Tietokannat-kesäkurssi

Tietokannan suunnittelu

Lappeenrannan teknillinen yliopisto
Tietotekniikan koulutusohjelma

CT60A4302 Tietokannat
Kesä 2018

Mikko Nummila
mikko.nummila@student.lut.fi

SISÄLLYSLUETTELO

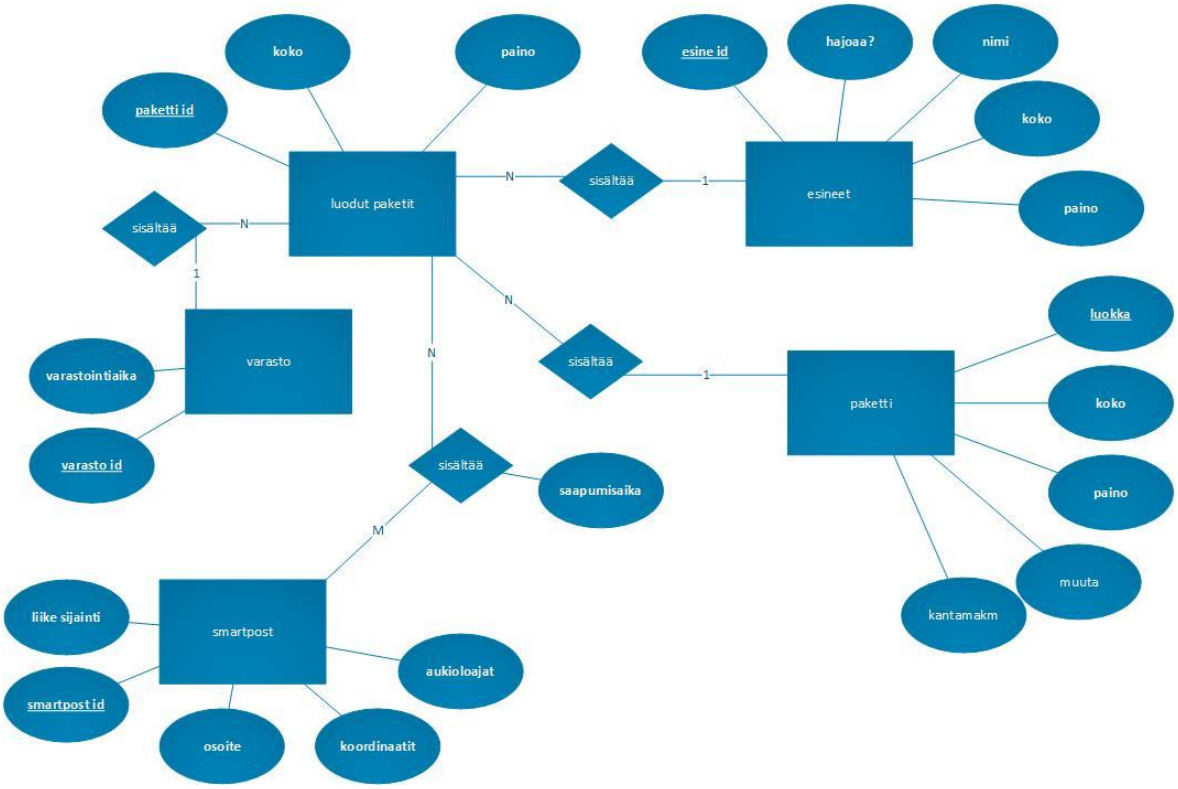
1 MÄÄRITYS

Harjoitustyössä toteutettiin Toiminnaltaan Itellaa Muistuttava Ohjelmisto, Tietokantaa Edellyttävä Integraatio eli TIMOTEI -ohjelmalle tietokanta. Tietokanta toimii javalla tehdyn graafisen käyttöliittymän taustalla ja sitä käyttävät ohjelman asiakkaat. Ohjelma luo tietokannan, jos sitä ei vielä ole ja käyttäjä voi halutessaan aloittaa aina puhtaalta pöydältä resetoimalla tietokannan. Ohjelman kaikki tiedon käyttäminen suoritettaisiin tietokannan kautta, jotta ohjelmaa voi käyttää samasta pisteestä myös uudelleenkäynnistämisen jälkeen.

TIMOTEI käyttää valmista SmartPost verkkoa ja niiden tietoja joita saadaan avoimesti. Tietokantaan pitää pystyä lisäämään smartpost-automaatteja ja niillä täytyy ainakin olla osoite, koordinaatit, aukioloajat, ja käyttäjäystävällisempi sijainti eli missä marketissa automaatti on. Tietokantaan pitää pystyä lisäämään paketteja, jotka kuuluvat kolmeen eri pakettiluokkaan ja joka luokalla on eri ominaisuuksia kuten paino, koko ja kantama. Paketit sisältävät erilaisia esineitä ja niilläkin on erilaisia ominaisuuksia kuten nimi, koko, paino ja tieto siitä onko esine särkyvä. Kun paketti luodaan, se laitetaan varastoon odottamaan lähetystä ja varastosta valitaan luotu paketti, joka sitten lähetetään automaattien välillä. Käyttäjän pitää pystyä käyttöliittymän kautta luomaan ja poistamaan luotuja esineitä ja paketteja sekä lisäämään ohjelman kartalle automaatteja ja kaikki tämä tieto kulkee tietokannan kautta. Käyttäjän tulisi nähdä myös kattava listaus tietokannassa olevista esineistä paketeista ja automaateista.

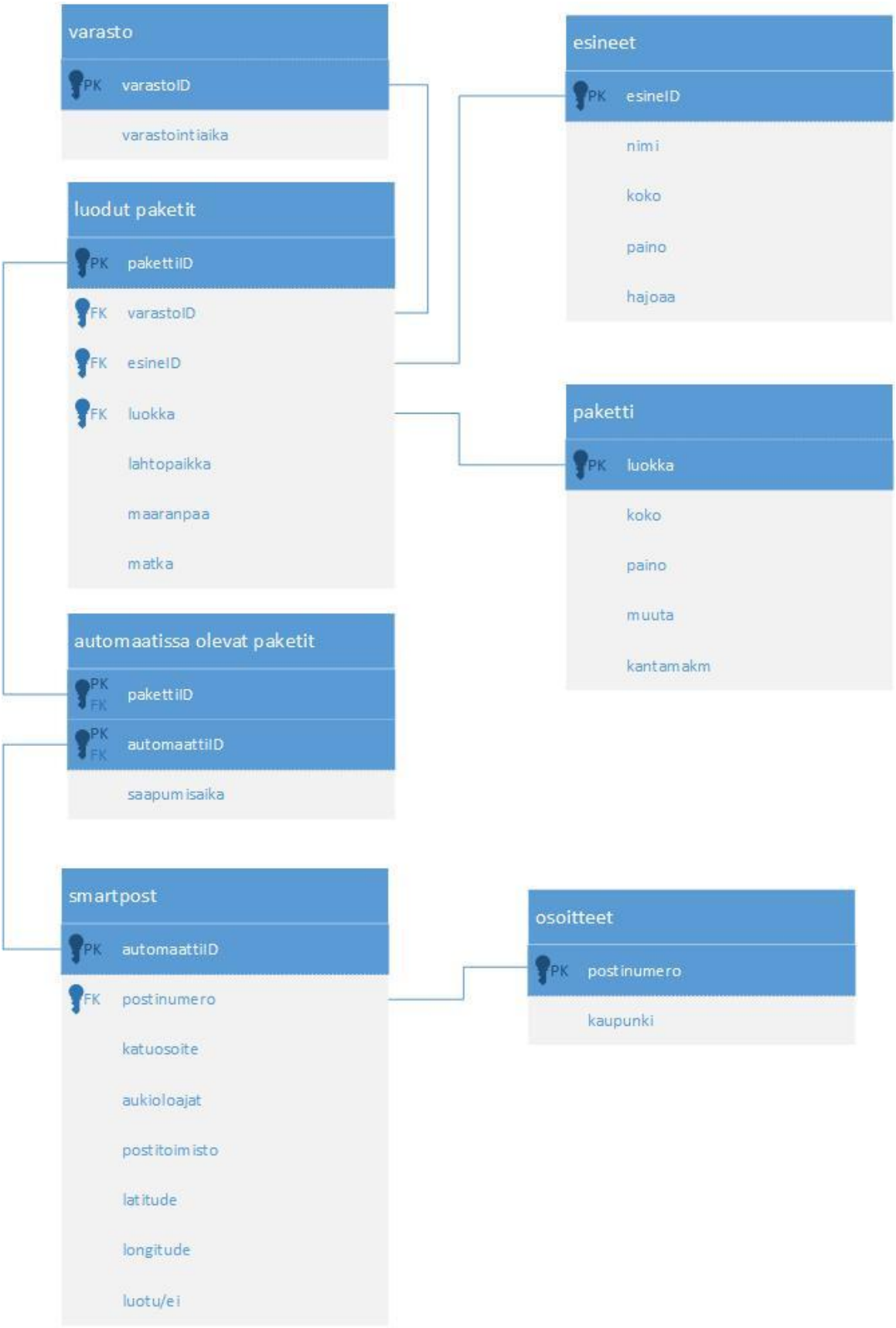
Työssä pitää toteuttaa ainakin seuraavat kyselyt: 1. Listaa kaikki automaatit tietoineen, 2. Listaa paketit siten, että siitä käy ilmi paketin luokka, minkä esineen se sisältää, mistä mihin paketti on menossa ja sen tämänhetkinen sijainti, 3. Listaa esineet tietoineen, 4. Hakea tietyn smartpostin koordinaatit, 5. Listata kartalle lisätyt smartpostit.

2 KÄSITEMALLI



Kuva 1.

Kuvassa 1. on ensimmäinen tehtävänannon perusteella tehty käsitemalli. Siihen on merkitty yksilötyypit, yhteystyypit ja niiden asteet sekä attribuutit. Pääavaimet ovat alleviivattu ja vaaditut attribuutit ovat boldattu. Relaatiomalliksi muuntamalla vierasavaimiksi tulisi 1, N suhteissa 1 puoleinen pääavain ja N, M suhteessa tulisi uusi taulu jossa on kaksoispääavain ja vierasavaimet. Lähes kaikille attribuuteille vaaditaan arvot ja kaikki mitattavat arvot kuten paino, koko, pituus pitää olla suurempia kuin nolla. Paketin attribuutit muuta defaultataan 'ei' ja kantama defaultataan 1000km. Esineet taululle ei löytynyt uniikkia arvoa millekään attribuutille, joten sille annettiin ID numero ja luodulla paketilla ja smartpostilla oli sama tilanne. Paketti kohdetyypin attribuutti luokka päätyi pääavaimeksi koska, jokaista luokkaa on vain yksi kappale. Kuvassa 2. on lopullinen tietokanta relaatiomalliksi muutettuna ja normalisoituna.


varasto
PK varastoID
varastointiaika
esineet
PK esineID
nimi
koko
paino
hajoaa
luodut paketit
PK pakettiID
FK varastoID
FK esineID
FK luokka
lahtopaikka
maaranpaa
matka
paketti
PK luokka
koko
paino
muuta
kantamakm
automaatissa olevat paketit
PK FK pakettiID
PK FK automaattiID
saapumisaika
smartpost
PK automaattiID
FK postinumero
katuosoite
aukioloajat
postitoimisto
latitude
longitude
luotu/ei
osoitteet
PK postinumero
kaupunki

Kuva 2.

3 TIETOKANTATOTEUTUS

Yleistä:

Muuttaessa käsitemallia relaatiomalliksi esiin tuli smartpostin osoitetietojen tarvitseminen tarkemmin, joten attribuutti osoite jaettiin katuosoitteeseen, kaupunkiin ja postinumeroon. Tästä seurasi normalisoinnissa se, että postinumerot ja kaupungit eivät menneet aina yhteen, joten luotiin uusi taulu postinumeroita ja kaupunkeja varten. Taulussa pääavaimena on postinumero ja se tulee vierasavaimeksi smartpost tauluun. Täten saatiin myös vaaditut 7 taulua tietokantaan, joiden keksiminen alkuun oli erittäin vaikeaa. Smartpostille lisättiin myös attribuutiksi boolean, että onko se luotu kartalle vai ei. Asia tuli esille, kun piti saada selville, mitkä automaatit on luotu ja mitkä ei. Luoduille paketeille lisättiin määränpää ja lähtöpaikka attribuutti. Tämä huomattiin, kun pakettia luodessa piti sille jotenkin saada talteen nämä paikat ja myöhemmin lisättiin vielä matka attribuutti, koska tuntui hyvältä idealta.

Toteutus:

Toteutuksessa käytettiin javafx graafista käyttöliittymää, jolla pystytään suorittamaan kaikki toiminnot käyttäjäystävällisesti ja vaatimatta tietoa SQL:stä. Käyttöliittymään ei tarvinnut tehdä mitään sellaisia muutoksia joita tietokanta erityisesti vaatisi. Kaikki muutokset olivat vain käyttöliittymän keskeisiä asioita

Testatessa huomioitavaa:

Moneen otteeseen tuli paljon virheitä kuten mm. unique key constraint. Kaikki nämä selvisivät lukemalla omaa koodiaan vähän tarkemmin ja alkoi syöttää SQL komennoille oikeita arvoja.

Ohjelmassa huomioitavaa:

Tietokantaan itseensä on laitettu eheyssääntöihin liittyviä muutamia CHECK lauseita varmistukseksi tarkistamaan, että painot ja koot ovat suurempia kuin nolla ja vaadittuihin arvoihin on laitettu NOT NULL. Kaikki on kuitenkin tarkistettu jo erikseen käyttöliittymän puolella ja ohjelma antaa viestiä jos esim. koittaa tehdä esinettä ilman painoa. Tämä on sen takia, että käyttäjäkokemus olisi parempi ja informatiivisempi ja ohjelma ei kaatuisi SQL erroreihin.