942385085, EarthX = -0.925039066431075,
EarthY = 0.36166124434783875, EarthZ = 1.5189972952178664E-7 WHERE Lightcur-
ve = 2 AND PointNumber = 1
UPDATE DataPoint SET MeasuringTime = to_timestamp('1000.1666666665114',
'J') , Magnitude = 142.42340000000002 , Error = 97653.4 , AsteroidX = -0.4161693916890852,
AsteroidY = 1.3932303042920355, AsteroidZ = 0.025604511485839036, EarthX = -
0.9253118242980993, EarthY = 0.3609931557396154, EarthZ = 1.512814559233872E-
7 WHERE Lightcurve = 2 AND PointNumber = 2

```

- Datapisteiden määrän vaihtuminen

Koska muunnetussa valokäyrässä saattaa olla eri määrä datapisteitä, kuin alkuperäisessä, täytyy metodin toimintaa testata sekä valokäyrien vähentyessä, että niiden määrän kasvaessa. Metodien toimintaa tulee myös testata tilanteissa, jossa datapisteen virhearvo muuttuu tyhjästä joksinkin luvuksi, sekä tilanteessa, jossa järjestelmässä jo oleva virhearvo poistetaan. Jo järjestelmästä löytyvän virhearvon muuttuminen testataan perustapauksen yhteydessä.

1. Annetut syötteen:

Columns {"0", "2"}

Data "33 234\n32 343\n43 343\n34 343\n"

2. Tietokantaan kohdistuvat oletukset:

Ainakin yksi muutettavan valokäyrän havaintopisteistä sisältää virhetiedon. Tehdyssä testissä tämä piste oli #3. Lisäksi muutettavassa valokäyrässä tulee olla vähemmän, kuin 4 havaintopistettä. Tehdyssä testissä niitä oli 3.

3. Metodi palautti:

```

SQL-lauseen UPDATE Observation SET Version = 1, Information = null, Reference
= null WHERE ObservationNumber = 2 AND Version = 0
UPDATE Lightcurve SET DefaultTrajectory = default_trajectory(2,to_timestamp('1001.375',
'J')), DefaultEarthTrajectory = default_trajectory(0, to_timestamp('1001.375', 'J')

```

```

), PhotometricSystem = null , Detector = null , TimeStandard = null , PhaseAngle
= 42.992957797358706 , DistanceFromSun = 0.9935573183349118 , DistanceFrom
Earth = 1.1581987793555217 , Longitude = 64.25350833512279 , Latitude =
1.282031165418034 WHERE ObservationNumber = 2
UPDATE DataPoint SET MeasuringTime = to_timestamp('1001.375', 'J') , Magni-
tude = 334.0 , AsteroidX = -0.4300261096723643, AsteroidY = 1.3844949699012772,
AsteroidZ = 0.025913438686656902, EarthX = -0.9330102581837718, EarthY =
0.34153770354192964, EarthZ = 1.3331834571819268E-7 WHERE Lightcurve = 2
AND PointNumber = 0
UPDATE DataPoint SET MeasuringTime = to_timestamp('1001.3333333334886',
'J') , Magnitude = 443.0 , AsteroidX = -0.42954929206227643, AsteroidY = 1.384799451885459,
AsteroidZ = 0.025902846725744826, EarthX = -0.9327516199140788, EarthY =
0.34221113589877405, EarthZ = 1.339387867606079E-7 WHERE Lightcurve = 2
AND PointNumber = 1
UPDATE DataPoint SET MeasuringTime = to_timestamp('1001.7916666665114',
'J') , Magnitude = 443.0 , Error = NULL , AsteroidX = -0.43479028303041956,
AsteroidY = 1.3814373139985718, AsteroidZ = 0.02601911759292124, EarthX = -
0.9355697425331426, EarthY = 0.33479359988888807, EarthZ = 1.2711014853850386E-
7 WHERE Lightcurve = 2 AND PointNumber = 2
INSERT INTO DataPoint (Lightcurve, PointNumber, MeasuringTime, Magnitu-
de, AsteroidX, AsteroidY, AsteroidZ, EarthX, EarthY, EarthZ) VALUES (2,3,
'to_timestamp('1001.4166666665114', 'J')', 443.0, -0.4305028548152453, 1.3841902545952924,
0.025902846725744826, -0.9332684077495687, 0.3408640922827399, 1.3269783483826557E-
7)

```

1. Annetut syötteet:

```
Columns {"0", "1", "2"}
```

```
Data "534 .5345 2354\n324 .543 234\n"
```

2. Tietokantaan kohdistuvat oletukset:

Ainakin yhden muutettavista havaintopisteistä tulee olla sellainen, ettei se sisällä virhetietoa. Tehdyssä testitapauksessa kumpikaan muutetuista havaintopisteistä ei sisältänyt virhetietoa. Lisäksi muutettavan valokäyrän tulee sisältää enemmän kuin kaksi havaintopistettä. Tehdyssä testissä havaintopisteitä oli 3.

3. Metodi palautti:

```
SQL-lauseen UPDATE Observation SET Version = 1, Information = null, Reference
= null WHERE ObservationNumber = 2 AND Version = 0
UPDATE Lightcurve SET DefaultTrajectory = default_trajectory(2,to_timestamp('1022.25',
'J')), DefaultEarthTrajectory = default_trajectory(0, to_timestamp('1022.25', 'J') ),
PhotometricSystem = null , Detector = null , TimeStandard = null , PhaseAngle
= 44.40113272870759 , DistanceFromSun = 0.9994080349697035 , DistanceFrom
Earth = 1.2605657199132911 , Longitude = 74.26022478268182 , Latitude =
1.3910676361120822 WHERE ObservationNumber = 2

```

```
UPDATE DataPoint SET MeasuringTime = to_timestamp('1022.25', 'J') , Magnitude = 2454.0 , Error = 0.5345 , AsteroidX = -0.6574995702245856, AsteroidY = 1.2022461083528762, AsteroidZ = 0.030601723944912, EarthX = -0.9993508032675655, EarthY = -0.010695436898205114, EarthZ = -1.7945058464481784E-7 WHERE Lightcurve = 2 AND PointNumber = 0
```

```
UPDATE DataPoint SET MeasuringTime = to_timestamp('1013.5', 'J') , Magnitude = 334.0 , Error = 0.543 , AsteroidX = 0.0, AsteroidY = 0.0, AsteroidZ = 0.0, EarthX = -0.987103151769523, EarthY = 0.13945961247248878, EarthZ = -4.8851044902388555E-8 WHERE Lightcurve = 2 AND PointNumber = 1
```

```
DELETE FROM DataPoint WHERE LightCurve = 2 AND PointNumber = 2
```

- *deleteLightcurve*

Koska valokäyrien poistaminen on kaksivaiheista, tarvitaan metodin testausta varten kaksi perustapausta. Ensimmäisessä tapauksessa valokäyrä merkitään poistetuksi ja jälkinmäisessä tapauksessa valokäyrän poistetaan järjestelmästä kokonaan. Testauksessa tulee kokeilla myös molempia tapauksia sekä syöttäjätason oikeuksilla, että ylläpitäjän oikeuksilla. Lisäksi tarvitaan testi, jossa syöttäjä yrittää poistaa jonkin muu, kuin itse järjestelmään syöttämänsä valokäyrän. Täydellisen lausekattavuuden aikaansaamiseksi poiston pitää myös jossain testitapauksessa epäonnistua.

1. Annetut syötteen:

LightcurveID 1

UserLevel 1

2. Tietokantaan kohdistuvat oletukset:

Lightcurve #1 on syöttänyt järjestelmään poistoa yrittävä käyttäjä.

3. Metodi palautti:

SQL-lauseen UPDATE Observation SET Deleted = TRUE, Version = 1 WHERE ObservationNumber = 1

1. Annetut syötteen:

LightcurveID 1

UserLevel 2

2. Tietokantaan kohdistuvat oletukset:

Lightcurve #1 on syöttänyt järjestelmään joku muu, kuin poistoa yrittävä käyttäjä

3. Metodi palautti:

SQL-lauseen UPDATE Observation SET Deleted = TRUE, Version = 1 WHERE ObservationNumber = 1

1. Annetut syötteen:

LightcurveID 3

UserLevel 2

2. Tietokantaan kohdistuvat oletukset:

Lightcurve #3 on jo merkitty poistetuksi.

3. Metodi palautti:

SQL-lauseen DELETE FROM Observation WHERE ObservationNumber = 3

1. Annetut syötteet:

LightcurveID 6346345

UserLevel 2

2. Tietokantaan kohdistuvat oletukset:

Lightcurve #6346345 ei löydy järjestelmästä

3. Metodi palautti:

false

Virhe TahitiLibrary.ERR_LIGHTCURVEID

1. Annetut syötteet:

LightcurveID 1

UserLevel 1

2. Tietokantaan kohdistuvat oletukset:

Lightcurve #1 ei ole käyttäjän syöttämä.

3. Metodi palautti:

false

Virhe TahitiLibrary.ERR_NOT_ALLOWED

- *restoreLightcurve*

Myös palautusmetodin toiminta on erittäin suoraviivaista, sillä varsinainen palautus on yksinkertainen SQL-operaatio. Lausekattavan testauksen toettamiseksi tarvitaan kaksi päätausta, onnistunut palautus sekä epäonnistuva palautus. Epäonnistuvaksi palautukseksi käy esimerkiksi yritys palauttaa sellainen valokäyrä, jota ei ole merkitty poistetuksi, tai valokäyrä, jota ei järjestelmästä löydy laisinkaan.

1. Annetut syötteet:

LightcurveID 1

2. Tietokantaan kohdistuvat oletukset:

Lightcurve #1 ei ole merkitty poistetuksi.

3. Metodi palautti:

false

Virhe TahitiLibrary.ERR_LIGHTCURVEID

1. Annetut syötteet:

LightcurveID 2345235

2. Tietokantaan kohdistuvat oletukset:

Lightcurve #2345235 ei löydy järjestelmästä.

3. Metodi palautti:

false

Virhe TahitiLibrary.ERR_LIGHTCURVEID

1. Annetut syötteen:

LightcurveID 3

2. Tietokantaan kohdistuvat oletukset:

Lightcurve #3 on merkitty poistetuksi.

3. Metodi palautti:

SQL-lauseen UPDATE Observation SET Deleted = FALSE, Version = 1 WHERE
ObservationNumber = 3

3.14 LogHandler

Luokan LogHandler testaamiseen käytetään erillistä testiluokkaa sekä lopullista tilannetta vastaavaa tietokantaa. Ilman oikeaa tietokantaa luokkaa olisi työlästä testata.

Pakkausnäkymän metodit hakevat lokimerkinnät tietyltä ajanjaksolta ja tekevät syötteistä sql-muotoisen lokimerkinnän tallettamislauseen.

- *getLogEntries*

1. Annetut syötteen:

TimeStart 29.04.2003

TimeEnd null

2. Metodi palautti:

LogEntry -containerin, jossa 17 kappaletta lokimerkintöjä. Lokimerkinnät olivat seuraavanlaisia sisällöltään (mainittu vain numero, päiväys ja tekijä):

2: Tue Apr 29 17:43:58 EEST 2003: Tahiti

3: Wed Apr 30 13:24:03 EEST 2003: Tahiti

107: Fri May 02 16:46:01 EEST 2003: Tahiti

108: Fri May 02 16:46:01 EEST 2003: Tahiti

109: Fri May 02 16:46:01 EEST 2003: Tahiti

110: Fri May 02 16:54:41 EEST 2003: Tahiti

111: Fri May 02 16:54:41 EEST 2003: Tahiti

112: Fri May 02 16:54:41 EEST 2003: Tahiti

113: Fri May 02 16:55:53 EEST 2003: Tahiti

114: Fri May 02 16:55:53 EEST 2003: Tahiti

115: Fri May 02 16:55:53 EEST 2003: Tahiti

116: Fri May 02 16:56:52 EEST 2003: Tahiti

117: Fri May 02 16:56:52 EEST 2003: Tahiti

118: Fri May 02 16:56:52 EEST 2003: Tahiti
 119: Fri May 02 17:04:20 EEST 2003: Tahiti
 120: Fri May 02 17:04:20 EEST 2003: Tahiti
 121: Fri May 02 17:04:20 EEST 2003: Tahiti

1. Annetut syötteet:

TimeStart 29.04.2003

TimeEnd 29.04.2003

2. Metodi palautti:

LogEntry –containertaulukon, jossa 1 kappale lokimerkintöjä. Lokimerkinnät olivat seuraavanlaisia sisällöltään (mainittu vain numero, päiväys ja tekijä):

2: Tue Apr 29 17:43:58 EEST 2003: Tahiti

1. Annetut syötteet:

TimeStart null

TimeEnd 28.04.2003

2. Metodi palautti:

LogEntry –containertaulukon, jossa 1 kappale lokimerkintöjä. Lokimerkinnät olivat seuraavanlaisia sisällöltään (mainittu vain numero, päiväys ja tekijä):

1: Mon Apr 28 17:43:58 EEST 2003: Tahiti

1. Annetut syötteet:

TimeStart null

TimeEnd null

2. Metodi palautti:

LogEntry –containertaulukon, jossa 18 kappaletta lokimerkintöjä. Lokimerkinnät olivat seuraavanlaisia sisällöltään (mainittu vain numero, päiväys ja tekijä):

2: Tue Apr 29 17:43:58 EEST 2003: Tahiti

3: Wed Apr 30 13:24:03 EEST 2003: Tahiti

1: Mon Apr 28 17:43:58 EEST 2003: Tahiti

107: Fri May 02 16:46:01 EEST 2003: Tahiti

108: Fri May 02 16:46:01 EEST 2003: Tahiti

109: Fri May 02 16:46:01 EEST 2003: Tahiti

110: Fri May 02 16:54:41 EEST 2003: Tahiti

111: Fri May 02 16:54:41 EEST 2003: Tahiti

112: Fri May 02 16:54:41 EEST 2003: Tahiti

113: Fri May 02 16:55:53 EEST 2003: Tahiti

114: Fri May 02 16:55:53 EEST 2003: Tahiti

115: Fri May 02 16:55:53 EEST 2003: Tahiti

116: Fri May 02 16:56:52 EEST 2003: Tahiti
 117: Fri May 02 16:56:52 EEST 2003: Tahiti
 118: Fri May 02 16:56:52 EEST 2003: Tahiti
 119: Fri May 02 17:04:20 EEST 2003: Tahiti
 120: Fri May 02 17:04:20 EEST 2003: Tahiti
 121: Fri May 02 17:04:20 EEST 2003: Tahiti

- *getSQLLog*

1. Annetut syötteet:

EntryID 1
EntryClass "KLASSE"
Entry "very lame, _tamed_logentry"
EntryFields "oldie", "older"
OldValues "baddie", "removable"
NewValues "newer", "newest"

2. Metodi palautti:

String -olion, jonka sisältö oli seuraava:
 insert into LogEntry (entrydate, entry, maker) values (now(), 'KLASSE||very lame,
 _tamed_logentry||oldie,older||baddie,removable||newer,newest||', 1)

1. Annetut syötteet:

EntryID 1
EntryClass ""
Entry ""
EntryFields ""
OldValues ""
NewValues ""

2. Metodi palautti:

String -olion, jonka sisältö oli seuraava:
 insert into LogEntry (entrydate, entry, maker) values (now(), '|||||||', 1)

1. Annetut syötteet:

EntryID -1
EntryClass ""
Entry ""
EntryFields ""
OldValues ""
NewValues ""

2. Metodi palautti:

null -viitteen

3.15 MailHandler

3.15.1 Julkiset metodit

- *spamRecommender(DBControl conn, User user)* Metodi on testattu lausekattavasti kahdella erilaisella testitapauksella. Ensimmäinen tapaus on niin sanottu virheetön suoritus, mikä tarkoittaa sitä, että metodissa kutsuttava getMessage-metodi palauttaa true. Esitellyssä tapauksessa metodin paluuarvo on true. Jälkimmäisessä tapauksessa kutsuttava getMessage-metodi palauttaa false, jolloin myös spamRecommender palauttaa false. Käytännössä getMessage-metodi saadaan helpoiten palauttamaan false suorittamalla DBControllin (julkiseksi muunnettu) closeConnection-metodi ennen metodin kutsumista.
 - Annetut syötteet:
 - conn** DBControl, jossa tietokantayhteys on olemassa.
 - user** Containerissa: UserID 12, Name Urpo, Level 1, EMail paasky@cs.helsinki.fi, Recommender Sandaalimies, RecommenderEMail paasky@cs.helsinki.fi
 - Tietokantaan kohdistuvat oletukset:
 - adminemail** huuhaa@cs.helsinki.fi
 - mailserver** localhost
 - serverport** 25
 - referencemessage** Reference: \$username \$login \$recommender \$password
 - referencemessageheader** SAPC Recommendations
 - Metodi palautti:
 - true**
 - Sähköpostissa: Reference: Urpo Urpos login Sandaalimies \$password
 - Annetut syötteet:
 - conn** DBControl, jossa tietokantayhteyttä ei ole olemassa.
 - user** Containerissa: UserID 12, Name Urpo, Level 1, EMail paasky@cs.helsinki.fi, Recommender Sandaalimies, RecommenderEMail paasky@cs.helsinki.fi
 - Tietokantaan kohdistuvat oletukset:
 - adminemail** huuhaa@cs.helsinki.fi
 - mailserver** localhost
 - serverport** 25
 - referencemessage** Reference: \$username \$login \$recommender \$password
 - referencemessageheader** SAPC Recommendations
 - Metodi palautti:
 - false**
 - Virhe** ERR_NO_CONNECTION_OR_ERRONEOUS_SQL
- *spamUser(DBControl conn, User newUser, String password)* Voidaan testata kuten spamRecommender(DBControl conn, User user).

- Annetut syötteet:
 - conn** DBControl, jossa tietokantayhteys on olemassa.
 - user** Containerissa: UserID 12, Name Urpo, Level 1, EMail paasky@cs.helsinki.fi, Recommender Sandaalimies, RecommenderEMail paasky@cs.helsinki.fi
 - password** ihkaomasalasana
- Tietokantaan kohdistuvat oletukset:
 - adminemail** huuhaa@cs.helsinki.fi
 - mailserver** localhost
 - serverport** 25
 - approvemessage** Approve: \$username \$login \$recommender \$password
 - approvemessageheader** SAPC Approve
- Metodi palautti:
 - true**
 - Sähköpostissa: Approve: Urpo Urpos login Sandaalimies ihkaomasalasana
- Annetut syötteet:
 - conn** DBControl, jossa tietokantayhteyttä ei ole olemassa.
 - user** Containerissa: UserID 12, Name Urpo, Level 1, EMail paasky@cs.helsinki.fi, Recommender Sandaalimies, RecommenderEMail paasky@cs.helsinki.fi
 - password** ihkaomasalasana
- Tietokantaan kohdistuvat oletukset:
 - adminemail** huuhaa@cs.helsinki.fi
 - mailserver** localhost
 - serverport** 25
 - approvemessage** Approve: \$username \$login \$recommender \$password
 - approvemessageheader** SAPC Approve
- Metodi palautti:
 - false**
 - Virhe** ERR_NO_CONNECTION_OR_ERRONEOUS_SQL
- *sendNewPassWord(DBControl conn, User user, String newPassword)* Voidaan testata kuten `spamRecommender(DBControl conn, User user)`.
 - Annetut syötteet:
 - conn** DBControl, jossa tietokantayhteys on olemassa.
 - user** Containerissa: UserID 12, Name Urpo, Level 1, EMail paasky@cs.helsinki.fi, Recommender Sandaalimies, RecommenderEMail paasky@cs.helsinki.fi
 - newPassword** uusisalasana
 - Tietokantaan kohdistuvat oletukset:
 - adminemail** huuhaa@cs.helsinki.fi

mailserver localhost

serverport 25

passwordmessage Password: \$username \$login \$recommender \$password

passwordmessageheader SAPC Password

– Metodi palautti:

true

Sähköpostissa: Password: Urpo Urpos login Sandaalimies uusisalasana

– Annetut syötteet:

conn DBControl, jossa tietokantayhteyttä ei ole olemassa.

user Containerissa: UserID 12, Name Urpo, Level 1, EMail paasky@cs.helsinki.fi, Recommender Sandaalimies, RecommenderEMail paasky@cs.helsinki.fi

– Tietokantaan kohdistuvat oletukset:

adminemail huuhaa@cs.helsinki.fi

mailserver localhost

serverport 25

passwordmessage Password: \$username \$login \$recommender \$password

passwordmessageheader SAPC Password

– Metodi palautti:

false

Virhe ERR_NO_CONNECTION_OR_ERRONEOUS_SQL

- *getErrorMessages()* Suoritetaan aina lausekattavasti.

Testattiin lausekattavasti.

3.15.2 Yksityiset metodit

- *getMessage(DBControl conn, String query)* Metodi testataan virheettömän suorituksen ja virheellisen suorituksen osalta. Jälkimmäiseen tapaukseen päädytään, jos tietokantayhteys ei ole kunnossa tai SQL-lause on virheellinen. Tällöin paluuarvo on false ja virhemuuttuja saa arvon TahitiLibrary.ERR_NO_CONNECTION_OR_ERRONEOUS_SQL. Virheettömän suorituksen paluuarvo on luonnollisesti true.

Testattiin lausekattavasti spamXXX-metodien yhteydessä.

- *processText(String text, String name, String login, String password, String recommender)* Metodi tulee suoritettua lausekattavasti silloin kun mikään syötteistä ei ole null.

Testattiin lausekattavasti spamXXX-metodien yhteydessä.

3.15.3 Aliluokka MailerThread

- *run()* Tulee testattua aina lausekattavasti.
Testattiin lausekattavasti spamXXX-metodien yhteydessä.
- *send()* Kun muuttuja DEBUG on true, metodi tulee testattua lausekattavasti.
Testattiin lausekattavasti spamXXX-metodien yhteydessä.

3.16 SystemHandler

Luokan testaamista varten luodaan erillinen testiluokka. Myös tietokantayhteys on tarpeen testausta varten.

3.16.1 Kirjastometodit

Kirjastometodit ovat staattisia metodeita, joilla toteutetaan syötteiden tyyppin tarkastaminen, päivämäärän muodon muuttamiset sekä poistetaan syötteestä mahdollisesti haitallisia merkkejä. Syötteiden tyyppitarkistus tarkoittaa lukuarvojen etsimistä merkkijonosyötteistä.

- *deSQL*
 1. Annetut syötteen:
 - String** SAPC = 'password' <header>
 2. Metodi palautti:
 - String** -olion, jonka sisältö oli seuraavanlainen:
SAPC "password" header
Testillä kokeiltiin haitallisten merkkien poistamista syötteestä, näitä olivat <, >, ' ja =. Muut merkit poistettiin, mutta ' merkittiin SQL-komentiksi ". Tulokset olivat haluttuja.
 1. Annetut syötteen:
 - null** -viite
 2. Metodi palautti:
 - null** -viitteen, kuten pitikin.
 1. Annetut syötteen:
 - String** ""
 2. Metodi palautti:
 - String** -olion, sisältönä tyhjä merkkijono, kuten oli tarkoituskin.
- *isDouble*
 1. Annetut syötteen:

String "3.3"

2. Metodi palautti:

boolean –muuttujan arvoltaan true

1. Annetut syötteet:

String "aurum"

2. Metodi palautti:

boolean –muuttujan arvoltaan false

- *isInt*

1. Annetut syötteet:

String "99999"

2. Metodi palautti:

boolean –muuttujan arvoltaan true

1. Annetut syötteet:

String "saurum"

2. Metodi palautti:

boolean –muuttujan arvoltaan false

- *isLong*

1. Annetut syötteet:

String "9999999999999999"

2. Metodi palautti:

boolean –muuttujan arvoltaan true

1. Annetut syötteet:

String "laurum"

2. Metodi palautti:

boolean –muuttujan arvoltaan false

- *parseDate*

1. Annetut syötteet:

String "6000"

String "saurum"

2. Metodi palautti:

null –viite

1. Annetut syötteet:

String "24.04.2003"

String "dd.MM.yyyy"

2. Metodi palautti:

Date -olion, joka palauttaa seuraavan arvon toString()-metodistaan:

Wed Apr 23 03:00:00 EEST 2003

- *toTimestamp*

Testataan lähinnä käytännössä muiden metodien toiminnan yhteydessä. Oletetaan toimivaksi.

3.16.2 Julkiset metodit

Pakkausnäkyvän metodit vastaavat varsinaisesta järjestelmätoiminnasta, ts. TahitiApista on kutsometodit jokaista pakkausnäkyvän metodia kohden.

- *changeSettings*

1. Annetut syötteet:

AdminEMail null

MailServer null

ServerPort null

ApproveMessage "You have been approved"

PasswordMessage "This is your new \$password"

ReferenceMessage "Have you recommended this user?"

[**ApproveMessageHeader** "SAPC approve"

PasswordMessageHeader "SAPC password"

ReferenceMessageHeader "SAPC Recommendations"

2. Metodi palautti:

boolean arvoltaan false, kuten pitikin

1. Annetut syötteet:

AdminEMail admin@mail.astro.helsinki.fi

MailServer null

ServerPort null

ApproveMessage "You have been approved"

PasswordMessage "This is your new \$password"

ReferenceMessage "Have you recommended this user?"

[**ApproveMessageHeader** "SAPC approve"

PasswordMessageHeader "SAPC password"

ReferenceMessageHeader "SAPC Recommendations"

2. Metodi palautti:

boolean arvoltaan false, kuten pitikin

1. Annetut syötteet:

AdminEMail admin@mail.astro.helsinki.fi

MailServer mail.astro.helsinki.fi

ServerPort null

ApproveMessage "You have been approved"

PasswordMessage "This is your new \$password"

ReferenceMessage "Have you recommended this user?"

[**ApproveMessageHeader** "SAPC approve"

PasswordMessageHeader "SAPC password"

ReferenceMessageHeader "SAPC Recommendations"

2. Metodi palautti:

boolean arvoltaan false, kuten pitikin. Porttinumeroa kokeiltiin myös syötteillä 86579568 ja -1, tulokset olivat tätä testiä vastaavia, kuten pitikin (epäkelpo porttinumero)

1. Annetut syötteet:

AdminEMail admin@mail.astro.helsinki.fi

MailServer mail.astro.helsinki.fi

ServerPort 25

ApproveMessage "You have been approved"

PasswordMessage "This is your new \$password"

ReferenceMessage "Have you recommended this user?"

[**ApproveMessageHeader** "SAPC approve"

PasswordMessageHeader "SAPC password"

ReferenceMessageHeader "SAPC Recommendations"

2. Metodi palautti:

boolean arvoltaan true, kuten pitikin

- *getSettings*

1. Annetut syötteet:

DBControl -luokan aliluokka Dummydb, joka ei pysty muodostamaan yhteyttä

2. Metodi palautti:

null -viitteen, kuten pitikin

1. Annetut syötteet:

DBControl -olio, tietokantayhteys kunnossa

2. Metodi palautti:

Settings -olio, jossa kannassa olevat arvot

3.17 TahitiLibrary

toDate

1. Annetut syötteet:

Julian Date 2448188.5

2. Metodi palautti:

Date Wed Oct 24 02:00:00 EET 1990

toJulian

1. Annetut syötteet:

Date Wed Oct 24 02:00:00 EET 1990

2. Metodi palautti:

Julian Date 2448188.5

distance

1. Annetut syötteet:

double[3] (0.855732, 0.507082, 0.000003)

2. Metodi palautti:

double 2.899632823190733

3. Metodi piti palauttaa:

double 2.8996

solarPhase

1. Annetut syötteet:

double[3] (0.855732, 0.507082, 0.000003)

double[3] (2.819902, -0.349044 0.026762)

2. Metodi palautti:

double 16.494845512325472

3. Metodi piti palauttaa:

double 16.50 pyöristettynä 0.1 tarkkuuteen

eclipticLongitude

1. Annetut syötteet:

`double[3]` (0.855732, 0.507082, 0.000003)

`double[3]` (2.819902, -0.349044 0.026762)

2. Metodi palautti:

double 336.4489783243571

3. Metodi piti palauttaa:

double 336.40 pyöristettynä 0.1 tarkkuuteen

eclipticLatitude

1. Annetut syötteet:

`double[3]` (0.855732, 0.507082, 0.000003)

`double[3]` (2.819902, -0.349044 0.026762)

2. Metodi palautti:

double 0.7155175395508387

3. Metodi piti palauttaa:

double 0.70 pyöristettynä 0.1 tarkkuuteen

countCoordinates

1. Annetut syötteet:

time Fri Feb 02 07:35:02 EET 2001

Epoch Thu Oct 18 03:00:00 EEST 2001

Axis 1.322691

Eccentricity 0.27961274

Inclination 1.72799

Argument 161.02783

Longitude 70.92064

Anomaly 107.8134613

2. Metodi palautti:

Asteroidx -0.951737837113183

Asteroidy 0.7700886676646176

Asteroidz 0.0347291121861053

3. Metodi piti palauttaa:

Asteroidx -0.951738642

Asteroidy 0.770087699

Asteroidz 0.034729127

lighttimeCorrect

1. Annetut syötteet:

time 2451942.732663

asteroidX -0.951738642

asteroidY 0.770087699

asteroidZ 0.034729127

earthX 0.274172392 - 0.951738642

earthY -0.0544258374 + 0.770087699

earthZ -0.0347286299 + 0.034729127

2. Metodi palautti:

double 2451942.731036202

3. Metodi piti palauttaa:

double 2451942.73104

toIntensity(reduceMagnitude())

1. Annetut syötteet:

magnitudi 18.344

asteroidX -0.951738642

asteroidY 0.770087699

asteroidZ 0.034729127

earthX 0.274172392 - 0.951738642

earthY -0.0544258374 + 0.770087699

earthZ -0.0347286299 + 0.034729127

2. Metodi palautti:

double 5.470038713568135E-4

3. Metodi piti palauttaa:

double 0.5470039E-03

3.18 TrajectoryHandler

getDataRows

1. Annetut syötteet:

String "Rai\nrai\nraa\n\nmina\nhaistan\nihmis\nlihaa"

2. Metodi palautti:

String[] (Rai, rai, raa, mina, haistan, ihmis, lihaa)

checkParameters

1. Annetut syötteet:

DBControl null-viite

TrajectoryFormContainer Normaali container

User Normaali Container

2. Metodi palautti:

boolean false

3. Virheenä oli:

Virhe 28

1. Annetut syötteet:

DBControl normaali DBControl

TrajectoryFormContainer null-viite

User Normaali Container

2. Metodi palautti:

boolean false

3. Virheenä oli:

Virhe 28

1. Annetut syötteet:

DBControl normaali DBControl

TrajectoryFormContainer Normaali container

User null-viite

2. Metodi palautti:

boolean false

3. Virheenä oli:

Virhe 28

1. Annetut syötteet:

DBControl normaali DBControl

TrajectoryFormContainer Container, jossa Data-attribuutti on null-viite

User normaali Container

2. Metodi palautti:

boolean false

3. Virheenä oli:

Virhe 28

1. Annetut syötteet:

DBControl normaali DBControl

TrajectoryFormContainer Container, jossa Columns-attribuutti on null-viite

User normaali Container

2. Metodi palautti:

boolean false

3. Virheenä oli:

Virhe 28

1. Annetut syötteet:

DBControl normaali DBControl

TrajectoryFormContainer Container, jossa Columns-attribuutin sisällä null-viite

User normaali Container

2. Metodi palautti:

boolean false

3. Virheenä oli:

Virhe 28

1. Annetut syötteet:

DBControl normaali DBControl

TrajectoryFormContainer Container, jossa ei numero columns-aulukossa

User normaali Container

2. Metodi palautti:

boolean false

3. Virheenä oli:

Virhe 28

1. Annetut syötteet:

DBControl normaali DBControl

TrajectoryFormContainer Container, jossa columns-tila sisältää vain 8 alkua

User normaali Container

2. Metodi palautti:

boolean false

3. Virheenä oli:

Virhe 28

1. Annetut syötteet:

DBControl normaali DBControl

TrajectoryFormContainer normaali Container

User normaali Container

2. Metodi palautti:

boolean true

3. Virheenä oli:

Virhe 0

arrangeColumns

1. Annetut syötteet:

Validi columns-tila (TahitiLibrary.COL_DESIGNATION, TahitiLibrary.COL_EPOCH
TahitiLibrary.COL_REFERENCE, TahitiLibrary.COL_AXIS,
TahitiLibrary.COL_INCLINATION, TahitiLibrary.COL_ECCENTRICITY,
TahitiLibrary.COL_ARGUMENT, TahitiLibrary.COL_LONGITUDE,
TahitiLibrary.COL_ANOMALY)

2. Metodi palautti:

boolean true

3. Metodi virhe oli:

Virhe 0

4. Metodi asetti:

Designation should be set at location 0 and it was: 0

Epoch should be set at location 1 and it was: 1

Reference should be set at location 2 and it was: 2

Axis should be set at location 3 and it was: 3

Inclination should be set at location 4 and it was: 4

Eccentricity should be set at location 5 and it was: 5

Argument should be set at location 6 and it was: 6

Longitude should be set at location 7 and it was: 7

Anomaly should be set at location 8 and it was: 8

1. Annetut syötteet:

Virheellinen columns-tilukko (TahitiLibrary.COL_DESIGNATION, TahitiLibrary.COL_EPOCH, TahitiLibrary.COL_REFERENCE, TahitiLibrary.COL_AXIS, -1, TahitiLibrary.COL_ECCENTRICITY, TahitiLibrary.COL_ARGUMENT, TahitiLibrary.COL_LONGITUDE, TahitiLibrary.COL_ANOMALY)

2. Metodi palautti:

boolean false

3. Metodi virhe oli:

Virhe 31

parseDesignation

1. Annetut syötteet:

Designation asetettuna kohtaan 1

String "20020506 1 Ceres JPL#24 2.7664122 0.07911582 10.58348 80.48630 73.98448 189.2751236"

2. Metodi palautti:

boolean true

3. Metodi asetti parseAsteroidiin:

AsteroidID 3

Number 1

Name Ceres

Designation null-viite

1. Annetut syötteet:

Designation asetettuna kohtaan 0

String "572 Rebekka (1905 RB) 20020506 MPC25034 2.4005588 0.15620300 10.55764 194.76662
191.86925 335.0162000"

2. Metodi palautti:

boolean true

3. Metodi asetti parseAsteroidiin:

AsteroidID 4

Number 572

Name Rebekka

Designation 1905 RB

4. Metodi asetti parseTrajectoryyn:

Asteroid 4

TrajectoryID 1

1. Annetut syötteet:

Designation asetettuna kohtaan 0

String "7499 (1996 OO2) 20020506 MPO01401 3.1530959 0.12994080 9.97660 325.17348
71.27753 13.4125300 "

2. Metodi palautti:

boolean true

3. Metodi asetti parseAsteroidiin:

AsteroidID 5

Number 7499

Name null-viite

Designation 1996 002

4. Metodi asetti parseTrajectoryyn:

Asteroid 5

TrajectoryID 1

1. Annetut syötteet:

Designation asetettuna kohtaan 0

String "(1999 TG189) 20020506 MPC12175 2.6366161 0.08044730 12.46897 347.51730 89.69662 174.8446900"

2. Metodi palautti:

boolean true

3. Metodi asetti parseAsteroidiin:

AsteroidID 6

Number -1

Name null-viite

Designation 1999 TG189

4. Metodi asetti parseTrajectoryyn:

Asteroid 6

TrajectoryID 1

1. Annetut syötteet:

Designation asetettuna kohtaan 0

String "1802 Zhang Heng (1964 TW1) 20020506 MPC24718 2.8436664 0.03335770 2.68596 142.74740 300.04847 235.8949500"

2. Metodi palautti:

boolean true

3. Metodi asetti parseAsteroidiin:

AsteroidID 7

Number 1802

Name Zhang Heng

Designation 1964 TW1

4. Metodi asetti parseTrajectoryyn:

Asteroid 7

TrajectoryID 1

1. Annetut syötteet:

Designation asetettuna kohtaan 0

String "(TW1) 20020506 MPC24718 2.8436664 0.03335770 2.68596 142.74740 300.04847 235.8949500"

2. Metodi palautti:

boolean true

3. Metodi asetti parseAsteroidiin:

AsteroidID 8

Number -1

Name null-viite

Designation TW1

4. Metodi asetti parseTrajectoryyn:

Asteroid 8

TrajectoryID 1

1. Annetut syötteet:

Designation asetettuna kohtaan 0

String "25143 (1998 SF36) 20020506 JPL#63 1.3228213 0.27948049"

2. Metodi palautti:

boolean true

3. Oletettiin:

Kanta Kannassa vastaava asteroidi (ID 2), jolla ennestään 2 rataelementtiä

4. Metodi asetti parseAsteroidiin:

parseAsteroid null-viite

5. Metodi asetti parseTrajectoryyn:

Asteroid 2

TrajectoryID 3

1. Annetut syötteet:

Designation asetettuna kohtaan 0

String "25143 Ida 20020506 JPL#63 1.3228213 0.27948049 1.72805 70.92126 161.01582
237.4062178"

2. Metodi palautti:

boolean false

3. Oletettiin:

Kanta Kannassa vastaavat eri asteroidid ida ja asteroidi numero 25143

4. Virhe oli:

Virhe 30

parseRow

1. Annetut syötteet:

Designation asetettuna kohtaan 0

String "25143 (1998 SF36) 20020506 JPL#63 1.3228213 0.27948049 1.72805 70.92126 161.01582
237.4062178"

2. Metodi palautti:

boolean true

3. Oletettiin:

Kanta Kannassa vastaava asteroidi (ID 2), jolla ennestään 2 rataelementtiä

4. Metodi asetti parseTrajectoryyn:

asteroid should be 2 is: 2

TrajectoryID should be 3 is: 3

Epoch Mon May 06 03:00:00 EEST 2002

Axis 1.3228213

Eccentricity 0.27948049

Inclination 1.72805

Argument 161.01582

Longitude 70.92126

Anomaly 237.4062178

Reference JPL#63

5. Metodi asetti parseAsteroidiin:

parseAsteroid null-viite

1. Annetut syötteet:

Designation asetettuna kohtaan 0

String "572 Rebekka (1905 RB) 20020506 MPC25034 2.4005588 0.15620300 10.55764 194.76662
191.86925 335.0162000"

2. Metodi palautti:

boolean true

3. Metodi asetti parseTrajectoryyn:

asteroid 9

TrajectoryID 1
Epoch Mon May 06 03:00:00 EEST 2002
Axis 2.4005588
Eccentricity 0.156203
Inclination 10.55764
Argument 191.86925
Longitude 194.76662
Anomaly 335.0162
Reference MPC25034

4. Metodi asetti parseAsteroidiin:

AsteroidID 9
Number 572
Name Rebekka
Designation 1905 RB

1. Annetut syötteet:

Kenttien Sijainnit kaikkia yhdeksää kenttää ei ole sijoitettu
String Validi syöte

2. Metodi palautti:

boolean false

3. Virhe oli:

Virhe 29

1. Annetut syötteet:

Designation asetettuna kohtaan 0
String "572 Rebekka (1905 RB) 20020506 MPC25034 RAI!!!!!! 0.15620300 10.55764 194.76662
 191.86925 335.0162000"

2. Metodi palautti:

boolean false

3. Virhe oli:

Virhe 13

1. Annetut syötteet:

Designation asetettuna kohtaan 0

String "C1 C2 C3"

2. Metodi palautti:

boolean false

3. Virhe oli:

Virhe 32

parseAsteroidInsert

1. Annetut syötteet:

AsteroidID 1

Number 2

Name Rai

Designation Haallejula

2. Metodi palautti:

String insert into asteroid (asteroidnumber, designation, name, number) values (1,'Haallejula','Rai',2);

1. Annetut syötteet:

AsteroidID 1

Number -1

Name Rai

Designation Haallejula

2. Metodi palautti:

String insert into asteroid (asteroidnumber, designation, name, number) values (1,'Haallejula','Rai', null);

1. Annetut syötteet:

AsteroidID 1

Number 2

Name null

Designation Haallejula

2. Metodi palautti:

String insert into asteroid (asteroidnumber, designation, name, number) values (1,'Haallejula',null,2);

1. Annetut syötteet:

AsteroidID 1
Number 2
Name Rai
Designation null

2. Metodi palautti:

String insert into asteroid (asteroidnumber, designation, name, number) values (1,null,'Rai',2);

parseAsteroidInsert

1. Annetut syötteet:

asteroid 1
TrajectoryID 2
Epoch new Date(System.currentTimeMillis())
Axis 2.0
Eccentricity 4.0
Inclination 5.0
Argument 6.0
Longitude 7.0
AnomalyDBC, 2 8.0
Reference REF

2. Metodi palautti:

String insert into trajectory (asteroid, trajectorynumber, axis, eccentricity, inclination, argument, longitude, anomaly, epoch, orbitreference) values (1,2,3.0,4.0,5.0,6.0,7.0,8.0,to_timestamp('123:2000-01-01T00:00:00','REF'));

aspectUpdate

1. Oletettiin:

Kanta Kannassa valokäyrä, jolla 5 pistettä lightcurveUpdate-näkymässä

2. Metodi palautti:

String Update Datapoint set asteroidx = 1.4983025205184939, asteroidy = 0.5667701362638181, asteroidz = -0.0371290296767427, earthx = -0.7082896123872449, earthy = -0.7220260194444483, earthz = -7.386545563259279E-7 where lightcurve = 1 and pointnumber = 1;

String Update Datapoint set asteroidx = 1.4982879816628762, asteroidy = 0.5668533680135103, asteroidz = -0.03712779438376585, earthx = -0.7082056194220001, earthy = -0.722110437722276, earthz = -7.387124939547136E-7 where lightcurve = 1 and pointnumber = 2;

String Update Datapoint set asteroidx = 1.4982755293192733, asteroidy = 0.5669246293150616, asteroidz = -0.037126736623859985, earthx = -0.7081336968887838, earthy = -0.7221827084694198, earthz = -7.387620918277651E-7 where lightcurve = 1 and pointnumber = 3;

String Update Datapoint set asteroidx = 1.4982485199149722, asteroidy = 0.5670791154777791, asteroidz = -0.03712444311986183, earthx = -0.707977747176947, earthy = -0.7223393626022367, earthz = -7.388695916364177E-7 where lightcurve = 1 and pointnumber = 4;

String Update Datapoint set asteroidx = 1.4982256330665304, asteroidy = 0.5672099349766158, asteroidz = -0.037122500538811665, earthx = -0.7078456561513436, earthy = -0.7224719961196291, earthz = -7.389605987370974E-7 where lightcurve = 1 and pointnumber = 5;

String Update Lightcurve set defaultEarthTrajectory = 1, defaultTrajectory = 2, phaseAngle = 15.206109747637987, distancefromsun = 1.6018247637160818, distancefromearth = 2.5556801760357395, Longitude = 210.14386640720468, Latitude = 0.833320487815045 where ObservationNumber = 1;

getTrajectory

1. Annetut syötteet:

DBControl Validi DBControl

AsteroidID 2

2. Oletettiin:

Kanta Kannassa kahdet rataelementit asteroidilla 2

3. Metodi palautti:

Trajectory[0 .TrajectoryID] 2

Trajectory[0 .Epoch] Mon Jan 01 02:00:00 EET 2001

Trajectory[0 .Axis] 1.322691

Trajectory[0 .Eccentricity] 0.27961274

Trajectory[0 .Inclination] 1.72799

Trajectory[0 .Argument] 161.02783

Trajectory[0 .Longitude] 70.92064

Trajectory[0 .Anomaly] 107.8134613

Trajectory[0 .Reference] Kaasalainen

Trajectory[1 .TrajectoryID] 1

Trajectory[1 .Epoch] Sat Jan 05 02:00:00 EET 2002

Trajectory[1 .Axis] 1.3228213

Trajectory[1 .Eccentricity] 0.27948049

Trajectory[1 .Inclination] 1.72805

Trajectory[1 .Argument] 161.01582

Trajectory[1 .Longitude] 70.92126
Trajectory[1 .Anomaly] 237.4062178
Trajectory[1 .Reference] JPL#63

1. Annetut syötteet:

DBControl null
AsteroidID 2

2. Metodi palautti:

Trajectory null-viite

3. Virhe oli:

Virhe 28

addTrajectory

1. Annetut syötteet:

DBControl Validi DBControl

TrajectoryFormContainer.columns (TahitiLibrary.COL_DESIGNATION, TahitiLibrary.COL_EPOCH,
TahitiLibrary.COL_REFERENCE, TahitiLibrary.COL_AXIS, TahitiLibrary.COL_INCLINATION, TahitiLibrary.COL_ARGUMENT,
TahitiLibrary.COL_LONGITUDE, TahitiLibrary.COL_ANOMALY)

TrajectoryFormContainer.data "20020506 1 Ceres JPL#24 2.7664122 0.07911582 10.58348
80.48630 73.98448 189.2751236\n 20030610 7 Iris JPL#27 2.3859372 0.22979201 5.52380
259.85969 145.24594 28.9681205"

2. Oletettiin:

Kanta Kannassa ei asteroideja Iris eikä Ceres

3. Metodi palautti:

boolean true

4. Metodi lisäsi kantaan:

Asteroidi Ceres

Asteroidi Iris

Trajectory Ceres

Trajectory Iris

1. Annetut syötteet:

DBControl Validi DBControl

TrajectoryFormContainer.columns (TahitiLibrary.COL_DESIGNATION, TahitiLibrary.COL_EPOCH, TahitiLibrary.COL_REFERENCE, TahitiLibrary.COL_AXIS, TahitiLibrary.COL_INCLINATION, TahitiLibrary.COL_ARGUMENT, TahitiLibrary.COL_LONGITUDE, TahitiLibrary.COL_ANOMALY)

```
TrajectoryFormContainer.data "1 Ceres 20020506 JPL#24 2.7664122 0.07911582 10.58348
80.48630 73.98448 189.2751236\n
1 Ceres 20030424 Muhonen 2.7664122 0.07311582 10.58348 33.48630 72.98448 333.2751236\n
2 Pallas 20020506 JPL#21 2.7734826 0.22993723 34.84248 173.17166 310.39442 175.4142995
\n
23644 Yamaneko (1997 AW17) 20020506 MPC11151 2.3957585 0.12661210 2.13874 78.27102
112.20855 105.3140400 \n
3 Juno 20030610 JPL#22 2.6670328 0.25874667 12.97191 170.13172 247.93499 164.4353501
\n
4 Vesta 20020506 JPL#25 2.3626478 0.08887781 7.13485 103.94712 149.67895 213.2933256\n
5 Astraea 20011018 JPL#17 2.5733413 0.19281794 5.36909 141.69049 357.68118 196.0781904\n
6 Hebe 20020506 JPL#22 2.4255618 0.20145268 14.76672 138.84763 238.92435 269.4231085\n
7 Iris 20030610 JPL#27 2.3859372 0.22979201 5.52380 259.85969 145.24594 28.9681205\n
8 Flora 20030610 JPL#27 2.2013023 0.15656121 5.88723 111.03661 284.99710 244.9471365\n
9 Metis 20020506 JPL#28 2.3865330 0.12228810 5.57684 68.99317 5.65049 60.0990845\n
10 Hygiea 20030610 JPL#31 3.1355430 0.11941886 3.84404 283.65291 313.84348 202.4910439\n
"
```

2. Oletettiin:

Kanta Asteroidit Ceres ja Iris. Lisäksi yksi Cereksen ja Iriksen rataelementeistä on duplikaatti.

3. Metodi palautti:

boolean true

4. Metodi lisäsi kantaan:

Asteroidi Kaikki paitsi Ceres ja Iris

Trajectory Kaikki paitsi Cereksen ja Iriksen duplikaatit.

3.19 UserHandler

3.19.1 Yksityiset metodit

- *encrypt(String plaintext)*

Encrypt-metodin toiminta on hyvin suoraviivaista, ja jokaisella käyttökerralla lähes jokainen metodin rivi tulee suoritetuksi. Vain alun poikkeuskäsittely muodostaa poikkeuksen tähän ja poikkeuskäsittelyn testaamiseksi joudutaan luokan sisäiselle HASH_ALGORITHM-muuttujalle antamaan järjetön arvo. Jotta kaikki rivit tulevat suoritetuksi, tarkistetaan palautettavasta hexadecimaali jonosta, että se sisältää vähintään yhden nollalla alkavan tavun, sekä yhden tavun, jonka numeerinen arvo ylittää luvun 128.

Oletus SALT –muuttujan arvolla tälläinen testiaineisto on esimerkiksi ”salasana”, joka palauttaa SHA–algoritmilla ja ”Just type something here”–suolalla merkkijonon *Ox2a4a117c1ac82c679472031f498e2*. Tämän merkkijonon tavu 03 sisältää tarvittavan lisänollan, ja tavu C8 on selvästi suurempi kuin raja-arvo 128.

Virheellisellä HASH_ALGORITHM –arvolla metodi palautti oletetusti null. Lisäksi virheen nouto getErrorMessages–metodilla onnistui.

- *generatePassword()*

Koska generetaPasswords metodin koko toimintaidea liittyy satunnaisuuteen, ei sen testaaminen lausekattavasti ole täysin mahdollista ilman muutoksien tekemistä metodiin. Liisämällä jokaiseen metodin ehtoon tulostuslausekkeen voitiin varmistua siitä, että kaikki metodin lauseet tulivat suoritetuiksi. Metodin lopussa olevaa poikkeuskäsittelyä pystyttiin testamaan muuntamalla käytettävä merkkijärjestelmä ASCII:sta johonkin sellaiseen koodistoon, jota ei ole olemassa.

- *parseForm(UserFormContainer data)*

Koska parseForm ottaa syötteenä UserFormContainerin, tarvittiin sen testaamista varten ilmentymä kyseisestä containerista. Koska pakkauksesta fi.helsinki.cs.group.tahiti.ui löytyy rajapinnan toteuttava UserInfoDataBean–luokka, käytetään testauksessa sitä.

Testauksessa käytetty perustapaus oli seuraavan:

1. Annetut syötteen:

Name aapo
E-Mail haista@paska.net
Status Professional
UserID 1
UserName Aapo Abraham
Recommender Eetu
RecommenderE-Mail ekkekekke@jossain.net
Level 1
Version 2
DefaultDeceptor detector
DefaultPhotometricSystem photosys
DefaultLocation location
DefaultTimeStandard timestandard
Information information
Comment comment

2. Metodi palautti:

ApiUser –containerin, jossa vastaavat arvot vastaavissa kentissä.

Seuraavassa on lueteltu luokan eri haarautumisvaihtoehdot, ja näistä saatavat edellytykset testitapauksille. Käytetystä testiaineistosta on ilmoitettu vain muutokset perustapaukseen.

- UserID

Alun try-catch rakenteessa muunnetaan annettu userID long-tyypiksi. Testauksessa tarvitaan aineisto, jossa userID on oikeaa muotoa, sekä tapaus, jossa se ei ole oikeaa muotoa. Mikäli kentän sisältö on väärää muotoa, keskeytyy metodin suoritus virheeseen.

1. Annetut syötteen:

UserID null

2. Metodi palautti:

Null-viitteen

Virhe TahitiLibrary.ERR_NUMBERFORMAT

- Level ja version

Käyttäjätasolle ja versiolle tehdään vastaavat tarkistukset, joten myös niitä kohtaan joudutaan tekemään testi, jossa se on oikean tyyppinen, numeerinen sekä testi, jossa kentän arvona on jotain muuta. Tällöin metodin suoritus keskeytyy virheeseen.

1. Annetut syötteen:

Tapaus 1: level isoleveli

Tapaus 2: versio null

2. Metodi palautti:

Null-viitteen

Virhe TahitiLibrary.ERR_NUMBERFORMAT

- Käyttäjätasus

Mahdolliset käyttäjätasuksesta ovat "Amateur" sekä "Professional". Testit pitää suorittaa molemmilla arvoilla, sekä jollain ei-validilla arvolla. Mikäli arvo on väärä, keskeytyy metodin suoritus virheeseen.

1. Annetut syötteen:

Status Amateur

2. Metodi palautti:

ApiUser -containerin, jossa vastaavat arvot vastaavissa kentissä.

1. Annetut syötteen:

Status kingoftheworld

2. Metodi palautti:

null-viitteen

Virhe TahitiLibrary.ERR_DATAFORMAT

- Muut pakolliset kentät

Koska kaikista pakollisista kentistä tarkistetaan että ne ovat täytettyinä, joudutaan testauksessa jokainen kenttä jättämään järjestyksessä tyhjäksi. Lausekattavuuden täyttämiseksi kentät pitää erikseen täyttää tyhjällä Stringillä (), ja null-viitteellä. Pakollisten kenttien puuttuessa metodin suoritus päättyy virheeseen.

1. Annetut syötteet:

Tapaus 1: Name <tyhjä String>

Tapaus 2: Name null

Tapaus 3: EMail <tyhjä String>

Tapaus 4: EMail null

Tapaus 5: Status null

Tapaus 6: UserName <tyhjä String>

Tapaus 7: UserName null

Tapaus 8: Recommender <tyhjä String>

Tapaus 9: Recommender null

Tapaus 10: RecommenderEMail <tyhjä String>

Tapaus 11: RecommenderEMail null

2. Metodi palautti:

Null-viitteen

Virhe TahitiLibrary.ERR_MANDANTORY_DATA_MISSING

- Vapaaehtoiset kentät

Koska metodi ei tee vapaaehtoisille kentille muita tarkistuksia, kuin tyhjä stringit muutetaan null-viittauksiksi, saadaan lausekattavuus täytettyä laittamalla kaikkiin vapaaehtoisin kenttiin tyhjä String-arvot testauksessa.

1. Annetut syötteet:

DefaultDeceptor <tyhjä String>

DefaultPhotometricSystem <tyhjä String>

DefaultLocation <tyhjä String>

DefaultTimeStandard <tyhjä String>

Information <tyhjä String>

Comment <tyhjä String>

2. Metodi palautti:

ApiUser -containerin, jossa vapaaehtoiset kentät ovat null-viitteitä.

3.19.2 Normaalit metodit

Myös normaalien metodien testauksessa käytetään testausluokkaa `UserHandlerTester`. Koska normaalit metodit saattavat olla riippuvaisia yksityisistä metodeista, `SystemHandler`in apumetodeista, tai `DBControl`in tarjoamasta tietokantayhteydestä, ei täysin puhdas modulitestaus ole mahdollista. Tietokantaan tehtäviä kirjoituksia ei suoriteta loppuun asti, vaan testauksessa käytetään modifioitua `DBControl`-luokkaa, joka vain tulostaa mahdollisest `Insert`, `Update` ja `Delete` -lauseet ruudulle. Haut kuitenkin suoritetaan oikeasti.

- *registrationRequest*

Tämänkin metodin testaamiseen käytetään pakkauksesta `fi.helsinki.cs.group.tahiti.ui` löytyvää `UserInfoDataBean`-luokkaa. Perustapauksessa kaikki kentät ovat täytettyinä, eikä tietokannasta tule löytyä annettua `loginID`:tä vastaavaa käyttäjätunnusta. `UserNumber`-kentän arvo saadaan automaattisesti, eikä metodin käyttäjä voi siihen vaikuttaa. Testauksen perustapauksessa käytetyt syötteen:

1. Annetut syötteen:

```

UserID <tyhjä String>
Name aapo
EMail haista@paska.net
Status Professional
UserName Aapo Abraham
Recommender Eetu
RecommenderEMail ekkekekke@jossain.net
Level <tyhjä String>
Version <tyhjä String>
DefaultDeceptor DETECTOR: Arskan tarkka näkö
DefaultPhotometricSystem PHOTOMETRICSYSTEM: Mikä 'tun systeemi?
DefaultLocation LOCATION: paarissahan me
DefaultTimeStandard TIMESTD: olalla oli kello!
Information INFO: tää on ihan totta, joo
Comment tarkkaa info! (jos tämä näkyy niin se on VOIVOI

```

2. Metodi palautti:

```

Muodostettava SQL-lause: INSERT INTO TahitiUser (UserNumber, UserName,
    UserLevel, Password, Name, EMail, Status, Recommender, RecommenderEMail, In-
    formation, DefaultDetector, DefaultLocation, DefaultPhotometricSystem, Default-
    TimeStandard, LastLogin, Comment) VALUES (57, 'Aapo Abraham',0, 'Ox2c7d7f1c384aa0a64e7a2
    'aapo', 'haista@paska.net', 'Professional', 'Eetu', 'ekkekekke@jossain.net', 'INFO: tää on
    ihan totta, joo', 'DETECTOR: Arskan tarkka näkö', 'LOCATION: paarissahan me', 'PHOTOMETRIC
    Mikä "tun systeemi?', 'TIMESTD: olalla oli kello!', null, null);

true

```

Seuraavassa on listattu perustapauksen lisäksi tarvittavat testitapaukset.

- Virheelliset tiedot annetussa `UserFormContainer`issa

Koska annetun `UserFormContainer`in tulkitseminen tapahtuu yksityisessä `parseForm`-metodissa, ei tämän luokan yhteydessä erikseen testata kaikki eri syöteyhdistelmiä. Testauksessa vain tarkistetaan, ettei satunnainen, epäkelvo syöte mene läpi testistä.

1. Annetut syötteen:

```

Status King of the world!

```

2. Metodi palautti:

Null-viitteen

Virhe TahitiLibrary.ERR_DATAFORMAT

- Tunnuksen olemassaolon tarkistus

Olemassaolo tarkistuksessa tarvitaan tietokantaa, joten testausta varten tarvitaan sellainen aineisto, että ensimmäisellä ajolla luotavaa käyttäjätunnusta ei löydy kannasta, ja toisella ajolla se löytyy.

1. Annetut syötteet:

Status King of the world!

2. Tietokantaan kohdistuvat oletukset:

Username Käyttäjän Tahiti tulee löytyä järjestelmästä

3. Metodi palautti:

Null-viitteen

Virhe TahitiLibrary.ERR_UNABLE_TO_WRITE_REQUEST

- Vapaaehtoisten kenttien täyttäminen

Vapaaehtoisten kenttien täyttämistä tutkivat lauseet testaavat antamalla metodille syötteitä, joissa jokaisesta puuttuu vuorollaan jokin vapaaehtoinen kenttä. Tämän jälkeen modifiodun DBControl-luokan ruudulle tulostamaa SQL-lauseetta verrataan odotettuun tulokseen.

1. Annetut syötteet:

Tapaus 1: Information <tyhjä String>

Tapaus 2: DefaultDetector <tyhjä String>

Tapaus 3: DefaultPhotometricSystem <tyhjä String>

Tapaus 4: DefaultLocation <tyhjä String>

Tapaus 5: DefaultTimeStandard

2. Metodi palautti:

Tapaus 1, muodostettu SQL-lause: INSERT INTO TahitiUser (UserNumber, UserName, UserLevel, Password, Name, EMail, Status, Recommender, RecommenderEMail, Information, DefaultDetector, DefaultLocation, DefaultPhotometricSystem, DefaultTimeStandard, LastLogin, Comment) VALUES (3, 'Kyllä elämä on ihanaa',0, 'Ox4861206d28690480763dfc2e22141d4f3c0a1684', 'aapo', 'haista@paska.net', 'Professional', 'EArskan tarkka näkö', 'LOCATION: paarissahan me', 'PHOTOMETRICSYSTEM: Mikä "tun systeemi"?', 'TIMESTD: olalla oli kello!', null, null);

true

Tapaus 2, muodostettu SQL-lause: INSERT INTO TahitiUser (UserNumber, UserName, UserLevel, Password, Name, EMail, Status, Recommender, RecommenderEMail, Information, DefaultDetector, DefaultLocation, DefaultPhotometricSystem, DefaultTimeStandard, LastLogin, Comment) VALUES (4, 'Kyllä elämä on ihanaa',0, 'Oxe4261274516ea44b8a0455ca200c902c01004c0e54', 'aapo', 'haista@paska.net', 'Professional',

paarissahan me', 'PHOTOMETRICSYSTEM: Mikä ”tun systeemi?’, 'TIMESTD: olalla oli kello!', null, null);

true

Tapaus 3, muodostettu SQL-lause: INSERT INTO TahitiUser (UserNumber, UserName, UserLevel, Password, Name, EMail, Status, Recommender, RecommenderEMail, Information, DefaultDetector, DefaultLocation, DefaultPhotometricSystem, DefaultTimeStandard, LastLogin, Comment) VALUES (5, 'Kyllä elämä on ihaanaa', 0, 'Ox142f823ec43db4ecc0aac2583f2d61420fe84a3b', 'aapo', 'haista@paska.net', 'Professional', 'Epaarissahan me', null, 'TIMESTD: olalla oli kello!', null, null);

true

Tapaus 4, muodostettu SQL-lause: INSERT INTO TahitiUser (UserNumber, UserName, UserLevel, Password, Name, EMail, Status, Recommender, RecommenderEMail, Information, DefaultDetector, DefaultLocation, DefaultPhotometricSystem, DefaultTimeStandard, LastLogin, Comment) VALUES (6, 'Kyllä elämä on ihaanaa', 0, 'Oxd8669a1aa27a3c48763b2003364d73a46e7f74ce', 'aapo', 'haista@paska.net', 'Professional', 'Epaarissahan me', null, 'TIMESTD: olalla oli kello!', null, null);

true

Tapaus 5, muodostettu SQL-lause: INSERT INTO TahitiUser (UserNumber, UserName, UserLevel, Password, Name, EMail, Status, Recommender, RecommenderEMail, Information, DefaultDetector, DefaultLocation, DefaultPhotometricSystem, DefaultTimeStandard, LastLogin, Comment) VALUES (7, 'Kyllä elämä on ihaanaa', 0, 'Ox26e62216826ba4469a90474f47150243075e7a7b', 'aapo', 'haista@paska.net', 'Professional', 'Epaarissahan me', null, 'TIMESTD: olalla oli kello!', null, null);

true

- *login*

Kirjautumisen suoraviivaisen luoteen vuoksi testauskin on hyvin yksinkertaista. Testausta varten tarvitaan sellainen aineisto, että ensimmäisellä ajolla käyttäjä löytyy järjestelmästä, ja toisella ei.

1. Annetut syötteen:

Username Tahiti

Password password

2. Tietokantaan kohdistuvat oletukset:

Username Tahiti

Password Ox2a4a117c1ac82c679472031f498e24174f1ab0c6

3. Metodi palautti:

ApiUser –containerin, jossa käyttäjän Tahiti tietokannasta löytyvä informaatio.

1. Annetut syötteen:

Username Tahiti

Password 3l1t3

2. Tietokantaan kohdistuvat oletukset:

Username Tahiti

Password Ox2a4a117c1ac82c679472031f498e24174f1ab0c6

3. Metodi palautti:

Null-viitteen

Virhe TahitiLibrary.ERR_BAD_USERNAME_OR_PASSWORD

- *changePassword*

Myös salasanan vaihto on erittäin suoraviivainen operaatio. Testauksessa täytyy kokeilla vaihtoa sellaisilla syötteillä, joilla vaihdon pitäisi onnistua, ja lisäksi tilanteita, joissa annettu uusi salasana on liian lyhyt tai vanha salasana on väärä.

1. Annetut syötteet:

UserID 1

oldPassword password

newPassword tarpeekspitka

2. Tietokantaan kohdistuvat oletukset:

UserNumber 1

Password Ox2a4a117c1ac82c679472031f498e24174f1ab0c6

3. Metodi palautti:

SQL-lauseen UPDATE TahitiUser SET Password = 'Ox33614e6436ee202b024935200d164a6b383513b4
WHERE UserNumber = 1 AND Password = 'Ox2a4a117c1ac82c679472031f498e24174f1ab0c6'
true

1. Annetut syötteet:

UserID 1

oldPassword password

newPassword a

2. Tietokantaan kohdistuvat oletukset:

UserNumber 1

Password Ox2a4a117c1ac82c679472031f498e24174f1ab0c6

3. Metodi palautti:

false

Virhe TahitiLibrary.ERR_PASSWORD_TOO_SHORT

1. Annetut syötteet:

UserID 1

oldPassword hihii, arvasinko oikein?

newPassword 3l1t3

2. Tietokantaan kohdistuvat oletukset:

UserNumber 1

Password Ox2a4a117c1ac82c679472031f498e24174f1ab0c6

3. Metodi palautti:

SQL—**lauseen** UPDATE TahitiUser SET Password = 'Ox5c4cfa1e1ab25ca43a904e266ca67a6c66788a07
WHERE UserNumber = 1 AND Password = 'Ox7c4e52745a1c277c6300f0123202c0418c0ea84e';

Virheen TahitiLibrary.ERR_BAD_USERNAME_OR_PASSWORD

- *getUsers*

Koska getUsers-metodi ei toimi kuin eräänlaisena välittäjänä getUserFullille, tarvitaan sen testaamiseen vain kaksi tapausta. Toisessa getUsersFull palauttaa kannasta löytyviä tuloksia, ja toisessa palautetaan null.

1. Annetut syötteet:

UserName <tyhjä String>

Level <tyhjä String>

2. Metodi palautti (Palautettava informaatio riippuu kannassa olevasta tiedosta):

ApiUser –containerin arrayn, jossa 1 elementti seuraavalla sisällöllä:

UserNumber 1

level 2

version 0

name Administrator

userName <tyhjä String>

eMail email-address

Status Professional

Recommender suosittelija

RecommenderEMail suos-email

defaultDetector joo

defaultPhotometricSystem silmä

defaultTimeStandard kello

defaultLocation koti

information info

comment <tyhjä String>

lastLogin Thu Apr 17 11:35:36 EEST 2003

1. Annetut syötteet:

UserName joulu on taas, kattilat täynnä puuroo

Level <tyhjä String>

2. Metodi palautti (Palautettava informaatio riippuu kannassa olevasta tiedosta):

ApiUser –containerin arrayn, jonka pituus 0

- *getUsersFull*

GetUsersFull-metodi suorittaa vain yhden SQL kyselyn, ja palauttaa kyselyn tulokset kysyjälle, joten testaustan varten tarvitaan vain suorittaa kysely, jolle löytyy tuloksia (esimerkiksi voidaan hakea tyhjällä merkkijonolla), kysely, jolle ei löydy yhtään vastinetta, sekä kysely null-viitteelle. Näiden lisäksi myös versionumeron täyttämistä tulee kokeilla. Lausekattavuuden saavuttamiseksi riittää, että yhdessä haussa versionumeroa ei ole määritelty, ja toisessa on.

1. Annetut syötteen:

UserName <tyhjä String>

Level <tyhjä String>

2. Metodi palautti (Palautettava informaatio riippuu kannassa olevasta tietodosta):

ApiUser –containerin arrayn, jossa 1 elementti seuraavalla sisällöllä:

UserNumber 1

level 2

version 0

name Administrator

userName Tahiti

eMail email-address

Status Professional

Recommender suosittelija

RecommenderEMail suos-email

defaultDetector joo

defaultPhotometricSystem silmä

defaultTimeStandard kello

defaultLocation koti

information info

comment kommentti

lastLogin Thu Apr 17 11:35:36 EEST 2003

1. Annetut syötteen:

UserName joulu on taas, kattilat täynnä puuroo

Level -1

2. Metodi palautti (Palautettava informaatio riippuu kannassa olevasta tietodosta):

ApiUser –containerin arrayn, jonka pituus 0 (kun ei siellä kannassa ollu tommosia)

1. Annetut syötteen:

UserName joulu on taas, kattilat täynnä puuroo

Level <tyhjä String>

2. Metodi palautti (Palautettava informaatio riippuu kannassa olevasta tietodosta):

ApiUser –containerin arrayn, jonka pituus 0

- *setUserData*

Käyttäjätietojen muuttaminen voidaan jakaa kahteen pääluokkaan, käyttäjän itsensä tekemiin muutoksiin, sekä ylläpitäjän tekemään muutokseen. Näiden lisäksi testataan vielä tilanne, jossa joku muu tavallinen käyttäjä pyrkii muuttamaan toisen käyttäjän tietoja. Tehtyjen testien perustapaus on seuraava:

1. Annetut syötteet:

Muutoksen tekijä UserID = 2, UserLevel = 1

UserID 2

Name SuperMan

EMail newAddress@newdomain.net

Status Amateur

UserName My new loginID

Recommender Tomi

RecommenderEMail hgopispofsdg

Level 1

Version 0

DefaultDeceptor t425

DefaultPhotometricSystem PHOTOMETRICSYSTEM: Mikä 'tun systeemi?

DefaultLocation uhvoalus

DefaultTimeStandard Manne myi halvan kellon

Information Tärkeätä tietoa, on on on!

Comment Mitä, en pysty ite muuttamaan omaa kommenttia?

2. Tietokantaan kohdistuvat oletukset:

Tietokannasta tulee löytyä muutosta tekevä käyttäjä, ja kaikkien tietokenttien pitää sisältää jotain muuta informaatiota, kuin muutoksessa kantaan kirjataan. Käyttäjä tasona tulee olla 1.

3. Metodi palautti:

Muodostettava SQL-lause: INSERT INTO TahitiUser (UserNumber, UserName, UserLevel, Password, Name, EMail, Status, Recommender, RecommenderEMail, Information, DefaultDetector, DefaultLocation, DefaultPhotometricSystem, DefaultTimeStandard, LastLogin, Comment) VALUES (13, 'Kyllä elämä on ihanaa',0, 'Ox26293a567be41a043278021b6c740139d4362746', 'aapo', 'haista@paska.net', 'Professional', 'Eetu', 'Eetu', null);

true

Seuraavanlaisia testitapauksia tarvitaan, jotta lausekattavuus täyttyisi:

- Kirjautumistunnuksen muutos

Kirjautumistunnuksen muutoksessa tulee testata tilanne, jossa haluttu uusi käyttäjätunnus on jo järjestelmässä. Tällöin muutos keskeytetään.

1. Annetut syötteet:

Muutoksen tekijä UserID = 2, UserLevel = 1

UserID 2

UserName Tahiti

2. Tietokantaan kohdistuvat oletukset:

Tietokannasta tulee löytyä muutosta tekevän käyttäjän lisäksi käyttäjä Tahiti, jonka käyttäjännumero on eri, kuin 2.

3. Metodi palautti:

false

Virhe TahitiLibrary.ERR_UNABLE_TO_CHANGE_USERINFO

- Olemattoman käyttäjän vaihto

Koska ylläpitäjät pystyvät vaihtamaan myös muiden, kuin itsensä tietoja, tulee testauksessa ottaa huomioon tilanne, jossa muutettavaa käyttäjää ei löydy järjestelmästä laisinkaan. Tällöin muutosta ei luonnollisesti voida suorittaa.

1. Annetut syötteet:

Muutoksen tekijä UserID = 2, UserLevel = 2

UserID 2345234

2. Tietokantaan kohdistuvat oletukset:

Käyttäjännumeroa 2345234 ei saa löytyä kannasta.

3. Metodi palautti:

false

Virhe TahitiLibrary.ERR_UNABLE_TO_CHANGE_USERINFO

- Käyttäjätasoon kohdistuva muutos

Ylläpitäjän tekemien käyttäjätietojen muutoksen testaamiseen riittää yksi onnistuva testitapaus, jossa myös käyttäjätaso vaihtuu. Mikäli käyttäjätasoa ei vaihdeta, suoritetaan muutama rivi vähemmän, joten lausekattavan testauksen kannalta asialla ei ole merkitystä.

1. Annetut syötteet:

Muutoksen tekijä UserID = 1, UserLevel = 2

UserID 2

Name SuperMan

EMail newAddress@newdomain.net

Status Amateur

UserName My new loginID

Recommender Tomi

RecommenderEMail hgopispofsdg

Level 2

Version 0

DefaultDeceptor t425

DefaultPhotometricSystem PHOTOMETRICSYSTEM: Mikä 'tun systeemi?

DefaultLocation uhvoalus

DefaultTimeStandard Manne myi halvan kellon

Information Tärkeätä tietoa, on on on!

Comment Mitä, en pysty ite muuttamaan omaa kommenttia?

2. Tietokantaan kohdistuvat oletukset:

Tietokannasta tulee löytyä muutosta tekevä käyttäjä, ja kaikkien tietokenttien pitää sisältää jotain muuta informaatiota, kuin muutoksessa kantaan kirjataan. Käyttäjätasolla tulee olla 1.

3. Metodi palautti:

Muodostettava SQL-lause: UPDATE TahitiUser SET UserName = 'ekke', Name = 'SuperMan', EMail = 'newAddress@newdomain.net', Status = 'Professional', Version = 1, Information = 'Tärkeätä tietoa, on on on!', DefaultDetector = 't425', DefaultLocation = 'uhvoalus', DefaultPhotometricSystem = 'pullo', DefaultTimeStandard = 'Manne myi halvan kellon', Comment = 'The user is always drunk, error +/- 4532465346', Recommender = 'Tomi', RecommenderEMail = 'hgopispofsdg', UserLevel = 2 WHERE UserNumber = 2 AND Version = 0

true

- *resetPassword*

resetPasswordin lausekattavaa testausta varten täytyy yrittää nollata sellaisen käyttäjän salasanaa, jota kannasta ei löydy, normaalin nollauksen lisäksi.

- *getUser*

getUser jakautuu oikeasti kahteen metodiin, kirjautumistunnuksen ja sisäisen käyttäjänumeron perusteella tehtäviin hakuihin. Ensimmäinen, kirjautumistunnukseen perustuva haku hakee vain käyttäjän sisäisen käyttäjänumeron, ja kutsuu tämän avulla käyttäjänumeron perusteella tapahtuvaa hakua. Tätä metodia varten testiaineistoksi riittää järjestelmästä löytyvä kirjautumistunnus, sekä tunnus jota järjestelmästä ei löydy.

Varsinaisen käyttäjähaun tekevä getUser on myöskin hyvin yksinkertainen. Testiaineistoksi tarvitaan vain käyttäjännumero, joka löytyy järjestelmästä ja numero, jota järjestelmästä ei löydy.

1. Annetut syötteen:

UserID 1

2. Tietokantaan kohdistuvat oletukset:

Käyttäjännumero 1 löytyy tietokannasta

3. Metodi palautti:

ApiUser -containerin arrayn, jossa 1 elementti seuraavalla sisällöllä:

UserNumber 1
 level 2
 version 0
 name Administrator
 userName Tahiti
 eMail email-address
 Status Professional
 Recommender suosittelija
 RecommenderEMail suos-email
 defaultDetector joo
 defaultPhotometricSystem silmä
 defaultTimeStandard kello
 defaultLocation koti
 information info
 comment kommentti
 lastLogin Thu Apr 17 11:35:36 EEST 2003

1. Annetut syötteet:

UserID 2345234

2. Tietokantaan kohdistuvat oletukset:

Käyttäjänumeroa 2345234 ei saa löytyä kannasta.

3. Metodi palautti:

null-viitteen

Virhe TahitiLibrary.ERR_USER_NOT_FOUND

1. Annetut syötteet:

UserName Tahiti

2. Tietokantaan kohdistuvat oletukset:

Käyttäjä Tahiti tulee löytyä kannasta.

3. Metodi palautti:

ApiUser –containerin, jossa UserID oli 1. Tarkemmat tiedot löytyvät käyttäjänumeron perusteella tehdystä testistä.

1. Annetut syötteet:

UserName Joulupukki juupajee

2. Tietokantaan kohdistuvat oletukset:

Käyttäjää "Joulupukki juupajee" ei saa löytyä kannasta.

3. Metodi palautti:

null-viitteen

Virhe TahitiLibrary.ERR_USER_NOT_FOUND

- *removeUser*

Poistoa testattaessa tarvitaan tietokantaan sellainen aineisto, jossa ensimmäisessä ajossa käyttäjän toiminta ei ole aiheuttanut yhtään merkintään järjestelmän lokiin ja toisessa tapauksessa tällainen merkintä täytyy löytyä. Lisäksi lausekattavuuden saamiseksi tarvitaan tapaus, jossa poistaminen ei onnistu.

1. Annetut syötteet:

UserID 1

2. Tietokantaan kohdistuvat oletukset:

Käyttäjää #1 on tehnyt järjestelmässä jotain, joka on aiheuttanut lokimerkintöjä.

Käyttäjän versionumero on nolla.

3. Metodi palautti:

SQL-lauseen UPDATE TahitiUser SET UserLevel = -1, Version = 1 WHERE User-
Number = 1 AND Version = 0;

true

1. Annetut syötteet:

UserID 13

2. Tietokantaan kohdistuvat oletukset:

Käyttäjää #13 (versio 0), ei ole aiheuttanut yhtään merkintään järjestelmän lokiin.

3. Metodi palautti:

SQL-lauseen DELETE FROM TahitiUser WHERE UserNumber = 13 AND Version
= 0;

true

1. Annetut syötteet:

UserID 234234

2. Tietokantaan kohdistuvat oletukset:

Käyttäjää #234234 ei ole järjestelmässä

3. Metodi palautti:

SQL-lauseen DELETE FROM TahitiUser WHERE UserNumber = 234234 AND Ver-
sion = 0;

False

4 Integraatiotestaus

Integraatiotestaus suoritettiin ilman käyttöliittymää ns. Black-box testauksena bottom-up -mallin mukaisesti. Integraatiotestauksessa pyrittiin siihen, ettei sama henkilö suorita sekä moduulitestauksista että integraatiotestauksista tekemälleen luokalle. Testausvastuut TahitiApin metodeista arvottiin testaajien kesken.

Black-Box -testaustyyppistä joustetaan kuitenkin siinä määrin, ettei jokaisen merkkijonoja parametrinaan ottavan metodin yhteydessä testata erikseen erilaisia SQL-injektio tapauksia. Integraatiotestauksen yhteydessä luetaan siihen, että handler-luokat suorittavat jokaiselle String-tyyppiselle muuttujalle SystemHandlerin deSQL-metodin, jolloin mahdolliset SQL-lauseet ja erikoismerkit tehdään vaarattomiksi. Tämän metodin toiminta on jo testattu moduulitestauksen yhteydessä, ja se oletetaan toimivaksi. Validointitestauksen yhteydessä tätä toimintaa testataan pistokein.

Tahiti-Apin integraatiotestaus suoritetaan bottom-up mallin mukaisesti. Tämä tarkoittaa sitä, että aluksi testataan Apin alapuolella olevat metodit ja luokat, ja vasta tämän jälkeen varsinaisen Apin toiminta yhdessä eri toiminnot toteuttavien metodien ja luokkien kanssa. Alapuolella olevien metodien toiminta on kuitenkin suoritettu jo moduulitestauksen yhteydessä, joten testaaminen voidaan aloittaa suoraan kokonaisen TahitiApin kanssa.

Seuraavissa aliluvuissa käsitellään jokaisen eri metodin testaukseen tarvittavat aineistot. Aineistot on pyritty muodostamaan siten, että mahdollisimman moni metodin oikea käyttötapa tulisi testatuksi, sekä mahdolliset virhetilanteet saataisiin selville. Virhetilojen löytämiseksi metodien toimintaa kokeillaan raja-arvoilla, jotka ovat sallittujen rajojen molemmin puolin. Lisäksi luokkien toimintaa kokeillaan selvästi virheellisillä arvoilla. Myös testien tulokset ovat nähtävissä.

4.1 getAsteroids

Metodi testataan aluksi *tyhjällä haulla*, jonka ainoana ehtona on, että valokäyrien lukumäärä on yksi tai useampia. Tämän jälkeen tutkitaan tapaus, jossa valokäyrien lukumääräksi asetetaan arvo, joka on pienempi kuin yksi.

Tyhjän haun testauksen jälkeen hausta testataan vuorotellen alla luetellut tapaukset. Jokaisen tapauksen yhteydessä, jossa oletetaan haun toimivan testataan sekä tapaus, että tietokannasta löytyy ehdon täyttävä tapaus, että tapaus, jossa hakuehto ei täyty millään kannan alkiolla.

- Hakuehdoksi asetetaan null-viite
- Johonkin Suunnitteludokumentin versio 1.3 luvusta 3.2 löytyvään kenttään asetetaan null-viite
- Suunnitteludokumentin versio 1.3 luvusta 3.2 löytyvien lukuarvoehtojen mukaiset ylä- ja alarajat sekä ylärajat ylittävät ja alarajat alittavat arvot.
- Validi haku jokaisen suunnitteludokumentin versio 1.3 luvusta 3.2 löytyvän kentän kohdalla erikseen. Testauksessa suoritetaan sekä tapaus, jossa haun tulisi tuottaa tulosta, että tapaus, jossa tuloksena on tyhjä joukko.

- Haku, jossa kaikki kentät on täytetty. Sekä tyhjä, että tuottoisa tapaus.
- Erikseen testataan vielä tapaus, jossa filttareita on hakuehdoissa useampia kuin yksi

Seuraavat testit tehtiin:

Testataan tyhjä haku.

- Annetut syötteet

Asteroid Name

Asteroid Number

ObsTimeStart

ObsTimeEnd

Observer

Observing Site

MinimumLCs

MinimumDPs

Min/Max PA

Min/Max Lat

Min/Max Long

Min/Max HDist

Min/Max GDist

Abs. Photometry

LT Corrected

Information

Filters

Detectors

Detectors other

- Metodi palautti

asteroidID 1

number 234

numberOfLightcurves 1

numberOfObservers 1

name Ida

designation null

lastObservation Sat Oct 27 03:12:28 EET 1990

asteroidID 2

number 25143
numberOfLightcurves 2
numberOfObservers 1
name null
designation 1998 SF36
lastObservation Sat Feb 03 06:00:53 EET 2001

Testataan haku, jossa syötteenä null-viite.

- Annetut syötteenä
 - AsteroidQueryContainer** null-viite
- Metodi palautti
 - Virhe 0** Operation aborted, mandantory data missing, 11

Testataan haku, jossa null-viite containerin sisällä.

- Annetut syötteenä
 - Asteroid Name** null
 - Asteroid Number** null
 - ObsTimeStart** null
 - ObsTimeEnd** null
 - Observer** null
 - Observing Site**
 - MinimumLCs** null
 - MinimumDPs** null
 - Min/Max PA** null /
 - Min/Max Lat** /
 - Min/Max Long** /
 - Min/Max HDist** null /
 - Min/Max GDist** / null
 - Abs. Photometry**
 - LT Corrected**
 - Information**
 - Filters**
 - Detectors**
 - Detectors other**

- Metodi palautti

asteroidID 1
number 234
numberOfLightcurves 1
numberOfObservers 1
name Ida
designation null
lastObservation Sat Oct 27 03:12:28 EET 1990
asteroidID 2
number 25143
numberOfLightcurves 2
numberOfObservers 1
name null
designation 1998 SF36
lastObservation Sat Feb 03 06:00:53 EET 2001

Testataan haku, jossa useita filterinä null.

- Annetut syötteet

Asteroid Name
Asteroid Number
ObsTimeStart
ObsTimeEnd
Observer
Observing Site
MinimumLCs
MinimumDPs
Min/Max PA /
Min/Max Lat /
Min/Max Long /
Min/Max HDist /
Min/Max GDist /
Abs. Photometry
LT Corrected
Information
Filters null,

Detectors**Detectors other**

- Metodi palautti

Virhe 0 Error parsing numerical data, 12

Testataan tapaus, jossa vaihekulma < 0.

- Annetut syötteet

Asteroid Name

Asteroid Number

ObsTimeStart

ObsTimeEnd

Observer

Observing Site

MinimumLCs

MinimumDPs

Min/Max PA -1 /

Min/Max Lat /

Min/Max Long /

Min/Max HDist /

Min/Max GDist /

Abs. Photometry

LT Corrected

Information

Filters

Detectors

Detectors other

- Metodi palautti

asteroidID 1

number 234

numberOfLightcurves 1

numberOfObservers 1

name Ida

designation null

lastObservation Sat Oct 27 03:12:28 EET 1990

Testataan tapaus, jossa useita filttäreitä.

- Annetut syötteen

Asteroid Name

Asteroid Number

ObsTimeStart

ObsTimeEnd

Observer

Observing Site

MinimumLCs

MinimumDPs

Min/Max PA /

Min/Max Lat /

Min/Max Long /

Min/Max HDist /

Min/Max GDist /

Abs. Photometry

LT Corrected

Information

Filters 2, 4,

Detectors

Detectors other

- Tietokantaan kohdistuvat oletukset

Kannassa on idalla yksi valokäyrä, jossa filttəri 4 ja 1998 SF36:lla kaksi välökäyrää, joista toinen on filttəriä 4 ja toinen filttəriä 2

- Metodi palautti

asteroidID 1

number 234

numberOfLightcurves 1

numberOfObservers 1

name Ida

designation null

lastObservation Sat Oct 27 03:12:28 EET 1990

asteroidID 2

number 25143

numberOfLightcurves 2
numberOfObservers 1
name null
designation 1998 SF36
lastObservation Sat Feb 03 06:00:53 EET 2001

Testataan tapaus, jossa yksi filtteri.

- Annetut syötteen

Asteroid Name

Asteroid Number

ObsTimeStart

ObsTimeEnd

Observer

Observing Site

MinimumLCs

MinimumDPs

Min/Max PA /

Min/Max Lat /

Min/Max Long /

Min/Max HDist /

Min/Max GDist /

Abs. Photometry

LT Corrected

Information

Filters 4,

Detectors

Detectors other

- Tietokantaan kohdistuvat oletukset

Kannassa on idalla yksi valokäyrä, jossa filtteri 4 ja 1998 SF36:lla kaksi välokäyrää, joista toinen on filtteriä 4 ja toinen filtteriä 2

- Metodi palautti

asteroidID 1

number 234

numberOfLightcurves 1

numberOfObservers 1

name Ida
designation null
lastObservation Sat Oct 27 03:12:28 EET 1990
asteroidID 2
number 25143
numberOfLightcurves 1
numberOfObservers 1
name null
designation 1998 SF36
lastObservation Fri Feb 02 05:32:42 EET 2001

Testataan tapaus, jossa asteroidin nimi on validi.

- Annetut syötteen

Asteroid Name Ida

Asteroid Number

ObsTimeStart

ObsTimeEnd

Observer

Observing Site

MinimumLCs

MinimumDPs

Min/Max PA /

Min/Max Lat /

Min/Max Long /

Min/Max HDist /

Min/Max GDist /

Abs. Photometry

LT Corrected

Information

Filters

Detectors

Detectors other

- Tietokantaan kohdistuvat oletukset
 Kannassa on idalla yksi valokäyrä
- Metodi palautti

asteroidID 1
number 234
numberOfLightcurves 1
numberOfObservers 1
name Ida
designation null
lastObservation Sat Oct 27 03:12:28 EET 1990

Testataan tapaus, jossa asteroidin nimi on Huuheli.

- Annetut syötteet

Asteroid Name Huuheli

Asteroid Number

ObsTimeStart

ObsTimeEnd

Observer

Observing Site

MinimumLCs

MinimumDPs

Min/Max PA /

Min/Max Lat /

Min/Max Long /

Min/Max HDist /

Min/Max GDist /

Abs. Photometry

LT Corrected

Information

Filters

Detectors

Detectors other

- Tietokantaan kohdistuvat oletukset
Kannassa ei ole Huuheli-nimistä asteroidia

- Metodi palautti:

Tyhjä taulukko

Testataan tapaus, jossa asteroidin numero on kannassa.

- Annetut syötteet

Asteroid Name

Asteroid Number 25143

ObsTimeStart

ObsTimeEnd

Observer

Observing Site

MinimumLCs

MinimumDPs

Min/Max PA /

Min/Max Lat /

Min/Max Long /

Min/Max HDist /

Min/Max GDist /

Abs. Photometry

LT Corrected

Information

Filters

Detectors

Detectors other

- Tietokantaan kohdistuvat oletukset

Kannassa on asteroidi, jonka numero on 25143

- Metodi palautti:

asteroidID 2

number 25143

numberOfLightcurves 2

numberOfObservers 1

name null

designation 1998 SF36

lastObservation Sat Feb 03 06:00:53 EET 2001

Testataan tapaus, jossa asteroidin numeroa ei ole kannassa.

- Annetut syötteet

Asteroid Name

Asteroid Number 3

ObsTimeStart

ObsTimeEnd

Observer

Observing Site

MinimumLCs

MinimumDPs

Min/Max PA /

Min/Max Lat /

Min/Max Long /

Min/Max HDist /

Min/Max GDist /

Abs. Photometry

LT Corrected

Information

Filters

Detectors

Detectors other

- Tietokantaan kohdistuvat oletukset
Kannassa ei ole asteroidia, jonka numero on 3
- Metodi palautti

Tyhjä taulukko

Testataan tapaus, jossa kaikki asteroidin kentät on täytetty.

- Annetut syötteet

Asteroid Name Ida

Asteroid Number 234

ObsTimeStart 1.1.1990

ObsTimeEnd 1.12.1990

Observer Binzel

Observing Site

MinimumLCs 1

MinimumDPs 3

Min/Max PA 15 / 17

Min/Max Lat 0.7 / 0.7

Min/Max Long 336 / 337

Min/Max HDist 2.5 / 3

Min/Max GDist 2 / 2.2

Abs. Photometry false

LT Corrected

Information

Filters 4,

Detectors Photo,

Detectors other

- Tietokantaan kohdistuvat oletukset
Kannassa on ehtoihin sopiva asteroidi
- Metodi palautti

asteroidID 1

number 234

numberOfLightcurves 1

numberOfObservers 1

name Ida

designation null

lastObservation Sat Oct 27 03:12:28 EET 1990

4.2 getLightcurves

Metodi testataan samoin kuin getAsteroids. Aluksi siis testataan *tyhjä haku*, jonka ainoana ehtona on, että valokäyrien lukumäärä on yksi tai useampia. Tämän jälkeen tutkitaan tapaus, jossa valokäyrien lukumääräksi asetetaan arvo, joka on pienempi kuin yksi.

Tämän jälkeen hausta testataan vuorotellen alla luetellut tapaukset.

- Hakuehdoksi asetetaan null-viite
- Johonkin Suunnitteludokumentin luvusta 3.2 löytyvään kenttään asetetaan null-viite
- Suunnitteludokumentin luvusta 3.2 löytyvien lukuarvoehtojen mukaiset ylä- ja alarajat sekä ylärajat ylittävät ja alarajat alittavat arvot.
- Validi haku joidenkin suunnitteludokumentin luvusta 3.2 löytyvien yksittäisten kenttien osalta erikseen. Testauksessa suoritetaan sekä tapaus, jossa haun tulisi tuottaa tulosta, että tapaus, jossa tuloksena on tyhjä joukko.
- Erikseen testataan vielä tapaus, jossa filttereitä on hakuehdoissa useampia kuin yksi

Seuraavat testit tehtiin:

- Annetut syötteen containerissa:
MinimumLightcurves 1
- Palauttaa:
Kaikki tietokannan valokäyrät.
- Annetut syötteen containerissa:
MinimumLightcurves -1
- Palauttaa:
Kaikki tietokannan valokäyrät.
- Annetut syötteen containerissa:
Kaikki arvot null
- Palauttaa:
Kaikki tietokannan valokäyrät.
- Annetut syötteen containerissa:
MinimumLightcurves 1
setMinimumPhaseAngle 0
setMaximumPhaseAngle 0
setMinimumDistanceFromSun 0
setMaximumDistanceFromSun 0
setMinimumDistanceFromEarth 0
setMaximumDistanceFromEarth 0
setMinimumLongitude -90
setMaximumLongitude -90
setMinimumLatitude -90
setMaximumLatitude -90
setMinimumDataPoints 1
- Palauttaa:
Tyhjä (ei haun ehdot täyttäviä valokäyriä)
- Annetut syötteen containerissa:
MinimumseAngle 180

MaximumPhaseAngle 180

MinimumLongitude 90

MaximumLongitude 90

MinimumLatitude 90

MaximumLatitude 90

- Palauttaa:

Tyhjä (ei haun ehdot täyttäviä valokäyriä)

- Annetut syötteet containerissa:

AsteroidNumber -1

MinimumLightcurves 0

setMinimumPhaseAngle -1

setMaximumPhaseAngle -1

setMinimumDistanceFromSun -1

setMaximumDistanceFromSun -1

setMinimumDistanceFromEarth -1

setMaximumDistanceFromEarth -1

setMinimumLongitude -91

setMaximumLongitude -91

setMinimumLatitude -91

setMaximumLatitude -91

setMinimumDataPoints 0

- Palauttaa:

Tyhjä

- Annetut syötteet containerissa:

MinimumPhaseAngle 181

MaximumPhaseAngle 181

MinimumLongitude 91

MaximumLongitude 91

MinimumLatitude 91

MaximumLatitude 91

- Palauttaa:

Tyhjä

- Annetut syötteet containerissa:

AsteroidName Ida

- Palauttaa:

Idaan liittyvät valokäyrät

- Annetut syötteet containerissa:

AsteroidName "Minua ei ole"

- Palauttaa:

Tyhjä

- Annetut syötteet containerissa:

AsteroidNumber 22

- Palauttaa:

Asteroidiin numero 22 liittyvät valokäyrätiedot

- Annetut syötteet containerissa:

AsteroidNumber 99999999999

- Palauttaa:

Tyhjä

- Annetut syötteet containerissa:

Observer 1036_Ganymed.apc+

- Palauttaa:

Kyseiseen Observeriin liittyvät valokäyrätiedot

- Annetut syötteet containerissa:

Observer "Ei ole minua"

- Palauttaa:

Tyhjä

- Annetut syötteet containerissa:

Detector Photoelectric

- Palauttaa:

Valokäyrätiedot, joissa detector photoelectric

- Annetut syötteen containerissa:

Detector Eioleminua

- Palauttaa:

Tyhjä

- Annetut syötteen containerissa:

Filter 4

- Palauttaa:

Valokäyrätiedot, joissa filter on 4

- Annetut syötteen containerissa:

Filter 1

- Palauttaa:

Tyhjä (ei löydy tietokannasta)

- Annetut syötteen containerissa:

Location "W 117.68 +34.38 Table Mountain Observatory, California, USA"

- Palauttaa:

Valokäyrätiedot, joissa location on yllä mainittu

- Annetut syötteen containerissa:

Location Muaeiole

- Palauttaa:

Tyhjä

- Annetut syötteen containerissa:

Filter 2, 4

- Palauttaa:

Valokäyrätiedot, joissa filteri 2 tai 4

4.3 getTrajectory

Metodin vastaanottama long-tyyppinen muuttuja ei anna testaukselle paljoakaan riippumavaraa. Metodin toimintaan testataan virheellisellä syötteellä (negatiivinen asteroidinnumero), järjestelmästä löytyvällä asteroidinumerolla sekä oikealla, positiivisella asteroidnumerolla, jota järjestelmästä ei löydy. Testauksessa on olettu, että Asteroidille #1 löytyy rataelementtejä, eikä asteroideja -432 ja 4326653 ole olemassa. Lisäksi asteroidilla #2 on tietokannassa 2 rataelementtitietoa.

- Annetut syötteen:

AsteroidID -432

- Api palautti:

Trajectory-taulukon , jossa ei yhtään alkioa.

- Annetut syötteen:

AsteroidID 43266534

- Api palautti:

Trajectory-taulukon , jossa ei yhtään alkioa.

- Annetut syötteen:

AsteroidID 1

- Api palautti:

Trajectory-taulukon , jossa seuraava informaatiota:

```
Elementti 0, AsteroidID 1
Elementti 0, TrajectoryID 1
Elementti 0, Axis 2.8600955
Elementti 0, Eccentricity 0.04667024
Elementti 0, Inclination 1.13806
Elementti 0, Argument 108.99459
Elementti 0, Longitude 324.20683
Elementti 0, Anomaly 143.3734946
Elementti 0, Reference JPL#30
Elementti 0, Epoch Mon Jan 06 02:00:00 EET 2003
Elementti 0, EntryDate 2003-04-28
```

- Annetut syötteen:

AsteroidID 2

- Api palautti:

Trajectory-taulukon , jossa seuraava informaatiota:

```

Elementti 0, AsteroidID 2
Elementti 0, TrajectoryID 2
Elementti 0, Axis 1.322691
Elementti 0, Eccentricity 0.27961274
Elementti 0, Inclination 1.72799
Elementti 0, Argument 161.02783
Elementti 0, Longitude 70.92064
Elementti 0, Anomaly 107.8134613
Elementti 0, Reference Kaasalainen
Elementti 0, Epoch Mon Jan 01 02:00:00 EET 2001
Elementti 0, EntryDate 2003-04-28
Elementti 1, AsteroidID 2
Elementti 1, TrajectoryID 1
Elementti 1, Axis 1.3228213
Elementti 1, Eccentricity 0.27948049
Elementti 1, Inclination 1.72805
Elementti 1, Argument 161.01582
Elementti 1, Longitude 70.92126
Elementti 1, Anomaly 237.4062178
Elementti 1, Reference JPL#63
Elementti 1, Epoch Sat Jan 05 02:00:00 EET 2002
Elementti 1, EntryDate 2003-04-28

```

4.4 makeDataFile

Metodi testataan ensin antamalla syötteenä null-arvot vuorotellen kuhunkin taulukkopaikkaan. Tämän jälkeen testataan tapaukset:

- Taulukoissa on eri määrät alkioita
- Kussakin taulukossa on yksi oikeellinen alkio ja system on TahitiLibrary.SYSTEM_MAGNITUDE
- Kussakin taulukossa on yksi oikeellinen alkio ja system on TahitiLibrary.SYSTEM_INTENSITY
- Kussakin taulukossa on yksi oikeellinen alkio ja system on -1
- Kussakin taulukossa on yksi alkio, ja lightcurve-aulun alkio on null-viite
- Kussakin taulukossa on yksi alkio, ja earthTrajectory-aulun alkio on null-viite
- Kussakin taulukossa on yksi alkio, ja asteroidTrajectory-aulun alkio on null-viite
- Kussakin taulukossa on yksi alkio, ja molempien Trajectory-aulujen alkio on null-viite
- Tauluissa on nolla alkioita
- Taulussa lightcurve on useampia kuin yksi alkio ja molempien Trajectory taulujen jokin alkio on null-viite

Seuraavat tapaukset testattiin:

Testataan tapaukset, joissa jokin taulukoista on null-viite

- Annetut syötteet

lightcurves null

asteroidTrajectories Validi yhden trajectoryn taulukko

earthTrajectories Validi yhden trajectoryn taulukko

- Metodi palautti

Virhe 0 There was missing data when attempting to make the output file, 42

Testataan tapaukset, joissa jokin taulukoista on null-viite

- Annetut syötteet

lightcurves Validi valokäyrätaulukko, jossa yksi valokäyrä

asteroidTrajectories null

earthTrajectories Validi yhden trajectoryn taulukko

- Metodi palautti

Virhe 0 Missing trajectory information, 47

Testataan tapaus, jossa lightcurve-taulukossa kaksi alkia ja trajectory-tauluissa yhden.

- Annetut syötteet

lightcurves Validi valokäyrätaulukko, jossa yksi kaksi valokäyrää

asteroidTrajectories Validi yhden trajectoryn taulukko

earthTrajectories Validi yhden trajectoryn taulukko

- Metodi palautti:

Virhe 0 Output error: parameter length mismatch, 46

Testataan tapaus, jossa system on jotakin muuta kuin intensity tai magnitudi

- Annetut syötteet

ApiLightcurve

asteroidID 2

lightcurveID 1

defaultTrajectoryID 1

earthTrajectoryID 1

submitter 1

numberOfPoints 5
filter 4
version 1
phaseAngle 0.0
distanceFromSun 0.0
distanceFromEarth 0.0
longitude 0.0
latitude 0.0
absolutePhotometry false
lightTimeCorrected true
deleted true
location TKTL
observer J. Muhonen
information 5 ensimmäistä pistettä valokäyrästä 1
detector Detector1
photometricSystem Photo1
timeStandard My clock
reference null
observationTime Fri Feb 02 05:32:42 EET 2001
entryDate Sat May 03 16:10:15 EEST 2003
dataPoints (Fri Feb 02 05:32:42 EET 2001, 18.344,+0.027,eX:0.274172392,eY:-0.0544258374,eZ:-0.0347286299,aX:0.951738642,aY:-0.770087699,aZ:-0.034729127), (Fri Feb 02 05:42:47 EET 2001, 18.371,+0.041,eX:0.274089109,eY:-0.0543880879,eZ:-0.0347290925,aX:0.951821398,aY:-0.769895325,aZ:-0.0347295894), (Fri Feb 02 05:51:25 EET 2001, 18.158,+0.017,eX:0.274089109,eY:-0.0543880879,eZ:-0.0347290925,aX:0.951821398,aY:-0.769895325,aZ:-0.0347295894), (Fri Feb 02 06:10:08 EET 2001, 17.949,+0.0090,eX:0.274005836,eY:-0.0543503547,eZ:-0.0347295541,aX:0.951821398,aY:-0.769895325,aZ:-0.0347295894), (Fri Feb 02 06:25:59 EET 2001, 17.833,+0.0080,eX:0.273935381,eY:-0.0543184395,eZ:-0.0347299439,aX:0.95197411,aY:-0.769540118,aZ:-0.0347304405),
ApiTrajectory
trajectoryID 0
epoch Thu Oct 18 03:00:00 EEST 2001
axis 1.322691
eccentricity 0.27961274
inclination 1.72799
argument 161.02783
longitude 70.92064

anomaly 107.8134613
reference null
entryDate null
ApiTrajectory
trajectoryID 0
epoch Sat Jan 01 14:00:00 EET 2000
axis 1.00000011
eccentricity 0.01671022
inclination 5.0E-5
argument 114.20783
longitude -11.26064
anomaly -2.48284
reference null
entryDate null
system -1

- Metodi palautti

String 1
 5 0

0.9511963926510935, -0.7713453118296646, -0.034726067228701085
 0.2747162308833505, -0.05467376758899578, -0.03472556913181759
 0.0, 18.344
 0.9512410625394295, -0.7712417678914776, -0.034726319732357736
 0.2746713367815873, -0.05465337425255912, -0.03472582172190229
 0.007002314552664757, 18.371
 0.9512793028612717, -0.771153108902268, -0.0347265357078345
 0.2746329007087924, -0.05463591730635764, -0.03472603777138462
 0.012997685000300407, 18.158
 0.9513621872064765, -0.7709608850074681, -0.03472700324276473
 0.2745495800088886, -0.05459808347830153, -0.034726505466774464
 0.025995370000600815, 17.949
 0.9514323566608737, -0.7707980859480545, -0.034727398431052756
 0.27447902821297254, -0.054566057185149286, -0.034726900790966256
 0.03700231481343508, 17.833

Testataan tapaus, jossa lightcurvessa on valokäyrä, mutta ratatietoja ei ole annettu

- Annetut syötteet

ApiLightcurve

asteroidID 2

lightcurveID 1

defaultTrajectoryID 1

earthTrajectoryID 1

submitter 1

numberOfPoints 5

filter 4

version 1

phaseAngle 0.0

distanceFromSun 0.0

distanceFromEarth 0.0

longitude 0.0

latitude 0.0

absolutePhotometry false

lightTimeCorrected true

deleted true

location TKTL

observer J. Muhonen

information 5 ensimmäistä pistettä valokäyrästä 1

detector Detector1

photometricSystem Photo1

timeStandard My clock

reference null

observationTime: Fri Feb 02 05:32:42 EET 2001

entryDate Sat May 03 16:10:15 EEST 2003

dataPoints (Fri Feb 02 05:32:42 EET 2001, 18.344,+0.027,eX:0.274172392,eY:-0.0544258374,eZ:-0.0347286299,aX:0.951738642,aY:-0.770087699,aZ:-0.034729127), (Fri Feb 02 05:42:47 EET 2001, 18.371,+0.041,eX:0.274089109,eY:-0.0543880879,eZ:-0.0347290925,aX:0.951821398,aY:-0.769895325,aZ:-0.0347295894), (Fri Feb 02 05:51:25 EET 2001, 18.158,+0.017,eX:0.274089109,eY:-0.0543880879,eZ:-0.0347290925,aX:0.951821398,aY:-0.769895325,aZ:-0.0347295894), (Fri Feb 02 06:10:08 EET 2001, 17.949,+0.0090,eX:0.274005836,eY:-0.0543503547,eZ:-0.0347295541,aX:0.951821398,aY:-0.76970293,aZ:-0.0347300508), (Fri Feb 02 06:25:59 EET 2001, 17.833,+0.0080,eX:0.273935381,eY:-0.0543184395,eZ:-0.0347299439,aX:0.95197411,aY:-0.769540118,aZ:-0.0347304405),

asteroidTrajectories Validi yhden trajectoryn taulukko, jossa alkio null-viite

earthTrajectories Validi yhden trajectoryn taulukko, jossa alkio null-viite

system TahitiLibrary.SYSTEM_INTENSITY

- Metodi palautti

String 1

5 0

```
-0.951738642, 0.770087699, 0.034729127
-0.67756625, 0.7156618616, 4.970999999964754E-7
0.0 0.004596211446660012
-0.951821398, 0.769895325, 0.0347295894
-0.6777322889999999, 0.7155072371, 4.968999999938051E-7
0.007002314552664757 0.004483322704327333
-0.951821398, 0.769895325, 0.0347295894
-0.6777322889999999, 0.7155072371, 4.968999999938051E-7
0.012997685000300407 0.0054550658817037285
-0.951904128, 0.76970293, 0.0347300508
-0.6778982920000001, 0.7153525753, 4.967000000050126E-7
0.025995370000600815 0.006613022494817659
-0.95197411, 0.769540118, 0.0347304405
-0.6780387290000001, 0.7152216785, 4.966000000036774E-7
0.03700231481343508 0.007358681396510455
```

4.5 makeRawData

Metodi testataan siten, että joku syötteistä on null; ja siten, että syötenä annettussa taulukossa tai AsteroidContainerissa on null-arvoja. Lisäksi metodia testataan vielä valideilla syötearvoilla.

Metodia testattiin seuraavilla tapauksilla:

- Annetut syötteen:

asteroid null

curves null

- Palautti:

null

- Annetut syötteen:

asteroid AsteroidID 123, Number 223, Name null, Designation null, NumberOfLightcurves
2, NumberOfObservers 1, lastObservation null

curves {null,null}

- Palautti:

(tyhjä String)

- Annetut syötteet:

asteroid AsteroidID 123L, Number 223, Name Pödö, Designation 1998 SF36,NumberOfLightcurves 2, NumberOfObservers 1, lastObservation new Date

curves {Lightcurve1,Lightcurve2} Valokäyrien arvot ovat luonnin yhteydessä alustetut arvot

- Palautti:

```
*****
OBJECT   : 223, 1998 SF36, Pödö
LIGHTCURVE   : 0
LT CORRECTED   : false
ABS PHOTOMETRY : false
*****
GEOMETRY:
ASPECT DATA   : 0.0 0.0 0.0 0.0 0.0
DATA:
*****
OBJECT   : 223, 1998 SF36, Pödö
LIGHTCURVE   : 0
LT CORRECTED   : false
ABS PHOTOMETRY : false
*****
GEOMETRY:
ASPECT DATA   : 0.0 0.0 0.0 0.0 0.0
DATA:
```

4.6 registrationRequest

Koska rekisteröintipyyntöön suorittamiseen ei tarvita aikaisempia oikeuksia, keskitytään integraatiotestauksessa vain UserFormContainer-containerin sisältämien kenttien arvoihin, ja näiden arvojen eri variaatioihin. Normaalin käyttäjäpyynnön, jossa kaikki pakolliset kentät ovat täyttyinä testataan metodin toimintaa tilanteissa, joissa jokin pakollisista kentistä on null ja/tai tyhjä merkijono. Lisäksi status-kenttää kokeillaan syöttää järjettömiä statuksia, joita järjestelmän ei tulisi hyväksyä. Tehdyissä testeissä oletuksena on ollut, että tietokannassa ei ole testitapaus [01-08] nimisiä käyttäjiä.

Ensimmäinen testitapaus. Pakolliset kentät täytetty, vapaaehtoiset tyhjiä

- Annetut syötteet:

UserID null

Name RegReq

EMail postmaster@localhost
Status Amateur
UserName testitapaus 01
Recommender Suosittelija
RecommenderEMail suosittelija@mail
DefaultDetector <tyhjä String>
DefaultLocation null
DefaultPhotometricSystem null
DefaultTimeStandard null
Information <tyhjä String>

- Api palautti:

true

Kokeillaan samaa uudestaan. Koska ei ole kirjaututtu sisään, ei pitäisi onnistua

- Annetut syötteet:

User Käyttäjä ei ole kirjautunut sisään.

- Api palautti:

false

Virhe Action not allowed (#2)

Sitten kokeillaan pakollisten tietojen puuttumisia.

- Annetut syötteet:

Tapaus 1, Name <tyhjä String>

Tapaus 2, EMail null

Tapaus 3, UserName <tyhjä String>

Tapaus 4, Recommender null

Tapaus 5, RecommenderEMail <tyhjä String>

- Api palautti:

false

Virhe Operation aborted, mandantory data missing (#11)

Kokeillaan järjetöntä arvoa Status-kentässä

- Annetut syötteet:

Status King_of_the_world

- Api palautti:

false

Virhe Incorrect parameter dataformat or value (#13)

Metodin toimintaa pitää myös testata myös vapaaehtoisten kenttien ollessa täytettyinä. Koska vapaaehtoisia kenttiä on viisi kappaletta, ei kaikkia erilaisia täyttöyhdistelmiä tulla erikseen testaamaan (tällöin testitapauksia olisi pelkästään näitä varten $5! (=120)$). Sen sijaan vapaaehtoisten kenttien toimintaa tulla testaamaan kuudella eri testillä, jossa viidessä ensimmäisessä yksi jokaisesta vapaaehtoisesta kentästä on täytettynä, ja kuudennessa tapauksessa kaikki kentät sisältävät tietoa.

- Annetut syötteet:

UserName testitapaus 02

DefaultDetector DETECTOR

DefaultLocation null

DefaultPhotometricSystem null

DefaultTimeStandard null

Information <tyhjä String>

- Api palautti:

true

- Annetut syötteet:

UserName testitapaus 03

DefaultDetector null

DefaultLocation LOCATION

DefaultPhotometricSystem null

DefaultTimeStandard null

Information <tyhjä String>

- Api palautti:

true

- Annetut syötteet:

UserName testitapaus 04

DefaultDetector null

DefaultLocation <tyhjä String>

DefaultPhotometricSystem PHOTOMETRICSYSTEM

DefaultTimeStandard null

Information <tyhjä String>

- Api palautti:

true

- Annetut syötteet:

UserName testitapaus 05

DefaultDetector null

DefaultLocation <tyhjä String>

DefaultPhotometricSystem null

DefaultTimeStandard TIMESTANDARD

Information <tyhjä String>

- Api palautti:

true

- Annetut syötteet:

UserName testitapaus 06

DefaultDetector null

DefaultLocation <tyhjä String>

DefaultPhotometricSystem null

DefaultTimeStandard <tyhjä String>

Information INFORMATION

- Api palautti:

true

- Annetut syötteet:

UserName testitapaus 07

DefaultDetector DETECTOR

DefaultLocation LOCATION

DefaultPhotometricSystem PHOTOMETRICSYSTEM

DefaultTimeStandard TIMESTANDARD

Information INFORMATION

- Api palautti:

true

Myös tilannetta, jossa haluttu kirjautumistunnus on jo käytössä tulee testata. Tällöin metodin pitäisi ilmoittaa käytössä olevasta tunnuksesta, sekä suorituksen keskeytyä. Muita tällaisia tilanteita, jotka vaativat erityistä testausta ovat sellaiset, jossa jokin kentistä, joista metodin ei tulisi välittää, ovat täytettyinä. Tällaisia kenttiä ovat

- **UserName** (Järjestelmän sisäinen tunnus)
- **Level** (Käyttäjätaso, uusilla käyttäjillä aina nolla)
- **Comment** (Ylläpitäjän käyttäjästä antama kommentti)
- **Version** (Versionumero, uusilla käyttäjäpyynnöillä aina nolla)

Tämä toiminnallisuus voidaan testata tekemällä käyttäjärekisteröintipyyntö, jossa kaikki edellämainitut kentät ovat täytettyinä. Toivottu tulos on, että metodi kirjaa käyttäjäpyynnön tietokantaan, mutta ei välitä edellämainittujen kenttien sisällöstä.

- Annetut syötteet:

UserName testitapaus 07

DefaultDetector DETECTOR

DefaultLocation LOCATION

DefaultPhotometricSystem PHOTOMETRICSYSTEM

DefaultTimeStandard TIMESTANDARD

Information INFORMATION

- Tietokantaan kohdistuvat oletukset:

Username testitapaus 07 löytyy järjestelmästä.

- Api palautti:

false

Virhe Unable to complete registration request (#33)

- Annetut syötteet:

UserID 1

Name RegReq

EMail postmaster@localhost

Status Professional

UserName testitapaus 08

Recommender Suosittelija

RecommenderEMail suosittelija@mail
DefaultDetector DETECTOR
DefaultLocation LOCATION
DefaultPhotometricSystem PHOTOMETRICSYSTEM
DefaultTimeStandard TIMESTANDARD
Information INFORMATION
Level 2
Comment kommentti on
Version 103

- Api palautti:

true

Viimeisessä testitapauksessa vielä tarkistetaan, ettei metodi kaadu saadessaan null-viitteen.

- Annetut syötteen:

UserFormContainer null
- Api palautti:

false

Virhe Null parameters in method call (#28)

Lopuksi tietokannasta tarkistettiin, että kantaan ilmeistyi kahdeksaan rekistörintipyyntöä. Näissä pyynnöissä käyttäjänimet olivat muotoa testitapaus [01-08].

4.7 login

Metodin toiminnallisuutta testataan kolmenlaisilla syötteillä; kelvollisilla, epäkelvoilla ja väärin tyypillisillä. Kelvollisessa syötteessä sekä tunnus että salasana ovat oikein, epäkelvoissa jompi kumpi tai molemmat ovat väärin ja väärin tyypisessä syötteessä arvona on esim. null-viite. Oikeellisuus tarkoittaa tiedon löytymistä kannasta ja sen vastaavuutta toisen tiedon kanssa.

- Annetut syötteen:

String "Tahiti"

String "password"
- Api palautti:

User -olion
- Annetut syötteen:

String "not_valid"

String "passport"

- Api palautti:

null -viite. Kokeiltiin myös puolittain virheellisillä arvoilla. Tulos silloinkin null-viite.

- Vääränmuotoiset arvot eivät menneet testiluokan kääntäjästä lävitse.

4.8 insertLightcurve

Metodin käyttö edellyttää syöttäjätason oikeuksia, joten sen käyttöä testataan ensin kaikilla kolmella eri käyttäjätasolla, kun syötteenä on oikeellinen yhden datapisteen ja aika ja visual sarakkeet sisältävä LightCurveForm-container. Tämän jälkeen testitapauksia varioidaan varioimalla LightcurveFormContainerin sisältöä seuraavasti. Jatkossa kenttiä AsteroidName, AsteroidNumber ja AsteroidDesignation kutsutaan *nimitiedoiksi*.

- Kaikki nimitiedot tyhjiä merkkijonoja.
- Nimitiedoista on annettu vain numero
- Nimitiedoista on annettu vain nimi
- Nimitiedoista on annettu vain tunnus
- Nimitiedot on asetettu siten, että numero vastaa yhtä asteroidia ja nimi jotakin toista
- ZeroTime kenttä on nolla tai pienempi kuin nolla (koska suunnitteludokumentissa oli näköjään virhe)
- Jokin pakollisista kentistä on null-viite tai tyhjä merkkijono
- Suunnitteludokumentin luvusta 3.5 lyötyvät ylä- ja alarajat sekä ne ylittävät että alittavat arvot

Kun ollaan varmistuttu, että metodi toimii muiden kuin data- ja columns- kenttien osalta, pidetään seuraavissa testeissä muut kentät vakioina ja varioidaan data ja columns-kenttiä. Testataan seuraavat tapaukset.

- Columns-taulukossa eri määrä sarakkeita kuin datakentässä
- Columns-taulukossa kaksi samaa saraketta
- Columns-taulukossa ei aikasaraketta
- Columns-taulukossa vakio, joka ei ole TahtitiLibraryssä
- Tyhjä datakenttä
- Tyhjiä rivejä oikeiden rivien välissä

- Yksi epäkelpo rivi
- Kaikki datapisteet ovat täsmälleen samanlaisia
- Oikeanlainen data usealla datapisteellä

Lopuksi testataan vielä täsmälleen samojen tietojen syöttämistä kaksi kertaa peräkkäin.

Testataan tapaus, jossa käyttäjätaso on selaaja

- Annetut syötteet

LightcurveID

AsteroidNumber 234

AsteroidDesignation

AsteroidName Ida

ObservingSite Juhon testilaboratorio

Observer J.M.

ZeroTime 2433282.5

UnitOfTime 0

AbsolutePhotometry false

ZeroMagnitude 0

LightTimeCorrected false

Columns "0""2"

Data 0.5 18.54

PhotometricSystem Boonussydeemi

TimeStandard Juhon pää

Reference Ei oo julkastu

Information Tää on ihan vaan testitapaus

Detector Korvien välissä

Version

- Metodi palautti

Virhe 0 Unable to get userdata from session, 1

Testataan tapaus, jossa käyttäjätaso on syöttäjä Login: Tahiti2

- Annetut syötteet

LightcurveID

AsteroidNumber 234

AsteroidDesignation

AsteroidName Ida
ObservingSite Juhon testilaboratorio
Observer J.M.
ZeroTime 2433282.5
UnitOfTime 0
AbsolutePhotometry false
ZeroMagnitude 0
LightTimeCorrected false
Columns "0""2"
Data 0.5 18.54
PhotometricSystem Boonussydeemi
TimeStandard Juhon pää
Reference Ei oo julkastu
Information Tää on ihan vaan testitapaus
Detector Korvien välissä
Version

- Metodi palautti

boolean true

Testataan tapaus, jossa nimitiedot ovat tyhjiä Login: Tahiti2

- Annetut syötteet

LightcurveID
AsteroidNumber
AsteroidDesignation
AsteroidName
ObservingSite Juhon testilaboratorio
Observer J.M.
ZeroTime 2433282.5
UnitOfTime 0
AbsolutePhotometry false
ZeroMagnitude 0
LightTimeCorrected false
Columns "0""2"
Data 0.5 18.54
PhotometricSystem Boonussydeemi

TimeStandard Juhon pää

Reference Ei oo julkastu

Information Tää on ihan vaan testitapaus

Detector Korvien välissä

Version

- Metodi palautti

Virhe 0 Could not identify asteroid unambiguously, or Asteroid not found, 15

Testataan tapaus, jossa nimitiedoista on annettuna vain numero Login: Tahiti2

- Annetut syötteet

LightcurveID

AsteroidNumber 234

AsteroidDesignation

AsteroidName

ObservingSite Juhon testilaboratorio

Observer J.M.

ZeroTime 2433282.5

UnitOfTime 0

AbsolutePhotometry false

ZeroMagnitude 0

LightTimeCorrected false

Columns "0""2"

Data 0.5 18.54

PhotometricSystem Boonussydeemi

TimeStandard Juhon pää

Reference Ei oo julkastu

Information Tää on ihan vaan testitapaus

Detector Korvien välissä

Version

- Metodi palautti

boolean true

Testataan tapaus, jossa nimitiedoista on annettuna vain nimi Login: Tahiti2

- Annetut syötteet

LightcurveID
AsteroidNumber
AsteroidDesignation
AsteroidName Ida
ObservingSite Juhon testilaboratorio
Observer J.M.
ZeroTime 2433282.5
UnitOfTime 0
AbsolutePhotometry false
ZeroMagnitude 0
LightTimeCorrected false
Columns "0"2"
Data 0.5 18.54
PhotometricSystem Boonussydeemi
TimeStandard Juhon pää
Reference Ei oo julkastu
Information Tää on ihan vaan testitapaus
Detector Korvien välissä
Version

- Metodi palautti

boolean true

Testataan tapaus, jossa nimitiedoista on annettuna vain tunnus Login: Tahiti2

- Annetut syötteet

LightcurveID
AsteroidNumber
AsteroidDesignation 1998 SF36
AsteroidName
ObservingSite Juhon testilaboratorio
Observer J.M.
ZeroTime 2433282.5
UnitOfTime 0
AbsolutePhotometry false
ZeroMagnitude 0
LightTimeCorrected false

Columns "0"2"
Data 0.5 18.54
PhotometricSystem Boonussydeemi
TimeStandard Juhon pää
Reference Ei oo julkastu
Information Tää on ihan vaan testitapaus
Detector Korvien välissä
Version

- Metodi palautti

boolean true

Testataan tapaus, jossa nimitiedoista on annettuna eri asteroidien nimi ja tunnus Login: Tahiti2

- Annetut syötteet

LightcurveID
AsteroidNumber
AsteroidDesignation 1998 SF36
AsteroidName Ida
ObservingSite Juhon testilaboratorio
Observer J.M.
ZeroTime 2433282.5
UnitOfTime 0
AbsolutePhotometry false
ZeroMagnitude 0
LightTimeCorrected false
Columns "0"2"
Data 0.5 18.54
PhotometricSystem Boonussydeemi
TimeStandard Juhon pää
Reference Ei oo julkastu
Information Tää on ihan vaan testitapaus
Detector Korvien välissä
Version

- Metodi palautti

Virhe 0 Could not identify asteroid unambiguously, or Asteroid not found, 15

Testataan tapaus, jossa zero-time on pienempi kuin nolla Login: Tahiti2

- Annetut syötteet

LightcurveID

AsteroidNumber 234

AsteroidDesignation

AsteroidName Ida

ObservingSite Juhon testilaboratorio

Observer J.M.

ZeroTime -1

UnitOfTime 0

AbsolutePhotometry false

ZeroMagnitude 0

LightTimeCorrected false

Columns {"0""2"}

Data 2433300 18.54

PhotometricSystem Boonussydeemi

TimeStandard Juhon pää

Reference Ei oo julkastu

Information Tää on ihan vaan testitapaus

Detector Korvien välissä

Version

- Metodi palautti

boolean true

Testataan tapaus, jossa zero-time on nolla Login: Tahiti2

- Annetut syötteet

LightcurveID

AsteroidNumber 234

AsteroidDesignation

AsteroidName Ida

ObservingSite Juhon testilaboratorio

Observer J.M.

ZeroTime 0

UnitOfTime 0

AbsolutePhotometry false
ZeroMagnitude 0
LightTimeCorrected false
Columns {"0""2"}
Data 2433300 18.54
PhotometricSystem Boonussydeemi
TimeStandard Juhon pää
Reference Ei oo julkastu
Information Tää on ihan vaan testitapaus
Detector Korvien välissä
Version

- Metodi palautti

boolean true

Testataan tapaus, jossa pakollinen kenttä on null Login: Tahiti2

- Annetut syötteet

LightcurveID
AsteroidNumber 234
AsteroidDesignation
AsteroidName Ida
ObservingSite Juhon testilaboratorio
Observer null
ZeroTime 2433282.5
UnitOfTime 0
AbsolutePhotometry false
ZeroMagnitude 0
LightTimeCorrected false
Columns {"0""2"}
Data 0.5 18.54
PhotometricSystem Boonussydeemi
TimeStandard Juhon pää
Reference Ei oo julkastu
Information Tää on ihan vaan testitapaus
Detector Korvien välissä
Version

- Metodi palautti

Virhe 0 Operation aborted, mandantory data missing, 11

Testataan tapaus, jossa columns-sarakkeita eri määrä kuin datasarakkeita (dataa enemmän)

- Annetut syötteen

LightcurveID

AsteroidNumber 234

AsteroidDesignation

AsteroidName Ida

ObservingSite Juhon testilaboratorio

Observer J.M.

ZeroTime 2433282.5

UnitOfTime 0

AbsolutePhotometry false

ZeroMagnitude 0

LightTimeCorrected false

Columns {"0""2"}

Data 0.5 18.54 0.94

PhotometricSystem Boonussydeemi

TimeStandard Juhon pää

Reference Ei oo julkaistu

Information Tää on ihan vaan testitapaus

Detector Korvien välissä

Version

- Metodi palautti

Virhe 0 Incorrect parameter dataformat or value, 13

Testataan tapaus, jossa kaksi samaa columns-saraketta

- Annetut syötteen

LightcurveID

AsteroidNumber 234

AsteroidDesignation

AsteroidName Ida

ObservingSite Juhon testilaboratorio

Observer J.M.
ZeroTime 2433282.5
UnitOfTime 0
AbsolutePhotometry false
ZeroMagnitude 0
LightTimeCorrected false
Columns {"0""2""2"}
Data 0.5 18.54 20.00
PhotometricSystem Boonussydeemi
TimeStandard Juhon pää
Reference Ei oo julkastu
Information Tää on ihan vaan testitapaus
Detector Korvien välissä
Version

- Metodi palautti

boolean true

Testataan tapaus, jossa columnssissa ei aikasaraketta

- Annetut syötteen

LightcurveID
AsteroidNumber 234
AsteroidDesignation
AsteroidName Ida
ObservingSite Juhon testilaboratorio
Observer J.M.
ZeroTime 2433282.5
UnitOfTime 0
AbsolutePhotometry false
ZeroMagnitude 0
LightTimeCorrected false
Columns {"2""4"}
Data 18.54 20.00
PhotometricSystem Boonussydeemi
TimeStandard Juhon pää
Reference Ei oo julkastu

Information Tää on ihan vaan testitapaus

Detector Korvien välissä

Version

- Metodi palautti

Virhe 0 No Epoch-column found from data, 20

Testataan tapaus, jossa columnssissa vakio, joka ei tahitilibraryssa

- Annetut syötteen

LightcurveID

AsteroidNumber 234

AsteroidDesignation

AsteroidName Ida

ObservingSite Juhon testilaboratorio

Observer J.M.

ZeroTime 2433282.5

UnitOfTime 0

AbsolutePhotometry false

ZeroMagnitude 0

LightTimeCorrected false

Columns {"0"-1"}

Data 0.5 18.54

PhotometricSystem Boonussydeemi

TimeStandard Juhon pää

Reference Ei oo julkastu

Information Tää on ihan vaan testitapaus

Detector Korvien välissä

Version

- Metodi palautti

Virhe 0 Incorrect parameter dataformat or value, 13

Testataan tapaus, jossa datakentän erottimena useampia välilyöntejä

- Annetut syötteen

LightcurveID

AsteroidNumber 234

AsteroidDesignation
AsteroidName Ida
ObservingSite Juhon testilaboratorio
Observer J.M.
ZeroTime 2433282.5
UnitOfTime 0
AbsolutePhotometry false
ZeroMagnitude 0
LightTimeCorrected false
Columns {"0""2"}
Data 0.5 18.54
PhotometricSystem Boonussydeemi
TimeStandard Juhon pää
Reference Ei oo julkastu
Information Tää on ihan vaan testitapaus
Detector Korvien välissä
Version

- Metodi palautti

boolean true

Testataan tapaus, jossa datakentän erottimena tabulaattori

- Annetut syötteet

LightcurveID
AsteroidNumber 234
AsteroidDesignation
AsteroidName Ida
ObservingSite Juhon testilaboratorio
Observer J.M.
ZeroTime 2433282.5
UnitOfTime 0
AbsolutePhotometry false
ZeroMagnitude 0
LightTimeCorrected false
Columns {"0""2"}
Data 0.5 18.54

PhotometricSystem Boonussydeemi
TimeStandard Juhon pää
Reference Ei oo julkastu
Information Tää on ihan vaan testitapaus
Detector Korvien välissä
Version

- Metodi palautti

boolean true

Testataan tapaus, jossa datakentän erottimena tabulaattori ja välilyönti

- Annetut syötteen

LightcurveID
AsteroidNumber 234
AsteroidDesignation
AsteroidName Ida
ObservingSite Juhon testilaboratorio
Observer J.M.
ZeroTime 2433282.5
UnitOfTime 0
AbsolutePhotometry false
ZeroMagnitude 0
LightTimeCorrected false
Columns {"0" "2"}
Data 0.5 18.54
PhotometricSystem Boonussydeemi
TimeStandard Juhon pää
Reference Ei oo julkastu
Information Tää on ihan vaan testitapaus
Detector Korvien välissä
Version

- Metodi palautti

boolean true

Testataan tapaus, jossa datakentässä useita rivejä

- Annetut syötteet

LightcurveID

AsteroidNumber 234

AsteroidDesignation

AsteroidName Ida

ObservingSite Juhon testilaboratorio

Observer J.M.

ZeroTime 2433282.5

UnitOfTime 0

AbsolutePhotometry false

ZeroMagnitude 0

LightTimeCorrected false

Columns {"0""2"}

Data 0.5 18.54

0.6 18.55

0.7 17.23

0.8 20.20

PhotometricSystem Boonussydeemi

TimeStandard Juhon pää

Reference Ei oo julkastu

Information Tää on ihan vaan testitapaus

Detector Korvien välissä

Version

- Metodi palautti

boolean true

Testataan tapaus, jossa datakentässä tyhjä rivi usean datapisteen välissä

- Annetut syötteet

LightcurveID

AsteroidNumber 234

AsteroidDesignation

AsteroidName Ida

ObservingSite Juhon testilaboratorio

Observer J.M.

ZeroTime 2433282.5
UnitOfTime 0
AbsolutePhotometry false
ZeroMagnitude 0
LightTimeCorrected false
Columns {"0""2"}
Data 0.5 18.54 0.6 18.55
 0.7 17.23 0.8 20.20
PhotometricSystem Boonussydeemi
TimeStandard Juhon pää
Reference Ei oo julkastu
Information Tää on ihan vaan testitapaus
Detector Korvien välissä
Version

- Metodi palautti

boolean true

4.9 insertLightcurveAtlas

Metodi tulee testattua järjestelmätestausvaiheessa käytännön kannalta varsin kattavasti, joten integraatiotestaukseen sisällytetään vain muutama erikoistapaus. Nämä erikoistapaukset ovat seuraavat:

- Syötteenä annetaan LightcurveFormContainerin sijasta null.
- LightcurveFormContainer sisältää null-viitteitä.

Testi suoritettiin seuraavasti:

- Syötteet:
 - data** null
 - earth[]** {1.2,2.2,3.3,4.4,5.5,6.6,7.7}
 - asteroid[]** {1.2,2.2,3.3,4.4,5.5,6.6,7.7}
 - aspectTime** 3.3
 - atlasAspect** {1.2,2.2,3.3,4.4,5.5,6.6,7.7}
- Palautti:
 - false**

- Syötteet:

data AtlasLightcurveContainer, jossa kaikki arvot null.

earth[] {1.2,2.2,3.3,4.4,5.5,6.6,7.7}

asteroid[] {1.2,2.2,3.3,4.4,5.5,6.6,7.7}

aspectTime 3.3

atlasAspect {1.2,2.2,3.3,4.4,5.5,6.6,7.7}

- Palautti:

false

4.10 changePassword

Metodi vaatii syöttäjätason oikeudet, joten aluksi metodia testataan kaikilla käyttäjä- tasoilla oikeellisilla parametreilla eli oikein olevalla vanhalla salasanalla sekä 5-merkkisellä uudella salasanalla. Tämän jälkeen testataan tapaukset:

- Null-arvot parametreina
- Vanha salasana on väärin

Testataan tapaus, jossa käyttäjätaso on selaaja

- Annetut syötteet

String Tahiti

String password

- Metodi palautti

Virhe 0 Unable to get userdata from session, 1

Testataan tapaus, jossa käyttäjätaso on syöttäjä

- Annetut syötteet

String password

String passwd

- Metodi palautti

boolean true

Testataan tapaus, jossa käyttäjätaso on admin

- Annetut syötteet

String password

String passwd

- Metodi palautti

boolean true

Testataan tapaus, jossa salasanat nulleja

- Annetut syötteet

String null

String null

- Metodi palautti

Virhe 0 Null parameters in method call, 28

Testataan tapaus, jossa vanha salasana väärin:

- Annetut syötteet

String vaarin!

String password

- Metodi palautti

Virhe 0 Invalid username or password, 34

4.11 getUsers

Metodin syöte sisältää vain etsittävän merkkijonon, joten testaus on helppo rajata. Testattavat tapaukset:

- Null-arvo parametrinä
- Merkkijono, jonka avulla löydetään ainakin yksi käyttäjä
- Merkkijono, jonka avulla ei löydetä käyttäjiä
- Eri tasoilla käyttäjillä hakeminen

Metodin tulisi palauttaa null ensimmäisessä tapauksessa ja tyhjä Users-taulukko viimeisessä. Toisessa tapauksessa tuloksena olisi vähintään yhden alkion kokoinen Users-taulukko.

- Annetut syötteet:

User usernamella null

- Api palautti:

null -viite, kuten pitikin. HUOM: käyttäjänä nollakäyttäjä!

- Annetut syötteet:
`User` usernamella "persebus"
- Api palautti:
`null` –viite, kuten pitikin. HUOM: käyttäjänä nollakäyttäjä!
- Annetut syötteet:
`User` usernamella "Tahiti"
- Api palautti:
`null` –viite, kuten pitikin. HUOM: käyttäjänä nollakäyttäjä!
- Annetut syötteet:
`User` usernamella null
- Api palautti:
`User` –taulukko ilman alkioita. HUOM: käyttäjänä syöttäjä!
- Annetut syötteet:
`User` usernamella "persebus"
- Api palautti:
`User` –taulukko ilman alkioita. HUOM: käyttäjänä syöttäjä!
- Annetut syötteet:
`User` usernamella "Tahiti"
- Api palautti:
`User` –taulukko yhdellä alkiolla käyttäjän Tahiti tiedoilla. HUOM: käyttäjänä syöttäjä!
- Ylläpitäjä–tason käyttäjälle pätevät samat tulokset kuin syöttäjälle, lukuunottamatta kenttiä, jotka näkyvät vain ylläpitäjälle. Syöttäjä ei näitä tietoja saa.

4.12 `setUserData`

Metodille annetaan parametrina `UserFormContainer`–olio, jonka kenttien kelvollisuudesta riippuu metodin paluuarvo. Testitapauksia ovat siis epäkelvoilla arvoilla täytetty `UserFormContainer`, `null`–viite ja kelvollinen `UserFormContainer`.

Lisää testattavaa aiheuttaa käyttäjäoikeusvaatimus. Metodista ei luonnollisestikaan voi käyttää se-laajan oikeuksin, joten paluuarvon tulee olla tällöin `false`. Lisäksi pakolliset kentät, joita on suurin osa syöttestä, tulee olla täytettynä, tai muutos ei onnistu. Syötteen kentistä lisäksi `UserID:n` on oltava positiivinen.

Paluuarvo epäonnistuneissa muutoksissa on `false`. Onnistuneissa muutoksissa palautetaan `true`.

- Annetut syötteet:
UserFormContainer null
- Api palautti:
boolean arvoltaa false kaikilla käyttäjäoikeustasoilla
- Annetut syötteet:
UserFormContainer new UserFormContainer
- Api palautti:
boolean arvoltaa false kaikilla käyttäjäoikeustasoilla
- Annetut syötteet:
UserFormContainer käyttäjän kernel tiedoilla (väliaikainen käyttäjä)
- Api palautti:
boolean arvoltaa false nollakäyttäjällä ja muulla syöttäjällä (kernel oli syöttäjä). Arvo oli true jos käyttäjänä oli kernel tai ylläpitäjä

4.13 resetPassword

Testataan tapauksilla: loginID on null, loginID:ssä annettu käyttäjä on olemassa ja loginID:ssä annettua käyttäjää ei ole olemassa.

- Annetut syötteet:
logID null
- Palautti:
false
- Annetut syötteet:
logID "kukkelikumuaeioo" (käyttäjää ei ole tietokannassa)
- Palautti:
false
- Annetut syötteet:
logID "Luseri" (käyttäjää on tietokannassa)
- Palautti:
true
- Lähetti sähköpostin tietokannassa määriteltyyn osoitteeseen

4.14 changeLightcurve

Valokäyrien muuntaminen on mahdollista vain ylläpitäjä-tason käyttäjille. Tämän takia metodin ensimmäisesti testitapaukset yksinkertaisesti koostuvat käyttöyrityksistä ilman tarvittavia oikeuksia. Metodin toimintaa pitää kokeilla ennen kirjautumista, sekä syöttäjän oikeuksilla. Kummassakaan tapauksessa suorittamisen ei pitäisi onnistua. Testiajon alussa kannassa tulee olla valokäyrä 2, versio 0 sekä Atlas-syöttimellä siirretty valokäyrä 3.

- Annetut syötteet:

User Käyttäjä ei ole kirjautunut sisään.

- Api palautti:

false

Virhe Unable to get userdata from session (#1)

- Annetut syötteet:

UserLevel 1 (TahitiLibrary.USER_SUBMITTER)

- Api palautti:

false

Virhe Action not allowed (#2)

Annettava syöte, LightcurveFormContainer sisältää lukuisia kenttiä, jotka aiheuttavat runsaasti erilaisia testaustapauksia. Osa annettavista kentistä sisältää pakollista tietoa, joten metodin toimintaa tulee ainakin testata tilanteessa, jossa jonkin näistä pakollisista kentistä on tyhjä, tai sisältää pelkän null-viittauksen.

- Annetut syötteet:

Observer <tyhjä String>

ObservingSite Site

ZeroTime 2483282.5

UnitOfTime 1

AbsolutePhotometry true

ZeroMagnitude 0

LightTimeCorrected false

- Api palautti:

false

Virhe Operation aborted, mandantory data missing (#11)

- Annetut syötteet:

Observer Observer
ObservingSite null
ZeroTime 2483282.5
UnitOfTime 1
AbsolutePhotometry true
ZeroMagnitude 0
LightTimeCorrected false

- Api palautti:

false

Virhe Operation aborted, mandantory data missing (#11)

Valinnaiset kentät (Information, Reference, PhotometricSystem, Detector, TimeStandard) antavat myös erilaisia testaustapauksia. Testauksessa pitää ainakin olla tapaukset, jossa näitä tietoja ei alunperin tietokannassa ole, ja niille annetaan uudet arvot. Lisäksi tarvitaan tilanne, jossa tietokannassa olevat kentät sisältävät informaatiota, mutta uudet kentät eivät. Kolmas muutostyyppi saadaan, kun tietokannassa oleva informaation muuttuu. Nämä kaikki ehdot saadaan yhteen testitapaukseen.

- Annetut syötteet:

LightcurveID 2
AsteroidNumber 234
AsteroidDesignation
AsteroidName
ObservingSite Site
Observer isosilmä
ZeroTime 2483282.5
UnitOfTime 1
AbsolutePhotometry true
ZeroMagnitude 0
LightTimeCorrected false
Columns {"0""2"}
Data "24.43 2352\n43.43 4532\n44.54 543"
PhotometricSystem CCD
TimeStandard tik-tak, tik-tak
Reference referenssiin en keksi mitään
Information huuhaa

Detector Havaintoasema 2001

Version 0

- Api palautti:

true

Ekassa muutoksessa kaikki vapaaehtoisetkin kentät täytettiin tavaralla, joten nyt tiedään, että sitä on. Nyt muutetaan joitain kenttiä, ja tyhjennetään joitain.

- Annetut syötteen:

LightcurveID 2

AsteroidNumber 234

AsteroidDesignation

AsteroidName

ObservingSite ObservingSite

Observer Observer

ZeroTime 2483282.5

UnitOfTime 1

AbsolutePhotometry true

ZeroMagnitude 0

LightTimeCorrected false

Columns {"0""2"}

Data "24.43 2352\n43.43 4532\n44.54 543"

PhotometricSystem null

TimeStandard uusi, uljas timestandard

Reference

Information

Detector Kuun pimeämäpi puoli

Version 1

- Api palautti:

true

Koska muutos voi koskea vain yhtä valokäyrään, täytyy metodin toimintaa myös testata sellaisessa tapauksessa, jossa syötteenä annettava data sisältää enemmän, kuin yhden valokäyrän havaintotietoja. Tällöin metodin ei pitäisi tehdä minkäänlaisia muutoksia tietokantaan. Lisäksi datapisteisiin liittyen pitää tehdä testejä, joissa käytettävä filteri vaihtuu, mittausarvot vaihtuvat sekä mittausajat vaihtuvat. Näiden lisäksi tarvitaan myös testi tilanteesta, jossa mahdollinen error-kentässä jo oleva arvo katoaa, sekä tilanteesta, jossa errortieto talletetaan tietokantaan.

- Annetut syötteet:

Columns {"0", "2", "3"}

Data "23 234 5643\n43 5435 .4234\n54 5.53 .235"

- Api palautti:

false

Virhe Lightcurve changing can only contain one lightcurve (#18)

Testaan errorin lisäämistä, vaihdetaan filteri samalla

- Annetut syötteet:

Columns {"0", "1", "3"}

Data "23 234 5643\n43 5435 .4234\n54 5.53 .235"

- Api palautti:

true

- Annetut syötteet:

Columns {"0", "2"}

Data "24.43 2352\n43.43 4532\n44.54 543"

- Api palautti:

true

Koska `changeLightcurve`-metodilla voidaan myös vaihtaa asteroidin kohdetta, täytyy metodin toiminta testata tilanteissa, joissa uutta kohdetta ei löydy järjestelmästä lainkaan. Toinen vastaanvotettava testitapaus on havainnon muuttaminen siten, että sen kohteena on maapallo, eli asteroid 0.

- Annetut syötteet:

AsteroidNumber

AsteroidDesignation 1998 SF36

- Api palautti:

true

- Annetut syötteet:

AsteroidName Earth

AsteroidDesignation null

- Api palautti:

false

Virhe Could not identify asteroid unambiguously, or Asteroid not found (#15)

- Annetut syötteet:

AsteroidNumber 234

AsteroidName null

AsteroidDesignation 1998 SF36

- Api palautti:

false

Virhe Could not identify asteroid unambiguously, or Asteroid not found (#15)

Atlas-siirtimellä järjestelmään lisättyjä valokäyriä ei pitäisi pystyä muuttamaan, joten metodia pitää myös kokeilla tällaisen valokäyrän muuttamiseen. Lisäksi muuttamista pitää yrittää sellaiselle valokäyrälle, jota järjestelmästä ei löydy lainkaan, sekä väärälle versionumerolle, jolloin muutoksen ei myöskään pitäisi onnistua. Valokäyrä #3 on siirretty järjestelmään Atlas-syöttimellä.

- Annetut syötteet:

LightcurveID 3

- Api palautti:

false

Virhe Lightcurve Atlas-imported, cannot be changed (#52)

- Annetut syötteet:

LightcurveID 32453223

- Api palautti:

false

Virhe LightcurveID not found, or error searching it (#17)

- Annetut syötteet:

Version 4342

- Api palautti:

false

Virhe LightcurveID not found, or error searching it (#17)

Koska tiettyjen kenttien (`UnitOfTime`, `zeroTime`, `zeroMagnitude`, `columns`-taulukon sisältö) annettavassa datassa tulee sisältää vain tietynlaisia arvoja, pitää metodin toimintaa testata oikeiden arvojen ulkopuolella olevilla arvoilla sekä rajoilla olevilla arvoilla. Seuraavassa on lueteltu tällaiset arvoraajat:

- `UnitOfTime`
 Kentän `unitOfTime` tulee olla joko 0 (`UNIT_DAY`) tai 1 (`UNIT_HOUR`). Toimintaa kokeillaan näiden arvojen lisäksi arvolla 2.
- `Magnitude`
`Magnitude` ei voi olla negatiivinen. Metodin toimintaa testataan tilanteessa, jossa jonkin havaintopisteen magnituden arvoksi tulee negatiivinen arvo.
- `columns`-taulukko
`Columns`-taulukon sisältäminen arvojen tulee olla `TahitiLibrary`stä löytyviä vakioita, eli valideja arvoja ovat luvut väliltä 0...7. Lisäksi jokaisesta syötteenä annettavasta `columns`-taulukosta tulee löytyä aikasarake (0), sekä vähintään yksi datasarake. `Error`-saraketta (1) ei lasketa varsinaiseksi datasarakkeeksi. Tämän perusteella metodin toimintaa pitää ainakin kokeilla tilanteissa, jossa saraketiedot sisältävät sarakkeet -1 ja 8, sekä tilanteessa, jossa varsinainen havaintotieto puuttuu. `Columns`-taulukkoon yritetään myös laittaa ei-numeerista tietoa, jolloin suorituksen pitäisi keskeytyä.
- Annetut syötteet:
UnitOfTime 0
- Api palautti:
true
- Annetut syötteet:
UnitOfTime 2
- Api palautti:
false
Virhe Incorrect parameter dataformat or value (#13)
- Annetut syötteet:
Data "24.43 -2\n43.43 4532\n44.54 543"
- Api palautti:
false
Virhe Incorrect parameter dataformat or value (#13)

- Annetut syötteet:

LightcurveID 2
AsteroidNumber 234
AsteroidDesignation null
AsteroidName null
ObservingSite ObservingSite
Observer Observer
ZeroTime 2483282.5
UnitOfTime 1
AbsolutePhotometry true
ZeroMagnitude 0
LightTimeCorrected false
Columns {"0"-1"}
Data 4.2 234\n4.6 434\n4.75 34.23
PhotometricSystem null
TimeStandard uusi, uljas timestandard
Reference
Information
Detector Kuun pimeämäpi puoli
Version 7

- Api palautti:

false
Virhe Incorrect parameter dataformat or value (#13)

- Annetut syötteet:

Columns {"0", "8"}

- Api palautti:

false
Virhe Incorrect parameter dataformat or value (#13)

- Annetut syötteet:

Columns {"0", "1"}

- Api palautti:

false

Virhe No measurement resultdata found (#21)

- Annetut syötteet:

Columns {"2", "1"}

- Api palautti:

false

Virhe No Epoch-column found from data (#20)

- Annetut syötteet:

Columns {"0", "Roska"}

- Api palautti:

false

Virhe Error parsing numerical data (#12)

Tehdään vielä testi, jossa dataa olis 3 columnia, mutta columni-infoa kahdesta

- Annetut syötteet:

LightcurveID 2

AsteroidNumber 234

AsteroidDesignation null

AsteroidName null

ObservingSite ObservingSite

Observer Observer

ZeroTime 2483282.5

UnitOfTime 1

AbsolutePhotometry true

ZeroMagnitude 0

LightTimeCorrected false

Columns {"0""2"}

Data 234 34 432\n23 343 532

PhotometricSystem null

TimeStandard uusi, uljas timestandard

Reference

Information

Detector Kuun pimeämäpi puoli

Version 7

- Api palautti:

false**Virhe** Incorrect parameter dataformat or value (#13)

Viimeisessä testitapaus luokassa muutetaan kannassa olevan valokäyrän havaintopisteiden määrää. Aluksi määrä kasvatetaan neljään, ja tämän jälkeen lasketaan yhteen. Lopuksi vielä testataan, ettei suoritus kaadu virheeseen, mikäli metodille annetaan null-pointter.

- Annetut syötteet:

LightcurveID 2**AsteroidNumber** 234**AsteroidDesignation** null**AsteroidName** null**ObservingSite** ObservingSite**Observer** Observer**ZeroTime** 2483282.5**UnitOfTime** 1**AbsolutePhotometry** true**ZeroMagnitude** 0**LightTimeCorrected** false**Columns** {"0""2"}**Data** "234 34\n23 343\n4 243\n443 234"**PhotometricSystem** null**TimeStandard** uusi, uljas timestandard**Reference****Information****Detector** Kuun pimeämäpi puoli**Version** 7

- Api palautti:

true

- Annetut syötteet:

LightcurveID 2**AsteroidNumber** 234**AsteroidDesignation** null

AsteroidName null
ObservingSite ObservingSite
Observer Observer
ZeroTime 2483282.5
UnitOfTime 1
AbsolutePhotometry true
ZeroMagnitude 0
LightTimeCorrected false
Columns {"0""2"}
Data 234 34
PhotometricSystem null
TimeStandard uusi, uljas timestandard
Reference
Information
Detector Kuun pimeämäpi puoli
Version 8

- Api palautti:

true

- Annetut syötteet:

LightcurveFormContainer null

- Api palautti:

false

Virhe Operation aborted, mandantory data missing (#11)

4.15 logoff

Koska logoff-metodi ei tarvitse minkäänlaista syötettä, on sen testaaminen hyvin yksinkertaista. Metodin toimintaa tulee testata tilanteessa, joissa on ensiksi kirjaututtu sisään, sekä tilanteessa, jossa käyttäjä ei vielä ole kirjautunut sisään. Jotta metodin toiminnasta voidaan varmistua, tulee ennen ja jälkeen metodin suorittamista yrittää suorittaa jokin erillisiä käyttöoikeuksia tarvitseva metodi.

Testaus suoritettiin siten, että aluksi kirjaututtiin ylläpito-tason käyttäjällä sisään, ja tehtiin muutamia rekistöröintiopyyntöjä. Tämän jälkeen kutsuttiin logoffia, ja yritettiin tehdä uusi rekistöröintiopyyntö. Tämä ei onnistunut, joten logoff-metodi toimii odotetulla tavalla. Toisen logoff-kutsun tekeminen peräkkäin ei vaikuttanut asiaan millään tavalla.

4.16 removeUser

Metodi tarvitsee syötteenä saamastaan User-oliosta vain käyttäjän järjestelmän sisäisen tunnistenumeron. Arvon oikeellisuudesta riippuu poiston onnistuminen. Tunnisteen on oltava positiivinen, minkä lisäksi kyseiseen numeroon sidotun käyttäjän on oltava olemassa järjestelmän tietokannassa. Myös tietokantayhteyden on oltava kunnossa, jotta operaatio voisi onnistua.

Testiaineiston on koostuttava ainakin null-viitteestä, epäkelvosta User-oliosta (epäkelpo tunniste) sekä kelvollisesta User-oliosta. Myös tietokantayhteyden tilan vaikutus tulokseen testataan.

Metodi palauttaa onnistuessaan arvon true. Epäonnistuessaan metodi palauttaa arvon false.

- Annetut syötteen:

User null

- Api palautti:

boolean arvoltaan false kaikilla käyttäjätasoilla

- Annetut syötteen:

User new User()

- Api palautti:

boolean arvoltaan false kaikilla käyttäjätasoilla

- Annetut syötteen:

User userID:llä 39 (silloinen väliaikaiskäyttäjä)

- Api palautti:

boolean arvoltaan false syöttäjän ja nollakäyttäjän tapauksessa, ylläpitäjän tapauksessa arvoltaan true

4.17 getSettings

Metodin toiminta edellyttää ylläpitäjätason oikeuksia, joten sen kutsumista testataan kaikilla käyttäjätasoilla. Testataan, että metodi palauttaa kantaan talletetut asetukset.

Testataan tapaus, jossa käyttäjätaso on selaaja

- Metodi palautti

Virhe 0 Unable to get userdata from session, 1

Testataan tapaus, jossa käyttäjätaso on syöttäjä

- Metodi palautti

Virhe 0 Action not allowed, 2

Testataan tapaus, jossa käyttäjätaso on ylläpitäjä

- Metodi palautti

ApiSettings

serverPort 25

adminEMail admin@mail.astro

mailServer localhost

approveMessage You have been approved

passwordMessage This is your new password

referenceMessage Have you recommended this user] *username*,recommender ?

approveMessageHeader SAPC Approve

passwordMessageHeader SAPC Password

referenceMessageHeader SAPC Recommendation request

4.18 changeSettings

Metodia testataan ensiksi selaaja-, syöttäjä- ja ylläpitäjätason oikeuksilla siten, että syötteenä annettavan SettingsFormContainerin jokaiseen kenttään laitetaan validi arvo (so. kenttä ei ole null ja ServerPort>=0). Tämän jälkeen ylläpitäjätason oikeuksilla testataan seuraavat tapaukset:

- Syötteenä annettu newSettings on null
- Joku syötteenä annetun SettingsFormContainerin kentistä on null
- ServerPort<0

Metodi testattiin seuraavasti:

- Käyttöoikeustila: Browser
- Annetut syötteenä:
 - newSettings** AdminEmail "ukko@kukko.fi",MailServer "host", ServerPort "54", Approve-MessageHeader "aaha", ApproveMessage "joo", PasswordMessageHeader "jaa", Password-Message "naah", ReferenceMessageHeader "fokopk", ReferenceMessage "koplop"
- Api palautti:
 - false**
- Käyttöoikeustila: Submitter
- Annetut syötteenä:

newSettings AdminEmail "ukko@kukko.fi",MailServer "host", ServerPort "54", ApproveMessageHeader "aaha", ApproveMessage "joo", PasswordMessageHeader "jaa", PasswordMessage "naah", ReferenceMessageHeader "fokopk", ReferenceMessage "koplop"

- Api palautti:

false

- Käyttöoikeustila: Admin

- Annetut syötteet:

newSettings AdminEmail "ukko@kukko.fi",MailServer "host", ServerPort "54", ApproveMessageHeader "aaha", ApproveMessage "joo", PasswordMessageHeader "jaa", PasswordMessage "naah", ReferenceMessageHeader "fokopk", ReferenceMessage "koplop"

- Api palautti:

true

- Tietokantaan tallettuivat esitetyt arvot

- Käyttöoikeustila: Admin

- Annetut syötteet:

newSettings null

- Api palautti:

false

- Käyttöoikeustila: Admin

- Annetut syötteet:

newSettings AdminEmail null,MailServer null, ServerPort null, ApproveMessageHeader null, ApproveMessage null, PasswordMessageHeader null, PasswordMessage null, ReferenceMessageHeader null, ReferenceMessage null

- Api palautti:

false

- Käyttöoikeustila: Admin

- Annetut syötteet:

newSettings AdminEmail "ukko@kukko.fi",MailServer "host", ServerPort "-11", ApproveMessageHeader "aaha", ApproveMessage "joo", PasswordMessageHeader "jaa", PasswordMessage "naah", ReferenceMessageHeader "fokopk", ReferenceMessage "koplop"

- Api palautti:

false

4.19 getLogEntries

Koska suunnitteludokumentin mukaan getLogEntries-metodin käyttämiseen tarvitaan ylläpitäjän oikeudet, testataan metodin toimintaa tilanteissa, joissa kirjautuneella käyttäjällä on syöttäjän oikeudet, sekä tilanteessa, jossa käyttäjä ei ole kirjautunut järjestelmään sisään laisinkaan.

- Annetut syötteet:

User Käyttäjä ei ole kirjautunut sisään.

- Api palautti:

null-viitteen

Virhe Unable to get userdata from session (#1)

- Annetut syötteet:

UserLevel 1 (TahitiLibrary.USER.SUBMITTER)

- Api palautti:

null-viitteen

Virhe Action not allowed (#2)

Metodin syötteenään ottava logQueryContainer sisältää kaksi merkkijonokenttää, timeStart ja timeEnd. Metodin toimintaa tullaan testaamaan tilanteissa, joissa molemmat tai jompikumpi merkkijono on null tai tyhjä. Tällaisissa tilanteissa metodin tulisi toimia rajoittamatta, toisinsanoen, esimerkiksi timeEndin ollessa tyhjä, palautetaan kaikki kenttää timeStart myöhäisemmät havainnot. Luonnollisesti metodin toimintaa testataan myös molempien kenttien ollessa täytettyinä.

Tyhjässä haussa metodi palautti seuraavaa. Tästä siis selviää, mitä ajohetkellä tietokannassa oli, joten seuraavissa tapauksissa oletetaan, ettei tietokannan sisältö ole muuttunut.

- Annetut syötteet:

TimeStart <tyhjä String>

TimeEnd null

- Api palautti:

LogEntry-taulukon , jossa 3 alkia. Alkioiden sisältönä oli seuraavaa:

Elementti 0, entryID 2

Elementti 0, Maker Tahiti

Elementti 0, EntryDate Tue Apr 29 17:43:58 EEST 2003

Elementti 0, Entry Unknown class||No entry||Unknown fields||Unknown old values||Unknown present values||

Elementti 1, entryID 3

Elementti 1, Maker Tahiti

Elementti 1, EntryDate Wed Apr 30 13:24:03 EEST 2003

Elementti 1, Entry rour||mnerr||Unknown fields||Unknown old values||Unknown present values||

Elementti 2, entryID 1

Elementti 2, Maker Tahiti

Elementti 2, EntryDate Mon Apr 28 17:43:58 EEST 2003

Elementti 2, Entry KLASSE||very lame, _tamed_ logentry||mertol,bested||kurrikulum,vitae||life,story||

- Annetut syötteet:

TimeStart 29.04.2003

TimeEnd null

- Api palautti:

LogEntry-taulukon , jossa 2 alkia. Alkioiden sisältönä oli seuraavaa:

Elementti 0, entryID 2

Elementti 0, Maker Tahiti

Elementti 0, EntryDate Tue Apr 29 17:43:58 EEST 2003

Elementti 0, Entry Unknown class||No entry||Unknown fields||Unknown old values||Unknown present values||

Elementti 1, entryID 3

Elementti 1, Maker Tahiti

Elementti 1, EntryDate Wed Apr 30 13:24:03 EEST 2003

Elementti 1, Entry rour||mnerr||Unknown fields||Unknown old values||Unknown present values||

- Annetut syötteet:

TimeStart 29.04.2003

TimeEnd 29.04.2003

- Api palautti:

LogEntry-taulukon , jossa 1 alkia. Alkioiden sisältönä oli seuraavaa:

Elementti 0, entryID 2

Elementti 0, Maker Tahiti

Elementti 0, EntryDate Tue Apr 29 17:43:58 EEST 2003

Elementti 0, Entry Unknown class||No entry||Unknown fields||Unknown old values||Unknown present values||

- Annetut syötteet:

TimeStart 28.4.2003

TimeEnd 29.04.2003

- Api palautti:

LogEntry-*taulukon* , jossa 2 alkiota. Alkioiden sisältönä oli seuraavaa:

Elementti 0, entryID 1

Elementti 0, Maker Tahiti

Elementti 0, EntryDate Mon Apr 28 17:43:58 EEST 2003

Elementti 0, Entry KLASSE||very lame, _tamed_ logentry||mertol,bested||kurrikulum,vitae||life,story||

Elementti 1, entryID 2

Elementti 1, Maker Tahiti

Elementti 1, EntryDate Tue Apr 29 17:43:58 EEST 2003

Elementti 1, Entry Unknown class||No entry||Unknown fields||Unknown old values||Unknown present values||

- Annetut syötteen:

TimeStart <tyhjä String>

TimeEnd 29.04.2003

- Api palautti:

LogEntry-*taulukon* , jossa 2 alkiota. Alkioiden sisältönä oli seuraavaa:

Elementti 0, entryID 2

Elementti 0, Maker Tahiti

Elementti 0, EntryDate Tue Apr 29 17:43:58 EEST 2003

Elementti 0, Entry Unknown class||No entry||Unknown fields||Unknown old values||Unknown present values||

Elementti 1, entryID 1

Elementti 1, Maker Tahiti

Elementti 1, EntryDate Mon Apr 28 17:43:58 EEST 2003

Elementti 1, Entry KLASSE||very lame, _tamed_ logentry||mertol,bested||kurrikulum,vitae||life,story||

Lisäksi kenttiin kokeillaan laittaa satunnaista tekstiä, jolloin päivämäärien purkaminen syöteken-
tistä pitäisi olla mahdotonta, ja haun epäonnistua. Vaikkakaan metodin ei sinäänsä tarvitsekaan
varautua tilanteeseen, jossa timeEnd on aikaisempi, kuin timeStart, testataan metodin toimin-
taa myös tällaisessa tilanteessa. Oletettavasti tällaisessa tilanteessa metodi palauttaa logEntry-
taulukon, jossa ei ole yhtään alkiota.

- Annetut syötteen:

TimeStart 30.04.2003

TimeEnd 29.04.2003

- Api palautti:

LogEntry-taulukon , jossa 0 alkiota.

- Annetut syötteet:

TimeStart 76.34.2342

TimeEnd 29.04.2003

- Api palautti:

LogEntry-taulukon , jossa 0 alkiota.

- Annetut syötteet:

TimeStart 12.12.2000

TimeEnd roskaa

- Api palautti:

null-viitteen

Virhe Incorrect parameter dataformat or value (#13)

- Annetut syötteet:

LogQueryContainer null

- Api palautti:

null-viitteen

Virhe Missing logquery object (#48)

4.20 deleteLightcurve

Metodille annetaan syötteenä vain valokäyrän tunnistenumero, joten testiaineisto rajoittuu kelloiseen ja epäkelpoon lukuun. Onnistuessaan metodi palauttaa arvon true, muutoin false. Ensimmäisellä kerralla käyrä vain poistomerkitään, joten sen löytyminen kannasta voidaan tarkistaa tarkoitukseen sopivalla metodilla. Toisella kerralla käyrä poistetaan kokonaan järjestelmästä.

- Annetut syötteet:

int 200 (ei ko. valokäyrää)

- Api palautti:

boolean arvoltaan false kaikilla käyttäjillä

- Annetut syötteet:

int 5

- Api palautti:

boolean arvoltaan false syöttäjän ja nollakäyttäjän tapauksessa, ylläpitäjän tapauksessa arvoltaan true

4.21 restoreLightcurve

Metodin käyttö vaatii ylläpitäjätason oikeudet, joten sen kutsumista testataan ensin kaikilla käyttäjätasoilla oikeellisella valokäyrän tunnuksella valokäyrälle, joka on merkattu poistetuksi. Sen jälkeen testataan seuraavat tapaukset:

- Valokäyrätunnusta ei löydy kannasta
- Valokäyrätunnuksen valokäyrää ei ole merkattu poistetuksi

Testataan tapaus, jossa käyttäjätaso on selaaja

- Annetut syötteet

LightcurveID 1

- Tietokantaan kohdistuvat oletukset
Valokäyrä 1 on merkitty poistetuksi
- Metodi palautti

Virhe 0 Unable to get userdata from session, 1

Testataan tapaus, jossa käyttäjätaso on syöttäjä

- Annetut syötteet

LightcurveID 1

- Tietokantaan kohdistuvat oletukset
Valokäyrä 1 on merkitty poistetuksi
- Metodi palautti

Virhe 0 Action not allowed, 2

Testataan tapaus, jossa käyttäjätaso on ylläpitäjä

- Annetut syötteet

LightcurveID 1

- Tietokantaan kohdistuvat oletukset
Valokäyrä 1 on merkitty poistetuksi
- Metodi palautti

boolean true

Testataan tapaus, jossa käyttäjätaso on ylläpitäjä ja valokäyrää ei ole poistettu

- Annetut syötteet

LightcurveID 1

- Tietokantaan kohdistuvat oletukset
Valokäyrää 1 ei ole merkitty poistetuksi
- Metodi palautti

Virhe 0 LightcurveID not found, or error searching it, 17

Testataan tapaus, jossa käyttäjätaso on ylläpitäjä ja valokäyrää ei ole

- Annetut syötteet

LightcurveID -1

- Tietokantaan kohdistuvat oletukset
Valokäyrää -1 ei ole kannassa
- Metodi palautti

Virhe 0 LightcurveID not found, or error searching it, 17

4.22 addTrajectory

Metodi testataan seuraavilla eri tapauksilla:

- Syöte on null-viite
- Syötteenä annetun TrajectoryFormContainerin joku kenttä on null

Metodi testattiin seuraavasti:

- Annetut syötteet

data null

- Palauttaa

false

- Annetut syötteet

data containerissa kaikki arvot null

- Palauttaa

false

4.23 spamRecommender

Myös spamRecommender-metodi saa syötteenä vain long-tyyppisen muuttujan, joka tässä tapauksessa edustaa käyttäjän sisäistä järjestysnumeroa. Metodin toimintaa tulee testata selvästi virheellisellä syötteellä (negatiivinen käyttäjännumero), sellaisella positiivisella käyttäjänumerolla, jota järjestelmästä ei löydy, sekä järjestelmästä löytyvillä käyttäjännumeroilla. Lisäksi metodin toiminnan testaamiseksi tietokantaan tarvitaan käyttäjä, jonka ilmoittama suosittelijan sähköpostiosoite on epäkelpo. Testeissä ylläpitäjän sähköpostiosoitteena on käytetty tjhannin@cs.helsinki.fi, jotta myös tieto epäonnistuuneesta lähettämisestä saadaan perille.

- Annetut syötteenä:

UserID 1

RecommenderEMail tjhannin@cs.helsinki.fi

- Api palautti:

true

- Vastaanotettu sähköpostiviesti:

From: tjhannin@cs.helsinki.fi

To: tjhannin@cs.helsinki.fi

Subject: SAPC Recommendations

This is default reference message, username = Administrator, recommender = Tomi, login = Tahiti, password = \$password

- Annetut syötteenä:

UserID 2

RecommenderEMail ei_oo_mailiosote

- Api palautti:

true

- Vastaanotettu sähköpostiviesti:

Returned mail: see transcript for details

— The following addresses had permanent fatal errors —

<ei_oo_mailiosote>

(reason: 550 User unknown.)

(expanded from: <ei_oo_mailiosote>)

- Annetut syötteenä:

UserID -1234

- Api palautti:

false

Virhe User not found (#37)

- Annetut syötteet:

UserID 123544

- Api palautti:

false

Virhe User not found (#37)

Toinen testityyppi metodille saadaan sen tarvitsemista käyttöoikeuksista. Metodien toimintaa tulee testata tilanteissa, jossa sen käyttäjä ei vielä ole kirjautunut järjestelmään ollenkaan, sekä tilanteissa, jossa kirjautunut käyttäjä on käyttäjätasoltaan syöttäjä, eikä ylläpitäjä.

- Annetut syötteet:

User Käyttäjä ei ole kirjautunut sisään.

- Api palautti:

false

Virhe Unable to get userdata from session (#1)

- Annetut syötteet:

UserLevel 2 (TahitiLibrary.USER_SUBMITTER)

- Api palautti:

false

Virhe Action not allowed (#2)

4.24 spamUser

Metodille annetaan syötteenä vain käyttäjän järjestelmän sisäinen tunnistenumero, joten testi voidaan suorittaa kahdella lukuarvolla. Toinen on epäkelvo, toinen kelvollinen. Kelvollisen luvun ilmaiseva käyttäjä löytyy järjestelmän tietokannasta, epäkelvon ei. Paluuarvo toiminnon onnistuessa on true, muutoin false.

- Annetut syötteet:

int 0 (epäkelvo käyttäjä testin aikoihin)

- Api palautti:

boolean arvoltaan false kaikilla käyttäjillä

- Annetut syötteet:
 - `int` 100 (Väliaikaiskäyttäjä)
- Api palautti:
 - `boolean` arvoltaan `false` syöttäjän ja `nollakäyttäjän` tapauksessa, `ylläpitäjän` tapauksessa arvoltaan `true`

5 Validointitestausta

Validointitestausta varten ryhmälle perustettiin koneelle `db.cs.helsinki.fi` ryhmähakemisto, jossa järjestelmätestaus suoritettiin. Näin tehtiin, jotta testaus olisi mahdollisimman determinististä, ts. testitapausten ulkopuoliset ärsykkeet olisivat minimoituja. Käytännössä asuaa kontrolloidaan siten, että ennen muutoksien tekemistä tietokantaan tai testiluokkiin tulee asiasta saada lupa ryhmän IRC-kanavalla olijoilta. Käytännössä tämä tarkoittaa sitä, että aina testausta suoritettaessa testaaajien pitää olla tavattavissa ryhmän IRC-kanavalla (EFNet: #ohtu-tahiti).

Validointitestauksen testaustapaukset ovat jossain määrin samanlaisia integraatiotestauksen kanssa. Validointitestauksessa kuitenkin kiinnitetään enemmän huomiota järjestelmän toimiseen kokonaisuutena, ja testitapausten perustapauksina pidetään määrittelydokumentista löytyviä toimintoja. Validointitestauksen yksi päätavoitteista onkin tarkistaa, että kaikki määritellyt järjestelmän toiminnot toimivat oikealla tavalla. Lisäksi validointitestauksessa suoritetaan testausta integraatiotestauksessa esiintyvien tapausten mukaisesti, mutta testit suoritetaan käyttöliittymän kautta.

Validointitestauksessa myös tarkistettiin, että järjestelmä toimii määrittelydokumentissa mainituilla selaimilla. Näitä selaimia ovat Microsoft Internet Explorer 4.0, Netscape Navigator 4.0+ ja Mozilla 1.1.0.

5.1 Valokäyrien haku

Valokäyrän haun testauksen perustapaukset löytyvät integraatiotestauksen kohdasta `getLightcurves`. Näiden testitapausten ohessa validointitestauksessa kiinnitetään erityistä huomiota saatujen tietojen oikeellisuuteen. Testauksessa tarkistetaan, että tiedot näytetään käyttäjille juuri sellaisina, kuin ne tietokannassa ovat. Lisäksi tarkistetaan, että valokäyrää visualisoiva kuva näyttää oikealta.

5.2 Valokäyrien syöttäminen järjestelmään

Testauksen perustapaukset löytyvät integraatiotestauksen `insertLightcurve`-osiosta. Validointitestauksessa tarkistetaan, että lisätyt valokäyrän näkyvät järjestelmässä oikealla tavalla, ja että käyttöliittymän antamat virhetiedot vastaavat todellisuutta. Lisäys toimi käyttöliittymän kautta vastaavalla tavalla, kuin integraatiotestauksessakin. Lisäyksistä aiheutuneet lokimerkinnot vastasivat operaatioita, sekä havaintopisteiden arvot laskettiin oikein..

5.3 Valokäyrien tulostaminen

Testauksen perustapaukset löytyvät integraatiotestauksen `makeDataFile`-osiosta sekä `makeRawFile`-osiosta, riippuen kumman tulostusmuodon käyttäjä valitsi. Validointitestauksessa tarkistetaan, että muodostetut tiedostot tulevat muodostetuiksi käyttäjän valitsemien parametrien mukaisesti. Käyttäjä voi valita jokaiselle erilliselle valokäyrälle oman maarataelementin sekä asteroidirataelementin. Lisäksi laskettavan tulostiedoston yhteydessä käyttäjä saa määritellä haluaako hän tulostiedoston havaintopisteet intensiteetti vai magnitudimuodossa.

Testitapauksina käytettiin 3 erillistä tapausta pienin variaatioin (ratatietoja muuteltiin):

1. Käyttäjä ei valitse yhtään valokäyrää ja haluaa raakadatan.
2. Käyttäjä valitsee kaikki valokäyrät ja haluaa raakadatan.
3. Käyttäjä valitsee muutaman valokäyrän ja valitsee niille järjestelmän tarjoamat ratatiedot.
4. Käyttäjä valitsee muutaman valokäyrän ja valitsee niille omavalintaiset ratatiedot.
5. Käyttäjä ei valitse yhtään valokäyrää ja haluaa tulosdatan intensiteettimuodossa/magnitudimuodossa.
6. Käyttäjä valitsee kaikki valokäyrät ja haluaa tulosdatan intensiteettimuodossa/magnitudimuodossa.
7. Käyttäjä valitsee muutaman valokäyrän, valitsee niille järjestelmän tarjoamat ratatiedot ja haluaa tulosdatan intensiteettimuodossa/magnitudimuodossa.
8. Käyttäjä valitsee muutaman valokäyrän, valitsee niille omavalintaiset ratatiedot ja haluaa tulosdatan intensiteettimuodossa/magnitudimuodossa.

5.4 Valokäyrien muuttaminen

Testauksen perustapaukset ovat integraatiotestauksen `changeLightcurve`-osiossa. Validointitestauksessa tarkistetaan, että muutettu valokäyrä todellakin on muutettu oikein, ja että virheellisistä arvoista kerrotaan tyydyttävällä tavalla.

5.5 Omien tietojen muuttaminen

Testaaminen suoritettiin integraatiotestauksessa määritellyin syöttein. Tulokset olivat toivotunlaisia, joten osio toimii moitteetta.

5.6 Salasanan nollaaminen

Salasanan nollaamisessa käytetyt perustestitapaukset on käsitelty integraatiotestauksen `resetPassword`-metodin yhteydessä. Järjestelmätestauksessa katsotaan myös, että järjestelmän lähettämä sähköposti on oikeaa muotoa ja varmistetaan että uusi, muutettu salasana toimii järjestelmään kirjautumisessa.

5.7 Käyttäjien tietojen hakeminen

Käyttäjien tietojen hakemisen perustestitapaukset ovat käsitelty integraatiotestauksen `getUsers`-sekä `getUsersFull`-metodien yhteydessä. Validointitestauksen tarkoituksena on varmistaa että käyttäjän antaessa syötteenä osittaisen tekstin käyttäjän nimestä, järjestelmä palauttaa käyttäjälle kaikki osittaista nimeä vastaavat käyttäjät sekä antaa käyttäjätasosta riippuen mahdollisuuden poistaa, muokata tai tarkastella valittavan käyttäjän tarkempia tietoja.

1. Tyhjä käyttäjähaku, valitaan jokin ylläpitäjä-tason käyttäjä.
2. Tyhjä käyttäjähaku, valitaan jokin syöttäjä-tason käyttäjä.
3. Tyhjä käyttäjähaku, valitaan jokin hyväksymistä odottava käyttäjä.
4. Haetaan käyttäjää osittaisella nimellä.
5. Haetaan käyttäjää kokonimellä.
6. Haetaan käyttäjää nimellä, jolla ei löydy yhtään tulosta.

5.8 Käyttäjien lisääminen ja poistaminen järjestelmästä

Käyttäjien lisääminen järjestelmään koostuu ylläpitäjän tekemästä rekisteröinninpyynnöstä, ja tämän pyynnön hyväksymisestä. Koska nämä ovat erillisiä toimintoja, jotka testataan myös itsenäisinä operaatioina, ei käyttäjän lisääminen vaadi erillistä testitapausta.

Käyttäjän poistamisen perustestitapaukset löytyvät integraatiotestauksen `removeUser`-metodin yhteydestä. Validointitestauksessa myös tarkistetaan, että käyttäjä todella poistui järjestelmästä, ja että operaation aiheuttama lokimerkintä on oikea. Koska lokimerkintöjä aiheuttaneita käyttäjiä ei poisteta järjestelmästä kokonaan, pitää toimintaa testata myös tällaisessa tilanteessa. Tämän jälkeen tarkistetaan, että käyttäjää ei enää näy käyttäjähaun listoissa. Järjestelmä toimi poistoissa odotetulla tavalla. Mikäli käyttäjä ei ollut aiheuttanut merkintöjä järjestelmän lokiin, poistui käyttäjä kokonaan. Jos taas lokimerkintöjä oli tullut, merkittiin käyttäjä poistetuksi asettamalla hänelle käyttäjätasoksi -1. Tällöin käyttäjää ei näkynyt käyttäjähaun listoissa.

Testattiin tapaukset:

- Rekisteröintipyynnön täyttäminen selaajana
- Rekisteröintipyynnön täyttäminen selaajana useaan kertaan samalta selaimelta
- Rekisteröintipyynnön täyttäminen vajavaisesti selaajana
- Rekisteröintipyynnön poistaminen
- Rekisteröintipyynnön hyväksyminen (käyttäjätason vaihto)
- Lokimerkintöjä tehneen käyttäjän poisto

Rekisteripyyntöä täytettäessä vajavaiset kentät tulivat oikein havaituiksi. Myös vapaaehtoiset kentät toimivat kuten piti. Useita rekisteröintipyyntöjä ei samasta sessiosta voinut suorittaa.

Rekisteröintipyyntöä poistettaessa eekisteripyyntö poistui järjestelmästä kokonaan, ja lokiin tuli oikea merkintä. Rekisteröintipyyntöä hyväksyttäessä testattiin myös kaikkien sähköpostien lähetys. Sähköpostit tulivat perille oikein, ja niissä ilmoitetulla salasanalla pystyi kirjautumaan sisään järjestelmään. Lokimerkintä oli jälleen oikeanlainen.

Kun lokimerkintöjä tehnyt käyttäjä poistettiin, hänestä jäi merkintä järjestelmään, ja oikeuksiksi hänelle oli asetettu -1. Poistomerkitty käyttäjä ei näkynyt käyttäjälistauksessa. Lokiin tuli oikeanlainen merkintä poistosta.

5.9 Asetusten muuttaminen

Asetusten muuttaminen on käyty aiemmin läpi metodin `changeSettings` osalta. Testiin kuuluu kelpojen ja epäkelpojen arvojen läpimenon tarkistaminen. Lisäksi pakolliset kentät `AdminEMail`, `MailServer` ja `ServerPort` tulee olla täytettyinä.

Käyttöliittymän antamaan `User Information`-kenttäpohjaan kokeiltiin seuraavia muutoksia sekä submitterin että adminin käyttöoikeuksilla

- Username kohta asetettiin tyhjäksi.
- New password ja Confirm password samoja, old password oikea salasana.
- New password ja Confirm password samoja, old password ei ole oikea.
- New password ja Confirm password eivät ole samoja, old password oikea.
- New password ja Confirm password eivät ole samoja, old password ei ole oikea.
- Email-kenttään laitetaan epäkelvollinen sähköpostiosoite
- Kaikki kentät asetetaan tyhjäksi.
- Kaikkiin kenttiin asetetaan joku kelvollinen arvo.

Arvot muuttuivat oikein

5.10 Järjestelmän tapahtumien selaaminen

Järjestelmän tapahtumien selaamisen perustestit ovat kuvattu metodin `getLogEntries` integraatio-testauksen yhteydessä. Validointitestauksessa tarkastetaan, että käyttäjän antama aikahaarukka vastaa ruudulle ilmestyvien tapahtumien aikahaarukkaa. Myös varmistetaan tyhjän aikahaarukan hakevan kahden edellisen kuukauden tapahtumat.

1. Alkupäivämäärä on suurempi kuin loppupäivämäärä.
2. Alkupäivämäärä tai loppupäivämäärä tai molemmat ovat tyhjiä.
3. Alkupäivämäärä tai loppupäivämäärä tai molemmat eivät ole valideja päiväyksiä.

4. Alkupäivämäärä sekä loppupäivämäärä muodostavat aikavälin, jolla on tehty merkintöjä tapahtumahistoriaan.

5.11 Kirjautuminen

Testitapaukset esitellään integraatiotestauksen login-osiossa. Testattavana ovat kelvolliset ja epäkelvolliset syötteet.

Sisäänkirjautuminen testattiin kelvollisilla ja epäkelvollisilla arvoilla arvoilla. Epäkelvollisia arvoja testattiin siten, että toinen syötteistä on epäkelpo ja siten, että molemmat on epäkelpoja. Kelvollisilla arvoilla sisäänkirjautuminen onnistui ja ruutuun tulostui käyttäjän käyttäjätason mukaiset valikot. Jos molemmat arvot oli epäkelvollisia tai salasana olemassa oleva ja käyttäjätunnus väärä, tulostui: "Invalid username or password", eikä sisäänkirjautuminen onnistunut. Kun username oli kelvollinen, mutta password väärä, tulostui: "Invalid password". Tällöinkään sisäänkirjautuminen ei onnistunut.

5.12 Käyttäjätietojen muuttaminen

Käyttäjätietojen muuttaminen on vastaa integraatiotestauksen setUserData-metodin testausta. Testi muodostuu kelvollisten ja epäkelvollisten arvojen läpimenon tarkastamisesta.

5.13 Rataelementtien lisääminen

Asteroidien syöttö testattiin syöttämällä järjestelmään kaikki Nasalta saadut asteroidit. Syötettyä materiaalia tarkastettiin pistokokein.

5.14 Atlas-syötin

Koska Atlas-syötin toteutetaan varsinaisen järjestelmän ulkoisena komponenttina, ei sen testausprioriteetti ole kaikista korkein. Lisäksi kaikki aineisto, jota syöttimellä tullaan käsittelemään on jo olemassa, eikä uutta Atlas-muodossa olevaa aineistoa pitäisi enää tulla. Näiden seikkojen vuoksi Atlas-syöttimen testaus tapahtuu suoraviivaisesti syöttimen käyttämisellä. Ajon jälkeen tarkistetaan syöttimen tietokantaan laittamat havainnot ja tarkistetaan siirrettyjen tietojen oikeellisuus. Ennen testausta järjestelmään lisättiin tieto jokaisesta testiaineiston kohde asteroidista, lukuunottamatta asteroidia 243 Ida. Testauksessa käytettiin seuraavia Atlas-tiedostoja, listassa kerrottu käytetyn Atlas-tiedoston nimi, sekä md5sum.

- 4c655e717717e267e8aa1694e7fdec29 1036_Ganymed.apc
- 583a6b6c8a9dc1f5dd1c4ab0f97514dd 1580_Betulia.apc
- 0516e5e31052c991eee94da962198623 201_Penelope.apc
- a2e7a8d45fd822f43e3ddb616ec189d3 22_Kalliope.apc
- 7780b240f55fcb5717e5bec73804739 243_Ida.apc

- cd63fcc14fe393a9e1ebe4420984fa00 354_Eleonora.apc
- 868d374331f552b619c3b5ec6cc6f9ef 3908.apc
- e9c943b8051b94c265208a8bbd460283 44_Nysa.apc
- 5288fbfcc664108d1c145fd739e3e728 532_Herculina.apc
- 3b01075bad77dcfc02575115fb22141c 5587.apc
- 5841ead8b7624ca330322c10584867c6 704_Interamnia.apc
- 0ec150abbf19c2ca021ddd4053776755 87_Sylvia.apc

Testiajossa 234 valokäyrää siirtyi järjestelmään. Siirretyistä Atlas-tiedostoista tarkistettiin pistokokein, että siirtynyt informaatio vastasi Atlas-tiedostoissa olevaa informaatiota. Kun Atlas-siirrin ajettiin kaksi kertaa peräkkäin tällä samalla aineistolla, ei tietokantaan tullut uusia havaintoja.

Myös tiedostot, jotka Atlas-syötin hylkää tarkistettiin pistokokein. Tällä pyritään varmistamaan, ettei syötin hylkää oikean muotoisia tiedostoja. Itse Atlas-syötin hylkäsi tapauksia, joissa oli selvästi puutteellisia tietoja, kuten havaintopaikan (ObservationSite) puuttuminen. Lisäksi joissain tiedostoissa oli havaintoja, joissa oli kolmen pisteen havaintotiedot, mutta otsikkolistalla mainitaan vain kaksi sarake.

Atlas-siirrin hyväksyi joitain sellaisia tietoja, joita TahitiApin insertLightcurveAtlas-metodi ei hyväksynyt. Tällaisia tilanteita syntyi, mikä siirron kohdetta ei löytynyt järjestelmästä, tai havaintoajoissa oli vääränlaista informaatiota. Joissain tiedostoissa zeroTimenä oli esimerkiksi arvo 1980, ja havaintoajat oli luokkaa 20. Tämä muutettuna dateksi on 4708-05-16 BC, mikä ei ole oikea havaintoaika.

5.15 Rekisteröintipyynnön täyttäminen

Rekisteröintipyynnön tekemisen testitapaukset löytyvät integraatiotestaus-luvusta alaotsikon registrationRequest alta. Kuitenkin siten, että versionhallintaan liittyviä asioita ei testata validointitestauksen yhteydessä.

5.16 Käyttäjien hyväksyminen tai hylkääminen

Testataan hyväksymällä ja hylkäämällä rekisteröintipyyntö. Hyväksyntä ja hylkääminen tapahtuu nappia painamalla.

5.17 Valokäyrän poistaminen

Valokäyrän poistamista testataan ensinnäkin tapauksessa, jolloin käyttäjä ei ole ylläpitäjä eikä ole lisännyt valokäyrää järjestelmään. Tällöin poiston ei tulisi onnistua. Tämän jälkeen testataan tapaus, jossa käyttäjä ei ole ylläpitäjä, mutta on syöttänyt poistettavan valokäyrän järjestelmään. Tällöin valokäyrä pitäisi poistomerkitsemään, muttei kokonaan poistamaan. Lopuksi testataan, että ylläpitäjä voi sekä poistomerkitä, että poistaa valokäyriä järjestelmästä.

Testattiin tapaukset:

- Ylläpitäjä merkitsi valokäyrän poistetuksi
- Ylläpitäjä palautti valokäyrän
- Ylläpitäjä poisti poistomerkatun valokäyrän
- Valokäyrän syöttäjä merkitsi oman valokäyränsä poistetuksi

Tapaukset toimivat odotetusti. Ylläpitä pystyi sekä poistomerkitsemään, että poistamaan valokäyrän. Syöttäjä kykeni poistomerkitsemään ainoastaan syöttämänsä valokäyrän. Lokimerkinnot olivat oikeelliset.

Salasanan nollaus ja lähettäminen toimii oikein. Kun lähetysohjelmaa painaa, sähköpostilla saapuu uusi salasana, joka käy samantien järjestelmään.

Valokäyrien hakeminen:

Kokeilin erilaisia arvoja jokaiseen kenttään valokäyrien etsintäomakkeessa asteroidi Sylvian kohdalla. Softa toimi ihan niin kuin sen pitikin.

Valokäyrien muuttaminen:

Valokäyrien muuttaminen pelaa erinomaisesti. Atlas-valokäyriä ei voi muuttaa, mutta se oli tarkoituskin.

Liite 1. Atlasimportterin testimateriaalia 1

```

OBSERVER.....: Joojoo
OBJECT.....: 22 Kalliope
OBSERVING SITE: W 104.02 +30.67 McDonald Observatory, Mount Locke, Texas, USA
REFERENCE.....: Zappala and van Houten-Groeneveld (1979)
INFORMATION...: ASPECT ERROR !!
RELATIVE PHOT.: T
LT CORRECTED...: T
PHOT. SYSTEM...: Onmikäon
DETECTOR.....: Photoelectric
COLUMNS.....: \#UBVRI?.
UNIT OF TIME...: 1.0 day
ZERO TIME.....: 2400000.0
ASPECT DATA...: 2.6774 3.1850 17.10 85.04 6.79 2445083.60000 J2000.0
GEOMETRY.....:
    system: J2000.0 ecliptic
    epoch: 2445083.6 (1982 Apr 24.1)
    Earth: -0.835301 -0.560150 -0.000013 0.009302 -0.014353 0.000000
    object: -0.561880 2.590544 0.376720 -0.010536 -0.001884 0.002182
UPDATE.....: 0
ACQUISITION...: Original digital data
TELESCOPE.....: 82 inch reflector
REDUCED MAG...: F
CURVE IDENTITY: Fig. 9, filled circles
OBSERVING TIME: 1953 Feb 15.4
COMPOSITE.....: T
DATA:
1 2 3 4 5 6 7 8
=====
OBJECT.....: Kalliope
OBSERVING SITE: W 104.02 +30.67 McDonald Observatory, Mount Locke, Texas, USA
REFERENCE.....: Zappala and van Houten-Groeneveld (1979)
INFORMATION...: ASPECT ERROR !!
RELATIVE PHOT.: F
LT CORRECTED...: F
PHOT. SYSTEM...: Onmikäon
DETECTOR.....: Photoelectric
COLUMNS.....: \#UBVRI?.
UNIT OF TIME...: hour
ZERO TIME.....: 2400000.0
ASPECT DATA...: 2.6774 3.1850
GEOMETRY.....:
    system: J2000.0 ecliptic

```

```

epoch: 2445083.6 (1982 Apr 24.1)
Earth:  -0.835301  -0.560150  -0.000013   0.009302 -0.014353  0.000000
object:  -0.561880   2.590544   0.376720  -0.010536 -0.001884  0.002182
UPDATE.....: 0
ACQUISITION...: Original digital data
TELESCOPE.....: 82 inch reflector
REDUCED MAG...: F
CURVE IDENTITY: Fig. 9, filled circles
OBSERVING TIME: 1953 Feb 15.4
COMPOSITE.....: T
DATA:
1 2 3 4 5 6 7 8
=====
OBJECT.....: 123 Kalliope
OBSERVING SITE: W 104.02 +30.67 McDonald Observatory, Mount Locke, Texas, USA
REFERENCE.....: Zappala and van Houten-Groeneveld (1979)
INFORMATION...: ASPECT ERROR !!
LT CORRECTED...: F
PHOT. SYSTEM...: Onmikäon
DETECTOR.....: Photoelectric
COLUMNS.....: \#UBVRI?.
ZERO TIME.....: 2400000.0
ASPECT DATA...: 2.6774 3.1850
GEOMETRY.....:
    system: J2000.0 ecliptic
    epoch: 2445083.6 (1982 Apr 24.1)
    Earth:  -0.835301  -0.560150  -0.000013   0.009302 -0.014353  0.000000
    object:  -0.561880   2.590544   0.376720  -0.010536 -0.001884  0.002182
UPDATE.....: 0
ACQUISITION...: Original digital data
TELESCOPE.....: 82 inch reflector
REDUCED MAG...: F
CURVE IDENTITY: Fig. 9, filled circles
OBSERVING TIME: 1953 Feb 15.4
COMPOSITE.....: T
DATA:
1 2 3 4 5 6 7 8
=====
INFORMATION...: ASPECT ERROR !!
=====
OBJECT.....: 123 Kalliope
=====
OBJECT.....: 123 Kalliope

```

OBSERVING SITE: W 104.02 +30.67 McDonald Observatory, Mount Locke, Texas, USA

=====

OBJECT.....: 123 Kalliope

OBSERVING SITE: W 104.02 +30.67 McDonald Observatory, Mount Locke, Texas, USA

ZERO TIME.....: 2400000.0

=====

OBJECT.....: 123 Kalliope

OBSERVING SITE: W 104.02 +30.67 McDonald Observatory, Mount Locke, Texas, USA

ZERO TIME.....: 2400000.0

LT CORRECTED...: F

=====

OBJECT.....: 123 Kalliope

OBSERVING SITE: W 104.02 +30.67 McDonald Observatory, Mount Locke, Texas, USA

ZERO TIME.....: 2400000.0

LT CORRECTED...: F

COLUMNS.....: \#UBVRI?.

=====

OBJECT.....: 123 Kalliope

OBSERVING SITE: W 104.02 +30.67 McDonald Observatory, Mount Locke, Texas, USA

ZERO TIME.....: 2400000.0

LT CORRECTED...: F

COLUMNS.....: \#UBVRI?.

DATA:

1 2 3 4 5 6 7 8

=====

OBJECT.....: 123 Kalliope

OBSERVING SITE: W 104.02 +30.67 McDonald Observatory, Mount Locke, Texas, USA

ZERO TIME.....: 2400000.0

LT CORRECTED...: F

COLUMNS.....: \#UBVRI?.

GEOMETRY.....:

 system: J2000.0 ecliptic

 epoch: 2445083.6 (1982 Apr 24.1)

DATA:

1 2 3 4 5 6 7 8

=====

OBJECT.....: 123 Kalliope

OBSERVING SITE: W 104.02 +30.67 McDonald Observatory, Mount Locke, Texas, USA

ZERO TIME.....: 2400000.0

LT CORRECTED...: F

COLUMNS.....: \#UBVRI?.

GEOMETRY.....:

 system: J2000.0 ecliptic

epoch: 2445083.6 (1982 Apr 24.1)
Earth: -0.835301 -0.560150 -0.000013 0.009302 -0.014353 0.000000

DATA:

1 2 3 4 5 6 7 8

```
=====
OBJECT.....: 123 Kalliope
OBSERVING SITE: W 104.02 +30.67 McDonald Observatory, Mount Locke, Texas, USA
ZERO TIME.....: 2400000.0
LT CORRECTED...: F
COLUMNS.....: \#UBVRI?.
GEOMETRY.....:
    system: J2000.0 ecliptic
    epoch: 2445083.6 (1982 Apr 24.1)
    Earth: -0.835301 -0.560150 -0.000013 0.009302 -0.014353 0.000000
    object: -0.561880 2.590544 0.376720 -0.010536 -0.001884 0.002182
```

DATA:

1 2 3 4 5 6 7 8

```
=====
```

Liite 2. Atlasimportterin testimateriaalia 2

Operation started succesfully 2.5.2003 18:24:54

*** Parsing file: hakemisto/AtlasInTestifile ***

LightcurveID: null

AsteroidNumber: 22

AsteroidName: Kalliope

AsteroidDesignation: null

ObservingSite: W 104.02 +30.67 McDonald Observatory, Mount Locke, Texas, USA

Observer: Joojoo

ZeroTime: 2400000.0

UnitOfTime: 0

AbsolutePhotometry: false

ZeroMagnitude: null

LightTimeCorrected: true

Columns: 0 2 3 4 5 6 7 1

Data:

1 2 3 4 5 6 7 8

PhotometricSystem: Onmikäon

Detector: Photoelectric

TimeStandard: null

Reference: Zappala and van Houten-Groeneveld (1979)

Information: ASPECT ERROR !!

UPDATE.....: 0

ACQUISITION...: Original digital data

TELESCOPE.....: 82 inch reflector

REDUCED MAG...: F

CURVE IDENTITY: Fig. 9, filled circles

OBSERVING TIME: 1953 Feb 15.4

COMPOSITE.....: T

System: J2000.0 ecliptic

Aspect Time: 2445083.6

Aspect Data: 2.6774 3.185 17.1 85.04 6.79

Earth: -0.835301 -0.56015 -1.3E-5 0.009302 -0.014353 0.0

Object: -0.56188 2.590544 0.37672 -0.010536 -0.001884 0.002182 0.0

Lightcurve imported succesfully.

A lightcurve-section ended. Line: 28

LightcurveID: null
 AsteroidNumber: null
 AsteroidName:
 AsteroidDesignation: null
 ObservingSite: W 104.02 +30.67 McDonald Observatory, Mount Locke, Texas, USA
 Observer: null
 ZeroTime: 2400000.0
 UnitOfTime: 1
 AbsolutePhotometry: true
 ZeroMagnitude: null
 LightTimeCorrected: false
 Columns: 0 2 3 4 5 6 7 1
 Data:
 1 2 3 4 5 6 7 8

PhotometricSystem: Onmikäon
 Detector: Photoelectric
 TimeStandard: null
 Reference: Zappala and van Houten-Groeneveld (1979)
 Information: ASPECT ERROR !!

UPDATE.....: 0
 ACQUISITION...: Original digital data
 TELESCOPE.....: 82 inch reflector
 REDUCED MAG...: F
 CURVE IDENTITY: Fig. 9, filled circles
 OBSERVING TIME: 1953 Feb 15.4
 COMPOSITE.....: T

System: J2000.0 ecliptic
 Aspect Time: 2445083.6
 Aspect Data: null
 Earth: -0.835301 -0.56015 -1.3E-5 0.009302 -0.014353 0.0
 Object: -0.56188 2.590544 0.37672 -0.010536 -0.001884 0.002182 0.0

Error: AspectData not set.
 A lightcurve-section ended. Line: 55

 LightcurveID: null
 AsteroidNumber: 123
 AsteroidName: Kalliope
 AsteroidDesignation: null
 ObservingSite: W 104.02 +30.67 McDonald Observatory, Mount Locke, Texas, USA

Observer: null
 ZeroTime: 2400000.0
 UnitOfTime: null
 AbsolutePhotometry: null
 ZeroMagnitude: null
 LightTimeCorrected: false
 Columns: 0 2 3 4 5 6 7 1
 Data:
 1 2 3 4 5 6 7 8

PhotometricSystem: Onmikäon
 Detector: Photoelectric
 TimeStandard: null
 Reference: Zappala and van Houten-Groeneveld (1979)
 Information: ASPECT ERROR !!

UPDATE.....: 0
 ACQUISITION...: Original digital data
 TELESCOPE.....: 82 inch reflector
 REDUCED MAG...: F
 CURVE IDENTITY: Fig. 9, filled circles
 OBSERVING TIME: 1953 Feb 15.4
 COMPOSITE.....: T

System: J2000.0 ecliptic
 Aspect Time: 2445083.6
 Aspect Data: null
 Earth: -0.835301 -0.56015 -1.3E-5 0.009302 -0.014353 0.0
 Object: -0.56188 2.590544 0.37672 -0.010536 -0.001884 0.002182 0.0

Error: AspectData not set.
 A lightcurve-section ended. Line: 80

 LightcurveID: null
 AsteroidNumber: null
 AsteroidName: null
 AsteroidDesignation: null
 ObservingSite: null
 Observer: null
 ZeroTime: null
 UnitOfTime: null
 AbsolutePhotometry: null
 ZeroMagnitude: null

LightTimeCorrected: null
Columns: null
Data:
null
PhotometricSystem: null
Detector: null
TimeStandard: null
Reference: null
Information: ASPECT ERROR !!
System: null
Aspect Time: NaN
Aspect Data: null
Earth: null
Object: null

Error: AsteroidName, AsteroidNumber and AsteroidDesignation not set. Fill at least one.
Error: ObservingSite not set.
Error: ZeroTime not set.
Error: LightTimeCorrected not set.
Error: Columns not set.
Error: Data not set.
Error: Geometry earth not set.
Error: Geometry object not set.
Error: AspectData not set.
A lightcurve-section ended. Line: 82

LightcurveID: null
AsteroidNumber: 123
AsteroidName: Kalliope
AsteroidDesignation: null
ObservingSite: null
Observer: null
ZeroTime: null
UnitOfTime: null
AbsolutePhotometry: null
ZeroMagnitude: null
LightTimeCorrected: null
Columns: null
Data:
null
PhotometricSystem: null

Detector: null
TimeStandard: null
Reference: null
Information:
System: null
Aspect Time: NaN
Aspect Data: null
Earth: null
Object: null

Error: ObservingSite not set.
Error: ZeroTime not set.
Error: LightTimeCorrected not set.
Error: Columns not set.
Error: Data not set.
Error: Geometry earth not set.
Error: Geometry object not set.
Error: AspectData not set.
A lightcurve-section ended. Line: 84

LightcurveID: null
AsteroidNumber: 123
AsteroidName: Kalliope
AsteroidDesignation: null
ObservingSite: W 104.02 +30.67 McDonald Observatory, Mount Locke, Texas, USA
Observer: null
ZeroTime: null
UnitOfTime: null
AbsolutePhotometry: null
ZeroMagnitude: null
LightTimeCorrected: null
Columns: null
Data:
null
PhotometricSystem: null
Detector: null
TimeStandard: null
Reference: null
Information:
System: null
Aspect Time: NaN

Aspect Data: null
Earth: null
Object: null

Error: ZeroTime not set.
Error: LightTimeCorrected not set.
Error: Columns not set.
Error: Data not set.
Error: Geometry earth not set.
Error: Geometry object not set.
Error: AspectData not set.
A lightcurve-section ended. Line: 87

LightcurveID: null
AsteroidNumber: 123
AsteroidName: Kalliope
AsteroidDesignation: null
ObservingSite: W 104.02 +30.67 McDonald Observatory, Mount Locke, Texas, USA
Observer: null
ZeroTime: 2400000.0
UnitOfTime: null
AbsolutePhotometry: null
ZeroMagnitude: null
LightTimeCorrected: null
Columns: null
Data:
null
PhotometricSystem: null
Detector: null
TimeStandard: null
Reference: null
Information:
System: null
Aspect Time: NaN
Aspect Data: null
Earth: null
Object: null

Error: LightTimeCorrected not set.
Error: Columns not set.

Error: Data not set.
Error: Geometry earth not set.
Error: Geometry object not set.
Error: AspectData not set.
A lightcurve-section ended. Line: 91

LightcurveID: null
AsteroidNumber: 123
AsteroidName: Kalliope
AsteroidDesignation: null
ObservingSite: W 104.02 +30.67 McDonald Observatory, Mount Locke, Texas, USA
Observer: null
ZeroTime: 2400000.0
UnitOfTime: null
AbsolutePhotometry: null
ZeroMagnitude: null
LightTimeCorrected: false
Columns: null
Data:
null
PhotometricSystem: null
Detector: null
TimeStandard: null
Reference: null
Information:
System: null
Aspect Time: NaN
Aspect Data: null
Earth: null
Object: null

Error: Columns not set.
Error: Data not set.
Error: Geometry earth not set.
Error: Geometry object not set.
Error: AspectData not set.
A lightcurve-section ended. Line: 96

LightcurveID: null
AsteroidNumber: 123

AsteroidName: Kalliope
AsteroidDesignation: null
ObservingSite: W 104.02 +30.67 McDonald Observatory, Mount Locke, Texas, USA
Observer: null
ZeroTime: 2400000.0
UnitOfTime: null
AbsolutePhotometry: null
ZeroMagnitude: null
LightTimeCorrected: false
Columns: 0 2 3 4 5 6 7 1
Data:
null
PhotometricSystem: null
Detector: null
TimeStandard: null
Reference: null
Information:
System: null
Aspect Time: NaN
Aspect Data: null
Earth: null
Object: null

Error: Data not set.
Error: Geometry earth not set.
Error: Geometry object not set.
Error: AspectData not set.
A lightcurve-section ended. Line: 102

LightcurveID: null
AsteroidNumber: 123
AsteroidName: Kalliope
AsteroidDesignation: null
ObservingSite: W 104.02 +30.67 McDonald Observatory, Mount Locke, Texas, USA
Observer: null
ZeroTime: 2400000.0
UnitOfTime: null
AbsolutePhotometry: null
ZeroMagnitude: null
LightTimeCorrected: false
Columns: 0 2 3 4 5 6 7 1

Data:

1 2 3 4 5 6 7 8

PhotometricSystem: null

Detector: null

TimeStandard: null

Reference: null

Information:

System: null

Aspect Time: NaN

Aspect Data: null

Earth: null

Object: null

Error: Geometry earth not set.

Error: Geometry object not set.

Error: AspectData not set.

A lightcurve-section ended. Line: 110

LightcurveID: null

AsteroidNumber: 123

AsteroidName: Kalliope

AsteroidDesignation: null

ObservingSite: W 104.02 +30.67 McDonald Observatory, Mount Locke, Texas, USA

Observer: null

ZeroTime: 2400000.0

UnitOfTime: null

AbsolutePhotometry: null

ZeroMagnitude: null

LightTimeCorrected: false

Columns: 0 2 3 4 5 6 7 1

Data:

1 2 3 4 5 6 7 8

PhotometricSystem: null

Detector: null

TimeStandard: null

Reference: null

Information:

System: J2000.0 ecliptic

Aspect Time: 2445083.6

Aspect Data: null
Earth: null
Object: null

Error: Geometry earth not set.
Error: Geometry object not set.
Error: AspectData not set.
A lightcurve-section ended. Line: 121

LightcurveID: null
AsteroidNumber: 123
AsteroidName: Kalliope
AsteroidDesignation: null
ObservingSite: W 104.02 +30.67 McDonald Observatory, Mount Locke, Texas, USA
Observer: null
ZeroTime: 2400000.0
UnitOfTime: null
AbsolutePhotometry: null
ZeroMagnitude: null
LightTimeCorrected: false
Columns: 0 2 3 4 5 6 7 1
Data:
1 2 3 4 5 6 7 8

PhotometricSystem: null
Detector: null
TimeStandard: null
Reference: null
Information:
System: J2000.0 ecliptic
Aspect Time: 2445083.6
Aspect Data: null
Earth: -0.835301 -0.56015 -1.3E-5 0.009302 -0.014353 0.0
Object: null

Error: Geometry object not set.
Error: AspectData not set.
A lightcurve-section ended. Line: 133

LightcurveID: null
AsteroidNumber: 123
AsteroidName: Kalliope
AsteroidDesignation: null
ObservingSite: W 104.02 +30.67 McDonald Observatory, Mount Locke, Texas, USA
Observer: null
ZeroTime: 2400000.0
UnitOfTime: null
AbsolutePhotometry: null
ZeroMagnitude: null
LightTimeCorrected: false
Columns: 0 2 3 4 5 6 7 1
Data:
1 2 3 4 5 6 7 8

PhotometricSystem: null
Detector: null
TimeStandard: null
Reference: null
Information:
System: J2000.0 ecliptic
Aspect Time: 2445083.6
Aspect Data: null
Earth: -0.835301 -0.56015 -1.3E-5 0.009302 -0.014353 0.0
Object: -0.56188 2.590544 0.37672 -0.010536 -0.001884 0.002182 0.0

Error: AspectData not set.
A lightcurve-section ended. Line: 146

All the lightcurves were not imported.
Copied file hakemisto/AtlasInTestifile to kakku.

Operation ended 2.5.2003 18:24:54
