CT60A4301 Tietokannat

**Harjoitustyö 1 – Tietokannan suunnittelu**

Lappeenrannan teknillinen yliopisto

Innovation and Software (IS), LUT LBM

CT60A4302 Tietokannat

Kevät 2017

0439302(op. nro.) Joni Kettunen

Joni.kettunen@student.lut.fi

SISÄLLYSLUETTELO

[SISÄLLYSLUETTELO 1](#_Toc482780392)

[1 määritys 2](#_Toc482780393)

[2 käsitemalli 3](#_Toc482780394)

[3 tietokantatoteutus 5](#_Toc482780395)

[4 keskustelu 9](#_Toc482780396)

# määritys

Tietokanta lähettiin kehittämään varastonhallintaan. Tärkein tietokannan toiminnallisuus on varastoida tietoa varastosta sekä varastotasosta. Tietokantaan pystyy asettamaan asiakkaiden tilausrivejä poispäin varastosta sekä toimittajien tuontitilausrivejä varastoon päin. Tietokannan käyttäjiä ovat yrityksen johto, sihteerit ja työntekijät (varastomiehet). Tietokannan python kielellä toteutettu käyttöliittymä on tarkoitettu manuaalista syöttöä varten, ja jos mikäli tietokanta olisi oikeassa käytössä se olisi integroitu automaattiseen graafiseen tilausjärjestelmään.

Tietokannasta löytyy myös toiminnanohjausjärjestelmään liittyviä elementtejä, joiden avulla pystytään hallinnoimaan varastoa tehokkaammin. Näitä on varmuusvarastotaso ja sen alittamisen tarkkailu, työntekijöiden työtehokkuuden valvonta ja toimittajien toimitusaikojen valvonta.

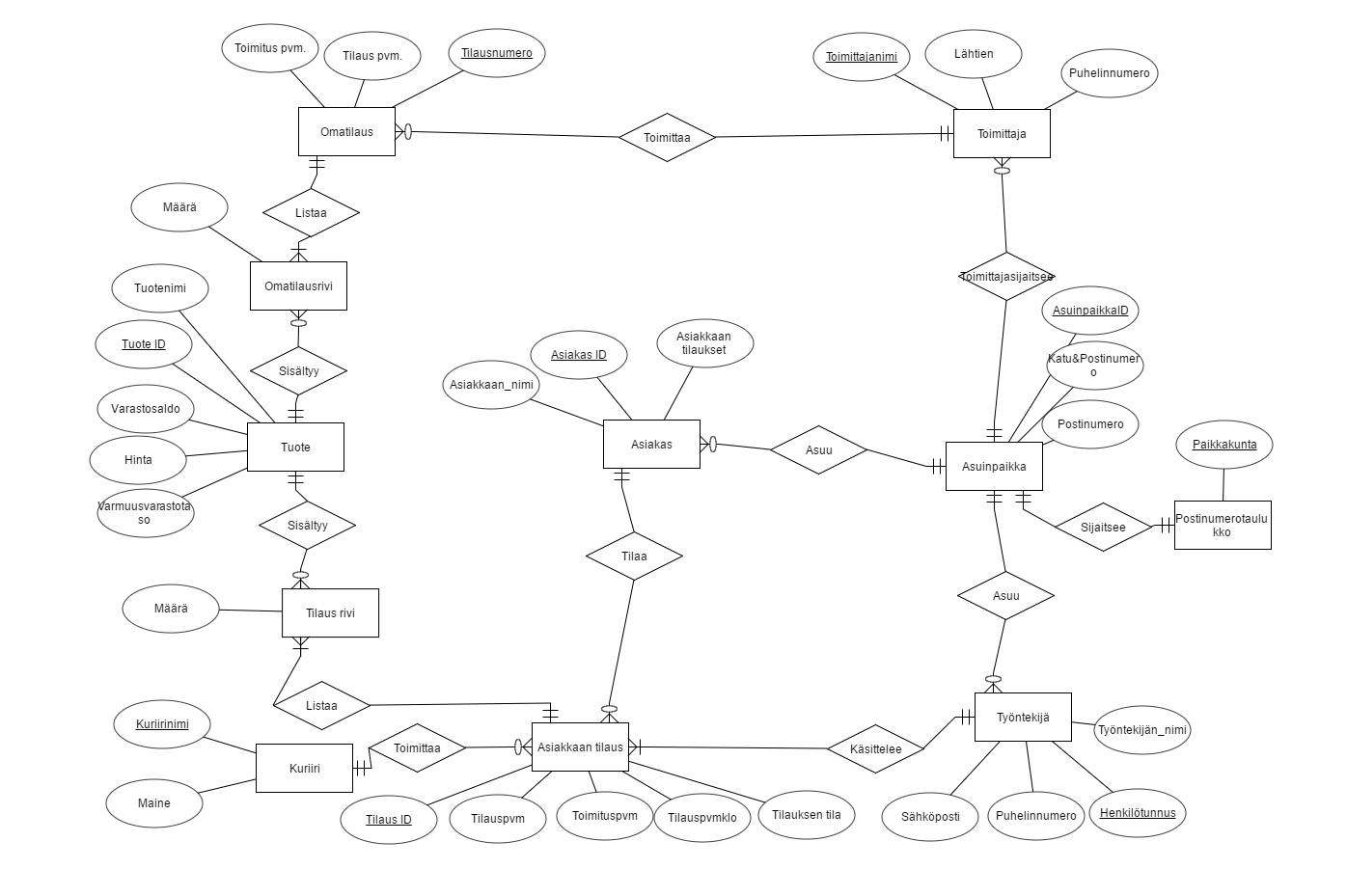
**Tietokannan perustoiminnallisuus**

Tietokannassa on kolme käyttäjäryhmää. Ylläpitäjä (admin) , sihteeri sekä varastomies. Varastomiehellä on pääsy vain tarkkailemaan asiakkaille päin lähteviä paketteja, jotta he voisivat paketoida ja käsitellä näitä eteenpäin. Sihteerillä on oikeudet toteuttaa lähes kaikkia toimintoja tietokannassa. Sihteeri pääsee käsiksi omiin tilauksiin sekä asiakkaille päin meneviin tilauksiin. Sihteeri ei kuitenkaan pääse käsiksi toiminnanohjausjärjestelmän elementteihin. Admin käyttäjällä on pääsy kaikkiin tietokannan toimintoihin.

Tehdessä asiakkaalle tilausta tulee olla tiedossa, kuka on tilaamassa, mitä hän on tilaamassa ja miten paljon. Tilaukset syötetään järjestelmään tilausriveinä, joten käyttöliittymän kautta tilausten teko on varsin hidasta. Syötettäessä tilausta järjestelmään voidaan valita onko tilaus juuri vastaanotettu, käsittelyssä vai lähetetty.

Omaa tilausta tehdessä tulee olla tiedossa toimittaja, tilattava esine sekä tilattava määrä. Tarvittaessa luodaan uusi tuote tuotekatalogiin ja lisätään uusi toimittaja toimittajien listaan.

# käsitemalli

Suunnittelin tietokannan ER- käsitemallin erdplus (www.erdplus.com) ohjelmalla.

Kuva 1. ER- diagrammi

ER-diagrammin keskiössä on asiakas, joka on tietokantaa palvelevalle yritykselle kaiken A ja O. ER- malli koostuu kaikenkaikkiaan 31 eri attribuutista, 12 ei suhteesta ja 11 eri yksilötyypistä. Pääavaintyypit on alleviivattu kuvioihin. Tietokannan väliset suhteet on esitetty Hoffer, Prescott & McFaddenin notaation mukaisesti. Vierasavaimia ei suoraan pysty erottamaan tästä ER-diagrammista, mutta ne voidaan katsoa nopeasti relaatiotaulumallista, jota on käytetty enemmän avuksi tietokantaa kirjoittaessa SQLLite:llä. En myöskään sisällyttänyt ER malliini attribuuttien tyyppiä (INT, VARCHAR, DATE, DATETIME ) jne. Ne löytyvät helpoiten tietokannnanluontikoodista palautuksen oheessa. Myös CHECK ja NOT NULL yms. Rajoitukset löytyvät helpoiten luettavissa tietokannanluontikooditiedostosta.

Käsitemallia tehdessäni tajusin, että yksinkertainen idea varastonhallintatietokannasta olisikin huomattavasti kompleksimpi mitä ajattelin. Vaikeuksia oli miettiessä, miten toteutan tilaukseen sisältyvät tuotteet, mutta päädyin ratkaisuun, jossa tietokantaan syötetään tilausrivejä. Tilausriveistä saa tällöin haettua sopivalla Queryllä kaikki yhden asiakkaan tiettynä ajanhetkenä tilaamat tuotteet, jotka voidaan pakata samaan lähetykseen. Näin jälkikäteen ajatellen olisi pitänyt rakentaa insert – queryt niin, että query olisi osanut luoda usean tilausrivin kerralla, mutta tämä ei juolahtanut mieleeni kun suunnittelin tietokantaa ja insery query rakenteita.

# https://puu.sh/vRDkL/96e8f42cb4.pngtietokantatoteutus

Kuva 2. Relaatiotaulut

**Yleistä:**

Suurin hankaluus ER mallia relaatiotauluihin transformoidessa oli Omatilausrivikoostuu ja asiakastilausrivikoostuu taulut. Molemmat ottaa vierasavaimet omalta tilaukselta tai asiakkaan tilaukselta, sekä tuote taululta tuote\_ID:en. Mieleeni tuli erilaisia yhdistelmä pääavain rakenteita, mutta näiden käsittely oli jäänyt kurssilla todella pieneksi. Päädyin rakenteeseen, jossa jokaisella tilausrivilläkin oli vielä tunnistava tilausrivi\_ID.

Vaikka en ER- diagrammia tehdessäni kiinnittänyt huomiota normaalisääntöihin ja niiden toteutumiseen tietokannassa, huomasin että olin jo ottanut normaalisääntöjä er-diagrammissa huomioon. Tietokannassa on yksi asuinpaikka taulu, joka pystyy antamaan asuintiedot Toimittaja, Asiakas ja työntekijä tauluille. Myöskin postinumerotaulu on tehty erilliseksi, jotta rakenne ei rikkoisi normaalisääntöjä.

Käyttöliittymän rakentaminen muodostui suureksi haasteeksi, koska vähän kaikki oli linkitettyinä toisiinsa tietokannassa. Päädyin tekemään kaikki tietokantaan tiedon lisäykset customoiduiksi, enkä lisää tietokantaan missään vaiheessa tietoa annetulla geneerisellä aliohjelmalla. Taulun muutokset ovat tietokannassa geneerisen mallin mukaisia, joka hankaloittaa käyttöliittymän käsittelyä ainakin kokemattomalla käyttäjällä. Mikäli aika ei olisi loppunut kesken, olisin tehnyt oman tilauksen ja asiakkaan tilauksen muokkaukset customoiduiksi varianteiksi, sillä tilaus pitää saada nopeasti asetettua lähetetyksi tai omatilaus vastaanotetuksi, joka olisi johtanut varaston kasvamiseen tai vähentymiseen automaattisesti.

**Toteutustapa:**

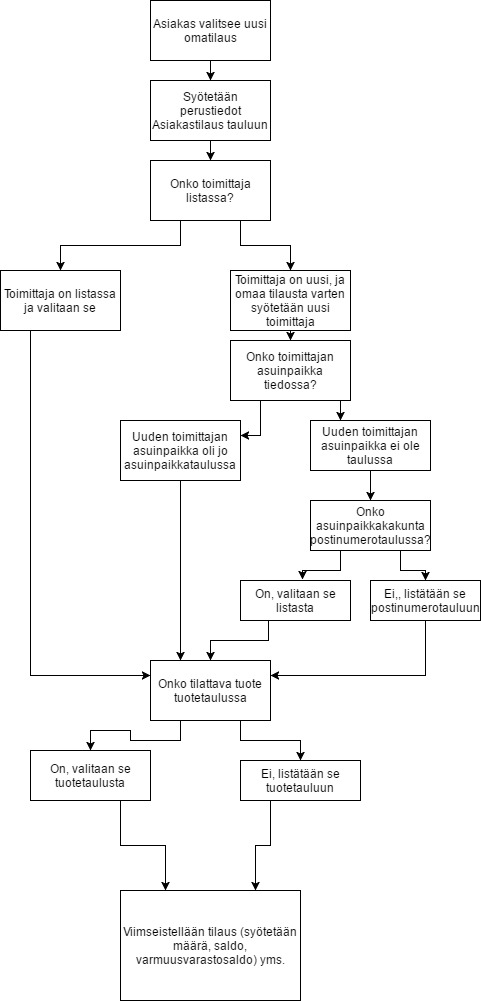
Käytin käyttöliittymässä python koodipohjaa. Annettua ohjelmapohjaa on muutettu useassakin kohtaa, jotta sain sen yhteensopivaksi tietokantani kanssa. Lisäksi insert queryt on kirjoitettu kaikki customeiksi.

**Testatessa huomioitavaa:**

Käyttöliittymän pitäisi toimia, kun sille antaa vain oikeassa muodossa olevia syötteitä. Mikäli käyttöliittymään antaa vääränmuotoisia syötteitä, koituu tämä joissain tilanteissa vielä ohjelman kaatumiseen.

**Koodissa huomioitavaa:**

Eheyssääntöihin liittyviä tarkistuksia on toteutettu CHECK- määrityksin taulujen rakenteissa. Käyttöliittymä osaa tehdä joitakin perussyötteiden tarkastuksia. Käyttöoikeudet on toteutettu ohjelmapohjan näyttämällä tavalla.



Kuva 3. UI- ketju omaa tilausta luodessa

Tauluhin tietueiden lisäykset on suoritettu ”ketjuina”. Esimerkiksi syötettäessä omaa täydennystilausta järjestelmään, ohjelma kysyy käyttäjältä ensin perustiedot tilauksesta. Tämän jälkeen valitaan toimittaja listasta, ja mikäli se ei ole siellä, ohjelma osaa automaattisesti mennä lisäämään toimittajaa käyttäjän niin valitsiessa. Ja mikäli uudella toimittajalla on uusi asuinpaikka, osaa ohjelma mennä seuraavaksi lisäämään tietoja asuinpaikkatauluun. Ja mikäli asuinapaikka sijaitsee uudessa kaupungissa, menee ohjelma lisäämään tietoja postinumerotauluun postinumeroaliohjelmalla. Jälleen kun valitaan omaan tilaukseen tuotteita, valitaan onko tilattava tuote jo nykyisessä tuotekatalogissa. Mikäli se ei ole osaa ohjelma mennä omantilauksensyöttöprosessin yhteydessä lisäämään tuotteen tuote tauluun.

# keskustelu

Pyrin tekemään harjoitustyön niin, että saisin ”kiittäen hyväksytty” muodosta 10 lisäpistettä. Tietokannassa on 11 taulua, ja eheyssänntöjä tarkkaillaan CHECK- lauseilla taulujen rakenteissa. Kyselyissä on lukuisia JOIN lauseilla toteutettuja kyselyitä. Laajat kyselylauseet ovat olleen välttämättömiä varastonhallintatietokannan sisältöä tarkkailtaessa.

Opin tietokantaa suunnitellessa ja käyttöliittymää luodessa, miten jännää tietokannan toiminnan suunnittelu voi olla. On mielenkiintoista luoda kuvaa reaalimaailmasta tietokantaan, jota voidaan sitten hyödyntää toimintaa tehostaessa.