

Aihe: Verkkopankkisovellus

Sovelluksen kuvaus

Toteutetaan sovellus, jossa käyttäjä pääsee tarkastelemaan omia pankkitietojaan selaimen kautta.

Tavoitteena on, että käyttäjä pystyy kirjautumaan omalla tunnuksellaan järjestelmään, jossa hän voi tarkastella tiliensä saldoja ja niiden tilitapahtumia, lähettää rahaa toiselle tilille sekä päivittää omia henkilökohtaisia tietojaan. Ylläpitäjän tulee pystyä myös lisäämään uusia käyttäjiä. Ajan salliessa ohjelmaan voidaan toteuttaa myös viestiominaisuus, jonka avulla käyttäjät pystyvät lähettämään toisilleen yksityisiä viestejä, jotka ovat luettavissa kirjauduttua.

Sovellus toteutetaan Javalla ja PostgreSQL:llä. Se toimii laitoksen users-palvelimella Tomcat-palvelimen alla.

Käsiteanalyysi

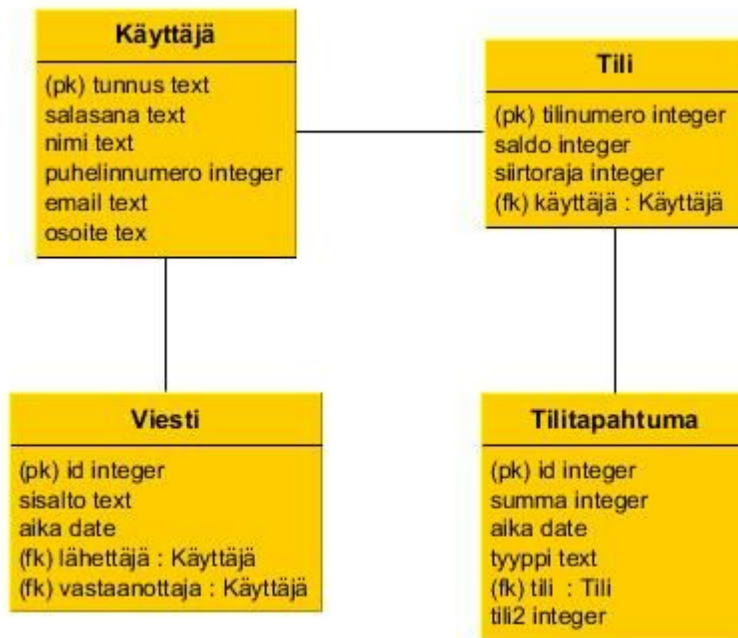
Aihemäärittelystä löytyi seuraavat käsitteet:

Sovellus	// tätä toteutetaan
Käyttäjä	
Pankkitiedot	// käyttäjän tietoja, eli attribuutteja
Selain	// apuna sovelluksen käyttämiseen
Tunnus	// käyttäjän ominaisuus
Järjestelmä	// sama kuin sovellus
Tili	
Saldo	// tiin ominaisuus
Tilitapahtumat	
Raha	// määre, joka liittyy tiliin ja tilitapahtumiin
Henkilökohtaiset tiedot	// käyttäjän ominaisuuksia
Java	// ohjelmalogiikan toteutuskieli
Viesti	

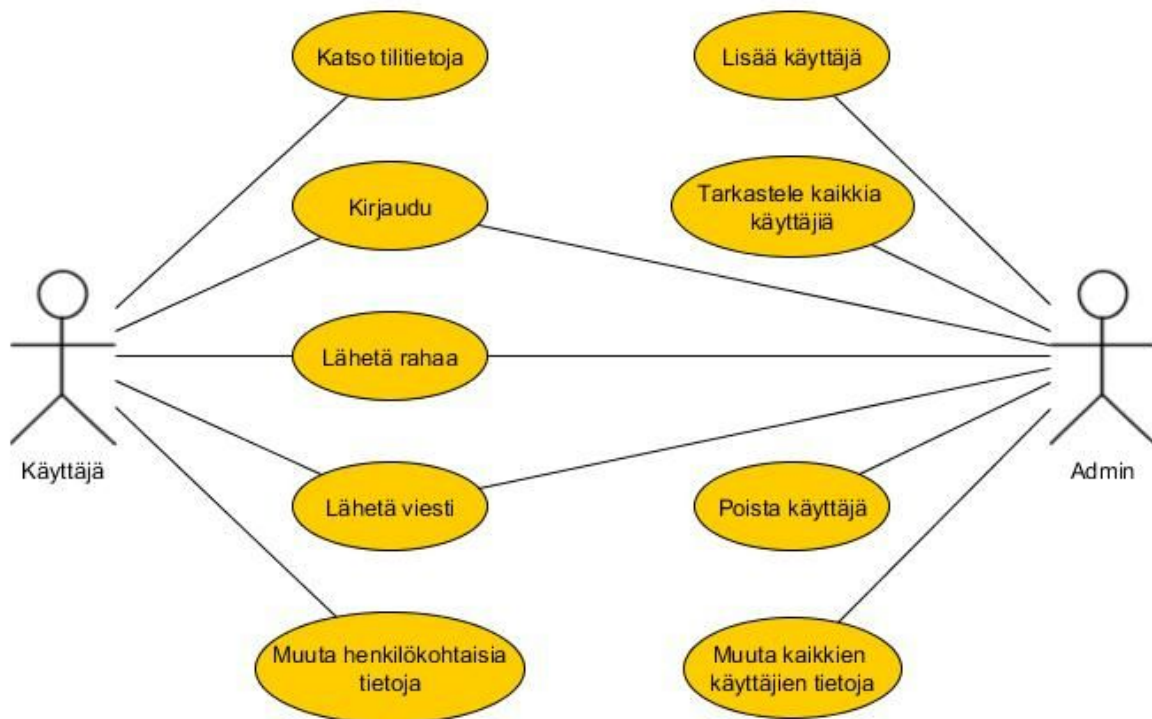
Tietokantatauluiksi kelpaaviksi osoittautui siis:

Käyttäjä
Tili
Tilitapahtumat
Viesti

Tietokantakaavio



Käyttötapauskaavio



Esimerkkikyselyitä

Tietokantataulujen luomiskyselyt:

```
CREATE TABLE Kayttaja (  
  tunnus text PRIMARY KEY,  
  nimi text NOT NULL,  
  puhnro text,  
  email text,  
  salasana text NOT NULL  
);
```

```
CREATE TABLE Tili (  
  tilinumero integer PRIMARY KEY,  
  saldo integer NOT NULL,  
  nostoraja integer,  
  siirtoraja integer,  
  kayttaja text NOT NULL,  
  FOREIGN KEY (kayttaja) REFERENCES Kayttaja (tunnus)  
);
```

```
CREATE TABLE Tilitapahtuma (  
  id integer PRIMARY KEY,  
  summa integer NOT NULL,  
  ajankohta date NOT NULL,  
  tyyppi text NOT NULL,  
  tili integer NOT NULL,  
  tili2 integer,  
  FOREIGN KEY (tili) REFERENCES Tili (tilinumero)  
);
```

```
CREATE TABLE Viesti (  
  id integer PRIMARY KEY,  
  sisalto text NOT NULL,  
  aika date NOT NULL,  
  lahettaja text NOT NULL,  
  vastaanottaja text NOT NULL,  
  FOREIGN KEY (lahettaja) REFERENCES Kayttaja (tunnus),  
  FOREIGN KEY (vastaanottaja) REFERENCES Kayttaja (tunnus)  
);
```

HUOM!

-Saldo tallennetaan tileille sentteinä.

-Tilitapahtuman tyytit ovat:

Kredit (saldoa lisätään)
Debet (saldoa vähennetään)
Nosto
Talletus
Korko

Lisäyskyselyt:

```
INSERT INTO Kayttaja (tunnus, nimi, puhnro, email, salasana)
VALUES ('locx', 'Oskari Johansson', '12345', 'keksitty.email@nan.com', 'banaani1');
```

```
INSERT INTO Tili (saldo, nostoraja, kayttaja) VALUES (0, 100000, 1);
```

```
INSERT INTO Tilitapahtuma (summa, ajankohta, tyyppi, tili, tili2)
VALUES (100, datetime(), 'Kredit', 1, 10);
```

```
INSERT INTO Viesti (sisalto, aika, lahettaja, vastaanottaja) VALUES ('Hello World!',
datetime(), 'locx', 'maailma');
```

Tililtä 1 siirretään 100.00 euroa tilille 10:

```
BEGIN TRANSACTION;
UPDATE Tili SET saldo = saldo - 10000 WHERE tilinumero = 1;
INSERT INTO Tilitapahtuma (summa, ajankohta, tyyppi, tili, tili2) VALUES (10000,
datetime(), 'Debet', 1, 10);
UPDATE Tili SET saldo = saldo + 10000 WHERE tilinumero = 10;
INSERT INTO Tilitapahtuma (summa, ajankohta, tyyppi, tili, tili2) VALUES (10000,
datetime(), 'Kredit', 10, 1);
COMMIT;
```

Kysely, jolla saadaan listattua kaikki tilin 1 tilitapahtumat:

```
SELECT * FROM Tilitapahtumat
WHERE tili = 1
ORDER BY ajankohta DESC;
```

Kysely, jolla saadaan listattua käyttäjien locx ja maailma välinen keskusteluhistoria:

```
SELECT * FROM Viesti
WHERE (lahettaja = 'locx' AND vastaanottaja = 'maailma')
OR (lahettaja = 'maailma' AND vastaanottaja = 'locx');
```