

**Tietokantasovellus**

**Drinkkiarkisto**

Arto Kiviluoto

013748799

[arto.kiviluoto@cs.helsinki.fi](mailto:arto.kiviluoto@cs.helsinki.fi)



# 1.Johdanto

Harjoitustyöni aiheena on melko yksinkertainen drinkkiarkisto, joka tarjoaa käyttäjille mahdollisuuden hakea tietokannasta erilaisia juomasekoituksia, sekä vaikuttaa drinkkiarkiston sisältöön. Tarkoituksena on tuottaa selainpohjainen sovellus, jota käyttäjät pystyvät käyttämään kotikoneeltaan. Ajatuksena järjestelmän takana on tarjota drinkkien ystäville kanava, jossa jakaa kokemuksia ja löytää uusia ideoita. Järjestelmä on tarkoitettu rekisteröityneille käyttäjille, eikä sisältöä näytetä palvelun ulkopuolisille tahoille.

Harjoitustyön toteutan käyttäen ohjelmointikielenä PHP:tä ja tietokantana PostgreSQL:ää. PHP:n valitsin siksi, että se on melko yksinkertainen ja sen opettelu harjoitustyötä tehdessä onnistuu varmasti. PHP on entuudestaan jossain määrin tuttu, mutta tarkoitukseni on syventää osaamistani sen suhteen.

PostgreSQL on valittu harjoitustyöhön lähinnä siksi, ettei tietokantojen eroista ole kovinkaan tarkkaa käsitystä ja PostgreSQL oli tässä tapauksessa kurssin järjestäjien suositus.

Versionhallintaan käytetään Github -verkkopalvelua, jolloin harjoitustyö on kaikille osapuolille saatavilla sijainnista ja ajasta huolimatta.

Järjestelmä tarjotaan Helsingin yliopiston Tietojenkäsittelytieteen laitoksen Users -palvelimelta. Käyttäjien näkökulmasta järjestelmä ei vaadi mitään ylimääräistä, vaan sen tulisi toimia ”tavallisen” verkkosivuston tapaan. Eroja selaimien käyttäytymisessä esimerkiksi ulkoasun suhteen saattaa löytyä.

Työ on testattu vain PostgreSQL:llä, mutta siinä käytetyt SQL-komennot ovat melko yleisiä, joten tietokannan vaihtaminen on melko vaivatonta.

## 2. Yleiskuva järjestelmästä



### 2.1 Käyttötapauskaavio

### Käyttäjärühmät

**Vierailija** – Vierailija on henkilö, joka ei ole vielä käyttänyt järjestelmää. Vierailija rekisteröityy järjestelmään alkaakseen käyttää sitä

**Käyttäjä** – Käyttäjä on järjestelmää ulkopuolelta (kotikone, julkinen tietokone etc..) käyttävä henkilö, joka kirjautuu järjestelmään selaillakseen arkistosta löytyviä drinkkejä, sekä vaikuttaakseen järjestelmän sisältöön (omien reseptien lisäys/muokkaus/poisto).

**Ylläpitäjä** – Ylläpitäjä on järjestelmän tuottanut taho, joka pitää yllä järjestelmää sen kokonaisuudessaan.

### 3. Käyttötapaukset

#### Vierailija

Vierailija siirtyy selaimella Drinkkiarkisto -sivulle ja ohjautuu etusivulle. Etusivulla Vierailijalle kerrotaan sivuston tarkoituksesta ja hänelle tarjotaan mahdollisuus rekisteröityä.

#### Käyttäjä

Käyttäjä siirtyy selaimella Drinkkiarkisto -sivulle ja kirjautuu tunnuksillaan sisään syöttämällä käyttäjänimen ja salasanan. Hän siirtyy haku -toimintoon ja hakee drinkkivaihtoehtoja raaka-aineen perusteella. Raaka-aineeksi käyttäjä valitsee appelsiinimehun ja hänelle listataan drinkit jotka sisältää appelsiinimehua.

Käyttäjä on kirjautuneena palveluun ja haluasi tehdä Long Island Iced Tean. Käyttäjä muistaa drinkin nimen ja yhden raaka-aineista, mutta haut eivät tuota suotuisia tuloksia. Drinkkiä ei siis ole järjestelmässä. Käyttäjä löytää reseptin toisaalta ja lisää sen järjestelmään muiden nähtäväksi.

Käyttäjä huomaa, että äsken lisäämänsä Long Island Iced Tea:han tuli virhe. Hän hakee reseptin arkistosta, valitsee sen ja painaa 'Muokkaa reseptiä' -painiketta. Käyttäjä tekee haluamansa muutokset ja tallentaa reseptin.

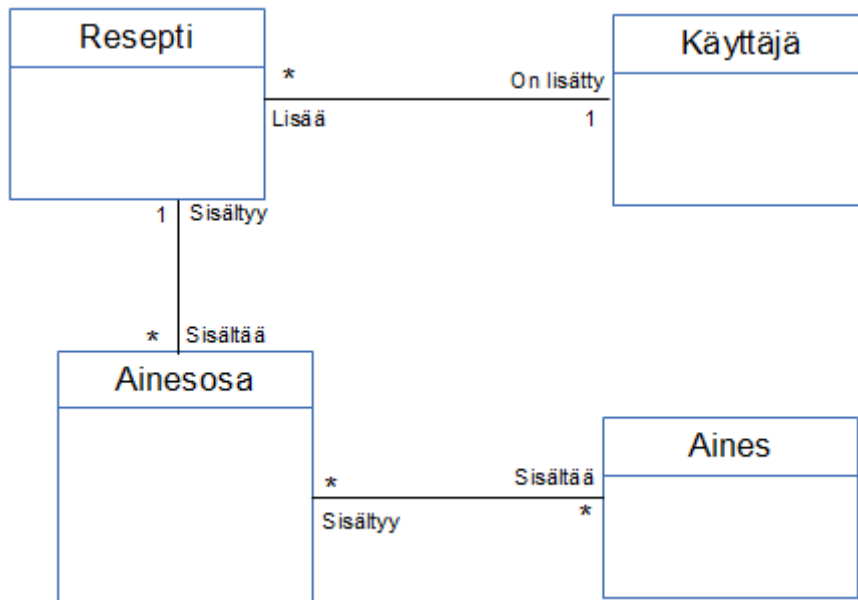
#### Ylläpitäjä

Ylläpitäjä huomaa, että Käyttäjän lisäämä resepti on epäkelpo järjestelmään ja poistaa sen.

Ylläpitäjä huomaa virheen Käyttäjän lisäämässä reseptissä ja muokkaa sitä haluamallaan tavalla.

## 4. Järjestelmän tietosisältö

Järjestelmä tarvitsee toimiakseen neljä tietokantataulua. Käyttäjä, mihin tallennetaan järjestelmään rekisteröityneiden henkilöiden antamat tiedot. Resepti eri reseptien tietojen tallentamiseen. Aines reseptissä käytettävien aineksien tallennukseen sekä välitauluksi Ainesosa, jonka avulla reseptiin kuuluvat ainesosat ja niiden määrät saadaan talletettua.



4.1 Käsitekaavio

## Resepti

### Resepti

Attribuutti	Arvojoukko	Kuvaus
id	Serial	Reseptin tunnus
Nimi	Merkkijono, 20	Juoman nimi
Ohje	Merkkijono, 100	Valmistusohje
Lisähuomio	Merkkijono, 100	Mahdolliset huomiot
Hyväksytty	Boolean	Onko resepti julkinen
Lisääjä	Merkkijono, 20	Kenen lisäämä resepti on

#### 4.2 Reseptin tietosisältö

Reseptillä on aina yksilöivä tunnistenumero. Jokaisella reseptillä on oltava myös nimi, ohje, ja lisääjä. Lisähuomio on vapaaehtoinen kenttä, sillä jokaiseen reseptiin ei välttämättä sellaista tarvita. Hyväksytty -kenttä on tarkoitettu järjestelmän jatkokehitysidealle, josta myöhemmässä osiossa lisää

## Käyttäjä

### Käyttäjä

Attribuutti	Arvojoukko	Kuvaus
ktunnus	Merkkijono, 15	Käyttäjätunnus
salasana	Merkkijono, 16	Käyttäjän salasana
Pro -status	Boolean	Onko status myönnetty

#### 4.3 Käyttäjän tietosisältö

Käyttäjä yksilöidään käyttäjätunnuksen perusteella. Samoja käyttäjätunnuksia ei voi olla. Jotta käyttäjä voidaan tunnistaa on jokaisella oltava salasana. Pro -status on tarkoitettu jatkokehitysideaa varten, josta myöhemmässä osiossa lisää.

## Aines

### Aines

Attribuutti	Arvojoukko	Kuvaus
aines	Merkkijono, 20	Aineksen nimi

#### 4.4 Aineksen tietosisältö

Aines -taulu sisältää vain tiedon siitä minkä nimisiä aineita järjestelmään on lisätty. Samannimiset ainekset eivät ole mahdollisia.

## Ainesosa

### Ainesosa

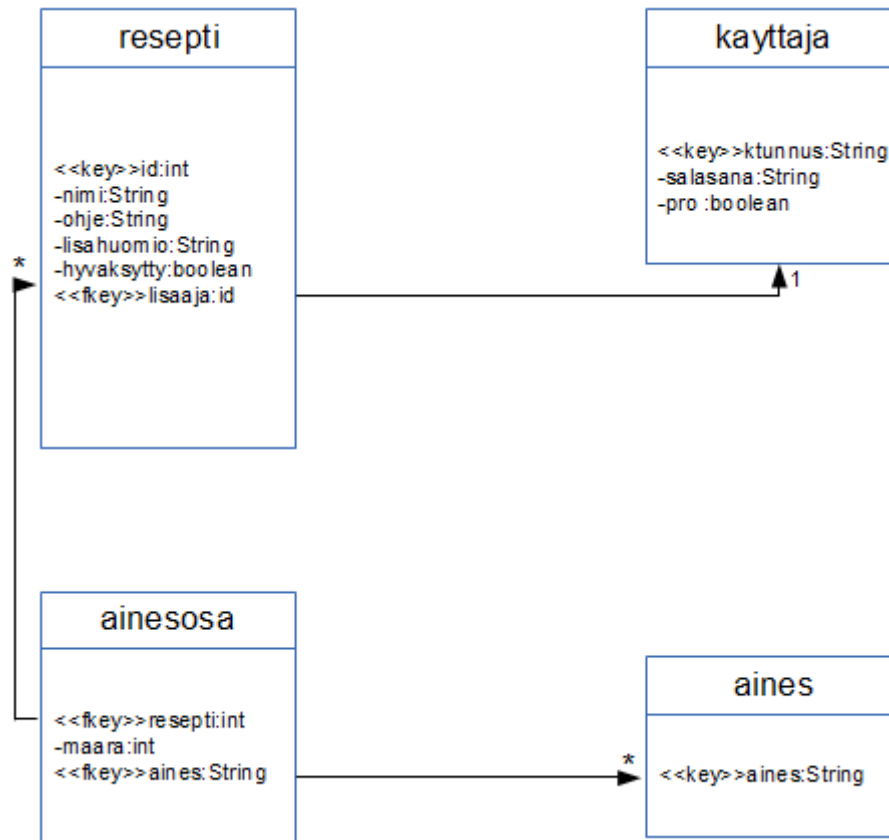
Attribuutti	Arvojoukko	Kuvaus
resepti	int	Reseptin ID
maara	int	Aineksen määrä
aines	Merkkijono,20	Aineksen nimi

#### 4.5 Ainesosan tietosisältö

Ainesosa on aputaulu, jonka avulla reseptiin kuuluvien aineiden määrä saadaan talletettua. 'resepti' -vierasavain viittaa reseptin tunnisteeseen, 'aines' viittaa aines -taulun yhteen riviin ja määrä kertoo kuinka paljon kyseistä ainesta reseptiin kuuluu.



## 5. Relaatietietokantakaavio



5.1 Relaatietietokantakaavio

### Create table -lauseet:

```
CREATE TABLE kayttaja (  
    ktunnus char(15) PRIMARY KEY NOT NULL,  
    salasana char(16) NOT NULL,  
    pro boolean  
);
```

```
CREATE TABLE resepti (  
    id SERIAL PRIMARY KEY,  
    nimi char(20) NOT NULL,  
    ohje char(100) NOT NULL,  
    lisahuomio char(100),  
    hyväksytty boolean,  
    lisaaja char(20) REFERENCES kayttaja(ktunnus) NOT NULL  
);
```

```
CREATE TABLE aines (  
    aines char(20) PRIMARY KEY NOT NULL  
);
```

```
CREATE TABLE ainesosa (  
    resepti int REFERENCES resepti(id) NOT NULL,  
    maara int NOT NULL,  
    aines char(20) REFERENCES aines(aines)  
);
```

## 6. Järjestelmän yleisrakenne

Harjoitustyössä on pyritty kolmijakoiseen MVC -malliin, jossa mallit, näkymät ja kontrollerit ovat omissa hakemistoissaan. Käytännössä tämä jako toteutuu järjestelmässä seuraavalla tavalla:

Views -kansion alta löytyvät sekä koko sivun kattava runko tietostosta pohja.php, että erilaiset sisältönäkymä. Käytännössä tämä runko tarjoaa muuttuvalle sisällölle navigointipalkin, sivun jaon ja ulkoasun header -palkkeineen. Libs -kansioista löytyvät erilaiset kooditiedostot, jotka hallinnoivat

järjestelmän tietosisältöä käyttäjän syötteiden mukaisesti. Juuresta löytyvät