

Sisällysluettelo

OHJELMISTON PROTOTYYPIN DOKUMENTOINTI	2
Kuvaus ohjelman tietosisällöstä	2
Tietokannan rakenne	2
Tietokannan SQL-luontilauseet	2
Tietojen lisääminen käyttötapauksena	3
Käyttötapauksen kuvaus lomakkeella	3
Ohjelmiston käyttöliittymän kuvaus	4
Päävalikko ”Main menu”	4
All employees	5
Add employee	6
Puutteelliset tiedot	7
Delete employee	8
Update employee	9
Change employee data	9
All employees (JSON)	10
Ohjelmiston moduulikaavio	11
Kommentit ohjelmakoodissa	12
Esimerkki 1: henkilonlisays.php (Add employee)	12
Esimerkki 2: hakuHenkilonumerolla.php (Update employee)	13
Esimerkki 3: tkanta.html (All employees (JSON))	13

OHJELMISTON PROTOTYYPIN DOKUMENTOINTI

Kuvaus ohjelman tietosisällöstä

Tietokannan rakenne

Ohjelma sisältää tietokannan, johon työntekijöiden tiedot talletetaan. Käytössä on SQL-tietokanta katta. Tietokannassa on yksi taulu henkilosto, jonka rakenne näkyy alla. Perusavain on henkilöNumero. Osa tiedoista on pakollisia.

henkilosto			
Sarake	Tietotyyppi	Pakollisuus	Rajoitukset
henkiloNumero	int(11)	kyllä	perusavain
etunimi	varchar(20)	kyllä	
sukunimi	varchar(30)	kyllä	
puhelin	varchar(15)	kyllä	
osasto	varchar(15)	ei	
palkka	decimal(6,2)	ei	

Tietokannan SQL-luontilauseet

SQL-luontilause löytyy tiedostosta henkilosto.sql. Alla näkyy myös esimerkkilause kokeiluarvojen syöttämisestä tietokantaan.

```
henkilosto.sql
1 CREATE TABLE henkilosto(
2   henkiloNumero INTEGER NOT NULL PRIMARY KEY,
3   etunimi VARCHAR(20) NOT NULL,
4   sukunimi VARCHAR(30) NOT NULL,
5   puhelin VARCHAR(15) NOT NULL,
6   osasto VARCHAR(15),
7   palkka DECIMAL(6,2)
8 );
9
10 INSERT INTO henkilosto VALUES
11 ('1', 'Matti', 'Puro', '0403234255', 'IT', '4400.00'),
12 ('2', 'Pirjo', 'Hakkila', '0445144086', 'Hallinto', '4100.00'),
13 ('3', 'Liisa', 'Wallin', '0503250251', 'Markkinointi', '3800.00');
```

Tietojen lisääminen käyttötapauksena

Ohjelmassa voidaan hakea kaikki työntekijät taulukossa tai JSON-muodossa, muuttaa yksittäisen työntekijän tietoja ja lisätä tai poistaa yksittäinen työntekijä. Alla esitetään uuden työntekijän tietojen lisäämisen käyttötapaus "Add employee".

Käyttötapauksen kuvaus lomakkeella

Käyttötapauksen nimi	Add employee
Yleiskuvaus	Käyttäjä lisää uuden työntekijän tiedot järjestelmään.

Päiväys Versio Laatija	21.11.2019 v1 Mia Kauppila
Prosessi	Työntekijöiden tietojen tallennus ja käsittely
Käyttäjäroolit	Sihteeri, henkilöstöhallinto
Laukaisija	Tarve tallentaa uuden työntekijän tiedot
Esitiedot	Käyttäjä on käynnistänyt ohjelman ja tietokantayhteys toimii. Käyttäjä on ohjelman päävalikossa "Main menu".

Käyttötapauksen kuvaus	
1	Käyttäjä valitsee päävalikosta "Main menu" painikkeen "Add employee", joka avaa uuden sivun työntekijän tietojen syöttämistä varten.
2	Käyttäjä syöttää uuden työntekijän tiedot lomakkeen kenttiin ja tallentaa "Save"-painikkeella tiedot järjestelmään.
3	Järjestelmä tarkistaa, että pakolliset kentät on täytetty ja tietojen muoto on oikea. Mikäli tiedoissa on puutteita, siirrytään poikkeukseen (P1).
4	Järjestelmä tekee lisäyskyselyn tietokantaan. Jos lisäyskysely ei onnistu, siirrytään poikkeukseen (P2).
5	Järjestelmä tallentaa tiedot kantaan ja ilmoittaa käyttäjälle tietojen tallennuksen onnistumisesta. Viesti esim.: "Employee number 5 added".

Poikkeukset	
P1	Käyttäjälle näytetään ohjeteksti koskien kenttää, jossa on puutteita. Ohjeteksti tyhjälle kentälle on "Täytä tämä kenttä" ja väärin täytetylle kentälle "Käytä pyydettyä muotoilua" + kenttää koskeva muodon aputeksti. Käyttäjä voi korjata tiedot ja jatkaa eteenpäin painamalla uudestaan "Save"-painiketta, jolloin palataan käyttötapauksen normaaliin kulkuun.
P2	Tietokannassa on jo työntekijä syötetyllä numerolla. Järjestelmä ilmoittaa käyttäjälle, että työntekijän lisäys tietokantaan ei onnistunut. Viesti: It was not possible to add the employee. Käyttäjä voi palata päävalikkoon "Back to menu"-painikkeella.

Tarvittava syöte käyttäjältä	Käyttäjän tulee syöttää lomakkeeseen seuraavat tiedot (pakolliset tiedot tummennettu): 1. Number 2. First name 3. Last name 4. Telephone 5. Department 6. Salary
Järjestelmän ulosanti käyttäjälle	Käyttäjä näkee aputekstin liikuttaessaan hiirtä pakollisten kenttien kohdalla tai ohjetekstin selaimen kielellä painaessaan "Save" ennen kuin pakolliset kentät on täytetty. Pakollisten kenttien ulosanti on seuraavanlainen:

	<ol style="list-style-type: none">1. aputeksti: Required format '001' ohjeteksti: Käytä pyydettyä muotoilua. + Required format: '001'2. apu-/ohjeteksti: Täytä tämä kenttä.3. apu-/ohjeteksti: Täytä tämä kenttä.4. aputeksti: Required format '0401234567' ohjeteksti: Käytä pyydettyä muotoilua. + Required format: '0401234567'
Lopputulokset	Työntekijän tiedot on tallennettu tietokantaan.
Muuta	Tiedot tallennetaan SQL-tietokantaan katta, jota voi hallinnoida phpMyAdminin kautta.

Ohjelmiston käyttöliittymän kuvaus

Päävalikko "Main menu"

- sisältää viisi painiketta, joiden kautta ohjelma toimii.

Main menu

All employees

Add employee

Delete employee

Update employee

All employees (JSON)

All employees

- avaa listauksen kaikista työntekijöistä.

All employees

Number	First name	Last name	Telephone	Department	Salary
1	Viivi	Stromberg	0503256520	Hallinto	4100.00
2	Pirjo	Hakkila	0445144086	Hallinto	4100.00
3	Liisa	Wallin	0503250251	IT-tuki	3500.00
4	Jussi	Penttinen	0408788100	IT	4000.00
5	Mikko	Salonen	0443234255	Markkinointi	3600.00

Back to menu

[Add employee](#)

- avaa uuden sivun, jolle syötetään uuden työntekijän tiedot.

Add new employee

Number:	<input type="text"/>
First name:	<input type="text"/>
Last name:	<input type="text"/>
Mobile phone:	<input type="text"/>
Department:	<input type="text"/>
Salary:	<input type="text"/>


Save

Back to menu

Puutteelliset tiedot

- Jos käyttäjän syöttämissä tiedoissa on puutteita, näytetään ohjeteksti.

Add new employee

Number:	<input type="text" value="005"/>
First name:	<input type="text" value="Matti"/>
Last name:	<input type="text" value="Matti"/>
Mobile phone:	<input type="text" value="050-5322144"/>
Department:	<div> Käytä pyydettyä muotoilua. Required format: '0401234567'</div>
Salary:	<input type="text" value="0400"/>

Save

Back to menu

Delete employee

- avaa sivun, jolla voi poistaa työntekijän tiedot. Lista tietokannassa olevien työntekijöiden tiedoista nimitasolla on myös samanaikaisesti näkyvissä. Kun yksi työntekijä on poistettu, lista työntekijöistä päivittyy.

Delete employee

Give the number of the employee to be deleted:

All employees

Number	First name	Last name
1	Viivi	Stromberg
2	Pirjo	Hakkila
3	Liisa	Wallin
4	Jussi	Penttinen
5	Mikko	Salonen

[Update employee](#)

- avaa sivun, jolle täytyy syöttää työntekijän numero etsimistä varten.

Change employee data

Give the number of the employee to be changed:

[Change employee data](#)

- Jos työntekijä löytyy, avautuu tämän jälkeen sivu, jolla työntekijän tietoja voi muuttaa ja lopuksi päivittää tiedot "Update"-painikkeella.

Change employee data

Number:

First name:

Last name:

Telephone:

Department:

Salary:

All employees (JSON)

- näyttää kaikkien työntekijöiden tiedot JSON-muodossa

All employees (JSON)

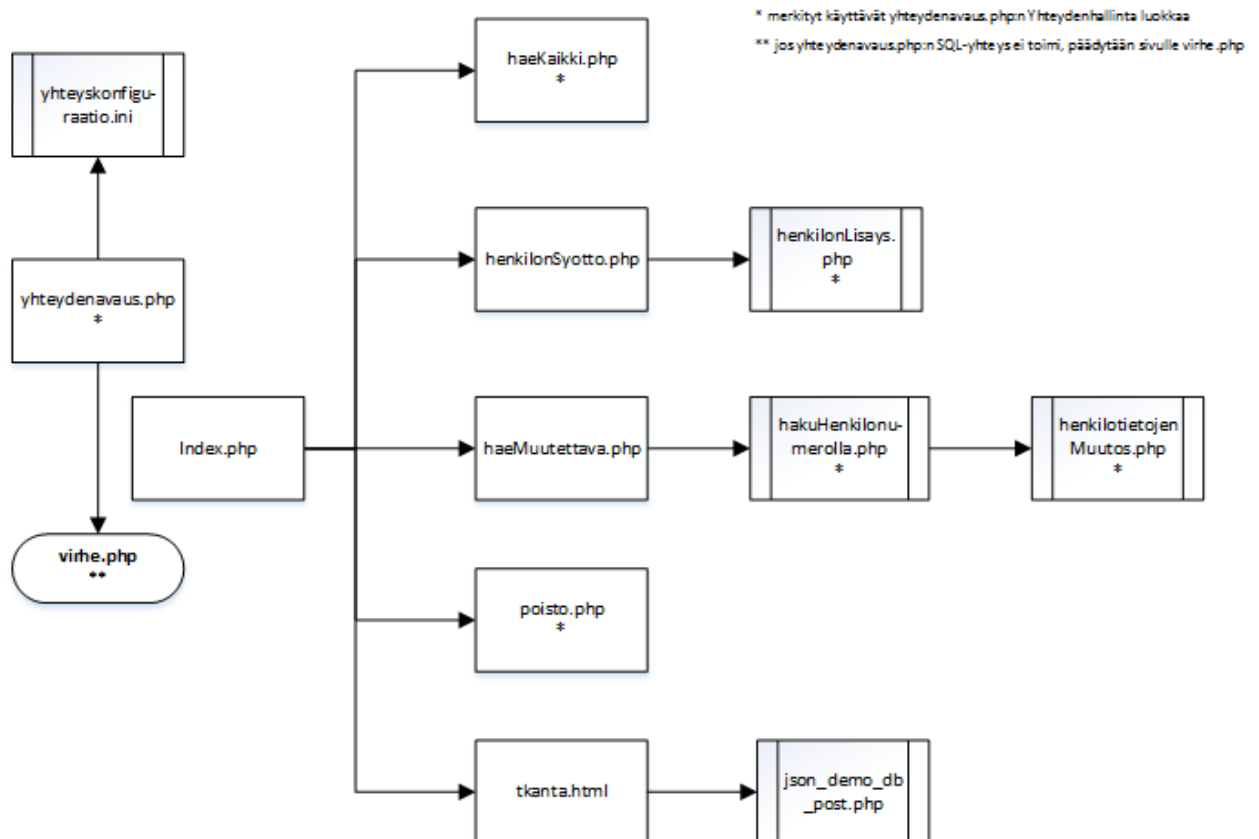
```
[{"henkiloNumero": "1", "etunimi": "Viivi", "sukunimi": "Stromberg",  
  "osasto": "Hallinto", "palkka": "4100.00"},  
{"henkiloNumero": "2", "etunimi": "Pirjo", "sukunimi": "Hakkila", "os  
asto": "Hallinto", "palkka": "4100.00"},  
{"henkiloNumero": "3", "etunimi": "Liisa", "sukunimi": "Wallin", "osa  
sto": "IT-tuki", "palkka": "3500.00"},  
{"henkiloNumero": "4", "etunimi": "Jussi", "sukunimi": "Penttinen", "  
osasto": "IT", "palkka": "4000.00"},  
{"henkiloNumero": "5", "etunimi": "Mikko", "sukunimi": "Salonen", "os  
asto": "Markkinointi", "palkka": "3600.00"}]
```

Back to menu

Ohjelmiston moduulikaavio

Moduulikaavio

7. joulukuuta 2019



Kommentit ohjelmakoodissa

Esimerkki 1: henkilonlisays.php (Add employee)

```
// tarkistetaan että pakolliset kentät eivät ole tyhjiä
if(empty($henkilonumero) || empty($etunimi) || empty($sukunimi) || empty($puhelin)){
    print("<h3>Please fill in the employee data.</h3>");
}
else{
    //suoritetaan lisäyskysely jos pakolliset tiedot ok
    $lisattyLkm = $yhteydenhallinta->suoritaPaivitysLause(
        "insert into henkilosto "
        . "(henkiloNumero, etunimi, sukunimi, puhelin, osasto, palkka) "
        . "values (?, ?, ?, ?, ?, ?)",
        Array($henkilonumero, $etunimi, $sukunimi, $puhelin, $osasto, $palkka));

    //jos lisättiin uusi työntekijä, printataan tiedot
    if($lisattyLkm > 0){
        print("<h3>Employee number " . $henkilonumero . " added</h3>");
        //tai virheviesti
    } else{
        print("<h3>It was not possible to add the employee.</h3>");
    }
}
```

Esimerkki 2: hakuHenkilonumerolla.php (Update employee)

```
$henkilonumero = htmlspecialchars(trim($_POST['henkilonumero']),  
ENT_QUOTES, 'UTF-8');  
//suoritetaan sql-kysely  
$haettava = $yhteydenhallinta->suoritaHakuLause(  
"select henkiloNumero,etunimi,sukunimi,puhelin,osasto,palkka"  
."from henkilosto where henkiloNumero=?",  
Array($henkilonumero));  
//jos haettavaa henkilöä ei löydy tietokannasta  
if(empty($haettava)){  
    ...echo "The employee number entered does not exist in the database.";  
}  
//jos henkilö löytyi  
//tulostetaan lomakkeen kenttiin muutettavan henkilön tiedot  
else{  
    ...$muutettavaHenkilo = $haettava[0];  
    ...print_r($haettava);  
    ...echo "  
    ...<form action='henkilotietojenMuutos.php' method='post'>  
    ...<p>  
    ...<label>  
    ...<span class='selite'>Number:</span>  
    ...<input type='text'  
    ...name='henkilonumero'  
    ...size='6' readonly='true'  
    ...value='$muutettavaHenkilo[henkiloNumero]'  
    ...</label>  
    ...</p>
```

Esimerkki 3: tkanta.html (All employees (JSON))

```
xmlhttp = new XMLHttpRequest();  
xmlhttp.onreadystatechange = function(){  
    ...//pyyntö käsitelty, vastaus saatu ja serverin antama status 200  
    ...if (this.readyState == 4 && this.status == 200){  
    ...    console.log(this.responseText);  
    ...    //sijoitetaan vastaus tulosalueeseen  
    ...    tulosalue.textContent=this.responseText;
```