

TIETOKANTA MERIKOTKIEN SEURANTAAN

Suunnitteludokumentti

Versiohistoria:

Versio	Päivämäärä	Kuvaus	Tekijä
0.1	5.3.2004	Ensimmäinen luonnos	Henri Jääskeläinen
0.2	8.3.2004	Toinen luonnos	Henri Jääskeläinen
0.3	14.3.2004	Kolmas luonnos	Henri Jääskeläinen
1.0	17.3.2004	Ensimmäinen versio	Henri Jääskeläinen
1.1	19.3.2004	Toinen versio	Henri Jääskeläinen
1.2	28.3.2004	Kolmas versio	Henri Jääskeläinen

Helsinki, 28.03.2004

Harri Honkanen
Henri Jääskeläinen
Jouko Kleemola
Timo Maaranen
Markus Mattila
Sipi Turner

HELSINGIN YLIOPISTO
Tietojenkäsittelytieteen laitos
Ohjelmistotuotantoprojekti Hali2

Sisällys

1 JOHDANTO.....	1
1.1 TUOTTEEN TAUSTA JA TARKOITUS.....	1
1.2 TÄRKEIMMÄT VAATIMUKSET.....	2
1.3 ERIKOISSANASTO JA KÄYTETYT LYHENTEET.....	2
2 SUUNNITTELUN JA TOTEUTUKSEN RAJOITTEET.....	4
2.1 NOUDATETTAVAT STANDARDIT JA TARVITTAVAT OHJELMAT.....	4
2.2 LUOKKAKIRJASTOT JA AJURIT.....	5
2.3 OHJELMOINTIKIELI JA -TYYLII.....	5
3 JÄRJESTELMÄN YLEISKUVAUS.....	6
4 ARKKITEHTUURIKUVAUS.....	7
4.1 KOMPONENTTIEN VÄLISET SUHTEET.....	7
4.2 LUOKKIEN VÄLISET SUHTEET.....	8
4.2.1 Luokkakaavio.....	8
4.3 PAKKAUS FI.HY.HALL.DB.....	9
4.3.1 ConnectionPool.....	10
4.3.2 Table.....	12
4.3.3 OperationResults.....	15
4.3.4 SearchResults.....	15
4.3.5 DatabaseOperation.....	17
4.3.6 SearchOperation.....	18
4.3.7 InsertOperation.....	20
4.3.8 UpdateOperation.....	21
4.3.9 DeleteOperation.....	22
4.4 PAKKAUS FI.HY.HALL.SERVLET.....	23
4.4.1 Hali.....	23
4.5 PAKKAUS FI.HY.HALL.GENERAL.....	24
4.5.1 HaliProperties.....	24
4.5.2 FileSave.....	25
4.6 PAKKAUS FI.HY.HALL.ACTION.....	26
4.6.1 Action.....	26
4.6.2 LoginValidation.....	26
4.7 PAKKAUS FI.HY.HALL.ACTION.SEARCH.....	27
4.7.1 Search.....	27
4.7.2 PesaSearch.....	27
4.7.3 KuntaSearch.....	27
4.7.4 TarkastajaSearch.....	27
4.7.5 LajiSearch.....	28
4.7.6 AputauluSearch.....	28
4.7.7 HistoriaSearch.....	28
4.7.8 ReviiriSearch.....	29
4.7.9 SaalisSearch.....	29
4.7.10 MyrkkySearch.....	29
4.8 PAKKAUS FI.HY.HALL.ACTION.UPDATE.....	30
4.8.1 Update.....	30
4.8.2 PesaUpdate.....	30
4.8.3 KuntaUpdate.....	30
4.8.4 TarkastajaUpdate.....	30
4.8.5 LajiUpdate.....	31
4.8.6 AputauluUpdate.....	31
4.8.7 HistoriaUpdate.....	31
4.8.8 ReviiriUpdate.....	32
4.8.9 SaalisUpdate.....	32
4.8.10 MyrkkyUpdate.....	32
4.9 PAKKAUS FI.HY.HALL.ACTION.REPORT.....	33
4.9.1 Report.....	33
4.9.2 ReportA.....	33
4.9.3 ReportB.....	33
4.9.4 ReportC.....	33
4.9.5 ReportD.....	34

4.9.6ReportE.....	34
4.9.7ReportF.....	34
4.9.8ReportG.....	35
4.9.9ReportH.....	35
4.9.10ReportI.....	35
4.9.11ReportJ.....	35
4.9.12ReportK.....	36
4.9.13ReportL.....	36
4.9.14ReportM.....	36
4.9.15ReportN.....	36
4.9.16ReportO.....	37
4.9.17ReportP.....	37
4.9.18ReportQ.....	37
4.9.19ReportR.....	37
4.10PAKKAUS FI.HY.HALI.VIEW.....	38
4.10.1View.....	38
4.10.2ViewLogin.....	38
4.10.3ViewNavi.....	38
4.10.4ViewPesaHaku.....	39
4.10.5ViewPesa.....	39
4.10.6ViewAputaulu.....	39
4.10.7ViewRaportit.....	39
4.10.8ViewHistoria.....	39
4.10.9ViewKunnat.....	40
4.10.10ViewMyrkyt.....	40
4.10.11ViewSaaliit.....	40
4.10.12ViewSaaliitYllapito.....	40
4.10.13ViewTarkastajat.....	40
4.10.14ViewReviirit.....	41
4.10.15ViewLajit.....	41
4.11PAKKAUS FI.HY.HALI.VALIDATE.....	41
4.11.1Validate.....	41
4.11.2ValidateLogin.....	41
4.11.3ValidateHaku.....	42
4.11.4ValidatePesatiedot.....	42
4.11.5ValidateRaportit.....	42
4.11.6ValidateHistoria.....	42
4.11.7ValidateKunnat.....	43
4.11.8ValidateMyrkyt.....	43
4.11.9ValidateSaaliit.....	43
4.11.10ValidateTarkastajat.....	43
4.11.11ValidateReviirit.....	44
4.12PAKKAUS FI.HY.HALI.LOG.....	45
4.12.1Log.....	45
4.13DATAMALLI.....	45
4.14MUUT TIEDOSTOT.....	46
4.14.1Template-tiedostot.....	46
4.14.1.1navi.ftl.....	47
4.14.1.2login.ftl.....	47
4.14.1.3haku.ftl.....	47
4.14.1.4pesatiedot.....	48
4.14.1.5myrkyt.ftl.....	57
4.14.1.6saalis.ftl.....	58
4.14.1.7saalis_yllapito.ftl.....	59
4.14.1.8historia.ftl.....	60
4.14.1.9reviirit.....	60
4.14.1.10kunnat.ftl.....	61
4.14.1.11tarkastajat.ftl.....	61
4.14.1.12aputaulut.ftl.....	62
4.14.1.13raportit.ftl.....	62
4.14.1.14lajit.ftl.....	63
4.14.2Properties-tiedostot.....	63
4.14.2.1Kielituen tiedostot.....	64
4.14.3Tyylitiedosto.....	65
5KÄYTTÖLIITTYMÄ.....	65

5.1	KÄYTTÖTAPAUKSET.....	66
5.1.1	Käyttötapaus 1: Uuden pesän lisääminen.....	66
5.1.2	Variaatio käyttötapauksesta 1.....	67
5.1.3	Käyttötapaus 2: Tiedossa olevan reviirin tarkastus.....	68
5.1.4	Käyttötapaus 3: Raportin toimittaminen.....	69
5.1.5	Käyttötapaus 4: Pesän hakeminen.....	69
5.2	KUVASARJAT.....	70
5.2.1	Käyttäjän sisäänkirjautuminen.....	70
5.2.2	Uuden pesän lisääminen.....	72
5.2.3	Aputaulun päivitys.....	75
5.2.4	Vanhan pesän tarkastus.....	77
5.2.5	Raportin toimittaminen.....	80
5.2.6	Pesän hakeminen.....	81
5.2.7	Historia-näyttö.....	83
5.2.8	Kunnat-näyttö.....	84
5.2.9	Myrkyt-näyttö.....	85
5.2.10	Saaliit-näytöt.....	86
5.2.11	Tarkastajat-näyttö.....	88
5.2.12	Reviirit-näyttö.....	88
5.2.13	Lajit-näyttö.....	89
6	TESTAUSSUUNNITELMA.....	90
6.1	KOMPONENTTITESTAUS.....	90
6.2	INTEGROINTITESTAUS.....	90
6.3	JÄRJESTELMÄTESTAUS.....	91

LÄHTEET

1 Johdanto

Tämä suunnitteludokumentti kuvaa toteutettavan Haliaeetus-järjestelmän teknisen toteutuksen näkökulmasta. Dokumentin perusteella kuka tahansa teknisesti osaava henkilö voisi toteuttaa kuvatun järjestelmän. Suunnitteludokumentti on siis ohje siitä, kuinka järjestelmä tulisi kasata. Dokumentti kuvaa järjestelmän luokkarakenteen ja käyttöliittymän.

Suunnittelun pohjana on käytetty ohjelmistotuotantoprojekti Halin [16] tuottamaa koodia ja dokumentaatiota. Järjestelmän rakennetta on kuitenkin huomattavasti muutettu myös ohjelmistotuotantoprojekti Halin [16] toteuttamilta osin.

1.1 Tuotteen tausta ja tarkoitus

WWF:n merikotkatyöryhmän johtama suojelutyö merikotkan pelastamiseksi Suomessa sukupuuton partaalta elinvoimaisiksi populaatioksi saaristossa ja Pohjois-Suomessa on eräs luonnonsuojelun menestystarinoita.

Merikotkakannan romahduksen jälkeen pesimätietoja alettiin kerätä 1960-luvulla yksittäisten tutkijoiden toimesta. WWF:n merikotkatyöryhmän perustamisen jälkeen 1972 pesimätietojen kerääminen muuttui systemaattiseksi. Kolmenkymmenen vuoden aikana on tunnetut merikotkan pesät tarkastettu vuosittain. Tarkastuksen yhteydessä on kerätty tietoja mm. pesimistuloksesta, pesinnän epäonnistumisen syistä, poikasista, pesäpuusta ja -paikasta sekä kerätty kuoriutumattomia munia ja muita näytteitä myrkkyanalyysjää varten.

Merikotkille tarkoitettuja tekopesiä on rakennettu eri puolille Suomea pesien tahattoman häirinnän minimoimiseksi pesinnän herkimmissä vaiheissa. Merikotkien pesimäpiirit ovat verraten pysyviä. Kannan kasvaessa merikotkat perustavat uusia elinpiirejä vuosittain sekä aiemmin merikotkien asuttamille alueille että uusille seuduille.

Seuranta jatkuu. Tuotteen avulla Luonnontieteellisessä keskusmuseossa syötetään ja ylläpidetään merikotkatietoja. Tiedoista kootaan raportteja tutkijoiden sekä suojelu- ja muiden viranomaisten käyttöön

Ohjelmistotuotantoprojektiryhmän tarkoituksena on suunnitella ja toteuttaa merikotkien pesätarkastusten yhteydessä kerättyjen tietojen tallettamiseen ja käyttöön soveltuva tietokanta sekä käyttöliittymä tietokantaan.

Pesätarkastusten tiedot on kerätty lomakkeilla. Talletettavat tiedot selviävät lomakkeilta ja asiakasta haastatteleamalla. Käyttöliittymä tarjoaa rajapinnan tietokannan tauluihin, tarkistaa syötetyt tiedot ja tallettaa tiedot tietokantaan. Käyttöliittymän kautta tuotetaan säännönmukaisia raportteja tutkijoiden ja viranomaisten käyttöön.

Tietokantajärjestelmää laadittaessa otetaan huomioon laajennusmahdollisuus muiden petolintujen pesätarkastusten tallettamiseen. Käyttööntovaiheessa oleva vastaava sääksijärjestelmä tarjoaa suunnittelulle lähtökohdan. Järjestelmä

asennetaan Helsingin yliopiston atk-osaston tietokonejärjestelmään tuotantokäyttöä varten.

1.2 Tärkeimmät vaatimukset

Järjestelmän on oltava turvallinen niin, että järjestelmään pääsee vain tunnuksella ja salasanaalla. Tietokannasta on pystyttävä hakemaan, lisäämään ja muuttamaan tietoa ja lisäksi on pystyttävä tuottamaan lukuisa määrä erilaisia raportteja. Käyttöliittymän tulee olla mahdollisimman selkeä ja helppokäyttöinen ja loppukäyttäjältä ei voida odottaa teknisen arkkitehtuurin tuntemusta eikä muutakaan ”teknistä” osaamista. Lisäksi järjestelmän täytyy toimia oikeellisesti myös silloin kun käyttäjä tekee virheen, esim. niin että järjestelmä ei hyväksy virheellistä syötettä.

Tarkemmin järjestelmän vaatimukset on kuvattu vaatimusdokumentissa.

1.3 Erikoissanasto ja käytetyt lyhenteet

Black-box Testausmenetelmä, jossa testaus tehdään ohjelmiston spesifikaation perusteella.

CVS Concurrent Versions System. Versionhallintaohjelmisto, joka on luotu helpottamaan ohjelmistojen versionhallintaa.

Haliaeetus järjestelmä, järjestelmä
Näillä tarkoitetaan koko toteutettavaa järjestelmää, joka sisältää käyttöliittymän, tietokannan ja näiden välillä olevat toiminnallisuudet.

HTML HyperText Markup Language. World Wide Webin eli WWW:n julkaisukieli.

HTTP Hypertext Transfer Protocol. Siirtokäytäntö eli protokolla, jonka varaan WWW rakentuu. Hypertekstidokumenttien siirtoa verkossa tukeva komentokieli.

HTTPS HTTP over Secure Sockets Layer. HTTP:n salakirjoitettu versio.

Istunto Samalta selaimelta tuleva sarja kyselyjä, jotka tapahtuvat määrättyssä ajanjaksossa.

Java Ohjelmointikieli, jota käytetään projektin toteutuksessa.

JDBC-ajuri Java DataBase Connectivity -ajuri, mahdollistaa tietokantakutsut palvelinsovelmista.

JDBC-yhteys
Hoitaa yhteyden tietokantaan, jonne järjestelmän tiedot on talletettu.

Käyttöliittymä
Se osa järjestelmästä, joka näkyy loppukäyttäjälle ja jolla järjestelmää käytetään.

Käyttötapaus

Käyttötapauksessa kuvataan käyttäjän tavoite jonkin päämäärän saavuttamiseksi, ja mahdollisimman yksityiskohtaiset tiedot tilanteen taustoista (tilatiedot). Käyttötapauksessa ei oteta mitään kantaa tekniseen toteutukseen. Käyttötapaukset ovat olennainen osa käyttöliittymäsuunnittelua.

Luokka Java-ohjelmointikielessä yksi kokonaisuus, kuten taulu tietokannassa.

Luokkarakenne

Kuvaa järjestelmän teknisen (luokkien väliset suhteet) rakenteen korkealla tasolla.

Mallipohja Sama kuin template. Dokumentin runko, joka sisältää tietyllä tavalla merkittyjä kohtia, joihin voidaan ohjelmallisesti lisätä vaihtuvia arvoja.

Merikotka Merikotka (*Haliaeetus albicilla*) on Suomen suurin petolintu. Sen siipien väli on 190–240 cm ja pituus 76–94 cm. Merikotkalla on tasaruskea höyhenpuku ja suorakaiteen muotoiset siivet, jotka harottavat kärjistään. Pyrstö on lyhyt ja kiilamainen ja vanhoilla linnuilla valkea.

Merikotkakanta

Merikotkan biologinen kanta.

Merikotkatyöryhmä

Merikotkatyöryhmä on Torsten Stjernbergin johtama merikotkien tutkimusta ja suojelua tekevä työryhmä Maailman Luonnonsäätiössä (WWF).

Metodi Java-luokan sisällä oleva aliohjelma, jota voidaan kutsua itse luokasta tai toisesta Java-luokasta.

Olio Java-luokan ilmentymä.

Pakkaus Pakkaus on tapa kerätä yhteen toisiinsa jollain tavalla liittyvät ohjelmaluokat.

Pesä Merikotkan pesä on hyvin kookas risupesä suuren puun latvassa tai kallionkielekkeellä. Pesä voi olla myös vaihtopesä tai tekopesä.

Reviiri Yhdellä reviirillä elää yksi merikotkapariskunta, jolla saattaa olla useampikin pesä reviirinsä alueella.

Servlet Java-ohjelmointikielellä kirjoitettu palvelinsovelma, servletti.

SQL Structured Query Language. Standardoitu kieli, jolla voidaan määrittää erilaisia tietokantaoperaatioita.

- Tarkastus** Pesille tehdään tarkastuskäyntejä muutaman kerran vuodessa, jolloin saadaan tietoa mm. poikasten määrästä ja pesällä vallitsevista olosuhteista.
- Template** Sama kuin mallipohja. Dokumentin runko, joka sisältää tietyllä tavalla merkittyjä kohtia, joihin voidaan ohjelmallisesti lisätä vaihtuvia arvoja.
- Tietokanta** Jotain käyttötarkoitusta varten laadittu kokoelma toisiinsa liittyviä säilytettäviä tietoja. Tietokannan teknisiä ominaisuuksia ovat mm. tiedon riippumattomuus sitä käsittelevistä ohjelmista, tietojen samanaikainen käyttö, monipuoliset tiedonhakumahdollisuudet, tietojen suojaus, mutkikkaat riippuvuudet tietojen välillä ja automaattinen varmistus ja elpyminen häiriöistä.
- Tilatieto** Käyttötapauksissa tilatiedot kertovat kaikki taustatekijät, jotka vaikuttavat käyttäjän tavoitteen saavuttamiseen.
- TKTL** Helsingin yliopisto, Tietojenkäsittelytieteen laitos.
- White-box** Rakenteellinen testausmenetelmä, eli testaus tehdään ohjelmiston sisäisen rakenteen perusteella.
- WWF** WWF (World Wildlife Fund) eli Maailman Luonnonsäätiö on maailmanlaajuinen luonnonsuojelujärjestö, joka työskentelee luonnon monimuotoisuuden suojelemiseksi ja ekologisten toimintojen ylläpitämiseksi.

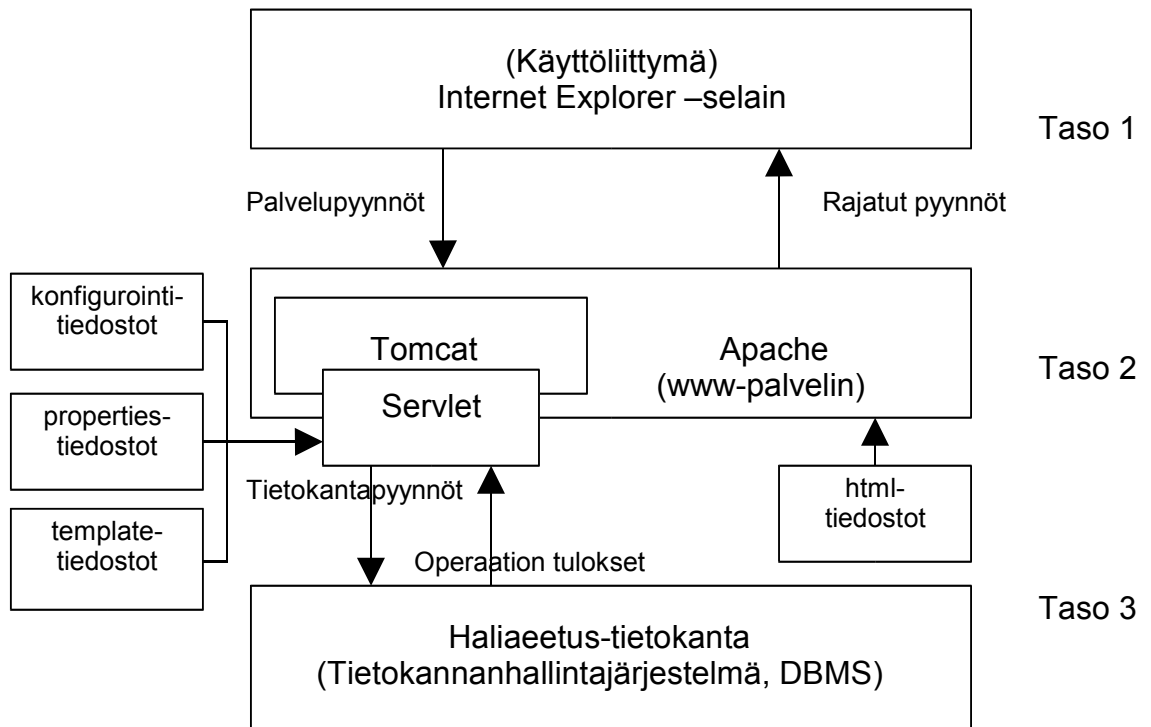
2 Suunnittelun ja toteutuksen rajoitteet

Tässä luvussa määritellään rajoitukset suunnittelulle ja toteutukselle. Rajoitukset koskevat teknistä toimintaympäristöä ja eräitä standardeja. Lisäksi on huomattava asiakkaan vaatimus siitä, ettei mitään merikotkiin liittyvää kriittistä suojelutietoa vuoda ulkopuolisille esim. dokumenttien kautta.

2.1 Noudatettavat standardit ja tarvittavat ohjelmat

Käyttäjän selaimen ja WWW-palvelimen välinen kommunikointi tapahtuu suojatulla HTTPS-protokollalla 6.3. Käyttäjälle näkyvissä HTML-sivuissa käytetään W3C:n 6.3 määrittelemää HTML 4.01-spesifikaatiota 6.3.

Loppukäyttäjän työasemasta on oltava tietoliikenneyhteys palvelimelle, jossa Java-servletit ajetaan. Käyttöliittymä toimii IE 6.0 (tai uudempi) selaimella 6.3. Lisäksi työasemassa on oltava Java- ja Oracle 9i -ympäristöt. Palvelimen tietokantana on kyseinen Oracle 9i 6.3.



Kuva 1: Yleiskuva järjestelmän Haliaetus toiminnasta

Templatet on toteutettu FreeMarker-kirjastolla 6.3, joka on julkaistu GNU General Public License:n 6.3 alla.

Oracle 9i:n JDBC-ajurista 6.3 käytetään toteutusvaiheessa versiota 9.x for Java 1.4.x.

2.3 Ohjelmointikieli ja -tyyli

Ohjelmisto kirjoitetaan Sunin kehittämällä Java-ohjelmointikielellä 6.3. Koodin ulkoasussa pyritään noudattamaan Java Code Conventions –spesifikaatiota 6.3. Koodi kommentoidaan niin, siitä voidaan generoida Javadoc-dokumentaatio 6.3.

Dokumentointi- ja kommentointikieli on suomi. Java-luokkien nimet ovat englanniksi, kuten myös metodien ja muuttujien nimet, lukuun ottamatta muuttujia, jotka vastaavat tietokannan attribuutteja. Tietokannan attribuutteja vastaavat muuttujat on nimetty suomeksi ja vastaavilla nimillä kuin tietokannassa, jotta vältetään sekaannuksilta.

Versionhallintaan käytetään TKTL:n CVS-versionhallintajärjestelmää 6.3.

3 Järjestelmän yleiskuvaus

Tässä luvussa kuvataan yleisellä tasolla Haliaeetus-järjestelmän toimintaa. Käyttöliittymä on kuvattu tämän dokumentin luvussa 5 ja tietokanta on kuvattu projektin vaatimusdokumentissa.

Järjestelmä perustuu käyttäjän selaimen, servlet-palvelimen ja tietokantapalvelimen väliseen vuorovaikutukseen (**Kuva 1**). Yhteys käyttöliittymän ja tietokannan välillä toteutetaan kolmitasoisella arkkitehtuurilla.

Taso 1:

Tasolla 1 oleva selain on vuorovaikutuksessa sekä käyttäjän että tasolla 2 olevan ohjelmiston kanssa.

Taso 2:

Palvelinkoneella pyörii Apache WWW-palvelin, sekä *Tomcat 4.1.10*-palvelin 6.3. Palvelimella sijaitsevat JDBC-ajuri, servletit, JDBC-tietokantaluokat, Freemarker 6.3 kirjastoluokat sekä staattiset template-, html- ja properties-tiedostot.

Apache-Tomcat – pari käyttää HTTPS-protokollaa 6.3 kommunikoidessaan käyttäjän selaimen kanssa ja JDBC-ajuria 6.3 kommunikoidessaan Haliaeetus-tietokannan kanssa.

Servletit 6.3 käyttävät tietokantaa JDBC-tietokantaluokkien avustuksella. Servletit myös generoivat dynaamista HTML:ää ja kommunikoivat käyttäjän selaimen kanssa. Servletit saavat tulostamiensa HTML-sivujen rungot template-tiedostoista, ja esitettävät tiedot Haliaeetus-tietokannasta.

Taso 3

Tasolla 3 on Haliaeetus-tietokanta, jossa täytyy olla vastaava JDBC-ajuri, joka tukee JDBC API:a.

4 Arkkitehtuurikuvaus

Tässä luvussa kuvataan ohjelmiston käyttämät tiedostot, niiden keskinäiset suhteet ja sijainti järjestelmässä.

Ohjelmisto koostuu kymmenestä pakkauksesta:

<code>fi.hy.hali.db</code>	Tietokantaoperaatiot
<code>fi.hy.hali.general</code>	Tiedostotalleminen ja properties-tiedostojen käsittely
<code>fi.hy.hali.servlet</code>	Kontrolliservletti Hali
<code>fi.hy.hali.action</code>	Abstrakti Action-luokka
<code>fi.hy.hali.action.search</code>	Hakuoperaatiot
<code>fi.hy.hali.action.update</code>	Päivitysoperaatiot
<code>fi.hy.hali.action.report</code>	Raporttiluokat
<code>fi.hy.hali.validate</code>	Tarkistusluokat
<code>fi.hy.hali.log</code>	Loki
<code>fi.hy.hali.view</code>	Näyttöluokat

Lisäksi html-sivujen tuottamiseen käytetään mallipohjia eli template-tiedostoja ja tyylitiedostoa. Kielituessa käytetään properties-tiedostoja.

Pakkauksen `fi.hy.hali.general` luokka `HaliProperties` ja tietokantaluokat on otettu tipu4:stä 6.3. Luokan `HaliProperties` pohjana on tipu4:n `TipuProperties`, josta ryhmä Hali [16] on poistanut joitain metodeja. Myös tietokantaluokkia ryhmä Hali [16] on hieman muuttanut. Ainoastaan luokat `DatabaseOperation` ja `DeleteOperation` ovat muuttamattomina tipu4:stä.

4.1 Komponenttien väliset suhteet

Käyttäjän syöttämä informaatio lähetetään selaimen toimintopainikkeista pakkauksen `fi.hy.hali.servlet` `Hali`-luokalle, joka toimii kontrolliservlettinä. `Hali`-servletti kutsuu tarvittavia `fi.hy.hali.action` ja `fi.hy.hali.view` pakettien luokkia, jotka toteuttavat kutsun mukaisen toiminnallisuuden.

Action luokat `fi.hy.hali.action` paketissa tarkistavat hakuehdot ja kantaan syötettävät tiedot `fi.hy.hali.validate` pakkauksen luokkien avulla ja suorittavat tietokantaoperaatiot pakkauksen `fi.hy.hali.db` tietokantaluokkien avulla. Suoritettava tietokantaoperaatio voi olla haku-, lisäys-, muokkaus- tai poisto- operaatio. Tietokantaoperaatiota varten tarvitsemansa metadatan (Luku 4.3.2) Action-luokat saavat luomalla ilmentymän `Table`-luokasta, joka käyttää apunaan tiedostoa `tabledata`.

Action-luokat palauttavat tietokantaluokilta saamansa vastauksen `Hali`-servletille, joka välittää sen `View`-luokalle uuden näytön generoimista varten. Näytön luonnissa `View`-luokka käyttää apuna valmiita mallipohjia sivun ulkoasun

määrittelemisessä ja tietojen oikeaan kenttään kohdentamisessa. Kielituki toteutetaan View-luokassa, joka hakee tarvitsemansa erikieliset tekstit properties-tiedostoista.

Konfigurointitiedot laitetaan tiedostoon `haliaaetus.config`, josta niitä voidaan lukea.

4.2 Luokkien väliset suhteet

4.2.1 Luokkakaavio

Luokkakaaviossa (**kuva 2**) on esitetty järjestelmän luokkien suhde toisiinsa.

Luokkakaavion luokat pakkauksittain:

fi.hy.hali.servlet

Hali

fi.hy.hali.action

Action (abstrakti)

LoginValidation

fi.hy.hali.action.search

Search (abstrakti), PesaSearch, KuntaSearch, AputauluSearch, ReviiriSearch, TarkastajaSearch, HistoriapesaSearch, LajiSearch, SaalisSearch, MyrkkySearch

fi.hy.hali.action.update

Update (abstrakti), PesaUpdate, KuntaUpdate, AputauluUpdate, ReviiriUpdate, TarkastajaUpdate, HistoriapesaUpdate, LajiUpdate, SaalisUpdate, MyrkkyUpdate

fi.hy.hali.action.report

Report (abstrakti), ReportA-ReportR

fi.hy.hali.validate

Validate (abstrakti), ValidateLogin, ValidateKunta, ValidatePesatiedot, ValidateLaji, ValidateAputaulu, ValidateTarkastaja, ValidateHistoria, ValidateReviiri, ValidateHaku, ValidateRaportit, ValidateSaalis, ValidateMyrkky

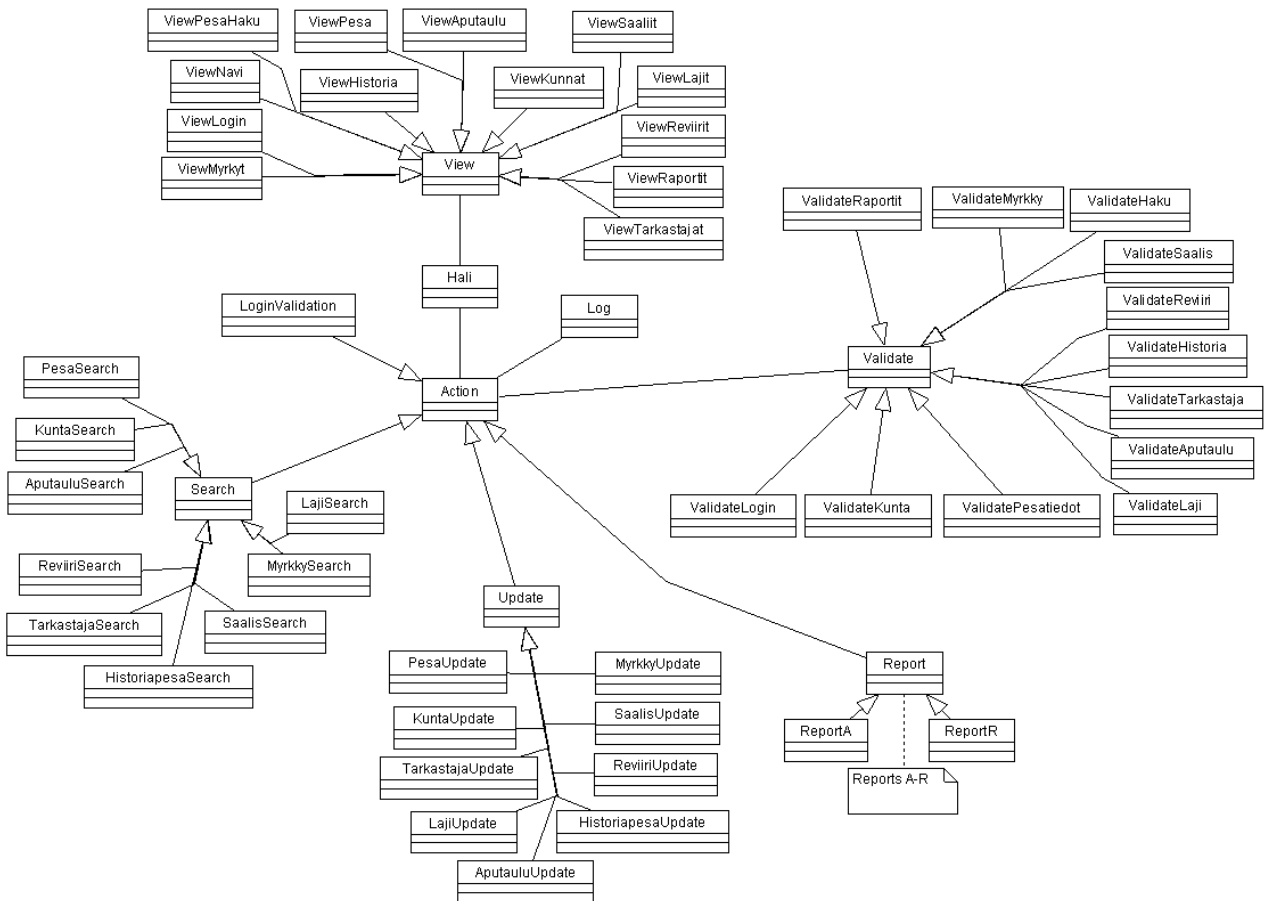
fi.hy.hali.log

Log

fi.hy.hali.view

View (abstrakti), ViewLogin, ViewNavi, ViewPesaHaku, ViewPesa, ViewHistoria, ViewAputaulu, ViewKunnat, ViewLajit, ViewReviirit, ViewRaportit, ViewTarkastajat, ViewSaaliit, ViewMyrkyt

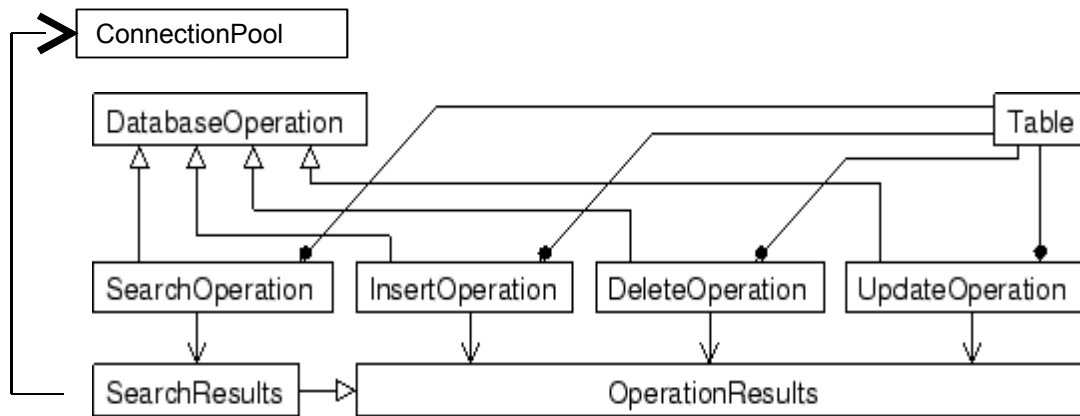
Kuvassa ei ole pakkauksia `fi.hy.hali.db` ja `fi.hy.haligeneral`. Tarkempi esitys näiden pakkauksien sisältämistä luokista on kyseistä pakkausta käsittelevässä luvussa.



Kuva 2: Luokkien väliset suhteet

4.3 Pakkaus `fi.hy.hali.db`

Pakkaus `fi.hy.hali.db` pitää sisällään tietokantaluokat (**kuva 3**), jotka suorittavat JDBC-yhteyden avulla haku, lisäys, muokkaus tai poisto – operaatioita tietokantaan. Operaatiot on toteutettu luokissa `SearchOperation`, `InsertOperation`, `UpdateOperation` ja `DeleteOperation`, jotka periytyvät yhteisestä abstraktista ylliluokasta `DatabaseOperation`.



Kuva 3: Tietokantaoperaatioiden luokkarakene ja tulostyytit.

Operaatiot palauttavat kutsujalleen lisäys-, muokkaus- ja poistotapauksissa totuusarvon, joka indikoi halutun operaation onnistumista. Haku-operaatio palauttaa erillisen `SearchResults`-olion, jolla hakutuloksia voidaan tarkastella lähemmin.

Myös JDBC-yhteys tietokantaan hoidetaan tietokantaluokkien avulla. Yhteyksiä jaetaan yhteysvarannon (connection pool) avulla. Luokka `DatabaseOperation` tarjoaa tuen luokan `ConnectionPool` dynaamiselle yhteysjaolle.

4.3.1 ConnectionPool

`ConnectionPool` eli yhteysvaranto sisältää JDBC-tietokantayhteyksiä. Yhtäaikaisten yhteyksien maksimilukumäärä sekä käytettävä JDBC-ajuri ja tietokannan url ilmoitetaan tiedostossa `haliaaetus.config`.

Konstruktori:

```
private ConnectionPool(String user, String pw)
```

Konstruktori alustaa ja luo `ConnectionPool`-olion. Tarvittavia tietoja ovat muun muassa käytettävä ajuri, tietokannan url ja yhtäaikaisten yhteyksien maksimilukumäärä. Parametreina annettavat käyttäjätunnus ja salasana sijoitetaan luokan muuttujiin, joten kaikki yhteydet tietokantaan luodaan pelkästään näitä tunnuksia käyttäen. Koska servletit saavat käyttää vain luokan staattisia metodeja, konstruktori määrittellään näkyvyydeltään yksityiseksi.

Julkiset metodit:

```
public static Connection requestConnection
    (String username, String password)
```

Metodilla pyydetään tietokantaoperaatioon tarvittava yhteys. Metodia on tarkoitus käyttää siten, että ennen tietokantaoperaatioita servlet pyytää luokalta käyttäjälle yhteyden.

Käyttäjän käyttäjätunnus ja salasana tarkistetaan joka kerta yhteyttä pyydettyäessä. Jos tunnukset eivät ole kelvolliset, palautetaan `null`. Parametreja `username` (käyttäjätunnus) ja `password` (salasana) verrataan varannon tiedossa olevaan tietokannan käyttäjätunnukseen ja salasanaan. Kaikkien käyttäjien tulee käyttää samaa tietokantatunnusta kirjautuessaan Haliaetus-järjestelmään, koska suunnitellussa varannossa on tuki vain yksille tunnuksille.

Suoritettuaan kaikki saman pyynnön aikana tehtävät tietokantaoperaatiot, servletin tulisi palauttaa yhteys varantoon. Yhteys pysyy avoimena, vaikka se luovutetaankin muiden operaatioiden käyttöön.

```
public static void releaseConnection(Connection con)
```

Metodi palauttaa aiemmin varatun yhteyden takaisin yhteysvarantoon, jossa se on kaikkien käyttäjien käytettävissä.

```
public void run()
```

Säikeen tarvitsema metodi.

Yksityiset metodit:

```
private synchronized Connection getConnection()
```

Metodi hankkii yhteyden jollakin seuraavista keinoista:

1. Vapaita yhteyksiä löytyy varannosta: otetaan ensimmäinen vapaa yhteys sieltä käyttöön. Jos saatu yhteys oli ajurin puolesta suljettu, luodaan tilalle uusi yhteys ja yhteyden varausta yritetään uudelleen. Tässä tapauksessa yhteyden pyytäjä joutuu kilpailemaan muiden yhteyttä pyytämään tulleiden käyttäjien kanssa tasavertaisesti.

2. Vapaita yhteyksiä ei ole varannossa: Jos yhteyksien maksimimäärää ei ole vielä ylitetty, luodaan uusi yhteys ja kilpailutetaan se käyttäjien kesken. Jos yhteyksiä on jo maksimimäärä, jäädään odottamaan yhteyden vapautumista.

```
private void makeBackgroundConnection()
```

Metodi käynnistää säikeen muodostamaan yhteyttä silloin, kun yhteyksiä ei ole vapaana. Yhteyden muodostus tapahtuu säikeessä, koska yhteyden luominen voi kestää useamman sekunnin. Kun toiminto suoritetaan taustalla, voidaan palvella muita yhteyden pyytäjiä ja vapauttajia rinnalla.

```
private Connection makeNewConnection()
```

Metodi luo uuden yhteyden tietokantaan. Jos yhteyden luonti ei onnistunut, palautetaan `null`.

```
private synchronized void free(Connection connection)
```

Metodi palauttaa yhteysvarantoon kuuluvan yhteyden. Yhteys siirretään vapaiden yhteyksien joukkoon ja yhteyttä mahdollisesti odottaville ilmoitetaan

vapautuneesta yhteydestä.

```
private synchronized int totalConnections()
```

Metodi laskee vapaiden ja käytettyjen yhteyksien lukumäärän.

```
private boolean checkUser(String user, String pw)
```

Metodi tarkastaa, että käyttäjän tunnus ja salasana vastaavat yhteysvarannon tiedossa olevaa käyttäjätunnusta ja salasanaa.

```
private synchronized void closeAllConnections()
```

Metodilla suljetaan kaikki yhteydet, niin vapaat kuin käytössä olevatkin. Metodia käytetään vain `ConnectionPool` -olion tuhoamisen yhteydessä.

```
private void closeConnections(Set connections)
```

Metodi sulkee kaikki annetussa yhteysjoukossa (vapaat/käytössä) olevat yhteydet. Metodia käyttää vain `closeAllConnections()`.

Pakkausnäkyvyyden metodit:

```
protected void finalize()
```

Lopetusmetodi kutsuu `closeAllConnections()`-metodia. Tällä tavoin varmistetaan, että `ConnectionPool` -olion tuhoamisen yhteydessä kaikki yhteydet tulevat suljetuiksi.

4.3.2 Table

`Table`-luokan avulla saadaan tietokantataulujen metadata eli nimet, attribuutit ja jokaiseen attribuuttiin liittyvät tiedot tekstitiedostosta `tabledata`. Tiedosto `tabledata` sijoitetaan toteutusympäristössä palvelinkoneelle `alkokrunni.cs.helsinki.fi` kansioon `/jserv/etc/`. Tiedosto sisältää taulujen tiedot siten, että jokainen tiedoston rivi sisältää yhden taulun tiedot muodossa

```
taulu;attribuutti:avain:null:type:kommentti;attribuutti:avain:...
```

missä

- `taulu` on taulun nimi,
- `attribuutti` on taulun sarakkeen eli attribuutin nimi,
- `avain` 'Y' jos attribuutti on taulun pääavain, muutoin 'N',
- `null` 'Y' jos attribuutti voi olla null, muutoin 'N',
- `type` on attribuutin tyyppi (`NUMBER`, `VARCHAR2`, `DATE`..),
- `kommentti` on attribuuttia koskeva kommentti.

Taulun nimen ja eri attribuuttien erottimena on puolipiste ja attribuuttien tietokenttien erottimena kaksoispiste.

Servlet hakee käynnistyessään jokaisen kohdetaulunsa `Table`-olion, ja säilyttää niitä ilmentymämuuttujissaan koko elinkaarensa ajan.

Sisäluokka:

```
private static class FieldInfo
```

`FieldInfo` on `Table`-luokan staattinen sisäluokka, joka toimii yksinkertaisena taulun attribuuttia kuvaavana tietorakenteena.

Sisäluokan muuttujat:

```
int type
```

Muuttujan arvona on sarakkeen tyyppi.

```
String comment
```

Muuttujaan sijoitetaan sarakkeen kommentti.

```
boolean key
```

Muuttujan arvona on `true`, mikäli sarake on taulun pääavain.

```
boolean nullable
```

Muuttujan arvona on `true`, mikäli arvo voi olla `NULL`.

Muuttujat:

```
private String tableName
```

Muuttujassa on taulun nimi.

```
private SortedSet attributes
```

Muuttujassa on taulun attribuuttien nimet (`String`) aakkosjärjestettynä joukkona.

```
private Map fields
```

Muuttujassa on taulun attribuuttien tiedot attribuuttien nimien (`String`) osoittamina `Table.FieldInfo`-olioina.

Konstruktori:

```
private Table(String tableName, SortedSet attributes,
              Map fields)
```

Konstruktori luo `Table`-olion sijoittaen parametrit ilmentymämuuttujiinsa. Parametri `tableName` ilmoittaa tietokannan taulun nimen, `attributes` sisältää tietokannan attribuutit järjestyksessä ja `fields` kohdentaa jokaiseen attribuuttiin kyseisen attribuutin muut tiedot (avain, null, type, ja kommentti). Koska konstruktoria saa käyttää vain luokan staattinen metodi, konstruktori on näkyvyydeltään yksityinen.

Julkiset metodit:

```
public static Table getTable(String tableName)
```

Metodi palauttaa parametria vastaavan taulun tiedot. Metodin ensimmäisellä kutsukerralla ennen taulutietojen hakua haetaan `tabledata`-tiedostosta taulukuvaukset.

```
public String getName()
```

Metodi palauttaa taulun nimen.

```
public SortedSet getKeys()
```

Metodi palauttaa taulun attribuuteista ne, jotka ovat taulun pääavaimia. Avain tai avaimet palautetaan järjestettynä joukkona merkkijonoja (`SortedSet`). Merkkijonoilla järjestys on yleensä nouseva.

```
public SortedSet getAttributes()
```

Metodi palauttaa taulun attribuuttien nimet järjestettynä joukkona merkkijonoja. Merkkijonoilla järjestys on yleensä nouseva.

```
public int getType(String attr)
```

Metodilla voi selvittää parametrina annetun attribuutin tyyppin (`NUMBER`, `VARCHAR2`, `DATE`...).

```
public String getComment(String attr)
```

Metodi palauttaa parametrina annettuun attribuuttiin liittyvän kommentin.

```
public boolean isKey(String attr)
```

Metodilla voidaan selvittää onko parametrina annettu attribuutti taulun avain.

```
public boolean isNullable(String attr)
```

Metodi palauttaa tiedon siitä, voiko parametrina annettu attribuutti olla null.

Yksityiset metodit:

```
private static FieldInfo parseField(StringTokenizer fields)
```

Metodi `loadTabledata()` käyttää tätä metodia apunaan. Metodilla saadaan koottua attribuutin muut tiedot (avain, null, type, ja kommentti) `FieldInfo`-sisäluokan ilmentymään.

```
private static Map loadTableData()
```

Metodi lukee tiedostosta `tabledata` kaikkien tietokannassa käytettyjen

taulujen kuvaukset. Tiedosto luetaan rivi riviltä, ja jokaisesta rivistä tehdään Taulu-olio.

4.3.3 OperationResults

`OperationResults`-luokkaa käytetään tietokantaoperaatioiden tulosten tutkimiseen. Tietokantaoperaatio palauttaa `OperationResults`-tyyppisen olion, jossa on operaation totuusarvoinen tulos ja mahdollinen virheilmoitus. Hakutuloksille luodaan oma tarkempi tulosolio, joka on tyyppiä `SearchResults`. Luokka `SearchResults` perii luokan `OperationResults` ominaisuudet.

Konstruktori:

```
public OperationResults()
```

Metodi luo tulosolion, jossa virhettä ei ole tapahtunut, eli tulosoliota luotaessa oletetaan tietokantaoperaation onnistuvan.

Julkiset metodit:

```
public void setErrorMessage(String msg)
```

Jos tietokantaoperaatio ei jostain syystä onnistunut, tietokantaluokka asettaa tällä metodilla virheilmoituksen tulosolioon, ja operaatio merkitään epäonnistuneeksi.

```
public boolean succeeded()
```

Servlet voi tämän metodin avulla selvittää tietokantaoperaation onnistumisen / epäonnistumisen. Jos metodi palauttaa `false` merkinä operaation epäonnistumisesta, `getErrorMessage()` -metodia voidaan käyttää virheen tarkempaan tutkimiseen.

```
public String getErrorMessage()
```

Tällä metodilla saa kysytyä operaation epäonnistumisen syyn. Palautettu teksti on tarkoitettu ainoastaan virheen raportointiin.

4.3.4 SearchResults

`SearchResults` on luokka, jolla hakuoperaation tuloksia voi selata. Luokka periytyy `OperationResults`-luokasta. `SearchResults`-luokka sisältää listan haussa löytyneiden kohdetaulujen riveistä. Servlet pääsee käsiksi hakutuloksiin luokan julkisilla metodeilla. Osa metodeista vaatii, että käyttäjän tarvitsee antaa parametreina myös tietokannan käyttäjätunnus ja salasana (koska nämä metodit puolestaan käyttävät luokkaa `ConnectionPool`).

Konstruktori:

```
public SearchResults(ResultSet rs, Table t)
```

Konstruktorin tehtävänä on kutsua ylluokan konstruktoria ja asettaa tietokantaoperaation tuottaman tulosjoukon ROWID:t listaan ja operaatioissa ollut kohdetaulu SearchResults-olion tiedoiksi.

Julkiset metodit:

```
public int getRowCount()
```

Metodi palauttaa tietokantahaussa löytyneiden rivien määrän.

```
public int getRowNumber()
```

Metodi palauttaa sen rivin numeron, missä kursori tällä hetkellä on. Palautettu arvo on väliltä [1..n], missä n on tulosjoukon rivien lukumäärä.

```
public SortedMap getCurrentRow(String username,
                               String password)
```

Metodi palauttaa tämänhetkisen vastausjoukon rivin järjestettynä SortedMap-oliona (attribuutti -> arvo).

```
public SortedMap getNextRow(String username,
                             String password)
```

Metodi palauttaa vastausjoukossa nykyistä kohtaa seuraavan rivin, ja kursoria siirretään yksi rivi eteenpäin. Jos kursori on jo viimeisellä rivillä, palautetaan null.

```
public SortedMap getPreviousRow(String username,
                                 String password)
```

Metodi palauttaa vastausjoukossa nykyistä kohtaa edeltävän rivin, ja kursoria siirretään yksi rivi taaksepäin. Jos kursori on jo ensimmäisellä rivillä, palautetaan null.

```
public SortedMap getAbsoluteRow(int rowNum,
                                 String username, String password)
```

Metodi palauttaa vastausjoukosta parametrilla rowNum määritellyn rivin. Haluttu rivi annetaan väliltä [1..n], missä n on tulosjoukon rivien lukumäärä. (Hausssa löytyneiden rivien määrän saa selville luokan metodilla getRowCount()).

```
public SortedMap getAbsoluteRow(String rowId,
                                 String username, String password)
```

Metodi palauttaa vastausjoukosta parametrilla rowId määritellyn rivin.

```
public void removeRow(ROWID rowId)
```

Metodi poistaa (tietokannasta juuri poistetun) rivin myös SearchResults-olion hakutulostista. Jos riviä ei ole, ei tehdä mitään. Metodia kutsutaan servletistä käsin, jotta juuri poistettu tietue ei näkyisi vastausjoukossa.

```
public void removeRow(String rowId)
```

Metodi poistaa (tietokannasta juuri poistetun) rivin myös `SearchResults`-olion hakutulostilasta. Metodi kutsuu `removeRow(ROWID rowId)`:tä. Metodia kutsutaan servletistä käsin, jotta juuri poistettu tietue ei näkyisi vastausjoukossa.

Yksityiset metodit:

```
private void setResultSet(ResultSet rs)
```

Metodi sijoittaa haussa löydettyjen rivien ROWID:t listaan. Tämän jälkeen vastauskursori suljetaan.

```
private SortedMap getAbsoluteRow(ROWID rowId,
                                  String username, String password)
```

Metodi palauttaa käyttäjän määrittelemän rivin järjestettynä `SortedMap`-oliona. Luokan julkiset rivihaakumetodit käyttävät tätä metodia riviä hakiessaan.

4.3.5 DatabaseOperation

Luokka `DatabaseOperation` on abstrakti yleistys kaikista mahdollisista servlettien tietokantaan kohdistamista operaatioista. Jokainen tietokantaoperaatio on oltava valmisteltavissa siten, ettei joka kerta kyseistä operaatiota tiettyyn tauluun kohdistettaessa tarvitse kyselylauseketta kääntää uudestaan. Siksi jokaisen tietokantaoperaation on toteutettava tietty tietokantaoperaatioiden hallinnalle yhteinen rajapinta.

Jokaisella operaatiolla (poislukien `SearchOperation`) oletetaan olevan yksi tiettyyn tauluun kohdistuva valmisteltu lauseke (`java.sql.PreparedStatement`), jollaisen ilmentymää `DatabaseOperation`-luokka säilyttää jokaista ko. operaation suoritukseen tarjottua tietokantayhteyttä kohti. Näin täytyy tehdä, sillä valmistellut lausekkeet ovat sidottuja siihen yhteyteen, jolla valmistelu on alun perin suoritettu.

Ainoa asia, mitä geneerinen tietokantaoperaatio ei voi määrittellä, on SQL-lause, joka määrää mitä kyseinen operaatio käytännössä tekee. Siksi jokaisen erikoistuneen operaatioluokan on toteutettava luokan `DatabaseOperation` abstrakti metodi `getSQLString()`.

Muuttujat:

```
protected OperationResults result
```

Operaation tulos talletetaan muuttujaan `result`. Toteuttava tietokantaoperaatio voi missä tahansa vaiheessa tämän muuttujan avulla asettaa operaation tilan epäonnistuneeksi, jolloin se raportoidaan käyttäjälle.

```
protected Table table
```

Muuttujaan `table` laitetaan operaation kohdetaulun ilmentymä.

```
private Map statementMap
```

Muuttuja `statementMap` sisältää jokaiselle käytössä olevalle tietokantayhteydelle (`java.sql.Connection`) käännetyt SQL-lauseet (`java.sql.PreparedStatement`).

Konstruktori:

```
protected DatabaseOperation(Table table)
```

Konstruktori luo generisen tietokantaoperaation, joka kohdistuu parametrin `table` määrittelemään tietokantatauluun.

Pakkausnäkyvyyden metodit:

```
protected final PreparedStatement
    getPreparedStatement(Connection con)
```

Metodi palauttaa annetulle yhteydelle valmistellun tätä operaatiota vastaavan lausekkeen. Mikäli valmistelua ei juuri tälle yhteydelle vielä ole tehty, se tehdään metodia kutsuttaessa ja talletetaan tämän operaation ilmentymään käytettäväksi juuri tälle yhteydelle tämän operaation tulevia suorituksia varten.

```
protected int getResultSetType()
```

Metodi palauttaa operaation vaatiman/salliman vastausjoukon kursorityypin, jonka tulee olla yksi luokan `java.sql.ResultSet` arvoista `TYPE_FORWARD_ONLY`, `TYPE_SCROLL_INSENSITIVE` tai `TYPE_SCROLL_SENSITIVE`. Oletuksena palautetaan arvo `TYPE_FORWARD_ONLY`.

```
protected int getResultSetConcurrency()
```

Metodi palauttaa operaation vaatiman/salliman tietueiden eristyneisyysasteen, jonka tulee olla toinen `java.sql.ResultSet` luokan arvoista `CONCUR_READ_ONLY` tai `CONCUR_UPDATABLE`. Oletuksena palautetaan arvo `CONCUR_READ_ONLY`.

Abstraktit metodit:

```
protected abstract String getSQLString()
```

Metodi palauttaa toteuttavan operaation SQL-kielisen lausekkeen, johon on ilmentymää luotaessa luotu valmis parametrilista annetun kohdetaulun avainjoukon avulla.

4.3.6 SearchOperation

Luokka `SearchOperation` perii luokan `DatabaseOperation`. `SearchOperation` suorittaa `SELECT`-lauseella haun tietokannan tauluun annetuilla attribuuttimaskkeilla ja muilla hakuehdoilla. Attribuuttimaskivertailut tehdään `LIKE`-operaattoreilla, joten käyttäjät voivat itse syöttää jokerimerkkejä

hakukenttiin. Muissa hakuehdoissa voi käyttää mitä tahansa operaattoreita. Tulokset palautetaan erillisessä `SearchResults`-oliossa.

Keskustelu hakuoperaation kanssa on hyvin suoraviivaista. Ensin operaatio konstruoidaan antamalla parametriksi kohdetaulu. Tämän jälkeen koostetaan hakuarvot kartaksi ja muodostetaan mahdollinen muut hakuehdot määrittelevä merkkijono ja kutsutaan niillä metodia `executeSearch()`, joka suorittaa itse operaation. Tuloksena käyttäjä saa erillisen tulosolion `SearchResults`.

Konstruktori:

```
public SearchOperation(Table table)
```

Konstruktori luo uuden hakuoperaation, joka kohdistuu parametrin `table` määrittelemään tietokantatauluun.

Julkiset metodit:

```
public OperationResults executeSearch(Connection con,
                                     Map values,
                                     String where)
```

Metodi suorittaa hakuoperaation kohdetauluun annetulla yhteydellä `con` käyttäen hakuehtoina hakuarvoja `values` ja muita hakuehtoja `where`. Muut hakuehdot määrittelevä merkkijono `where` voi sisältää mitä tahansa SQL:n WHERE-lauseessa sallittuja hakuehtoja. Kutsu palauttaa haun tuloksena `SearchResults`-olion.

```
public OperationResults executeSearch(Connection con,
                                     Map values,
                                     String where,
                                     String sortKey)
```

Metodi suorittaa hakuoperaation kohdetauluun annetulla yhteydellä `con` käyttäen hakuehtoina hakuarvoja `values` ja muita hakuehtoja `where`. Muut hakuehdot määrittelevä merkkijono `where` voi sisältää mitä tahansa SQL:n WHERE-lauseessa sallittuja hakuehtoja. Tulokset järjestetään kohdetaulun sarakkeen `sortKey` mukaan. Kutsu palauttaa haun tuloksena `SearchResults`-olion.

Yksityiset metodit:

```
private String escapeQuotes(String s)
```

Metodi palauttaa parametrinaan saamansa merkkijonon SQL-lauseeseen sopivassa muodossa, jossa mahdolliset heittomerkit on kahdennettu, jotta niitä ei tulkittaisi merkkijonoskalaarin erottimiksi.

Pakkausnäkyvyyden metodit:

```
protected String getSQLString(Map values, String where,
                              String sortKey)
```

Metodi palauttaa hakulausekkeen muotoa

```
SELECT ROWID FROM taulu
WHERE attr1 LIKE 'value1'
AND attr2 LIKE 'value2'
...
AND attrN LIKE 'valueN'
AND (where)
ORDER BY sortKey
```

Kentät *attr?* ovat kohdetaulun attribuutteja (parametrin *values* avaimia). Kentät *value?* ovat hakuarvoja (parametrin *values* annettua avainta vastaavia arvoja). Jos hakuarvo on `null`, attribuuttia ja arvoa ei lisätä kyselyyn. Merkkijono *where* on muut hakuehdot määrittelevä merkkijono, joka voi sisältää mitä tahansa SQL:n WHERE-lauseessa sallittuja hakuehtoja, ja se lisätään kyselyyn vain jos se ei ole `null`. Lause `ORDER BY` lisätään vain, mikäli *sortKey* ei ole `null`.

4.3.7 InsertOperation

Luokka `InsertOperation` perii luokan `DatabaseOperation`. Lisäysoperaatiolla voidaan lisätä yksi rivi määrättyyn tietokannan tauluun. Lisäys onnistuu, mikäli taulussa ei vielä ole samoilla avaimilla varustettua tietuetta, taulun eheys- ja avainrajoitteet toteutuvat eikä tiedonsiirtovirheitä tapahdu. Tulokset palautetaan `OperationResults`-oliona.

Keskustelu lisäysoperaation kanssa on hyvin suoraviivaista. Ensin operaatio konstruoidaan antamalla parametriksi kohdetaulu. Tämän jälkeen koostetaan arvot kartaksi ja kutsutaan niillä metodia `executeInsert()`, joka suorittaa itse operaation. Kutsu palauttaa lisäysoperaation tuloksena `OperationResults`-olion.

Konstruktori:

```
public InsertOperation(Table table)
```

Konstruktori luo uuden lisäysoperaation, joka kohdistuu parametrin *table* määrittelemään tietokantatauluun.

Julkiset metodit:

```
public OperationResults executeInsert(Connection con, Map
values)
```

Metodi lisää rivin kohdetauluun arvoilla *values*. Kyseisestä arvokartasta tulee löytyä arvot ainakin kaikille pääavaimille sekä muille 'NOT NULL'-attribuuteille.

Yksityiset metodit:


```
private void setParameters(PreparedStatement stmt,
                          Map values)
```

Metodi kopioi annetut arvot SQL-lausekkeeseen. Tätä kutsutaan vain metodin `executeInsert()` sisältä.

```
private Date StringToDate(String strDate)
```

Koska käyttäjät ilmoittavat päiväyksen merkkijonona, tarvitaan apumetodi, joka muuntaa merkkijonona annetun päiväyksen `strDate` `java.sql.Date`-olioksi.

Pakkausnäkyvyyden metodit:

```
protected String getSQLString()
```

Metodi palauttaa SQL-lauseen muotoa

```
INSERT INTO taulu (attr1, attr2, ..., attrN)
VALUES (?, ?, ..., ?)
```

Kentät `attr?` ovat kohdetaulun attribuutteja.

4.3.8 UpdateOperation

Luokka `UpdateOperation` perii luokan `DatabaseOperation`. Päivitysoperaatio muuttaa jonkin jo olemassa olevan rivin attribuuttien arvoja. Operaatio onnistuu, mikäli kohdetaulusta löytyy annetun `ROWID:n` osoittama rivi eikä tiedonsiirtovirheitä tapahdu.

Keskustelu päivitysoperaation kanssa on hyvin suoraviivaista. Ensin operaatio konstruoidaan antamalla parametriksi kohdetaulu. Tämän jälkeen koostetaan arvot kartaksi ja kutsutaan niillä metodia `executeUpdate`, joka saa parametrikseen uusien attribuuttien arvojen lisäksi päivitettävän tietueen `ROWID:n`. Kutsu palauttaa lisäysoperaation tuloksena `OperationResults`-olion. Päivitykset tehdään vain annetuille attribuuteille.

Muuttujat:

```
private SimpleDateFormat dateFormat
```

Muuttujaan sijoitetaan tietokannassa käytettävä päiväysten esitysmuoto.

Konstruktori:

```
public UpdateOperation(Table table)
```

Konstruktori luo uuden päivitysoperaation, joka kohdistuu parametrin `table` määrittelemään tietokantatauluun.

Julkiset metodit:

```
public OperationResults executeUpdate(Connection con,
                                     Map values, String rowId)
```

Metodi suorittaa päivitysoperaation taulun riville `rowId` muuttaen tämän rivin attribuuttien arvot kartan `values` mukaisiksi. Mikäli `Map`-olio `values` ei sisällä jonkin attribuutin arvoa, kyseinen attribuutti jätetään päivittämättä ja näin ollen sen arvo ei muutu.

Yksityiset metodit:

```
private void setParameters(PreparedStatement stmt,
                           Map values, String rowId)
```

Metodi kopioi uudet attribuuttien arvot sekä `ROWID`:n kyselylausekkeeseen. Annetut arvot korvaavat kaikki vanhat arvot. Mikäli arvokartasta `values` ei jotain arvoa löydy, kyseiseen kohtaan asetetaan arvo `NULL`. Eheysrajoitteen `NOT NULL` sisältävien kenttien kohdalla tästä aiheutuu väistämättä poikkeus ja toiminnon epäonnistuminen.

Käytännössä kaikkien kenttien päivityksestä ei kuitenkaan synny ongelmaa, sillä päivitysoperaatiota suoritettaessa servletin kutsuolio `javax.servlet.HttpServletRequest` sisältää tietueen kaikkien attribuuttien arvot – myös vanhat arvot niistä kentistä, joita käyttäjä ei ole muuttanut.

Pakkausnäkyvyyden metodit:

```
protected String getSQLString()
```

Metodi palauttaa SQL-lauseen muotoa

```
UPDATE taulu
SET attr1 = ?, attr2 = ?, ..., attrN = ?
WHERE ROWID = ?
```

Kentät `attr?` ovat kohdetaulun attribuutteja.

4.3.9 DeleteOperation

Luokka `DeleteOperation` perii luokan `DatabaseOperation`. Poisto-operaatio poistaa yhden rivin kohdetaulusta. Operaatio onnistuu, mikäli taulusta löytyy annettu rivi eikä tiedonsiirtovirheitä tapahdu.

Keskustelu poisto-operaation kanssa on hyvin suoraviivaista. Ensin operaatio konstruoidaan antamalla parametriksi kohdetaulu. Tämän jälkeen poisto-operaatio käynnistetään kutsumalla metodia `executeDelete()`, joka saa parametrinaan poistettavan rivin `ROWID`:n. Kutsu palauttaa poisto-operaation tuloksena `OperationResults`-instanssin.

Luokkaa `DeleteOperation` ei käytetä tämän suunnitteludokumentin pohjalta toteutettavassa Haliaeetus-järjestelmän versiossa, mutta se suunnitellaan ja toteutetaan tulevaisuuden tarpeita silmälläpitäen.

Konstruktori:

```
public DeleteOperation(Table table)
```

Konstruktori luo uuden poisto-operaation, joka kohdistuu parametrin `table` määrittelemään tietokantatauluun.

Julkiset metodit:

```
public OperationResults executeDelete(Connection con,
                                     String rowId)
```

Metodi poistaa kohdetaulusta rivin annetulla `ROWID`:llä.

Yksityiset metodit:

```
private void setParameters(PreparedStatement stmt,
                           String rowId)
```

Metodi kopioi `ROWID`:n arvon SQL-lausekkeeseen.

Pakkausnäkyvyyden metodit:

```
protected String getSQLString()
```

Metodi palauttaa SQL-lauseen muotoa

```
DELETE FROM taulu
WHERE ROWID = ?
```

4.4 Pakkaus `fi.hy.hali.servlet`

4.4.1 Hali

Hali on järjestelmän ainoa servletiluokka, joka hoitaa Haliaeetus-järjestelmän toiminnanohjauksen. Toiminnanohjaus tapahtuu if-else-rakenteen avulla haarautumalla istunnon tilan ja saatujen Action-parametrien mukaan. Servletti hoitaa parametrien käsittelyn oikeaan muotoon ja niiden syöttämisen eteenpäin Action-luokille ja View-luokille.

Julkiset metodit:

```
public void doGet(HttpServletRequestRequest, HttpServletResponse)
```

Metodia kutsutaan kun servletiltä pyydetään palvelua get-menetelmällä. Metodi sisältää kokonaisuudessaan Haliaeetus-järjestelmän toiminnanohjauksen.

```
public void doPost(HttpServletRequest request, HttpServletResponse)
```

Metodia kutsutaan kun servletiltä pyydetään palvelua post-menetelmällä. Metodi samaistetaan doGet-metodiin yksinkertaisuuden vuoksi kutsumalla doGet-metodia.

```
public void init(ServletConfig config)
```

Metodia kutsutaan aina kun servletti käynnistetään. Metodi mm. alustaa tarvittavia parametrejä.

4.5 Pakkaus fi.hy.hali.general

Pakkauksen `fi.hy.hali.general` luokat suorittavat erilaisia järjestelmän tarvitsemia aputoimintoja.

4.5.1 HaliProperties

Haliaeetus-ohjelmiston asetustiedot löytyvät tiedostosta `haliaeetus.config.HaliProperties`-luokan avulla tiedostosta saadaan luettua konfigurointitietoja muille luokille, kuten esimerkiksi käytettävä ajuri JDCB-yhteydelle.

`HaliProperties` perii Javan oman valmiin luokan `java.util.Properties`. Yliluokan metodeja käytetään avain-arvo merkkijonoparien lukemiseen tiedostosta. Kun luokkaa käytetään sen julkisin, staattisten metodein ensimmäisen kerran, asetustiedot luetaan tiedostosta `haliaeetus.config`.

Konstruktori:

```
private HaliProperties()
```

Konstruktoria tehtävänä on kutsua yliluokan konstruktoria. Konstruktori määrittää yksityiseksi eli luokasta ei voi luoda ilmentymiä, sillä luokkaa käytetään kirjastoluokan tavoin staattisten metodien avulla.

Julkiset metodit:

```
public static String get(String key)
```

Metodi palauttaa parametrilla `key` asetustiedostosta löytyvän arvon. Jos parametria `key` ei löydy asetustiedostosta, palautetaan `null`.

```
public static getInt(String key)
```

Metodi palauttaa parametrilla `key` asetustiedostosta löytyvän arvon, mutta muutettuna kokonaisluvuksi. Jos annetulla parametrilla ei löytynyt arvoa tiedostosta tai arvoa ei voida tulkita kokonaisluvuksi, palautetaan `-1`.

```
public static getBoolean(String key)
```

Metodi palauttaa parametrilla `key` asetustiedostosta löytyvän arvon, mutta muutettuna totuusarvoksi. Jos annetulla parametrilla ei löytynyt arvoa tiedostosta tai arvoa ei voida tulkita totuusarvoksi, palautetaan `false`.

```
public static Template getPageTemplate(String
                                     templateName)
```

Metodi palauttaa kutsujalleen `Template`-olion, joka vastaa parametrissa määritellyä templatetiedostoa. Parametrina `TemplateName` annettavassa mallipohjan tiedostonimessä täytyy olla mukana tiedoston `ftl`-pääte.

4.5.2 FileSave

`FileSave`-luokan tehtävänä on kirjoittaa `fi.hy.hali.action.reports-` pakkauksen luokkien suorittaman hakuoperaation tulostietoja (raportteja) määriteltyyn tekstitiedostoon. Myös luokka `Log` käyttää tätä luokkaa lokitietojen kirjoittamisessa tiedostoon.

Konstruktori:

```
public FileSave(String fileName)
```

Konstruktori luo `FileSave` -olion, jonka avulla voidaan tallettaa tietoja `fileName`-parametrilla määriteltyyn tiedostoon.

Julkiset metodit:

```
public static boolean fileExists(String fileName)
```

Metodilla voidaan selvittää ennen `FileSave` -olion luomista, onko parametrina annettu tiedosto jo olemassa. Jos tiedosto löytyy käyttäjää voidaan varoittaa ylikirjoituksesta tai pyytää määrittelemään uusi tiedoston nimi.

```
public void saveData(String data)
```

Metodi tallettaa parametrin `data` tiedostoon ja lisää perään määritellyn määrän tabulaattoreita. Tabulaattorien määrän saa asetettua metodilla `setNumberOfTabs(int number)`. Jos asetusta ei ole tehty, käytetään oletusarvoa.

```
public void nextRow()
```

Metodi suorittaa tekstitiedostossa rivinvaihdon.

```
public void saveRow(String[] values)
```

Metodi tallettaa tiedostoon yhdelle riville parametrina annetun merkkijonotaulukon tiedot sijoittaen tietojen väliin määritellyn määrän / oletusmäärän tabulaattoreja ja suorittaa lopuksi rivinvaihdon.

4.6 Pakkaus `fi.hy.hali.action`

4.6.1 Action

Abstrakti luokka `Action` on pakkauksen `fi.hy.hali.action`-pakkauksen luokkien ylliluokka. Siihen on koottu kaikille `fi.hy.hali.action`-pakkauksen luokille yhteiset toiminnot.

Abstraktit metodit:

```
public abstract Map execute(Map oldDatamodel, Map  
newDatamodel)
```

Abstrakti `execute`-metodi toteutetaan aliluokissa. Metodi käynnistää aliluokan ilmentymän toiminnan. Saa parametrina `Map`-olion, joka sisältää tarvittavan datan toiminnon suorittamista varten ja palauttaa `Map`-olion, joka sisältää tulosdatan.

4.6.2 LoginValidation

Luokka `LoginValidation` kutsuu luokkaa `ValidateLogin`, joka suorittaa tunnuksen ja salasanan tarkistuksen.

Julkiset metodit:

```
public Map execute(Map oldDatamodel, Map newDatamodel)
```

Välittää parametrina saadun `Map`-olion sisältämän tunnuksen ja salasanan `ValidateLogin`-luokalle ja palauttaa `ValidateLogin`-luokalta saamansa tuloksen `Map`-oliassa.

4.7 Pakkaus fi.hy.hali.action.search

4.7.1 Search

Abstrakti luokka `Search` on pakkauksen `fi.hy.hali.action.search` luokkien ylluokka. Siihen on koottu kaikille hauille yhteisiä muuttujia ja metodeita.

4.7.2 PesaSearch

`PesaSearch` luokka suorittaa pesätietojen haun. Ennen haun suorittamista hakuehdot tarkistetaan `fi.hy.hali.validate` pakkauksen `ValidateHaku`-luokan avulla.

Julkiset metodit:

```
public Map execute(Map oldDatamodel, Map newDatamodel)
```

Metodi suorittaa tietokantahaun parametrina saamansa `Map`-olion sisältämien hakuehtojen mukaisesti ja palauttaa `Map`-oliona haussa saadut pesätiedot.

4.7.3 KuntaSearch

`KuntaSearch` luokka suorittaa kuntatietojen haun. Ennen haun suorittamista hakuehdot tarkistetaan `fi.hy.hali.validate` pakkauksen `ValidateHaku`-luokan avulla.

Julkiset metodit:

```
public Map execute(Map oldDatamodel, Map newDatamodel)
```

Metodi suorittaa tietokantahaun parametrina saamansa `Map`-olion sisältämien hakuehtojen mukaisesti ja palauttaa `Map`-oliona haussa saadut kuntatiedot.

4.7.4 TarkastajaSearch

`TarkastajaSearch` luokka suorittaa tarkastajatietojen haun. Ennen haun suorittamista hakuehdot tarkistetaan `fi.hy.hali.validate` pakkauksen `ValidateHaku`-luokan avulla.

Julkiset metodit:

```
public Map execute(Map oldDatamodel, Map newDatamodel)
```

Metodi suorittaa tietokantahaun parametrina saamansa `Map`-olion sisältämien hakuehtojen mukaisesti ja palauttaa `Map`-oliona haussa saadut tarkastajatiedot.

4.7.5 LajiSearch

`LajiSearch` luokka suorittaa lajitietojen haun. Ennen haun suorittamista hakuehdot tarkistetaan `fi.hy.hali.validate` pakkauksen `ValidateHaku`-luokan avulla.

Julkiset metodit:

```
public Map execute(Map oldDatamodel, Map newDatamodel)
```

Metodi suorittaa tietokantahaun parametrina saamansa `Map`-olion sisältämien hakuehtojen mukaisesti ja palauttaa `Map`-oliona haussa saadut lajitiedot.

4.7.6 AputauluSearch

`AputauluSearch` luokka suorittaa aputaulutietojen haun. Ennen haun suorittamista hakuehdot tarkistetaan `fi.hy.hali.validate` pakkauksen `ValidateHaku`-luokan avulla.

Julkiset metodit:

```
public Map execute(Map oldDatamodel, Map newDatamodel)
```

Metodi suorittaa tietokantahaun parametrina saamansa `Map`-olion sisältämien hakuehtojen mukaisesti ja palauttaa `Map`-oliona haussa saadut aputaulutiedot.

4.7.7 HistoriaSearch

`HistoriaSearch` luokka suorittaa historiapesätietojen haun. Ennen haun suorittamista hakuehdot tarkistetaan `fi.hy.hali.validate` pakkauksen `ValidateHaku`-luokan avulla.

Julkiset metodit:

```
public Map execute(Map oldDatamodel, Map newDatamodel)
```

Metodi suorittaa tietokantahaun parametrina saamansa `Map`-olion sisältämien hakuehtojen mukaisesti ja palauttaa `Map`-oliona haussa saadut historiapesätiedot.

4.7.8 ReviiriSearch

`ReviiriSearch` luokka suorittaa reviiritietojen haun. Ennen haun suorittamista hakuehdot tarkistetaan `fi.hy.hali.validate` pakkauksen `ValidateHaku`-luokan avulla.

Julkiset metodit:

```
public Map execute(Map oldDatamodel, Map newDatamodel)
```

Metodi suorittaa tietokantahaun parametrina saamansa `Map`-olion sisältämien hakuehtojen mukaisesti ja palauttaa `Map`-oliona haussa saadut reviiritiedot.

4.7.9 SaalisSearch

`SaalisSearch` luokka suorittaa saalistietojen haun. Ennen haun suorittamista hakuehdot tarkistetaan `fi.hy.hali.validate` pakkauksen `ValidateHaku`-luokan avulla.

Julkiset metodit:

```
public Map execute(Map oldDatamodel, Map newDatamodel)
```

Metodi suorittaa tietokantahaun parametrina saamansa `Map`-olion sisältämien hakuehtojen mukaisesti ja palauttaa `Map`-oliona haussa saadut saalistiedot.

`SaalisSearch`-luokkaa ei toteuteta Hali2 [17] projektin puitteissa.

4.7.10 MyrkkySearch

`MyrkkySearch` luokka suorittaa myrkkytietojen haun. Ennen haun suorittamista hakuehdot tarkistetaan `fi.hy.hali.validate` pakkauksen `ValidateHaku`-luokan avulla.

Julkiset metodit:

```
public Map execute(Map oldDatamodel, Map newDatamodel)
```

Metodi suorittaa tietokantahaun parametrina saamansa `Map`-olion sisältämien hakuehtojen mukaisesti ja palauttaa `Map`-oliona haussa saadut myrkkytiedot.

`MyrkkySearch`-luokkaa ei toteuteta Hali2 [17] projektin puitteissa.

4.8 Pakkaus fi.hy.hali.action.update

4.8.1 Update

Abstrakti luokka `Update` on pakkauksen `fi.hy.hali.action.update` luokkien ylluokka. Siihen on koottu kaikille aliluokille yhteisiä muuttujia ja metodeita.

4.8.2 PesaUpdate

`PesaUpdate`-luokka suorittaa pesätietojen päivityksen. Ennen tietokantapäivityksen tekemistä se kutsuu `ValidatePesatiedot`-luokkaa tarkistaakseen päivitettävien tietojen oikeellisuuden.

Julkiset metodit:

```
public Map execute(Map oldDatamodel, Map newDatamodel)
```

Metodi suorittaa `Map`-oliossa saamiensa pesätietojen päivityksen tietokantaan ja palauttaa `Map`-oliossa tiedon päivityksen onnistumisesta.

4.8.3 KuntaUpdate

`KuntaUpdate`-luokka suorittaa kuntatietojen päivityksen. Ennen tietokantapäivityksen tekemistä se kutsuu `ValidateKunnat`-luokkaa tarkistaakseen päivitettävien tietojen oikeellisuuden.

Julkiset metodit:

```
public Map execute(Map oldDatamodel, Map newDatamodel)
```

Metodi suorittaa `Map`-oliossa saamiensa kuntatietojen päivityksen tietokantaan ja palauttaa `Map`-oliossa tiedon päivityksen onnistumisesta.

4.8.4 TarkastajaUpdate

`TarkastajaUpdate`-luokka suorittaa tarkastajatietojen päivityksen. Ennen tietokantapäivityksen tekemistä se kutsuu `ValidateTarkastajat`-luokkaa tarkistaakseen päivitettävien tietojen oikeellisuuden.

Julkiset metodit:

```
public Map execute(Map oldDatamodel, Map newDatamodel)
```

Metodi suorittaa `Map`-oliossa saamiensa tarkastajätietojen päivityksen tietokantaan ja palauttaa `Map`-oliossa tiedon päivityksen onnistumisesta.

4.8.5 LajiUpdate

`LajiUpdate`-luokka suorittaa lajitietojen päivityksen. Ennen tietokantapäivityksen tekemistä se kutsuu `ValidateLajit`-luokkaa tarkistaakseen päivitettävien tietojen oikeellisuuden.

Julkiset metodit:

```
public Map execute(Map oldDatamodel, Map newDatamodel)
```

Metodi suorittaa `Map`-oliossa saamiensa lajitietojen päivityksen tietokantaan ja palauttaa `Map`-oliossa tiedon päivityksen onnistumisesta.

4.8.6 AputauluUpdate

`AputauluUpdate`-luokka suorittaa aputaulutietojen päivityksen. Ennen tietokantapäivityksen tekemistä se kutsuu `ValidateAputaulut`-luokkaa tarkistaakseen päivitettävien tietojen oikeellisuuden.

Julkiset metodit:

```
public Map execute(Map oldDatamodel, Map newDatamodel)
```

Metodi suorittaa `Map`-oliossa saamiensa lajitietojen päivityksen tietokantaan ja palauttaa `Map`-oliossa tiedon päivityksen onnistumisesta.

4.8.7 HistoriaUpdate

`HistoriaUpdate`-luokka suorittaa historiapesätietojen päivityksen. Ennen tietokantapäivityksen tekemistä se kutsuu `ValidateHistoria`-luokkaa tarkistaakseen päivitettävien tietojen oikeellisuuden.

Julkiset metodit:

```
public Map execute(Map oldDatamodel, Map newDatamodel)
```

Metodi suorittaa `Map`-oliossa saamiensa lajitietojen päivityksen tietokantaan ja palauttaa `Map`-oliossa tiedon päivityksen onnistumisesta.

4.8.8 ReviiriUpdate

ReviiriUpdate-luokka suorittaa reviiritietojen päivityksen. Ennen tietokantapäivityksen tekemistä se kutsuu ValidateReviirit-luokkaa tarkistaakseen päivitettävien tietojen oikeellisuuden.

Julkiset metodit:

```
public Map execute(Map oldDatamodel, Map newDatamodel)
```

Metodi suorittaa Map-oliossa saamiensa lajitietojen päivityksen tietokantaan ja palauttaa Map-oliossa tiedon päivityksen onnistumisesta.

4.8.9 SaalisUpdate

SaalisUpdate-luokka suorittaa saalistietojen päivityksen. Ennen tietokantapäivityksen tekemistä se kutsuu ValidateSaaliit-luokkaa tarkistaakseen päivitettävien tietojen oikeellisuuden.

Julkiset metodit:

```
public Map execute(Map oldDatamodel, Map newDatamodel)
```

Metodi suorittaa Map-oliossa saamiensa lajitietojen päivityksen tietokantaan ja palauttaa Map-oliossa tiedon päivityksen onnistumisesta.

SaalisUpdate-luokkaa ei toteuteta Hali2 [17] projektin puitteissa.

4.8.10 MyrkkyUpdate

MyrkkyUpdate-luokka suorittaa myrkkytietojen päivityksen. Ennen tietokantapäivityksen tekemistä se kutsuu ValidateMyrkkyt-luokkaa tarkistaakseen päivitettävien tietojen oikeellisuuden.

Julkiset metodit:

```
public Map execute(Map oldDatamodel, Map newDatamodel)
```

Metodi suorittaa Map-oliossa saamiensa lajitietojen päivityksen tietokantaan ja palauttaa Map-oliossa tiedon päivityksen onnistumisesta.

MyrkkyUpdate-luokkaa ei toteuteta Hali2 [17] projektin puitteissa.

4.9 Pakkaus `fi.hy.hali.action.report`

Pakkauksen `fi.hy.hali.action.report` luokat generoivat käyttäjän pyytämät raportit kahteen tekstitiedostoon. Toinen tiedosto on muotoiltu ja toinen yksinkertainen tekstitiedosto, jossa datasarakkeet on eroteltu tabulaattorilla. Muotoillun tiedoston malli haetaan FreeMarker templatesta. Pakkaukseen kuuluvat Report -yliluokka ja sen aliluokat ReportA:sta ReportR:ään. Tiedostot tallennetaan `fi.hy.hali.general-pakkauksen` `FileSave`-luokalla.

Tarkemmat kuvaukset raporttien ulkoasusta löytyvät vaatimusdokumentista.

4.9.1 Report

Pakkauksen `fi.hy.hali.action.report` luokkien abstrakti ylliluokka. Sisältää kaikille paketin luokille yhteisiä muuttujia ja metodeita.

4.9.2 ReportA

Luokka tuottaa yhteenvedon suojeluviranomaisten käyttöön. Listaa reviirin perustiedot ja kaikkien reviirillä olevien pesien perustiedot, kuten puusto, näkyvyys pesältä ja tarkastajan nimi.

Julkiset metodit:

```
public Map execute(Map oldDatamodel, Map newDatamodel)
```

4.9.3 ReportB

Tarkastajien tekemät pesätarkastukset. Listaa kunkin tarkastajan tarkastamien pesien lukumäärän annetulla aikavälillä.

Julkiset metodit:

```
public Map execute(Map oldDatamodel, Map newDatamodel)
```

4.9.4 ReportC

Reviiri- ja pesäkohtaiset pesimistulos ja kuoriutumistiedot. Listaa pesäkohtaiset pesimistulos- ja kuoriutumistiedot maa-, suur- alue-, kunta- tai reviiritarkkuudella.

Julkiset metodit:

```
public Map execute(Map oldDatamodel, Map newDatamodel)
```

4.9.5 ReportD

Kuoriutumispäivät. Listaa haluttujen vuosien kuoriutumispäivät ja -määrät vuorokauden tarkkuudella.

Julkiset metodit:

```
public Map execute(Map oldDatamodel, Map newDatamodel)
```

4.9.6 ReportE

Yhteenvedo kohdan D raportista. Raportissa on listattuna kuoriutumispäivät. Kuoriutumispäivien mediaanit, keskiarvot, keskiarvojen, lukumäärät keskivirheet listataan kutakin poikasmäärää kohti.

Julkiset metodit:

```
public Map execute(Map oldDatamodel, Map newDatamodel)
```

4.9.7 ReportF

Pesimistulokset reviireittäin. Raportissa on kaksi taulua. Toisessa listataan pesimistulokset (A:sta V:hen) kutakin vuotta kohti. Kussakin pesimistuloksessa ilmoitetaan kyseisten tulosten lukumäärät reviireittäin. Paras tulos reviirissä huomioidaan. Toisessa taulussa listataan vuosittaiset reviirikohtaiset poikasmäärät, asuttujen reviirien lukumäärät eriteltynä luokkiin pesimistuloksien mukaan, munapesien lukumäärät ja poikaspesien lukumäärät.

Julkiset metodit:

```
public Map execute(Map oldDatamodel, Map newDatamodel)
```

4.9.8 ReportG

Puustot. Raportissa listataan vuosittaisten pesintöjen lukumäärät puustokohtaisesti. Kutakin puustoa kohti ilmoitetaan pesintöjen määrä kyseisessä puustossa sekä lukumäärän prosentuaalinen osuus kaikista pesinnöistä.

Julkiset metodit:

```
public Map execute(Map oldDatamodel, Map newDatamodel)
```

4.9.9 ReportH

Tarkat poikuekoot rengastushetkellä. Listaa vuosittaisen rengastettujen lintujen poikuekoot.

Julkiset metodit:

```
public Map execute(Map oldDatamodel, Map newDatamodel)
```

4.9.10 ReportI

Pesäpuun elävyys. Listaa pesäpuiden vuosittaisen elävyysasteen.

Julkiset metodit:

```
public Map execute(Map oldDatamodel, Map newDatamodel)
```

4.9.11 ReportJ

Pesäpuun puulaji (luonnonpesät). Listataan vuosittain pesäpuiden puulajien (luonnonpesät). Tulos ilmoitetaan prosenttiosuuksina. Lasketaan myös keskiarvo-osuudet vuosien yli.

Julkiset metodit:

```
public Map execute(Map oldDatamodel, Map newDatamodel)
```

4.9.12 ReportK

Pesäpuun puulaji (tekopesät). Raportin J kaltainen tuloste, mutta tekopesille.

Julkiset metodit:

```
public Map execute(Map oldDatamodel, Map newDatamodel)
```

4.9.13 ReportL

Pesän sijainti. Listataan vuosittain, mihin pesä on rakennettu (sijainti eri kohdissa puuta, tekopesä yms.). Tulos ilmoitetaan prosenttiosuuksina. Lasketaan myös keskiarvo-osuudet vuosien yli.

Julkiset metodit:

```
public Map execute(Map oldDatamodel, Map newDatamodel)
```

4.9.14 ReportM

Puustojen käsittelyasteet. Listataan vuosittain pesää ympäröivän puuston käsittelyaste. Tulos ilmoitetaan prosenttiosuuksina. Lasketaan myös keskiarvo-osuudet vuosien yli.

Julkiset metodit:

```
public Map execute(Map oldDatamodel, Map newDatamodel)
```

4.9.15 ReportN

Pesien sijainti pesäsaaren tyyppin mukaan. Listataan vuosittain pesän sijaintisaaren tyyppi. Tulos ilmoitetaan prosenttiosuuksina. Lasketaan myös keskiarvo-osuudet vuosien yli.

Julkiset metodit:

```
public Map execute(Map oldDatamodel, Map newDatamodel)
```


4.9.16 ReportO

Pesien sijainti maastotyyppin mukaan. Listataan vuosittain, millaisessa maastotyyppissä pesä sijaitsee. Tulos ilmoitetaan prosenttiosuuksina. Lasketaan myös keskiarvo-osuudet vuosien yli.

Julkiset metodit:

```
public Map execute(Map oldDatamodel, Map newDatamodel)
```

4.9.17 ReportP

Palstan rauhoitusaste ja pesimistulos (vuosittainen raportointi ja yhteenveto) Listataan vuosittain kaksiulotteinen taulukko, jossa riveinä on palstan rauhoitusaste ja sarakkeina pesimistulos. Tulos ilmoitetaan esiintymisfrekvenssinä. Tulostetaan myös vastaava taulukko, joka on yhteenveto kaikista vuosista.

Julkiset metodit:

```
public Map execute(Map oldDatamodel, Map newDatamodel)
```

4.9.18 ReportQ

Uusi Pesä. Uusi Pesä -raportin avulla kirjataan uuden pesän pesätarkastuksen tiedot tietokantaan. Raportti sisältää kohdat pesän sijainnin ja rauhoitustietojen, pesäpuun ja pesän mittojen, pesän ympäristön ja muihin pesätarkastukseen liittyvien tietojen kirjaamiseen.

Julkiset metodit:

```
public Map execute(Map oldDatamodel, Map newDatamodel)
```

4.9.19 ReportR

Vanha pesä. Vanha Pesä -raportti on Uusi Pesä -raportin kaltainen, mutta poikkeaa siinä, että pesän ja sen ympäristön, reviirin ja muut vastaavat ei-muuttuvat tai harvoin muuttuvat tiedot on esitetyt. Raportti tulostetaan säännöllistä pesätarkastusta varten. Pesätarkastuksen yhteydessä lomakkeeseen täytetään uudet ja muuttuneet tiedot.

Julkiset metodit:

```
public Map execute(Map oldDatamodel, Map newDatamodel)
```

4.10 Pakkaus fi.hy.hali.view

Pakkauksen luokkien tehtävä on tuottaa kaikki järjestelmän näytöt. Jokaista tuotettavaa näyttöä varten on oma aliluokka. Luokat kokoavat näytölle tulostettavat tiedot yhdistämällä saamansa parametrit ja näytölle tulevat otsikkotekstit, tallentavat ne `Map`-olioon ja välittävät sen eteenpäin varsinaiselle tulostusosalle, joka käyttää template-tiedostoja (.ftl) niiden tulostamiseen. Template-tiedostoissa määritellään kunkin näytön ulkoasu. Näyttöjen osat jotka esiintyvät useassa eri näytössä, kuten navigointiosio, tuotetaan erillisessä metodissa.

4.10.1 View

Pakkauksen `fi.hy.hali.view` luokkien abstrakti ylliluokka. Luokka sisältää kaikille näyttöluokille yhteisiä metodeita ja muuttujia. Lisäksi luokassa on aliluokkien toiminnan käynnistävä abstrakti `execute`-metodi, joka toteutetaan aliluokissa.

Abstraktit metodit:

```
public abstract String execute(Map datamodel)
```

4.10.2 ViewLogin

Tulostaa sisäänkirjautumisnäytön.

Julkiset metodit:

```
public String execute(Map datamodel)
```

4.10.3 ViewNavi

Tulostaa navigointipalkin.

Julkiset metodit:

```
public String execute(Map datamodel)
```

4.10.4 ViewPesaHaku

Tulostaa pesähakunäytön.

Julkiset metodit:

```
public String execute(Map datamodel)
```

4.10.5 ViewPesa

Tulostaa pesänäytön.

Julkiset metodit:

```
public String execute(Map datamodel)
```

4.10.6 ViewAputaulu

Tulostaa aputaulunäytön.

Julkiset metodit:

```
public String execute(Map datamodel)
```

4.10.7 ViewRaportit

Tulostaa raporttinäytön.

Julkiset metodit:

```
public String execute(Map datamodel)
```

4.10.8 ViewHistoria

Tulostaa historiapesänäytön.

Julkiset metodit:

```
public String execute(Map datamodel)
```

4.10.9 ViewKunnat

Tulostaa kuntanäytön.

Julkiset metodit:

```
public String execute(Map datamodel)
```

4.10.10 ViewMyrkyt

Tulostaa myrkkynäytön.

Julkiset metodit:

```
public String execute(Map datamodel)
```

4.10.11 ViewSaaliit

Tulostaa saalisnäytön.

Julkiset metodit:

```
public String execute(Map datamodel)
```

4.10.12 ViewSaaliitYllapito

Tulostaa saaliiden ylläpitoäytön.

Julkiset metodit:

```
public String execute(Map datamodel)
```

4.10.13 ViewTarkastajat

Tulostaa tarkastajanäytön.

Julkiset metodit:

```
public String execute(Map datamodel)
```

4.10.14 ViewReviirit

Tulostaa reviirinäytön.

Julkiset metodit:

```
public String execute(Map datamodel)
```

4.10.15 ViewLajit

Tulostaa lajinäytön.

Julkiset metodit:

```
public String execute(Map datamodel)
```

4.11 Pakkaus fi.hy.hali.validate

4.11.1 Validate

Validate on abstrakti luokka tietokannan syötetietojen tarkistukseen. Luokka sisältää yleisiä validaatioon liittyviä yksityisiä metodeita sekä abstraktin metodin `execute`.

Abstraktit metodit:

```
public abstract List execute(Map values)
```

Metodi `execute` saa parametrina `Map`-olion, josta se etsii mahdolliset virheet ja palauttaa virheet sisältävän `List`-olion jos virheitä löytyy tai null, jos virheitä ei esiinny. Toteutetaan aliluokissa.

4.11.2 ValidateLogin

Tarkistaa vastaavatko parametrina saatu tunnus ja salasana tietokannan tietoja.

Julkiset metodit:

```
public List execute(Map values)
```

4.11.3 ValidateHaku

Tarkistaa ovatko haun parametrit oikeassa muodossa.

Julkiset metodit:

```
public List execute(Map values)
```

4.11.4 ValidatePesätiedot

Tarkistaa ovatko pesätiedot oikeassa muodossa.

Julkiset metodit:

```
public List execute(Map values)
```

4.11.5 ValidateRaportit

Tarkistaa ovatko raporttien luomisparametrit oikeassa muodossa.

Julkiset metodit:

```
public List execute(Map values)
```

4.11.6 ValidateHistoria

Tarkistaa ovatko historiatiedot oikeellisia.

Julkiset metodit:

```
public List execute(Map values)
```

4.11.7 ValidateKunnat

Tarkistaa ovatko kuntatiedot oikeellisia.

Julkiset metodit:

```
public List execute(Map values)
```

4.11.8 ValidateMyrkyt

Tarkistaa ovatko myrkkytiedot oikeellisia.

Julkiset metodit:

```
public List execute(Map values)
```

4.11.9 ValidateSaaliit

Tarkistaa ovatko saalistiedot oikeellisia.

Julkiset metodit:

```
public List execute(Map values)
```

4.11.10 ValidateTarkastajat

Tarkistaa ovatko tarkastajan tiedot oikeellisia.

Julkiset metodit:

```
public List execute(Map values)
```

4.11.11ValidateReviirit

Tarkistaa ovatko reviiritiedot oikeellisia.

Julkiset metodit:

```
public List execute(Map values)
```


4.12 Pakkaus `fi.hy.hali.log`

4.12.1 Log

Pakkauksen `fi.hy.hali.log` ainoan luokan `Log` tehtävänä on kirjata tietokannan taulujen attribuutteihin tehdyt muutokset tekstitiedostoihin. Muutoksella tarkoitetaan tilannetta, jossa arvo muutetaan toiseksi. Uuden rivin lisäys ei siis ole lokiin kirjattava muutos. Jokaista muutosta kohden tuotetaan oma tiedostonsa. Lokitiedostoon merkitään muutetun taulun nimi, attribuuttien nimet ja muutos aika. `Update` -luokkien suorittaessa tietokantapäivitystä ne kutsuvat `Log` -luokan metodia `compareMap` joka suorittaa vertailun päivitettävän ja tietokannassa olevan tiedon välillä.

Julkiset metodit:

```
public boolean compareMap(Map oldDatamodel, Map
newDatamodel)
```

Metodi vertailee parametreina saamiensa `Map` -olioiden sisältämien kenttien arvoja. Mikäli tietokantaan vietävässä uudessa versiossa kentän sisältö on muuttunut tietokannassa olevan vanhan version vastaavaan kenttään verrattuna, muutoksesta tehdään lokikirjaus tekstitiedostoon. Metodi palauttaa `true`, jos lokikirjaus onnistui tai lokiin ei ole mitään kirjattavaa ja `false`, jos lokikirjaus epäonnistui.

4.13 Datamalli

Datamalli on puutietorakenne, jonka avulla siirretään dataa järjestelmän sisällä. Datamalli on `Map`-olio, jonka sisällä voi olla `Map`-olioita, sekä `List`-oliota. Järjestelmän toteutuksessa `Map`-olioina käytetään `Map`-rajapinnan toteuttavia `HashMap`-luokan ilmentymiä ja `List`-oliona `List`-rajapinnan toteuttavia `ArrayList`-luokan ilmentymiä. `Map`- ja `List`-rajapintojen käyttäminen mahdollistaa kuitenkin kaikkien rajapinnan toteuttavien luokkien käytön datamallissa.

Datamalli sisältää järjestelmän toiminnassa välitettäviä parametreja. Järjestelmän luokat muuttavat datamallin sisältöä tarpeen mukaan, eli se sisältää aina vain tarvittavat parametrit. Datamalli voi sisältää tunnuksen ja salasanan, tietokantaan talletettavia attribuutteja, tietokantahakuun tarvittavia määreitä, virheilmoituksia, tietoja tuotettavien `html`-sivujen dropdown-listoista tai konfigurointitiedoston polkumääreen.

Datamallissa näyttöihin tulevat staattiset tekstit on sijoitettu `text`-oksaan muodossa `text.näyttö.kenttä`. Datamallin tekstikenttiin tietokannasta ladattavat arvot on esitetty muodossa `data[].taulu.attribuutti`. Lisäksi

datamallissa on `error`-oksa virheilmoituksia varten ja `searchparameters`-oksa hakumääreitä varten.

Tässä dokumentissa datamalli kuvataan seuraavasti puuna:

```
(root)
+solmu
  +muuttuja
+sekvenssi[]
  +solmu
    +muuttuja1
    +muuttuja2
```

Mallissa jokainen puun normaali solmu voi toimia tyhjänä solmuna (=hakemisto) tai muuttujana (sisältää arvon). Sekvenssityyppiset solmut taas toimivat kuin taulukko. Esimerkissä siis `sekvenssi` toimii kuin taulukko, ja viittaus `sekvenssi[1]` viittaa sekvenssin toiseen muuttujaan/alkioon. Sekvenssin alkiot voivat toimia myös hakemistorakenteina, ja yhdessä alkiossa voi olla monta muuttujaa. Sekvenssirakenteet toteutetaan `List`-rajapinnan toteuttavan luokan ilmentymien avulla. Tavallisiin muuttujiin viitataan templatessa esim `solmu.muuttuja`.

4.14 Muut tiedostot

Järjestelmä käyttää mallipohjia (template-tiedostot) dynaamisten html-sivujen tuottamiseen. Järjestelmän kielituki toteutetaan Javan `properties`-tiedostoilla.

4.14.1 Template-tiedostot

Servlet-luokkien tuottamat tulosteet muunnetaan käyttäjän ymmärtämään HTML-muotoon FreeMarker-mallipohjien 6.3 eli template-tiedostojen (`.ftl`) avulla. Jokaista näyttöä kohden on yksi mallipohja paitsi näytöt Uusi pesä, Pesätiedot ja Uusi tarkastus käyttävät samaa mallipohjaa. Lisäksi on mallipohja sivujen yläosan linkkejä varten.

Template-tiedostot sisältävät tavallista HTML-koodia sekä FreeMarker-komentoja, joiden avulla on toteutettu mm. ehtorakenteet ja luettelomuotoisen tiedon esitys. Template-tiedostojen avulla servletti voi generoida käyttöliittymän näytölle valmiita lomakepohjia, listata list-operaatiolla dynaamisesti tulosjoukkoja näytölle ja yhdistellä eri tavoin näytöllä esitettäviä osia `include`-komennoilla. Template-tiedostot vastaavat käyttöliittymän eri näyttöjä ja niiden avulla voidaan esim. `if`-lauseita käyttäen generoida näytön virheilmoituksia.

HTML-lomakkeissa käytetyt kenttien ja FreeMarker-muuttujien nimet vastaavat (soveltuvilta osin) tietokannan taulujen ja sarakkeiden nimiä (muodossa `taulu.attribuutti`).

Olenlaisin asia templateissa on datamalli, jonka tietoa käsittelevä luokka luo. Datamallista template voi suoraan sijoittaa html:n sekaan muuttujia. Tässä luvussa keskitytään datamallin rakenteeseen, koska itse `ftl`-tiedostot ovat melko triviaaleja html-kieltä ja perusohjelmointia tuntevalle, ja tiedostojen perusrakenteen pystyy näin ollen päättämään näyttöjen kuvista kappaleessa 5.

Haliaeetus-ratkaisussa kaikki näyttöjen staattiset tekstit haetaan kielituen takia tiedostosta. Tämä aiheuttaa sen, että datamallipuista tulee isoja ja tällä esitystavalla graafisesti korkeita (itse puuhan on melko matala).

4.14.1.1navi.ftl

```
(root)
+ text
|
|   +haku
|   +uusi_pesa
|   +raportit
|   +reviirit
|   +aputaulut
|   +kunnat
|   +tarkastajat
|   +kirjaudu_ulos
```

`navi.ftl` on navigointipalkin template, joka näkyy jokaisen näytön ylälaidassa. Se sisällytetään muihin templateihin include-lauseella.

4.14.1.2login.ftl

```
(root)
|
+userid          ("tunnus")
|
+password        ("salasana")
|
+haliURL         ("https://db.cs.helsinki.fi/s/tunnus/")
|
+error
|   +login_userid   ("virheellinen käyttäjätunnus")
|   +login_password ("virheellinen salasana")
|
+text
|   +login_userid   ("Käyttäjätunnus :")
|   +login_password ("Salasana :")
|   +login_submit   ("SISÄÄN")
|   +login_title    ("HALIAEETUS - Sisäänkirjautuminen")
|   +login_halisystem ("HALIAEETUS-JÄRJESTELMÄ")
```

`login.ftl` tuottaa sisäänkirjautumissivun.

4.14.1.3haku.ftl

```
(root)
|
+error
|   +haku_pesanimi   ("Virheellinen pesänimi")
|   +haku_pesa_kunta ("Virheellinen kunta")
|   +haku_pesa_id    ("Virheellinen pesäID")
|   +haku_reviiri_nimi ("Virheellinen reviirinimi")
|   +haku_reviiri_kunta ("Virheellinen kunta")
|   +haku_reviiri_id ("Virheellinen reviiriID")
|   +haku_yht_pituus ("Virheellinen pituuskoordinaatti")
|   +haku_yht_leveys ("Virheellinen leveyskoordinaatti")
|   +haku_ast_pituus ("Virheellinen pituuskoordinaatti")
|   +haku_ast_leveys ("Virheellinen leveyskoordinaatti")
|
+text
|   +haku_title      ("Haku")
```

```

+haku_hakuehdot_title      ("Hakuehdot")
+haku_vuosivali_title      ("VUOSIVÄLI")
+haku_pesa_title           ("Pesä")
+haku_pesanimi             ("nimi:")
+haku_pesa_kunta           ("kunta:")
+haku_pesa_id              ("id:")
+haku_reviiri_nimi         ("nimi:")
+haku_reviiri_kunta        ("kunta:")
+haku_reviiri_id           ("id:")
+haku_suur_alue            ("Suuralue:")
+haku_ymp_keskus           ("Ympäristökeskus:")
+haku_koord_title          ("Koordinaatit:")
+haku_yht_koord            ("Yhtenäiskoordinaatisto")
+haku_yht_pituus           ("Pituus")
+haku_yht_leveys           ("Leveys")
+haku_ast_koord            ("Astekoordinaatisto")
+haku_ast_pituus           ("Pituus")
+haku_ast_leveys           ("Leveys")
+haku_submitbutton1       ("Hae pesät")
+haku_haetut_pesat_title   ("Haetut pesät")
+haku_haetut_pesat_pesa_id ("Numero")
+haku_haetut_pesat_pesanimi ("Nimi")
+haku_haetut_pesat_reviiri_nimi ("Reviiri")
+haku_haetut_pesat_kunta_nimi ("Kunta")
+haku_haetut_pesat_submitlink1 ("Uusi Tarkastus")

+ddlist
+ddlist_suur_alue
+ddlist_suur_alue_selected
+ddlist_ymp_keskus
+ddlist_ymp_keskus_selected

+searchparameters
+reviiri
+reviiri_nimi
+reviiri_id
+pesavakio
+pesa_id
+pesanimi
+yht_leveys
+yht_pituus
+ast_leveys
+ast_pituus
+kunta
+kunta_nimi
+suur_alue
+ymp_keskus
+vuosi
+vuosialku
+vuosiloppu

+data[]
+reviiri
+reviiri_nimi
+pesavakio
+pesa_id
+pesanimi
+kunta
+kunta_nimi

```

Tuottaa hakusivun.

4.14.1.4 pesatiedot

```

(root)
|
+error
|
+pesatiedot_tark_pvm      ("Virheellinen päivämäärä")
+pesatiedot_yht_pituus   ("Virhe koordinaatissa")
+pesatiedot_yht_leveys   ("Virhe koordinaatissa")
+pesatiedot_ast_pituus   ("Virhe koordinaatissa")
+pesatiedot_ast_leveys   ("Virhe koordinaatissa")
+pesatiedot_rak_vuosi    ("Virheellinen vuosi")
+pesatiedot_loyt_vuosi   ("Virheellinen vuosi")
+pesatiedot_rak_laji     ("Virheellinen lajitunnus")
+pesatiedot_r_taulu_pvm  ("Virheellinen päivämäärä")

```

+pesatiedot_r_taulu_nro	("Virheellinen taulun numero")
+pesatiedot_palsta_rauh_pvm	("Virheellinen päivämäärä")
+pesatiedot_rauh_aika_alku	("Virheellinen päivämäärä")
+pesatiedot_rauh_aika_loppu	("Virheellinen päivämäärä")
+pesatiedot_suojelualue	("Virheellinen suojelualan nimi")
+pesatiedot_pesa_mit_pvm	("Virheellinen päivämäärä")
+pesatiedot_korkeus	("Virheellinen puun korkeus")
+pesatiedot_et_maasta	("Virheellinen etäisyys")
+pesatiedot_et_latvasta	("Virheellinen etäisyys")
+pesatiedot_tyviymparys	("Virheellinen rungon ympärysmitta")
+pesatiedot_tyvihalkaisija	("Virheellinen rungon halkaisija")
+pesatiedot_latvaymparys	("Virheellinen rungon ympärysmitta")
+pesatiedot_latvahalkaisija	("Virheellinen rungon halkaisija")
+pesatiedot_pesa_korkeus	("Virheellinen pesän korkeus")
+pesatiedot_pesa_halk_max	("Virheellinen pesan suurin halkaisija")
+pesatiedot_pesa_halk_min	("Virheellinen pesan pienin halkaisija")
+pesatiedot_et_meri	("Virheellinen etäisyys")
+pesatiedot_et_jarvi	("Virheellinen etäisyys")
+pesatiedot_et_avosuo	("Virheellinen etäisyys")
+pesatiedot_et_as	("Virheellinen etäisyys")
+pesatiedot_et_autotie	("Virheellinen etäisyys")
+pesatiedot_et_moottorikelkka	("Virheellinen etäisyys")
+pesatiedot_et_talvitie	("Virheellinen etäisyys")
+pesatiedot_et_kalaviljely	("Virheellinen etäisyys")
+pesatiedot_et_ilmajohto	("Virheellinen etäisyys")
+pesatiedot_et_viljapelto	("Virheellinen etäisyys")
+pesatiedot_et_avohakkuu	("Virheellinen etäisyys")
+pesatiedot_et_lahipuu	("Virheellinen etäisyys")
+pesatiedot_as_lkm_1000	("Virheellinen asuntojen määrä")
+pesatiedot_as_lkm_500	("Virheellinen asuntojen määrä")
+pesatiedot_manty_lkm_p	("Virheellinen mäntyjien lukumäärä (p)")
+pesatiedot_manty_pit_p	("Virheellinen mäntyjien keskipituus (p)")
+pesatiedot_manty_lkm_i	("Virheellinen mäntyjien määrä (i)")
+pesatiedot_manty_pit_i	("Virheellinen mäntyjien keskipituus (i)")
+pesatiedot_manty_lkm_e	("Virheellinen mäntyjien määrä (e)")
+pesatiedot_manty_pit_e	("Virheellinen mäntyjien keskipituus (e)")
+pesatiedot_manty_lkm_l	("Virheellinen mäntyjien määrä (l) ")
+pesatiedot_manty_pit_l	("Virheellinen mäntyjien keskipituus (l)")
+pesatiedot_kuusi_lkm_p	("Virheellinen kuusten lukumäärä (p)")
+pesatiedot_kuusi_pit_p	("Virheellinen kuusten keskipituus (p)")
+pesatiedot_kuusi_lkm_i	("Virheellinen kuusten lukumäärä (i)")
+pesatiedot_kuusi_pit_i	("Virheellinen kuusten keskipituus (i)")
+pesatiedot_kuusi_lkm_e	("Virheellinen kuusten lukumäärä (e)")
+pesatiedot_kuusi_pit_e	("Virheellinen kuusten keskipituus (e)")
+pesatiedot_kuusi_lkm_l	("Virheellinen kuusten lukumäärä (l)")
+pesatiedot_kuusi_pit_l	("Virheellinen kuusten keskipituus (l)")
+pesatiedot_muu_lkm_p	("Virheellinen puiden lukumäärä (p)")
+pesatiedot_muu_pit_p	("Virheellinen puiden keskipituus (p)")
+pesatiedot_muu_lkm_i	("Virheellinen puiden lukumäärä (i)")
+pesatiedot_muu_pit_i	("Virheellinen puiden keskipituus (i)")
+pesatiedot_muu_lkm_e	("Virheellinen puiden lukumäärä (e)")
+pesatiedot_muu_pit_e	("Virheellinen puiden keskipituus (e)")
+pesatiedot_muu_lkm_l	("Virheellinen puiden lukumäärä (l)")
+pesatiedot_muu_pit_l	("Virheellinen puiden keskipituus (l)")
+pesatiedot_tarkastus_vuosi	("Virheellinen vuosi")
+pesatiedot_tarkastus_tark_pvm	("Virheellinen päivämäärä")
+pesatiedot_tark_tunti	("Virheellinen kellonaika")
+pesatiedot_munia_lkm	("Virheellinen munien lukumäärä")
+pesatiedot_kuoriutumattomia_lkm	("Virheellinen kuoriutumattomien lukumäärä")
+pesatiedot_kuolleita_lkm	("Virheellinen kuolleiden lukumäärä")
+pesatiedot_elavia_lkm	("Virheellinen elävien lukumäärä")
+pesatiedot_reng_poik_lkm	("Virheellinen rengastusikäisten lukumäärä")
+pesatiedot_lentopoik_lkm	("Virheellinen lentopoikasten lukumäärä")
+pesatiedot_k_rengas_vasen	("Virheellinen renkaan tunnus")
+pesatiedot_k_rengas_oikea	("Virheellinen renkaan tunnus")
+pesatiedot_n_rengas_vasen	("Virheellinen renkaan tunnus")
+pesatiedot_n_rengas_oikea	("Virheellinen renkaan tunnus")
+pesatiedot_rengas_vasen	("Virheellinen renkaan tunnus")
+pesatiedot_rengas_oikea	("Virheellinen renkaan tunnus")
+pesatiedot_poikanen1_rengas_vasen	("Virheellinen renkaan tunnus")
+pesatiedot_poikanen1_rengas_oikea	("Virheellinen renkaan tunnus")
+pesatiedot_poikanen2_rengas_vasen	("Virheellinen renkaan tunnus")
+pesatiedot_poikanen2_rengas_oikea	("Virheellinen renkaan tunnus")
+pesatiedot_poikanen3_rengas_vasen	("Virheellinen renkaan tunnus")
+pesatiedot_poikanen3_rengas_oikea	("Virheellinen renkaan tunnus")
+pesatiedot_poikanen4_rengas_vasen	("Virheellinen renkaan tunnus")
+pesatiedot_poikanen4_rengas_oikea	("Virheellinen renkaan tunnus")
+pesatiedot_poikanen1_siipi_pituus	("Virheellinen siiven pituus")
+pesatiedot_poikanen2_siipi_pituus	("Virheellinen siiven pituus")
+pesatiedot_poikanen3_siipi_pituus	("Virheellinen siiven pituus")
+pesatiedot_poikanen4_siipi_pituus	("Virheellinen siiven pituus")

+pesatiedot_poikanen1_nilkka_max	("Virheellinen nilkan maksimipaksuus")
+pesatiedot_poikanen1_nilkka_min	("Virheellinen nilkan minimipaksuus")
+pesatiedot_poikanen2_nilkka_max	("Virheellinen nilkan maksimipaksuus")
+pesatiedot_poikanen2_nilkka_min	("Virheellinen nilkan minimipaksuus")
+pesatiedot_poikanen3_nilkka_max	("Virheellinen nilkan maksimipaksuus")
+pesatiedot_poikanen3_nilkka_min	("Virheellinen nilkan minimipaksuus")
+pesatiedot_poikanen4_nilkka_max	("Virheellinen nilkan maksimipaksuus")
+pesatiedot_poikanen4_nilkka_min	("Virheellinen nilkan minimipaksuus")
+pesatiedot_poikanen1_nokka_pituus	("Virheellinen nokan pituus")
+pesatiedot_poikanen1_nokka_tyvi	("Virheellinen nokan korkeus")
+pesatiedot_poikanen2_nokka_pituus	("Virheellinen nokan pituus")
+pesatiedot_poikanen2_nokka_tyvi	("Virheellinen nokan korkeus")
+pesatiedot_poikanen3_nokka_pituus	("Virheellinen nokan pituus")
+pesatiedot_poikanen3_nokka_tyvi	("Virheellinen nokan korkeus")
+pesatiedot_poikanen4_nokka_pituus	("Virheellinen nokan pituus")
+pesatiedot_poikanen4_nokka_tyvi	("Virheellinen nokan korkeus")
+pesatiedot_poikanen1_paino	("Virheellinen paino")
+pesatiedot_poikanen2_paino	("Virheellinen paino")
+pesatiedot_poikanen3_paino	("Virheellinen paino")
+pesatiedot_poikanen4_paino	("Virheellinen paino")
+pesatiedot_nayte_m	("Virheellinen munien lukumäärä")
+pesatiedot_nayte_p	("Virheellinen kuolleiden poikasten lukumäärä")
+pesatiedot_nayte_a	("Virheellinen kuolleiden aikuisten lukumäärä")
+pesatiedot_nayte_r	("Virheellinen saalisnäytteiden lukumäärä")
+text	
+pesatiedot_title	("PESÄTIEDOT")
+pesatiedot_tark_pvm	("Tarkastuspäivä*:")
+pesatiedot_ddlist_tark_pvm_tark	("Päivämäärän tarkkuus:")
+pesatiedot_kirj_pvm	("Kirjauspäivä:")
+pesatiedot_muutos_pvm	("Muutospäivä:")
+pesatiedot_tarkastaja1_sukunimi	("Tarkastaja 1: Sukunimi:")
+pesatiedot_tarkastaja1_etunimi	("Etunimi:")
+pesatiedot_tarkastaja1_id	("Id*:")
+pesatiedot_tarkastaja2_sukunimi	("Tarkastaja 2: Sukunimi:")
+pesatiedot_tarkastaja2_etunimi	("Etunimi:")
+pesatiedot_tarkastaja2_id	("Id:")
+pesatiedot_ddlist_kunta	("Kunta*:")
+pesatiedot_tarkka_sijainti	("Kylä, saari tms.:")
+pesatiedot_reviiri_nimi	("Reviirin nimi*:")
+pesatiedot_pesa_nimi	("Pesän nimi*:")
+pesatiedot_pesa_id	("Id:")
+pesatiedot_vanha_pesanro	("Pesän tunnus ennen tietojärjestelmää")
+pesatiedot_koord	("Koordinaatit*:")
+pesatiedot_ddlist_koord_mittaus	("Mittaustapa*:")
+pesatiedot_radiobutton_yht	("")
+pesatiedot_radiobutton_aste	("")
+pesatiedot_koord_yht	("Yhtenäis:")
+pesatiedot_yht_pituus	("Pituus 3")
+pesatiedot_yht_leveys	("Leveys")
+pesatiedot_koord_ast	("Aste:")
+pesatiedot_ast_pituus	("Pituus")
+pesatiedot_ast_leveys	("Leveys")
+pesatiedot_rak_vuosi	("Rakentamivuosi:")
+pesatiedot_ddlist_rak_vuosi_tark	("Tarkkuus:")
+pesatiedot_loyt_vuosi	("Löytymivuosi:")
+pesatiedot_rak_laji	("Pesän rakentanut laji:")
+pesatiedot_r_taulu_pvm	("Kiinnityspäivä:")
+pesatiedot_ddlist_r_taulu_kieli	("Kieli:")
+pesatiedot_r_taulu_nro	("Taulun nro:")
+pesatiedot_ddlist_puulaji	("Pesäpuun puulaji:")
+pesatiedot_ddlist_elavyys	("Elävyys:")
+pesatiedot_ddlist_pesa_kunto	("Pesän kunto:")
+pesatiedot_ddlist_sijainti	("Pesän sijainti")
+pesatiedot_pesa_kommentti	("Kommentti pesän sijaintiin liittyen:")
+pesatiedot_rauh_title	("RAUHOITUSTIEDOT")
+pesatiedot_palsta_rauh_pvm	("Tietojen päivämäärä*:")
+pesatiedot_ddlist_palsta_rauhoitus	("Pesän palstan rauhoitusilanne:")
+pesatiedot_rauh_aika_alku	("Rauhoituksen alkupvm:")
+pesatiedot_rauh_aika_loppu	("Rauhoituksen loppupvm:")
+pesatiedot_suojelualue	("Jos pesä suojelualueella, niin suojelualueen virallinen nimi:")
+pesatiedot_ddlist_palsta_omistaja	("Pesän palstan omistaja:")
+pesatiedot_omist_kommentti	("Kommentti:")
+pesatiedot_uhat	("Havaitut uhkatekijät:")
+pesatiedot_pesa_title	("PESÄPUUN JA PESÄN MITTOJA")
+pesatiedot_pesa_mit_pvm	("Mittauspäivämäärä:")
+pesatiedot_korkeus	("Puun korkeus:")
+pesatiedot_ddlist_korkeus_tark	("Tarkkuus:")
+pesatiedot_et_maasta	("Pesän yläpinnan etäisyys maasta:")
+pesatiedot_ddlist_et_maasta_tark	("Tarkkuus:")
+pesatiedot_et_latvasta	("Latvasta:")
+pesatiedot_ddlist_et_latvasta_tark	("Tarkkuus:")

+pesatiedot_tyvi	("130 cm maasta:")
+pesatiedot_tyviymparys	("Rungon ympärys:")
+pesatiedot_tyvihalkaisija	("Halkaisija:")
+pesatiedot_ddlist_tyvihai_tark	("Tarkkuus:")
+pesatiedot_latva	("Välittömästi pesän alla:")
+pesatiedot_latvaymparys	("Rungon ympärys:")
+pesatiedot_latvahalkaisija	("Halkaisija:")
+pesatiedot_ddlist_latvahai_tark	("Tarkkuus:")
+pesatiedot_pesa_korkeus	("Pesän korkeus:")
+pesatiedot_pesa_halk_max	("Pesän pinnan suurin halkaisija")
+pesatiedot_pesa_halk_min	("ja pienin halkaisija")
+pesatiedot_pesa_ymparisto_title	("PESÄN YMPÄRISTÖN TIETOJA")
+pesatiedot_pesa_ymparisto_etaisyydet_title	("Etäisyydet pesästä:")
+pesatiedot_et_meri	("Meren rantaan:")
+pesatiedot_et_jarvi	("Järven tai lammen rantaan:")
+pesatiedot_et_avosu	("Avosuon reunaan:")
+pesatiedot_et_as	("(Kesä)asuntoon:")
+pesatiedot_et_tie	("Autolla ajettavaan tiehen:")
+pesatiedot_et_moottorikelkka	("Moottorikelkkareittiin:")
+pesatiedot_et_talvitie	("Talvitiehen:")
+pesatiedot_et_kalaviljely	("Kalanviljelylaitokseen:")
+pesatiedot_et_ilmajohto	("Ilmajohtoon:")
+pesatiedot_et_viljapelto	("Viljeltyyn peltoon:")
+pesatiedot_et_avohakkuu	("Avohakkuun/siemenpuuston reunaan:")
+pesatiedot_et_lahipuu	("Sopivaan toiseen pesäpuuhun:")
+pesatiedot_pesa_ymparisto_sijainti_title	("Pesän sijainti:")
+pesatiedot_ddlist_saari_tyyppi	("Saarityyppi:")
+pesatiedot_ddlist_autoyhteys")	("Onko saareen tieyhteys (myös autoja kuljettavalla lautalla tms):")
+pesatiedot_as_lkm_1000	("Asuintalojen ja asuntojen määrä 1000m säteellä:")
+pesatiedot_as_lkm_500	("ja 500m säteellä:")
+pesatiedot_ddlist_pesan_nakyyvyys	("Pesän näkyvyys maastossa/vesiltä:")
+pesatiedot_pesa_ymparisto_puusto_title	("Puuston relaskooppimittaukset:")
+pesatiedot_puusto_p	("25 m pohjoiseen")
+pesatiedot_puusto_i	("25 m itään")
+pesatiedot_puusto_e	("25 m etelään")
+pesatiedot_puusto_l	("25 m länteen")
+pesatiedot_puusto_p_lkm	("lkm")
+pesatiedot_puusto_p_kskipituus	("keskipituus")
+pesatiedot_puusto_i_lkm	("lkm")
+pesatiedot_puusto_i_kskipituus	("keskipituus")
+pesatiedot_puusto_e_lkm	("lkm")
+pesatiedot_puusto_e_kskipituus	("keskipituus")
+pesatiedot_puusto_l_lkm	("lkm")
+pesatiedot_puusto_l_kskipituus	("keskipituus")
+pesatiedot_mannyt	("Männyt")
+pesatiedot_kuuset	("Kuuset")
+pesatiedot_muut	("Muut")
+pesatiedot_ddlist_puusto	("Puustotyyppi:")
+pesatiedot_ddlist_puusto_kasittely	("Käsittelyaste:")
+pesatiedot_ddlist_puusto_ika	("Puuston ikä:")
+pesatiedot_ddlist_maastotyyppi	("Maastotyyppi:")
+pesatiedot_valokuva	("Pesän ympäristöstä valokuva eläinmuseossa:")
+pesatiedot_tarkastus_title	("TARKASTUSTIEDOT")
+pesatiedot_tarkastus_vuosi	("Vuosi:")
+pesatiedot_tarkastus_tarkastaja1	("Tarkastaja 1:")
+pesatiedot_tarkastus_tarkastaja1_sukunimi	("Sukunimi:")
+pesatiedot_tarkastus_tarkastaja1_etunimi	("Etunimi:")
+pesatiedot_tarkastus_tarkastaja1_id	("Rengastaja/havainnojanro:")
+pesatiedot_tarkastus_tarkastaja2	("Tarkastaja 2:")
+pesatiedot_tarkastus_tarkastaja2_sukunimi	("Sukunimi:")
+pesatiedot_tarkastus_tarkastaja2_etunimi	("Etunimi:")
+pesatiedot_tarkastus_tarkastaja2_id	("Rengastaja/havainnojanro:")
+pesatiedot_tarkastus_tark_pvm	("Tarkastuspäivä*:")
+pesatiedot_ddlist_tarkastus_tark_pvm_tark	("Tarkkuus:")
+pesatiedot_tark_tunti	("Kellonaika:")
+pesatiedot_ddlist_tark_tapa	("Tarkastustapa:")
+pesatiedot_ddlist_muulaji	("Pesivä laji, jos ei merikotka:")
+pesatiedot_ddlist_pesimistulos	("Pesimistulos")
+pesatiedot_ddlist_pesimist_tark	("Tarkkuus:")
+pesatiedot_ddlist_nahdyt_merkit	("Nähdyt pesinnän merkit:")
+pesatiedot_ddlist_epaonni_syy	("Epäonnistumisen syy:")
+pesatiedot_pesimist_kommentti	("Pesimistuloksen kommentti:")
+pesatiedot_ddlist_pesa_kunto	("Pesän kunto:")
+pesatiedot_ddlist_pesa_merkit	("Merkit pesän ympärillä")
+pesatiedot_munia_lkm	("Munien lopullinen määrä")
+pesatiedot_kuoriutumattomia_lkm	("Kuoriutumattomia munia:")
+pesatiedot_kuolleita_lkm	("Kuolleita poikasia:")
+pesatiedot_elavia_lkm	("Eläviä poikasia:")
+pesatiedot_reng_poik_lkm	("Rengastusikäisiä poikasia")
+pesatiedot_lentopoik_lkm	("Lentopoikasia:")

+pesatiedot_aikuiset_title	("TIEDOT AIKUISISTA")
+pesatiedot_ddlist_aikuisia_lkm	("Lukumäärä")
+pesatiedot_koiras	("Koiras:")
+pesatiedot_k_rengas_v	("Rengas vasen:")
+pesatiedot_k_rengas_v_tunnus	("Tunnus:")
+pesatiedot_ddlist_k_rengas_v_vari	("Väri:")
+pesatiedot_k_rengas_o	("Rengas oikea:")
+pesatiedot_k_rengas_o_tunnus	("Tunnus:")
+pesatiedot_ddlist_k_rengas_o_vari	("Väri:")
+pesatiedot_k_rengas	("Rengas nähty:")
+pesatiedot_naaras	("Naaras:")
+pesatiedot_n_rengas_v	("Rengas vasen:")
+pesatiedot_n_rengas_v_tunnus	("Tunnus:")
+pesatiedot_ddlist_n_rengas_v_vari	("Väri:")
+pesatiedot_n_rengas_o	("Rengas oikea:")
+pesatiedot_n_rengas_o_tunnus	("Tunnus:")
+pesatiedot_ddlist_n_rengas_o_vari	("Väri:")
+pesatiedot_n_rengas	("Rengas nähty:")
+pesatiedot_koiras_tai_naaras	("Koiras tai naaras:")
+pesatiedot_rengas_v	("Rengas vasen:")
+pesatiedot_rengas_v_tunnus	("Tunnus:")
+pesatiedot_ddlist_rengas_v_vari	("Väri:")
+pesatiedot_rengas_o	("Rengas oikea:")
+pesatiedot_rengas_o_tunnus	("Tunnus:")
+pesatiedot_ddlist_rengas_o_vari	("Väri:")
+pesatiedot_rengas	("Rengas nähty:")
+pesatiedot_poikaset_title	("TIEDOT POIKASISTA")
+pesatiedot_poikanen	("Poikanen")
+pesatiedot_poikanen1	("1")
+pesatiedot_poikanen2	("2")
+pesatiedot_poikanen3	("3")
+pesatiedot_poikanen4	("4")
+pesatiedot_ddlist_sukupuoli	("Sukupuoli")
+pesatiedot_rengas_vasen	("Vasemman jalan renkaan tunnus")
+pesatiedot_ddlist_varit_vasen	("Vasemman jalan renkaan väri")
+pesatiedot_rengas_oikea	("Oikean jalan renkaan tunnus")
+pesatiedot_ddlist_varit_oikea	("Oikean jalan renkaan väri")
+pesatiedot_siipi_pituus	("Siiven pituus")
+pesatiedot_ddlist_siipi_pituus_m	("Siiven mittausmenetelmä")
+pesatiedot_nilkka_max	("Nilkan paksuus maksimi")
+pesatiedot_nilkka_min	("Nilkan paksuus minimi")
+pesatiedot_nokka_pituus	("Nokan pituus")
+pesatiedot_nokka_tyvi	("Nokan korkeus tyvestä")
+pesatiedot_paino	("Paino")
+pesatiedot_ddlist_kupu	("Kupu")
+pesatiedot_ddlist_dna_nayte	("Höyhen/verinäyte otettu")
+pesatiedot_naytteet_title	("NÄYTTEITÄ LÄHETETTY")
+pesatiedot_nayte_i	("Merikotkan sulkia ja höyheniä:")
+pesatiedot_nayte_m	("Munia:")
+pesatiedot_nayte_s	("Munansiruja:")
+pesatiedot_nayte_p	("Kuolleita poikasia:")
+pesatiedot_nayte_a	("Kuolleita aikuisia:")
+pesatiedot_nayte_r	("Saalisnäytteitä:")
+pesatiedot_nayte_o	("Oksennuspalloja")
+pesatiedot_submitbutton1_title	("Lisää")
+pesatiedot_submitbutton2_title	("Muuta")
+ddlist	
+ddlist_pesatiedot_tark_pvm_tark	
+ddlist_pesatiedot_tark_pvm_tark_selected	
+ddlist_pesatiedot_kunta_id	
+ddlist_pesatiedot_kunta_id_selected	
+ddlist_pesatiedot_koord_mittaus	
+ddlist_pesatiedot_koord_mittaus_selected	
+ddlist_pesatiedot_rak_vuosi_tarkkuus	
+ddlist_pesatiedot_rak_vuosi_tarkkuus_selected	
+ddlist_pesatiedot_r_taulu_kieli	
+ddlist_pesatiedot_r_taulu_kieli_selected	
+ddlist_pesatiedot_puulaji	
+ddlist_pesatiedot_puulaji_selected	
+ddlist_pesatiedot_elavyys	
+ddlist_pesatiedot_elavyys_selected	
+ddlist_pesatiedot_pesa_kunto	
+ddlist_pesatiedot_pesa_kunto_selected	
+ddlist_pesatiedot_sijainti	
+ddlist_pesatiedot_sijainti_selected	
+ddlist_pesatiedot_palsta_rauhoitus	
+ddlist_pesatiedot_palsta_rauhoitus_selected	
+ddlist_pesatiedot_palsta_omistaja	
+ddlist_pesatiedot_palsta_omistaja_selected	
+ddlist_pesatiedot_korkeus_tark	
+ddlist_pesatiedot_korkeus_tark_selected	

+ddlist_pesatiedot_et_maasta_tark
+ddlist_pesatiedot_et_maasta_tark_selected
+ddlist_pesatiedot_et_latvasta_tark
+ddlist_pesatiedot_et_latvasta_tark_selected
+ddlist_pesatiedot_saari_tyyppi
+ddlist_pesatiedot_saari_tyyppi_selected
+ddlist_pesatiedot_tieyhitys
+ddlist_pesatiedot_tieyhitys_selected
+ddlist_pesatiedot_nakvyvyys
+ddlist_pesatiedot_nakvyvyys_selected
+ddlist_pesatiedot_puusto
+ddlist_pesatiedot_puusto_selected
+ddlist_pesatiedot_puusto_kasittely
+ddlist_pesatiedot_puusto_kasittely_selected
+ddlist_pesatiedot_puusto_ika
+ddlist_pesatiedot_puusto_ika_selected
+ddlist_pesatiedot_maastotyyppi
+ddlist_pesatiedot_maastotyyppi_selected
+ddlist_pesatiedot_tarkastus_tark_pvm_tarkkuus
+ddlist_pesatiedot_tarkastus_tark_pvm_tarkkuus_selected
+ddlist_pesatiedot_tark_tapa
+ddlist_pesatiedot_tark_tapa_selected
+ddlist_pesatiedot_muulaji
+ddlist_pesatiedot_muulaji_selected
+ddlist_pesatiedot_pesimistulos
+ddlist_pesatiedot_pesimistulos_selected
+ddlist_pesatiedot_pesimist_tark
+ddlist_pesatiedot_pesimist_tark_selected
+ddlist_pesatiedot_nahdyt_merkit
+ddlist_pesatiedot_nahdyt_merkit_selected
+ddlist_pesatiedot_epaonni_syy
+ddlist_pesatiedot_epaonni_syy_selected
+ddlist_pesatiedot_epaonni_tark
+ddlist_pesatiedot_epaonni_tark_selected
+ddlist_pesatiedot_pesa_merkit
+ddlist_pesatiedot_pesa_merkit_selected
+ddlist_pesatiedot_aikuisia_lkm
+ddlist_pesatiedot_aikuisia_lkm_selected
+ddlist_pesatiedot_k_rengas_v_vari
+ddlist_pesatiedot_k_rengas_v_vari_selected
+ddlist_pesatiedot_k_rengas_o_vari
+ddlist_pesatiedot_k_rengas_o_vari_selected
+ddlist_pesatiedot_n_rengas_v_vari
+ddlist_pesatiedot_n_rengas_v_vari_selected
+ddlist_pesatiedot_n_rengas_o_vari
+ddlist_pesatiedot_n_rengas_o_vari_selected
+ddlist_pesatiedot_rengas_v_vari
+ddlist_pesatiedot_rengas_v_vari_selected
+ddlist_pesatiedot_rengas_o_vari
+ddlist_pesatiedot_rengas_o_vari_selected
+ddlist_pesatiedot_poikanen1_sukupuoli
+ddlist_pesatiedot_poikanen1_sukupuoli_selected
+ddlist_pesatiedot_poikanen2_sukupuoli
+ddlist_pesatiedot_poikanen2_sukupuoli_selected
+ddlist_pesatiedot_poikanen3_sukupuoli
+ddlist_pesatiedot_poikanen3_sukupuoli_selected
+ddlist_pesatiedot_poikanen4_sukupuoli
+ddlist_pesatiedot_poikanen4_sukupuoli_selected
+ddlist_pesatiedot_poikanen1_varit_vasen
+ddlist_pesatiedot_poikanen1_varit_vasen_selected
+ddlist_pesatiedot_poikanen1_varit_oikea
+ddlist_pesatiedot_poikanen1_varit_oikea_selected
+ddlist_pesatiedot_poikanen2_varit_vasen
+ddlist_pesatiedot_poikanen2_varit_vasen_selected
+ddlist_pesatiedot_poikanen2_varit_oikea
+ddlist_pesatiedot_poikanen2_varit_oikea_selected
+ddlist_pesatiedot_poikanen3_varit_vasen
+ddlist_pesatiedot_poikanen3_varit_vasen_selected
+ddlist_pesatiedot_poikanen3_varit_oikea
+ddlist_pesatiedot_poikanen3_varit_oikea_selected
+ddlist_pesatiedot_poikanen4_varit_vasen
+ddlist_pesatiedot_poikanen4_varit_vasen_selected
+ddlist_pesatiedot_poikanen4_varit_oikea
+ddlist_pesatiedot_poikanen4_varit_oikea_selected
+ddlist_pesatiedot_poikanen1_siipi_pituus_m
+ddlist_pesatiedot_poikanen1_siipi_pituus_m_selected
+ddlist_pesatiedot_poikanen2_siipi_pituus_m
+ddlist_pesatiedot_poikanen2_siipi_pituus_m_selected
+ddlist_pesatiedot_poikanen3_siipi_pituus_m
+ddlist_pesatiedot_poikanen3_siipi_pituus_m_selected
+ddlist_pesatiedot_poikanen4_siipi_pituus_m

```

+ddlist_pesatiedot_poikanen4_siipi_pituus_m_selected
+ddlist_pesatiedot_poikanen1_kupu
+ddlist_pesatiedot_poikanen1_kupu_selected
+ddlist_pesatiedot_poikanen2_kupu
+ddlist_pesatiedot_poikanen2_kupu_selected
+ddlist_pesatiedot_poikanen3_kupu
+ddlist_pesatiedot_poikanen3_kupu_selected
+ddlist_pesatiedot_poikanen4_kupu
+ddlist_pesatiedot_poikanen4_kupu_selected
+ddlist_pesatiedot_poikanen1_dna_nayte
+ddlist_pesatiedot_poikanen1_dna_nayte_selected
+ddlist_pesatiedot_poikanen2_dna_nayte
+ddlist_pesatiedot_poikanen2_dna_nayte_selected
+ddlist_pesatiedot_poikanen3_dna_nayte
+ddlist_pesatiedot_poikanen3_dna_nayte_selected
+ddlist_pesatiedot_poikanen4_dna_nayte
+ddlist_pesatiedot_poikanen4_dna_nayteseleced
+data[]
+pesatarkastus
  +p_tarkastus_id
  +pesa_id
  +tarkastaja1_id
  +tarkastaja2_id
  +tark_pvm
  |
  |           +paiva
  |           +kuukausi
  |           +vuosi
  +kirj_pvm
  |
  |           +paiva
  |           +kuukausi
  |           +vuosi
  +muutos_pvm
  |
  |           +paiva
  |           +kuukausi
  |           +vuosi
  +tark_pvm_tark
  +tark_tunti
  +as_lkm_1000
  +as_lkm_500
  +et_as
  +et_tie
  +et_kalaviljely
  +et_moottorikelkka
  +et_talvitie
  +tark_tapa
  +pesa_kunto
  +pesa_merkit
  +aikuisia_lkm
  +k_rengas_vasen
  +k_rengas_v_vari
  +k_rengas_oikea
  +k_rengas_o_vari
  +n_rengas_vasen
  +n_rengas_v_vari
  +n_rengas_oikea
  +n_rengas_o_vari
  +k_rengas
  +n_rengas
  +rengas
  +rengas_vasen
  +rengas_v_vari
  +rengas_oikea
  rengas_o_vari
  +munia_lkm
  +munia_pvm
  |
  |           +paiva
  |           +kuukausi
  |           +vuosi
  +kuoriutumattomia_lkm
  +elavia_lkm
  +kuoriutumispaiava
  +kuolleita_lkm
  +lentopoik_lkm
  +reng_poik_lkm
  +pesimistulos
  +pesimist_tark
  +nahdyt_merkit
  +epaonni_syy
  +epaonni_tark
  +muulaji

```

```

+pesa_korkeus
+nahdyt_merkit
+pesa_halk_max
+pesa_halk_min
+uhat
+nayte_i
+nayte_m
+nayte_s
+nayte_p
+nayte_a
+nayte_r
+nayteo
+pesa_kommentti
+pesimist_kommentti
+pesavakio
+pesa_id
+tarkastaja1_id
+tarkastaja2_id
+kirj_pvm
|           +paiva
|           +kuukausi
|           +vuosi
+muutos_pvm
|           +paiva
|           +kuukausi
|           +vuosi
+tark_pvm
|           +paiva
|           +kuukausi
|           +vuosi
+tark_pvm_tark
+pesanimi
+vanha_pesanro
+tarkka_sijainti
+koord_mittaus
+koord_tyyppi
+koord_tark
+yht_leveys
+yht_pituus
+ast_leveys
|           +aste
|           +min
|           +sek
+ast_pituus
|           +aste
|           +min
|           +sek
+et_meri
+et_jarvi
+rak_vuosi
+rak_vuosi_tark
+rak_laji
+loyt_vuosi
+tuhoutumisvuosi
+puulaji
+valokuva
+r_taulu_nro
+r_taulu_kieli
+r_taulu_pvm
|           +paiva
|           +kuukausi
|           +vuosi
+kommentti
+pesamuuttuva
+pesamuuttuva_id
+pesa_id
+tarkastaja1_id
+tarkastaja2_id
+alku_pvm
|           +paiva
|           +kuukausi
|           +vuosi
+loppu_pvm
|           +paiva
|           +kuukausi
|           +vuosi
+kirj_pvm
|           +paiva
|           +kuukausi
|           +vuosi
+muutos_pvm

```

```

|           +paiva
|           +kuukausi
|           +vuosi
+pesa_mit_pvm
|           +paiva
|           +kuukausi
|           +vuosi
+elavyys
+korkeus
+korkeus_tark
+maastotyyppi
+saari_tyyppi
+autoyhteys
+puusto
+puusto_kasittely
+puusto_ika
+ymp_mit_pvm
|           +paiva
|           +kuukausi
|           +vuosi
+manty_lkm_p
+manty_pit_p
+kuusi_lkm_p
+kuusi_pit_p
+muu_lkm_p
+muu_pit_p
+manty_lkm_i
+manty_pit_i
+kuusi_lkm_i
+kuusi_pit_i
+muu_lkm_i
+muu_pit_i
+manty_lkm_e
+manty_pit_e
+kuusi_lkm_e
+kuusi_pit_e
+muu_lkm_e
+muu_pit_e
+manty_lkm_l
+manty_pit_l
+kuusi_lkm_l
+kuusi_pit_l
+muu_lkm_l
+muu_pit_l
+et_lahipuu
+et_avosuo
+et_avohakkuu
+et_viljapelto
+et_ilmajohto
+pesan_nakvyys
+palsta_rauh_pvm
|           +paiva
|           +kuukausi
|           +vuosi
+palsta_rauhoitus
+rauh_aika_alku
|           +paiva
|           +kuukausi
|           +vuosi
+rauh_aika_loppu
|           +paiva
|           +kuukausi
|           +vuosi
+palsta_omistaja
+suojelualue
+sijainti
+et_maasta
+et_maasta_tark
+et_latvasta
+et_latvasta_tark
+tyvihalkaisija
+tyvihal_tark
+tyviymparys
+latvahalkaisija
+latvahal_tark
+latvaymparys
+rauh_kommentti
+omist_kommentti
+tarkastaja
+tarkastaja_id
+etunimi

```

```

|
|+poikanen[]
|
|   +sukunimi
|   +poikanen_id
|   +tarkastus_id
|   +mittaaja_id
|   +mittaus_pvm
|   |           +paiva
|   |           +kuukausi
|   |           +vuosi
|   +kirj_pvm
|   |           +paiva
|   |           +kuukausi
|   |           +vuosi
|   +muutos_pvm
|   |           +paiva
|   |           +kuukausi
|   |           +vuosi
|   +rengas_oikea
|   +rengas_vasen
|   +varit_oikea
|   +varit_vasen
|   +sukupuoli
|   +siipi_pituus
|   +siipi_pituus_m
|   +nilkka_min
|   +nilkka_max
|   +nokka_pituus
|   +nokka_tyvi
|   +paino
|   +kupu
|   +poikasen_ika_
|   +dna_nayte
|   +kommentti
|
|+kunta
|   +kunta_nimi
|
|+laji
|   +laji_id

```

`Pesä.ftl` toimii näyttöjen Uusi pesä, Uusi tarkastus ja Pesätiedot pohjana. Datamallin solmun `text` alla on kaikki näytön staattiset tekstit. Suurin osa teksteistä ladataan jokaiselle kolmelle sivulle, mutta esim. sivun yläosan otsikkoja linkkitiedot vaihtelevat sivujen välillä.

4.14.1.5myrkky.ftl

```

(root)
|
|+error
|   +myrkky_kerayspvm          ("Virheellinen päivämäärä")
|   +myrkky_analyysipvm       ("Virheellinen päivämäärä")
|   +myrkky_muna_pit          ("Virheellinen pituus")
|   +myrkky_muna_lev          ("Virheellinen leveys")
|   +myrkky_muna_paks_ei_k    ("Virheellinen kuoren paksuus")
|   +myrkky_muna_paks_k      ("Virheellinen kuoren paksuus")
|   +myrkky_muna_paino        ("Virheellinen aino")
|   +myrkky_muna_paino_tark   ("Virheellinen painon tarkkuus")
|   +myrkky_muna_sisalto      ("Virheellinen sisällön laatu")
|   +myrkky_muna_sis_vari     ("Virheellinen sisällön väri")
|   +myrkky_muna_sikio        ("Virheellinen sikiön koko")
|   +myrkky_ddt               ("Virheellinen DDT arvo")
|   +myrkky_ddd               ("Virheellinen DDD arvo")
|   +myrkky_dde               ("Virheellinen DDE arvo")
|   +myrkky_pcb               ("Virheellinen PCB arvo")
|   +myrkky_hg                ("Virheellinen HG arvo")
|
|+text
|   +myrkky_title              ("Haliaeetus - Myrkky")
|   +myrkky_myrkkyt_link      ("MYRKKYT")
|   +myrkky_saaliit_link      ("SAALIIT")
|   +myrkky_historia_link     ("HISTORIA")
|   +myrkky_myrkkyt_title     ("MYRKKYT")
|   +myrkky_pesanimi          ("Pesä")
|   +myrkky_keraaja           ("Kerääjä*")
|   +myrkky_kerayspvm         ("Keräyspäivä*")
|   +myrkky_analysioija       ("Analysoija*")
|   +myrkky_analyysipvm      ("Analyysipäivä*")

```

```

+myrkky_muna_title          ("MUNA")
+myrkky_muna_pit           ("Pituus")
+myrkky_muna_lev           ("Leveys")
+myrkky_muna_paks_ei_k     ("Kuoren paksuus ilman kalvoa")
+myrkky_muna_paks_k       ("Kuoren paksuus kalvoineen")
+myrkky_muna_paino        ("Paino")
+myrkky_muna_paino_tark   ("Painon tarkkuus")
+myrkky_muna_sisalto      ("Sisällön laatu")
+myrkky_muna_sis_vari     ("Sisällön väri")
+myrkky_muna_sikio        ("Sikiön koko")
+myrkky_pitoisuudet_title ("PITOISUUDET")
//-----näämä on varmaan kaikille kielille samat
+myrkky_ddt                ("DDT")
+myrkky_ddd                ("DDD")
+myrkky_dde                ("DDE")
+myrkky_pcb                ("PCB")
+myrkky_hg                 ("HG")
//-----
+myrkky_anal_menet        ("Analyysimenetelmä")
+myrkky_kommentti         ("KOMMENTTI")
+myrkky_submit_title     ("TALLENNA")

+myrkky_ddlist_muna       ("muna") //tämä on teksti ddiistin sisälle.. --> muna 1, muna 2, muna 3
+ddlist
+ddlist_myrkky_muna[0...n] //tässä ddiistissä listattaisiin munat..
+ddlist_myrkky_muna_selected
+myrkky
+myrkky_id
+pesa_id
+keraaja_id
+kerays_pvm
+maarittaja_id
+analyysi_pvm
+muna_pit
+muna_lev
+muna_paks_ei_k
+muna_paks_k
+muna_paino
+muna_paino_tark
+muna_sisalto
+muna_sis_vari
+muna_sikio
+ddt
+ddd
+dde
+pcb
+hg
+anal_menet
+kommentti
+pesavakio
+pesanimi
+pesa_id
+tarkastaja
+tarkastaja_id
+etunimi
+sukunimi

```

myrkyt.ftl on pesän myrkytietojen ylläpitosivu.

4.14.1.6saalis.ftl

```

(root)
|
+haliURL                      ("https://db.cs.helsinki.fi/s/tunnus/")
|
+text
|
+saaliit_title                ("Saaliit")
+saaliit_pesä                 ("Pesä:")
+saaliit_keräystiedot        ("Keräystiedot")
+saaliit_kaikki              ("Kaikki")
+saaliit_uusi                 ("Uusi")
+saaliit_saaliit_keräyspäivittain_title ("Saaliit keräyspäivittain")

```

```

+saaliit_keräys_pvm      ("Keräyspäivä")
+saaliit_kerääjä        ("Kerääjä")
+saaliit_määrittäjä     ("Määrittäjä")
+saaliit_määrittäjä     ("Määrittäjä")
+saaliit_laji           ("Laji")
+saaliit_lkm            ("Lukumäärä")

+searchparameteres
+pesa_id

+data[...]
+saalis
+saalis_id
+pesa_id
+keraaja_id
+maarittaja_id
+kirj_pvm
+kerays_pvm
+maaritys_pvm
+laji_id
+laji_lkm
+kommentti

```

saalis.ftl pesän tallennettujen saalistietojen näyttösivu.

4.14.1.7 saalis_yllapito.ftl

```

(root)
+haliURL                  ("https://db.cs.helsinki.fi/s/tunnus/")
+error
+historia_alku           ("virheellinen alkuvuosi")
+historia_loppu         ("virheellinen loppuvuosi")
+text
+saaliit_title           ("Saaliit")
+saaliit_pesä            ("Pesä:")
+saaliit_keräystiedot    ("Keräystiedot")
+saaliit_kaikki          ("Kaikki") // Tarkoitus?
+saaliit_uusi            ("Uusi")
+saaliit_saaliit_keräyspäivittäin_title ("Saaliit keräyspäivittäin")
+saaliit_keräys_pvm     ("Keräyspäivä")
+saaliit_kerääjä        ("Kerääjä")
+saaliit_määrittäjä     ("Määrittäjä")
+saaliit_määrittäjä     ("Määrittäjä")
+saaliit_laji           ("Laji")
+saaliit_lkm            ("Lukumäärä")
+saaliit_keräyspaikka    ("Keräyspaikka")
+saaliit_submitbutton_muuta ("Muuta")
+saaliit_submitbutton_lisää ("Lisää")

+searchparameters
+pesa_id
+keräys_pvm

+data[...]
+saalis
+saalis_id
+pesa_id
+keraaja_id
+maarittaja_id
+kirj_pvm
+kerays_pvm
+maaritys_pvm
+laji_id
+laji_lkm
+kommentti

```

saalis_yllapito.ftl on pesän saalistietojen ylläpitosivu yhtä keräyspäivää kohti.

4.14.1.8historia.ftl

```

(root)
|
+halURL                               ("https://db.cs.helsinki.fi/s/tunnus/")
|
+error
|   +historia_alku                     ("virheellinen alkuvuosi")
|   +historia_loppu                   ("virheellinen loppuvuosi")
|
+text
|   +historia_title                   ("Historia")
|   +historia_pesä                    ("Pesä:")
|   +historia_kirj_pvm                ("Kirjauspäivä")
|   +historia_muutos_pvm             ("Muutospäivä")
|   +historia_alku                   ("Tunnetun asumisajan alkuvuosi:")
|   +historia_alku_tark               ("Alkuvuoden tarkkuus")
|   +historia_loppu                  ("Tunnetun asumisajan loppuvuosi")
|   +historia_loppu_tark             ("Alkuvuoden tarkkuus")
|   +historia_tiedon_lähde_title     ("Tiedon lähde")
|   +historia_kommentti_title        ("Kommentti")
|   +historia_submitbutton           ("Tallenna")
|
+searchparameters
|   +pesa_id
|
+ddlist
|   +ddlist_historia_alku_tark
|   +ddlist_historia_alku_tark_selected
|   +ddlist_historia_loppu_tark
|   +ddlist_historia_loppu_tark_selected
|
+data[...]
|   +historia
|   |   +kirj_pvm
|   |   +muutos_pvm
|   |   +alku
|   |   +alku_tark
|   |   +loppu
|   |   +loppu_tark
|   |   +tiedon_lähde
|   |   +kommentti
|   +pesävakio
|   +pesanimi

```

historia.ftl pesän historiallisten tietojen ylläpitosivu.

4.14.1.9reviirit

```

(root)
|
+error
|   +reviirit_reviiri_id              ("virheellinen reviiritunnus")
|   +reviirit_reviiri_nimi           ("virheellinen reviirinumero")
|   +reviirit_kunta_id               ("virheellinen kuntatunnus")
|   +reviirit_vuosi                  ("virheellinen vuosi")
|   +reviirit_vanha_reviirinro      ("virheellinen vanhassa reviirinumerossa")
|   +reviirit_kommentti              ("virheellinen kommentti")
|
+text
|   +reviirit_title                  ("Reviirit")
|   +reviirit_reviiri_id             ("id")
|   +reviirit_reviiri_nimi           ("Nimi")
|   +reviirit_kunta_id               ("Kunta")
|   +reviirit_vuosi                  ("Vuosi")
|   +reviirit_vanha_reviirinro      ("Vanha reviirinro")
|   +reviirit_kommentti              ("Kommentti")
|   +reviirit_submit_lisäätitle     ("Lisää :")
|   +reviirit_submit_muutatitle     ("Muuta :")
|
+ddlist
|   +ddlist_reviirit_reviiri_id[0...n]
|   +ddlist_reviirit_reviiri_id_selected
|
+data

```



```

+reviiri
|
|   +reviiri_id
|   +reviiri_nimi
|   +vanha_reviirinro
|   +kommentti
+kunta
|   +kunta_id
+vuosi
|   +vuosi

```

reviirit.ftl on reviirien lisäyssivu.

4.14.1.10kunnat.ftl

```

(root)
|
+error
|
|   +kunnat_kunta_tunnus      ("Virheellinen tunnus")
|   +kunnat_nimi             ("Virheellinen nimi")
|   +kunnat_des_leveys       ("Virheellinen leveys-koordinaatti")
|   +kunnat_des_pituus       ("Virheellinen pituus-koordinaatti")
|   +kunnat_kunta_sade       ("Virheellinen säde")
|   +kunnat_kirj_pvm         ("Virheellinen kirjauspäivämäärä")
|
+text
|
|   +kunnat_title             ("KUNNAT")
|   +kunnat_uusi_kunta_title  ("UUSI KUNTA")
|   +kunnat_kunta_tunnus     ("Tunnus*")
|   +kunnat_nimi             ("Nimi*")
|   +kunnat_suur_alue        ("Suuralue*")
|   +kunnat_ymp_keskus       ("Ympäristökeskus*")
|   +kunnat_keskipiste_title  ("Kunnan keskipiste:")
|   +kunnat_des_leveys       ("Leveys*")
|   +kunnat_des_pituus       ("Pituus*")
|   +kunnat_kunta_sade       ("Kunnan säde*")
|   +kunnat_kirj_pvm         ("Kirjauspäivämäärä")
|   +kunnat_submitbutton1    ("Lisää kunta")
|   +kunnat_kuntaliitokset_title ("KUNTALIITOKSET")
|   +kunnat_suuralueet_title  ("SUURALUEET")
|   +kunnat_submitbutton2    ("Hae kunnat")
|   +kunnat_liitettävät_title ("LIITETTÄVÄT KUNNAT")
|   +kunnat_submitbutton3    ("Yhdistä kunnat")
|
+ddlist
|
|   +kunnat_suur_alue
|   +kunnat_suur_alue_selected
|   +kunnat_ymp_keskus
|   +kunnat_ymp_keskus_selected
|
+searchparameters
|
|   +kunta
|       +suur_alue
|
+data[]
|
|   +kunta
|       +kunta_id
|       +kunta_nimi
|       +suur_alue
|       +kirj_pvm
|       +kunta_tunnus
|       +kunta_liitos
|       +ymp_keskus
|       +des_leveys
|       +des_pituus
|       +kunta_sade

```

kunnat.ftl on kuntatietojen ylläpitosivu, jolla voidaan luoda uusia kuntia ja tehdä kuntaliitoksia.

4.14.1.11tarkastajat.ftl

```

(root)
|
+error

```

```

|         +tarkastajat_etunimi          ("Virheellinen etunimi")
|         +tarkastajat_sukunimi        ("Virheellinen sukunimi")
|         +tarkastajat_takastaja_id    ("Virheellinen tarkastaja ID")
|
| +text
|
|         +tarkastajat_title            ("TARKASTAJAT")
|         +tarkastajat_header1         ("UUDEN TARKASTAJAN LISÄYS")
|         +tarkastajat_etunimi        ("Etunimi")
|         +tarkastajat_sukunimi       ("Sukunimi")
|         +tarkastajat_tarkastaja_id   ("Tarkastaja ID")
|         +tarkastajat_submitbutton1  ("Lisää")
|         +tarkastajat_header2        ("NIMEN MUUTTAMINEN")
|         +tarkastajat_submitbutton2  ("Muuta")
|
| +data[]
|
|         +tarkastaja
|         +tarkastaja_id
|         +etunimi
|         +sukunimi

```

tarkastajat.ftl on tarkastajatietojen ylläpitosivu, jolla voidaan lisätä tarkastajia ja muuttaa tarkastajien nimeä.

4.14.1.12 aputaulut.ftl

```

(root)
|
| +error
|
|         +aputaulut_koodi              ("Virhe: koodi on jo olemassa")
|
| +text
|
|         +aputaulut_title              ("APUTAULUT")
|         +aputaulut_ddlist_taulu      ("Taulu*:")
|         +aputaulut_ddlist_attribuutti ("Attribuutti*:")
|         +aputaulut_koodi             ("Koodi")
|         +aputaulut_selite            ("Selite")
|         +aputaulut_submitbutton1     ("Lisää")
|         +aputaulut_submitbutton2     ("Muuta")
|
| +ddlist
|
|         +ddlist_taulu
|         +ddlist_taulu_selected
|         +ddlist_attribuutti
|         +ddlist_attribuutti_selected
|
| data[]
|
|         +aputaulu
|         |
|         |         +taulu
|         |         +attribuutti
|         |         |
|         |         |         +koodi
|         |         |         +selite

```

aputaulut.ftl on aputaulujen ylläpitosivu, jolla voidaan lisätä aputauluun koodeja ja muuttaa koodien selitteitä.

4.14.1.13 raportit.ftl

```

(root)
|
| +error
|
|         +raportit_haku_alku           ("Virheellinen alkupäivämäärä")
|         +raportit_haku_loppu        ("Virheellinen loppupäivämäärä")
|
| +text
|
|         +raportit_title              ("Raportit")
|         +raportit_raporttityyppi_title ("Raporttityyppi")
|         +raportit_raporttiA         ("Yhteenveto suojeluviranomaisten käyttöön")
|         +raportit_raporttiB         ("Tarkastajien pesätarkastukset")
|         +raportit_raporttiC         ("Reviiri- ja pesäkohtaiset pesimistulokset ja kuoriutumistiedot")
|         +raportit_raporttiD         ("Kuoriutumispäivät")
|         +raportit_raporttiE         ("Yhteenveto kohdan D raportista")
|         +raportit_raporttiF         ("Pesimistulokset revireittäin")
|         +raportit_raporttiG         ("Puustot")

```

```

+raportit_raporttiH      ("Tarkat poikuekoot rengastushetkellä")
+raportit_raporttiI     ("Pesäpuiden elävyys")
+raportit_raporttiJ     ("Pesäpuiden puulajit luonnonpesissä")
+raportit_raporttiK     ("Pesäpuiden puulajit tekopesissä")
+raportit_raporttiL     ("Pesien sijainti")
+raportit_raporttiM     ("Puustojen käsittelyasteet")
+raportit_raporttiN     ("Pesien sijainnit pesäsaaren tyyppin mukaan")
+raportit_raporttiO     ("Pesien sijainnit maastotyyppin mukaan")
+raportit_raporttiP     ("Palstan rauhoitusaste ja pesimistulos")
+raportit_raporttiQ     ("Uusi Pesä")
+raportit_raporttiR     ("Vanha pesä")
+raportit_aikavali      ("Aikaväli")
+raportit_alueet        ("Alueet:")
+raportit_koko_maa      ("Koko maa:")
+raportit_suur_alue     ("Suuralue")
+raportit_ymp_keskus    ("Ympäristökeskus:")
+raportit_kunta_nimi    ("Kunta:")
+raportit_reviiri_nimi  ("Reviiri:")
+raportit_pesanimi      ("Pesä:")
+raportit_submitbutton1 ("Tallenna tiedostoon")

+ddlist
+ddlist_suur_alue
+ddlist_suur_alue_selected
+ddlist_ymp_keskus
+ddlist_ymp_keskus_selected

+searchparameters
+reviiri
+reviiri_nimi
+pesavakio
+pesanimi
+kunta
+kunta_nimi
+suur_alue
+ymp_keskus
+haku_alku
+haku_loppu
+koko_maa

```

raportit.ftl on raporttien tallennussivu.

4.14.1.14lajit.ftl

```

(root)
+error
+laji_laji_id           ("Virheellinen id")

+text
+laji_title             ("Haliaeetus - Laji")
+laji_lajit_title       ("Lajit")
+laji_muuttaminen_title ("LAJIN TIETOJEN MUUTTAMINEN")
+laji_laji_id           ("Laji ID")
+laji_kommentti        ("Kommentti")

+ddlist
+ddlist_laji_id
+ddlist_laji_id_selected

+laji
+laji_id
+kommentti

```

lajit.ftl on lajitietojen tallennussivu.

4.14.2 Properties-tiedostot

Properties-tiedostot ovat tekstitiedostoja, jotka sisältävät avain-arvoparin yhtäläisyysmerkillä erotettuna (esim. `language=Kieli`). Kukin avain-arvopari on tiedostossa omalla rivillään. Avaimissa ja yhtäläisyysmerkin ympärillä ei saa olla välilyöntimerkkejä, arvoissa sitä vastoin välilyönnit ovat sallittuja. Isot ja pienet kirjaimet ovat merkitseviä.

4.14.2.1 Kielituen tiedostot

Kielituki toteutetaan properties-tiedostojen avulla. Kielitukitiedostoihin sijoitetaan generoitavien html-sivujen staattiset tekstit, virheilmoitukset ja muut ilmoitukset. Tuettavat kielet ovat suomi, ruotsi ja englanti. Kullekin kielelle luodaan oma properties-tiedosto. Avaimet ovat kaikissa properties-tiedostoissa samat, mutta arvona tiedostossa on kyseisen kielen mukainen teksti merkkijonona. Haliaeetus-järjestelmässä avain on sama kuin template-tiedoston muuttuja, johon teksti kohdistetaan.

Haliaeetus-järjestelmässä kielituki käyttää tiedostoja:

```
haliResources_fi.properties
haliResources_sv.properties
haliResources_en.properties
```

Tiedostot nimetään seuraavan syntaksin mukaisesti:

```
haliResources_[ISO-639 kielikoodi].properties.
```

Alla on esimerkki kolmesta properties-tiedostosta suomeksi, ruotsiksi ja englanniksi. Esimerkeissä avaimet ovat englanniksi, mutta toteutusympäristössä ne ovat suomeksi, jolloin ne vastaavat template-tiedostoissa käytettäviä muuttujia.

Tiedosto `haliResources_fi.properties` sisältää tuen suomen kielelle.
Esimerkki tiedoston sisällöstä:

```
language=Kieli
compute=Laske
whiteTailedEagle=Merikotka
```

Tiedosto `haliResources_sv.properties` sisältää tuen ruotsin kielelle.
Esimerkki tiedoston sisällöstä:

```
language=Språk
compute=Räkna
whiteTailedEagle=Havsörn
```

Tiedosto `haliResources_en.properties` sisältää tuen englannin kielelle.
Esimerkki tiedoston sisällöstä:

```
language=Language
compute=Compute
whiteTailedEagle=White-tailed eagle
```

Ylläpitäjän on huolehdittava kielitiedostoista. Jos uusi käännettävä sana tarvitaan, ylläpitäjän on lisättävä se samalla avaimella ja kielen mukaisella arvolla jokaiseen properties-kielitiedostoon.

Java käyttää Unicode-merkistöä, jolloin kielituki voidaan toteuttaa tarvittaessa kaikilla maailman kielillä. Properties-tiedostojen avulla kielivalikoiman

laajentaminen on helppoa; servletteihin tarvitsee tehdä vain vähäisiä lisäyksiä uuden kielen tukea varten.

4.14.3 Tyyli tiedosto

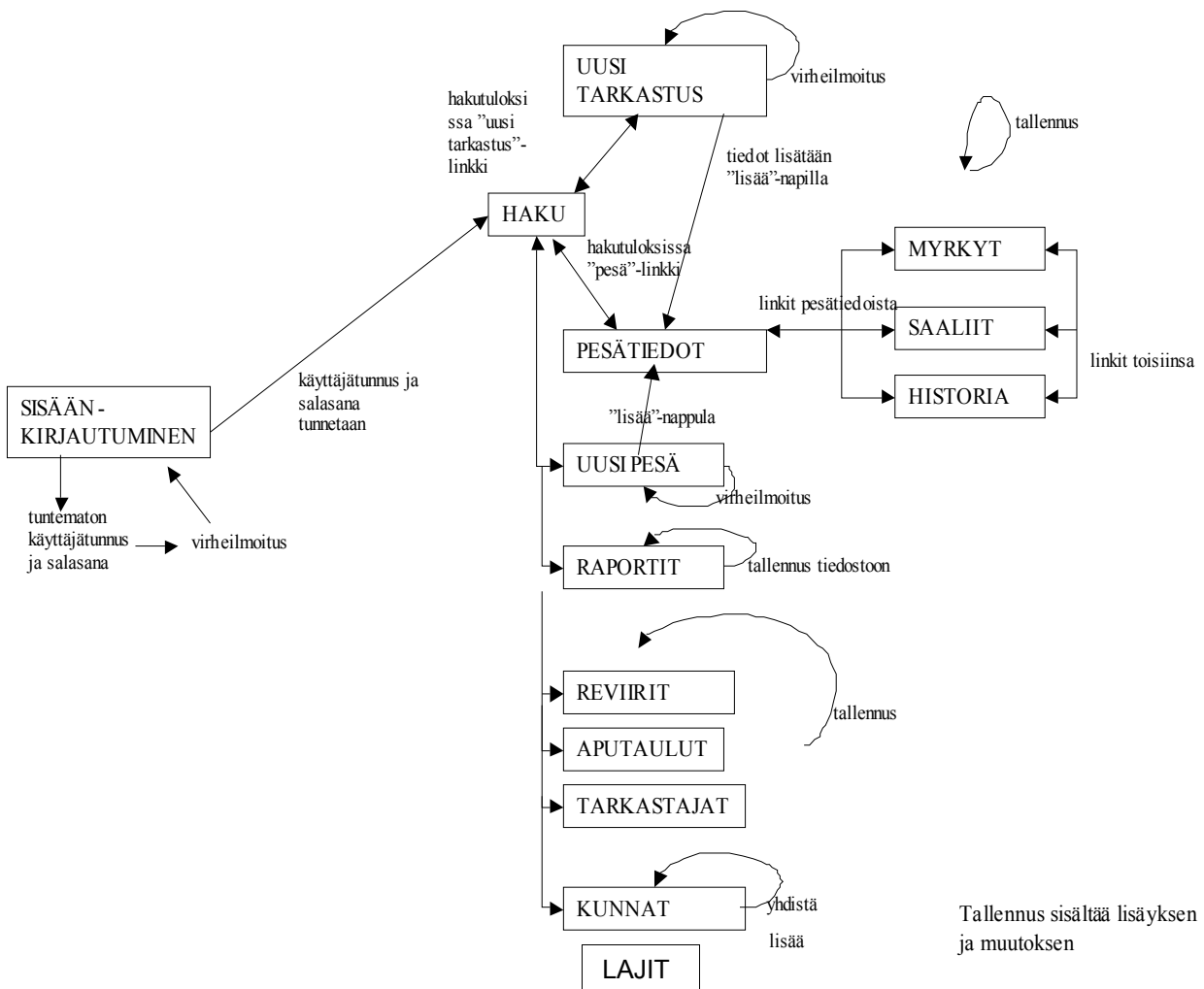
Tyyli tiedosto halityyli.css sijoitetaan samaan hakemistoon, kuin template-tiedostot. Tiedosto toimii html-tyyli tiedostona, joka kuvaa html-sivujen ulkoasua.

5 Käyttöliittymä

Tässä luvussa kuvataan järjestelmän käyttöliittymä. Suunnittelun lähtökohtana ovat käyttötapaukset, jotka on tehty käyttäjähaastattelujen perusteella. Käyttötapaukset on kuvattu sanallisesti, ja niiden pohjalta on tehty näyttökuvien sarjat, joissa käyttäjä saavuttaa tavoitteensa.

Käyttöliittymässä on 13 erilaista näyttöä: sisäänkirjautuminen, haku, uusi tarkastus, myrkyt, saaliit, historia, pesätiedot, uusi pesä, raportit, aputaulut, tarkastajat, kunnat ja laji.

Haku-sivu on sisäänkirjautumisen jälkeen ensimmäinen sivu. Navigointipalkista, joka on kaikkien sivujen yläreunassa, pääsee näyttöihin Haku, Uusi pesä, Raportit, Reviirit, Aputaulut, Tarkastajat ja Kunnat. Näistä kolme viimeistä ovat ylläpito näyttöjä. Uusi pesä-näytöllä lisätään uusi pesä ja Raportit-näytöltä saadaan aikaan erilaisia raportteja. Haku-sivulla tehdyn haun perusteella pääsee jo olemassa olevalle pesälle tekemään uuden tarkastuksen (Uusi tarkastus) tai katsomaan/muuttamaan vanhoja tietoja (Pesätiedot). Näistä näytöistä pääsee myös näyttöihin Myrkyt, Saaliit ja Historia, jotka liittyvät aina tiettyyn pesään. Käyttöliittymä on kuvattu tarkemmin luvussa 5.2.



Kuva 4: Käyttöliittymän näyttöjen väliset suhteet

5.1 Käyttötapaukset

Käyttötapaukset on tehty Heikki Lokin ja Torsten Stjernbergin haastattelujen perusteella. Kussakin käyttötapauksessa kuvataan käyttäjän tavoite ja tavoitteen saavuttamiseen vaikuttavat tilatiedot.

5.1.1 Käyttötapaus 1: Uuden pesän lisääminen

Tavoite: Merikotkakannan seurantatiedon ylläpitäminen ja päivittäminen uusien pesätietojen avulla.

Torsten "Totti" Stjernberg on löytänyt uuden merikotkapesän Ahvenanmaalta kesällä 2002. Totti on täyttänyt maastossa pesän perustietolomakkeen. Tiedot pitäisi arkistoida, jotta kannan seuraaminen nyt ja jatkossakin olisi mahdollista.

Tilatiedot

- Pesäpaikka sijaitsee Ahvenanmaalla Brändön kunnassa Lappon kylässä Alskärin saarella lounaaseen Gunnarstenistä.

- Pesäpaikan tarkka paikanmäärittely on tehty GPS-laitteella yhtenäiskoordinaateilla.
- Maastotyyppi on pieni lehtipuuta kasvava kallioinen saari, jossa on pieni kosteikko; pesä sijaitsee kosteikon reunalla. Lähiympäristön puusto on lehtimetsää, jossa kasvaa koivua ja leppää. Metsän käsittelyaste on hakkaamaton. Lähin pesäpuuksi soveltuva puu on noin kymmenen metrin päässä.
- Pesäpuu on elävä tervaleppä, jonka korkeus on arviolta 10 m ja halkaisija on 35 cm lännestä itään ja 38 cm pohjoisesta etelään. Latvian halkaisija juuri pesän alta on tarkasti 17 cm.
- Pesä sijaitsee tukevasti puussa monihaarisessa latvassa. Yläreuna on 8,5 m korkeudella. Etäisyys rantaan on 45 m. Halkaisija on 150 * 160 cm. Pesän korkeudeksi on mitattu 50 cm.
- Pesä näkyy huonosti kalliolle lännestä 10 m päässä (havaittavuus).
- Lähin asutus on Gunnarstenissä, tosin nyt vain kesäasutusta 1400 m päässä.
- Tarkastuskäynnillä 13.6.2002 havaittiin, että pesässä kaksi pesäpoikasta ja ei yhtään munaa. Pesän yläpuolella varoitteli kaksi aikuista.
 - Aikuisella koiraalla on oikeassa jalassa valkosininen rengas, jonka numeroita ei voitu lukea, ja Totti ei voinut päätellä, onko vasemmassa jalassa rengasta. Naaraalla on valkopunainen rengas oikeassa jalassa ja vasemmassa jalassa valkosininen.
 - Poikasista toinen on mahdollisesti koiras, jonka nilkka on 12,6 * 13,8 mm, siipi on 365 mm, painoa ei ole mitattu sekä nokan korkeus on 31,1 mm ja pituus on 31,4 mm. Totti rengasti sille oikeaan jalkaan punamustan renkaan numero E 16.202 ja vasempaan jalkaan mustakeltaisen renkaan numero N 207. Poikasen kupu on puoliksi täynnä.
 - Poikasista toinen on varmuudella koiras, jonka nilkka on 12,5 * 13,5 mm, siipi on 330 mm, painoa ei ole mitattu sekä nokan korkeus on 30,0 mm ja pituus 39,8 mm. Totti rengasti sille oikeaan jalkaan punamustan renkaan numero E 16.203 ja vasempaan jalkaan mustakeltaisen renkaan numero N 218. Poikasen kupu on tyhjä.
- Pesä oli asuttu vuonna 2002. Pesäpoikasten lukumäärä saatiin selville rengastettaessa.
- Paikalta löydettiin saaliita, joita ei voitu kerätä mukaan. Niitä olivat kolme naarashaahkan oikeaa siipeä, piisamin häntä, useita telkkiä ja noin yhden kilon puolikas hauki. Näytteinä on talletettu sulkia ja höyheniä, oksennuspalloja ja saalisnäytteitä sekä lisäksi poikasista nypittiin 5 selkähöyhentä DNA-näytteitä varten.
- Lisäksi Totti on kirjoittanut kohtaan ”pesän rakentamisvuosi”, että pesä on ollut joitakin vuosia olemassa. Saalistähteiden perusteella voidaan päätellä, että vuonna 2001 pesässä on ollut vähintään yksi poikanen (mahdollisesti myös jonakin aikaisempina vuosina); vuoden 2001 tulos kirjataan kohtaan ”pesimistulos aikaisempina vuosina” muotoon <1 poikanen.
- Pesäpuuhun ei ole kiinnitetty rauhoitustaulua. Pesäpalstaa tai pesimäsaarta ei ole rauhoitettu.
- Tietojen keräämisessä on oleellista, että havaintojen tekijä voi kirjata myös vapaamuotoisia kommentteja.

5.1.2 Variaatio käyttötapauksesta 1

Variaatio:

- Pesä sijaitsee tuulenpesässä, monihaaraisen latvan sijasta.
- Tuulenpesää ei kuitenkaan löydy järjestelmän valmiista koodistosta ja Totti haluaa lisätä tuulenpesälle oman koodin.
- Käytössä on Haliaeetus-järjestelmä, johon on talletettu valmiita koodeja pesän sijainnille.
- Muut tiedot ovat samat kuin yllä esitettyssä käyttötapauksessa 3.

5.1.3 Käyttötapaus 2: Tiedossa olevan reviirin tarkastus

Tavoite: Merikotkaporin seuranta ja dokumentaatio sekä uhkatekijöiden kartoitus.

Torsten ”Totti” Stjernberg on käynyt kesällä 1998 tarkastamassa aikaisemmin löydetyn reviirin Ahvenanmaalla. Totti on täyttänyt maastossa merikotkan reviirilomakkeen. Tiedot pitäisi arkistoida, jotta seuranta olisi jatkossakin mahdollista.

Tilatiedot:

- Reviiri sijaitsee Ahvenanmaalla Kumlingen kunnassa Enklingen kylässä.
- Reviirin tarkempi nimi Enklinge Blacksund.
- Reviirin vanha numero on 11501.
- Reviirin tila vuonna 1998 on poikaspesä. Aktiivisen pesän nimi on Lanto (1150103). Rengastettuja poikasia on kaksi kappaletta ja rengastamattomia ei ole. Aikuista koirasta ei näkynyt ja naaraalla oli vasen rengas ja oikeaa rengasta ei havaittu. Kerätyt saalisnäytteet olivat vanhoja jätteitä toisella vaihtopesällä (Blacksund NE)(1150102).
- Tarkistuspäivämäärä oli 2.6.1998.
- Ensimmäisellä poikasella on oikeassa jalassaan punamusta rengas, jonka numero on E 16.433, ja vasemmassa jalassaan valkomusta rengas, jonka numero on H-203. Sen sukupuoli on naaras, siiven pituus 255 mm, nilkka 13,8*16,0 mm, nokan korkeus 29,9 mm ja pituus 41,0 mm, painoa ei ole mitattu sekä kupu on luokaltaan 2.
- Toisella poikasella on oikeassa jalassaan punamusta rengas, jonka numero on E 16.434, ja vasemmassa jalassaan valkomusta rengas, jonka numero on H-204. Sen sukupuoli on koiras, siiven pituus 160 mm, nilkka 11,0*12,2 mm, nokan korkeus 24,8 mm ja pituus 34,4 mm, painoa ei ole mitattu sekä kupu on luokaltaan 2.
- Vaihtopesiä, joissa merikotka edelleen voisi pesiä, on kolme kappaletta. Ensimmäisen vaihtopesän nimi on Blacksund SE (1150101), se on tarkastettu 2.6.1998, tarkastus on tehty kiikaroimalla maasta ja pesän tila on koskematon. Pesän lähellä ei ole merkkejä, ympäristö on muuttumaton ja pesä on puussa. Toisen vaihtopesän nimi on Blacksund NE (1150102), se on tarkastettu 2.6.1998, tarkastus on tehty kiikaroimalla maasta ja pesän tila on koskematon. Pesän lähellä on yksittäisiä merkkejä, ympäristö on muuttumaton ja pesä on puussa. Lisäksi ympäristöstä löytyi yksi valkoinen untuva ja vanhoja ulosteita. Kolmannen vaihtopesän nimi on Blacksund Mell (1150104), se on tarkastettu 2.6.1998, tarkastus on tehty kiikaroimalla maasta ja pesän tila on puoliksi pudonnut. Pesän lähellä on yksittäisiä merkkejä, ympäristö on muuttumaton ja pesä on puussa. Lisäksi ympäristöstä löytyi vanhoja ulosteita.

- Asuttu pesä on Lanto tekopesä (1150103), joka sijaitsee männyssä, jonka ympärys on 148 cm / 1 m. Ensimmäinen vaihtopesä on Blacksund SE (1150101), joka sijaitsee männyssä, jonka ympärys on 127 cm / 1 m. Toinen vaihtopesä on Blacksund NE (1150102), joka sijaitsee männyssä, jonka ympärys on 100 cm / 1 m. Kolmas vaihtopesä on Blacksund Mell (1150104), joka sijaitsee männyssä, jonka ympärys on 141 cm / 1 m.
- Lanto tekopesän ympäristön metsätyyppi on mustikkatyypin tuore kangasmetsä ja metsän käsittelyaste on hakkaamaton, harvahko mäntymetsikkö. Blacksund SE:n (1150101) ympäristö on kanervatyypin kuiva kangasmetsä, metsän käsittelyaste on kallioinen mäntymetsä, etäisyys meren rantaan on 80 m ja etäisyys korpin pesään on 200 m. Blacksund NE:n (1150102) ympäristö on puolukka- ja kanervatyypin kuiva kangasmetsä ja etäisyys meren rantaan on 100 m. Black Mell:n etäisyys meren rantaan on 200 m.
- Tietojen keräämisessä on oleellista, että havaintojen tekijä voi kirjata myös vapaamuotoisia kommentteja.

5.1.4 Käyttötapaus 3: Raportin toimittaminen

Tavoite: Linnut –vuosikirjan seuranta-artikkelin tuottaminen.

Linnut -vuosikirjan toimittaja Ruokolainen on pyytänyt Torsten ”Totti” Stjernbergiltä vuoden 2003 alussa seuranta-artikkelin merikotkista 1970-, jossa olisi mukana tiedot myös vuosilta 2001-2002 suuralueittain. Edellinen katsaus päättyi vuoteen 2000.

Tilätiedot:

- Ruokolainen halusi alun perin artikkelin 28.2 mennessä, mutta Totti on saanut lisäaikaa maaliskuun loppuun.
- Tarvittavia vuosittaisia tietoja ovat suuralueittain: asuttujen reviirien lukumäärä, rengastusikäisten poikasten lukumäärä, onnistuneiden pesintöjen osuus, poikasten lukumäärä yhtä onnistunutta pesintää kohden, poikasten lukumäärä yhtä asuttua reviiriä kohden sekä asuttujen reviirien lukumäärä ja pesinnän onnistuminen suojelluilla ja suojelemattomilla alueilla.
- Totilla on jo muut raportit, paitsi asuttujen reviirien lukumäärä.
- Suuralueita ovat Ahvenanmaa, Kymenlaakso, Itä-Uusimaa, Länsi-Uusimaa, Varsinais-Suomi, Satakunta, Merenkurkku, Perämeri, Koillismaa ja Lappi.
- Vastaavat tiedot tarvitaan 2-3 vuoden välein alkuvuodesta.

5.1.5 Käyttötapaus 4: Pesän hakeminen

Tavoite: Mahdollisten pesien hakeminen alueella, jossa halutaan muuttaa maankäyttöä

Lounais-Suomen ympäristökeskuksen luonnonsuojeluyksikössä toimivalle virkamiehelle Esko Gustafssonille on tullut lausuntopyyntö rakennusosastolta selvittää mahdolliset merikotkan pesät, jotka sijaitsevat Turun saariston Högsåran saarella, johon Fortum on suunnitellut tuulivoimalan rakentamista.

Tilatiedot:

- Suunniteltu rakennusalue on Högsåran saaren länsiosassa Dragsfjärdin kunnassa.
- Esko Gustafsson haluaa selvittää alueella mahdollisesti esiintyvien pesien sijaintia.
- Esko etsii pesiä ensisijaisesti reviirin nimen perusteella. Reviirillä ja saarella on tässä tapauksessa sama nimi.
- Kun Eskolle selviää, että merikotka pesii Högsåran saarella, hän haluaa tietää pesien tarkat koordinaatit.
- Esko haluaa myös tietää, millainen on pesien käytön aste ja pesien kunto viimeisen kymmenen vuoden aikana, kuka on tehnyt havainnot pesästä ja kuinka hyvin pesä on havaittavissa.
- Havainnontekijöiden yhteystiedot löytyvät rengastajien vuosikirjasta.
- Tämänkaltaiset kyselyt ovat Eskolle jokapäiväistä työtä.

5.2 Kuvasarjat

Kuvasarjoissa kuvataan näyttö näytöltä, kun käyttäjä pyrkii tavoitteeseensa järjestelmän avulla. Kuvat ovat prototyyppejä ja yksityiskohtien suhteen muutoksia saatta tulla. Tarkoituksena onkin kuvata mahdollisimman hyvin käyttöliittymän toimintalogiikka, eikä kaikkia yksityiskohtia.

5.2.1 Käyttäjän sisäänkirjautuminen

Käyttäjä antaa käyttäjätunnuksen ja -salasanan sivulla Sisäänkirjautuminen ja painaa Sisään-nappia (**Kuva 5**).

Haliaeetus - sisään

Käyttäjätunnus	<input type="text"/>
Salasana	<input type="password"/>
	<input type="button" value="Sisään"/>

[Suomeksi](#) [Svenska](#) [English](#)

Kuva 5: Sisäänkirjautuminen 1

Aloitussivuna on Haku-sivu. Käyttäjä kirjautuu ulos Kirjautu ulos-linkistä (**Kuva 6**). Uloskirjautumisen jälkeen sivuksi tulee sivu Sisäänkirjautuminen, jossa Sisään-napin vieressä on kommentti ”Olet kirjautunut ulos järjestelmästä”.



HAKU

HAKUEHDOT

VUOSIVÄLI: -

PESÄ nimi: kunta: id:

REVIIRI: nimi: kunta: id:

SUURALUE: YMPÄRISTÖKESKUS:

KOORDINAATTI: YHTENÄISKOORDINAATISTO PITUUS m LEVEYS m

ASTEKOORDINAATISTO PITUUS ° ' " LEVEYS ° ' "

HAETUT PESÄT

NUMERO	NIMI	REVIIRI	KUNTA	
123456	HÖGSÅRA	HÖGSÅRA	DRAGSEJÄRD	UUSI TARKASTUS

Kuva 6: Haku-sivu

5.2.2 Uuden pesän lisääminen

Käyttäjä painaa navigointipalkin Uusi pesä-linkkiä. Sivulla on lomake, johon uuden pesän tiedot lisätään. Lopuksi painetaan sivun alareunassa olevaa Lisää-nappia (Kuvat 7, 8, 9 ja 10).

HAKU	UUSI PESÄ	REVIIRIT	RAPORTIT	APUTAULUT	KUNNAT	TARKASTAJAT	KIRJAUDU ULOS
----------------------	---------------------------	--------------------------	--------------------------	---------------------------	------------------------	-----------------------------	-------------------------------

UUSI PESÄ

Reviirin muut pesät: [BLACKSUND SE](#) [BLACKSUND NE](#) [BLACKSUND MELL](#)
[UUSI TARKASTUS](#) [UUSI TARKASTUS](#) [UUSI TARKASTUS](#)

[SAALIIT MYRKYT HISTORIA](#)

Tarkastuspäivä*: Päivämäärän tarkkuus*:

Tarkastaja 1: Sukunimi: Etunimi: Id*:

Tarkastaja 2: Sukunimi: Etunimi: Id:

Kunta*: Kylä,saari trns:

Reviirin nimi*: Pesän nimi*: Id: 1234567

Pesän tunnus ennen tietojärjestelmää:

Koordinaatit*: Mittaustapa*: Yhtenäis: Pituus 3 m Leveys m
Aste: Pituus ° ' " Leveys ° ' "

Rakentamisvuosi: Tarkkuus: Tuhoutumisvuosi:

Löytymisvuosi: Pesän rakentanut laji:

Rauhoitustaulun kiinnityspäivä: Kieli: Taulun nro:

Pesäpuun puulaji: Elävyys: Pesän kunto: Pesän sijainti:

Kommentti pesän sijaintiin liittyen:

Kuva 7: Uuden pesän lisääminen 1

RAUHOITUSTIEDOT

Tietojen päivämäärä:

Pesän palstan rauhoitus tilanne: Rauhoituksen alkupvm: Rauhoituksen loppupvm:

Jos pesä suojelualueella, niin suojelualueen virallinen nimi:

Pesän palstan omistaja: Kommentti:

Havaitut uhkatekijät:

PESÄPUUN JA PESÄN MITTOJA

Mittauspäivämäärä:

Puun korkeus: m Tarkkuus:

Pesän yläpinnan etäisyys maasta: m Tarkkuus: Latvasta: m Tarkkuus:

130cm maasta: Rungon ympärys: cm Halkaisija: cm Tark.:

Valittomasti pesän alla: Rungon ympärys: cm Halkaisija: cm Tark.:

Pesän korkeus: cm Pesän pinnan suurin halkaisija: cm ja pienin halkaisija cm

PESÄN YMPÄRISTÖN TIETOJA**Etäisyydet pesästä:**

Meren rantaan:	<input type="text"/> m	Järven tai lammen rantaan:	<input type="text"/> m
Avosuon reunaan:	<input type="text"/> m	(Kesä)asuntoon:	<input type="text"/> m
Autolla ajettavaan tiehen:	<input type="text"/> m	Moottorikelkkareittiin:	<input type="text"/> m
Talvitiehen:	<input type="text"/> m	Kalanviljelylatokseen:	<input type="text"/> m
Ilmajohtoon:	<input type="text"/> m	Viljeltyyn peltoon:	<input type="text"/> m
Avohakkuun/siemenpuuston reunaan:	<input type="text"/> m	Sopivaan toiseen pesäpuuhun:	<input type="text"/> m

Kuva 8: Uuden pesän lisääminen 2

Pesän sijainti:

Saarityyppi: Onko saareen tieyhteys (myös autolla kuljettavalla lautalla tms.):

Asuintalojen ja asuntojen määrä 1000m säteellä: ja 500m säteellä:

Pesän näkyvyys maastossa/vesiltä:

Puuston relaskooppimittaukset:

	25 m pohjoiseen		25 m itään		25 m etelään		25 m länteen	
	lkm	keskipituus	lkm	keskipituus	lkm	keskipituus	lkm	keskipituus
Männyt	<input type="text"/> kpl	<input type="text"/> m	<input type="text"/> kpl	<input type="text"/> m	<input type="text"/> kpl	<input type="text"/> m	<input type="text"/> kpl	<input type="text"/> m
Kuuset	<input type="text"/> kpl	<input type="text"/> m	<input type="text"/> kpl	<input type="text"/> m	<input type="text"/> kpl	<input type="text"/> m	<input type="text"/> kpl	<input type="text"/> m
Muut	<input type="text"/> kpl	<input type="text"/> m	<input type="text"/> kpl	<input type="text"/> m	<input type="text"/> kpl	<input type="text"/> m	<input type="text"/> kpl	<input type="text"/> m

Puustotyyppi: Käsitelyaste: Puuston ikä: Maastotyyppi:

Pesän ympäristöstä otettu valokuva ja lähetetty eläinmuseoon:

TARKASTUSTIEDOT

Pesän nimi: Tarkastusvuosi:

Tarkastaja 1: Sukunimi: Etunimi: Rengastaja/havainnoijanro*:

Tarkastaja 2: Sukunimi: Etunimi: Rengastaja/havainnoijanro:

Tarkastuspäivä*: Päivämäärän tarkkuus*: Kellonaika: Tarkastustapa:

Pesivä laji, jos ei merkitä:

Pesimistulos: Tarkkuus: Nähdyt pesinnän merkit

Epäonnistumisen syy: Tarkkuus:

Pesimistuloksen kommentti:

Pesän kunto: Merkit pesän ympärillä:

Lopullisen munamäärän tarkastuspäivämäärä:

Munien lopullinen määrä: Kuoriutumattomia munia: Kuolleita poikasia:

Eläviä poikasia: Rengastusikäisiä poikasia: Lentopoikasia:

Kuva 9: Uuden pesän lisääminen 3

HALIAETUS- Uusi pesä - Microsoft Internet Explorer

Tiedosto Muokkaa Näytä Suosikit Työkalut Ohje

Edellinen

Osoite <http://www.cs.helsinki.fi/u/hhokane/hali/uusipesa.html> Siirry Linkit >>

TIEDOT AIKUISISTA

Lukumäärä:

Koiras: Rengas vasen: Tunnus: Väri: Rengas oikea: Tunnus: Väri: Rengas nähty:

Naaras: Rengas vasen: Tunnus: Väri: Rengas oikea: Tunnus: Väri: Rengas nähty:

Koiras tai naaras: Rengas vasen: Tunnus: Väri: Rengas oikea: Tunnus: Väri: Rengas nähty:

TIEDOT POIKASISTA

Poikanen	1	2	3	4
Sukupuoli	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Vasemman jalan renkaan tunnus:	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Vasemman jalan renkaan väri:	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Oikean jalan renkaan tunnus:	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Oikean jalan renkaan väri:	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Sivun pituus	<input type="text"/> mm	<input type="text"/> mm	<input type="text"/> mm	<input type="text"/> mm
Sivun mittausmenetelmä:	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Nilkan paksuus maksimi:	<input type="text"/> mm	<input type="text"/> mm	<input type="text"/> mm	<input type="text"/> mm
Nilkan paksuus minimi:	<input type="text"/> mm	<input type="text"/> mm	<input type="text"/> mm	<input type="text"/> mm
Nokan pituus:	<input type="text"/> mm	<input type="text"/> mm	<input type="text"/> mm	<input type="text"/> mm
Nokan korkeus tyvestä:	<input type="text"/> mm	<input type="text"/> mm	<input type="text"/> mm	<input type="text"/> mm
Paino:	<input type="text"/> g	<input type="text"/> g	<input type="text"/> g	<input type="text"/> g
Kupu:	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Höyhen/verinäyte otettu:	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

NÄYTTEITÄ LÄHETETTY

Merikotkan sulkia ja höyheniä: Munia: kpl Munansiruja:

Kuolleita poikasia: kpl Kuolleita aikuisia: kpl

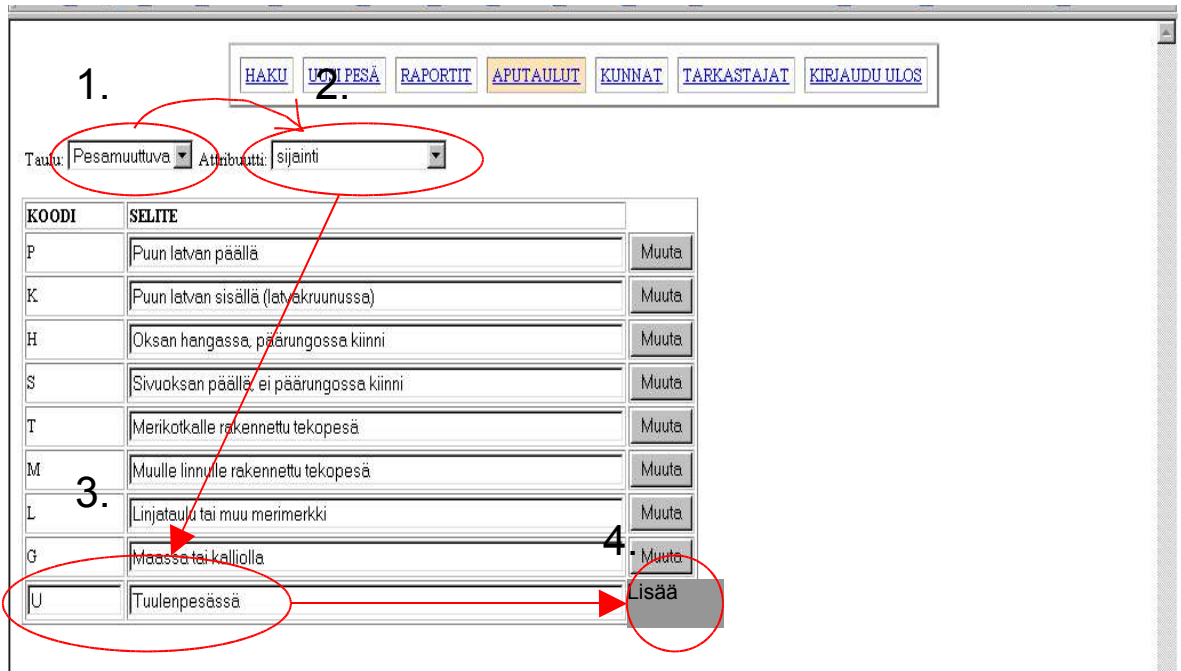
Saalisnäytteitä: pussia Oksennuspalloja:

Tallenna

Kuva 10: Uuden pesän lisääminen 4

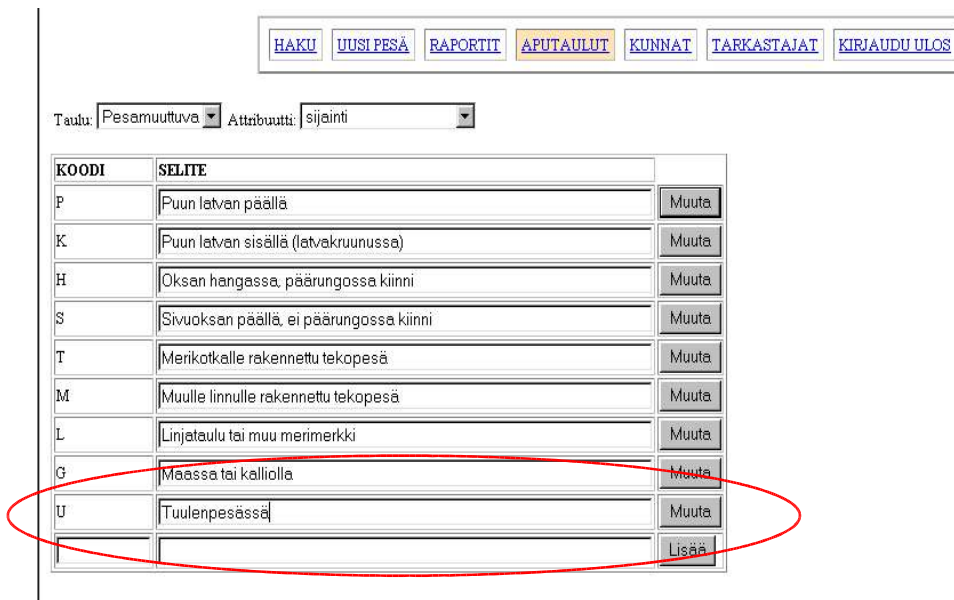
5.2.3 Aputaulun päivitys

Käyttötapaus 1:n variaatiossa, käyttäjä joutuu lisäämään aputauluun uuden koodin. Tämä johtuu siitä, että haluttua koodia ei järjestelmästä vielä löydy. Käyttäjä menee Aputaulujen ylläpito -sivulle navigointipalkin linkillä Aputaulut. Sivulla käyttäjä valitsee haluamansa taulun alavetovalikosta (**Kuva 12, 1.**). Tämä aiheuttaa sen, että Attribuutti-alavetovalikkoon ilmestyy kaikki kyseisen taulun aputaulua käyttävät attribuutit. Kun oikea attribuutti valitaan (**Kuva 12, 2.**), alapuolelle ilmestyy lista kyseisen attribuutin selitteistä. Tyhjälle riville lisätään uusi koodi ja selite (**Kuva 12, 3.**), sekä painetaan Lisää-nappia (**Kuva 12, 4.**).



Kuva 12: Aputaulun päivitys 1

Lisää-napin painamisen jälkeen uusi koodi ja selite ilmestyy listaan. Selitteitä voi korjata tekemällä korjaukset suoraan selite-kenttään ja painamalla kyseisen rivin Muuta-nappia. Tällöin järjestelmä antaa varoituksen siitä, että käyttäjä varmasti tietää mitä on tekemässä. Mahdolliselle uudelle koodille on myös ilmestynyt tyhjä rivi.



Kuva 13: Aputaulun päivitys 2

5.2.4 Vanhan pesän tarkastus

Käyttäjä hakee Haku-sivulla haluamansa pesät täyttämällä sopivat hakuehdot (tässä tapauksessa suuralue, reviirin nimi ja kunta), ja painamalla Hae Pesät-nappia.

Hakutulokset ilmestyvät listaksi näytön alareunaan. Pesän nimi-linkillä pääsee katselemaan/muuttamaan kyseisen pesän tietoja. Uusi Tarkastus-linkillä pääsee tekemään uuden tarkastuksen kyseiselle pesälle (**Kuva 14**).

HAKU
[UUSI PESÄ](#)
[REVIIRIT](#)
[RAPORTIT](#)
[APUTAULUT](#)
[KUNNAT](#)
[TARKASTAJAT](#)
[KIRJAUDU ULOS](#)

HAKU

HAKUEHDOT

PESÄ nimi: kunta: id:

REVIIRI: nimi: kunta: id:

SUURALUE: YMPÄRISTÖKESKUS:

KOORDINAATTI: YHTENÄISKOORDINAATISTO PITUUS m LEVEYS m

ASTEKOORDINAATISTO PITUUS ° ' " LEVEYS ° ' "

HAETUT PESÄT

NUMERO	NIMI	REVIIRI	KUNTA	
123456	HÖGSÅRA	HÖGSÅRA	DRAGSFJÄRD	UUSI TARKASTUS

Kuva 14: Vanhan pesän tarkastus 1

Lomake on pääosin esitäytetty edellisen vuoden tietojen perusteella. Tarkastuspäivä on kuitenkin syötettävä. Ja jos muutoksia on tullut, voi vanhoja tietoja muuttaa. Reviirin muihin pesiin ja pesän vanhempiin tarkastuksiin pääsee yläosassa olevilla linkeillä. Reviirin nimi-linkillä pääsee muuttamaan ja katsomaan tietoja ja Uusi Tarkastus-linkillä lisäämään kyseiselle vaihtopesälle uuden tarkastuksen. **(Kuva 15).**

HAKU	UUSI PESÄ	REVIIRIT	RAPORTIT	APUTAULUT	KUNNAT	TARKASTAJAT	KIRJAUDU ULOS
----------------------	---------------------------	--------------------------	--------------------------	---------------------------	------------------------	-----------------------------	-------------------------------

PESÄTARKASTUS

Reviirin muut pesät: [BLACKSUND SE](#) [BLACKSUND NE](#) [BLACKSUND MELI](#)

[UUSI TARKASTUS](#) [UUSI TARKASTUS](#) [UUSI TARKASTUS](#)

Tarkastusvuodet:

2003 [2002](#) [2001](#) [2000](#)

SAALIIT MYRKYT HISTORIA

Tarkastuspäivä*:	<input type="text" value="11"/>	<input type="text" value="06"/>	<input type="text" value="2003"/>	Päivämäärän tarkkuus*:	<input type="text"/>	Kirjauspäivä: 4.8.2003	Muutospäivä: 4.8.2003
Tarkastaja 1: Sukunimi:	<input type="text" value="Stjernberg"/>	Etinimi:	<input type="text" value="Torsten"/>	Id*:	<input type="text" value="246"/>		
Tarkastaja 2: Sukunimi:	<input type="text"/>	Etinimi:	<input type="text"/>	Id:	<input type="text"/>		
Kunta*:	<input type="text" value="Kumlinge"/>	Kylä,saari tms:	<input type="text" value="Outoskari"/>				
Reviirin nimi*:	<input type="text" value="Merikotkamännynkari"/>	Pesän nimi*:	<input type="text" value="Isoluodonpetäjä"/>	Id:	<input type="text" value="1234567"/>		
Pesän tunnus ennen tietojärjestelmää:	<input type="text" value="G2061103"/>						
Koordinaatit*:	Mittaustapa*:	<input type="text" value="G"/>	Yhtenäis:	<input checked="" type="radio"/>	Pituus:	<input type="text" value="313455"/>	m
				<input type="radio"/>	Leveys:	<input type="text" value="654654"/>	m
				Aste:	<input type="text"/>	°	<input type="text"/>
					<input type="text"/>	'	<input type="text"/>
					<input type="text"/>	"	<input type="text"/>
Rakentamivuosi:	<input type="text" value="1999"/>	Tarkkuus:	<input type="text" value="0"/>	Tuhoutumisvuosi:	<input type="text"/>		
Löytymisvuosi:	<input type="text" value="1999"/>	Pesän rakentanut laji:	<input type="text" value="Halalb"/>				
Rauhoitustaulun kiinnityspäivä:	<input type="text" value="05"/>	<input type="text" value="05"/>	<input type="text" value="2000"/>	Kieli:	<input type="text" value="FIN"/>	Taulun nro:	<input type="text" value="12321"/>
Pesäpuun puulaji:	<input type="text" value="M"/>	Elävyys:	<input type="text" value="E"/>	Pesän kunto:	<input type="text" value="K"/>	Pesän sijainti:	<input type="text" value="P"/>
Kommentti pesän sijaintiin liittyen:	<input type="text"/>						

Kuva 15: Vanhan pesän tarkastus 2

Tarkastustietoja (Näytöllä otsikot Tarkastus, Aikuiset linnut ja Poikaset) ei ole esitetyt, eli lomakkeen alaosa on tyhjennetty edellisen vuoden tiedoista. Käyttäjä täyttää nämä tiedot ja painaa Tallenna-nappia (**Kuva 16**). Seuraus tästä on sama kuin Uuden pesän lisäämisessä. Muut reviiirin pesät lisätään samalla tavalla menemällä ensin sivun yläosan linkeistä kyseisen vaihtopesän tietoihin.

TIEDOT AIKUISISTA

Lukumäärä:

Koiras:	Rengas vasen: Tunnus: <input type="text"/>	Väri: <input type="text"/>	Rengas oikea: Tunnus: <input type="text"/>	Väri: <input type="text"/>	Rengas nähty: <input type="checkbox"/>
Naaras:	Rengas vasen: Tunnus: <input type="text"/>	Väri: <input type="text"/>	Rengas oikea: Tunnus: <input type="text"/>	Väri: <input type="text"/>	Rengas nähty: <input type="checkbox"/>
Koiras tai naaras:	Rengas vasen: Tunnus: <input type="text"/>	Väri: <input type="text"/>	Rengas oikea: Tunnus: <input type="text"/>	Väri: <input type="text"/>	Rengas nähty: <input type="checkbox"/>

TIEDOT POIKASISTA

Poikanen	1	2	3	4
Sukupuoli	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Vasemman jalan renkaan tunnus:	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Vasemman jalan renkaan väri:	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Oikean jalan renkaan tunnus:	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Oikean jalan renkaan väri:	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Siiven pituus	<input type="text"/> mm	<input type="text"/> mm	<input type="text"/> mm	<input type="text"/> mm
Siiven mittausmenetelmä:	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Nilkan paksuus maksimi:	<input type="text"/> mm	<input type="text"/> mm	<input type="text"/> mm	<input type="text"/> mm
Nilkan paksuus minimi:	<input type="text"/> mm	<input type="text"/> mm	<input type="text"/> mm	<input type="text"/> mm
Nokan pituus:	<input type="text"/> mm	<input type="text"/> mm	<input type="text"/> mm	<input type="text"/> mm
Nokan korkeus tyvestä:	<input type="text"/> mm	<input type="text"/> mm	<input type="text"/> mm	<input type="text"/> mm
Paino:	<input type="text"/> g	<input type="text"/> g	<input type="text"/> g	<input type="text"/> g
Kupu:	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Höyhen/verinäyte otettu:	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

NÄYTTEITÄ LÄHETETTY

Merikotkan sulkia ja höyheniä: Munia: kpl Munansiruja:

Kuolleita poikasia: kpl Kuolleita aikuisia: kpl

Saalisnäytteitä: pussia Oksennuspalloja:



Kuva 16: Vanhan pesän tarkastus 3

5.2.5 Raportin toimittaminen

Käyttäjä pääsee navigointipalkin Raportit-linkin avulla Raportit-sivulle. Kyseisellä sivulla valitaan haluttu aikaväli ja alue (tässä tapauksessa koko maa) ja haluttu raportti ja painetaan Tallenna tiedostoon-nappia jonka seurauksena tiedosto tallentuu asetustiedostossa määritettyyn hakemistoon. **(Kuva 18).**

HAKU	UUSI PESÄ	REVIIRIT	RAPORTIT	APUTAULUT	KUNNAT	TARKASTAJAT	KIRJAUDU ULOS
----------------------	---------------------------	--------------------------	--------------------------	---------------------------	------------------------	-----------------------------	-------------------------------

RAPORTIT

RAPORTTITYYPPI

- Yhteenveto suojeleviranomaisten käyttöön
- Tarkastajien pesätarkastukset
- Revii- ja pesäkohtaiset pesimistulokset ja kuoriutumistiedot
- Kuoriutumispäivät
- Yhteenveto kuoriutumismisraportista.
- Pesimistulokset revireittäin
- Puustot
- Tarkat poikuekoot rengastushetkellä
- Pesäpuiden elävyys
- Pesäpuiden puulajit luonnonpesissä
- Pesäpuiden puulajit tekopesissä
- Pesien sijainti
- Puustojen käsittelyasteet
- Pesien sijainnit pesäsaaren tyyppin mukaan
- Pesien sijainnit maastotyyppin mukaan
- Palstan rauhoitusaste ja pesimistulos
- Uusi pesä
- Vanha pesä

AIKAVÄLI -

ALUEET: Koko maa:

Suuralue:	<input type="text"/>	Ympäristökeskus:	<input type="text"/>	Kunta:	<input type="text"/>
Revii:	<input type="text"/>	Pesä:	<input type="text"/>		

Kuva 18: Raportin toimittaminen

5.2.6 Pesän hakeminen

Käyttäjä syöttää Haku-sivulla haluamansa hakuehdot, ja painaa Hae pesät nappia (**Kuva 19**). Tekstikenttien hakuehtojen ei tarvitse olla täydellisiä, vaan käyttäjä voi syöttää esim. pari ensimmäistä kirjainta reviirin nimestä ja jokerimerkin.

HAKU
UUSI PESÄ
REVIIRIT
RAPORTIT
APUTAULUT
KUNNAT
TARKASTAJAT
KIRJAUDU ULOS

HAKU

HAKUEHDOT

PESÄ nimi: kunta: id:

REVIIRI: nimi: kunta: id:

SUURALUE: YMPÄRISTÖKESKUS:

KOORDINAATTI: YHTENÄISKOORDINAATISTO PITUUS m LEVEYS m

ASTEKOORDINAATISTO PITUUS ° ' " LEVEYS ° ' "

HAETUT PESÄT

NUMERO	NIMI	REVIIRI	KUNTA	
123456	HÖGSÅRA	HÖGSÅRA	DRAGSFJÄRD	UUSI TARKASTUS

Kuva 19: Pesän hakeminen 1

Hakutulokset ilmestyvät listaksi (tässä tapauksessa yksi pesä), ja Pesän nimi-linkillä (Högsåra) päästään katsomaan kyseisen pesän tietoja (**Kuva 20**).

Käyttäjä voi tarkastella Pesätiedot-sivulla pesän tietoja. Tämän jälkeen käyttäjä painaa Raportit-linkkiä.

HAKU	UUSI PESÄ	REVIIRIT	RAPORTIT	APUTAULUT	KUNNAT	TARKASTAJAT	KIRJAUDU ULOS
----------------------	---------------------------	--------------------------	--------------------------	---------------------------	------------------------	-----------------------------	-------------------------------

PESÄTIEDOT

Reviirin muut pesät: [BLACKSUND SE](#) [BLACKSUND NE](#) [BLACKSUND MELL](#)
[UUSI TARKASTUS](#) [UUSI TARKASTUS](#) [UUSI TARKASTUS](#)

Tarkastusvuodet:

[2003](#) [2002](#) [2001](#) [2000](#)

[SAALITT MYRKYT HISTORIA](#) [UUSI TARKASTUS](#)

Tarkastuspäivä*: 11 06 2003 Päivämäärän tarkkuus*: 1 Kirjauspäivä: 4.8.2003 Muutospäivä: 4.8.2003

Tarkastaja 1: Sukunimi: Stjernberg Etunimi: Torsten Id*: 246

Tarkastaja 2: Sukunimi: Etunimi: Id:

Kunta*: Kumlinge Kylä,saari tms: Outoskeri

Reviirin nimi*: Merikotkamännynkari Pesän nimi*: Isoluodonpetäjä Id: 1234567

Pesän tunnus ennen tietojärjestelmää: G2061103

Koordinaatit*: Mittaustapa*: G Yhtenäs: Pituus 313544 m Leveys 654654 m
 Aste: Pituus ° ' " Leveys ° ' "

Rakentamisvuosi: 1999 Tarkkuus: 0 Tuhoutumisvuosi:

Löytymisvuosi: 1999 Pesän rakentanut laji: Halalb

Rauhoitustaulun kiinnityspäivä: 05 05 2000 Kieli: FIN Taulun nro: 12321

Pesäpuun puulaji: M Elävyys: E Pesän kunto: K Pesän sijainti: P

Kommentti pesän sijaintiin liittyen:

Kuva 20: Pesän hakeminen 2

Raportit-sivulla käyttäjä tekee haluamiensa tietojen perusteella raportin samalla tavalla kuin edellisessä käyttötapauksessa (Raportin toimittaminen).

5.2.7 Historia-näyttö

Historia-näytöllä voidaan tallettaa, katsella ja muuttaa tietyn pesän historiatietoja. Historia-sivulle pääsee tietyn pesän Pesätiedot-näytöltä.

SAALIIT MYRKYT HISTORIA

HISTORIA

Pesä: Kirjausvm: Muuttamisvm:

Tunnetun asumisajan alkuvuosi: Alkuvuoden tarkkuus: Tunnetun asumisajan loppuvuosi: Alkuvuoden tarkkuus:

TIEDON LÄHDE	KOMMENTTI
<input type="text"/>	<input type="text"/>

TALLENNA

Kuva 23: Historia-näyttö

5.2.8 Kunnat-näyttö

Kunnat-näytöllä voi lisätä uuden kunnan. Kuntaliitosta varten on ensin haettavat halutut kunnat tietyltä suuralueelta tai -alueilta ruksimalla suuralueet ja painamalla Hae kunnat-nappia. Tämän seurauksena näytön alareunaan tulee lista kunnista valituilla suuralueilla. Tästä listasta ruksitaan yhdistettävät kunnat ja kirjoitetaan uuden kunnan tunnus, valitaan suuralue, ympäristökeskus ja painetaan Yhdistä kunnat-nappia.

HAKU
UUSI PESÄ
REVIIRIT
RAPORTIT
APUTAULUT
KUNNAT
TARKASTAJAT
KIRJAUDU ULOS

KUNNAT

UUSI KUNTA

Tunnus* Nimi* Suuralue* Ympäristökeskus*

Kunnan keskipiste: Leveys* Pituus*

Kunnan säde* km

KUNTALIITOKSET

SUURALUEET

- Ahvenanmaa (A)
- Kymenlaakso (K)
- Itä-Uusimaa (I)
- Länsi-Uusimaa (U)
- Varsinais-Suomi (R)
- Satakunta (S)
- Merenkurkku (M)
- Perämeri (P)
- Koillismaa (O)
- Lappi (L)

LITETTÄVÄT KUNNAT*

-
-

UUSI KUNTA

Tunnus* Suuralue* Ympäristökeskus*

Kuva 24: Kunnat-näyttö

5.2.9 Myrkyt-näyttö

Myrkyt-näytölle päästään tietyn pesän Pesätiedot-näytön linkiltä. Myrkyt-näytöllä voi katsoa, lisätä ja muuttaa tietyn pesän myrkkytietoja/vuosi (**Kuva 25**).

[HAKU](#)
[UUSI PESÄ](#)
[REVIIRIT](#)
[RAPORTIT](#)
[APUTAULUT](#)
[KUNNAT](#)
[TARKASTAJAT](#)
[KIRJAUDU ULOS](#)

MYRKYT [SAALIT HISTORIA](#)

MYRKYT

Pesä: [pesän nimi](#)

Keräyspäivä*: Analyysipäivä: Keraaja*: Analysoija*:

MUNA

Pituus	Leveys	Kuoren paksuus ilman kalvoa	Kuoren paksuus kalvoineen	Paino	Painon tarkkuus	Sisällön laatu	Sisällön väri	Sikiön koko
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

PITOISUUDET

DDT: 0.1 ppm DDD: 0.1 ppm DDE: 0.1 ppm PCB: 0.1 ppm HG: 0.1 ppm

ANALYYSIMENETELMÄ

KOMMENTTI

TALLENNA

Kuva 25: Myrkyt-näyttö

5.2.10 Saaliit-näytöt

Saaliit_vanhat-näytölle (**Kuva 26**) päästään kuten Myrkyt- ja Historia-näytöillekin. Saaliit_vanhat-näytöllä voi tarkastella tietyn pesän saalistietoja/vuosi. Koska samalle pesälle voi tehdä useita saalistarkastuksia/vuosi, on saaliille oma ylläpito-näyttö. Saaliit_vanhat sivun päivämäärälinkeistä pääsee muuttamaan yhden saalistarkastuksen tietoja, ja Uusi-linkistä pääsee lisäämään uuden tarkastuksen. Saaliit_yllapito-näytöllä voi lisätä ja muuttaa yksittäisen saalistarkastuksen tietoja (**Kuva 27**).

The screenshot shows a web application interface for bird banding records. At the top, there is a navigation bar with buttons for HAKU, UUSI PESÄ, REVIIRIT, RAPORTIT, APUTAULUT, KUNNAT, TARKASTAJAT, and KIRJAUDU ULOS. Below this, the page title is SAALIIT MYRKYT HISTORIA. The main heading is SAALIIT. There are links for Pesä: Pesän nimi and Keräyshedot: KAIKKI, pp.kk.vvvv, pp.kk.vvvv, UUSI. The section SAALIIT KERÄYSPÄIVITTÄIN contains two identical data blocks. Each block shows Keräyspäivä, pp.kk.vvvv, Kerääjä, Kalle Kerääjä, and Määrityspäivä, pp.kk.vvvv, Määrittäjä, Minna Määrittäjä. Below each block is a table with two columns: Laji and Lukumäärä.

Laji	Lukumäärä
ANAPLA	3
SOMMOL	2

Kuva 26: Saaliit_vanhat-näyttö

HAKU UUSI PESÄ REVIIRIT RAPORTIT APUTAULUT KUNNAT TARKASTAJAT KIRJAUDU ULOS

SAALIIT [MYRKYT HISTORIA](#)

SAALIIT

Pesä: [Pesän nimi](#)

Keräyshedot: [KAIKKI](#) [pp.kk.vvvv](#) [pp.kk.vvvv](#) [UUSI](#)

Keräyspäivä* Keräjä*

Määrittäjä* Määrittäjä

SAALIIT

Laji	Lukumäärä	Keräyspaikka	
ANAPLA	<input type="text" value="3"/>	<input type="text"/>	Muuta
SOMMOL	<input type="text" value="2"/>	<input type="text"/>	Muuta
<input type="text"/>	<input type="text"/>	<input type="text"/>	Lisää
<input type="text"/>	<input type="text"/>	<input type="text"/>	Lisää
<input type="text"/>	<input type="text"/>	<input type="text"/>	Lisää
<input type="text"/>	<input type="text"/>	<input type="text"/>	Lisää
<input type="text"/>	<input type="text"/>	<input type="text"/>	Lisää
<input type="text"/>	<input type="text"/>	<input type="text"/>	Lisää
<input type="text"/>	<input type="text"/>	<input type="text"/>	Lisää
<input type="text"/>	<input type="text"/>	<input type="text"/>	Lisää

KOMMENTTI

Kuva 27: Saaliit_yllapito-näyttö

5.2.11 Tarkastajat-näyttö

Tarkastajat-näytölle pääsee yläosan navigointipalkista. Sivulla voi lisätä uuden tarkastajan, tai muuttaa jo olemassa olevien tarkastajien nimiä. Sivulle listautuu kaikki tietokantaan talletetut tarkastajat. Tarkastajan ID:tä ei ole mahdollista muuttaa.

TARKASTAJAT

UUDEN TARKASTAJAN LISÄYS

Etunimi*: Sukunimi*: Tarkastaja ID*:

NIMEN MUUTTAMINEN

Etunimi*	Sukunimi*	Tarkastaja ID*
<input type="text"/>	<input type="text"/>	11111
<input type="text"/>	<input type="text"/>	22222
<input type="text"/>	<input type="text"/>	33333

Kuva 28: Tarkastajat-näyttö

5.2.12 Reviirit-näyttö

Reviirit-näytöllä voidaan lisätä järjestelmään uusi reviiri (**Kuva 29**)

[HAKU](#)
[UUSI PESÄ](#)
[REVIIRIT](#)
[RAPORTIT](#)
[APUTAULUT](#)
[KUNNAT](#)
[TARKASTAJAT](#)
[KIRJAUDU ULOS](#)

Reviirit

Uusi reviiri

Reviirinumero Nimi Kunta Vuosi Vanha reviirinumero Kommentti

Vanhojen tietojen päivitys

Valitse muutettava reviiri

Reviirinumero Nimi Kunta Vuosi Vanha reviirinumero Kommentti

Kuva 29: Reviirit-näyttö

5.2.13 Lajit-näyttö

Lajit-näytöllä voidaan lisätä järjestelmään uusi laji (**Kuva 30**)

[HAKU](#)
[UUSI PESÄ](#)
[REVIIRIT](#)
[RAPORTIT](#)
[APUTAULUT](#)
[KUNNAT](#)
[TARKASTAJAT](#)
[LAJIT](#)
[KIRJAUDU ULOS](#)

Lajit

UUSI LAJI

Laji ID	Kommentti
<input type="text"/>	<input type="text"/>

LAJIN TIETOJEN MUUTTAMINEN

Laji ID	Kommentti	
ABDUTA	Harvinainen sääksilaji Ghanasta.	<input type="button" value="Muuta"/>
DUBÄTD	Punainen.	<input type="button" value="Muuta"/>
PAKTUS	-	<input type="button" value="Muuta"/>

Kuva 30: Lajit-näyttö

6 Testaussuunnitelma

Tässä luvussa kuvataan ohjelmistolle tehtävän testauksen eri vaiheet, joita ovat komponenttitestaus (6.1), integrointitestaus (6.2) ja järjestelmätestaus (6.3). Itse testauksen tarkasta kuvauksesta ja testitapauksista tehdään erillinen dokumentti testivaiheen lopuksi. Testivaihe alkaa toteutusvaiheen loppupuolella.

Vaikka ohjelmointi pyritään tekemään mahdollisimman virheettömästi, virheitä ohjelmiin jää aina, jollei kyse ole aivan triviaalista ohjelmasta. Testauksen tarkoitus on havaita aikaisemmin tuntemattomat virheet. Täydellinen testaus on usein mahdotonta, mutta testauksella pyritään mahdollisimman pienellä työmäärällä löytämään mahdollisimman paljon virheitä. Testauksessa on tärkeää, että se on suunnitelmallista ja järjestelmällistä, ja samat testitapaukset ovat toistettavissa.

6.1 Komponenttitestaus

Komponenttitestauksessa ohjelmiston jokainen luokka ja metodi testataan erikseen. Testaus tehdään kattavasti, eli jokainen ohjelmiston mahdollinen suorituspolku pyritään testaamaan. Tämä menetelmä on rakenteellinen, eli white-box menetelmä. Rakenteellisessa testauksessa testitapaukset tehdään ohjelmiston rakenteen perusteella. Rakenteelliseen testaukseen kuuluu myös pöytätestaus, eli ohjelman läpikäyminen paperilla. Pöytätestausta tehdään jo suunnittelun ja toteutuksen yhteydessä.

6.2 Integrointitestaus

Integrointitestauksen tarkoituksena on varmistaa eri luokkien yhteensopivuus. Kyseessä on black box-menetelmä, eli testausaineistona ovat ohjelman spesifikaation perusteella valitut syötteen. Syötteen jaetaan ekvivalenssiluokkiin siten, että samassa luokassa olevat syötteen todennäköisesti löytävät samat virheet. Näin testattavien syötteen lukumäärä pysyy kohtuullisena. Syötteen valinta on yleensä heuristista.

Ekvivalenssiluokkien lisäksi käytetään raja-arvoanalyysiä, jossa keskitytään rajatapauksiin. Virheet keskittyvät usein juuri syötteen raja-tapauksiin. Lisäksi näiden systemaattisten menetelmien lisäksi kannattaa käyttää vielä epäformaalia menetelmää, arvaamista. Ainakin kirjoittajan oman kokemuksen perusteella monet isoimmista virheistä löytyvät usein epämääräisellä tavalla, jota voisi kutsua vaistoksi.

6.3 Järjestelmätestaus

Järjestelmätestauksessa testataan koko ohjelmisto. Testauksessa kiinnitetään huomiota mm. toiminnallisuuteen, käyttöliittymään, virhetilanteisiin, turvallisuuteen ja suorituskykyyn. Järjestelmätestauksessa otetaan huomioon vaatimusdokumentissa määritellyt ohjelmiston vaatimukset.

Lähteet

- [1] HyperText Transfer Protocol – HTTP/1.1, 1999. <http://www.w3.org/Protocols/rfc2616/rfc2616.html>. [15.3.2003]
- [2] FreeMarker 2.x
<http://www.freemarker.org>. [15.3.2003]
- [3] Apache Tomcat <http://jakarta.apache.org/tomcat/>. [19.3.2004]
- [4] JDBC™ Technology.
<http://java.sun.com/products/jdbc/>. [15.3.2003]
- [5] Java™ Servlet technology. <http://java.sun.com/products/servlet/>. [15.3.2003]
- [6] GNU GENERAL PUBLIC LICENSE, Version 2, June 1991 <http://www.fsf.org/licenses/gpl.html>. [15.3.2003]
- [7] Ohjelmistotuotantoryhmä Tipu4.
<http://www.cs.helsinki.fi/group/tipu4/>. [15.3.2003]
- [8] The World Wide Web Consortium (W3C).
<http://www.w3.org/>. [7.4.2003]
- [9] HTML 4.01 Specification, 1999. <http://www.w3.org/TR/html-401/>. [7.4.2003]
- [10] Oracle9i Database.
<http://www.oracle.com/ip/dep/otn/database/oracle9i/>. [7.4.2003]
- [11] Java™ 2 Platform, Standard Edition (J2SE™).
<http://java.sun.com/j2se/1.4/>. [7.4.2003]
- [12] Code Conventions for the Java™ Programming Language, 1999.
<http://java.sun.com/docs/codeconv/>. [7.4.2003]
- [13] Javadoc Tool Home Page.
<http://java.sun.com/j2se/javadoc/>. [7.4.2003]
- [14] CVS – Concurrent Versions System.
<http://www.cs.helsinki.fi/group/oukki/doc/ohjeita/cvs-manual/>. [7.4.2003]
- [15] Internet Explorer Home Page.
<http://www.microsoft.com/windows/ie/default.asp>. [7.4.2003]
- [16] Ohjelmistotuotantoprojekti Hali.
<http://www.cs.helsinki.fi/group/hali/>. [10.2.2003]
- [17] Ohjelmistotuotantoprojekti Hali2
<http://www.cs.helsinki.fi/group/hali2/>. [4.3.2004]