

Ohjelmistokehityksen perusteet harjoitustyö

Janne Vierimaa TVT19KMO

GitHUB

<https://github.com/JanneVierimaa/projekti>

<http://www.students.oamk.fi/~t8vija01/projekti/>

## Aihe

Työni aihe on verkkokauppa, joka toimii siten, että tuote-sivun kautta voi selata erilaisia kaupasta saatavilla olevia tuotteita. Tuotteet on jaoteltu kolmeen eri kategoriaan: televisiot, puhelimet ja tietokoneet. Tuote-sivulla kategorian pudotusvalikosta valitsemalla saadaan näkyville vain valitun kategorian tuotelistaus.

Tuotteen tai useampia tuotteita voi lisätä ostoskoriin, joka kokoaa ja summaa yhteen valitut tuotteet, niiden kappalemäärät ja hinnat. Ostoskori-sivulta voi käydä katsomassa valitsemansa tuotteet, niiden yhteishinnan ja tilata tuotteet tai palata takaisin tekemään ostoksia.

Tuotteiden lisääminen ostoskoriin, ostoskorin katseleminen ja tuotteiden tilaaminen vaativat sen, että käyttäjä rekisteröi tunnuksen ja salasanan joilla ensin kirjautuu sisään. Pelkkä tuotelistauksen selaaminen onnistuu myös ilman tunnusta/kirjautumista.

Kirjautunut käyttäjä voi katsoa tilaamiaan tuotteita omat tilaukset sivulta.

## Käyttöliittymän kuvaus

Sovelluksen aloitussivu on seuraavanlainen:

VERKKOKAUPPA				kirjaudu sisään
etusivu	tuotteet	rekisteröidy	omat tilaukset	ostoskori

Tervetuloa Vieras!

Rekisteröitymällä ja Kirjautumalla sisään pääset tekemään ostoksia

[Rekisteröidy tästä](#)

Jos sinulla on jo tunnukset, voit kirjautua sisään tästä:

<input type="text"/>
<input type="password"/>
<input type="button" value="Kirjaudu"/>

Aloitussivu tarjoaa mahdollisuuden rekisteröitymiseen tai sisäänkirjautumiseen. Samat toiminnot löytyvät myös sivun yläreunasta, ”rekisteröidy” ja ”kirjaudu sisään” linkkien takaa.

tuotteet-linkin takaa avautuu tuotelistaus, omat tilaukset näyttää kirjautuneen käyttäjän omat tiedot ja tilatut tuotteet. Ostoskori sivu avautuu myöskin vain kirjautuneelle käyttäjälle ja näyttää ostoskoriin lisätyt tuotteet.

Rekisteröidy-sivu näyttää seuraavalta:

VERKKOKAUPPA				kirjaudu sisään
etusivu	tuotteet	rekisteröidy	omat tilaukset	ostoskori

**Rekisteröidy**

Rekisteröinti lisää tietokannan Asiakas-tauluun asiakkaan etunimen, sukunimen, osoitteen, käyttäjänimen ja salasanan. Salasana on kryptattuna. Kaikki tiedot ovat pakollisia syöttää. Käyttäjänimi on yksilöllinen tieto ja kahta samanlaista käyttäjänimeä ei sallita. Mikäli rekisteröinti epäonnistuu, sovellus antaa virheilmoituksen: "Antamasi käyttäjänimi on jo varattu." Mikäli käyttäjä on jo kirjautunut sisään sovellukseen, rekisteröinti-sivu ilmoittaa: "Olet jo kirjautunut sisään, tunnuksella nnn"

Onnistuneen rekisteröitymisen jälkeen sovellus siirtyy automaattisesti sisäänkirjautumiseen:

### Kirjaudu sisään

Kirjaudu

Sisäänkirjautumisessa sovellus kysyy käyttäjätunnusta ja salasanaa, jotka tarkistetaan tietokannasta Asiakas-taulusta. Jos vastaavuus löytyy, tulostetaan käyttäjälle ilmoitus ”olet kirjautunut sisään” ja siirrytään takaisin etusivulle. Kirjautuminen ja käyttäjänimi tallennetaan \$SESSION-muuttujiin, ja niitä voidaan käyttää kaikkialla sovelluksessa kirjautumissession ajan.

Kirjautumisen jälkeen etusivu näyttää tältä:

### Tervetuloa Testikäyttäjä

Voit nyt lisätä tuotteita ostoskoriin ja tilata niitä. Klikkaa tuotelistausta ja aloita shoppailu!

Etusivulle tulostuu Tervetulotoivotus ja kirjautumissession käyttäjänimi ja kehoitus siirtyä tuotesivulle.

Tämä on toteutettu php:n if-lauseella, jossa tarkistetaan onko käyttäjä kirjautunut vai ei. Samalla tekniikalla myös oikean yläkulman linkin teksti vaihtuu ”kirjaudu ulos” – muotoon ja myös linkki ohjaa uloskirjautumiseen.

Tuotesivu listaa kaikki saatavilla olevat tuotteet taulukkomuotoon:









VERKKOKAUPPA

[kirjaudu ulos](#)

[etusivu](#) [tuotteet](#) [rekisteröidy](#) [omat tilaukset](#) [ostoskori](#)

Näytä: [Kaikki tuotteet](#) [näytä](#)

### Tuotteet

<div>Television</div> <div>The best TV in the world</div> <div></div> <div>499.00 €</div> <div>Määrä: <input type="text"/></div> <div><a href="#">Lisää ostoskoriin</a></div>	<div>Phone</div> <div>The best smartphone in the world</div> <div></div> <div>999.00 €</div> <div>Määrä: <input type="text"/></div> <div><a href="#">Lisää ostoskoriin</a></div>	<div>Laptop</div> <div>The best laptop in the world</div> <div></div> <div>689.00 €</div> <div>Määrä: <input type="text"/></div> <div><a href="#">Lisää ostoskoriin</a></div>	<div>LG TV</div> <div>Super good TV</div> <div></div> <div>789.00 €</div> <div>Määrä: <input type="text"/></div> <div><a href="#">Lisää ostoskoriin</a></div>
<div>Samsung TV</div> <div>HyperSuper good TV</div> <div></div>	<div>Sony TV</div> <div>Megahyper good TV</div> <div></div>	<div>iPhone</div> <div>Overpriced and mediocre phone</div> <div></div>	<div>Oneplus</div> <div>Flagship killer smartphone</div> <div></div>

Pudotusvalikosta käyttäjä voi valita haluamansa kategorian tai kaikki tuotteet. Sovellus hakee tietokannasta valitun kategorian mukaiset tuotteet. Tuotetiedoista näkyvillä on tuotteen nimi, tuotekuvaus, kuva ja hinta. Määrä-kenttään käyttäjä voi määrittää haluamansa määrän tuotetta ja lisätä ne ostoskoriin. Kun tuote lisätään ostoskoriin, sovellus ilmoittaa lisäyksen onnistuneen ja palaa takaisin tuotesivulle.

Ostoskori-sivu näyttää kaikki kirjautuneen käyttäjän ostoskoriin lisäämät tuotteet:

VERKKOKAUPPA

kirjaudu ulos

etusivu

tuotteet

rekisteröidy

omat tilaukset

ostoskori

**Ostoskorisi sisältö**

- tuotenumeroa: 6  
Phone  
4 kpl  
999.00€ / kpl  
  
Kokonaishinta: 3996 €
- tuotenumeroa: 9  
Samsung TV  
5 kpl  
899.00€ / kpl  
  
Kokonaishinta: 4495 €

**KAIKKI YHTEENSÄ: 8491 €**

Tilaa

Jatka shoppailua

Ostoskori-sivu hakee tietokannasta Ostoskori ja Tuote – tauluista kirjautuneen käyttäjän ostoskoriin lisäämät tuotteet. Tuotteet näytetään tuotenumeron mukaan lajiteltuina kappalemäärittäin. Tuotteiden yhteishinnat ja lopullinen kokonaishinta on laskettu yhteen PHP-koodissa.

Tilaa-nappi tyhjentää ostoskorin kyseisen kirjautuneen käyttäjän osalta ja lisää tilauksen tuotteet Triggerin avulla ennen tyhjennystä tietokannan Tilaukset-tauluun. Jatka shoppailua-nappi palaa takaisin tuotteet-sivulle.

Omat tilaukset-sivu tulostaa näkyviin kirjautuneen käyttäjän tilaamat tuotteet:

VERKKOKAUPPA				kirjaudu ulos
etusivu	tuotteet	rekisteröidy	omat tilaukset	ostoskori

#### Omat tietosi:

TestiEtunimi  
TestiSukunimi  
TestiKuja 1  
Testikäyttäjä

#### Tähän mennessä tilaamasi tuotteet

2019-10-12 Tuotenro: 6 kpl: 4  
2019-10-12 Tuotenro: 9 kpl: 5  
2019-10-12 Tuotenro: 9 kpl: 5  
2019-10-12 Tuotenro: 6 kpl: 5  
2019-10-12 Tuotenro: 5 kpl: 5  
2019-10-12 Tuotenro: 16 kpl: 5  
2019-10-12 Tuotenro: 11 kpl: 5

Omat tilaukset-sivu hakee tietokannasta Asiakas-tilusta kirjautuneen käyttäjän omat tiedot ja tulostaa ne ruudulle salasanaa lukuunottamatta. Sivun hakee myös tietokannasta Tilaukset-tilusta kirjautuneen session käyttäjänimen perusteella tilauksen tiedot ja päivämäärän.

Kirjaudu ulos – linkistä sessio tuhoetaan ja käyttäjä kirjataan näin ulos. Käyttäjälle tulostetaan viesti: ”Olet nyt kirjautunut ulos”

## Php-koodi ja tietokanta

### Php

#### menu.php

Navigointipalkki yläreunassa. Sisältää linkit muihin sivuihin. Sijoittelu määritetty niin, että palkki jää näkyviin kun sivua ”scrollataan.” Session aloitus myös täällä. Kirjaudu sisään/ulos-linkki muokkautuu sen mukaan onko käyttäjä kirjautunut sisään.

#### connection.php

Muodostaa yhteyden mysli.oamk.fi-palvelimelle ja opisk\_t8vija01 tietokantaan tunnuksella: t8vija01 ja salasanalla: amorphis. Mikäli yhteyttä ei saada aikaan tulostuu virheilmoitus.

#### index.php

Aloitussivu, joka tulostaa käyttäjälle tervetuloilmoituksen. If-lauseella toteutettu tarkistus, onko käyttäjä kirjautunut vai ei. Jos käyttäjä on kirjautunut, näytetään erilainen viesti kuin kirjautumattomalle käyttäjälle. Kirjautumattomalle käyttäjälle näytetään rekisteröintilinkki ja myös kirjautumislomake.

#### [create\\_user.php](#)

Rekisteröimislomake, jossa kentät: Etunimi, Sukunimi, Osoite, username, password. Lomake välittää tiedot Post-metodilla Rekisteröidy-nappia painamalla [add\\_user.php](#) – sivulle. Kaikki kentät ovat pakollisia täyttää.

#### [add\\_user.php](#)

Lisää [create\\_user.php](#) lomakkeelta saadut tiedot tietokannan Asiakas-tauluun. Onnistuneen rekisteröinnin jälkeen sovellus siirtää käyttäjän [login\\_form.php](#) – sivulle. Muussa tapauksessa annetaan virheilmoitus ja palataan [create\\_user.php](#) sivulle.

#### [login\\_form.php](#)

Tarkistaa ensin, onko käyttäjä jo kirjautunut sisään. Jos ei, sovellus tulostaa kirjautumislomakkeen, jossa kysytään käyttäjätunnusta ja salasanaa. Annetut tiedot välitetään post-metodilla [login.php](#) – sivulle.

#### [login.php](#)

Post-metodilla saadut tiedot (käyttäjänimi ja salasana) tallennetaan muuttujiin ja suoritetaan kysely tietokannasta, jolla tarkistetaan löytyykö kyseinen käyttäjänimi ja salasana yhdistelmä Asiakas-taulusta. Mikäli löytyy, asetetaan sessio sisäänkirjautuneeksi kyseisellä käyttäjänimellä. Tämän jälkeen palataan etusivulle.

#### [tuote.php](#)

Tulostaa tietokantahaun avulla kaikki saatavilla olevat tuotteet. Tulostus tehdään riveittäin ja niin, että yhdellä rivillä on korkeintaan neljä tuotetta. Pudotusvalikolla voidaan valita vain yksi haluttu kategoria, jolloin tietokannasta haetaan vain kyseisen kategorian tuotteet. Valinta tarkistetaan if-lauseella ja "\$selected"-muuttujan sisällön avulla. Pudotusvalikon tehty valinta jää voimaan myös muuttujan "\$selected" avulla. Kirjautuneelle käyttäjälle näytetään myös "määrä"-kenttä ja "lisää ostoskoriin" – nappi, joilla tuotteen lisääminen ostoskoriin onnistuu. Lomake välittää tiedot ( idTuote, määrä, username) post-metodilla [add\\_ostoskori.php](#)-sivulle.

#### [add\\_ostoskori.php](#)

Post-metodilla saadut tiedot talletetaan muuttujiin ja suoritetaan insert-lause, jolla tietokannan Ostoskori-tauluun lisätään tiedot (idTuote, määrä, username). Sovellus tulostaa viestin onnistuneesta/epäonnistuneesta lisäyksestä, jonka jälkeen palaa takaisin tuotesivulle.

#### [ostoskori.php](#)

Ostoskorin sisältö näkyy vain kirjautuneelle käyttäjälle. Kirjautuminen tarkistetaan if-lauseella.

Tietokannasta haetaan Ostoskori ja Tuote – tauluista kirjautuneen käyttäjän ostoskoriin lisäämät tuotteet. Tuotteet näytetään tuotenumeron mukaan lajiteltuina kappalemäärittäin. Tuotteiden yhteishinnat ja lopullinen kokonaishinta on laskettu yhteen PHP-koodissa.

Tilaa-nappi välittää tiedot [order.php](#) sivulle, joka tyhjentää ostoskorin kyseisen kirjautuneen käyttäjän osalta ja lisää tilauksen tuotteet Triggerin avulla ennen tyhjennystä tietokannan Tilaukset-tauluun. Jatka shoppailua-nappi palaa takaisin tuotteet-sivulle.

#### [order.php](#)

Kuten edellä kuvattu, tyhjentää Ostoskori-taulusta tiedot kyseisen kirjautuneen käyttäjän osalta. Tämän jälkeen sovellus palaa ostoskori-sivulle.

#### [tilaukset.php](#)

tilaukset-sivu hakee tietokannasta Asiakas-taulusta kirjautuneen käyttäjän omat tiedot ja tulostaa ne ruudulle salasanaa lukuunottamatta. Sivu hakee myös tietokannasta Tilaukset-taulusta kirjautuneen session käyttäjänimen perusteella tilauksen tiedot ja päivämäärän. Nämä tulostetaan riveittäin.



logout.php

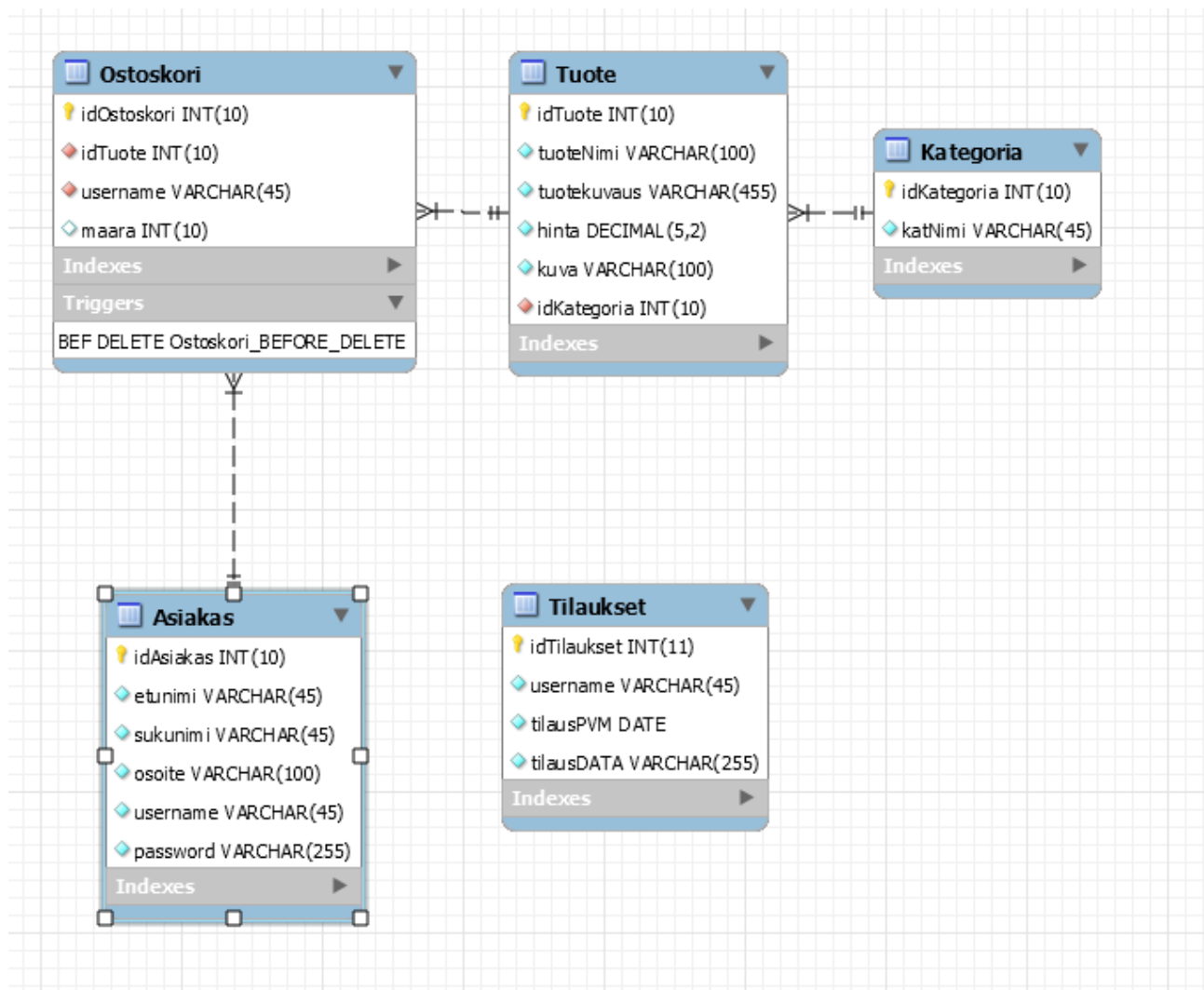
tuhoaa session ja käyttäjä kirjataan näin ulos

## Tietokanta

Työssä käytetty tietokanta "opisk\_t8vija01" sijaitsee mysli.oamk.fi – palvelimella.

Tietokannassa on viisi eri taulua: Kategoria, Tuote, Ostoskori, Asiakas ja Tilaukset

Tilaukset tauluun ei ole relaatioita, vaan siihen tietoja lisätään Ostoskori-taulun Before DELETE Triggerin avulla.



Tuote kuuluu kategoriaan, ja idKategoria on viiteavaimena Tuotetaulussa. Tuotteella on nimi, kuvaus, hinta ja polku kuvatiedostoon tuotteesta.

Ostoskori-taulu toimii Tuotteen ja Asiakas-taulun välissä ja sisältää viiteavaimet molemmista tauluista. Tuotteiden määrä on myös Ostoskori-taulussa. T

Asiakastaulussa username on yksilöllinen tieto, ja siksi sitä myös käytetään viiteavaimena Ostoskori-taulussa. Tämän lisäksi asiakastaulu sisältää tiedot: etunimi, sukunimi, osoite, salasana.

Tilaukset taulu on erillisenä muista ja se sisältää käyttäjänimen, päivämäärän ja CONCAT-toiminnon avulla kootun TilausDATAN. Tilaus-tauluun lisätään tietoja Triggerin avulla, ennenkuin ostoskorista poistetaan tietoa. TilausDATA sisältää Tuotenumeron ja kappalemäärän tekstimuotoisena datana. Nämä saadaan idTuote ja määrä tiedoista.

## SQL-kyselyt

Rekisteröinnissä tiedot lisätään Asiakas-tauluun (add\_user.php)

INSERT INTO Asiakas (etunimi,sukunimi,osoite,username,password) VALUES (:etunimi, :sukunimi,:osoite, :username, :password)

```
$encrypted_pass = password_hash($_POST['password'],PASSWORD_DEFAULT);
$stmt=$db->prepare("INSERT INTO Asiakas (etunimi,sukunimi,osoite,username,password) VALUES (:etunimi, :sukunimi,:osoite, :username, :password)");
$stmt->bindParam(':etunimi',$_POST['etunimi']);
$stmt->bindParam(':sukunimi',$_POST['sukunimi']);
$stmt->bindParam(':osoite',$_POST['osoite']);
$stmt->bindParam(':username',$_POST['username']);
$stmt->bindParam(':password',$encrypted_pass);
if($stmt->execute()){
```

Kirjautumisessa tarkistetaan löytyykö session käyttäjänimellä annettu salasana Asiakas-taulusta (login.php)

SELECT password from Asiakas where username=:username

```
$stmt=$db->prepare("SELECT password from Asiakas where username=:username");
$stmt->bindParam(':username', $annettu_tunnus);
$stmt->execute();
$oikea_salasana = $stmt->fetch(PDO::FETCH_COLUMN);
if(password_verify($annettu_salasana,$oikea_salasana)) {
```

Tuote-taulusta tehdään kyselyjä ja JOINin avulla otetaan mukaan myös Katategoria-taulu. Näin saadaan tulostettua joko kaikki tuotteet tai vain tietyn kategorian tuotteet. (tuote.php)

SELECT \* FROM Tuote

SELECT \* FROM Tuote JOIN Katategoria ON Tuote.idKatategoria=Katategoria.idKatategoria WHERE Katategoria.katNimi='\$selected'

```
if($selected==""){
    $sql="SELECT * FROM Tuote";
}
else{
    $sql="SELECT * FROM Tuote JOIN Katategoria ON Tuote.idKatategoria=Katategoria.idKatategoria WHERE Katategoria.katNimi='$selected'";
}
```

Ostoskori-tauluun lisätään tuotteita (add\_ostoskori.php)

INSERT INTO Ostoskori(idTuote,username,maara) VALUES (:idTuote,:username,:maara)

```
$stmt=$db->prepare("INSERT INTO Ostoskori(idTuote,username,maara) VALUES (:idTuote,:username,:maara)");
$stmt->bindParam(':idTuote',$id_products);
$stmt->bindParam(':username',$username);
$stmt->bindParam(':maara',$amount);
if($stmt->execute()){
```

Ostoskori-tauluun ja Tuote-tauluun tehdään kysely kirjautuneen käyttäjän osalta, lajitellaan tuotenumeron mukaan ja lasketaan kappalemäärät jokaiselta tuotteelta. Tuotetaulusta mukaan otetaan myös tuotteen nimi. (ostoskori.php)

```
$sql="SELECT Ostoskori.idTuote, tuoteNimi, SUM(maara), hinta from Ostoskori JOIN
Tuote ON Ostoskori.idTuote=Tuote.idTuote
WHERE username='$username'
GROUP BY idTuote";
$ostoskori=$db->query($sql);
```

Poistetaan tiedot Ostoskori-taulusta (order.php), jota ennen trigger lisää ne Tilaukset-tauluun

```
$sql1="DELETE FROM Ostoskori
WHERE username='$username'";
$ostoskori=$db->query($sql1);
```

```
1 • CREATE
2   DEFINER=`t8vija01`@`%`
3   TRIGGER `opisk_t8vija01`.`Ostoskori_BEFORE_DELETE`
4   BEFORE DELETE ON `opisk_t8vija01`.`Ostoskori`
5   FOR EACH ROW
6   BEGIN
7     DECLARE str VARCHAR(255);
8     SET str=CONCAT('Tuotenro: ',OLD.idTuote,' kpl: ',OLD.maara);
9     INSERT INTO Tilaukset VALUES(NULL,OLD.username,NOW(),str);
10  END
```

Tilaukset-taulusta haetaan kirjautuneen käyttäjän perusteella tiedot. (tilaus.php)

```
$sql1="SELECT * FROM Tilaukset WHERE username='$username'";
$tilaus=$db->query($sql1);
```

## Analyysi

Työ aiheutti paljon haastetta ja ajankäytön jakaminen oli ongelmallista. Tietokannan suunnittelu olisi voinut sujua paremminkin, mutta lopputulos on kuitenkin toimiva. Tilaukset-aulun olisi voinut toteuttaa myös niin, että siinä olisi ollut relaatio kannan muihin tauluihin.

Php-koodaaminen oli täysin uutta asiaa ja siinä vastaan tuli paljon haasteita. Oppiminen oli iso osa prosessia. Toteutus on omasta mielestäni tähänastisiin opintoihin nähden kelvollinen ja toimiva. Koodissa on varmasti paljon "turhaa" tai sitä voisi järkevoittaa monella tapaa. Omaan suoritukseen olen kuitenkin tyytyväinen.

Tulevaisuudessa vastaavissa projekteissa tärkeään asemaan nousee suunnitelmallisuus ja ajankäyttö. Nyt aikaa kului paljon siihen, että jokin asia toteutettiin toisin kuin alkuperäisessä suunnitelmassa oli määritelty. Tämä taas aiheutti uusia ongelmia ratkaistavaksi. Ongelmat kuitenkin ratkesivat ja projekti valmistui ajoissa.