

Verkkokauppa

WWW-palvelinohjelmointi
IIM50300

Jani Kähkönen

Raportti
04 2016

Ohjelmistotekniikan koulutusohjelma



Sisältö

| | | |
|----------|--------------------------------|----------|
| 1 | Kuvaus | 2 |
| 2 | Toteutus..... | 2 |
| 2.1 | Arkkitehtuuri..... | 2 |
| 2.2 | Sovelluksen luokkakaavio | 3 |
| 2.3 | Sovelluksen tietokanta | 3 |
| 3 | Aikataulu..... | 4 |
| 4 | Itsearvio | 4 |
| 4.1 | Myöhemmin lisättyä..... | 4 |
| 5 | Liitteet | 5 |

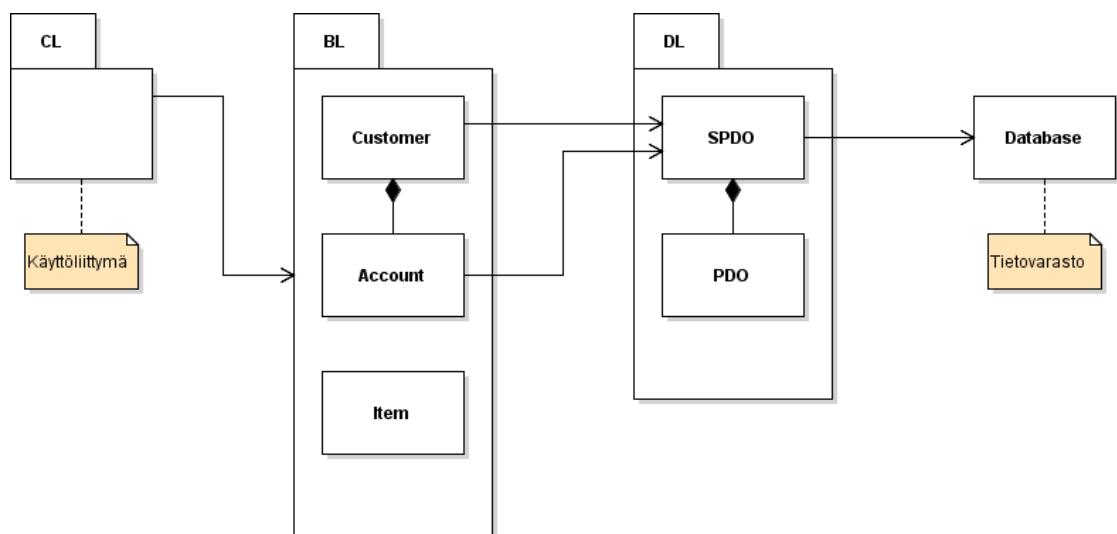
1 Kuvaus

Käytettävyydeltään yksinkertainen verkkokauppa, jonka avulla käyttäjät voivat selata ja tilata tuotteita, kuten tyypillisesti verkkokaupoissa. Sovelluksella voidaan myös katsella rekisteröinnin yhteydessä annettuja henkilötietoja sekä omaa tilaushistoriaa. Sovelluksen käyttö edellyttää rekisteröitymistä koska sovellukseen täytyy kirjautua. Verkkokaupan ideana oli että sitä voisi käyttää mahdollisimman moneen erilaiseen käyttötarkoitukseen. Tarkoituksena oli ottaa siis mahdollisimman vähän kantaa siihen, millaista tietoa verkkokaupassa esitetään.

2 Toteutus

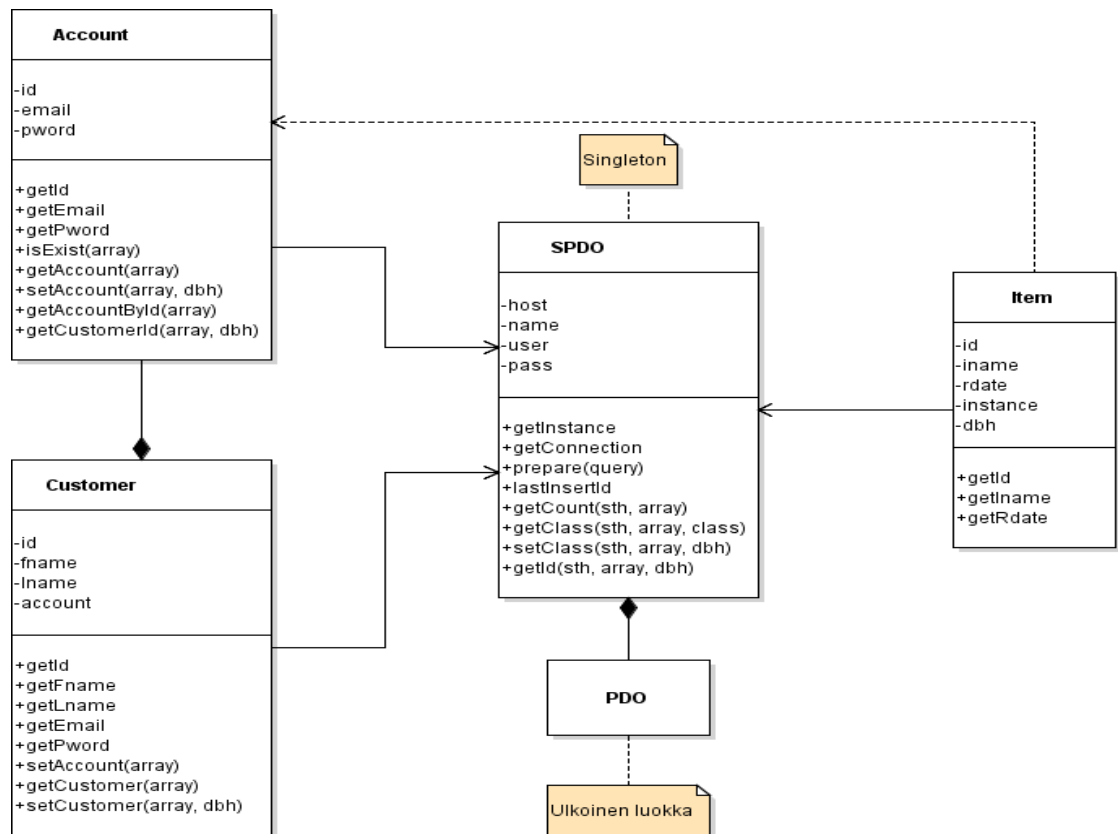
2.1 Arkkitehtuuri

Sovelluksen arkkitehtuuri muistuttaa 3-kerros arkkitehtuuria, jossa käyttöliittymä on erotettu sovelluslogiikka kerroksesta ja tietokanta kerroksesta. Esimerkiksi kirjautuessa käyttöliittymä kutsuu BL-kerroksen staattista `isExist` metodia, joka puolestaan kutsuu DL-kerroksen `getCount` metodia mikä palauttaa tietokannasta löydettyjen tulosten lukumäärän. Tämä lukumäärän palautetaan BL-kerrokselle jossa saatu tulos muunnetaan `true` tai `false` arvoksi ja palautetaan takaisin CL-kerrokselle.



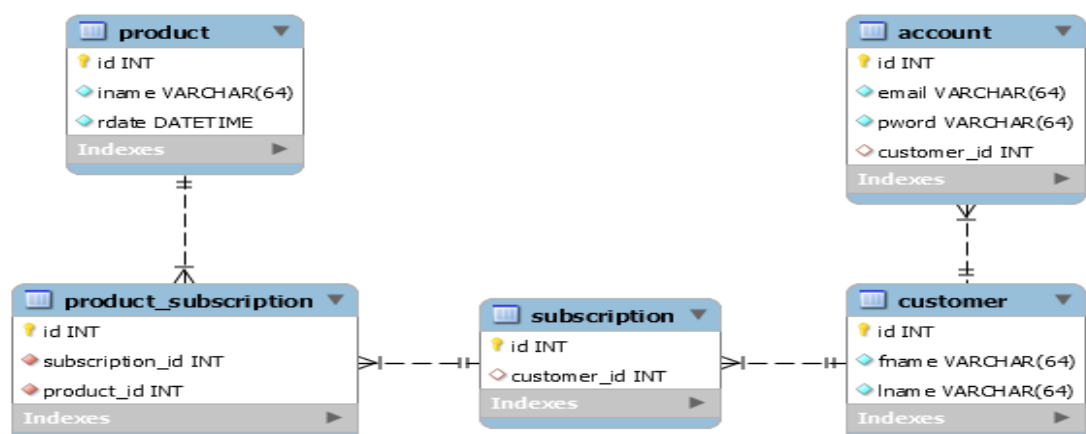
2.2 Sovelluksen luokkakaavio

Sovelluksen luokkakaavio kokonaisuudessaan, SPDO-luokka käsittelee sovelluksen tietokantaa ja luokissa Customer, Account ja Item sijaitsee sovelluslogiikka. Luokkien metodeissa välitetty array on sql-kyselyissä käytetty parametrilista.



2.3 Sovelluksen tietokanta

Sovellus käyttää MySQL-tietokantaa, joka sijaitsee student.labranet.jamk.fi palvelimella. Tietokantojen luontiin käytetyt lauseet on esitetty liitteessä 1.



3 Aikataulu

Harjoitustyön tekemiseen kului aikaa n. 60 tuntia. Vastuualueet: PHP, MySQL, HTML, CSS ja suunnittelu.

| VKO | Tehtävä | Aika |
|-----|-----------|-------------------------------------|
| 13 | 5h | Suunnitelma |
| 14 | 10h | Käyttöliittymä ja tietokanta. |
| 15 | 20h | Käyttöliittymä ja sovelluslogiikka. |
| 16 | 25h + 10h | Käyttöliittymä ja sovelluslogiikka. |

4 Itsearvio

Mielestäni harjoitustyönä ”verkkokauppa” oli sopivan haastava, monipuolinen mutta yllättävän työläs. Harjoitustyö eteni ilman merkittäviä ongelmia mutta omasta mielestäni sain odotettua vähemmän aikaiseksi. Onnistuin kuitenkin tavoitteissani ohjelmoimaan hyvin yksinkertaisen ja helppokäyttöisen verkkokauppa sovelluksen. Arvosana ehdotukseni kuitenkin on 4 – 5, perusteluina harjoitustyö tehty yksin, harjoitustyössä käytetty luokkia ja tietokantoja monipuolisesti sekä täyttää hyvin harjoitustyön vaatimuksia.

ZIP: <http://student.labranet.jamk.fi/~H9575/IIM50300-HTYO/IIM50300-HTYO.zip>

4.1 Myöhemmin lisättyä

- Rekisteröintitietojen validointi ja ilmoitukset käyttäjälle.
- Kirjautumistietojen validointi ja ilmoitukset käyttäjälle.
- Virheen sattuessa syötetyt tiedot lähetetään takaisin lomakkeella lukuun ottamatta salasanaa.
- Sovelluksen käyttö on estetty kirjautumattomilta käyttäjiltä.
- Logout ja tilaus painikkeet.
- Tilaus tallennetaan tietokantaan asianmukaisesti.
- Tilaushistoria.
- Koodia on muokattu enemmän oliopohjaiseksi.
- Korjattu bugeja.

Lisäyksien myötä harjoitustyön tuntimäärä nousi n. 70 tuntiin.

5 Liitteet

Liite 1. tietokannat.

Liite 1. tietokannat

```
-- -----
-- Table customer
-- -----

CREATE TABLE customer
(
    id INT NOT NULL AUTO_INCREMENT,
    fname VARCHAR(64) NOT NULL,
    lname VARCHAR(64) NOT NULL,
    PRIMARY KEY (id)
);

-- -----
-- Table account
-- -----

CREATE TABLE account
(
    id INT NOT NULL AUTO_INCREMENT,
    email VARCHAR(64) NOT NULL,
    pword VARCHAR(64) NOT NULL,
    customer_id INT NULL,
    PRIMARY KEY (id),
    CONSTRAINT fk_account FOREIGN KEY (customer_id) REFERENCES
customer (id)
);

-- -----
-- Table subscription
-- -----

CREATE TABLE subscription
(
    id INT NOT NULL AUTO_INCREMENT,
    customer_id INT NULL,
    PRIMARY KEY (id),
    CONSTRAINT fk_subscription FOREIGN KEY (customer_id) REFERENCE-
RENCES customer (id)
);

-- -----
-- Table product
-- -----

CREATE TABLE product
(
    id INT NOT NULL AUTO_INCREMENT,
    iname VARCHAR(64) NOT NULL,
    rdate DATETIME NOT NULL,
    PRIMARY KEY (id)
);

-- -----
-- Table product_subscription
-- -----

CREATE TABLE product_subscription
(
    id INT NOT NULL AUTO_INCREMENT,
    subscription_id INT NOT NULL,
    product_id INT NOT NULL,
    PRIMARY KEY (id),
    CONSTRAINT fk_product_subscription_1 FOREIGN KEY (sub-
scription_id) REFERENCES subscription (id),
    CONSTRAINT fk_product_subscription_2 FOREIGN KEY (pro-
duct_id) REFERENCES product (id)
);
```