

Henrik Aho – Nikke Tikka – Raoul Osman – Pauli Vuolle-Apiala

OTP1- Projektin dokumentaatio - Ryhmä 4

Varastopojat-varastonhallintasovellus

Metropolia Ammattikorkeakoulu
Tieto- ja viestintätekniikka
Ohjelmistotuotanto
Ohjelmointiprojekti
Päivämäärä

Sisällys

1	Johdanto	1
2	Työn tausta ja tavoite	1
3	Käsitteet, määritelmät	2
4	Toteutetut ja tulevat toiminnallisuudet	3
4.1	Kirjautumisikkuna	3
4.2	Inventaarionäkymä	3
4.3	Ostonäkymä	4
4.4	Myyntinäkymä	5
4.5	Laskutus ja raportointi	5
4.6	Muut toiminnallisuudet	6
5	Järjestelmän ohjelmointitekhninen toteutus	6
5.1	Käytetyt ohjelmointikielet ja kirjastot	7
5.2	Arkkitehtuuri	7
5.3	Käyttöliittymän kuvaus	8
5.4	Järjestelmän keskeinen logiikan kuvaus	9
5.5	Ulkoisten tietovarastojen (tiedostot, tietokannat) kuvaukset	10
5.6	Testaus	11
6	Varastosovelluksen käyttöohje	12
6.1	Inventaarinäkymän ohje (päänäkymä)	14
6.2	Myyntinäkymän ohje	16
6.3	Ostonäkymän ohje (Kesken)	17
6.4	Raportoinnin ohje	18

Liitteet

Liite 1. Javadoc-dokumentaatio, projektin doc-hakemistossa

Liite 2. UML-kaaviot, projektin doc-hakemistossa

Liite 3. Readme.md – Sisältää projektin asennuksen tiedot, projektin päänäkökymässä

1 Johdanto

Varastointi on tärkeä osa-alue useiden yritysten liiketoiminnassa. Toimiva varasto on elinehto yrityksille, jotka haluavat pärjätä jatkuvasti koventuvassa kilpailussa globaalissa maailmassa. Yrityksen koosta riippuen varastoitavien nimikkeiden määrät voivat kasvaa suuriksi ja niihin sitoutunut pääoma voi olla merkittävää.

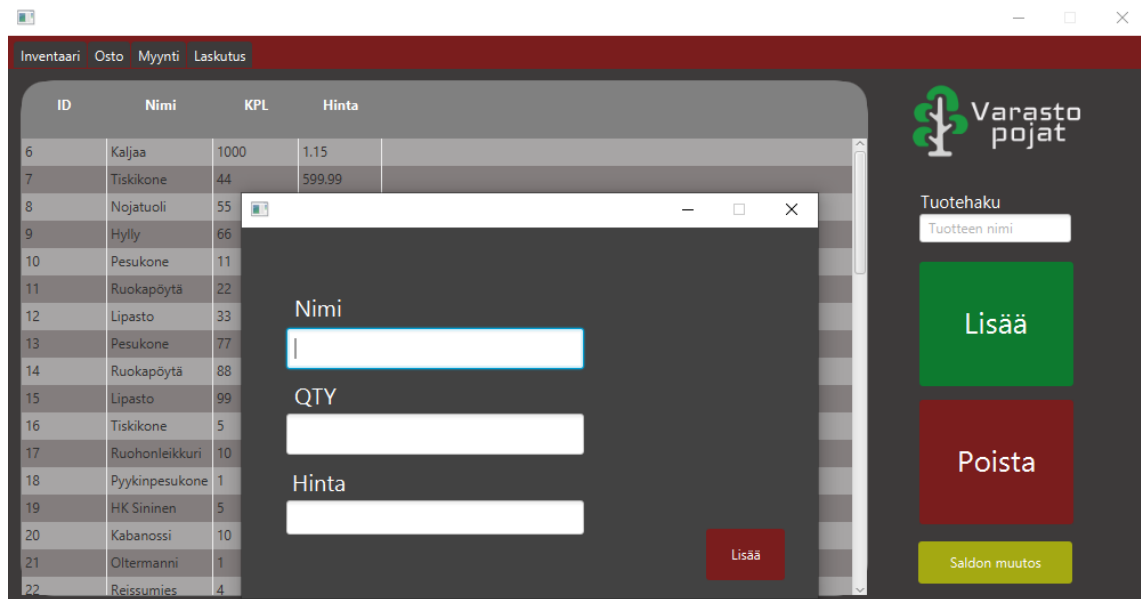
Yritysten voi olla hankala hallita alati kasvavien nimike-, tilaus- ja ostoihin liittyvää dataa. Täten tehokas varastointi edellyttääkin usein tietoteknisiä ratkaisuja helpottamaan yritysten työtaakkaa, liittyen varastointiin ja sen hallittavuuteen.

Projektin tarkoituksena on lähteä ratkaisemaan varastoinnin hallittavuuteen liittyviä ongelmia pienyrityksille tarkoitetulla varastohallintasovelluksella, joka helpottaisi yritysten arkea.

2 Työn tausta ja tavoite

Projektin tarkoituksena on luoda pienyrityksille suunnattu helppokäyttöinen varastohallintajärjestelmä. Varastohallintaohjelmalla tulisi voida tarkastaa ja päivittää varaston saldoja. Varastoon tulee pystyä tilamaan täydennyksiä, joista sovellus osaa pitää kirjaa. Käyttäjän tulisi voida myös myydä varastosta tuotteita, jotka päivittyvät saldoille ja menevät laskutukseen.

Ohjelmisto tulee toteuttamaan täysin toimivan inventaarion, johon liittyy useita tuotteita. Tuotteita pystyy poistamaan, päivittämään ja lisäämään inventaarionäkymästä (Kuva 1.), jolloin ne myös päivittyvät saldoille. Tuotteita pystytään myös hankkimaan varastoon (osto) ja myymään varastosta (myynti). Tapahtumat ovat täysin itsenäisiä toisistaan ja tallentavat tapahtumien tiedot aina omaan tietokantatauluunsa, joita käyttäjä pääsee helposti lukemaan ja muokkaamaan helppokäyttöisen käyttöliittymän kautta.



Kuva 1. Varastohallintasovelluksen inventaarionäkymä

3 Käsitteet, määritelmät

Tässä kuvataan dokumentissa käytetyt käsitteet, joita sen jälkeen käytetään järjestelmällisesti.

Inventaarilla tarkoitetaan listausta, josta näkee kaikki järjestelmässä olevat tuotteet.

Tuotteella (products) tarkoitetaan varastossa olevaa esinettä, tuotetta tai tavaraa.

Nimike tarkoittaa samaa kuin tuote.

Asiakkaalla (Customer) tarkoitetaan toimitusasiakasta, asiakas, jolle lähetetään järjestelmästä tuotteita.

Ostolla (Purchase) tarkoitetaan järjestelmään tehtävää ostotilausta, ostotilaukseen liittyy toimittaja jolta tuotteet tulevat järjestelmään.

Myynnillä (Orders) Tarkoitetaan järjestelmästä myytäviä tuotteita asiakkaille.

4 Toteutetut ja tulevat toiminnallisuudet

Tässä kappaleessa esitellään sovelluksen jo toteutuneet ja toteutukseen tulossa olevat ominaisuudet. Toiminnallisuudet on jaoteltu käyttöliittymännäkymien mukaisiin kokonaisuuksiin, joita erottavat väliotsikot. Nimet saattavat poiketa hieman Nektionissa esitetyistä.

4.1 Kirjautumisikkuna

Toteutetut:

Kirjautumisikkunan käyttöliittymä: kirjautumisella on selkeä käyttöliittymä.

Tulevat toteutukset:

Käyttäjähallinnan "backend" kokonaisuudessa: Luodaan pääkäyttäjä, joka pystyy lisäämään käyttäjiä järjestelmään.

Autentikointi: Lisätään käyttäjien autentikointi kirjautumisen yhteyteen.

4.2 Inventaarionäkymä

Toteutetut:

Varaston täydennys: varastoon pystyy lisäämään tuotteita. Tuote sisältää seuraavat attribuutit: Nimi, kpl ja hinta. Tuotteet päivittyvät tietokantaan ja ovat selkeästi nähtävissä käyttöliittymässä.

Inventaarionäkymän UI (Käyttöönotto): varaston täydennykseen liittyvät ominaisuudet ovat käytettävissä käyttöliittymästä.

Saldon korjaus (Hävikki): Varaston saldoja voi muuttaa jälkikäteen (Update).

Tulevat toteutukset:

Varaston saldojen summa käyttöliittymään: Listasta näkee rivien summan vaaka- ja pystyriveissä.

Varastopaikan lisääminen tuotteille: Varastotuotteille kuvitteelliset varastopaikat.

Raportointiin liittäminen: Inventaarion voi tulostaa esim. PDF-muodossa.

4.3 Ostonäkymä

Toteutuneet:

Työntekijä voi tilata tuotteita varastoon: Työntekijä pystyy luomaan ostotilauksen, joka päivittyy saldoille.

Ostoille liittyy toimittaja: Jokaiselle ostolle liittyy toimittaja

Toimiva käyttöliittymä ostoille: Luvattuja ominaisuuksia pystyy käyttämään käyttöliittymästä (Kesken osittain).

Tulevat toteutukset:

Raportointi: Käyttäjä pystyy tulostamaan tilauksen PDF-muodossa.

Useiden tuotteiden tilaus kerralla: Pystytään luomaan usean tuotteen tilaus (nyt voi tilata vain yhtä tuotetta kerrallaan).

Luo ostotilaus ominaisuuden tuominen UI:hin: toiminnallisuus kesken

4.4 Myyntinäkömä

Toteutuneet:

Työntekijä voi myydä tuotteita asiakkaille: Työntekijä pystyy luomaan myyntitilauksen, joka päivittyy saldoille.

Myynnille liittyy asiakas: Jokaiselle myynnillä on asiakas, jonka tiedot (osoite nimi ja kaupunki) päivittyvät myynnille.

Toimiva käyttöliittymä myynnille: Luvattuja ominaisuuksia pystyy käyttämään käyttöliittymästä.

Useiden tuotteiden myynti kerralla: Pystytään luomaan usean tuotteen tilaus (nyt voi tilata vain yhtä tuotetta kerrallaan).

Tulevat toteutukset:

Raportointi: Käyttäjä pystyy tulostamaan tilauksen PDF-muodossa.

4.5 Laskutus ja raportointi

Toteutetut:

Ei toteutuksia

Tulevat toteutukset:

Järjestelmä pystyy luomaan PDF-raportin: PDF-raportti vastaa tietokantataulussa olevaa dataa siistissä raportointimuodossa.

Järjestelmä luo laskun: Järjestelmä luo PDF-laskun. Laskussa näkyy asiakkaan tilaus ja laskutustiedot ja summan.

4.6 Muut toiminnallisuudet

Toteutuneet:

Ohjelma pystytään ajamaan ilman IDE-ympäristöä.

Ohjelmassa on kattavat yksikkötestaukset (Lisätietoja kohdassa 5.6).

Tulevat toteutukset:

Tukivalikko: järjestelmän ohjeet sovelluksessa itsessään.

5 Järjestelmän ohjelmointitekkinen toteutus

Projekti on toteutettu Apache Maven-projektinhallinta – ja kokoamistyökalun avulla. Kolmansien osapuolten kirjastot määritellään projektin src-hakemistossa olevassa POM.xml tiedostossa. Koska projekti on toteutettu ketteränohjelmistokehitysten (Scrum) periaatteiden mukaisesti, on jatkuva integraatio tärkeä osa pro-

jektia. Automatisoitu kokoaminen, testaus ja integrointi tapahtuu Jenkins-palvelimella osoitteessa: <http://10.114.32.140:8080/>. Projektin tehtävienhallinta on toteutettu Nektionilla.

5.1 Käytetyt ohjelmointikielet ja kirjastot

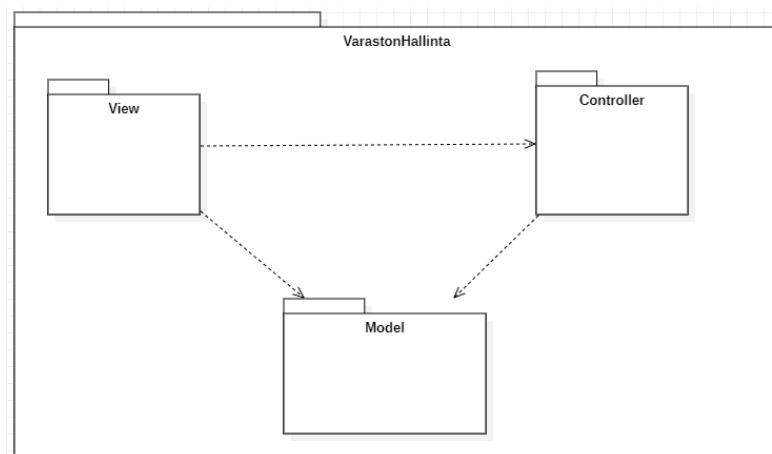
Projekti on toteutettu täysin Java-ohjelmointikielellä. Apache Maven-projektinhallinta – ja kokoamistyökalu määrittää projektin kirjastot, jotka ovat:

- Junit Jupiter 5.7.0
- Hibernate-core 5.4.27 Final
- MySQL connector 5.1.49
- JavaFX version 13

5.2 Arkkitehtuuri

Projektin noudattaa MVC-arkkitehtuuria (Kuva 2).

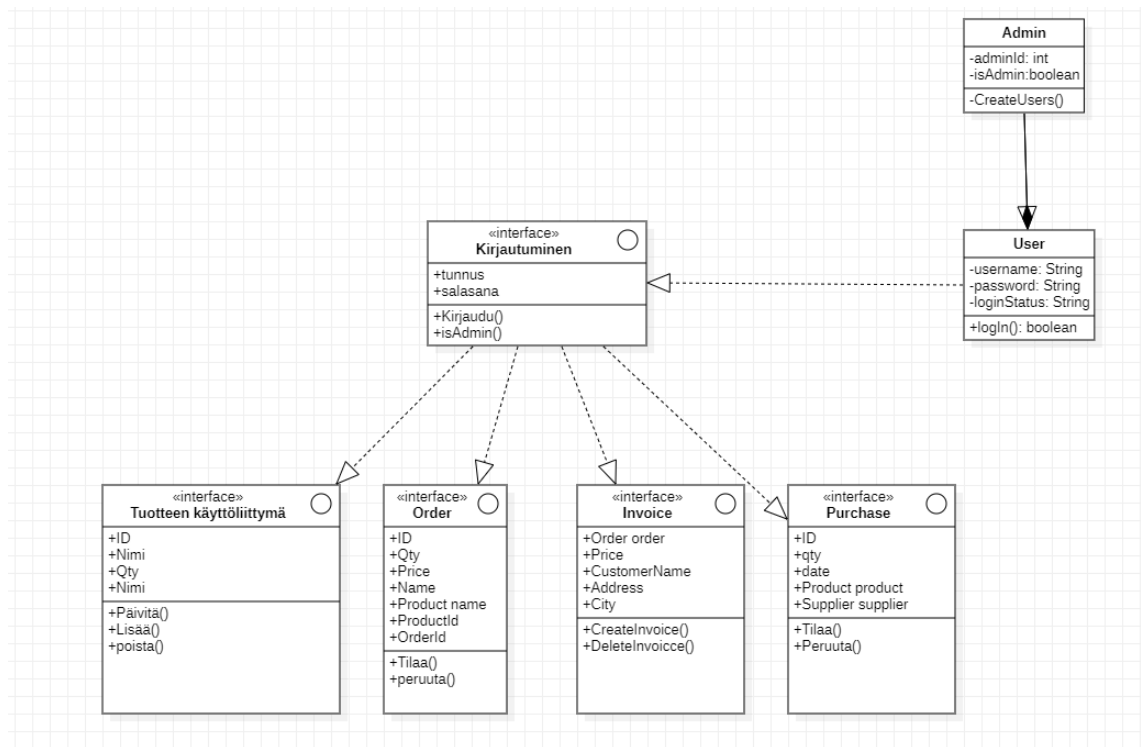
- Model-pakkaus sisältää tiedon luomiseen, tallentamiseen ja ylläpitoon liittyvät luokat.
- View-pakkaus sisältää kaikki käyttöliittymään liittyvät luokat
- Controller-pakkaus vastaanottaa käskyt ja näyttää käyttöliittymälle tarvittavat datat.



Kuva 2. pakkaukset ja niiden suhteet projektissa (MVC-malli)

5.3 Käyttöliittymän kuvaus

Ohjelmisto avautuu kirjautumisikkunalla, kirjautumisen jälkeen avautuu varsinainen ohjelma. Sovellus koostuu neljästä päänäköymästä, jotka ovat esitetty kuvassa 3 (Kuva 3). Näkymiin liittyvät toiminnot ja attribuutit ovat esitelty myös kuvassa 3. Kirjautumiseen liittyvät pääkäyttäjä ja käyttäjät ohjelmoidaan myöhemmin sovellukseen, ne eivät ole toistaiseksi käytössä.

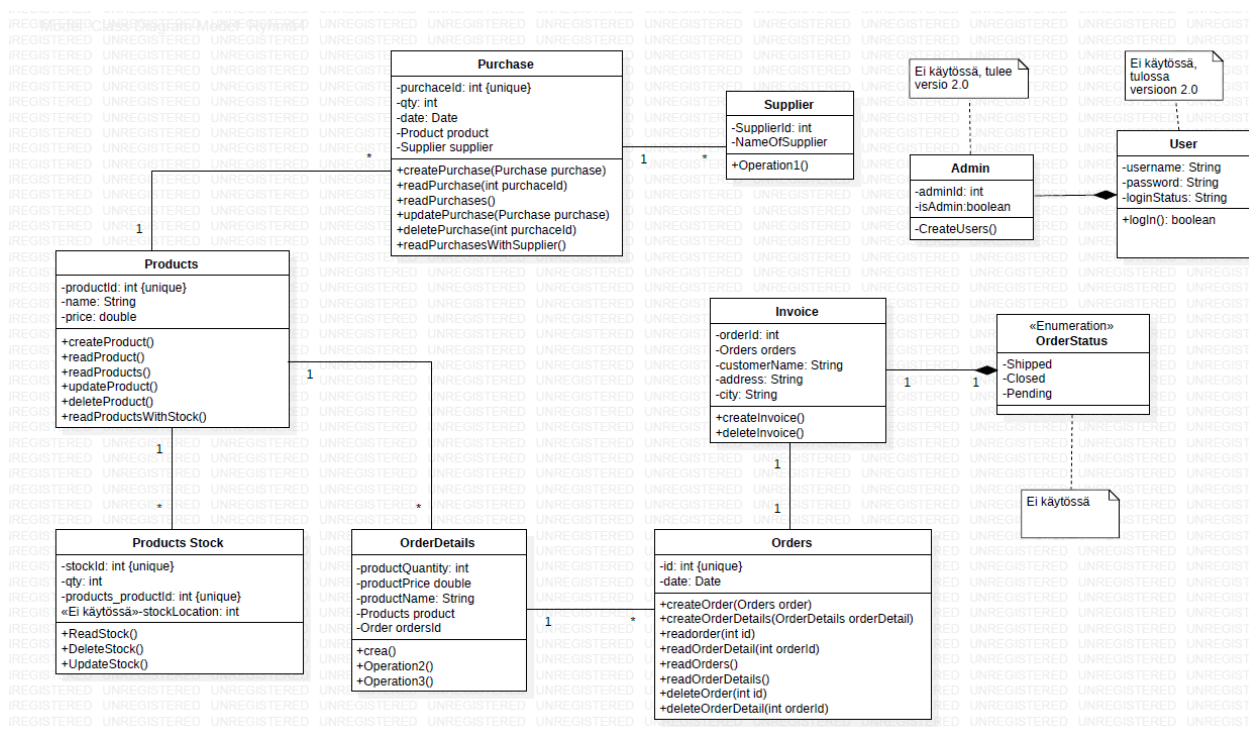


Kuva 3. kuvas ohjelmiston käyttöliittymänäkymästä

5.4 Järjestelmän keskeinen logiikan kuvaus

Järjestelmä rakentuu tietokantaolioiden ympärille (Kuva 4.), jokaiselle tietokantataululle on ohjelmoitu omat DAO:t (Data Access Object), jotka sisältävät kunkin luokan keskeiset operaatiot: luo, lue, poista, päivitä ja poista (CRUD-operaatiot). Nämä operaatiot ovat suoraan käytettävissä käyttöliittymässä, jotka kontrolleriluokka yhdistää suoraan malliin Huom. Kuvassa 4. ei ole esitetty kontrolleriluokkia.

(18)

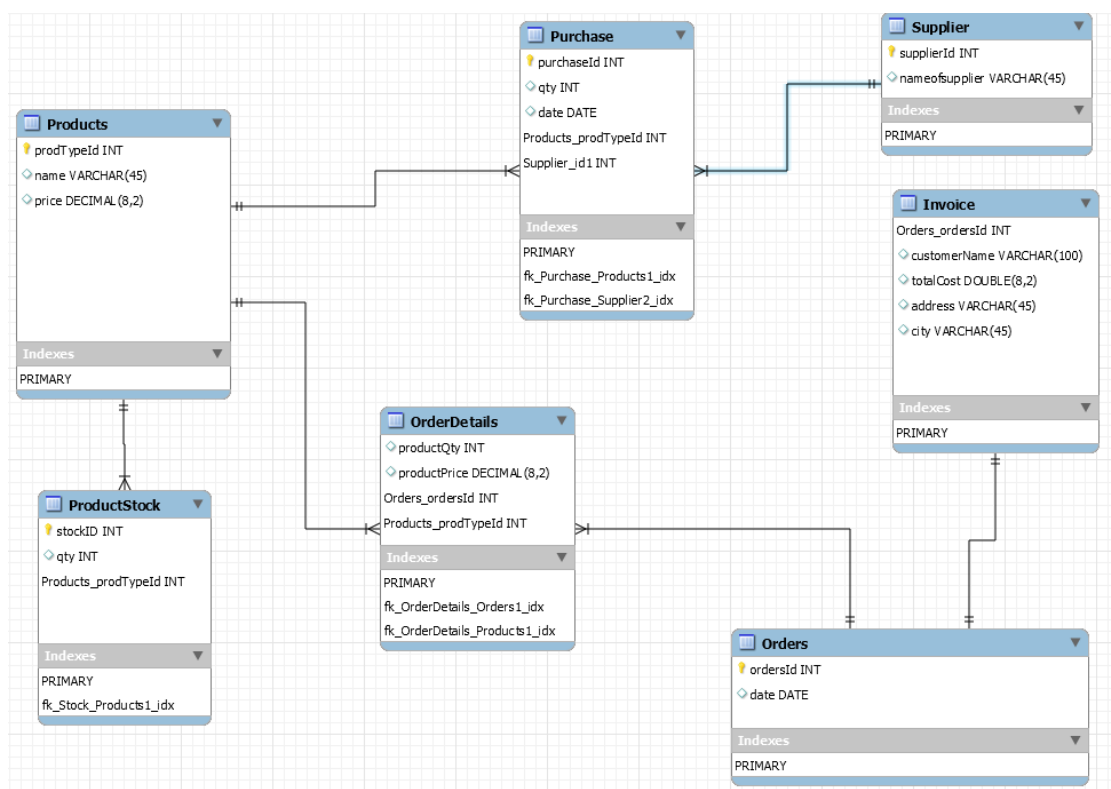


Kuva 4. Varastohallintajärjestelmän keskeisen logiikan kuvas luokkakaaviona

5.5 Ulkoisten tietovarastojen (tiedostot, tietokannat) kuvaukset

Sovelluksen tietokanta pyörii Metropolian educloud-palvelimella. Varsinainen tuotanto-tietokanta on osoitteessa 10.114.32.140:3306/varastohallinta. JUnit-testejä varten palvelimella on erikseen testaustietokanta osoitteessa 10.114.32.140:3306/testdb. Kuvassa 5. on esitetty kaikki projektin sisältämät tietokantataulut ja niiden suhteet toisiinsa.

(18)

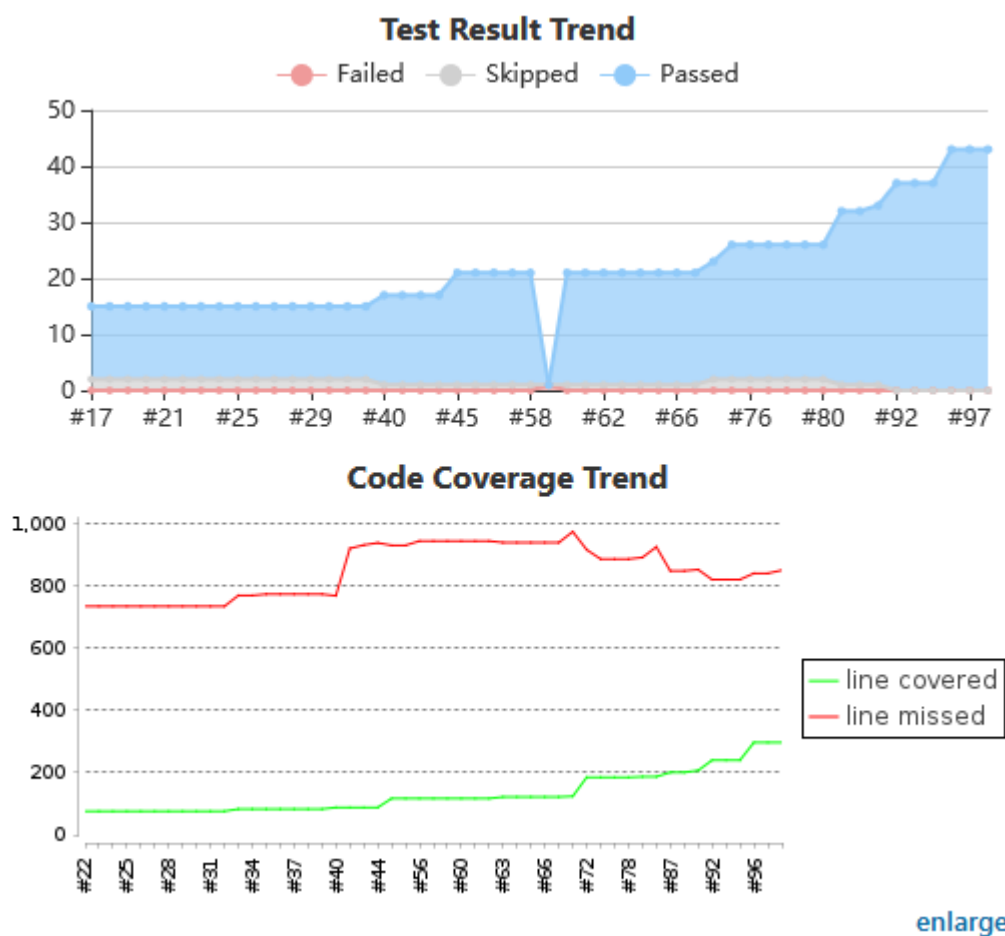


Kuva 5. Tietokantojen kuvas EER-diagrammin avulla

5.6 Testaus

Automatisoitu testaus on käytössä projektissa, jokaiselle toiminnallisuudelle on pyritty luomaan kattava testaus. Toiminnalliseen testaukseen käytämme projektissa Javan JUnit-testejä. Jokaista luokkaa kohtaan testataan tietokantaolio ja jokaista olioluokkaa testataan "*data access objektin*" kautta. Tietokantaluokkia testataan, jotta saadaan varmuus tietokantojen luomisen oikeellisuudesta. Testit on luotu myös lähes kaikkiin tietokantoihin liittyvistä olioista. Kaikki testit ajetaan jokaisen "buildin" jälkeen Jenkins-palvelimella, joka tuottaa kattavan raportoinnin koskien testaustamme (Kuva 6.)

(18)



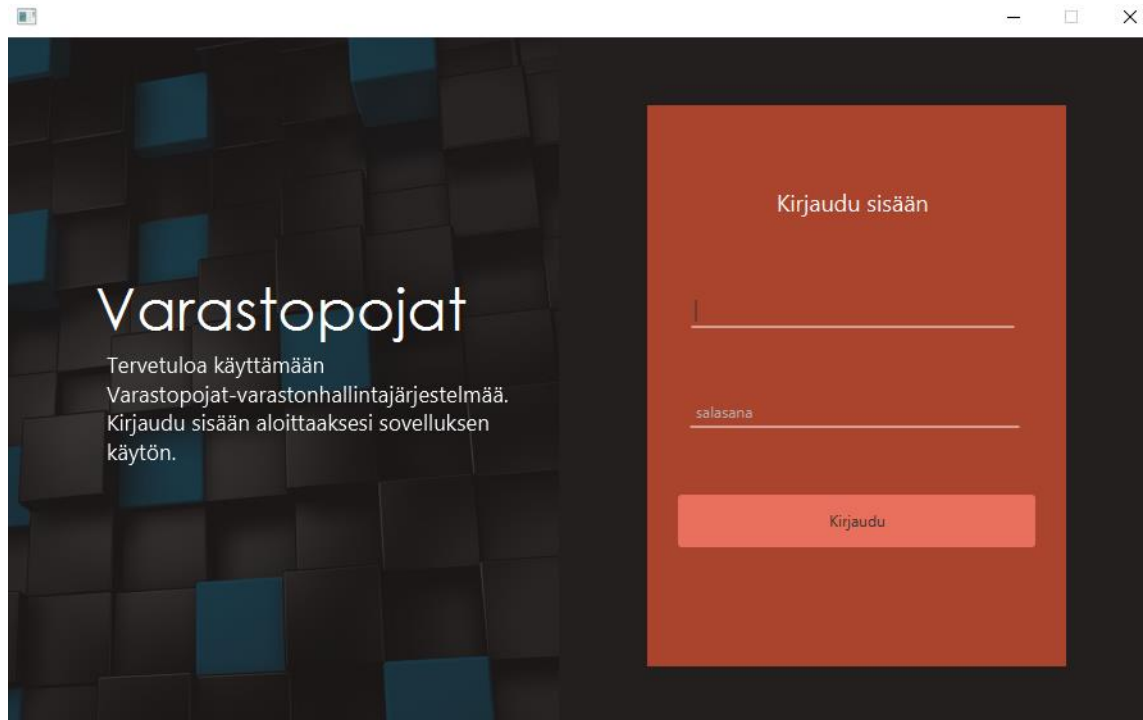
Kuva 6. Testauksen trendit Jenkins-palvelimella 10.3.2021

6 Varastosovelluksen käyttöohje

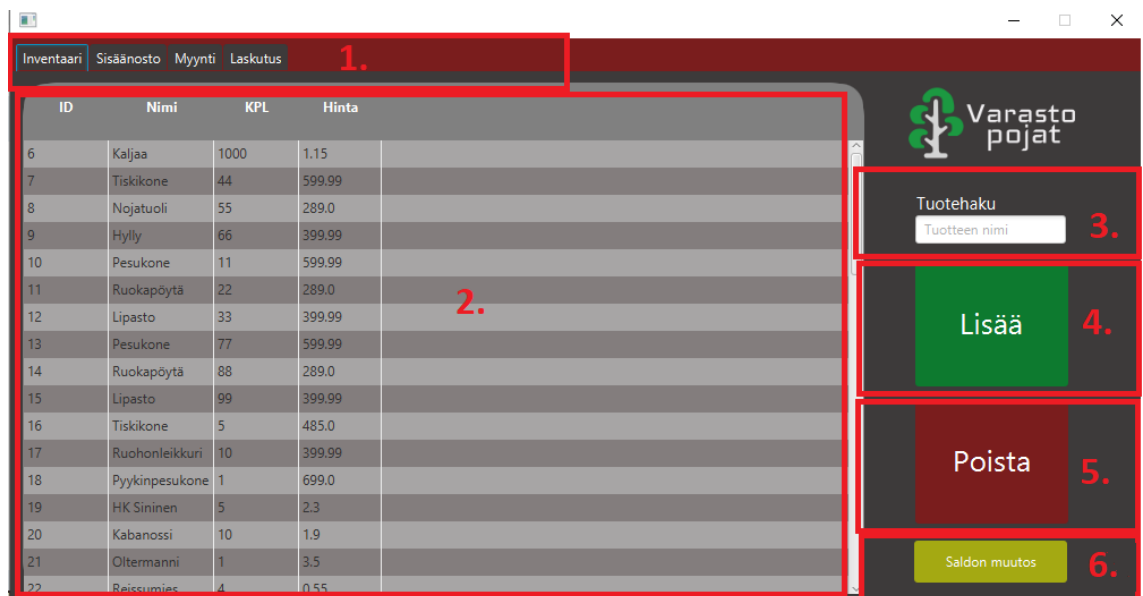
Varastopojat varastonhallintasovelluksen käynnistämisen jälkeen ohjelmistossa aukeaa kirjautumisikkuna (Kuva 7), jolla kirjaututaan varsinaiseen ohjelmaan. Ohjelmiston pääkäyttäjä (Admin) luo jokaiselle käyttäjälle tunnukset ohjelmistoon erikseen. Käyttäjä ei voi palauttaa tai luoda omaa käyttäjätunnusta, saati salasanaa.

Kirjautumisikkunassa (Kuva 7) käyttäjän tulee kirjoittaa hänelle erikseen luovutettu käyttäjätunnus ensin ja siihen liittyvä salasana, tämän jälkeen sovellukseen voi kirjautua ”kirjautu”-painikkeella.

(18)



Kuva 7. Varastopoijat-sovelluksen kirjautumisikkuna



Kuva 8. Sovelluksen päänäköymä

(18)

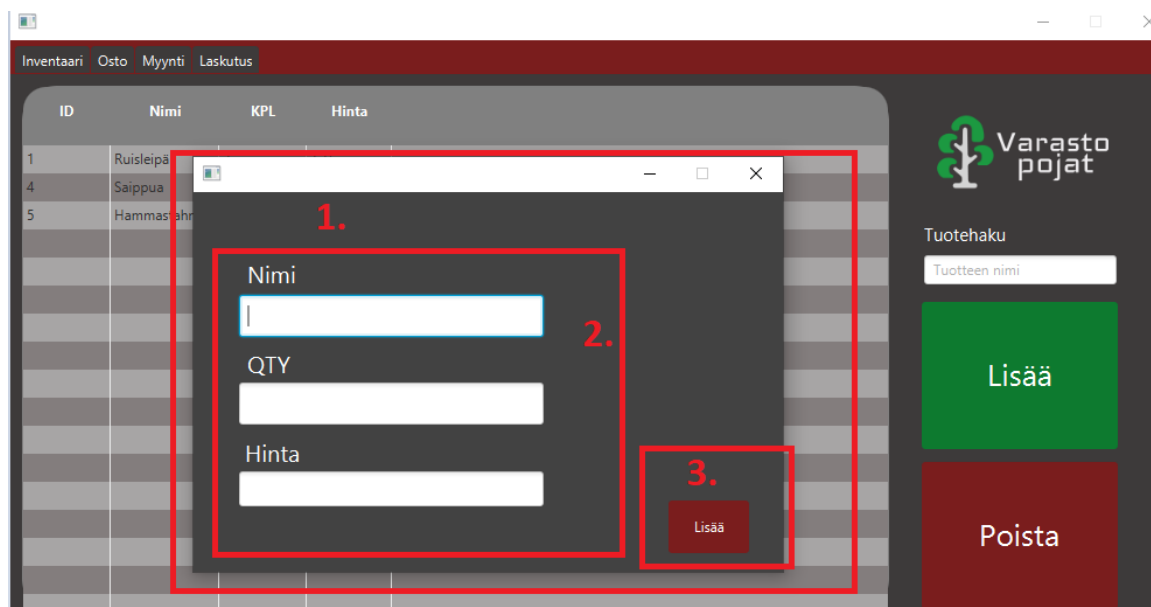
6.1 Inventaarinäkymän ohje (päänäkymä)

Kirjautumisen jälkeen ohjelma avaa automaattisesti sovelluksen pääikkunan eli inventaarinäkymän (Kuva 8.). Tämä näkymä muodostuu seuraavista toiminnallisuuksista:

1. Sovelluksessa navigointi on toteutettu yläreunassa löytyvillä välilehdillä (Kuva 8, kohta 1). Klikkaamalla välilehteä ohjelmisto päivittää näkymän käyttäjän haluamaan kohtaan (Inventaari, Osto, Myynti ja Laskutus).
2. Sovelluksen päänäkymä (Kuva 8, kohta 2). Tähän tulostuu näkymälle oleellinen tieto. Näkymän kentät vaihtelevat sen mukaan, minkä näkymän käyttäjä on sovelluksen yläreunasta kulloinkin valinnut. Kuvan esimerkissä käyttäjä on valinnut kohdan inventaari, jolloin sovellus tulostaa käyttäjälle tuotetiedot (Nimi, kpl ja hinta).
3. Näkymän haku (Kuva 8, kohta 3). Jokaisella näkymällä on oma haku (Haku nimellä). Haulla voi helposti löytää käyttäjän haluaman tuotteen, useiden tuotteiden joukosta.
4. Lisää-painike (Kuva 8, kohta 4). Painikkeella otsikointi vaihtelee kulloinkin valitun näkymän mukaisesti, toiminto on silti lähes sama jokaisessa näkymässä. Painikkeella voidaan lisätä tuotteita, tehdä tilauksia tai ostaa varastoon tuotteita. Painikkeen painamisen jälkeen sovellus avaa automaattisesti tarkentavan kyselylomakkeen (Kuva 9).
 1. Syöttökentät (Kuva 9, kohta 1). Näihin kenttiin syötetään kulloinkin pakollinen tieto liittyen tuotteen lisäykseen, myyntiin tai ostoon.
 2. Syöttöalue (vaihtelee näkymittäin)

(18)

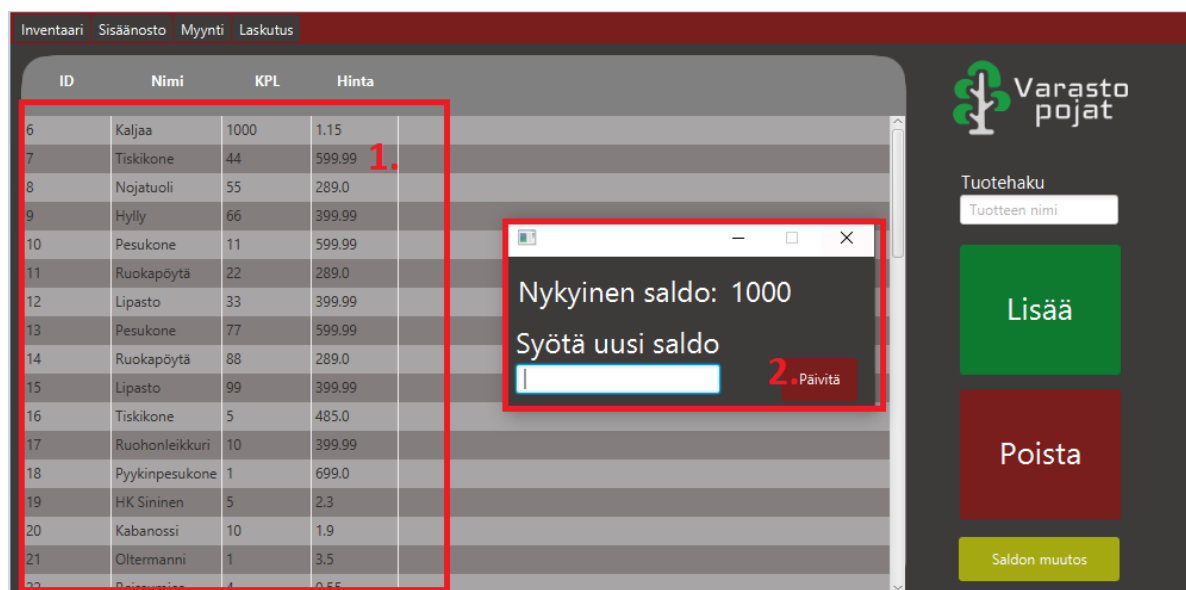
3. Pakollisten tietojen jälkeen painetaan "Lisää"-painiketta, jolloin tuotteet tallentuvat sovelluksessa (Kuva 9, kohta 3)



Kuva 9. Lisää-näkymä

5. Poista-painike (Kuva 9, kohta 5). Poista painikkeella voidaan poistaa yksittäisiä tuotteita, myyntitilauksia tai ostotilauksia. Poistaminen vaatii aina poistettavan tuotteen id-numeron. Poisto-nappulasta tapahtuva poistaminen on aina lopullinen ja kyseinen tieto poistuu tietokannasta pysyvästi.
6. Päivitä painike. **HUOM. Listata pitää valita päivitettävä rivi, ennen päivitysoperaation aloittamista** (Kuva 10). Avautuvaan kenttään syötetään haluttu saldo ja valitaan kohta "päivitä".

(18)



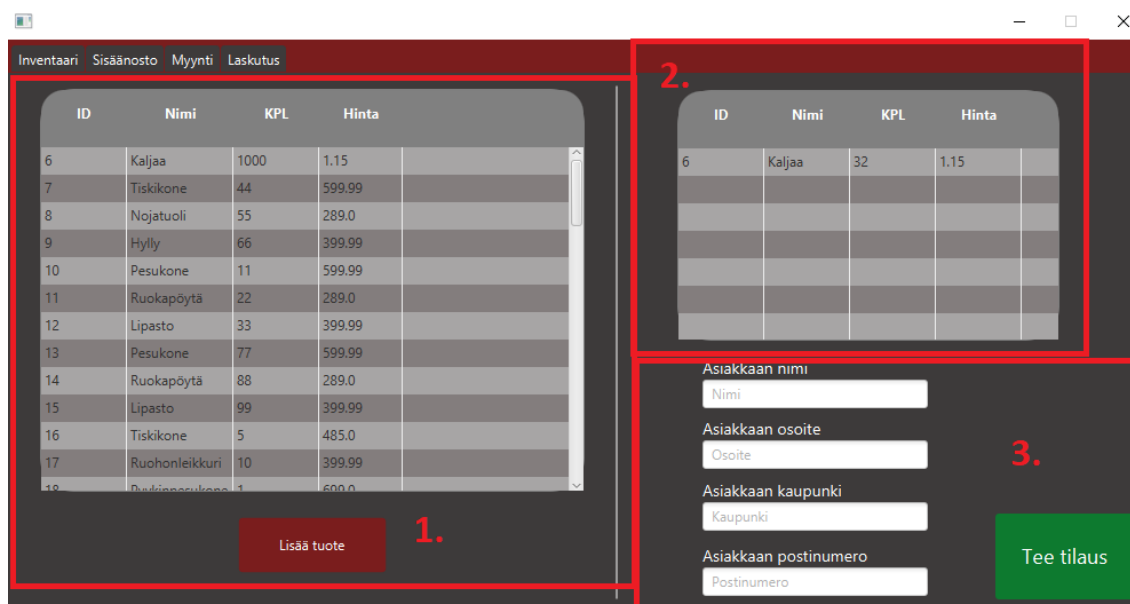
Kuva 10. Päivitä-näkymä

6.2 Myyntinäkömän ohje

Tuotteita pystyy myymään asiakkaille avaamalla myyntivälilehden. Myyntitilaus tapahtuu valitsemaan vasemmalta (Kuva 11, kohta 1) olevasta listasta halutun tuotteen. **HUOM.** **Tuoterivi pitää valita ennen kuin tuotteen voi lisätä** oikealle olevalle listalle, "lisää tuote"-nappulalla.

Haluttujen tuotteiden ollessa oikealla olevassa listassa (Kuva 11. kohta 2). Voidaan syöttää asiakkaan tiedot (Kuva11, kohta 3) jonka jälkeen "Tee tilaus"-nappula luo tilauksen ja tallentaa tilauksen tiedot tietokantaan.

(18)



Kuva 11. Myyntinäkömä varastohallintasovelluksessa.

6.3 Ostonäkymän ohje (Kesken)

Ostonäkymällä pystyy ostamaan tuotteita sisälle varastoon. Ostotilausikkuna toteuttaa samanlaista näkymää, kuten sovelluksen muutkin osastot (Kuva 12).

Sovelluksen päänäkömä (Kuva 12. kohta 1), Näyttää ostotilausten luontipäivämäärän, tuotteen Id:n ja ostettujen tuotteiden lukumäärän ja toimittajannimen. (Huom. Osio on hieman kesken varsinaisessa ohjelmassa).

Ostotilauksiin liittyy toimittaja, joita pystyy luomaan "luo toimittaja-napilla" (Kuva 12. kohta 2). Toimittajat tallentuvat omaan tietokantaansa.

Ostotilauksia voi luoda (tulee myöhemmässä versiossa) ja poistaa. Ostotilauksella luodut tilausrivit päivittyvät suoraan tuotteiden inventaarioon (Kuva 12. kohta 3 ja 4).

(18)

ID	Tuote	KPL	Päivämäärä	Toimittaja
5	10	10	10-03-2021	
6	10	10	10-03-2021	
8	10	10	10-03-2021	
30	15	10	10-03-2021	
31	10	10	10-03-2021	
32	20	10	10-03-2021	
33	25	10	10-03-2021	
32	40	10	10-03-2021	
34	10	10	10-03-2021	
35	10	10	10-03-2021	
39	5	10	10-03-2021	
48	10	10	10-03-2021	
49	10	10	10-03-2021	
51	10	10	10-03-2021	
73	15	10	10-03-2021	
74	10	10	10-03-2021	

Kuva 12. Ostonäkymä varastohallintasovelluksessa.

6.4 Raportoinnin ohje

Toteutetaan seuraavaan versioon.