

Aineopintojen harjoitustyö: Tietokantasovellus

Ilkka Koskinen

Helsinki 05.03.2017

HELSINGIN YLIOPISTO
Tietojenkäsittelytieteen laitos

Sisältö

Johdanto	3
Järjestelmän tarkoitus	3
Toteutusympäristö	3
Jatkokehitys	3
Yleiskuva järjestelmästä	4
Käyttötapauskaavio	4
Käyttäjärühmät	4
Käyttötapauskuvaukset	5
Kalareissujen CRUD	5
Vieheiden CRUD	5
Kalapaikkojen CRUD	5
Kalojen CRUD	6
Kalalajien selailu	6
Kalalajien CRUD	6
Käyttäjä	6
Järjestelmän tietosisältö	7
Käsitekaavio	7
Tietokohteiden kuvaukset	7
Player (Käyttäjä)	7
Lure (Viehe)	8
Trip (Kalareissu)	8
Spot (Kalapaikka)	9
Fish (Kala)	9
Species (Kalalaji)	9
Relaatiotietokantakaavio	10
Järjestelmän yleisrakenne	11
Istunnot	11
Javascript	11
Käyttöliittymä ja järjestelmän komponentit	12
Käyttöohje	13
Asennustiedot	14
Kalalajien asentaminen sovellukseen	14
Kalalajikuvien spesifikaatio	14
Taulujen luominen ja testidata	14
Testaus	15
Jatkokehitys	15
Omat kokemukset	15

Johdanto

Järjestelmän tarkoitus

KalaDB toimii kalastajan henkilökohtaisena saalispäiväkirjana. Käyttäjä lisää järjestelmään tiedot kalareissuistaan, kaloistaan, kalapaikoistaan sekä vieheistään, ja voi myöhemmin tarkastella saalishistoriaansa esim. sääolosuhteiden perusteella. Eri hakukriitereillä kyselyjen teko historiatietoihin tuottaa käyttäjälle toivon mukaan vinkkejä uusille kalareissuille.

Toteutusympäristö

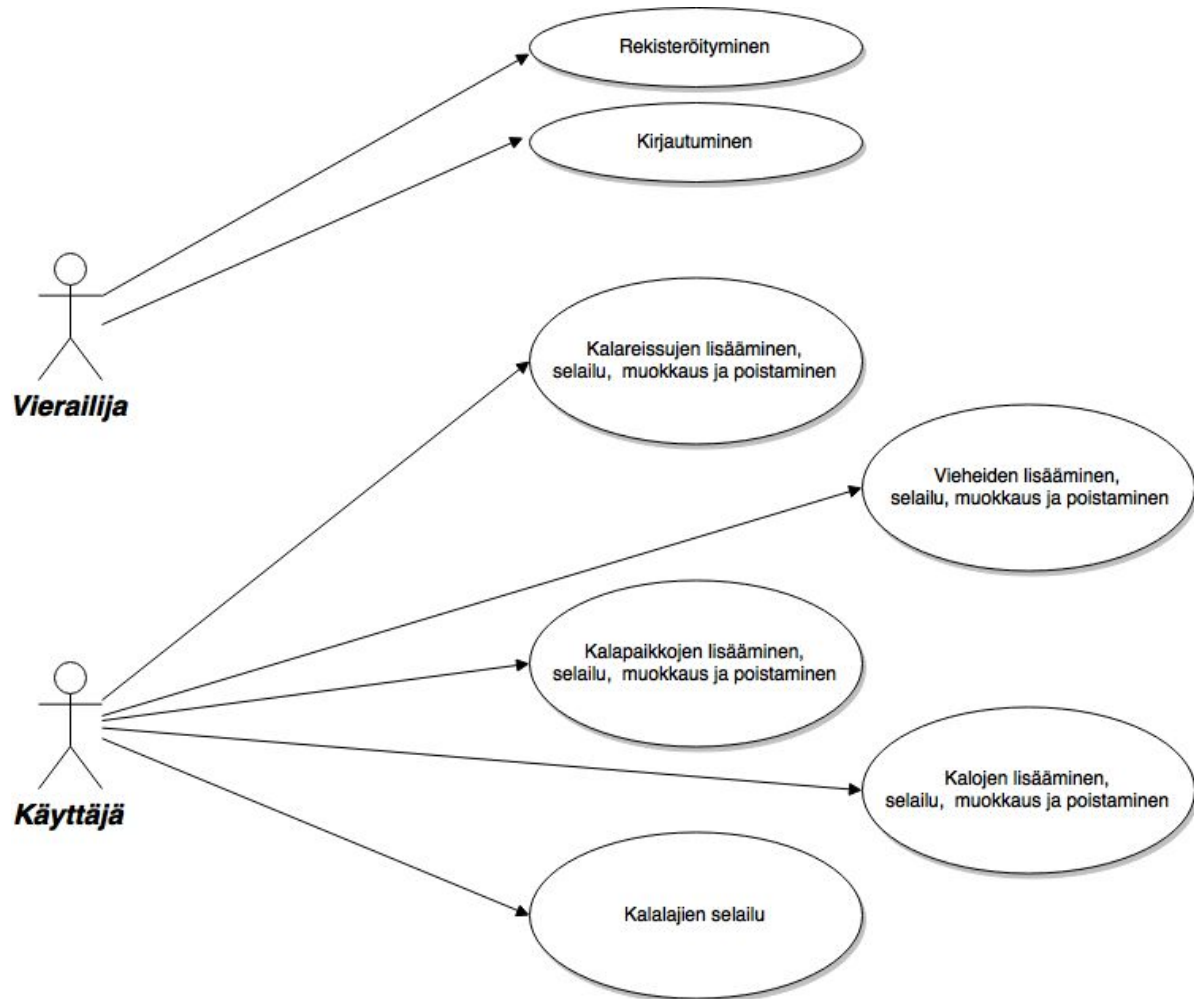
Järjestelmä toteutetaan tietojenkäsittelytieteen laitoksen palvelimelle PHP:lla ja PostgreSQL-tietokannalla.

Jatkokehitys

Harjoitustyönä toteutetaan yllä kuvatun kaltainen henkilökohtainen saalispäiväkirja, jossa jokainen käyttäjä näkee vain omat merkintänsä. Jatkokehityksellä järjestelmästä voisi luoda "some-version", jossa käyttäjät voivat jakaa koko käyttäjäkunnalle haluamiaan osia merkinnöistään.

Yleiskuva järjestelmästä

Käyttötapauskaavio



Käyttäjärühmät

Käyttäjärühmiä on kaksi kappaletta.

Vierailija tarkoittaa ketä tahansa sivustolle saapuvaa internet-käyttäjää. Rekisteröitymisen jälkeen vierailijasta tulee käyttäjä, joka voi kirjautua sisään sivustolle.

Käyttäjä on sivustolle rekisteröitynyt ja sisäänkirjautunut vierailija.

Ylläpitäjä hallinnoi kaikille käyttäjille yhteisiä resursseja, eli alustavasti vain kalalajeja.

Käyttötapauskuvaukset

Kalareissujen CRUD

- käyttäjä voi lisätä uuden kalareissun ja syöttää reissuun liittyvät tiedot:
 - nimi
 - pvm, aloitus- sekä lopetusaika
 - lämpötila, veden lämpötila
 - pilvisyys
 - tuulen nopeus ja suunta
 - kuvaus
- käyttäjä voi tarkastella listaa kalareissuistaan
 - ja valita yhden reissun tarkasteltavaksi
- käyttäjä voi valita kalareissun muokattavaksi
- käyttäjä voi poistaa kalareissun

Vieheiden CRUD

- käyttäjä voi lisätä uuden vieheen ja syöttää vieheen tiedot:
 - nimi
 - tyyppi
 - väritys
 - mahdollinen kuva
- käyttäjä voi tarkastella listaa vieheistään
 - ja valita yhden vieheen tarkasteltavaksi
- käyttäjä voi valita vieheen muokattavaksi
- käyttäjä voi poistaa vieheen

Kalapaikkojen CRUD

- käyttäjä voi lisätä uuden kalapaikan ja syöttää tiedot:
 - nimi
 - mahdollinen sanallinen kuvaus
 - mahdollinen kuva paikasta
- käyttäjä voi tarkastella listaa kalapaikoistaan
 - ja valita yhden kalapaikan tarkasteltavaksi
- käyttäjä voi valita kalapaikan muokattavaksi
- käyttäjä voi poistaa kalapaikan

Kalojen CRUD

- käyttäjä voi lisätä uuden kalan ja syöttää tiedot:
 - kalalaji
 - pyyntipaikka
 - viehe
 - paino
 - pituus
 - pyyntitapa
 - ottisyvyys
 - mahdollinen sanallinen kuvaus
 - mahdollinen kuva
- käyttäjä voi tarkastella listaa kaloistaan
 - ja vaikuttaa listauksen muodostumiskriteereihin, esim. sääolosuhteiden mukaisesti
- käyttäjä voi valita kalan muokattavaksi
- käyttäjä voi poistaa kalan

Kalalajien selailu

- käyttäjä voi tarkastella listaa kalalajeista
 - ja valita yhden kalalajin tarkasteltavaksi

Kalalajien CRUD

Kalalajit seedataan tietokantaan sovelluksen asennuksen yhteydessä.

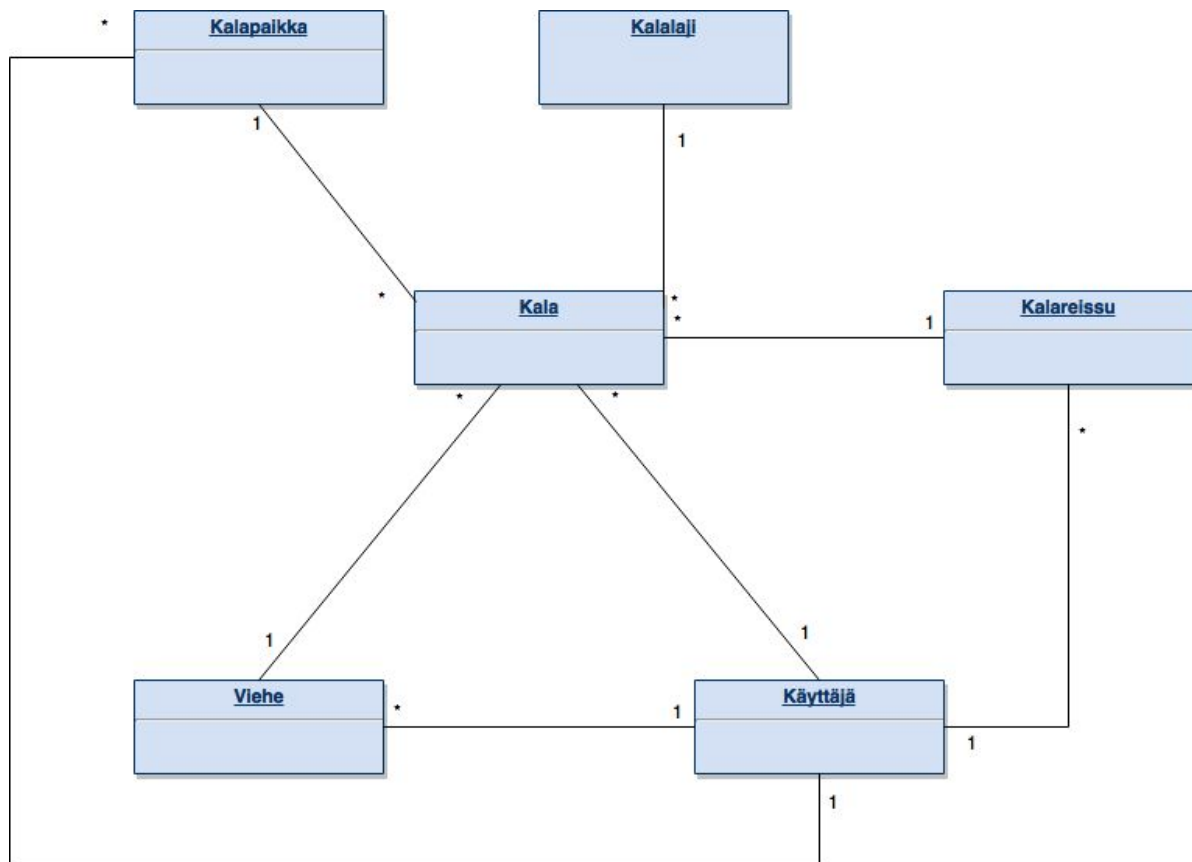
- ~~ylläpitäjä voi lisätä uuden kalalajin~~
- ~~ylläpitäjä voi tarkastella listaa kalalajeista~~
- ~~ylläpitäjä voi valita kalalajin muokattavaksi~~
- ~~ylläpitäjä voi poistaa kalalajin~~

Käyttäjä

- voi rekisteröidä uuden käyttäjätunnuksen
- voi kirjautua ja uloskirjautua
- muuttaa käyttäjänimeänsä

Järjestelmän tietosisältö

Käsitekaavio



Tietokohteiden kuvaukset

Player (Käyttäjä)

Attribuutti	Arvojoukko	Kuvailu
playername	varchar (20), NOT NULL	käyttäjänimi
password	varchar (30), NOT NULL	

Sivustolle rekisteröitynyt käyttäjä. Käyttäjällä voi omistaa useita kalapaikkoja, vieheitä, kalareissuja ja kaloja.

Lure (Viehe)

Attribuutti	Arvojoukko	Kuvailu
lurename	varchar(50), NOT NULL	vieheelle annettu nimi
luretype	varchar(50), NOT NULL	kiinteästä valikoimasta valittu tyyppiluokitus vieheelle
color	varchar(50), NOT NULL	kiinteästä valikoimasta valittu väri vieheelle

Viehe on jonkin rekisteröityneen käyttäjän lisäämä ja kuuluu hänelle.

Trip (Kalareissu)

Attribuutti	Arvojoukko	Kuvailu
tripname	varchar(50), NOT NULL	reissulle annettu nimi
tripday	date, NOT NULL	reissun päivämäärä
start_time	time, NOT NULL	reissun aloitusaika
end_time	time, NOT NULL	reissun lopetusaika
temperature	decimal	ilman lämpötila
water_temperature	decimal	
clouds	varchar(50), NOT NULL	kiinteästä valikoimasta valittu pilvisyyden aste
wind_mps	integer	
wind_direction	varchar(50), NOT NULL	kiinteästä valikoimasta valittu tuulen suunta
description	varchar(1000)	reissulle annettu kuvaus

Yhden käyttäjän syöttämä kalareissu.

Spot (Kalapaikka)

Attribuutti	Arvojoukko	Kuvailu
spotname	varchar(100), NOT NULL	
description	varchar(1000)	
Kuva	???	

Yhden käyttäjän omistama kalapaikka.

Fish (Kala)

Attribuutti	Arvojoukko	Kuvailu
weight	decimal	
length_cm	decimal	
fishing_method	varchar(50), NOT NULL	kiinteästä valikoimasta valittu pyyntitapa kalalle
fish_description	varchar(50)	
Kuva	???	

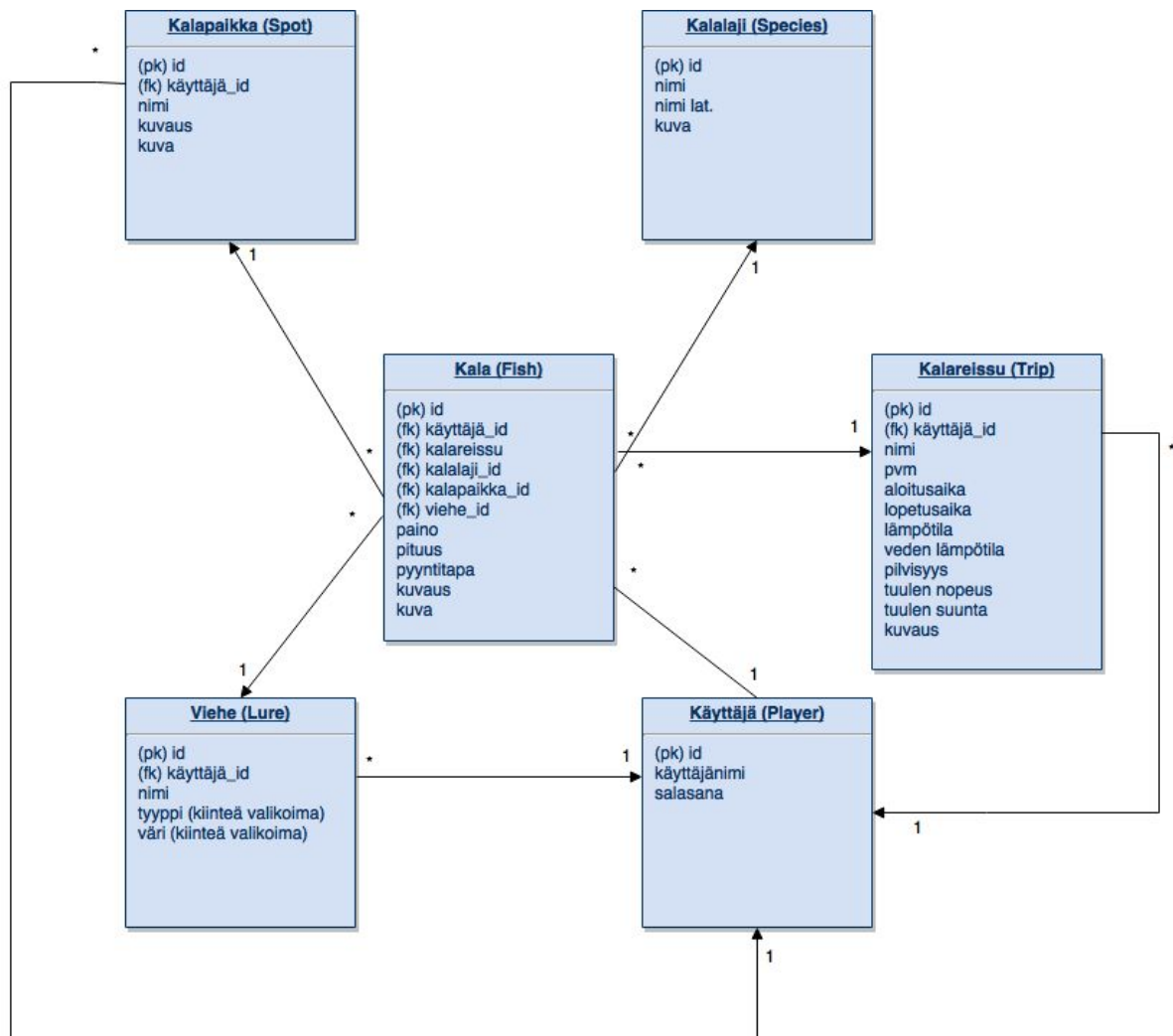
Käyttäjän lisäämä kala. Kalaan liittyy lisäksi viehe, kalapaikka, kalareissu ja kalalaji.

Species (Kalalaji)

Attribuutti	Arvojoukko	Kuvailu
name_fin	varchar(50), NOT NULL	
name_lat	varchar(50)	latinankielinen nimi
Kuva	???	lajikuva

Huom! Kalalajit seedataan tietokantaan perustamisen yhteydessä kiinteästi - katso asennusohjeet.

Relaatiotietokantakaavio

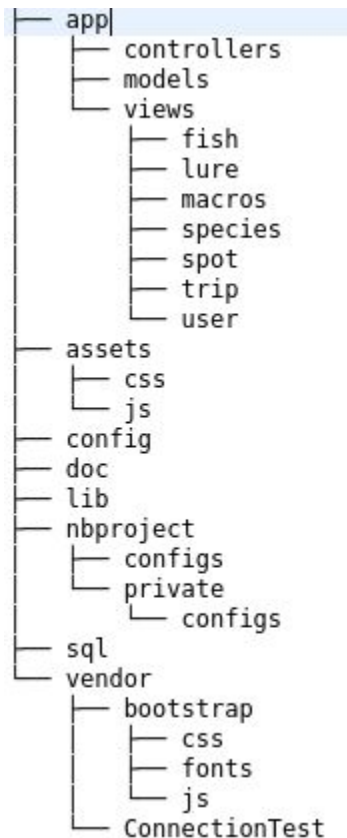


Suluissa toteutuksessa käytettävät englanninkieliset nimet tauluille. Muut osat päivitetään myöhemmin kun käännökset päätetty lopullisiksi.

Kaikki avaimet integer, muut datatyytit tietokohteiden kuvauksien mukaiset -> päivitetään kun selvinnyt PostgreSQL:stä sopivat datatyytit. Kysymyskohtia vielä ainakin enumien perustamiset kiinteään valikoiman attribuuteille sekä kuvien käsittely.

Järjestelmän yleisrakenne

Sovellus noudattaa MVC-mallia: toiminta on hajautettu malleihin, kontrollereihin ja näkymiin, jotka kaikki sijaitsevat omissa kansioissaan. Apukirjastot on sijoitettu hakemistoon lib, CSS-tyylitiedostot kansioon css ja javascriptit kansioon js.



Istunnot

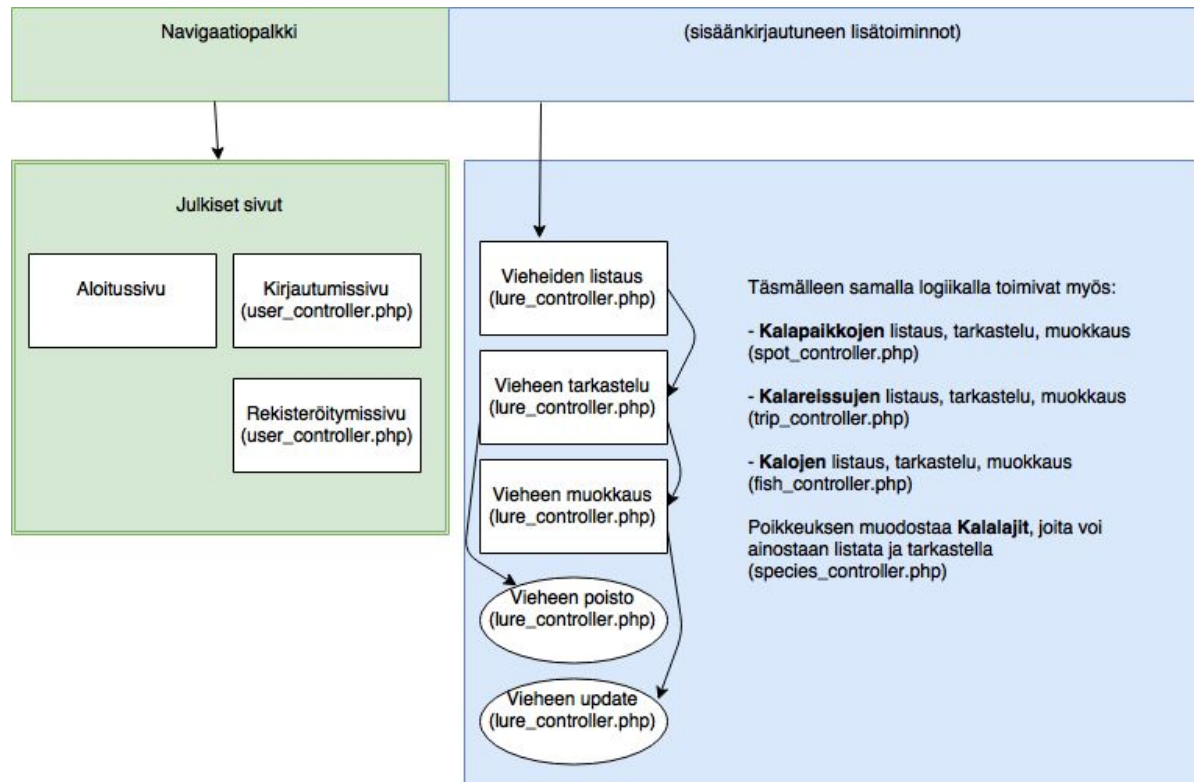
Sovellus hyödyntää kirjautuneen selaimen muistamiseen sessioita. Jos sessiosta ei löydy käyttäjää, näytetään vain kirjautumattomille näkyvät näkymät. Myöskään sovelluksen kontrolleritaso ei hyväksy pyyntöjä ilman sessiosta löytyvää käyttäjää.

Javascript

Käyttäjän kriittisten toimenpiteiden eli erilaisten asioiden poistamisen yhteydessä käyttäjältä pyydetään vahvistus ennen toimenpiteen suoritusta. Tähän hyödynnetään selaimessa javascriptiä.

Sovelluksen erilaiset taulukot ovat käyttäjän järjestettävissä. Tähän hyödynnetään jQuery-pluginia tablesorter (<http://tablesorter.com/docs/>).

Käyttöliittymä ja järjestelmän komponentit



Lisäksi sisäänkirjautuneelle käyttäjälle on toteutettu mahdollisuus

- muuttaa käyttäjänimeä
- poistaa käyttäjätili

Käyttöohje

Harjoitustyö on asennettuna users-palvelimella osoitteessa:

<http://itkoskin.users.cs.helsinki.fi/tsoha>

- login: KumiJ
- password: ColdAsIce

Sisäänkirjautunut käyttäjä voi (itsestään selvillä tavoilla sivustolla liikkumalla)

- katsella, lisätä, muokata ja poistaa vieheitään, kalapaikkojaan, kalareissujaan ja saatuja kaloja.

Yleisiä toimintaohjeita

- sivuston taulukoita voi järjestää otsakkeita klikkaamalla

Asennustiedot

Sovellus asennetaan kopioimalla tiedostot julkiseen kansioon halutulle palvelimelle, jolta löytyy valmius relaatiotietokannan käyttöön. Käytettävä tietokantayhteys määritellään luokkaan DataBaseConfig (config/database.php).

Kalalajien asentaminen sovellukseen

Sovellus tarvitsee toimiakseen kaikkien käyttäjien yhteiskäytössä olevia kalalajeja (Species), jotka tulee seedata tietokantaan asennuksen yhteydessä. Tätä varten löytyy valmiit SQL-lauseet tiedostosta **sql/seed_species.sql**. Kalalajit voi perustaa myös lisäämällä testidatan tiedostosta **sql/add_test_data.sql**.

Kalalajin tiedot tietokannassa koostuvat suomenkielisestä nimestä (**name_fin**) sekä latinankielisestä nimestä (**name_lat**). Jokaista lisättyä kalalajia kohti **on lisättävä sovellukseen myös png-muotoinen kuva**, jonka nimenä käytetään **name_fin** kenttään syötettyä arvoa. Kuvat tallennetaan sovelluksen kansioon **Assets/pics**.

Kalalajikuvien spesifikaatio

- nimeäminen kentän **name_fin** mukaisesti
- tiedostomuoto **png**
- suositeltu koko **500 x 500 px, alpha transparency**
- sijainti **Assets/pics**

Taulujen luominen ja testidata

Kansion **sql** alta löytyy tiedostot

- create_tables.sql
- drop_tables.sql
- add_test_data.sql
- seed_species.sql

joista löytyy valmiit sql-lauseet tietokantataulujen pystytykseen, poistamiseen ja testidatan lisäämiseen / kalalajien seedaamiseen. **Huom!** Testidata sisältää siis myös jo kokonaisuudessaan **seed_species.sql** tiedoston lauseet, ja kyseessä on vaihtoehtoiset tavat alustaa sovellus: pelkillä kalalajeilla, tai myös valmiilla testikäyttäjällä ja sisällöllä käyttäjälle.

Testaus

Sovellusta on testattu manuaalisesti selaimesta käymällä läpi kaikki toiminnot sekä hyväksyttävillä, että virheellisillä syötteillä. Kaikki ominaisuudet toimivat, ja bugeja ei pitäisi olla. Kuitenkin vain oikeat yksikkötestit ja automatisoitu järjestelmätestaus vasta takaisivat tämän.

Jatkokehitys

Sovellus toimii tällä toteutuksella vain käyttäjän henkilökohtaisena kalapäiväkirjana. Käyttäjien keskinäinen sekä sovelluksen ulkopuolelle tehtävä tietojen jakaminen olisi seuraava luonnollinen askel. Tämä jätettiin harjoitustyön ulkopuolelle, koska aihepiirin vuoksi toiminnot lähtevät hyvin nopeasti polveilemaan monimutkaisiksi. Tästä muutamia esimerkin kaltaisia ajatuksia alla:

- kalapaikat tarvitsevat koordinaatit ja algoritmin jolla tunnistaa useat kalapaikat "lähistöltä" samaksi paikaksi
- käyttäjien tulee kyetä monipuolisesti päättää julkisuusasetuksista - jos jokin kalasaalis halutaankin julkaista kaikille, tarkka kalapaikka voidaan haluta pitää kuitenkin yksityisenä tietona

Omat kokemukset

Harjoitustyö oli oikein opettavainen ja mukava projekti. Erinomainen materiaali ohjasi kädestä pitäen eteenpäin, ja vaikka PHP oli itselleni uusi tuttavuus, ei ongelmia tullut vastaan. MVC-mallin takominen takaraivoon oli erinomainen, ja sitä olin toivonutkin. Tietokantojen perusteet -kurssilla nämä jäivät vielä vähän hämäräksi.