

Unified webcomic/site subscriber

Popis aplikácie

Aplikácia je určená pre tých, ktorí radi čítajú napr. webové komiksy od viacerých tvorcov, a chcú by dostávať notifikácie o novom obsahu viacerých takýchto zdrojov spoločne.

Zdrojom je teda napr. stránka webkomiksu, pre ktorú má aplikácia nastavené sledovanie, v najjednoduchšej forme pravidelným automatizovaným načítaním danej stránky a porovnaním identifikačných znakov.

Užívatelia si v aplikácii môžu nastaviť, pre ktoré zdroje chcú alebo nechcú dostávať notifikácie, a akým štýlom ich chcú mať doručené, primárnym typom budú denné a týždenné maily.

Moderátori môžu pridávať vlastné posty, hlavne textové a dotazníkové, určené na prieskum medzi užívateľmi ohľadom žiadaných nových zdrojov. Ak sú im udelené príslušné práva, môžu nastavovať parametre zdrojov alebo vytvárať nové zdroje. Môžu tiež vytvárať skupiny zdrojov, pre pohodlnejšie odoberanie podobných vecí.

Administrátor môže udeľovať a odoberať práva užívateľom a moderátorom, automaticky má tiež maximálne možné práva moderátora.

Owner je špeciálny typ administrátora, ktorý má navyše právo udeľovať a odoberať administrátorské práva iným užívateľom. Právo ownera je udeľované mimo aplikácie samotnej, správcom systému.

Typy používateľov, právomoci

Typy používateľov sú nasledovné:

- guest
- user
- moderátor
- admin
- owner

Pre užívateľov ktorý na to nemajú explicitne práva, sú zdroje viditeľné len ako názov, popis a identifikácia posledného updatu.

Guest má len právo pozeráť si existujúce zdroje a posty od moderátorov. Nedokáže reagovať na dotazníky ani komentovať. Môže sa tiež zaregistrovať alebo prihlásiť, čím sa stane vyšším typom používateľa

User má okrem guest práv aj právo označiť pre zdroje že ich chce odoberať, prípadne že ich nechce odoberať ak ide o jednoduché (tzn. nie skupina) zdroje. Tak isto môže označiť, od ktorých moderátorov chce odoberať posty. Môže si tiež nastaviť preferencie notifikácií: či chce dostávať denné maily, či týždenné a ak áno ktorý deň v týždni. Na stránke si tiež môže pozrieť zoznam aktuálnych updatov pre jednoduché zdroje ktoré odoberá a označiť ich ako videné, na základe čoho budú vynechané z plánovaných mailov. Užívateľ tiež môže hlasovať v dotazníkoch a komentovať posty.

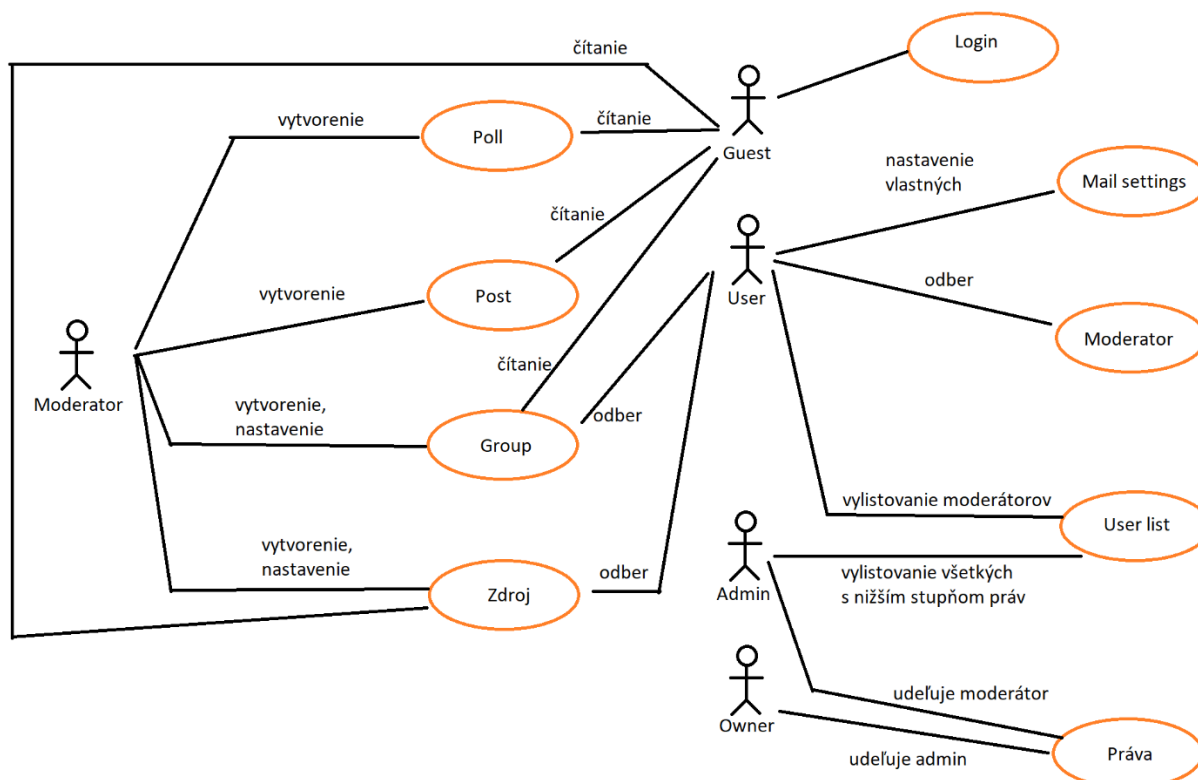
Moderátor môže mať okrem user práv udelené 4 dodatočné práva: vytváranie postov, úprava parametrov jednoduchého zdroja, vytváranie jednoduchých zdrojov, a management zložených zdrojov. Právo vytvárať zdroje umožňuje moderátorovi vytvoriť a zverejniť post. Právo nastavovať parametre zdroja umožňuje nastavovať parametre ako napríklad časový interval pollingu, kritérium pre detekciu nového obsahu a podobne. Právo vytvárať zdroje automaticky dáva aj právo úpravy zdrojov. Umožňuje pridanie nového zdroja a jeho nastavenie. Právo managementu zložených zdrojov umožňuje vytváranie a úpravu zložených zdrojov, t.j. pridávanie alebo uberanie z množiny zdrojov, ktoré ich tvoria.

Admin má automaticky maximálne možné práva moderátora. Navyše má právo udeľovať a odoberať moderátorské práva userom a moderátorom. Za týmto účelom si vie pozerať zoznam všetkých userov a moderátorov. Z usera môže spraviť moderátora tým, že mu pridá niektoré právo, a naopak odobratím všetkých práv.

Môže tiež iniciovať čistenie histórie pre jednoduchý zdroj, kde sa vymažú príliš staré (teda už nezaujímavé) updaty, pričom minimálne najnovší update zostane.

Owner má všetky práva administrátora, navyše má právo udeľovať a odoberať administrátorské práva. Vidí teda aj zoznam všetkých administrátorov. Ownerov potenciálne môže byť viacero, ale aplikácia zámerne neposkytuje možnosť pridať alebo uberať ownerov inak ako zásahom do databázy, ktorý vykoná správca systému (aj keď potenciálne na to bude tool, ktorý vygeneruje potrebné SQL).

Use case diagram



Používateľ vyššej úrovne vždy tiež môže spraviť všetko to, čo môže používateľ nižšej.

Databáza, tabuľky

Databáza bude typu SQL databázy, podľa možnosti odabstrahovaná pomocou knižnice Hibernate. Pre účely testov by mala byť použiteľná in-memory databáza, pre reálnu prevádzku asi MySQL alebo Postgres, ideálne by aplikácia mala byť nezávislá od DB, pre vývoj asi Postgres.

Tam, kde nespomínam explicitne prítomnosť id, alebo unikátne použitie foreign id ma tabuľka id pre každý záznam.

Tabuľka užívateľov bude obsahovať uid, meno a nejak vhodne spracované heslo (vhodná verzia salted hash), potom flagy pre owner, admin a moderátorské práva ako vhodný malý číselný typ.

Tabuľka mail settings bude pre užívateľa (uid) obsahovať 2 id zloženého zdroja (reprezentuje užívateľov zoznam odberov a ignorov) a flagy pre zapnutie denných a týždenných mailov, deň v týždni a timestampy posledných poslaných mailov.

Tabuľka postov obsahuje id postu, uid tvorca, názov a text postu a timestamp vytvorenia.

Tabuľka pollov obsahuje id postu a text možnosti. Post je poll ak ma aspoň 1.

Tabuľka hlasov obsahuje id možnosti a uid.

Tabuľka komentárov obsahuje uid, id postu a text.

Tabuľka jednoduchých zdrojov bude obsahovať id, názov a popis.

Tabuľka parametrov jednoduchého zdroja bude pre id zdroja mať id položky/atribútu a hodnotu uloženú ako reťazec.

Tabuľky pre zložené zdroje obsahuje id, názov, popis, a flag či ide o vlastný zoznam niektorého usera.

Tabuľka grafu zložených zdrojov obsahuje 2 id zloženého zdroja, parent a child; pravdepodobne k nej bude aj view realizujúci reflexívno-tranzitívny uzáver tohto vzťahu

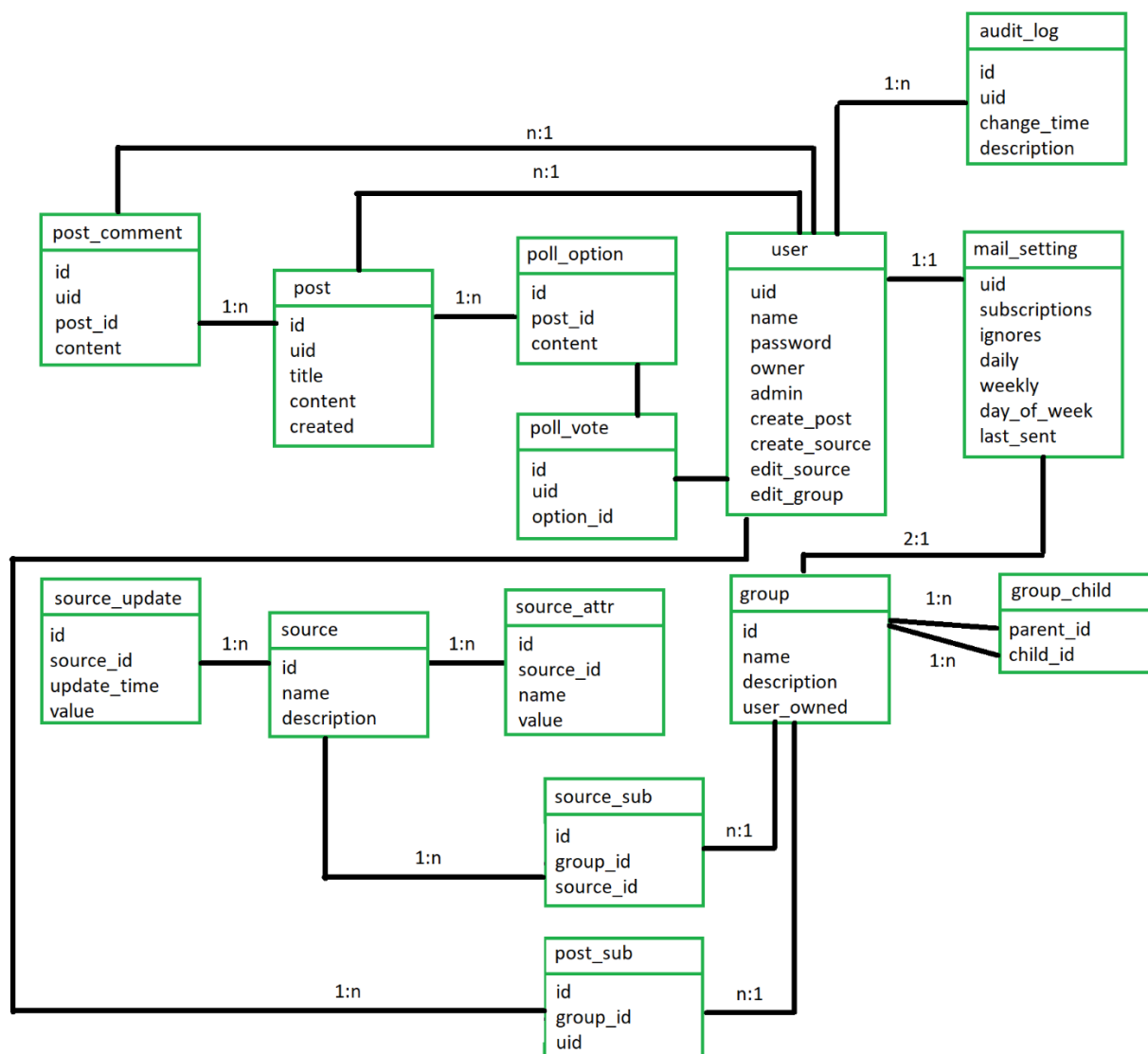
Tabuľka updatov obsahuje id jednoduchého zdroja, timestamp a nejaký identifikátor pridanej veci (napr. permalink...)

Tabuľka jednoduchých odberov obsahuje id zloženého zdroja, id jednoduchého zdroja.

Tabuľka postových odberov podobne obsahuje id zloženého zdroja a id usera (moderátor).

Tabuľka audit log obsahuje uid, timestamp a textový popis zmeny, ktorú vykonal

ER diagramy



Technologické požiadavky

Backend bude napísaný v Jave verzia 8, ako Springová aplikácia, s použitím Spring boot a Spring security pre spúšťanie a správu používateľov a bezpečnosti ako takej, poskytujúc REST interface, odpovede asi ako JSON (podľa toho, čo bude chcieť front end), požiadavky asi JSON. Logovanie bude asi log4j, k databáze pripojenie cez Hibernate.

Z pohľadu architektúry bude server interne obsahovať viacero servicov, ktoré združujú REST endpointy. Build bude pomocou Apache Maven, deploy asi ako WAR alebo ako JAR so zabundlovaným serverom.

Posielanie mailov bude časované pomocou Springu, a o poslanie samotné sa bude starať vhodná knižnica. Na generovanie tela mailu bude použité nejaké templatovanie, možno StringTemplate pre html formátované maily (hlavne ako tabuľky).

Pollovanie stránok bude realizované cez vhodnú Javovú knižnicu, pravdepodobne z rodiny springových.

Front end bude v nejakej rozumne aktuálnej JS knižnici.

Časový plán

týždeň od 7.3.

Základný setup projektu, Maven, databáza, configy atď.

týždeň od 14.3.

Triviálny front end, useri a základy REST interface

týždeň od 21.3.

REST interface

týždeň od 28.3.

Sledovanie zdrojov, emaily

týždeň od 4.4.

Front end

týždeň od 11.4.

dorábanie, testovanie atď.

Hodiny neviem rozumne odhadnúť, ale jednotlivé časti by mohli byť tak podobne náročné.