

lindeb

mau\Lu Link Database

Tietokantasovellus-harjoitustyö / Tulir Asokan

Sisällys

1 Johdanto	3
1.1 Tavoitteet ja toiminnallisuus	3
1.2 Teknologia	3
1.3 Toteutus- ja toimintaympäristö	4
2 Yleiskuva järjestelmästä	5
2.1 Käyttötapauskaavio	5
2.2 Käyttäjäryhmät	5
2.3 Käyttötapauskuvaukset	5
2.3.1 Jokaihmissen käyttötapaukset	5
2.3.2 Käyttäjän käyttötapaukset	6
3 Järjestelmän tietosisältö	7
3.1 Tietokohdekuvaukset	7
3.2.1 User	7
3.2.2 AuthToken	8
3.2.3 Link	8
3.2.4 Tag	8
4 Relaatiotietokantakaavio	9
5 Järjestelmän yleisrakenne	10
6 Käyttöliittymä ja sen komponentit	11
7 Asennustiedot	12
7.1 Kääntäminen	12
7.2 Konfiguraatio	12
8 Käynnistys- ja käyttöohje	13

1 Johdanto

1.1 Tavoitteet ja toiminnallisuus

Järjestelmän tavoitteena on sallia helppo linkkien tallentaminen ja tallennettujen linkkien selaaminen ja etsiminen.

Järjestelmän perustoiminnallisuuteen kuuluu linkkien tallennus ja tallennettujen linkkien hallinta, linkkien tagaaminen ja niiden selaaminen. Linkkien tallennuksen on onnistuttava tallennettavalta sivulta yhdellä napinpainalluksella, esimerkiksi kirjanmerkin tai selainlisäosan avulla.

Järjestelmä vaatii kirjautumisen. Käyttäjien lisäämä tieto pysyy yksityisenä vain käyttäjän nähtävänä.

Hakutoiminnon on kyettävä hakemaan tietoja myös sivun sisällön perusteella, eikä vain perustietojen kuten otsikon, sivuston ja tagien perusteella. Tämän lisäksi linkkejä on voitava järjestää ja suodattaa perustietojen perusteella.

Järjestelmään ei tule lisätä mitään sisäisiä sosiaalisia ominaisuuksia. Kaikki ihmisten välinen interaktio tapahtuu muissa palveluissa. Jos aikaa riittää, järjestelmään voidaan lisätä mahdollisuus jakaa linkkejä muihin palveluihin, mutta pelkkä linkin kopiointimahdollisuus riittää.

Järjestelmä on tarkoitus tulevaisuudessa integroida [mau\Lu](#)-palveluun.

1.2 Teknologia

Projektin palvelinpuoli toteutetaan [Go](#)-ohjelmointikielellä. Palvelimessa on REST-tyylinen HTTP/JSON-ohjelmointirajapinta. Itse ohjelman palvelimen lisäksi järjestelmä

hyödyntää myös [Elasticsearch](#)-palvelinta monimutkaisempien hakujen sallimiseksi. Tietokantana käytetään [MariaDB](#)-tietokannanhallintajärjestelmää.

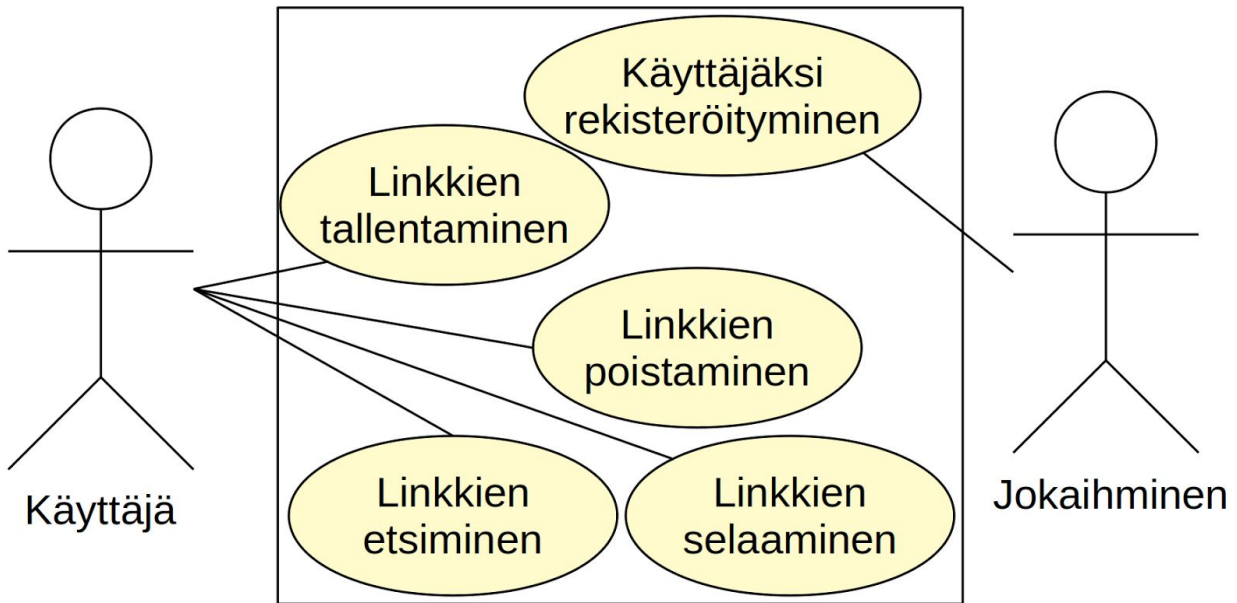
Käyttöliittymä toteutetaan web-ohjelmana [React](#)illa ja modernilla JavaScriptillä. Käyttöliittymän toimivutta vanhoilla selaimilla ei taata. Firefoxin uusin versio on suositeltu, mutta järjestelmä testataan myös Chromen uusimmalla versiolla.

1.3 Toteutus- ja toimintaympäristö

Projektia kehitetään JetBrainsin [GoLand](#)-kehitysympäristöllä. Sovellus asennetaan Maunium-palvelinympäristöön ja sen löytää osoitteesta lindeb.mau.lu.

2 Yleiskuva järjestelmästä

2.1 Käyttötapauskaavio



2.2 Käyttäjäryhmät

Jokaihmisellä tarkoitetaan ketä tahansa, joka internetin välityksellä tulee katsomaan lindebin www-sivuja.

Käyttäjällä tarkoitetaan järjestelmään rekisteröityneitä käyttäjiä.

2.3 Käyttötapauskuvaukset

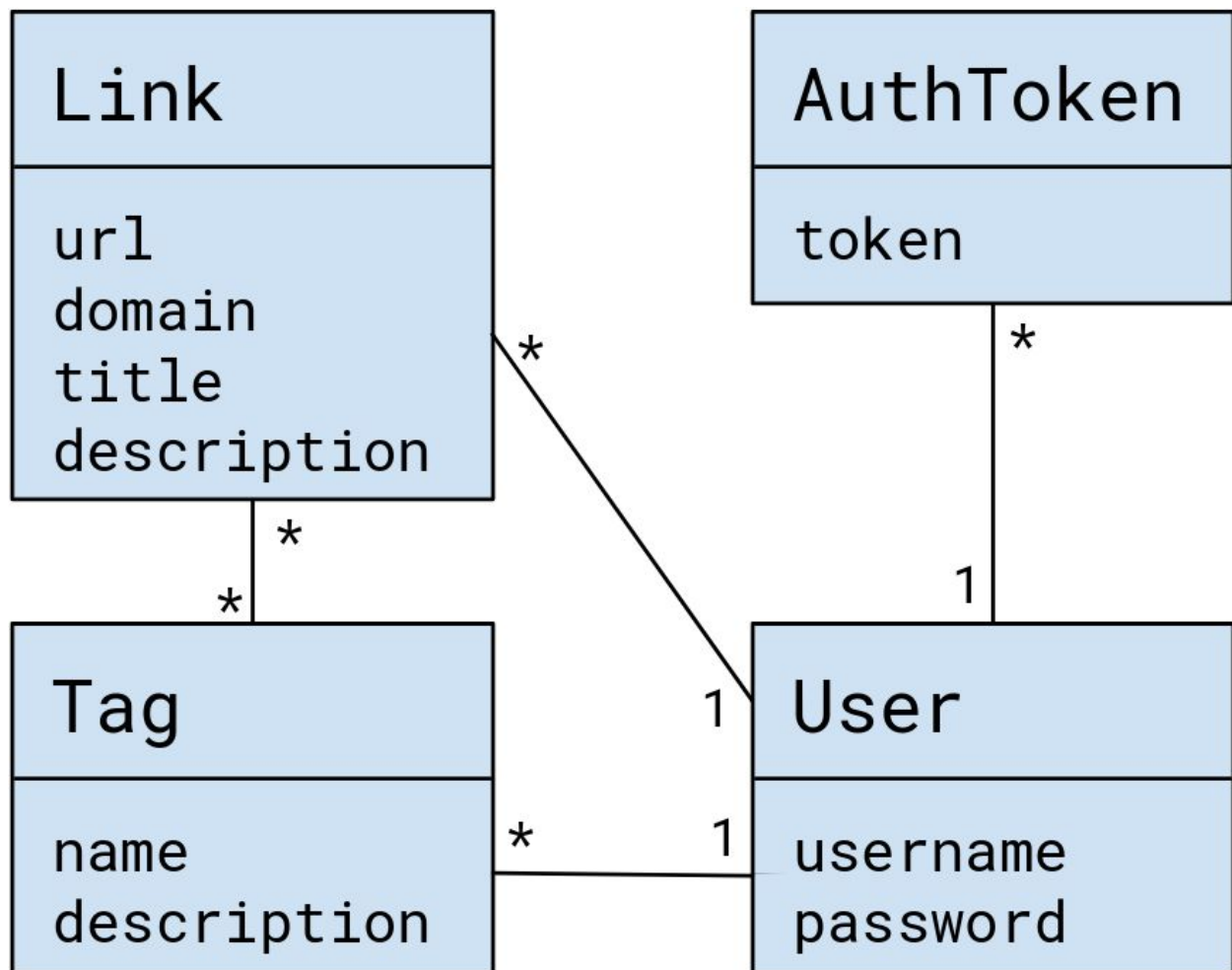
2.3.1 Jokaihmissen käyttötapaukset

- Rekisteröityminen: kuka tahansa voi rekisteröityä palvelun käyttäjäksi
- Muita käyttötapauksia: kirjautuminen

2.3.2 Käyttäjän käyttötapaukset

- Linkkien tallentaminen: käyttäjä voi tallentaa mielenkiintoisia linkkejä myöhemmin luettavaksi
- Linkkien etsiminen: käyttäjä voi etsiä linkkejä sivun sisällön perusteella
- Linkkien selaaminen: käyttäjä voi selata, järjestellä ja suodattaa tallennettuja linkkejä perustietojen perusteella
- Muita käyttötapauksia: linkkien poistaminen

3 Järjestelmän tietosisältö



3.1 Tietokohdekuvaukset

3.2.1 User

Rekisteröitynyt käyttäjä.

Attribuutti	Tyyppi	Kuvaus
username	Merkkijono	Käyttäjän asettama käyttäjänimi.
password	Merkkijono	bcrypt-hajautusfunktiolla suojattu salasana.

3.2.2 AuthToken

Käyttäjän kirjautumiskohtainen todennusmerkki.

Attribuutti	Tyyppi	Kuvaus
token	Merkkijono	SHA256-hajautusfunktiolla suojattu 64-merkkinen satunnaisgeneroitu merkkijono.

3.2.3 Link

Käyttäjän tallentama linkki, ja siihen liittyviä tietoja. Linkillä voi olla useita tageja, mutta vain yksi omistaja. Yhden omistajan rajoitus johtuu siitä, että omistaja voi vapaasti muokata esimerkiksi linkille tallennettua otsikkoa ja sen tageja.

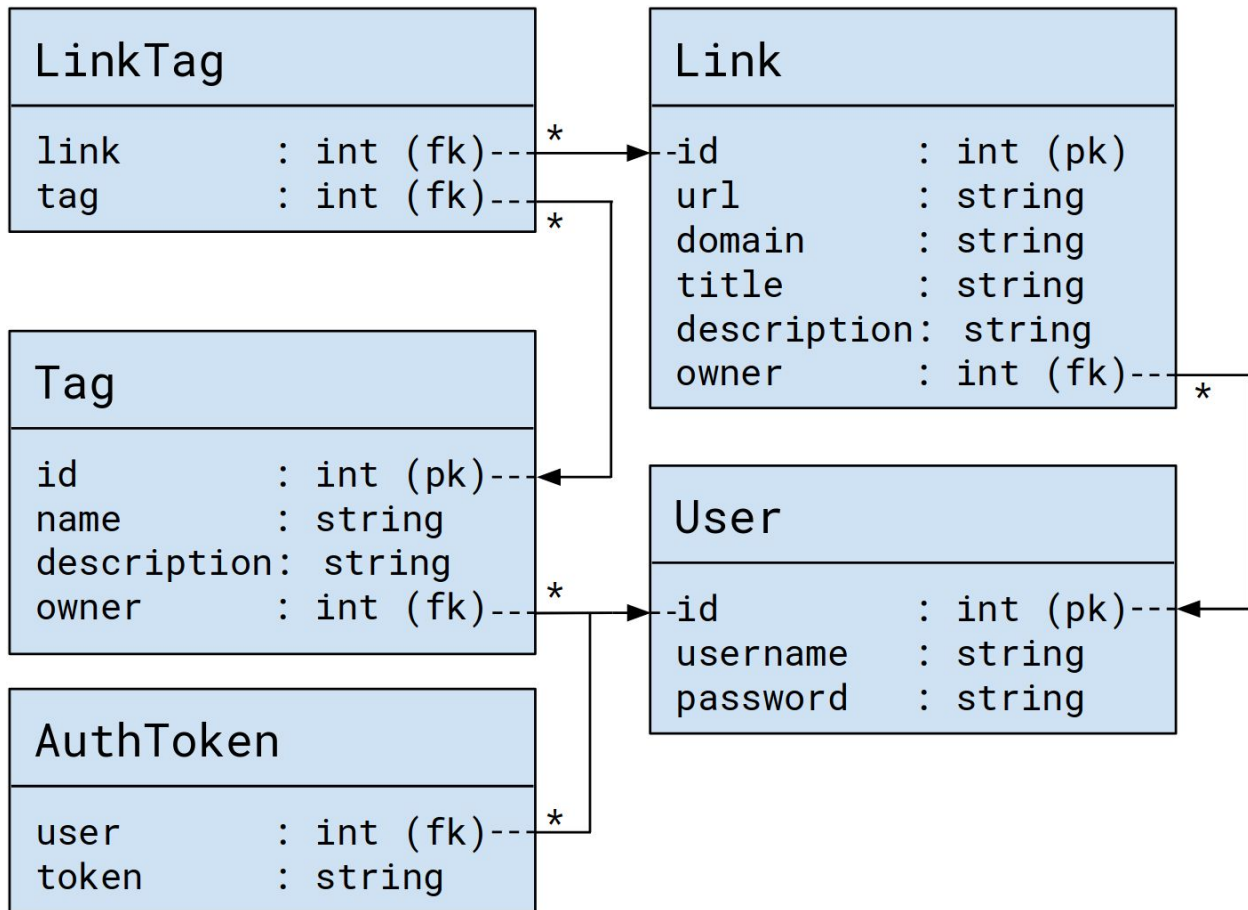
Attribuutti	Tyyppi	Kuvaus
url	Merkkijono	Koko linkki mukaanlukien protokolla ja kaikki parametrit.
domain	Merkkijono	Linkin domain-osa, joka näytetään käyttäjälle listauksissa.
title	Merkkijono	Linkin osoittaman sivun otsikko.
description	Merkkijono	Lyhyt kuvaus linkin osoittamasta sivusta.

3.2.4 Tag

Käyttäjäkohtainen tagi ja lyhyt kuvaus sille. Useampi linkki voidaan tagata samalla tagilla. Tagit ovat käyttäjäkohtaisia, sillä käyttäjä voi vapaasti määritellä tagin kuvauksen.

Attribuutti	Tyyppi	Kuvaus
name	Merkkijono	Tagin nimi, vain pieniä kirjaimia.
description	Merkkijono	Lyhyt kuvaus tagista.

4 Relaatiotietokantakaavio



5 Järjestelmän yleisrakenne

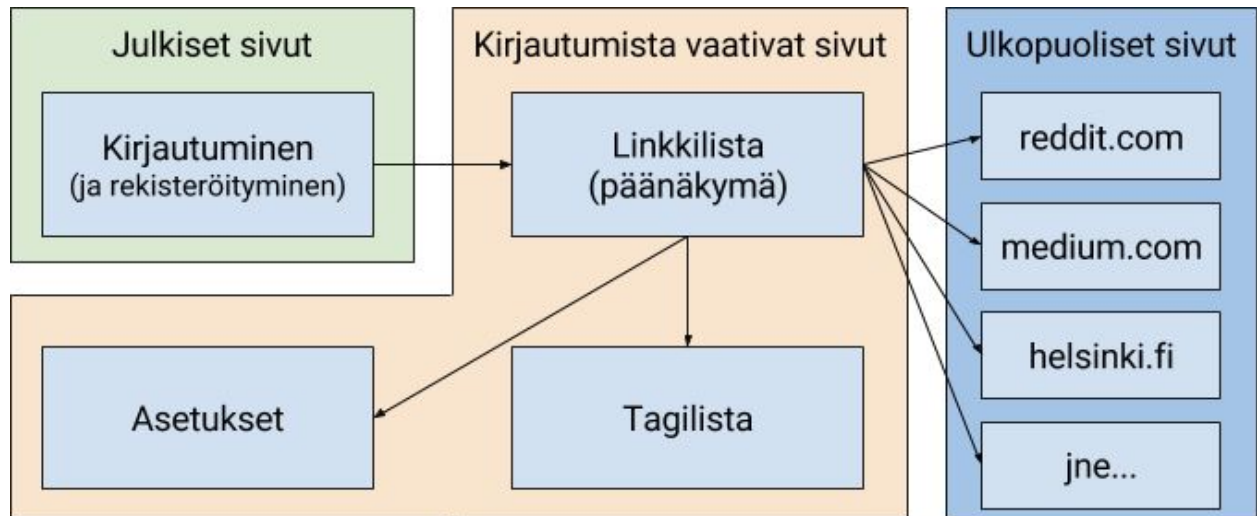
Tietokantasovellus on jaettu Reactilla toteutettuun frontendiin ja Go:lla toteutettuun backendiin.

Backend on jaettu kolmeen osaan: tietokantaa hallitsevaan "db"-pakettiin, HTTP/JSON-ohjelmointirajapinnan toteuttavaan "api"-pakettiin ja pääpakettiin, jossa on palvelimen konfiguraation lukeminen ja käynnistyksen hoitaminen. Kirjastoja hallinnoidaan Go:n virallisella, mutta kokeellisella [dep](#)-työkalulla, joka sijoittaa kirjastot vendor-hakemistoon. Konfiguraatio laitetaan tiedostoon nimeltä `config.yaml`. Konfiguraatiotiedosto kannattaa luoda kopioimalla `example-config.yaml`-tiedosto.

Tämä osio on keskeneräinen: siitä puuttuu frontendin rakenne.

6 Käyttöliittymä ja sen komponentit

Tämä osio on keskeneräinen.



7 Asennustiedot

7.1 Kääntäminen

Tämän vaiheen voi ohittaa, jos lataa Githubista valmiiksi käännetyt paketin [Releases](#)-osiosta.

Palvelimen itse kääntämiseen on oltava Go-työkalupakki asennettuna ja oikein konfiguroituna. Tämän jälkeen projektin voi ladata ja kompiloida ajamalla komennon

```
go get maunium.net/go/lindeb
```

jonka jälkeen projekti löytyy kansiota \$GOPATH/src/maunium.net/go/lindeb ja kompiloitu palvelinohjelma löytyy tiedostosta \$GOPATH/bin/lindeb.

Frontendin kääntäminen (*sitten, kun frontendin tekeminen on aloitettu*) tapahtuu ajamalla komento

```
npm run build
```

frontend-kansiossa. Kompiloinnin jälkeen staattiset tiedostot löytyvät frontendin kansiota dist.

7.2 Konfiguraatio

Konfiguraatiopohja löytyy tiedostosta `example-config.yaml`, josta sen voi kopioida vakiokonfiguraatitiedostoon `config.yaml`. Konfiguraatitiedoston polkua voi halutessaan muuttaa palvelinohjelman `--config-parametrillä`. Konfiguraatiopohjan pitäisi toimia tietokantakonfiguraatiota lukuunottamatta suoraan ainakin kehitysympäristössä. Konfiguraatiopohjassa on kommentit, jotka kertovat, mitä kenttiin pitäisi laittaa.

Jos et aktivoi palvelimen frontend-ominaisuutta, joudut käyttämään esimerkiksi Nginx-palvelinta backendin ja käyttäjän välissä frontendin jakamiseen.

8 Käynnistys- ja käyttöohje

Kun palvelin on asennettu ja konfiguroitu oikein, sen voi käynnistää yksinkertaisesti ajamalla palvelinohjelman. Yleensä ajaminen tapahtuu komennolla `lindeb` tai `./lindeb`.

Palvelimen käynnistyttyä voit avata sen selaimella. Virallinen instanssi löytyy osoitteesta lindeb.mau.lu. Mitään vakiotunnuksia ei ole: luo omat tunnukset kirjautumissivulla.