

Pizzapalvelu

Marko Hassinen
014029585

Tietokantasovellus
Syksy 2013

Sisällysluettelo

Johdanto	sivu 3
Yleiskuvaus järjestelmästä	sivu 4
Käyttötapaukset	sivu 5
Järjestelmän tietosisältö	sivu 6
Tietokohteet	sivu 7
Relaatiotietokantakaavio	sivu 10
Järjestelmän yleisrakenne	sivu 11
Järjestelmän komponentit	sivu 11
Sivukartta	sivu 15
Asennustiedot	sivu 16
Käyttöohje	sivu 16

Johdanto

Työn aiheeksi on valittu kurssimateriaalin Pizzapalvelu-tehtävä, jonka tarkoituksena on luoda virtuaalinen tilausjärjestelmä. Järjestelmässä tarjottavat tuotteet, kuten pizzat ja juomat, on jaettu eri tuoteryhmiksi. Kutakin tuotetta esitellään tuotekuvauksessa, osasta tuotteita on olemassa myös kuva. Järjestelmän mukaisiin valmiiksi suunniteltuihin pizzoihin voi halutessaan valita lisukkeita.

Pizzapalvelussa ilmaistaan tuotteiden hinnat, joiden muuttumista koskeva oikeus on säilytetty. Laskutuksessa huomioidaan kuitenkin tilausajankohtainen hinnasto.

Toimitusaika on ennalta sovittu. Jos toimitus sovittuun aikaan ei toteudu, asiakkaalle voidaan myöntää alennusta tilauksesta. Toimituksessa tapahtuneet häiriöt voivat johtaa tilauksen eväykseen. Asiakkaan on mahdollista purkaa tilauksensa, kuitenkin viimeistään tunti ennen sovittua toimitusaikaa.

Alla oleva toimintolista on lainattu osoitteesta

<http://advancedkittenry.github.io/aiheet/Pizzapalvelu.html> (luettu 3.11.2013, 15:07).

Toimintoja:

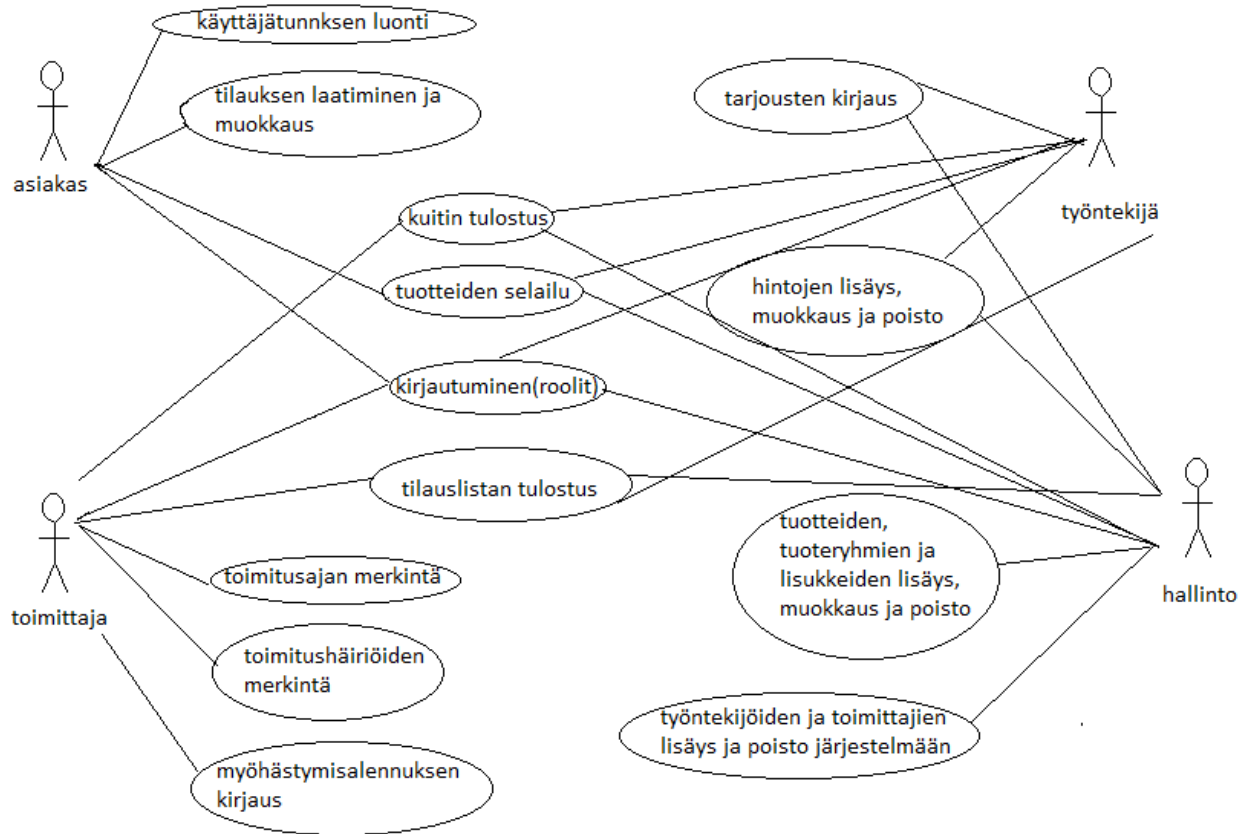
- tuotetietojen ja lisukkeiden lisäys, muokkaus ja poisto
- ajasta riippuvien hintojen lisäys, poisto ja muokkaus
- tuotteiden selailu
- tilauksen laatiminen ja muokkaus
- toimituksen ja mahdollisen myöhästymisalennuksen kirjaus
- tilauslistan tulostus
- kuitin tulostus
- tarjousten kirjaus
- ylläpidon kirjautuminen ja tilauksen tehneen asiakkaan tunnistaminen/kirjautuminen

Työssä hyödynnetään versionhallintaa, ja työ on nähtävissä osoitteessa

<https://github.com/mihassin/tietokantasovellus-s2013>. Toimiva ohjelmisto on tarkoitus laittaa users-palvelimelle. Työ toteutetaan PHP-ohjelmointikielellä. Työssä hyödynnetään tietokannan hallintajärjestelmää PostgreSQL. Alustavasti työ mahdollistaa vain yhden tietokannan käytön.

Yleiskuvaus järjestelmästä

Käyttötapauskaavio:



Käyttötapaukset

Asiakas

Asiakas avaa pizzapalvelun selaimellaan. Jos asiakkaalla ei ole käyttäjätunnuksia palveluun, voi hän luoda itselleen sellaisen. Järjestelmään kirjautumisen jälkeen asiakas voi selata tuotelistaa ja valita haluamansa tuotteet mahdollisine lisukkeineen tilaukseensa. Asiakastiedot saadaan käyttäjätiedoista. Asiakas merkitsee tilaukseen järjestelmän hyväksymän toimitusajan ja toimitusosoitteen. Tilausta voidaan muokata tai se voidaan poistaa vähintään tunti ennen sovittua toimitusaikaa.

Toimittaja

Toimittaja kirjautuu järjestelmään toimittajana ja tulostaa tilauslistan ennen toimitusta. Tilaus toimitetaan tilauksessa mainittuun toimitusosoitteeseen ja toimitus pyritään toteuttamaan sovittuun aikaan. Toimittaja merkitsee toimitusajan ja mahdolliset toimitushäiriöt sekä myöhästymisalennuksen. Toimittaja tulostaa asiakkaalle kuitin, johon on merkitty mahdollinen alennus.

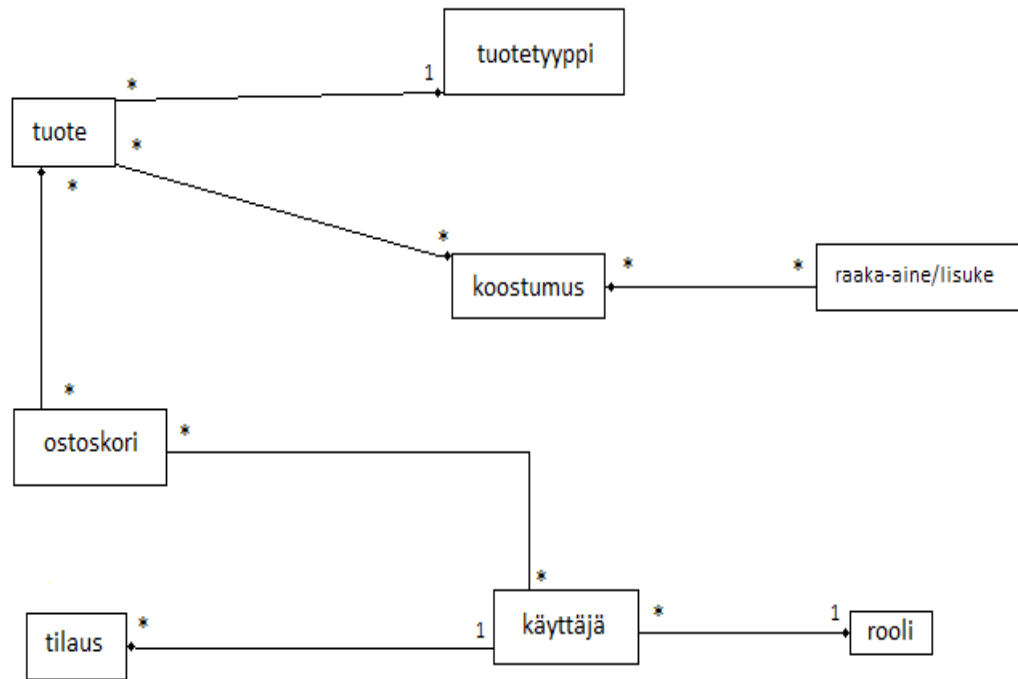
Työntekijä

Työntekijä kirjautuu järjestelmään työntekijänä. Työntekijän on kyettävä päivittämään hinnastoa ja merkitsemään tarjoukset järjestelmään. Lisäksi työntekijän pitää pystyä monitoroimaan sekä tuote- että tilauslistaa ja tarvittaessa tulostamaan kuitteja tilauksista.

Hallinto

Hallintoroolissa ei ole merkittäviä eroja työntekijän rooliin, mutta hallinnon on pystyttävä lisäämään, muokkaamaan ja poistamaan tuotteita, tuoteryhmiä tai lisukkeita. Hallinto myös sekä lisää että poistaa työntekijöitä ja toimittajia järjestelmään.

Järjestelmän tietosisältö



Tietokohteet

Työssä on kahdeksan tietokohdetta, joista kaksi on “mappereita”. Tietokohteille 'käyttäjä' ja 'tuote' on käytössä pienempiä hallinnointitaulukoita. Muutamalle tietokohteelle on luotu taulukko, josta ilmenevät avaimia lukuun ottamatta muut attribuutit.

Käyttäjä

Käyttäjä		
Attribuutti	Arvojoukko	Kuvailu
Etunimi	30 merkkiä pitkä merkkijono	Ilmaisee käyttäjän etunimen
Sukunimi	30 merkkiä pitkä merkkijono	Ilmaisee käyttäjän sukunimen
Sähköposti	255 merkkiä pitkä merkkijono	Ilmaisee käyttäjän sähköpostiosoitteen
Puhelin	255 merkkiä pitkä merkkijono	Ilmaisee käyttäjän puhelin numeron

Käyttäjille on oma yksilöivä id-attribuutti ja rooli-id-attribuutti, jolla ilmaistaan tietokohteiden 'käyttäjä' ja 'rooli' välinen relaatio. Sähköposti toimii järjestelmässä käyttäjätunnuksena. Salasana salataan tietokantaan, ja salauksen purkua varten on oma suola.

Rooli

Tietokohde 'rooli' toimii 'käyttäjä'-tietokohteen apuna yksilöimässä käyttäjien roolia järjestelmässä. Tietokohteella on kaksi attribuuttia: avain ja pienimuotoinen kuvaus.

Tuote

Tuote		
Attribuutti	Arvojoukko	Kuvailu
Nimi	30 merkkiä pitkä merkkijono	Ilmaisee tuotteen nimen
Kuvaus	255 merkkiä pitkä merkkijono	Mahdollista lisätietoa tuotteesta
Hinta	Reaaliluku	Ilmaisee tuotteelle merkityn hinnan

Tietokohde ilmaisee tuotteita, joita pizzapalvelu tarjoaa asiakkailleen. Palvelu vastaa tuotteen valmistuksesta ja toimituksesta. Tietokohteella on attribuuttina yksilöivä avain ja tuotetyypin avain, joka ilmaisee tietokohteiden välisen relaation.

Tuotetyyppi

Kyseessä on pienempi tietokohde, joka auttaa palvelun tuotteiden ja raaka-aineiden järjestelyssä. Tuotetyypillä on kaksi attribuuttia: yksilöivä avain ja pienimuotoinen kuvaus.

Raaka-aine

Raaka-aine		
Attribuutti	Arvojoukko	Kuvailu
Kuvaus	30 merkkiä pitkä merkkijono	Kertoo raaka-aineen nimen
Hinta	Reaaliluku	Kertoo raaka-aineen hinnan

Tietokohdetta 'raaka-aine' käytetään tuotteiden koostumuksen ilmaisemiseen. Järjestelmä sallii raaka-aineiden lisäyksen tuotteisiin. Tietokohteella on attribuutteina yksilöivä avain ja tuotetyypin avain.

Koostumus

'Koostumus'-tietokohdetta käytetään pizzatuotteiden koostumuksen ilmaisuun. Tietokohde toimii niin sanottuna mapperina, jolla liitetään raaka-aineet tuotteeseen. Tietokohteella on kolme attribuuttia: oma avain, tuotteen avain ja raaka-aineen avain.

Ostoskori

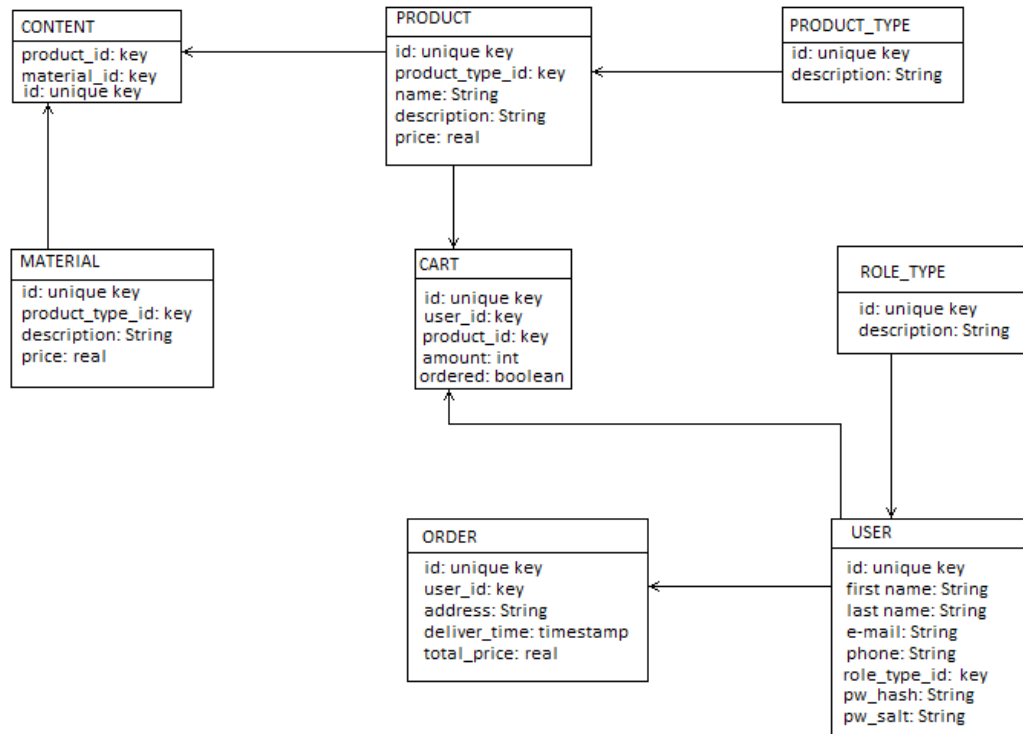
Ostoskori on järjestelmän toinen “mapperi”. Sillä ilmaistaan tilauksen sisältö. Tietokohteella on neljä attribuuttia: oma avain, käyttäjän avain, tuotteen avain, määrä (yhden tuotteen määrä) ja totuusarvo ordered.

Tilaus

Tilaus		
Attribuutti	Arvojoukko	Kuvailu
Toimitusosoite	255 merkkiä pitkä merkkijono	Kertoo asiakkaan valitseman toimitusosoitteen
Toimitusaika	Päivämäärä ja kellonaika	Kertoo asiakkaan valitseman toimitusajan
Yhteishinta	Reaaliluku	Toimituksen lopullinen hinta

'Tilaus'-tietokohde on tarkoitettu käyttäjille, jotka eivät ole asiakkaita. Tietokohteella on attribuuttina oma yksilöivä avain ja käyttäjän avain.

Relaatiotietokantakaavio



Järjestelmän yleisrakenne

Järjestelmä juurikansio on nimeltään htdocs. Htdocs sisältää kansiot sql, libs, css ja views, sekä php-tiedostoja. Sql kansio sisältää kaikki tarvittavat tiedostot tietokantojen alustukseen ja poistoon. Css kansiossa on tyylitiedostot. Libs kansiossa on php-tiedostoja, jotka tarjoavat järjestelmässä yleisesti käytettyjä toimintoja ja tietokannan hallinta toimintoja. Views kansion php-tiedostot ovat pääosin vastuussa html tägeistä, joilla määritetään sivujen rakennetta. Htdocsin php-tiedostot kasaavat tarvittavat views-kansion tiedostot ja liittävät libs-kansion toiminnallisuuden sivuihin, sekä määrittävät kyselyt, joita käytetään libs:n funktioissa.

Järjestelmän komponentit

index.php

Sivuston etusivu, jolla on näkymät (views/) header.php, etusivu.php ja footer.php.

etusivu.php

Etusivun html rakenteesta vastaava tiedosto, joka sijaitsee yleiskäyttöisten header.php ja footer.php välissä. Sisältää linkit ruokalistaan, juomalistaan ja lisukelistaan. Tiedosto sijaitsee views kansiossa

header.php

Sivuston navigaatio elementti, joka näkyy jokaisella sivulla. Elementin sisältö voi erota sisäänkirjautuneen käyttäjän roolista riippuen. Tiedosto ohjaa oikeaan tiedostoon, mikäli tarpeellista. Jos käyttäjä ei ole sisäänkirjautunut vastaa header.php navigaatio-elementin sisällöstä. Tiedosto sijaitsee views kansiossa.

customer-header.php/admin-header.php/employee-header.php/delivery-header.php

Sivustolla käytetään neljää eri roolia ja jokainen yllämainittu vastaa kysyisen roolin navigaatio-elementin sisällöstä. Jokainen näistä sijaitsee views kansiossa.

footer.php

Jokaisella sivun ala-reunassa on marginaali, jonka sisällöstä footer.php vastaa. Sivusto on suunniteltu näyttävän tässä elementissä sivuston päivityksestä kertovaa informaatiota, mutta sinne voidaan liittää esimerkiksi yhteystietoja tai muuta yleishyödyllistä informaatiota käyttäjälle.

menu.php/drinks.php/materials.php

Nämä tiedostot sijaitsevat juurikansiossa ja ovat vastuussa näkymien ja funktioiden kasaamisesta sivulle. Jokainen tiedosto vie uudelle sivulle sivustolla. Jokainen myös kutsuu libs/listing.php tiedoston funktiota getList(), jonka tehtävä on listata tietokannasta tarvittava tieto sivulle. Tiedostot pitävät sisällään kysely merkkijonot, joista osa haetaan libs/listqueries.php funktioista. Tiedostot myös mahdollistavat näkymän listojen tietojen lisäys, poisto ja muunnos toiminnoille. Toiminnot kuitenkin näkyvät vain tarvittavan roolin omaaville käyttäjille. Views kansiossa on, jokaista tiedostoa vastaava samanniminen tiedosto, joissa on rivi html-merkkauksia. Syy tähän on, että järjestelmässä on tarkoitus erottaa mahdollisimman hyvin html-merkkaukset views-kansioon.

libs/listqueries.php

Tiedosto sisältää kysely merkkijonoja.

libs/listing.php

Tiedosto sisältää funktion getList(\$kysely), jonka tarkoitus on listata parametrin \$kysely mukaisen tietokanta kyselyn tietoja html:n taulukko esityksenä. Tiedosto käyttää libs/db_connect.php yhteyden muodostamiseen tietokantaan.

libs/db_connect.php

Tiedosto sisältää funktion getConnection(), jonka on tarkoitus palauttaa yhteys kysyvälle tiedostolle. Tiedosto saa tarvittavat tiedot tietokanta yhteyden muodostamiseen tiedostolta secret.php, joka sijaitsee libs kansiossa.

views/listbuttons.php

Tiedosto sisältää nappi esityksen. Nappeja on kolme lisäys, poisto ja muunnos napit, jotka vievät uudelle sivulle. Lisäys ja poisto napit näkyvät vain admintason roolille ja muunnos työntekijöille ja kuljetukselle. Poisto napin lomake sisältää myös listaus esityksen, jossa valitaan poistettava tuote listasta. Listaus saadaan tiedoston libs/dropList.php funktiosta getDropList(). Lisäys nappi vie juuren tiedostoon add-item.php ja muutos add-offer.php.

libs/dropList.php

Tiedosto ottaa yhteyden tietokantaan db_connect.php tiedoston avulla ja listaa kyselyn mukaiset optiot listaan.

login.php

Lisää tiedoston views/login.php sisältämän lomakkeen.

views/login.php

Sisältää html kaavakkeet kirjautumista varten. Kaavakkeen lähetys vie juurikansion tiedostoon checklogin.php

checklogin.php

Tiedosto muodostaa yhteyden tietokantaan libs/db_connect.php tiedoston kautta. Kirjautumistietoina käytetään käyttäjän salasanaa ja sähköpostiosoitetta. Tiedot tarkistetaan tietokannasta. Mikäli tiedoissa on vikaa, näytetään käyttäjälle views/failure.php tiedoston sisältö. Jos tiedot ovat kunnossa, aloitetaan kirjautumissessio, jossa pidetään kirjaa käyttäjän roolista ja tunnuksesta, lopuksi käyttäjä ohjataan sivuston etusivulle.

views/failure.php

Tiedosto ilmoittaa kirjautumistietojen virheellisyydestä, muttei kerro käyttäjälle erityisesti missä oli vikaa tietoturvalisistä syistä. Käyttäjä näkee myös linkin jota kautta on mahdollista palata etusivulle.

logout.php

Päättää sisäänkirjautumisesta syntyneen session.

users.php

Tarjoaa taulukoituja tietoja käyttäjistä. Tietojen listaukseen käytetään libs kansion listing.php tiedostoa

jonka käyttämä kysely saadaan libs kansion listqueries.php tiedostosta. Sivun otsikko saadaan samannimisestä tiedostosta users.php, joka on views kansiossa.

orders.php

Tarjoaa taulukoituja tietoja tilauksista. Tietojen listaukseen käytetään libs kansion listing.php tiedostoa jonka käyttämä kysely saadaan libs kansion listqueries.php tiedostosta.

newuser.php

Sivun sisältö haetaan views kansion samannimisestä tiedotosta newuser.php.

views/newuser.php

Tiedosto sisältää html lomakkeen uuden käyttäjätunnuksen luontia varten. Mikäli jo sisäänkirjautunut käyttäjä yrittää päästä sivulle, hänet ohjataan takaisin etusivulle. Lomakkeen tiedot lähetetään juurikansion tiedoston newuserCheck.php tutkittavaksi.

newuserCheck.php

Sisäänkirjautunut käyttäjä ohjataan etusivulle. Tiedosto ottaa yhteyden tietokantaan libs kansion db_connect.php tiedoston kautta. Tiedosto tutkii uuden käyttäjän lomakkeen tiedot. Mikäli lomakkeessa annetussa tiedossa on vikaa, siitä kerrotaan ja näytetään views kansion newuserAgain.php tiedoston sisältämä linkki tunnuksen luontiin.

Sivusto sallii vain yhden käyttäjätunnuksen per sähköpostiosoite. Varattu sähköposti tutkitaan libs kansion checkEmail.php tiedoston avulla.

Jos tiedot olivat kunnossa luodaan kysely, jonka lähetetään tietokantaan. Lopuksi käyttäjälle ilmoitetaan tunnuksen luonnin onnistumisesta ja tarjotaan linkki etusivulle.

views/newuserAgain.php

Sisältää linkin juuren newuser.php tiedostoon.

remove-itemCheck.php

Tiedosto vastaa tuote-, juoma- ja lisuke listojen poisto toiminnon kyselyiden suorittamisesta tietokantaan. Sivulla ei ole esitystä ja käyttäjä ohjataan välittömästi etusivulle.

add-offer.php

Hakee tiedostot checkAccess.php ja dropList.php kansiota libs. Sivun sisältö on views kansion add-offer.php tiedostossa.

views/add-offer.php

Sisältää html lomakkeen tuotteen hinnan muutosta varten. Lomakkeen tarkistus ohjataan juuren tiedostoon add-offerCheck.php

add-offerCheck.php

Tiedosto luo yhteyden tietokantaan libs kansion db_connect.php tiedoston avulla. Libs kansion checkAccess.php ohjaa sisäänkirjaantumattomat ja asiakkaat etusivulle. Mikäli hinta ei ole numeerinen ohjataan käyttäjä takaisin lomakkeeseen. Lomakkeen tietojen pohjalta päivitetään tietokantaan tiedot. Lopuksi käyttäjä ohjataan etusivulle.

add-item.php

Hakee tiedostot checkAccess.php ja dropList.php kansiota libs. Sivun sisältö on views kansion add-item.php tiedostossa. Sisältöä varten luodaan kysely merkkijono.

views/add-item.php

Sisältää html lomakkeen uuden tuotteen luomista varten. Lomake mahdollistaa jokaisen tuotetyypin luonnin. Lomakeen tarkistus ohjataan juuren tiedostoon add-itemCheck.php

add-offerCheck.php

Tiedosto luo yhteyden tietokantaan libs kansion db_connect.php tiedoston avulla. Libs kansion checkAccess.php ohjaa sisäänkirjaantumattomat ja asiakkaat etusivulle. Mikäli hinta ei ole numeerinen ohjataan käyttäjä takaisin lomakkeeseen. Lomakkeen tietojen pohjalta päivitetään tietokantaan tiedot. Tietokanta kysely eroaa lisukkeiden ja muiden tuotetyyppien välillä. Lopuksi käyttäjä ohjataan etusivulle.

libs/checkAccess.php

Ohjaa sisäänkirjautumattomat ja asiakkaat etusivulle

libs/checkEmail.php

Sisältää funktion emailExists(\$email), joka tarkistaa kuinka monta parametrin \$email mukaista sähköposti osoitetta on tietokannassa. Yhteys tietokantaan luodaan db_connect.php tiedoston kautta.

cart.php

Huolehtii ostoskorin esityksen organisoinnista. Näkyvyys vain kirjautuneille. Listauksen suorittaa libs kansion list-cart.php. Kyselyt noudetaan libs kansion listqueries.php tiedostosta. Poisto ja lisäys napit haetaan views kansion cartbuttons.php tiedostosta.

libs/list-cart.php

Sisältää funktion getCartList(\$kysely), joka palauttaa parametrin \$kysely mukaisen esityksen ostoskorin listaukselle. Jos ostoskori on tyhjä, ilmoitetaan siitä.

views/cartbuttons.php

Tulostaa ostoskori sivulle toiminnallisuutta tarjoavat napit “Lisää tuote”, “Lisää juoma”, “Poista tuote” ja “Tilaa”. Kaksi jälkimmäistä näkyvät käyttäjälle vain, jos ostoskorissa on vähintään yksi tuote.

add-item-cartCheck.php

Tiedosto tarkistaa ostoskorin “Lisää tuote” napin lomakkeen ja päivittää tietokannan. Päivityksen jälkeen käyttäjä ohjataan takaisin ostoskori sivulle.

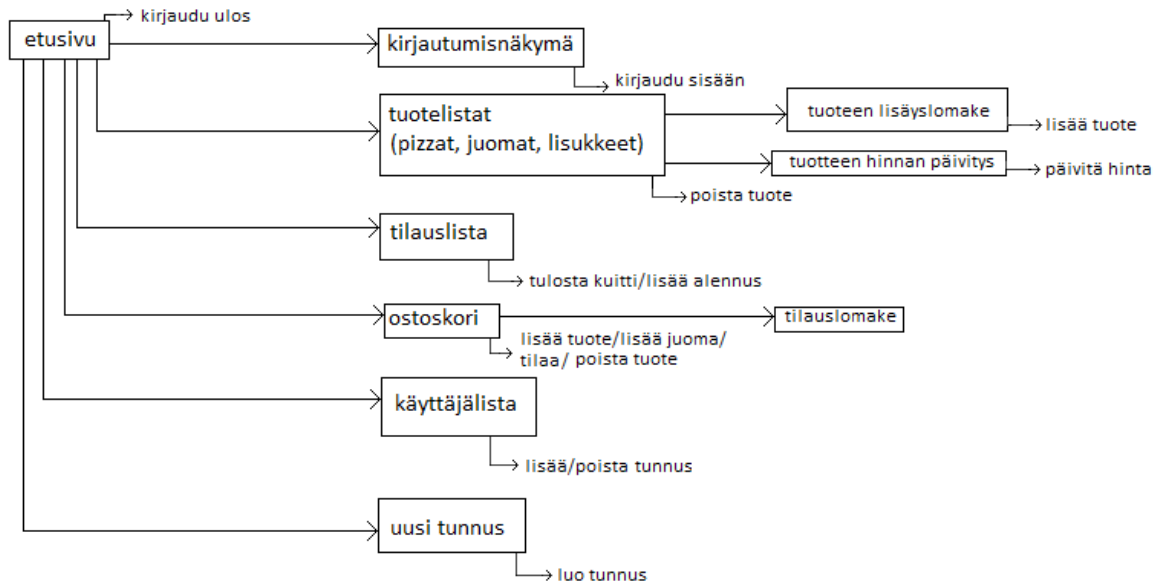
add-drink-cartCheck.php

Tiedosto tarkistaa ostoskorin “Lisää juoma” napin lomakkeen ja päivittää tietokannan. Päivityksen jälkeen käyttäjä ohjataan takaisin ostoskori sivulle.

remove-item-cartCheck.php

Tiedosto tarkistaa ostoskorin “Poista tuote” napin lomakkeen ja päivittää tietokannan. Päivityksen jälkeen käyttäjä ohjataan takaisin ostoskori sivulle.

Sivukartta



Asennustiedot

Sivuston käyttö edellyttää kaikkien siihen sisältyvien tiedostojen lisäämistä, mikä tulisi ottaa huomioon, jos projektia hyödynnetään harjoitustyön ulkopuolisessa yhteydessä.

Jos sivusto halutaan siirtää, tulee siinä olevia tietoja muuttaa niin, että yhteyden muodostaminen tietokantaan on mahdollista. Asennuksessa suositellaan harjoitustyössä käytettyä kansiorakennetta. Mikäli kansiorakenteeseen halutaan tehdä muutoksia, tulee huomioida mahdolliset muutokset tiedostojen osoittamisessa ja tiedostoissa, jotka osoittavat muihin tiedostoihin.

Oletuksena on, että tietokannan hallintajärjestelmä PostgreSQL ja ohjelmointikieli PHP ovat käytettävissä palvelimella, jolle sivusto asennetaan. Sivustoa käytetään oletuksellisesti selaimella, mutta selainohjelmaan ei oteta kantaa.

Vaikka sivusto on suunniteltu pizzeriayrityksen sähköiseen palveluun, voidaan järjestelmän käyttö yleistää pienin muutoksin muillekin ravintolayrityksille. Yksinkertaisten muutosten avulla on myös mahdollista kohentaa sivuston ulkoasua. Muutokset tehdään views/header.php-tiedostoon, jonne merkitään sivustossa hyödynnettävät tyylitiedostot. Tyylitiedostot tulisi asentaa css/-kansion alle.

Käyttöohje

Harjoitustyö löytyy osoitteesta <http://mihassin.users.cs.helsinki.fi/>. Sivuston käyttäminen ei edellytä kirjautumista, mutta kirjautuminen takaa laajemman joukon sivuston käytettömahdollisuuksia.

Kirjautumisessa huomioidaan käyttäjän rooli, joka määrää käyttäjälle tarpeelliset käyttösovellukset. Sivustolla on käytössä neljä eri roolia, joista jokaiselle tarjotaan testausmahdollisuutta. Alla olevassa taulukossa esitellään salasanoineen sivuston testaukseen suunniteltuja tunnuksia, jotka eroavat rooleiltaan.

tunnus	salasana	rooli
admin@test	test	admin
delivery@test	test	kuljetus
employee@test	test	työntekijä
user@test	test	käyttäjä