

1. Arkkitehtuurin yleiskatsaus

Järjestelmän tärkein komponentti on Ruby on Rails -web-sovellus, joka keskustelee kaikkien muiden järjestelmän komponenttien kanssa. Web-sovelluksen arkkitehtuurin perustana toimii MVC-arkkitehtuuri.

Sovellus koostuu kolmesta päämallista: lomakkeista (VisitationForm), kuvista (Image) ja linnuista (Bird). Kaiken selkärangana toimivat lomakkeet, joihin kuvat ja linnut liittyvät. Järjestelmän tärkeimmät toiminnot ovat lomakkeiden luonti, muokkaaminen, lähettäminen ja admin-käyttäjällä lomakkeiden hyväksyminen, sekä lintujen ja kuvien liittäminen näihin lomakkeisiin.

2. Järjestelmän tarkoitus

Järjestelmään on tarkoitus tallentaa kuvia merikotkista ja niiden pesistä, sekä näihin kuviin liittyviä metatietoja. Järjestelmän käyttäjät jakautuvat karkeasti kahteen ryhmään: kuvaajiin/ rengastajiin jotka lisäävät järjestelmään lomakkeita ja lähettävät ne hyväksyttäviksi, sekä admin-käyttäjiin jotka tarkastavat ja hyväksyvät lähetetyt lomakkeet.

2.1. Toiminnalliset vaatimukset

2.1.1. Toiminnalliset vaatimukset rengastajan kannalta

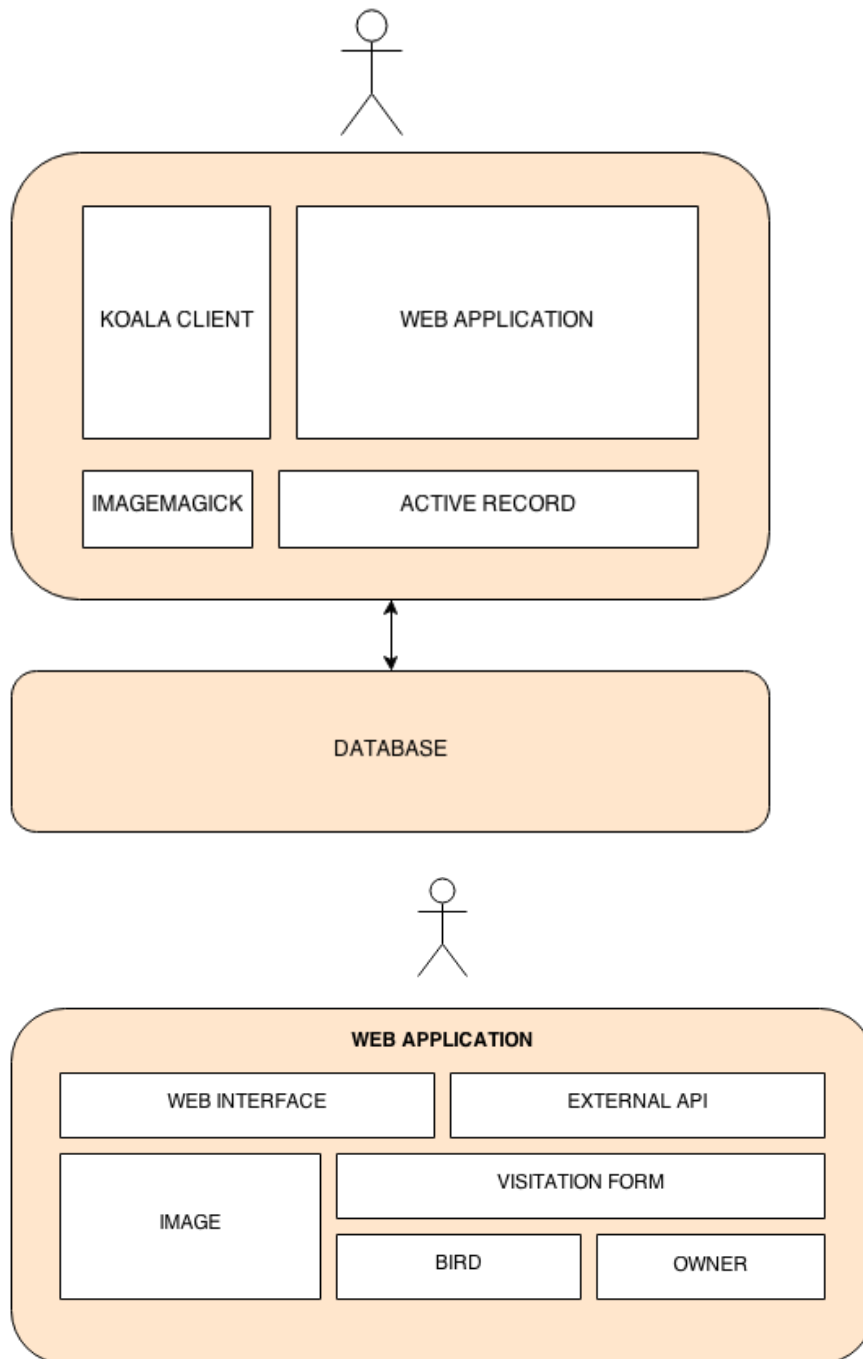
- Käyttäjän tulee pystyä luomaan uusia lomakkeita sekä poistamaan itse luomiaan lomakkeita
- Käyttäjän tulee pystyä tarkastelemaan itse lähettämiään lomakkeita
- Käyttäjän tulee pystyä liittämään lomakkeeseen kuvauskäyntikertaa ja kuvattua pesää identifioivat tiedot
- Käyttäjän tulee voida liittää lomakkeeseen kuvia pesän linnuista, itse pesästä, ja pesän ympäristöstä asianmukaisesti kategorioihin jaoteltuna, sekä poistaa lisäämiään kuvia
- Käyttäjän tulee voida liittää luomiinsa lomakkeisiin muita omistajia, jotka tällöin voivat myös nähdä ja muokata lomaketta
- Käyttäjän tulee pystyä lähettämään täyttämänsä lomake, sekä virheen havaittuaan peruuttamaan lomakkeen lähetys

2.1.2. Toiminnalliset vaatimukset admin-käyttäjän kannalta

- Käyttäjän tulee voida nähdä kaikki järjestelmään tallennetut lomakkeet
- Käyttäjän tulee pystyä hyväksymään tai hylkäämään lähetettyjä lomakkeita

3. Järjestelmän rakenne

3.1. Yleiskatsaus



3.2. Komponentit

3.2.1. ImageMagick

Käytetään esikatselukuvien luontiin.

- Vastaanottaa kuvan web-sovellukselta, pienentää kuvaa ja palauttaa esikatselukuvan.

3.2.2. Tietokanta

Käytetään lomakkeiden ja kuvien tallentamiseen.

- Vastaanottaa tallennettavan datan web-sovellukselta.

3.2.3. Koala-client

Käytetään käyttäjän autentikaatioon.

- Välittää käyttäjän kirjautumistiedot web-sovellukselle.

3.2.4. Web-sovellus

Ruby on Rails -sovellus jota käytetään lomakkeiden ja kuvien hallintaan.

- Kutsuu ImageMagickia käyttäjän ladatessa kuvia järjestelmään.
- Suorittaa tietokantapyyntöjä kun käyttäjä luo, muokkaa tai hakee lomakkeita ja kuvia.

3.2.4.1. Web-käyttöliittymä

Käytetään lomakkeiden ja kuvien luomiseen, muokkaamiseen ja poistamiseen.

3.2.4.2. Ulkoinen API

Käytetään lomakkeiden hakemiseen pesän ID:n, lintulajin ja päivämäärän perusteella.

3.2.4.3. Image-malli

Image-malliin tallennetaan itse kuvadata, sekä kuvaan liittyvät metatiedot. Kuvat liittyvät aina johonkin lomakkeeseen.

3.2.4.4. VisitationForm-malli

Sovelluksen tärkein malli. Lomakkeelle tallennetaan kuvattuun pesään ja kuvaajaan liittyviä tietoja, ja kaikki kuvat (Image) ja linnut (Bird) liittyvät johonkin lomakkeeseen.

3.2.4.5. Bird-malli

Bird-malli kuvaa yhtä pesän luona kuvattua aikuista lintua. Aikuiseen lintuun liittyviä erityisiä tietoja ovat esimerkiksi sen rengastustiedot, joita ei poikaslinnuilla ole.

3.2.4.6. Owner-malli

Jokaiseen tietokantaan tallennettuun lomakkeeseen liittyy yksi tai useampi omistaja (Owner). Admin-käyttäjiä lukuunottamatta käyttäjät voivat nähdä vain ne lomakkeet, joihin heidät on merkitty omistajiksi.

4. Järjestelmän käyttö

Järjestelmää voi käyttää kahdella tavalla: web-käyttöliittymällä kuten mitä tahansa verkkosivua, tai ulkoisen API:n kautta, jonka avulla järjestelmästä voi hakea lomakkeita ja kuvia.

4.1. Käyttöskenaariot

4.1.1. Käyttäjä luo lomakkeen ja lisää siihen kuvan

Käyttäjä kirjautuu järjestelmään ja Koala Client antaa verkkokäyttöliittymälle tiedon onnistuneesta kirjautumisesta. Kirjaututtuaan käyttäjä klikkaa käyttöliittymän "Luo uusi lomake"-linkkiä jolloin järjestelmä luo uuden lomakkeen ja käyttäjä pääsee muokkaamaan sen tietoja. Kun käyttäjä liittää lomakkeeseen uuden kuvan, lähetetään se palvelimelle asynkronisesti, joten käyttäjä voi samalla jatkaa lomakkeen täyttämistä ja kuvien lisäämistä. Palvelimella web-sovellus kutsuu ImageMagickia joka generoi ladatusta kuvasta pienemmän esikatselukuvan.

4.1.2. Käyttäjä lähettää täytetyn lomakkeen tarkistettavaksi

Kun käyttäjä on täyttänyt lomakkeelle tarvittavat tiedot, hän klikkaa "Lähetä"-nappia, jolloin selain lähettää lomakkeen sisällön palvelimelle. Palvelimella web-sovellus tarkistaa että lomakkeelle on täytetty vaaditut tiedot ja että nämä tiedot ovat oikeellisia. Mikäli lomake on täytetty virheellisesti, palautetaan käyttäjä muokkaus-näkymään. Mikäli lomake on täytetty oikein, tallennetaan tiedot tietokantaan.

4.1.3. Admin-käyttäjä tarkistaa ja hyväksyy lähetetyn lomakkeen

Admin-käyttäjä näkee "Lähetetyt"-kansiossa kaikkien käyttäjien lähettämät lomakkeet. Hän käy läpi lomakkeita yksi kerrallaan ja tarkistaa, että lomakkeen tiedot on täytetty oikein. Mikäli lomake on täytetty virheellisesti, admin voi joko palauttaa lomakkeen sen lähettäneelle käyttäjälle klikkaamalla "Peruuta lähetys"-nappia tai muokata itse tarvittavat tiedot oikeellisiksi. Kun lomake on täytetty oikein, admin klikkaa "Hyväksy"-nappia jolloin lomake merkitään arkistoiduksi eikä sitä voi enää muokata.

4.1.4. Käyttäjä hakee lomakkeita ulkoisen API:n avulla

Käyttäjä voi ulkoisen API:n kautta hakea lomakkeita haluamillaan hakuehdoilla antamalla ne osoitteen parametreina. API palauttaa hakuehtoja vastaavat lomakkeet JSON-objekteina.