



Kasi

Suunnitteludokumentti

Helsinki 06.05.07

Ohjelmistotuotantoprojekti

HELSINGIN YLIOPISTO

Tietojenkäsittelytieteen laitos

Kurssi

581260 Ohjelmistotuotantoprojekti (9+1op/6ov)

Projektiryhmä

Lauri Holmas

Veli-Pekka Kestilä

Joni Lahtinen

Tuukka Palomäki

Markus Penttilä

Antti-Pekka Sarin

Ilkka Tikkala

Asiakas

Harri Laine

Petri Kutvonen

Vastuhenkilö

Kimmo Simola

Ohjaaja

Jaakko Saaristo, Ilari Moilanen

Kotisivu

www.cs.helsinki.fi/group/kasi

Versiohistoria

Versio	Päiväys	Tehdyt muutokset
0.1	16.2.2007	VPK: Aloitettu dokumentin teko.
0.2	18.2.2007	MP: Lisätty alustavat luokkakuvaukset.
0.3	18.2.2007	VPK: asiakasliittymän tekstien parannus.
0.4	19.2.2007	VPK: Lisätty tietokantakuvaus.
0.5	21.2.2007	JKL: Editoitu GUI:n luokkakuvauksia.
0.52	21.2.2007	VPK: Lisätty raportointiliittymän kappale.
0.55	22.2.2007	VPK: Lisätty kuvat dokumentin liite-osaan.
0.56	23.2.2007	TP: Parannettu raportointipuolen kuvausta.
0.59	24.2.2007	MP: Lisätty pakettien nimet, tyyllittelyä.
0.60	24.2.2007	MP: Pieniä muutoksia tuoteolioihin
0.61	24.2.2007	AS: Sanasto päivitetty
0.62	24.2.2007	LH:tk-kuvaus ja yhteyskomponentti päivitetty
0.63	25.2.2007	MP: Product muuttui + sekalaista viilausta
0.64	25.2.2007	VPK: Tieto-olioihin muutoksia.
0.65	26.2.2007	MP: Muutin hiukan Main-komponenttia.
0.66	26.2.2007	MP: Sanasto taulukkoon, TK-kaavio, yms.
0.67	26.2.2007	VPK: Korjatut komponentti ja tieto-oliokuvat Korjattu FingerprintAuthentication.
0.68	27.2.2007	MP: Yleistä korjailua, päivitystä
0.69	27.2.2007	TP: Muutettu raporttiliittymää
0.70	1.3.2007	MP: Päivitetty Main ja Settings tarkastuksen mukaisesti
0.71	2.3.2007	MP: Korjattu muita osia tarkastuksen mukaisesti (ei kpl 6, kpl 4 osaksi).
0.82	4.3.2007	VPK: Lisätty GUI-komponentin arkkitehtuuri.
0.83	6.3.2007	MP: Lisätty AuthenticationSettings, päivitetty sen mukaisesti, korjailtu virheitä
0.84	6.3.2007	LH: TK-kaavio ja DBConnection-luokkakaavio korjattu.
0.85	6.3.2007	A-P:Authentication komponentin kuvaus ja kaavio päivitetty.
0.90	7.3.2007	MP: Korvattu GUI:n luokkakuvaukset uusilla
0.91	12.3.2007	MP: Pieniä muutoksia toteutuksen perusteella
0.92	13.3.2007	A-P:Authentication komponentin kuvaukset ja luokkakaavio päivitetty. Sanastoa lisätty.
0.93	13.3.2007	LH: virhe tk-kaaviossa korjattu
0.94	26.3.2007	IT: oikoluku, ulkoasua muokattu
1.2	6.5.2007	MP: Luku 6 (raportointi) uusiksi
1.3	6.5.2007	IT: Lopullinen versio

Sisällysluettelo

1 Johdanto.....	1
2 Sanasto.....	2
3 Arkkitehtuurisuunnitelma.....	4
4 Tietokanta.....	6
4.1 Userdata.....	6
4.2 Groupdata.....	8
4.3 Finger.....	8
4.4 Fingerprint.....	8
4.5 Userbalance.....	9
4.6 Rawproduct.....	9
4.7 Productsize.....	9
4.8 Finalproduct.....	10
4.9 Productmix.....	10
4.10 Productin.....	11
4.11 Productout.....	11
4.12 Log.....	11
5 Ostoliittymän arkkitehtuuri.....	13
5.1 Komponentit.....	13
5.1.1 GUI-komponentin alikomponentit.....	15
5.2 Tieto-oliot.....	16
5.2.1 UserBalance.....	17
5.2.2 UserInfo.....	17
5.2.3 Product.....	17
5.2.4 ExportProduct.....	17
5.2.5 ImportProduct.....	17
5.2.6 ProductGroup.....	18
5.2.7 CancelProduct.....	18
5.2.8 GUISettings.....	18
5.2.9 Language.....	18
5.2.10 LanguageEnum.....	18
5.2.11 DBSettings.....	18
5.2.12 FingerprintSettings.....	19
5.2.13 FPData.....	19
5.2.14 AuthenticationSettings.....	19
5.3 Rajapinnat.....	19
5.3.1 UserInterface.....	20
5.3.2 UserInterfaceEvent.....	20
5.3.3 Database.....	20
5.3.4 Configuration.....	20
5.3.5 UserAuthentication.....	20
5.3.6 Fingerprint.....	20
5.3.7 FingerprintEvent.....	21
5.3.8 GUI-komponentin alikomponenttien rajapinnat.....	21
6 Raportointiliittymän arkkitehtuuri.....	24
6.1 PHP:n erityisominaisuuksia.....	24
6.2 Tiedon välitys selaimesta PHP-luokkiin.....	24
6.3 Istunnonhallinta.....	24
6.3.1 Kielenvalinta ja kielen vaihtaminen.....	25
6.4 Perustoiminnallisuus.....	25
6.5 Raportointiliittymän hakemistorakenne.....	26

6.5.1 Alihakemistot	26
6.5.2 Tiedostopuun juuressa olevat tiedostot.....	27
6.6 Raportointiliittymän luokat.....	31
6.6.1 Language.....	31
6.6.2 Connection.....	31
6.6.3 ProductsAndGroupsBean.....	31
6.6.4 UsersBean.....	32
6.6.5 ReportBean.....	32
6.6.6 AuthenticationBean.....	32
6.6.7 Settings.....	32
6.6.8 DBQueries.....	32
7 Ostoliittymän luokkakuvaukset.....	33
7.1 MainProgram-komponentti.....	33
7.1.1 CoffeeTouch.....	33
7.1.2 MainGuiHandler.....	33
7.1.3 MainFingerprintHandler.....	34
7.2 Authentication-komponentti.....	34
7.2.1 Authentication.....	34
7.2.2 ConnectionData.....	34
7.2.3 Base64.....	34
7.3 Settings-komponentti.....	35
7.3.1 Settings.....	35
7.3.2 XML-asetustiedostot.....	35
7.4 Database-komponentti.....	35
7.4.1 DBCommunication.....	35
7.4.2 DBConLostException.....	36
7.4.3 ProductNotFoundException.....	36
7.5 FingerprintAuthentication-komponentti.....	36
7.5.1 FingerprintAuthentication.....	36
7.5.2 FingerprintCache.....	37
7.5.3 FingerprintCacheData.....	37
7.5.4 FingerprintData.....	37
7.5.5 ReaderListener.....	37
7.6 GUI-komponentti.....	38
7.6.1 GuiMain-alikomponentti.....	38
7.6.2 StartView-alikomponentti.....	38
7.6.3 RegisterView-alikomponentti.....	39
7.6.4 ConfigView-alikomponentti.....	40
7.6.5 ShopView-alikomponentti.....	41
7.6.6 ImportView-alikomponentti.....	42
7.6.7 AlertView-alikomponentti.....	43
7.6.8 General-alikomponentti.....	43
7.6.9 ProductLists - alikomponentti.....	44
7.6.10 fingerConf - alikomponentti.....	46
Liite A: Luokkakaaviot.....	47

1 Johdanto

Projektin tarkoituksena on tehdä työyhteisön kahvikassa. Valmis tuote tulee korvaamaan järjestelmän, jossa seinälle kiinnitettyyn paperiin on vedetty viivoja juodun kahvimäärän mukaan. Lisäksi projektin tehtävänä on tutustua sormenjälkitunnistuksen mahdollisuuksiin ja toteuttaa valmis komponentti sormenjälkitunnistusten tekemiseen.

Tässä dokumentissa käydään läpi projektin ohjelmistoarkkitehtuuri tarvittavalla tarkkuudella toteutuksen mahdollistamiseksi. Ensin arkkitehtuuri kuvataan yleisellä tasolla ja sen jälkeen esitellään ohjelmiston tietokannan rakenne.

Tietokannan lisäksi ohjelmistossa on kaksi osajärjestelmää. Näistä monimutkaisempi on asiakaskäyttöliittymä. Sen tehtävänä on mahdollistaa tuotteiden ostaminen (tai oikeammin käytön kirjaaminen, koska rahaa ei liiku) ja tuotteiden tuomisesta saatavan hyvityksen kirjaaminen helppokäyttöisesti. Tämän osajärjestelmän kautta myös lisätään ohjelmistoon uusia käyttäjiä. Asiakaskäyttöliittymä käyttää jo rekisteröityjen käyttäjien tunnistamiseen sormenjälkitunnistusta.

Ohjelmiston toinen osajärjestelmä on raportointikäyttöliittymä, joka puolestaan antaa verkkoselaimella käytettävän, HTML-pohjaisen näkymän ohjelman tietovarastoon. Sen kautta käyttäjät pystyvät seuraamaan omaa kahvinkulutustaan ja koko järjestelmän toimintaa. Lisäksi ylläpitokäyttäjät voivat tämän käyttöliittymän kautta muuttaa tarjolla olevaa tuotevalikoimaa ja poistaa käyttäjiä.

Molempien osajärjestelmien tarkempi arkkitehtuuri on kuvattu omissa luvuissaan. Näissä kuvataan niiden komponentit ja komponenttien väliset rajapinnat. Lisäksi kuvataan rajapintojen käyttämät tieto-oliot. Olio- ja komponenttikuvauksien tekemiseen on käytetty UML-kieltä, ja toteutukseen ja luokkasuunnitteluun tarvittavat tiedot on kirjoitettu auki.

Asiakaskäyttöliittymän komponenteista on lisäksi tarkat luokkakaaviot, joista selviää tarkemmin komponenttien sisäinen toteutus. Raportointikäyttöliittymän toisenlaisesta toteutustavasta johtuen siitä ei ole erillisiä luokkakaavioita, vaan kaikki toteutuksen kanalta tarpeellinen tieto on esitetty komponenttikuvauksen yhteydessä.

2 Sanasto

Käsite	Selitys
Aktiivikäyttö:	Tuote on näkyvissä käyttöliittymässä, kun se on merkattu tietokannassa olevan aktiivikäytössä
asetustiedosto	tietyn muotoinen XML-tiedosto, josta asiakasliittymä hakee asetuksensa
Ostoliittymä / asiakaskäyttöliittymä:	Osajärjestelmä, jonka kautta käyttäjät pystyvät ostamaan ja tuomaan tuotteita kahvihuoneessa
Csv-vienti:	Mahdollisuus tuottaa järjestelmän raporteista tiedosto, jossa kentät on eroteltu toisistaan erotinmerkillä
Event Interface:	Tapahtumarajapinta
Factory-olio:	Olio, joka on tarkoitettu uusien olioiden luomiseen
Griaule GrFinger:	Ulkopuolinen Java-kirjasto sormenjälkitunnistimen käyttöä varten
JDBC-rajapinta:	Java Database Connectivity, tarjoaa metodit tietokantakyselyihin
Käyttäjäsaldo:	Käyttäjän tuoteryhmäkohtainen saldo, joka kertoo tuontien ja vientien välisen tasapainon
Ostettava tuote:	Järjestelmästä ostettavissa olevat tuotteet, esim. kahvi maidolla tai espresso
Raporttiliittymä / raportointikäyttöliittymä:	Osajärjestelmä, joka on tarkoitettu käyttäjien ja tuotteiden hallinnointiin, sekä raporttien tuottamiseen
tuotava tuote	Tuote, jolla käyttäjä voi hyvittää ostoksiaan, esim. pakettillinen kahvia tai tölkillinen maitoa
TKTL:	Helsingin yliopiston tietojenkäsittelytieteen laitos
Tuotehälytys:	Tuotteelle asetettava hälytys, joka ilmaisee tuotteen olevan loppumaisillaan
Tuoteryhmä:	Usean vientituotteen muodostama kokonaisuus, jonka sisältämien tuotteiden arvot eroavat muista tuoteryhmistä
Tuontiarvo:	Järjestelmään tuotavasta tuotteesta omalle tilille saatava hyvitys perusyksikössä (kahvikupeissa)
Tuontitapahtuma:	Käyttäjän järjestelmään tekemä hyvitys, joka kasvattaa hänen saldoaan tuoteryhmässä

Käsite**Selitys****Vakiotuote :**

Käyttäjän itselleen asettama oletustuote

Vientitapahtuma:

Käyttäjän järjestelmällä tekemä ostos, joka vähentää hänen saldoaan tuoteryhmässä

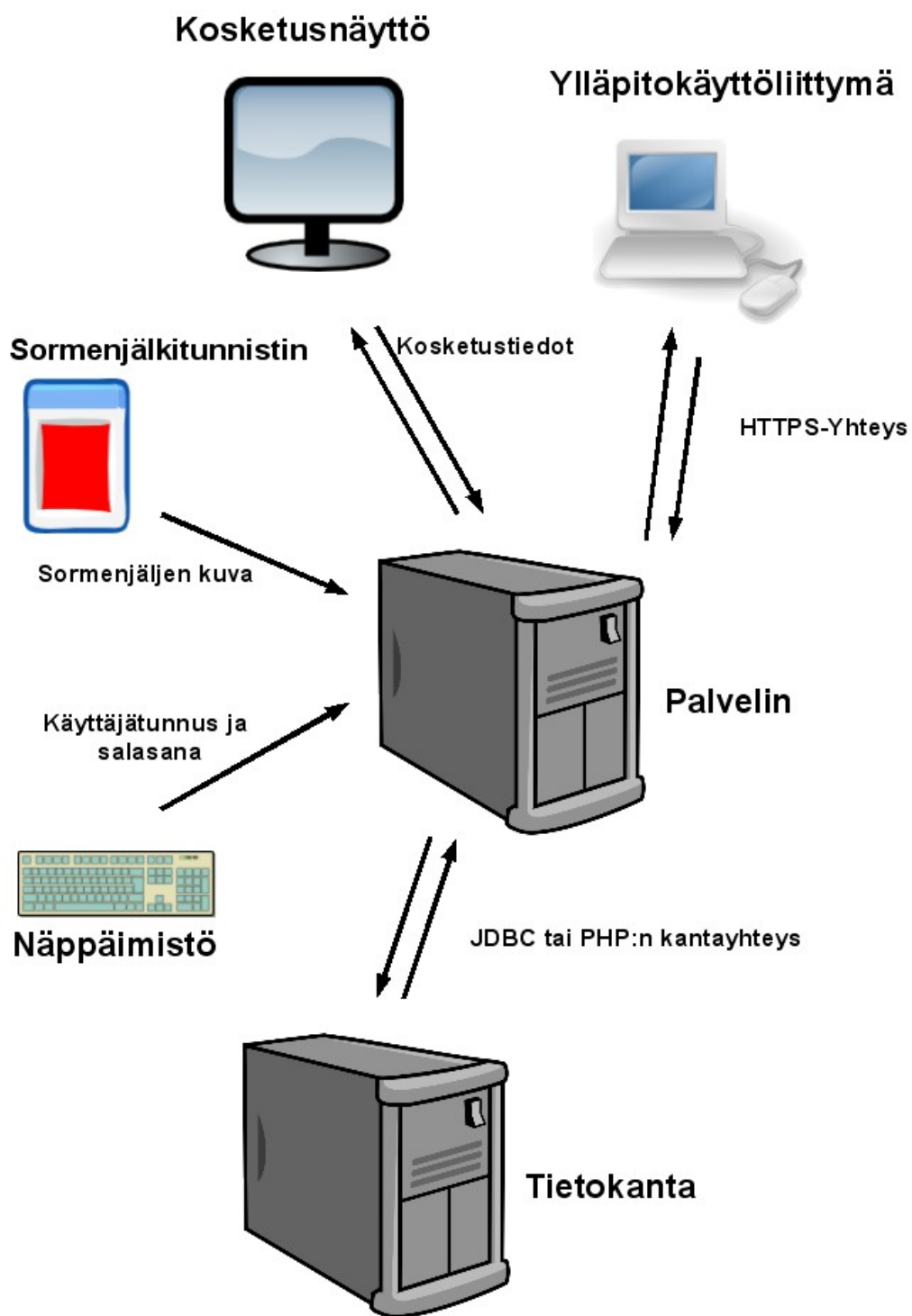
3 Arkkitehtuurisuunnitelma

Ohjelmistossa on kolme osajärjestelmää. Nämä osajärjestelmät ovat tietokanta, asiakasliittymä ja raportointiliittymä. Tietokanta toteutetaan siten, että se ei ole sidoksissa mihinkään määrättyyn tietokantajärjestelmään. Toteutusvaiheessa tämä pyritään varmistamaan siten, että tietokanta toimii PostgreSQL ja Oracle -kantamoottoreiden kanssa.

Asiakasliittymän tehtävänä on toimia kahvihuoneen näkymänä järjestelmään. Sen kautta käyttäjät pystyvät ostamaan tuotteita ja tuomaan tuotteita järjestelmään. Tämän järjestelmän osan toteutukseen käytetään Java-ohjelmointikieltä. Asiakaskäyttöliittymä keskustelee tietokannan lisäksi myös kosketusnäytön (joka korvaa hiiren osoitinlaitteena), sormenjälkilukijan ja näppäimistön kanssa. Kosketusnäyttöä käytetään pääasiallisena syötö ja osoitinlaitteena. Sormenjälkilukijan avulla käyttäjä tunnistetaan järjestelmässä, ja näppäimistön avulla käyttäjä syöttää käyttäjätunnuksensa ja salasanasensa rekisteröimisvaiheessa, jotta käyttäjä saadaan yhdistettyä Tietojenkäsittelytieteen laitoksen käyttäjätietoihin myöhempiä tunnistustarpeita varten.

Raportointikäyttöliittymä on toteutettu PHP-ohjelmointikielellä ja toimii Apache http-palvelimen päällä. Sen avulla käyttäjät pääsevät perumaan väärää ostoksia ja katselemaan raportteja osto- ja tuontitapahtumistaan. Lisäksi raportointikäyttöliittymän kautta järjestelmän ylläpitäjät pystyvät poistamaan järjestelmästä käyttäjiä, ja sekä lisäämään että poistamaan tuoteryhmiä ja tuotteita.

Kuvassa 1. esitellään ohjelmiston toimintaympäristö ja sen eri osat. Lisäksi kerrotaan minkälaisia syötteitä tai protokollia ohjelmiston osien ja toimintaympäristön välillä on. Ohjelmiston tietovarasto sijaitsee tietokantakoneella ja ohjelmakomponentit sijaitsevat palvelimella. Palvelimeen on myös yhdistetty kosketusnäyttö, sormenjälkitunnistin ja näppäimistö. Näiden avulla asiakas ohjaa ohjelmiston toimintaa kahvihuoneessa. Ylläpitokäyttöliittymään liitytään normaalin www-selaimen avulla käyttäen HTTPS-protokollaa.



Kuva 1: Yleiskuva ohjelmiston arkkitehtuurista.

4 Tietokanta

Tässä luvussa kuvataan ohjelmiston tietokanta. Siinä on esitelty jokainen taulu ja taulujen kentät. Tietokanta on suunniteltu mahdollisimman kantariippumattomaksi, mutta tietokannan yksinkertaistamiseksi käytetään joitakin tietokantojen ominaisuuksia, jotka täytyy toteuttaa kantariippuvaisesti.

Tästä kantariippuvuudesta johtuen on laadittu erilliset kannanluontilauseet kohdetietokannoiksi tarkoitetuille PostgreSQL:lle ja Oraclelle. Normaalit haut ja lisäykset pyritään toteuttamaan siten, että niihin ei tarvitse tehdä muutoksia, jos tietokantaa vaihdetaan. Jos lauseita kuitenkin joudutaan muuttamaan, ne on tallennettu ohjelmiston asetustiedostoihin, joissa niiden editoiminen on helppoa, ja ne ovat heti käytettävissä ohjelman uudelleenkäynnistyksen jälkeen.

Kuvassa 2 esitellään myös tietokantakaavio, josta on helpompi saada käsitys taulujen riippuvaisuussuhteista.

Postgresin boolean tietotyyppi on Oraclessa korvattu tietotyypillä char(1) check (ud_admin in ('T', 'F')).

4.1 Userdata

Taulu sisältää ohjelman käyttäjätiedot. Näihin tietoihin kuuluu esimerkiksi käyttäjätunnus ja käyttäjän etu- ja sukunimi. Lisäksi käyttäjästä talletetaan hänen oletuskielensä ja vakiotuotteensa. Myös tieto siitä, onko käyttäjä järjestelmän ylläpitäjä vai ei, on tallennettu tähän tauluun, kenttään ud_admin.

Kentän nimi	Tyyppi	Not null	Kuvaus
ud_username	Varchar(8)	X	TKTL:n käyttäjätunnus, pääavain
ud_givenname	Varchar(32)	X	Käyttäjän etunimi
ud_familyname	Varchar(64)	X	Käyttäjän sukunimi
ud_regularportion	Int	X	Käyttäjän valitsema vakiotuote. Viite finalproduct-tauluun
ud_language	Varchar(3)	X	Käyttäjän valitsema vakiokieli.
ud_admin	Boolean		Tieto käyttäjän ylläpitäjäoikeuksista

4.2 Groupdata

Taulu sisältää tiedot ohjelman tuoteryhmistä. Jokaisella tuoteryhmällä on ID-numero, jonka arvoa tietokanta kasvattaa automaattisesti, kun tauluun lisätään rivejä. Muita taulun tietoja ovat tuoteryhmän yleinen saldo, joka kertoo erotuksen siitä, paljonko tuoteryhmään on tuotu tuotteita, ja paljonko niitä on käytetty vakioannoksina. Lisäksi tuoteryhmästä on sen nimi suomeksi, ruotsiksi ja englanniksi. Käytöstä poisto päiväyksellinen tuoteryhmä ei näy käyttäjälle asiakaskäyttöliittymässä. Jos käytöstä poisto päiväys on tyhjä, on tuoteryhmä aktiivikäytössä.

Kentän nimi	Tyyppi	Not null	Kuvaus
gd_groupdataid	Int	X	Pääavain, kasvatetaan automaattisesti
gd_balance	Double	X	Tuoteryhmän yhteinen saldo
gd_finnname	Varchar(50)	X	Tuoteryhmän nimi suomeksi
gd_engname	Varchar(50)	X	Tuoteryhmän nimi englanniksi
gd_swename	Varchar(50)	X	Tuoteryhmän nimi ruotsiksi
gd_deacttime	Datetime		Käytöstä poisto päiväys

4.3 Finger

Taulu sisältää tiedon siitä, kenelle mikäkin sormenjälki kuuluu ja mihin sormeen se on liitetty.

Kentän nimi	Tyyppi	Not null	Kuvaus
f_finger	Int	X	0-4, eli mikä sormi on kyseessä
f_fpid	Int	X	Viite fingerprinttiin.
f_username	Varchar(8)		Käyttäjä, viiteavain userdata-tauluun

4.4 Fingerprint

Taulu sisältää sormenjäljen tiedot. Taulussa on lisäyksien myötä automaattisesti kasvava pääavain, joka yksilöi jokaisen tallennetun sormenjäljen. Sormenjäljistä tallennetaan tunnistustieto ja tieto sen laadusta.

Kentän nimi	Tyyppi	Not null	Kuvaus
fp_id	Int	X	Pääavain, kasvatetaan automaattisesti
fp_template	Varchar(1000)	X	Sormenjäljen tunnistustieto
fp_quality	Int	X	Tunnistustiedon laatu

4.5 Userbalance

Taulussa on tiedot käyttäjien saldoista eri tuoteryhmissä. Taulua päivitetään, kun käyttäjä tuo tuotteita järjestelmään tai vie niitä järjestelmästä. Taulussa on kaksi viiteavainta, toinen käyttäjään ja toinen tuoteryhmään. Lisäksi on kyseisen ryhmän saldotieto.

Kentän nimi	Tyyppi	Not null	Kuvaus
ub_username	Varchar(8)	X	Käyttäjä, viite userdata-tauluun
ub_groupid	Int	X	Tuoteryhmä, viite groupdata-tauluun
ub_balance	Double	X	Käyttäjän saldo kyseisessä tuoteryhmässä

4.6 Rawproduct

Taulu sisältää tiedot tuotavista tuotteista. Pääavain yksilöi tuotteet tietokannassa ja lisäksi jokainen tuote on yhdistetty johonkin tuoteryhmään. Taulussa on tieto siitä, onko tuotteella loppumishälytys ja mitä laatua tuotekoot ovat (l, g, kg, jne.). Tuotteen arvon laskemiseksi taulussa on tuotteen perustuontiarvo. Tuotavien tuotekokojen tuontiarvot lasketaan kertoimen ja perustuontiarvon mukaan. Käytöstä poisto päiväyksellinen tuote ei näy käyttäjille asiakaskäyttöliittymässä. Jos päivämäärä on tyhjä, tuote on aktiivikäytössä.

Kentän nimi	Tyyppi	Not null	Kuvaus
raw_rawproductid	Int	X	Pääavain, kasvatetaan automaattisesti
raw_groupid	Int	X	Tuoteryhmä, viite groupdata-tauluun
raw_picture	blob		Sisältää tuotteen kuvan.
raw_value	Double	X	Tuotteen tuontiarvo
raw_alert	Boolean		Tieto onko tuotteessa loppumishälytys
raw_sizeunit	Varchar(4)	X	Tuotteen koon määre (esim. kg, l, g)
raw_finname	Varchar(50)	X	Tuotteen nimi suomeksi
raw_engname	Varchar(50)	X	Tuotteen nimi englanniksi
raw_swename	Varchar(50)	X	Tuotteen nimi ruotsiksi
raw_deacttime	Datetime		Käytöstä poisto päiväys

4.7 Productsize

Taulu sisältää tiedot yhdestä tuotavaan tuotteeseen liittyvästä pakkauskoosta. Taulussa on pääavainkenttä, jonka luku kasvaa automaattisesti rivilisäyksen yhteydessä. Lisäksi taulussa on tieto kertoimesta, jolla tietyn koon tuontiarvo lasketaan tuotteen perustuontiarvosta, ja paketin tuotekoko. Käytöstä poisto päiväyksellinen tuotekoko ei näy käyttäjille asiakaskäyttöliittymässä. Jos päivämäärä on tyhjä, on tuotekoko aktiivikäytössä.

Kentän nimi	Tyyppi	Not null	Kuvaus
ps_productsizeid	Int	X	Pääavain, kasvatetaan automaattisesti
ps_value	Double	X	Tuotteen arvo kertoimena. Varsinainen arvo lasketaan kertomalla sen tuotteen arvo, johon tuotekoko liittyy, tällä kertoimella.
ps_size	Double	X	Paketin tuontikoko
ps_rawproductid	Int	X	Tuote, viite rawproduct-tauluun
ps_deacttime	Datetime		Käytöstä poisto päiväys

4.8 Finalproduct

Taulu sisältää tiedot ostettavista tuotteista. Jokaisella tuotteella on sen yksiselitteisesti tunnistava pääavain, jonka kanta luo tuotteen lisäämisen yhteydessä. Lisäksi tuotteessa on tieto siitä, mihin tuoteryhmään se kuuluu, ja muut tuotteen tiedot, kuten sen ostoarvo, kuva ja nimi suomeksi, ruotsiksi ja englanniksi. Käytöstä poisto päiväksellinen tuote ei näy käyttäjille asiakaskäyttöliittymässä. Jos päivämäärä on tyhjä, on tuote aktiivikäytössä.

Kentän nimi	Tyyppi	Not null	Kuvaus
final_finalproductid	Int	X	Pääavain, kasvatetaan automaattisesti
final_groupid	Int	X	Tuoteryhmä, viite groupdata-tauluun
final_value	Double	X	Tuotteen hinta (normaalisti 1)
final_picture	Blob		Kuva tuotteesta
final_finname	Varchar(50)	X	Tuotteen nimi suomeksi
final_engname	Varchar(50)	X	Tuotteen nimi englanniksi
final_swename	Varchar(50)	X	Tuotteen nimi ruotsiksi
final_deacttime	Datetime		Käytöstä poisto päiväys

4.9 Productmix

Taulussa kuvataan mitä raakatuotteita, eli tuotavia tuotteita (esim. kahvi, maito), johonkin valmiiseen tuotteeseen kuuluu. Taulussa on siis viittaukset kummankin tyyppisten tuotteiden pääavaimiin.

Kentän nimi	Tyyppi	Not null	Kuvaus
mix_rawproductid	Int	X	Raakatuote, viite rawproduct-tauluun
mix_finalproductid	Int	X	Valmistuote, viite finalproduct-tauluun

4.10 Productin

Taulu sisältää tapahtumakirjaukset käyttäjien tuonneista. Sen perusteella tuotavien tuotteiden listaa voidaan muokata sopivaan järjestykseen käyttäjälle. Lisäksi tämän perusteella voidaan tehdä raportteja käyttäjien tuonneista ja tuontitottumuksista.

Kentän nimi	Tyyppi	Not null	Kuvaus
in_productinid	Int	X	Pääavain, kasvatetaan automaattisesti
in_username	Varchar(8)	X	Käyttäjä
in_productsizeid	Int	X	Tuotekoko, viite productsize-tauluun
in_rawproductid	Int	X	Tuote, viite rawproduct-tauluun
in_acttime	Datetime	X	Tapahtuma-aika
in_value	Double	X	Tapahtuman arvo, saldoihin tehty muutos

4.11 Productout

Taulu sisältää tapahtumakirjaukset käyttäjien ostoista. Tästä taulusta lasketaan kuinka monta ostoa käyttäjä on tehnyt eri tuotteista, ja tämän tiedon perusteella järjestetään ostettavat tuotteet sellaiseen järjestykseen, että käyttäjän useimmin ostama tuote näkyy käyttöliittymässä listan ensimmäisenä ja harvimminkin ostama viimeisenä. Lisäksi taulu mahdollistaa tuotteiden ostojen perumiset ja erilaisten raporttien teon käyttäjien käyttötottumuksista.

Kentän nimi	Tyyppi	Not null	Kuvaus
out_productoutid	Int	X	Pääavain, kasvatetaan automaattisesti
out_username	Varchar(8)	X	Käyttäjä
out_finalproductid	Int	X	Tuote, viite finalproduct-tauluun
out_acttime	Datetime	X	Tapahtuma-aika
out_value	Double	X	Tapahtuman arvo, saldoihin tehty muutos
out_cancelevent	Int		Mahdollinen peruttava tapahtuma, viite productout-tauluun. Jos tyhjä niin normaalitapahtuma, jos olemassa niin peruutustapahtuma jolla korvataan viitattu tapahtuma.

4.12 Log

Taulu sisältää tapahtumakirjauksia arvomuutoksista tuonti- ja vientituotteissa, pakkauskoissa sekä tuoteryhmissä. Taulussa lukee log_type-kentässä tekstimuodossa mihin tauluun viitataan, sekä log_tableid-kentässä kyseisen taulun tunnus. Tauluun talletetaan myös tieto kuka teki muutoksen ja milloin.

Kentän nimi	Tyyppi	Not null	Kuvaus
log_id	Int	X	Pääavain, kasvatetaan automaattisesti
log_type	Varchar(5)	X	Tekstimuotoinen tieto mihin tauluun viitataan: raw, final, size tai group
log_tableid	Int	X	Viitattavan taulun tunnus
log_oldvalue	Double	X	Muutetun vanha arvo
log_newvalue	Double	X	Muutetun uusi arvo
log_acttime	Datetime	X	Tapahtuma-aika
log_username	Varchar(8)	X	Käyttäjä

5 Ostoliittymän arkkitehtuuri

Tässä luvussa esitellään ostoliittymän komponentit, tieto-oliot ja rajapinnat. Tarkemmat kuvaukset löytyvät ohjelman mukana toimitettavasta Javadocista ja myös osoitteesta: <http://www.cs.helsinki.fi/group/kasi/doc/index.html>.

5.1 Komponentit

Asiakasliittymässä on kuusi komponenttia, jotka on esitelty kuvassa 3. Näistä kuudesta MainProgram-komponentti hoitaa kaiken tiedonkulun muiden asiakasliittymän osien kesken. Loput viisi puolestaan hoitavat ohjelman toimintojen ja ulkoisten rajapintojen toteuttamisen. Nämä viisi muuta komponenttia ovat GUI, Authentication, Settings, Database ja FingerprintAuthentication.

MainProgram-komponentti sovittaa muut erilliset komponentit toisiinsa ja helpottaa niiden toteuttamista toisistaan erillisinä osina. Muiden apuna olevat Database ja Settings-komponenteista käyttävät kuitenkin esimerkiksi muiden komponenttien määrittelemiä asetusluokkia asetusten välittämiseen, joten täysin erillisiä ne eivät ole. Database-komponentin rajapinnoissa on suoraan jokaiselle kantahaulle tehty oma metodi, joka palauttaa kyselyn antaman tiedon oikeassa muodossa.

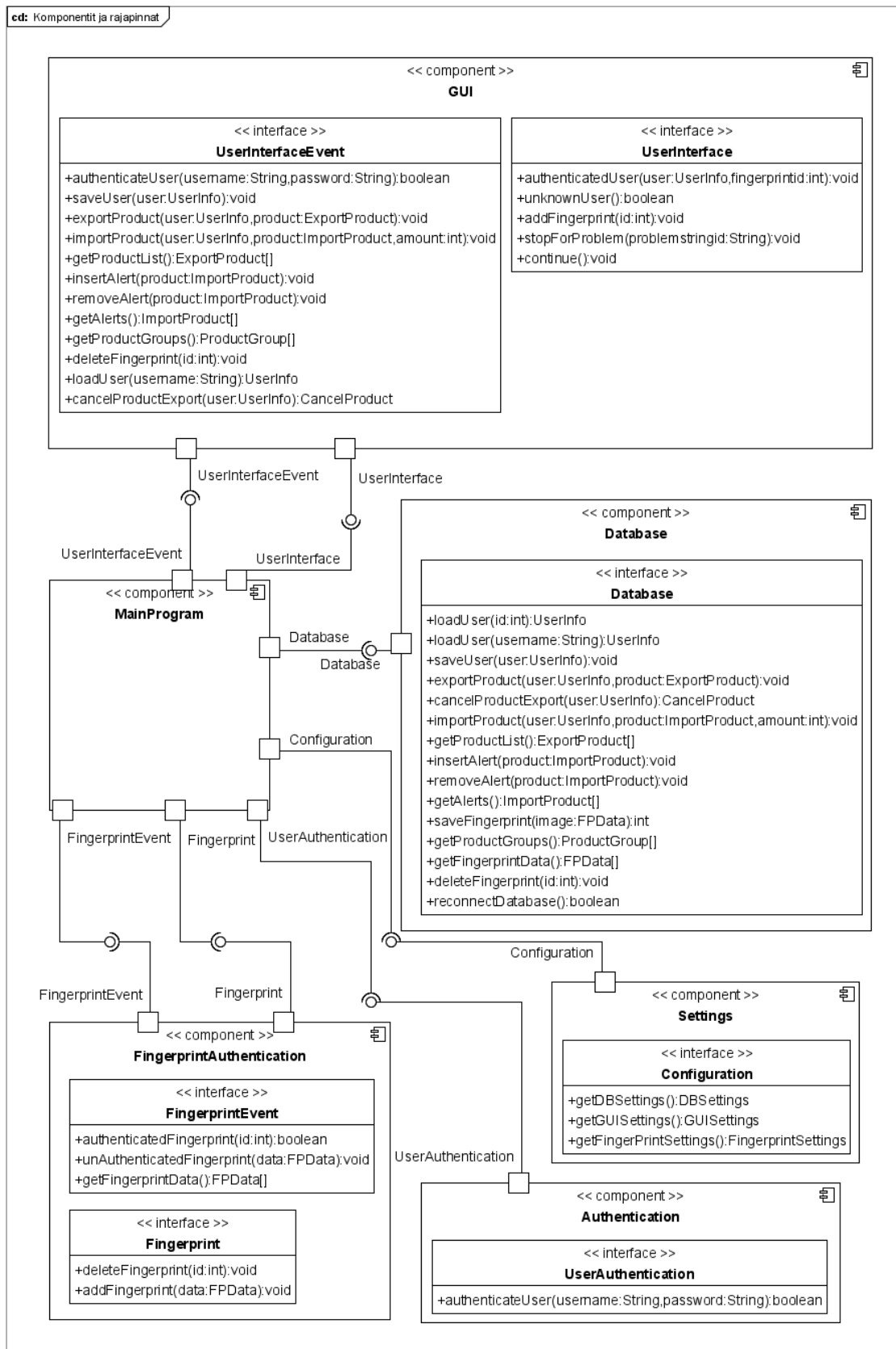
GUI-komponentti on ohjelman suurin ja monimutkaisin osa. Sen tehtävänä on toteuttaa käyttäjälle näkyvä ohjelman käyttöliittymä ja muuntaa käyttäjän ohjelmalle antamat käskyt sellaiseen muotoon, että ohjelman muiden osien on niitä helppo käsitellä. Lisäksi GUI-komponentti pitää kirjaa ohjelman sen hetkisestä tilasta ja reagoi sen mukaan myös MainProgram-komponentin kautta tuleviin sormenjälkien tunnistustapahtumiin.

Authentication-komponentin tehtävä on liittyä käyttöjärjestelmään tai muuhun vastaavaan käyttäjätietojen lähteeseen, jonka kautta käyttäjä pystytään tunnistamaan (TKTL:n) käyttäjätunnuksen ja salasanan avulla.

Settings-komponentin tehtävänä on hakea ohjelman tarvitsemat asetukset tiedostojärjestelmässä sijaitsevasta asetustiedostosta. Asetustiedoston sijainti voidaan määrittää kommentoriviltä, tai jos sitä ei ole määritelty, komponentti käyttää ohjelmakoodissa lukevaa vakiopaikkaa tiedoston sijainnille. Lisäksi komponentti käyttää GUI, Database ja FingerprintAuthentication-komponenttien määrittämiä asetusluokkia.

Database-komponentin tehtävänä on toimia ohjelman ja ohjelmiston tietokannan välisenä rajapintana. Sen rajapinnassa on jokaiselle kantahaulle tehty oma metodi, joka palauttaa kyselyn antaman tiedon oikeassa muodossa. Komponentti ottaa yhteyttä tietokantaan JDBC-rajapinnan kautta. Tietokannasta haettava tieto asetetaan myöhemmin tässä dokumentissa esiteltäviin tieto-olioihin.

FingerprintAuthentication-komponentti kuuntelee ohjelman käyttämää sormenjälkilukijaa. Tätä tarkoitusta varten se käyttää hyödykseen Griaulen GrFingerJava-kirjastoa ja -ajuria. Komponentti saa sormenjälkitunnistimelta ilmoituksia uudesta luetusta sormenjäljestä. Tämän jälkeen se vertaa luettua sormenjälkeä sillä varastossa oleviin sormenjälkitunnistustietoihin, ja jos tunnistus onnistuu, lähettää MainProgram-komponentille sormenjäljen ID-numeron. Muussa tapauksessa lähetetään sormenjäljen tunnistustieto sen mahdollista tallentamista varten.



Kuva 3: Komponentit ja rajapinnat

5.1.1 GUI-komponentin alikomponentit

Ohjelman GUI-komponentti on jaettu kuvan 4. mukaisesti 8-ali-komponenttiin. Niistä GuiMain-komponentti on GUI-komponentin ydin. General-komponentissa on muiden komponenttien yhdessä käyttämiä luokkia. Muut komponentit on puolestaan jaettu sen mukaan, että minkä ohjelman näytön ne toteuttavat. Tämä mahdollistaa uusien näyttöjen lisäämisen tarvittaessa ohjelmaan mahdollisimman pienin muutoksin.

GUI-Komponentin kaikki "älykäs"-toiminnallisuus on toteutettu GuiMain-ali-komponentissa. Tämän alikomponentin tehtävänä on olla GUI-komponentin muiden osien rajapintana itse pääohjelmaan. Se toteuttaa ViewEventInterface rajapinnan ja pitää sisällään käyttöliittymän tilatiedot.

StartView-ali-komponentti toteuttaa käyttäjälle ensimmäisenä näkyvän osan ohjelmasta. Tämän komponentin toteuttamassa näkymässä käyttäjä voi aloittaa rekisteröitymisen ohjelman käyttäjäksi, aloittaa tuotteen loppumisesta kertovan varoituksen lisäämisen, poistaa tuotteen loppumisen varoitus tai sormenjäljen antamalla kirjautua sisään ohjelmaan. Toteuttaa StartViewInterface-rajapinnan.

RegisterView-ali-komponentti mahdollistaa uuden käyttäjän rekisteröitymisen. Siinä käyttäjä kirjoittaa TKTL:n käyttäjätunnuksen ja salasanaansa. Jos nämä hyväksytään, niin sen jälkeen käyttäjälle annetaan mahdollisuus antaa itselleen etunimi ja sukunimi järjestelmään. Näiden tietojen antamisen jälkeen käyttäjä siirretään normaaliin ConfigView-ali-komponentin näkymään. Toteuttaa RegisterViewInterface-rajapinnan

ConfigView-ali-komponentin tuottama näkymä mahdollistaa käyttäjän käyttäjätietojen muuttamisen ohjelmassa. Sen avulla käyttäjä voi vaihtaa tuotetta, joka häneltä normaalisti veloitetaan ostoliittymässä. Ja lisää itselleen uusia tai korvata vanhoja sormenjälkiä, jotka mahdollistavat käyttäjän sisäänkirjautumisen sormenjälkilukijan avulla. ConfigView-ali-komponentin näkymästä käyttäjä voi siirtyä tuotteiden tuonti tai ostonäkymään. Alikomponentti toteuttaa ConfigViewInterface-rajapinnan.

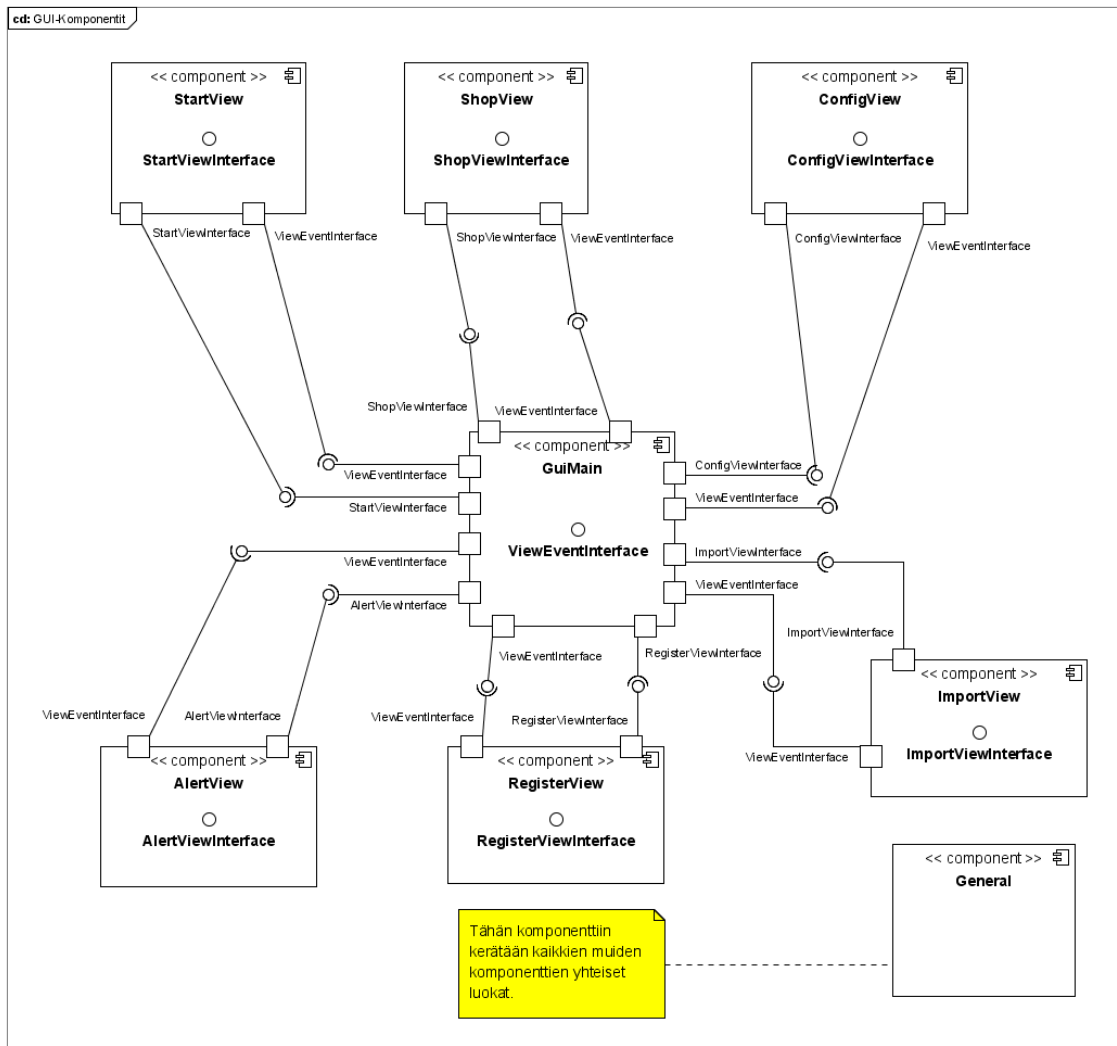
ShopView-ali-komponentti tuottaa näkymän, jonka kautta käyttäjä voi merkitä ostaneensa tuotteita järjestelmästä. Näkymä tulee näkyville, kun käyttäjä joko syöttää ohjelman alkuruudussa sormenjälkensä tai siirtyy siihen jostain muusta näkymästä nappia painamalla. Jos käyttäjä on syöttänyt sormenjälkensä alkuruudussa ja ei tee mitään muuta tämän jälkeen tai kirjautuu ulos järjestelmästä. Veloitetaan häneltä asetuksissa määritelty vakiotuote. Tämän komponentin näkymästä käyttäjä voi siirtyä tuotteiden tuonti tai omat tiedot näkymiin. Alikomponentti toteuttaa ShopViewInterface-rajapinnan.

ImportView-ali-komponentin tuottaman näkymän kautta käyttäjät pystyvät merkitsemään tiedon siitä, että ovat tuoneet tuotteita. Nämä merkinnät lisäävät sitten käyttäjän saldoa järjestelmässä. Näkymästä pääsee osto ja omat tiedot näkymiin. Alikomponentti toteuttaa ImportViewInterface-rajapinnan.

AlertView-ali-komponentti tuottaa näkymän, jossa käyttäjä pystyy lisäämään varoituksen jokin tuotavan tuotteen loppumisesta. Käyttäjän lisättyä varoituksen ohjelma siirtyy takaisin aloitusnäkymään. Alikomponentti toteuttaa AlertViewInterface-rajapinnan.

General-ali-komponenttiin on lisätty muiden komponenttien yhteisesti tarvitsemia luokkia. Näitä ovat esimerkiksi NPanel ja LanguageButton, joita monet alikomponenteista

käyttävät. Komponentilla ei ole omaa toiminnallisuutta ja se toimii lähinnä luokkavarastona.



Kuva 4: GUI-Komponentin alikomponentit

5.2 Tieto-oliot

Asiakasliittymän eri osien välistä tiedonkulkua varten ohjelmassa on tieto-olioita. Näiden tehtävänä on tiedon välittäminen komponentista toiseen sellaisessa muodossa, että tietoon liittyvät riippuvuussuhteet pysyvät helposti hallittavina. Ohjelmasta löytyvät seuraavissa luvuissa esiteltävät tieto-oliot. Tieto-olioiden luokat on laitettu niille läheisen järjestelmän komponentin kanssa samaan pakettiin, jotka on mainittu kunkin luokan kohdalla.

Tieto-oliot on esitelty UML-luokkakaaviona dokumentin liitteessä (liite A, kuva 1).

5.2.1 UserBalance

Paketti: gui

Luokka on UserInfo-luokan apuluokka, joka pitää sisällään käyttäjän saldon järjestelmässä olevassa tuoteryhmässä.

5.2.2 UserInfo

Paketti: gui

Luokan tehtävänä on välittää käyttäjään liittyvä tieto Database-komponentilta GUI-komponentille. Luokka sisältää kaiken tarpeellisen tiedon käyttäjästä, hänen tilinsä tilanteesta ja tuotteista, joita käyttäjä voi ostaa. Luokkaa käytetään myös apuna, kun uusi käyttäjä lisätään tai käyttäjän tietoihin halutaan tehdä muutoksia. Tällöin luokan avulla siirretään tieto siitä, että mitä asetuksia käyttäjästä on muutettu Database-komponentille, joka tallettaa tiedot kantaan.

5.2.3 Product

Paketti: main

Luokka on ExportProduct ja ImportProduct -luokkien ylikuokka. Se sisältää tuotteen perustiedot, jotka ovat molemmille tuotteille yhtenevät. Näitä ovat esimerkiksi tuotteen id-numero, nimi eri kielillä ja kuva.

5.2.4 ExportProduct

Paketti: main

Tämä luokka pitää sisällään tiedon yksittäisistä tuotteista, joita käyttäjät voivat järjestelmästä ostaa. Se perii suurimman osan toiminnallisuudestaan Product-luokalta ja lisää näihin ominaisuuksiin mahdollisuuden tallettaa kuinka paljon tuotteen käyttäminen vähentää käyttäjän saldoa.

5.2.5 ImportProduct

Paketti: main

Luokka perii Product-luokan. Sen avulla siirretään tietoa siitä minkälaisia tuotteita käyttäjät voivat järjestelmään tuoda. Tuodut tuotteet ovat jokainen luokan ilmentymiä. Luokassa on tieto siitä, mitä kokoja tuotteita voi tuoda (amounts), ja minkälainen arvo (values) näillä tuoduilla koilla on. Lisäksi luokka pitää kirjaa mitä määrettä tuotava tuote on, eli onko tuotteen määrä esimerkiksi litroissa (l) vai grammoissa (g).

5.2.6 ProductGroup

Paketti: main

Luokka kerää yhteen kaikki tiettyyn ryhmään kuuluvat tuotteet, joita järjestelmään voi tuoda.

5.2.7 CancelProduct

Paketti: main

Luokka sisältää tiedon viimeisimmästä ostosta, joka on peruttu. Tähän tarkoitukseen luokan luomaan ilmentymään tallennetaan kyseisen oston tapahtumapäivä ja ostetun tuotteen tiedot.

5.2.8 GUISettings

Paketti: gui

Nimensä mukaisesti luokka pitää sisällään käyttöliittymän asetukset. Sen avulla käyttöliittymä saa Settings-komponentilta tiedon muuttujista, joita järjestelmän hallinnoja voi muuttaa. Lisäksi luokkaan on Language-luokan avulla tallennettu käyttöliittymän käytämät kielet.

5.2.9 Language

Paketti: gui

Luokka on GUISettings-luokan apuluokka, jonka ilmentymiin järjestelmä tallentaa kaikkien käyttöliittymän tarvitsemien tekstien eri kieliversiot.

5.2.10 LanguageEnum

Paketti: gui.settings

Luokka kierrättää kielten tietoja ja pitää niiden integer arvot sisällään.

5.2.11 DBSettings

Paketti: database

Luokka pitää sisällään Database-komponentin asetukset. Nämä asetukset pitävät sisäl-

lään tiedon siitä miten tietokantaan saa yhteyden ja mikä on suurin sallittu aika edellisen vientitapahtuman perumiselle. Luokkaan varastoidaan myös kantaan tehtävien kyselyiden SQL-lauseet.

5.2.12 FingerprintSettings

Paketti: fingerprintauthentication

Tässä luokassa on tiedot sormenjälkilukijan asetuksista. Näitä asetuksia ovat lukijan sormenjäljille varaaman välimuistin koko ja polut GrFingerJavan tarvitsemiin lisenssi-tiedostoon ja natiivikirjastoihin.

5.2.13 FPData

Paketti: fingerprintauthentication

Tämän tieto-olion avulla Database ja FingerprintAuthentication -komponentit siirtävät sormenjälkien tunnistustietoja toistensa välillä. Olioon on tallennettuna tunnistustieto tavutaulukkona ja kokonaislukuna tallennetun tiedon laatu, sekä lisäksi sormenjäljen tunnusnumero.

5.2.14 AuthenticationSettings

Paketti: authentication

Tämän luokan avulla välitetään autentikointikomponentille tieto URL-osoitteesta, josta autentikointi tehdään.

5.3 Rajapinnat

Ohjelman komponenttien välissä on kuusi komponenttien välisen liikennöinnin hoitavaa rajapintaa. Näiden lisäksi komponenttien pääluokilla on konstruktorimetodit, jotka hoitavat asetus-tieto-olioiden ja tapahtumarajapintojen (eventinterface) rekisteröimisen luokille. Tapahtumarajapintojen avulla komponentit pääsevät ilmoittamaan muulle järjestelmälle käyttäjän suorittaneen jonkin toiminnon. Niiden lisäksi komponenteilla on normaaleja rajapintoja, joiden kautta komponentin tarjoamia palveluita voi käyttää.

Rajapinnat on esitelty aiemmin kuvassa 3, ja seuraavissa luvuissa ne kuvataan tarkemmin.

5.3.1 **UserInterface**

Rajapinnan tehtävänä on antaa mahdollisuus kutsua käyttöliittymäkomponentin tarjoamia palveluita. Rajapinta antaa mahdollisuuden kutsua käyttöliittymää, kun käyttäjä on pistänyt sormensa sormenjälkilukijaan. Käyttöliittymää voidaan kutsua myös, jos ohjelman suorituksessa tapahtuu jokin virhe, ja käyttöliittymä halutaan jädäyttää väliaikaisesti näyttämään virheilmoitusta.

5.3.2 **UserInterfaceEvent**

Rajapinnan tehtävänä on välittää käyttäjän tekemien valintojen vaikutukset käyttöliittymästä muulle ohjelmalle. Tämän kaltaisia tapahtumia ovat esimerkiksi ostosten teko tai tuotteiden tuominen, sekä käyttäjän omien asetusten muuttaminen.

Luokka nimetty hieman huonosti, koska kyseessä on tosiaan rajapinta eikä tapahtuma.

5.3.3 **Database**

Rajapinta tarjoaa tarvittavat metodit tiedon hakemiseen tietokannasta ja sen tallettamiseen sinne. Monet järjestelmän käyttämisestä tieto-olioista luodaan rajapinnan toteuttavassa komponentissa tietokannan tiedoista.

5.3.4 **Configuration**

Rajapinnan tehtävänä on antaa näkymä komponenttiin, jonka tehtävä on lukea asetukset ohjelman asetustiedostosta. Komponentti muuttaa asetukset asetuseroiksi, jotka MainProgram-komponentti noutaa ja välittää ohjelman muille osille.

5.3.5 **UserAuthentication**

Rajapinnan tehtävänä on antaa normaaliin käyttäjätunnistukseen yhtenäinen rajapinta. Rajapinnan avulla käyttäjät voivat tunnistautua TKTL:n normaalilla käyttäjätunnus ja salasana -parillaan.

5.3.6 **Fingerprint**

Rajapinnan avulla ohjelma voi antaa sormenjälkitunnistuskomponentille käskyjä.

5.3.7 FingerprintEvent

Rajapinnan tehtävänä on kuunnella FingerprintAuthentication-komponentilta tulevia tapahtumia ja välittää näitä eteenpäin muiden ohjelman osien käyttöön. Rajapintaan tulee pääasiassa tapahtumia, kun käyttäjä laittaa sormen sormenjälkilukijaan ja ohjelmaa käynnistettäessä.

5.3.8 GUI-komponentin alikomponenttien rajapinnat

GUI-komponentilla on useampia sisäisten osien välillä olevia rajapintoja. Näiden tehtävänä on mahdollistaa eri komponenttien välinen tiedonsiirto ja samalla selkeyttää, sekä jakaa GUI-komponentin rakennetta. Seuraavaksi kuvaamme rajapintojen rakenteen ja kerromme tarkemmin mitä ne ja niiden metodikutsut tekevät.

5.3.8.1 ViewEventInterface

Rajapinta tarjoaa tarvittavat metodit GUI-Komponentin muille alikomponenteille, jotta ne voivat kutsua GuiMain-komponenttia ja sen avulla muita sen aliosia tai muuta ohjelmaa. Rajapinnasta on esitelty kaikki sen tarjoamat metodit. Kaikki GUI-komponentin osat käyttävät tätä rajapintaa.

5.3.8.2 ViewEvent (rajapinta)

Luokka on kaikkien muiden komponenttikohtaisten rajapintaluokkien ylituokka. Se määrittelee kaikille yhteiset metodit, joita GuiMain tarvittaessa kutsuu. Yksikään komponentti ei suoraan toteuta tätä rajapintaa. Määriteltävät metodit ovat komponenttien tarjoamien näyttöjen näyttämiseen ja piilottamiseen, sekä kielenvaihtoon liittyviä.

5.3.8.3 TimerEvent (rajapinta)

Rajapinnan avulla paketin sisällä välitetään tieto aikakatkaisusta.

5.3.8.4 AlertEvent (rajapinta)

Rajapinnan avulla paketin sisällä välitetään tieto hälytyksien lisäämisestä ja poistamisesta.

5.3.8.5 FingerEvent (rajapinta)

Rajapinnan avulla voidaan kertoa kun sormenjälkeä halutaan muokata.

5.3.8.6 RegisterEvent (rajapinta)

Antaa mahdollisuuden kertoa rekisteröitymisyrityksestä ja nimen antamisesta.

5.3.8.7 BuyEvent (rajapinta)

Rajapinta, jonka avulla kerrotaan, että tuote on hyväksytty vietäväksi.

5.3.8.8 CancelEvent (rajapinta)

Rajapinta, jonka avulla kerrotaan, että käyttäjä tarvitsee lisää aikaa miettimiseen.

5.3.8.9 ImportEvent (rajapinta)

Rajapinnan avulla kerrotaan kun jokin tuote halutaan hyväksyä tuoduksi.

5.3.8.10 ViewEvent (rajapinta)

Rajapinnan avulla NPanelin navigointinapit kutsuvat NPanelia ja kertovat, että niitä on painettu.

5.3.8.11 CancelEvent (rajapinta)

Rajapinnan avulla kerrotaan, että käyttäjä haluaa perua ostoksensa.

5.3.8.12 ProductChangeEvent (rajapinta)

Rajapintaa käytetään kertomaan, että käyttäjä on valinnut jonkin tuotteen.

5.3.8.13 RegisterViewInterface

Tämä on RegisterView-komponentin rajapinta, jonka kautta sitä hallitaan. Se perii ViewInterface-rajapinnan metodit ja lisäksi määrittelee komponenttikohtaisia metodeja.

5.3.8.14 ShopViewInterface

Tämä on ShopView-komponentin rajapinta, jonka avulla sille annetaan tiedot käyttäjän ostettavissa olevista tuotteista ja tieto myös käyttäjän saldoista eri tuoteryhmissä. Se myös perii ViewInterface-rajapinnan metodit.

5.3.8.15 ImportViewIterface

Tämä on ImportView-komponentin rajapinta, joka myös perii ViewInterface-rajapinnan toiminnallisuuden. Se lisäksi määrittää omat metodinsa tuotavien tuotteiden ja käyttäjän saldotietojen syöttämiseen komponentille.

5.3.8.16 AlertViewInterface

Toteuttaa AlertView-komponentin ohjausrajapinnan. Rajapinta perii itse määrittelemiensä metodien lisäksi myös ViewInterface-rajapinnan metodit.

5.3.8.17 StartViewInterface

Rajapinta StartView-komponentin hallintaan. Rajapinta myös perii ViewInterface-rajapinnan metodit ja lisäksi toteuttaa komponentille ominaisia metodeita, joiden avulla sen tilaa voidaan muuttaa. Nämä metodit auttavat lisäämään hälytyksiä ja poistamaan niitä komponentista.

5.3.8.18 ConfigViewInterface

Rajapinnan avulla voidaan hallita ConfigView-komponentin tilaa. Sen kautta pystyy lisäämään komponentille tiedon järjestelmästä ostettavista tuotteista, lisätä sormenjalkia ja antaa komponentille tiedon käyttäjän sen hetkisestä saldosta järjestelmässä.

6 Raportointiliittymän arkkitehtuuri

Raportointiliittymän avulla käyttäjät pystyvät tulostamaan raportteja järjestelmän käytöstä. Näistä raporteista käy ilmi esimerkiksi tuotteiden tuonnit ja viennit, käyttäjien saldot järjestelmässä ja tuotteiden hintoihin tehdyt muutokset.

Raportointiliittymää käytetään myös järjestelmän yleiseen hallintaan. Sen avulla pääkäyttäjät voivat muokata käyttäjien tietoja tai poistaa käyttäjiä järjestelmästä. Myös tuoteryhmien ja tuotteiden lisäys ja deaktivointi, sekä niiden tietojen muokkaus, tapahtuu raportointiliittymän kautta.

Suunnitelmaan on tullut suuria muutoksia alkuperäisestä ja siten tämä raportointiliittymän suunnitelma on karsittu aiempaa lyhyemmäksi. Luokkien ja tiedostojen tarkempi kuvaus on päätetty jättää tiedostojen kommentteista generoituun PHPdoc ja Jsdoc-dokumentaatioon. Perusrakenne raportointiliittymässä on samanlainen kuin ennenkin, mutta JavaScriptin avulla toteutetut toiminnot ovat siirtäneet paljon toiminnallisuutta PHP-tiedostoista JavaScript-tiedostoihin.

6.1 PHP:n erityisominaisuuksia

Luokkien metodit ovat kaikki public-metodeita, koska PHP:n versio 4.x ei tue private-metodeita. Samoin luokissa olevat muuttujat ovat public-muuttujia. Muuttujien nimissä käytetään ns. unkarilaista notaatiota, jolla kerrotaan muuttujan tyyppi. Esimerkiksi `$a_` on array- tyyppinen muuttuja, `$s_` on string-tyyppinen muuttuja. Array-muuttujissa, eli taulukoissa voidaan hajautustaulun tyyppisesti säilyttää minkä tahansa tyyppin muuttujia, joita indeksoidaan hakuavaimilla.

6.2 Tiedon välitys selaimesta PHP-luokkiin

PHP-tiedostot saavat käskynsä verkkoselaimen suorittamista HTTP GET ja POST -operaatioista. Tietoihin viitataan globaaleilla muuttujilla `$_GET["muuttujan_nimi"]`, `$_POST["lomakkeen_kentän_nimi"]` ja `$_REQUEST["muuttujan_nimi"]`. JavaScript-tiedostoja varten on tehty erillisiä vain tietokannan ja JavaScriptin yhdistämiseen tarkoitettuja PHP-tiedostoja, joita ei ole tarkoitettu käyttäjän suoraan käytettäväksi.

6.3 Istunnonhallinta

Istunnonhallinta toteutetaan PHP:n `$_SESSION` muuttujien avulla. Kun käyttäjä ensi kertaa tulee raportointiliittymän aloitussivulle, hänelle luodaan istunto, joka on voimas-

sa raportointiliittymässä. Käyttäjä tunnistautuu tietojenkäsittelytieteen laitoksen käyttäjätunnuksella ja salasanallaan, jonka jälkeen tarkistetaan, että käyttäjä on rekisteröitynyt järjestelmään asiakasliittymän kautta.

`$_SESSION["currentUser"]` -muuttujaan asetetaan tunnistetun käyttäjän tunnus, jonka jälkeen se on helposti muun järjestelmän saatavissa. Tietokannasta tarkistetaan myös onko käyttäjälle asetettu ylläpito-oikeudet, joka merkitään `$_SESSION["isAdmin"]` -muuttujaan totuusarvona.

Mikäli käyttäjää yrittää tunnistautumatta jollekin raportointiliittymän sivulle, hänet ohjataan automaattisesti kirjautumissivulle. Istunnon voimassaoloaika riippuu PHP:n asenuksesta ja sen pituus on `php.ini`-tiedostossa määritellyn istunnon maksimipituus.

Tiedostossa `session.php`, joka liitetään kaikkien interaktiivisten PHP-sivujen alkuun, on toteutettu istunnon tarkistus ja kielen vaihtaminen. Käyttäjän sessio tuhoetaan tiedostossa `logout.php`, jonka jälkeen käyttäjä ohjataan takaisin kirjautumissivulle.

6.3.1 Kielenvalinta ja kielen vaihtaminen

Raportointiliittymän oletuskieli on suomi. Se tallennetaan muuttujaan `$_SESSION["lang"]`. Raportointiliittymän käyttämät sanat ovat tiedostoissa `settings/fin.xml`, `settings/swe.xml` ja `settings/eng.xml`. Sanojen lataamisen tiedostoista hoitaa `Language`-luokka (tiedostossa `language.php`), jonka `gettext`-funktiolla kaikki käyttäjälle näytettävät tekstit haetaan.

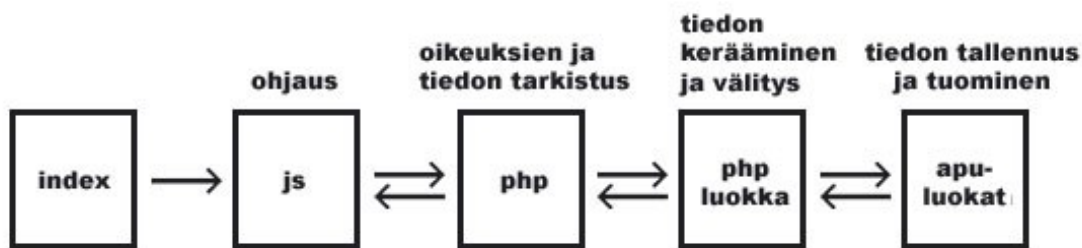
Kieltä voi vaihtaa klikkaamalla valikkopalkissa olevaa lipun kuvaa, jolloin suoritetaan uusi HTTP GET-kutsu samalle sivulle, ja `session.php` hoitaa kielen vaihtamisen edellä mainittuun muuttujaan.

6.4 Perustoiminnallisuus

Raportointiliittymässä tuote- ja käyttäjänhallintasivujen kohdalla periaate on, että sivut ladataan kerran, jonka jälkeen toiminnot käsitellään JavaScriptin avulla siten ettei koko sivua tarvitse ladata uudelleen, paitsi kieltä vaihtaessa. Kuvassa 5 sivujen toimintaperiaate.

JavaScript-tiedostot `product.js` ja `usr.js` luovat tarvittavan HTML-koodin kiinteillä HTML-sivuilla määriteltyihin DIV-elementteihin. Tietokannasta JavaScript-tiedostot hakevat tarvittavat tiedot kutsumalla suoraan niitä varten tehtyjä PHP-tiedostoja, jotka palauttava paljaana merkkieroteltuna tekstinä halutun tiedon. Tietojen talletus tapahtuu vastaavasti kutsumalla PHP-tiedostoja tarvittavin parametrein.

Tietojen validointi tapahtuu näissä JavaScriptin kutsumissa PHP-tiedostoissa ja lopullinen tietokantayhteys hoidetaan PHP-luokkien kautta.



Kuva 5: Raportointiliittymän tuote- ja käyttäjänhallintasivujen toimintaperiaate

Raportit luovat PHP-sivut on tästä poiketen toteutettu perinteisesti siten, että ne itse sekä luovat sivut että käyttävät eri PHP-luokkia tietokantahakujen suorittamiseen.

6.5 Raportointiliittymän hakemistorakenne

Hakemistorakenne on pidetty mahdollisimman yksinkertaisena. Itse ohjelman pääsivut sijaitsevat hakemistopuun juuressa ja eri hakemistoihin on sijoitettu tietyn tyyppistä tietoa. Tämä helpottaa ohjelman rakenteen pitämistä mahdollisimman siistinä ja helposti hallittavana.

6.5.1 Alihakemistot

beans

Hakemistossa on PHP:n luokat, jotka hoitavat tietokantayhteyksien suorittamisen (connection-luokan kautta) ja perustiedonkäsittelyn tietokannasta saatavaan dataan liittyen. Hakemistossa ovat productsAndGroupsBean, usersBean, reportBean ja authenticationBean.

css

Hakemistossa raportointiliittymän CSS-tyylitiedosto.

javascript

Hakemistossa on tuote- ja käyttäjäsiivuilla tarvittavat JavaScript-tiedostot product.js ja usr.js. JavaScript-tiedostot on esitelty tarkemmin ohjelman mukana toimitettavassa omassa dokumentaatioissaan.

pics

Hakemistossa on raportointiliittymän käyttämät kuvat.

include

Hakemistossa on tietokantayhteyden luomiseen sekä tietokantakyselyiden suorittamiseen tarvittavat tiedostot connection.php (PostgreSQL) ja connection_oracle.php. Kullonkin käytössä oleva tiedosto on nimettävä connection.php:ksi.

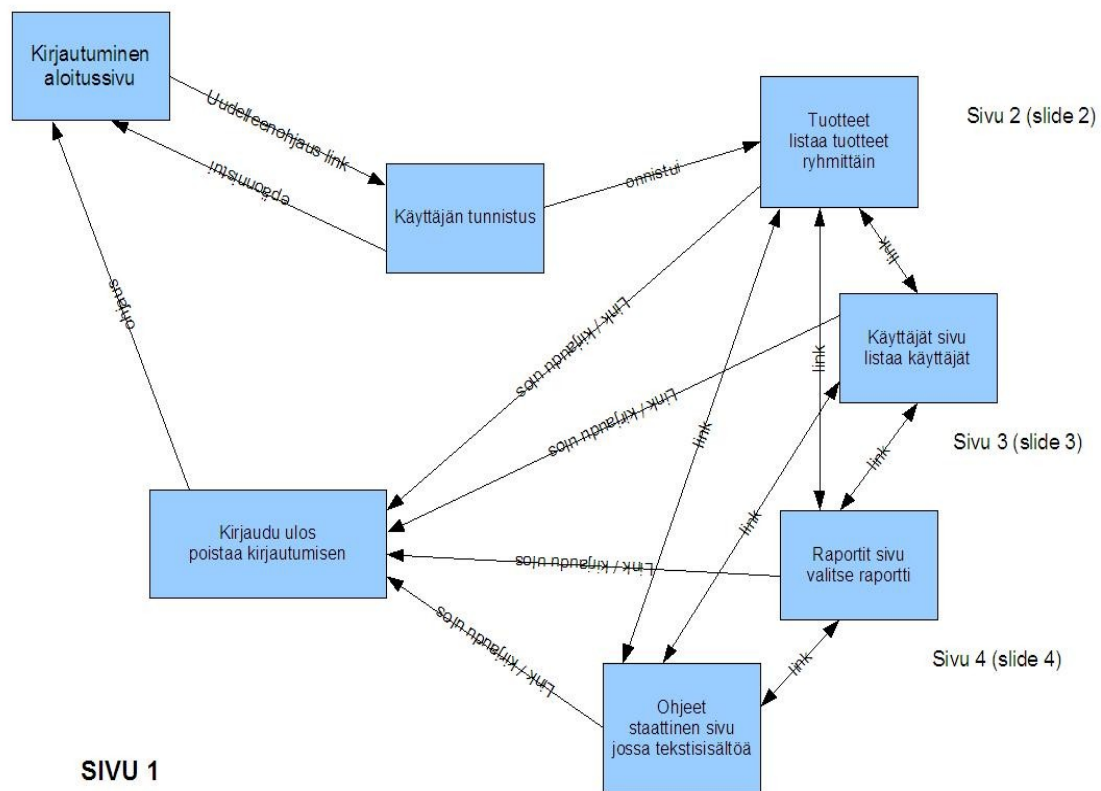
Lisäksi hakemistossa on istunnon hallintaan tarvittavat tiedostot session.php ja session_authentication_page.php sekä oikean kieliset tekstit hakeva language.php, asetustiedot noutava settings.php, tietokantakyselyt hakeva dbqueries.php, navigointivalikon luova top_navi.php ja HTML-sivujen alkuun ja loppuun liitettävät header.php ja footer.php.

settings

Hakemistossa on raportointiliittymän kielitiedostot eng.xml, fin.xml, swe.xml, joissa on sanasto eri kielillä, settings.xml, joka sisältää muut asetukset, ja dbqueries.xml, jossa on tietokantakyselyt.

6.5.2 Tiedostopuun juuressa olevat tiedostot

Seuraavaksi on kuvattu ne tiedostot, jotka ovat tiedostopuun juuressa. Nämä tiedostot toteuttavat käyttöliittymän toiminnot ja sisältävät ohjelman pääasiallisen toteutuksen. Juuressa on sekä tiedostoja, jotka luovat käyttäjälle näytettävän sivun itsenäisesti että ”aputiedostoja”, joita JavaScript-tiedostot käyttävät hyväkseen tuote- ja käyttäjä sivuilla.



Kuva 6: Aloitusnäky (osaksi vanhentunut)

JavaScriptin käyttämät tiedostot palauttavat tiedostojen kommentteissa kuvatus kaltaisen tekstirivin, mikäli tiedoston tarkoituksena on hakea tietoa tietokannasta. Jos tiedostoa käytetään tiedon tallentamiseen, se palauttaa onnistuessaan ”SUCCESS” ja epäonnistuessaan ”FAILURE” (sekä mahdollisen epäonnistumisen syyn, jos se on tiedossa).

index.html

Tiedosto, joka ohjaa kirjautumiseen. Tiedosto ohjaa sivulle authentication.php, eli kirjautumisikkunaan, jossa annetaan tietojenkäsittelytieteen laitoksen tunnus ja salasana. Tunnistuksen jälkeen siirrytään itse sovellukseen ja sivulle reports.php.

products.php

Tiedosto joka listaa tuotteet ryhmittäin. Kun klikataan ryhmän nimeä, ryhmän tuotteet listataan ryhmän nimen alla. Kun tuotteen nimeä klikataan tuotteen tiedot ladataan ja käyttäjälle näytetään muokkauslomake. Painamalla ”lisää tuote” -linkkiä käyttäjälle näytetään tyhjä lomake uuden tuotteen lisäämistä varten. Vastaavasti voidaan muokata tuoteryhmien tietoja ja lisätä uusia ryhmiä.

Käyttäjän on myös mahdollista poistaa (deaktivoida) tuotteita ja ryhmiä sekä palauttaa poistettuja sellaisia takaisin käyttöön. Sivun toiminnallisuus on toteutettu JavaScriptillä ja se tarjotaan navigointivalikossa ainoastaan ylläpitäjille, joskin sen näkyvyyttä ei sinänsä ole estetty tavallisilta käyttäjiltä. Eri PHP-aputiedostot huolehtivat siitä, että tavalliset käyttäjät eivät voi muokata tuotteiden tietoja.

getproducts.php

Tuotesivun JavaScript saa tältä tiedostolta kaikki tuoteryhmät tai jonkin tuoteryhmän kaikki tuonti- tai vientituotteet. Tämän perusteella se rakentaa tuotevalikon tuotesivulle.

getproductdata.php

Tämän tiedoston avulla tuotesivun JavaScript saa haettua tietyn tuotteen kaikki tiedot näytettäväksi muokkauslomakkeessa.

getgroupdata.php

Tämän tiedoston avulla tuotesivun JavaScript saa haettua tietyn tuoteryhmän nimet eri kielillä näytettäväksi muokkauslomakkeessa.

editExportProduct.php

Tuotesivulla käyttäjän muokattua jonkin vientituotteen tietoja kutsutaan tätä tiedostoa JavaScriptistä HTTP GET-kutsulla ja annetaan parametrina tuotteen tiedot. Tiedot tallennetaan tietokantaan. Sopivalla parametrilla tuote voidaan myös deaktivoida tai aktiivoida tämän tiedoston avulla.

editImportProduct.php

Tuotesivulla käyttäjän muokattua jonkin tuontituotteen tietoja kutsutaan tätä tiedostoa JavaScriptistä HTTP GET-kutsulla ja annetaan parametrina tuotteen tiedot. Tiedot tallennetaan tietokantaan. Sopivalla parametrilla tuote voidaan myös deaktivoida tai akti-

voida tämän tiedoston avulla.

addExportProduct.php

Jos käyttäjä lisää tuotesivulla uuden vientituotteen, kutsutaan tätä tiedostoa JavaScriptistä HTTP GET-kutsulla ja annetaan parametrina tuotteen tiedot. Tiedot tallennetaan tietokantaan.

addImportProduct.php

Jos käyttäjä lisää tuotesivulla uuden vientituotteen, kutsutaan tätä tiedostoa JavaScriptistä HTTP GET-kutsulla ja annetaan parametrina tuotteen tiedot. Tiedot tallennetaan tietokantaan.

addGroup.php

Jos tuotesivulla lisätään uusi tuoteryhmä, kutsutaan tätä tiedostoa JavaScriptistä HTTP GET-kutsulla ja annetaan uuden ryhmän tiedot parametreina.

editGroup.php

Mikäli tuoteryhmän tietoja muokataan tai se deaktivoidaan tai aktivoidaan, kutsutaan tätä PHP-tiedostoa JavaScriptistä.

users.php

Sivulla on JavaScriptillä toteutettu jatkuva käyttäjähaku. Käyttäjä valittaessa luodaan lomake, jonka avulla voidaan muuttaa käyttäjän nimeä, oletuskieltä, ylläpito-oikeuksia ja saldoa eri tuoteryhmissä. Ylläpitäjät voivat muokata kaikkia tietoja ja poistaa käyttäjiä, kun tavalliset käyttäjät voivat ainoastaan vaihtaa omaa nimeään ja oletuskieltään.

getusers.php

Tiedosto hakee tietokannasta ja palauttaa käyttäjäsivun JavaScriptille kaikki käyttäjät jatkuvan käyttäjähaun toteuttamista varten.

getuserdata.php

Käyttäjäsivun JavaScript-tiedosto hakee tältä PHP:ltä tietyn käyttäjän tiedot näytettäväksi muokkauslomakkeessa.

getbalances.php

Tämän tiedoston avulla käyttäjäsivun JavaScript saa tietyn käyttäjän saldotiedot kaikissa tuoteryhmissä.

getsession.php

Käyttäjäsivun JavaScript saa tämän tiedoston avulla \$_SESSION-muuttujaan tallennetun nykyisen käyttäjän ja tiedon siitä, onko tällä ylläpito-oikeudet. Näitä tietoja se käyttää rajoittamaan käyttäjälle sallittuja toimintoja. Käyttäjän oikeudet tarkistetaan toki myös muokkaukset välittävissä PHP-tiedostoissa, koska tällainen välitystapa ei ole ko-

vin turvallinen.

saveUser.php

Käyttäjäsivun JavaScript kutsuu tätä tiedostoa sopivin parametrein HTTP GET-kutsulla, kun käyttäjän tiedot halutaan tallettaa tietokantaan.

saveBalances.php

Käyttäjäsivun JavaScript kutsuu tätä tiedostoa sopivin parametrein HTTP GET-kutsulla, kun käyttäjän saldotiedot halutaan tallettaa tietokantaan.

removeUser.php

Käyttäjäsivun JavaScript kutsuu tätä tiedostoa antaen parametriksi käyttäjätunnuksen HTTP GET-kutsulla, kun käyttäjä halutaan poistaa järjestelmästä.

reports.php

Sivulla tuotetaan raportti, jossa näkyy tietyn käyttäjän kaikki tuontitapahtumat tietyllä aikavälillä. Käyttäjää voidaan etsiä käyttäjätunnuksen tai käyttäjän nimen perusteella hakulomakkeella, tai vaihtoehtoisesti voidaan hakea kaikki käyttäjät ja valita listasta jonkin käyttäjän raportti näytettäväksi. Aikavälin alku- ja päätepäivä valitaan omista valitsimistaan. Oletusaikaväli on 31 vuorokautta.

Kun käyttäjä on valittu, näytetään itse raportti, sekä linkki sarakkeiden erotinmerkeillä varustetun (CSV) tiedoston tallennukseen.

reportsOut.php

Sivulla tuotetaan edellisen raportin kaltainen, mutta vientituotteet kattavat raportti. Lisäksi raportissa näkyvät käyttäjän mahdollisesti asiakasliittymällä suorittamat vientien peruutukset.

reportsGroups.php

Sivulla tarjotaan valittavaksi mikä tahansa tuoteryhmä tai niiden alta löytyvä tuote, jonka jälkeen näytetään statistiikkatietoa kyseisen ryhmän tai tuotteen käytöstä (tuonneista/vienneistä) valitulla aikavälillä ja keskimäärin päivää kohden.

reportsValues.php

Sivulla näytetään tuotteet, joiden arvoa (tuonti/vientiarvo) on muutettu valitulla aikavälillä. Jokin tuote valittaessa näytetään raportti, jossa näkyvät arvonmuutokset (mistä mihin), niiden tapahtuma-ajankohta ja muutoksen tehneen käyttäjän käyttäjätunnus.

reportsBalances.php

Sivulla näytetään raportti, jossa näkyy kaikkien järjestelmään rekisteröityneiden saldo kaikissa aktiivisissa tuoteryhmissä. Negatiiviset arvot korostetaan punaisella värillä. Myös tämän raportin voi tulostaa tekstitiedostona.

about.php

Sivulla on yleistietoa järjestelmästä englanniksi sekä tekijöiden nimet.

authentication.php

Sivu tuottaa sisäänkirjautumissivun ja tallettaa käyttäjän tiedot \$_SESSION-muuttujaan onnistuneen kirjautumisen jälkeen.

logout.php

Sivu kirjaa käyttäjän ulos järjestelmästä tuhoten session ja sen tiedot. Käyttäjän selain ohjataan kirjautumissivulle.

6.6 Raportointiliittymän luokat

Raportointiliittymän luokat sisältävät raportointiliittymässä käytettävät monimutkaisemat tieto-oliot. Näitä luokkia ovat Connection, ProductsAndGroupsBean, UsersBean, AuthenticationBean ja ReportBean sekä Language, DBQueries ja Settings. Seuraavaksi esitellään lyhyesti jokainen näistä luokista. Luokkien tarkempi kuvaus löytyy ohjelman mukana toimitettavasta PHPdocista.

6.6.1 Language

Luokka hakee kielitiedostosta raportointiliittymässä näytettävät tekstit näiden tekstien nimien perusteella. Tarkoituksena on tarjota helppo mahdollisuus toteuttaa monikielisyys. Language-luokka luodaan jollekin tietylle kielelle, jonka jälkeen sen gettext-metodilla voidaan hakea tietyn niminen sana.

6.6.2 Connection

Luokka hoitaa tietokantayhteyden ottamisen, yhteyden sulkemisen sekä kyselyiden välittämisen tietokannalle. Connection-tiedostosta tehdään kaksi versiota, toinen PostgreSQL-kantaa ja toinen Oracle-kantaa varten. Käytössä olevan tiedoston pitää olla nimeltään connection.php. Muita tietokantojakin voidaan ottaa käyttöön kirjoittamalla uusi Connection-luokka, joka toteuttaa metodit samalla tavoin kuin PostgreSQL- ja Oracle-versiot. Lisäksi on tietysti huolehdittava tietokantakyselyjen toimivuudesta.

6.6.3 ProductsAndGroupsBean

Luokka hoitaa tuotteisiin ja tuoteryhmiin liittyvät tapahtumat. Sen kautta ladataan tuotetiedot näytettäväksi ja lisätään, muokataan ja poistetaan tuotteita sekä tuoteryhmiä tieto-

kannasta. Luokka myös huolehtii tuotteiden arvojen muutoksien kirjaamisesta lokitau-
luun.

6.6.4 UsersBean

Luokassa hoidetaan käyttäjien tietojen muokkaus, poisto ja lataus. Luokassa ei ole uu-
den käyttäjän lisäämistä, koska se tapahtuu asiakasliittymässä. Käyttäjän käyttäjätun-
nusta ei voi muuttaa, mutta muiden tietojen muokkaaminen on mahdollista. Luokan
avulla voidaan myös hakea ja muokata käyttäjän saldoja eri tuoteryhmissä. Lisäksi luo-
kan kautta tarkistetaan kirjautumisvaiheessa onko tunnistautunut käyttäjä rekisteröitynyt
ja onko tällä ylläpito-oikeudet.

6.6.5 ReportBean

Luokka hakee tietokannasta tiedot järjestelmän tuottamia raportteja ja statistiikkatietoja
varten, ja jäsentää tiedot taulukoihin käytännölliseen muotoon, jolloin tiedot on helppo
tulostaa raportiksi varsinaisilla raporttisivuilla.

6.6.6 AuthenticationBean

Luokka hoitaa käyttäjätunnusten autentikoinnin tietojenkäsittelytieteen laitoksen verk-
kopalvelinta varten. Koska TKTL:llä käytetty PHP ei tarjoa suoraan mahdollisuuksia
tarvittavan HTTPS-yhteyden ottamiseen, ajetaan luokassa palvelinkoneelta ulkoinen
cURL-ohjelma, jonka antamasta HTTP-protokollan vastauskoodista autentikaation on-
nistuminen voidaan päätellä. Autentikointiin käytetty verkkosivu voidaan säätää asetus-
tiedostossa. cURL on yleinen, useimmissa Linux-koneissa perusasennukseen kuuluva
ohjelma.

6.6.7 Settings

Luokka hakee asetustiedostosta settings.xml tarvittavat asetukset, joita ovat lähinnä tie-
tokannan ja autentikointisivun osoite.

6.6.8 DBQueries

Luokka hakee tietokantakyselytiedostosta dbqueries.xml tietokantakyselyt muiden luok-
kien käyttöön. Kyselyitä voidaan hakea luokasta niille annettujen nimien perusteella.

7 Ostoliittymän luokkakuvaukset

Tässä osiossa kuvataan aiemmin esiteltyjen Ostoliittymän komponenttien luokat. Luetavuuden parantamiseksi komponenttien luokkakaaviot ovat tämän dokumentin liitteenä. Luokat on laitettu paketteihin siten, että asiakasliittymän juuripaketti on nimeltään *cafe* ja eri komponentit ovat sen alaisissa paketeissa.

Komponenttien tarkempi kuvaus löytyy ohjelman mukana toimitettavasta Javadocista, ja myös osoitteessa <http://www.cs.helsinki.fi/group/kasi/doc/index.html>.

7.1 MainProgram-komponentti

Paketti: main

MainProgram-komponentti koostuu kolmesta luokasta. Se toteuttaa `UserInterfaceEvent`-rajapinnan GUI-komponentin ja `FingerprintEvent`-rajapinnan `FingerprintAuthentication`-komponentin käytettäväksi.

MainProgram-komponentti itse käyttää GUI:n toteuttamaa `UserInterface`-rajapintaa, Database-komponentin `Database`-rajapintaa, Settings-komponentin toteuttamaa `Configuration`-rajapintaa, Authentication-komponentin toteuttamaa `UserAuthentication`-rajapintaa ja `FingerprintAuthentication`-komponentin toteuttamaa `Fingerprint`-rajapintaa.

Komponentin UML-kaaviossa (liite A, kuva 2) on esitelty sen luokkien väliset suhteet ja tärkeimmät metodit. Lisäksi siitä näkee helposti komponentin käyttämät ulkoiset luokat.

7.1.1 CoffeeTouch

`CoffeeTouch` on komponentin ja ohjelman pääluokka, joka käynnistää muut komponentit, ja lisäksi välittää tarpeen vaatiessa käyttöliittymälle virheilmoituksen.

7.1.2 MainGuiHandler

Luokan tehtävä on toimia yhteytenä käyttöliittymän, tietokannan ja käyttäjätunnusten autentikoinnin välillä. Luokka toteuttaa `UserInterfaceEvent`-rajapinnan ja annetaan parametrina `GuiMain`-luokan konstruktorille, joka odottaa kyseisen rajapinnan toteuttavaa luokkaa. Luokan käyttämät (tai pikemminkin välittämät) `ProductGroup`, `ImportProduct`, `CancelProduct` ja `ExportProduct` on kuvattu luvussa 5.2, kuten myös `UserInfo`-luokka.

Tietokantaluokan metodeja kutsuttaessa pitää käsitellä tietokantaluokan mahdollisesti heittämä `DBConLostException`. Tällaisen sattuessa kutsutaan `CoffeeTouch`-luokan `displayDBError`-metodia.

7.1.3 MainFingerprintHandler

Luokan tehtävä on toimia yhteytenä sormenjälkitunnistuksen, tietokannan ja käyttöliittymän välillä. Luokka annetaan parametrina FingerprintAuthenticationin konstruktorille. Luokan välittämät FPData-luokan oliot on kuvattu luvussa 5.2. Luokka toteuttaa FingerprintEvent-rajapinnan.

Tietokantaluokan metodeja kutsuvien metodien pitää käsitellä tietokantaluokan mahdollisesti heittämä DBConLostException. Tällaisen sattuessa kutsutaan CoffeeTouch-luokan displayDBError-metodia.

7.2 Authentication-komponentti

Paketti: authentication

Komponentin tehtävä on tunnistaa käyttäjä TKTL:n käyttäjäksi vertailemalla syötettyjä tunnuksia TKTL:n vastaaviin käyttäen HTTP-protokollan basic authentication-menetelmää. Komponentti käyttää apunaan ulkoista luokkaa Base64, joka hoitaa käyttäjätunnuksen ja salasanan base64-koodauksen HTTP viestiä lähetettäessä. Jotta komponentti toimisi, käyttäjän täytyy lisätä TKTL:n SSL-varmenne Java-virtuaalikoneen varmenteiden (ns. keystore) käyttämällä Javan mukana tulevaa keytool-työkalua.

Komponentin UML-luokkakaavio on dokumentin liitteenä (liite A, kuva 3).

7.2.1 Authentication

Luokan tehtävänä on tunnistaa käyttäjä HTTP-protokollan basic authentication menetelmän avulla. Toteuttaa UserAuthentication-rajapinnan.

7.2.2 ConnectionData

Luokka toimii tietovarastona, jonne tallennetaan HTTP vastauskoodi, sekä vaikutusalue yhteyttä muodostaessa. Luokka jäsentää saadun vastauksen.

7.2.3 Base64

Ulkopuolinen luokka joka muuttaa tekstiä Base64:een ja takaisin. Tekijä Robert Harder ja luokan kuvaus löytyy www-sivustolta <http://iharder.net/base64>.

7.3 Settings-komponentti

Paketti: settings

Settings-komponentin tehtävä on kerätä ohjelman asetukset asetustiedostoista, ja välittää ne ohjelman muille komponenteille.

Komponentin UML-luokkakaavio on dokumentin liitteenä (liite A, kuva 4).

7.3.1 Settings

Komponentin ainoa luokka toteuttaa Configuration-rajapinnan.

7.3.2 XML-asetustiedostot

Komponentti hakee asiakasliittymän säädettävissä olevat asetukset XML-muotoisista tiedostoista. Ne noudattavat seuraavaa muotoa:

```
<PROPERTIES>
  <ENTRY KEY="avain">arvo</ENTRY>
</PROPERTIES>
```

Asetustiedostoja on viisi kappaletta. Pääasetustiedosto settings.xml:n osoite voidaan ohjelmalle antaa komentorivillä käynnistyksen yhteydessä. Muiden asetustiedostojen osoitteet on määritelty sen sisällä. Muut tiedostot ovat tietokantakyselyt sisältävä dbqueries.xml ja käyttöliittymän tekstit eri kielillä sisältävät english.xml, finnish.xml ja swedish.xml.

7.4 Database-komponentti

Paketti: database

Komponentti huolehtii asiakasliittymän tietokantayhteyden luomisesta ja sen tietokantakyselyistä. Sen tehtävänä on piilottaa tietokannan kanssa käytettävän SQL-kielen kyselyt muulta ohjelmalta ja muuttaa ne normaaliksi Javan metodirajapinnaksi.

Komponentin UML-luokkakaavio on dokumentin liitteenä (liite A, kuva 5).

7.4.1 DBCommunication

Luokan tarkoitus on välittää tietoa tietokannasta ohjelmalle ja ohjelmasta kantaan. Kun luokasta luodaan uusi ilmentymä, yhteys kantaan avataan ja sitä ei suljeta välissä. Kaik-

ki kantaa käyttävät metodit koettavat käynnistää yhteyden kerran jos se on alhaalla, ja heittävät sitten vasta poikkeuksen.

7.4.2 DBConLostException

Tämä poikkeusluokka on sitä varten, että tällä ilmoitetaan tietokantayhteyden olevan kadonnut. Tämä luokka perii Exception-luokan.

7.4.3 ProductNotFoundException

Tämä poikkeusluokka sitä varten että ilmoitetaan että tuote puuttuu, eli tuote on poistettu käytöstä. Viestiksi annetaan tuotteen nimi sekä id. Tämä luokka perii Exception-luokan.

7.5 FingerprintAuthentication-komponentti

Paketti: fingerprintauthentication

Komponentti huolehtii sormenjälkien lukemisesta ja tunnistamisesta, mihin käytetään Griaulen GrFinger -apuohjelmaa. Komponentti käyttää FingerprintEvent-rajapintaa, joka tarjoaa tarvittavat yhteydet Main-komponenttiin. UML-kaaviossa komponentista (Liite A, kuva 6) näkee helposti luokkien väliset yhteydet ja niiden metodit, sekä tärkeät sisäiset muuttujat.

7.5.1 FingerprintAuthentication

Komponentin pääluokka, joka ottaa vastaan käynnistysasetukset ja olion, joka toteuttaa FingerprintEvent-rajapinnan. Luokan tehtäviin kuuluu myös Griaulen GrFingerJava-kirjaston alustaminen ja käyttöönotto. Lisäksi se toteuttaa IStatusEventListener-tapahtumarakapinnan, jonka kautta sormenjälkitunnistuskirjasto ja -ajuri kertovat, onko koneessa kytkettynä sormenjälkitunnistinta vai ei.

Luokassa luodaan FingerprintCache-olio, jonka tehtävänä on tallettaa muistiin ohjelman tarvitsemat sormenjälkien tunnistustiedot, ja ReaderListener-olio jokaista sillä hetkellä kytkettynä olevaa sormenjälki-lukijaa varten. Myös MatchingContext-olioiden luominen ja käyttäminen on tämän luokan tehtävänä. Nämä GrFingerJava-kirjastoon kuuluvat oliot tekevät varsinaisen sormenjälkikuvien vertailun. Toinen rajapinta, jonka luokka toteuttaa on FingerprintData. Tämän rajapinnan avulla ReaderListener-olio lähettää uuden sormenjälkienlukijalta tulleen kuvan tälle luokalle käsiteltäväksi.

Komponentin UML-luokkakaaviosta (Liite A, kuva 6) käyvät ilmi komponentin luokkien suhteet toisiinsa ja GrFingerJava-kirjastoon.

7.5.2 FingerprintCache

Tämän luokan tehtävänä on luoda välimuistipuskuri komponentissa käytettäville sormenjälkien tunnistustiedoille. Tämän pitäisi nopeuttaa ohjelman toimivuutta ja lisäksi vähentää ohjelman tietokannalle aiheuttamaa kuormaa. Ohjelman asetuksissa voidaan säätää sitä kuinka monta sormenjälkeä pidetään muistissa, tai voidaan myös asettaa, että kaikki sormenjäljet talletetaan muistiin. Luokka on erittäin vahvasti riippuvainen Fingerprint-luokasta, joiden muodostamana taulukkona välimuisti toteutetaan.

7.5.3 FingerprintCacheData

Tämä luokka toimii komponentin sisällä sormenjälkien tallettajana, kun ne varastoidaan välimuistiin. Lisäksi se pitää kirjaa siitä, että kuinka monta kertaa mitäkin sormenjälkeä on oikeasti tarvittu, jotta käyttämättömät sormenjäljet voidaan siivota pois muistista. Tätä luokkaa käyttävät FingerprintCache ja FingerprintAuthentication -luokat. Sormenjälki on tallennettuna jo valmiiksi GrFingerJavan-tarvitsemaan FingerprintImage-muotoon.

7.5.4 FingerprintData

Tämä rajapinnan avulla ReaderListener-luokka kertoo FingerprintAuthentication luokalle saaneensa uuden kuvan ja välittää sen jatkokäsiteltäväksi. Rajapinnassa on vain yksi metodi.

7.5.5 ReaderListener

Luokan tehtävänä on ottaa vastaan sormenjälkien kuvia sormenjälkilukijalta ja välittää niitä eteenpäin jatkokäsiteltäväksi. Jokaista järjestelmään liitettyä lukijaa varten luodaan yksi näitä luokkia ja se toimii tämän jälkeen itsenäisesti kuunnellen tapahtumia sormenjälkilukijalta. Saatuaan uuden kuvan luokka lähettää sen eteenpäin käyttäen FingerprintData-rajapinnan newFingerprint-metodia.

7.6 GUI-komponentti

Paketti: gui ja sen alipaketit

GUI-komponentti toteuttaa järjestelmän asiakasliittymän graafisen käyttöliittymän. Monimutkaisuudestaan johtuen komponentti on jaettu useaan pakettiin, jotka on seuraavassa dokumentoitu omien alaotsikoidensa alla. Paketeissa käytetyt general-paketissa tai sen alipaketeissa olevat luokat on dokumentoitu vain kertaalleen omassa kohdassaan. Osa komponentissa käytettävistä olioista on dokumentoitu aikaisemmin tässä dokumentissa luvussa 5.3 (UserInfo ja GuiSettings sekä niiden apuluokat, ja rajapinnat UserInterfaceEvent ja UserInterface). GUI-komponentin sisäiset rajapinnat on dokumentoitu luvussa 5.3.8.

Jokainen käyttöliittymän ”nappula”, eli käyttäjän manipuloitavissa oleva osa, toteuttaa ActionListener-rajapinnan, joten sitä ei ole jokaisen kohdalla erikseen mainittu.

7.6.1 GuiMain-alikomponentti

Paketti: gui

Paketissa on käyttöliittymäkomponentin pääluokka sekä (kappaleessa 5 kuvatut) rajapinnat UserInterface ja UserInterfaceEvent, joiden kautta käyttöliittymä keskustele muun järjestelmän kanssa.

7.6.1.1 GuiMain

Luokka perii JFrame-luokan, eli luo käyttöliittymän ikkunan. Se toteuttaa UserInterface- ja ViewEventInterface-rajapinnat. UserInterfaceEvent-rajapinnan avulla luokka pystyy kutsumaan muuta järjestelmää tarpeen mukaan. Lisäksi luokka käyttää GUI-komponentin sisällä RegisterViewInterface, ShopViewInterface, ImportViewInterface, AlertViewInterface, StartViewInterface ja ConfigViewInterface -rajapintoja.

7.6.2 StartView-alikomponentti

Paketti: gui.startView

StartView-alikomponentti toteuttaa käyttäjälle ensimmäisenä näkyvän osan ohjelmasta. Tämän komponentin toteuttamassa näkymässä käyttäjä voi aloittaa rekisteröitymisen ohjelman käyttäjäksi, aloittaa tuotteen loppumisesta kertovan varoituksen lisäämisen, poistaa tuotteen loppumisen varoitus tai sormenjäljen antamalla kirjautua sisään ohjelmaan. Toteuttaa StartViewInterface-rajapinnan.

7.6.2.1 StartPanel

Luokka toimii paketin pääluokkana. Se toteuttaa startViewInterface- ja AlertEvent-rajapinnat. AlertEvent-rajapinnan avulla sille kerrotaan, että käyttäjä haluaa lisätä tai poistaa hälytyksen tuotteelta. Tästä kerrotaan eteenpäin ViewEventInterface-rajapinnan avulla. Luokka perii JPanel-luokan.

7.6.2.2 RemoveButton

Tulostaa napin alkunäkymään, jolla poistetaan hälytys.

7.6.2.3 ErrorPanel

Paneelissa on teksti hälytyksestä ja nappi, jolla hälytyksen voi poistaa. Luokka perii JPanel-luokan.

7.6.3 RegisterView-alikomponentti

Paketti: gui.registerView

RegisterView-alikomponentti mahdollistaa uuden käyttäjän rekisteröitymisen. Siinä käyttäjä kirjoittaa TKTL:n käyttäjätunnuksen ja salasanaansa. Jos nämä ovat oikein, käyttäjälle annetaan mahdollisuus syöttää itselleen etunimi ja sukunimi järjestelmään. Näiden tietojen antamisen jälkeen käyttäjä siirretään normaaliin ConfigView-alikomponentin näkymään. Toteuttaa RegisterViewInterface-rajapinnan

7.6.3.1 RegisterPanel

Luokka toteuttaa RegisterEvent- ja RegisterViewInterface-rajapinnat ja perii JPanel-luokan.

7.6.3.2 CPanel

Keskimmäinen paneeli, perii JPanel-luokan.

7.6.3.3 Lower

Näyttää käyttäjälle kentät, joihin hän voi kirjoittaa etu- ja sukunimensä. Perii JPanel-luokan.

7.6.3.4 Upper

Näyttää käyttäjälle kentät, joihin hän voi laittaa käyttäjätunnuksensa ja salasansansa. Perii JPanel-luokan.

7.6.3.5 CnPanel

Näyttää sivun otsikon. Perii JPanel-luokan.

7.6.4 ConfigView-alikomponentti

Paketti: gui.configView

ConfigView-alikomponentin tuottama näkymä mahdollistaa käyttäjän käyttäjätietojen muuttamisen ohjelmassa. Sen avulla käyttäjä voi vaihtaa vakiotuotettaan ja lisätä itselleen uusia, tai korvata vanhoja, sormenjälkiä, jotka mahdollistavat käyttäjän sisäänkirjautumisen sormenjälkilukijan avulla.

ConfigView-alikomponentin näkymästä käyttäjä voi siirtyä tuotteiden tuonti tai ostonäkymään. Alikomponentti toteuttaa ConfigViewInterface-rajapinnan.

7.6.4.1 ConfigPanel

Paketin pääluokka, joka toteuttaa ConfigViewInterface- ja FingerEvent-rajapinnat ja perii JPanel-luokan.

7.6.4.2 CPanel

Paneli on olemassa järjestääkseen sen sisällä olevat paneelit parempaan järjestykseen. Perii JPanel-luokan.

7.6.4.3 CnPanel

Näyttää sivun otsikon. Perii JPanel-luokan.

7.6.4.4 CcnPanel

Näyttää ohjeen käyttäjälle ja tiedon mikä on hänen tämän hetkinen vakiotuotteensa.

7.6.4.5 LeftSide

Näyttää käyttäjän saldot tuoteryhmissä.

7.6.5 ShopView-alikomponentti

Paketti: gui.shopView

ShopView-alikomponentti tuottaa näkymän, jonka kautta käyttäjä voi merkitä ostaneensa tuotteita järjestelmästä. Näkymä tulee näkyville, kun käyttäjä joko syöttää ohjelman alkuruudussa sormenjälkensä tai siirtyy siihen jostain muusta näkymästä nappia painamalla.

Jos käyttäjä on syöttänyt sormenjälkensä alkuruudussa ja ei tee mitään muuta tämän jälkeen tai kirjautuu ulos järjestelmästä, veloitetaan häneltä asetuksissa määritelty vakiotuote. Tämän komponentin näkymästä käyttäjä voi siirtyä tuotteiden tuonti- tai omat tiedot -näkyymiin. Alikomponentti toteuttaa ShopViewInterface-rajapinnan.

7.6.5.1 ShopPanel

Paketin pääluokka, joka keskustelelee muun järjestelmän kanssa ViewEventInterfacen avulla. Toteuttaa ShopViewInterface- ja BuyEvent-rajapinnat. Luo luokat SPanel, CPanel ja NPanel. Perii JPanel-luokan.

7.6.5.2 CancelCPanel

Toteuttaa keskialustalla olevan peruutustapahtuman.

7.6.5.3 CPanel

Näyttää osto- ja peruutusnapin ja tiedot mitä tuotetta käyttäjä on ostamassa sekä ohjetekstin. Perii Jpanel-luokan.

7.6.5.4 SPanel

Näyttää käyttäjän saldon ruudun alareunassa. Perii JPanel-luokan.

7.6.5.5 CancelButton

Nappi, jolla saadaan lisää aikaa miettimiseen, eli peruutetaan tuotteen veloitus tietyn ajan kuluttua.

7.6.5.6 BuyButton

Nappi, jolla vahvistetaan valittu tuote heti vietäväksi.

7.6.6 ImportView-alikomponentti

Paketti: gui.importView

ImportView-alikomponentin tuottaman näkymän kautta käyttäjät pystyvät merkitsemään tiedon siitä, että ovat tuoneet tuotteita. Nämä merkinnät lisäävät käyttäjän saldoa järjestelmässä. Näkymästä pääsee osto- ja omat tiedot -näkyymiin. Alikomponentti toteuttaa ImportViewInterface-rajapinnan.

7.6.6.1 ImportPanel

Paketin pääluokka, joka luodaan kun halutaan sivu näkyviin. Toteuttaa ImportEvent-rajapinnan, jonka avulla kerrotaan kun jokin tuote halutaan hyväksyä tuotavaksi. Perii JPanel-luokan.

7.6.6.2 SPanel

Luokka näyttää käyttäjän saldon sivun alareunassa.

7.6.6.3 CnPanel

Luokka näyttää yläreunan infotekstin.

7.6.6.4 CPanel

Luokka jakaa näkymän kahtia ja luo vasemmalle puolelle listan tuotavista tuotteista ja oikealle puolelle ohjeen ja hyväksymisnapin. Perii JPanel-luokan.

7.6.6.5 RightSide

Näytetään CPanelin oikealla puolella.

7.6.7 AlertView-alikomponentti

Paketti: gui.alertView

AlertView-alikomponentti tuottaa näkymän, jossa käyttäjä pystyy lisäämään varoituksen jonkin tuotavan tuotteen loppumisesta. Käyttäjän lisättyä varoituksen ohjelma siirtyy takaisin aloitusnäköön. Alikomponentti toteuttaa AlertViewInterface-rajapinnan.

7.6.7.1 CPanel

Keskimmäinen paneeli, perii JPanel-luokan.

7.6.7.2 AlertPanel

Paketin pääluokka, toteuttaa paketin rajapinnat ja keskustelee muun järjestelmän kanssa. Perii JPanel-luokan.

7.6.8 General-alikomponentti

Paketti: gui.general ja sen alipaketit

7.6.8.1 NPanel

Luokasta luodaan ilmentymä. Tämän jälkeen kutsutaan sopivaa metodia ja luokka näyttää tietyt navigointi nappulat. Luokka perii JPanel-luokan.

7.6.8.2 LanguageButton

Kielenvaihtonappi, perii JButton-luokan.

7.6.8.3 ImportButton

Nappi mahdollistaa siirtymisen tuotteiden tuonti-ikkunaan. Perii JButton-luokan.

7.6.8.4 ConfigButton

Nappi mahdollistaa omien asetusten näkymään siirtymisen. Perii JButton-luokan.

7.6.8.5 ShopButton

Nappi antaa mahdollisuuden siirtyä ostotilaan. Perii JButton-luokan.

7.6.8.6 LogoutButton

Nappi antaa mahdollisuuden siirtyä alkutilaan. Perii JButton-luokan.

7.6.8.7 CancelProductButton

Nappi antaa mahdollisuuden perua aikaisemmin tehty ostos. Mikäli peruutusta ei ole tai sitä ei sallita, nappi ei ole aktiivinen.

7.6.9 ProductLists - alikomponentti

Komponentti jakaa tuotavat tuotteet kahteen JList-komponenttiin. Vasemman puoleisessa näytetään ryhmät, mihin tuotteita voi tuoda ja oikeanpuoleisessa näytetään valitun ryhmän sisältävät tuotteet ja niiden mahdolliset koot. Käyttäjä voi valita näistä listoista tuovansa tuotteen.

7.6.9.1 AlertViewProductRenderer

Muokkaa tuotelistan joilla on hälytys

7.6.9.2 ImportableList

Luokka on paketin pääluokka joka luodaan kun halutaan luoda lista tuotavista tuotteista.

7.6.9.3 GroupList

GroupList sisältää tuotavien tuotteiden ryhmät ja näyttää ne JList-elementteinä.

7.6.9.4 GroupRenderer

Muokkaa tuotavien tuotteiden ryhmät.

7.6.9.5 ImportProductList

Listaa ryhmän tuotteet

7.6.9.6 ImportProductViewRenderer

Muokkaa tuotavien tuotteiden listan.

7.6.9.7 ProductList

ProductList sisältää tuotavan ryhmän kaikki tuotteet ja niiden tuotavissa olevat koot.

7.6.9.8 ProductRenderer

Tuotavien tuotteiden listan muokkaus.

7.6.9.9 ShopableList

7.6.10 fingerConf - alikomponentti

Tarjoaa käyttäjälle mahdollisuuden nähdä minkä sormien sormenjäljet järjestelmään on lisätty. Mahdollistaa myös sormenjälkien lisäämisen ja muokkauksen.

7.6.10.1 FingerConfigPanel

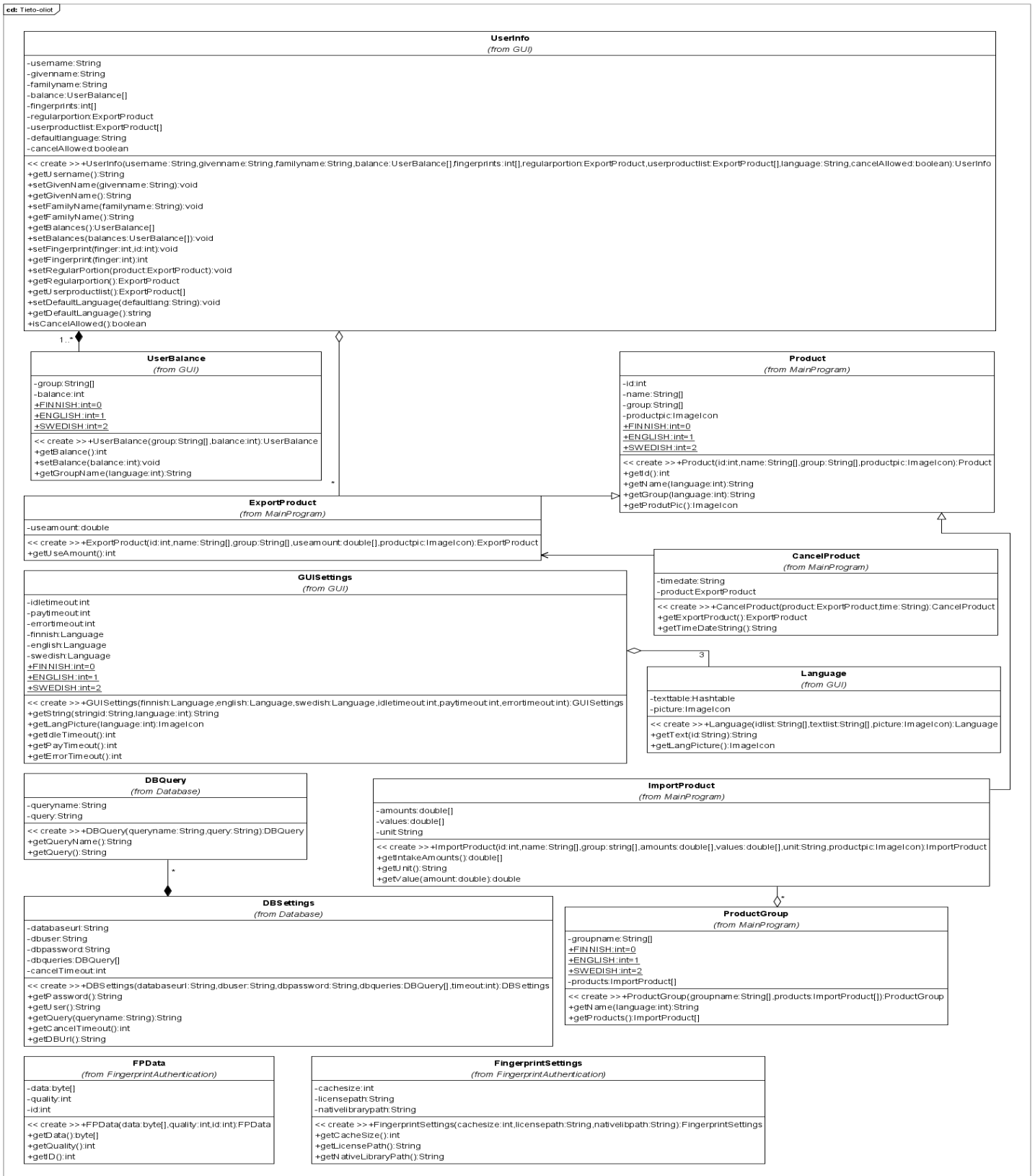
Paketin pääluokka, perii JPanel-luokan.

7.6.10.2 FingerButton

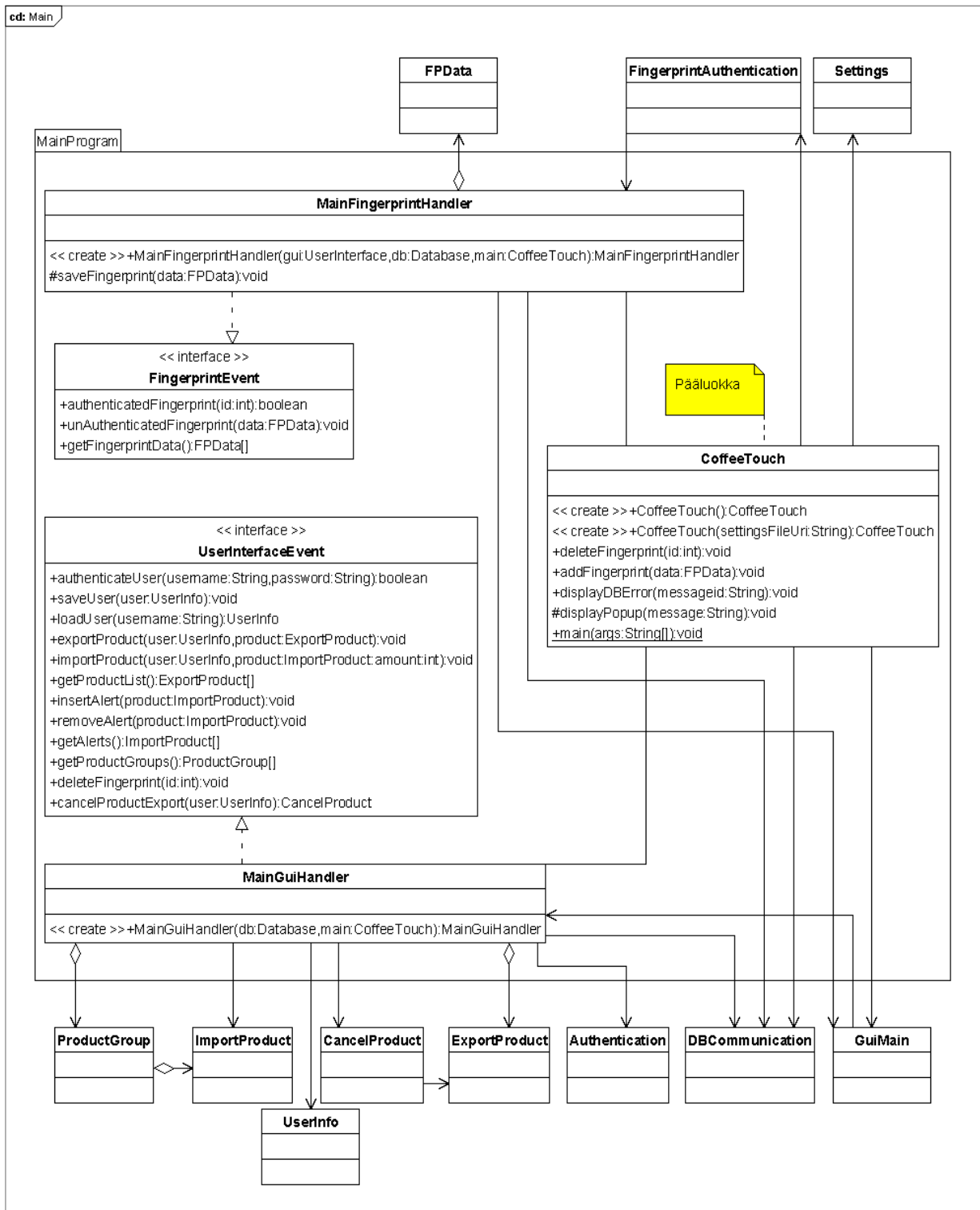
Nappi, jonka avulla käyttäjä kertoo mitä sormeaa hän haluaa muokata. Napin väri myös riippuu siitä onko siinä jo sormenjälki.

Liite A: Luokkakaaviot

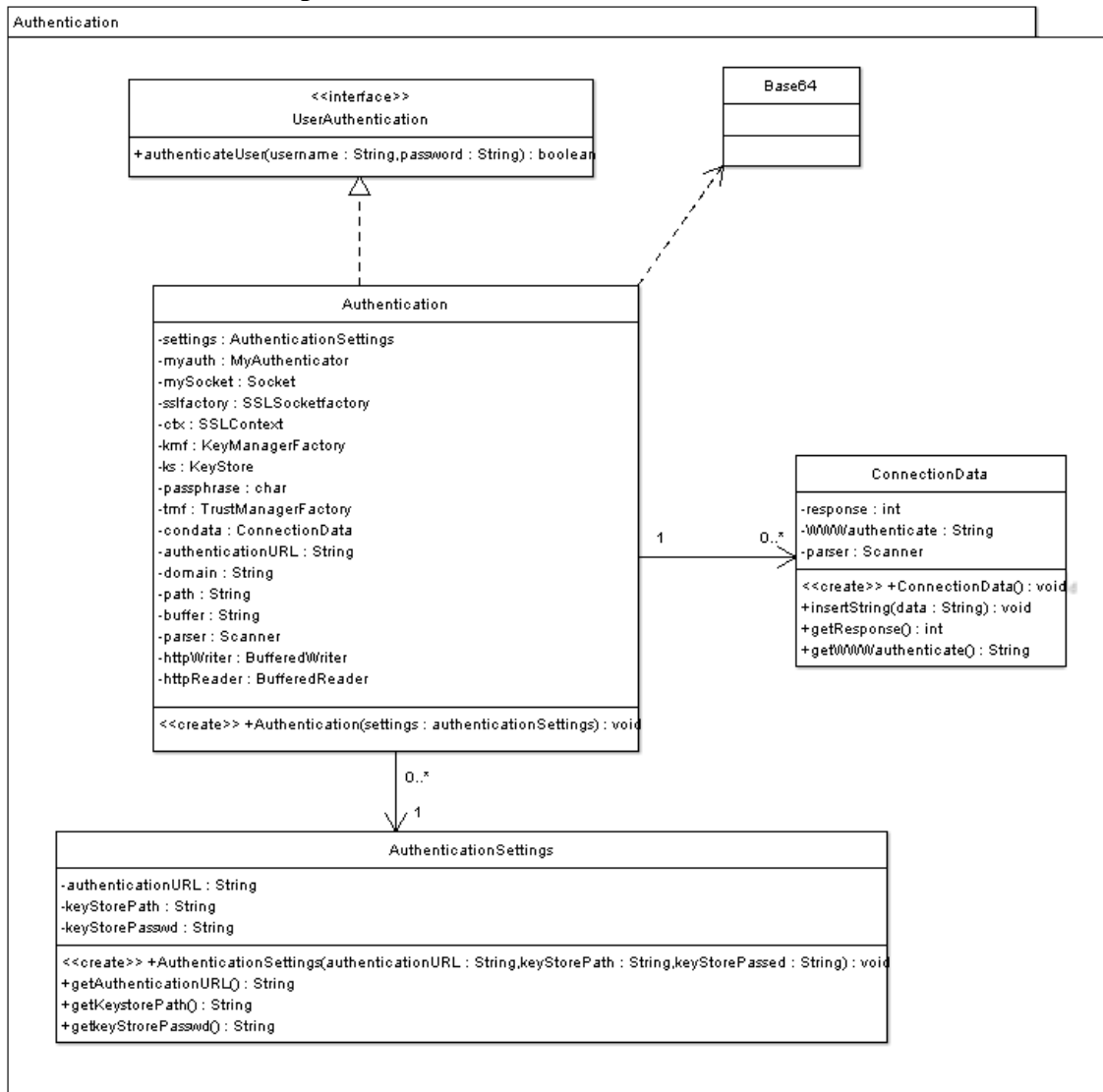
1. Asiakasliittymän tietoliit



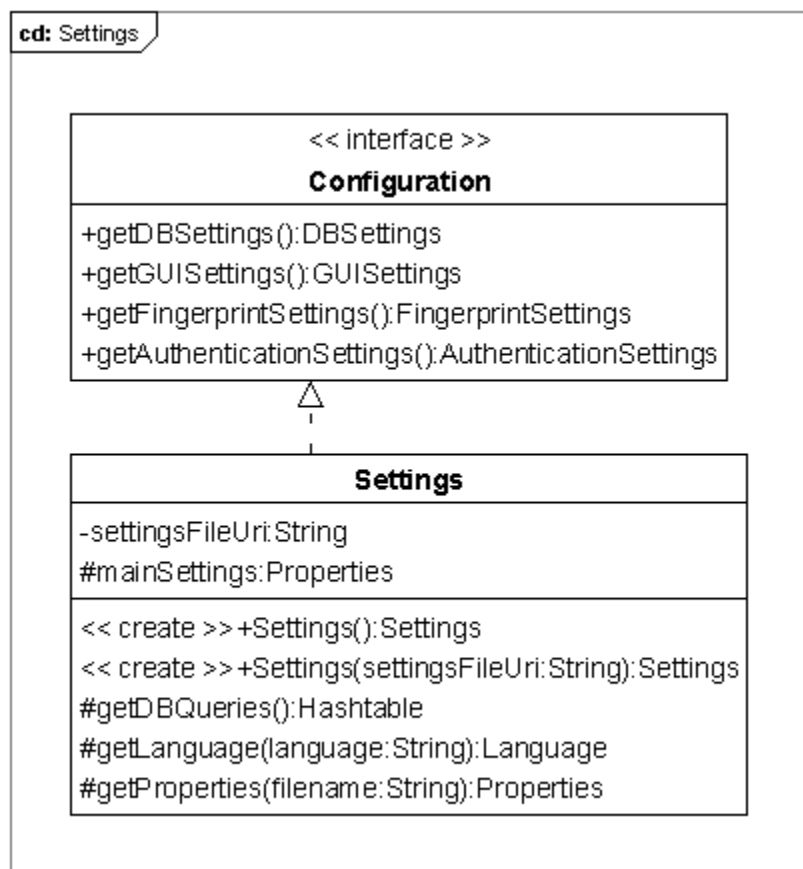
2. MainProgram-komponentin luokkakaavio



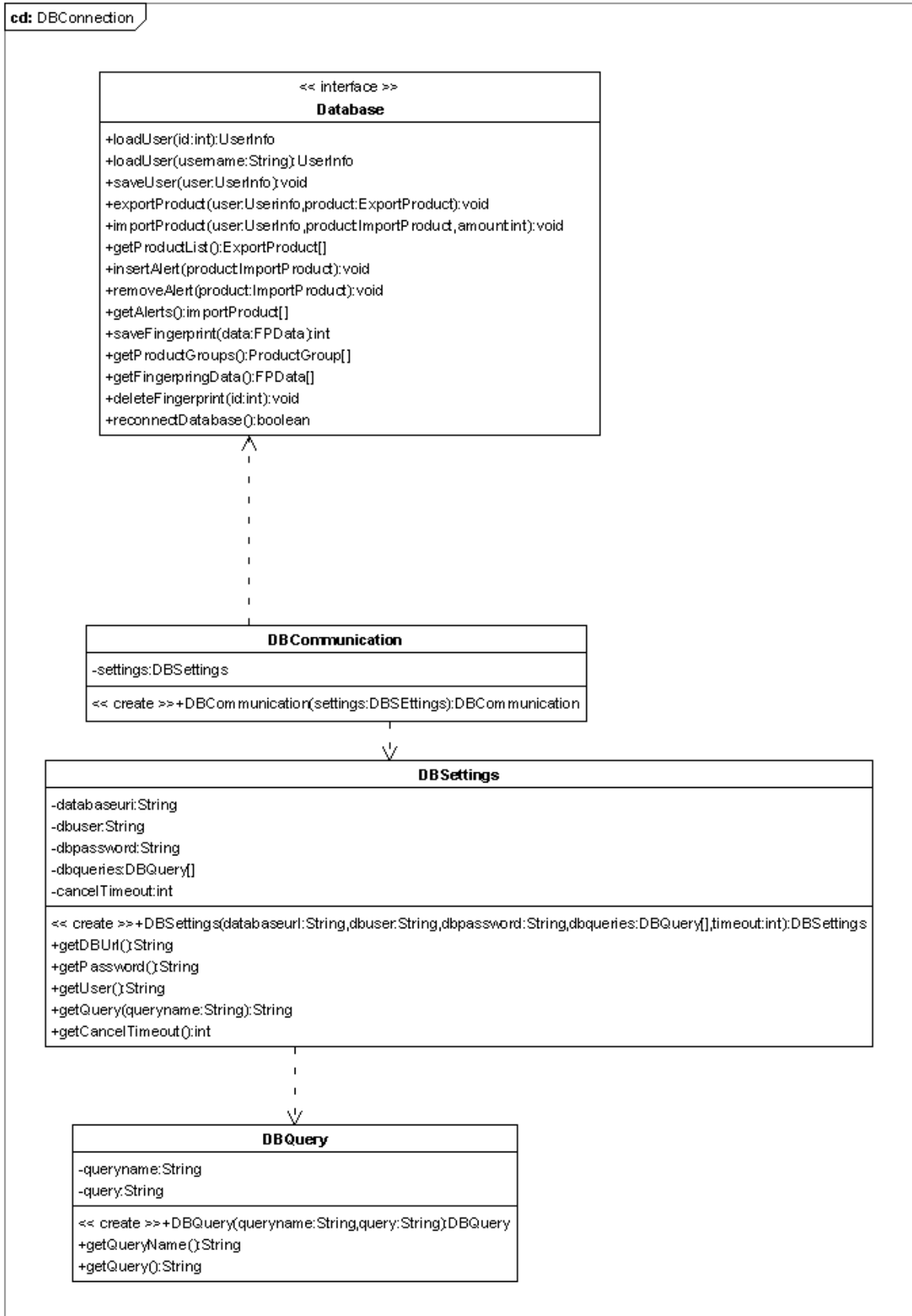
3. Authentication-komponentin luokkakaavio



4. Settings-komponentin luokkakaavio.



5. Database-komponentin luokkakaavio



7. GUI-komponentin luokkakaaviot

