**Databáze/streamovací služba anime filmů.**

Daniel Makovský (MAK-0065)

15.10.2022

1. **Specifikace**

**Motivace**

Už několik let naráží komunita milovníků anime na to, že neexistuje jednotná databáze anime filmů (dále jen **anime**) a seriálů, která by obsahovala kromě hodnocení v podobě komentářů a liků, taky kompletní seznam animátorů a editorů (dále jen „mangak“), hodnocení kritiků, a zároveň i možnost přímého streamování filmu či seriálu z našich serverů, nebo odkazy na jiné služby, kde je možno film shlédnout.

**Role**

Do systému budou přistupovat **správci** databáze, kteří budou moct upravovat veškeré záznamy a moderovat komunikaci v komentářové sekci. Taktéž budou mít do systému přístup vybrané spřátelené **vydavatelské společnosti**, které budou mít na starosti údržbu a aktuálnost filmů či seriálů pouze dané společnosti.

Další důležitá role v systému je běžný **uživatel (anonymní)**, který může zjišťovat informace o jakémkoli filmu, případně film přehrát. Někteří uživatelé můžou mít také předplatné, které jim otevře možnost přehrání ve vyšší kvalitě, nebo sledování zpoplatněných filmů. Pro některé akce jako komentáře, nebo ukládání stavu přehrání musí být uživatel **přihlášen**

**Vstupy**

Celý systém je hlavně zaměřen na data anime. Systém bude uchovávat data o délce filmů počtu přehrání, mangaky, stručný a podrobný popis a seznam všech přehratelných zdrojů. Každý zdroj má krom odkazu na soubor na úložišti, nebo případně externí hosting (netu.io a podobné), uloženou také dostupnou kvalitu přehrávání a typ zpoplatnění (zdarma, předplatné naší služby, předplatné externí služby).

V případě že se jedná o anime seriál budou tyto informace pro každou epizodu oddělené. K datům anime u seriálu přibydou číslo epizody a číslo série. Souhrnné data pro celý seriál budou uloženy jako epizoda 0 a série 0. Filmy nebudou mít tyto data vyplněny.

Každý mangaka může spolupracovat na více anime. Úpravy budou moct provádět pouze administrátoři nebo uživatelé vytvoření pro vydavatelské společnosti. Veškeré změny budou podléhat auditování.

Uživatelé budou vystupovat v několika rolích popsaných výše. Mezi informace které potřebujeme ukládat patří přezdívka, email, heslo, fakturační údaje jako jméno, příjmení, adresa a stát. Taktéž z důvodu lokalizace zvolený jazyk, čas posledního přihlášení, preference notifikací, zda má zaplacené předplatné a typ účtu. (Dostupné typy jsou Administrátor, Vydavatel, Návštěvník). Každý uživatel může měnit pouze vlastní profil (s výjimkou administrátorů, ti můžou měnit profil všech uživatelů).

Uživatelé budou mít možnost anime hodnotit v podobě liků, nebo komentářů. Komentář bude povinně obsahovat popis, odkaz na uživatele a hvězdičkové hodnocení 1 až 10, viditelnost a uživatele který komentář skryl. Komentář může odpovídat na jiný komentář, v tom případě bude mít uloženou také referenci na původní komentář. Administrátoři, nebo autor komentáře budou mít možnost komentář skrýt. Změny, nebo skrytí komentáře nebudou podléhat auditu. V případě že v odpovědi na komentář dojde ke skrytí původního komentáře, bude zobrazena informace, že komentář odkazuje na smazaný komentář. V případě že bude skrytá celá struktura, nezobrazí se žádný z komentářů. Smazání uživatele, neznamená skrytí jeho komentářů. Administrátor (případně uživatel pokud maže svůj účet), bude mít možnost při mazání uživatele skrýt všechny jeho komentáře jedním kliknutím.

Každý like bude obsahovat odkaz na anime a odkaz na uživatele a informace, zda byl již sečten do statistik a informaci, zda je like stále aktivní. V případě, že uživatel like odebere až po sečtení statistik, bude like deaktivován a označen jako nesečtený.

Jeden z důležitých vstupů jsou také žádosti o přidání anime. (Tyto žádosti budou obsahovat referenci na uživatele, jméno požadovaného anime, zprávu uživatele, a odkaz na administrátora, který požadavek vyřizuje). Tyto žádosti budou administrátoři postupně zpracovávat a anime přidávat do databáze.

Každé přehrání filmu, nebo kliknutí na odkaz externího zdroje uloží do databáze informaci o přehrání. Tato informace bude obsahovat odkaz na uživatele, odkaz na zdroj a čas. Opakované shlédnutí stejným uživatelem chceme počítat jako nové (Stejně jako například YouTube).

**Výstupy**

Jedním z hlavních výstupů, který bude dostupný všem uživatelům bude seznam anime a anime seriálů. Anime bude možné fulltextově filtrovat podle jména anime či epizody, jmen mangak a hvězdičkového hodnocení. Seriály budou obsahovat souhrnné statistiky jako například součet délek epizod, počet unikátních liků, komentářů a shlédnutí na všech epizodách. V neposlední řadě filtraci podle stavu shlédnutí a označení likem pro daného uživatele.

Pro administrátory bude aplikace umožňovat výpis všech uživatelů s vyhledáváním podle přezdívky a stavu předplatného. Taktéž výpis všech komentářů s filtrací podle stavu smazání a výpis všech žádostí o přidání anime s možností filtrace podle přiřazení.

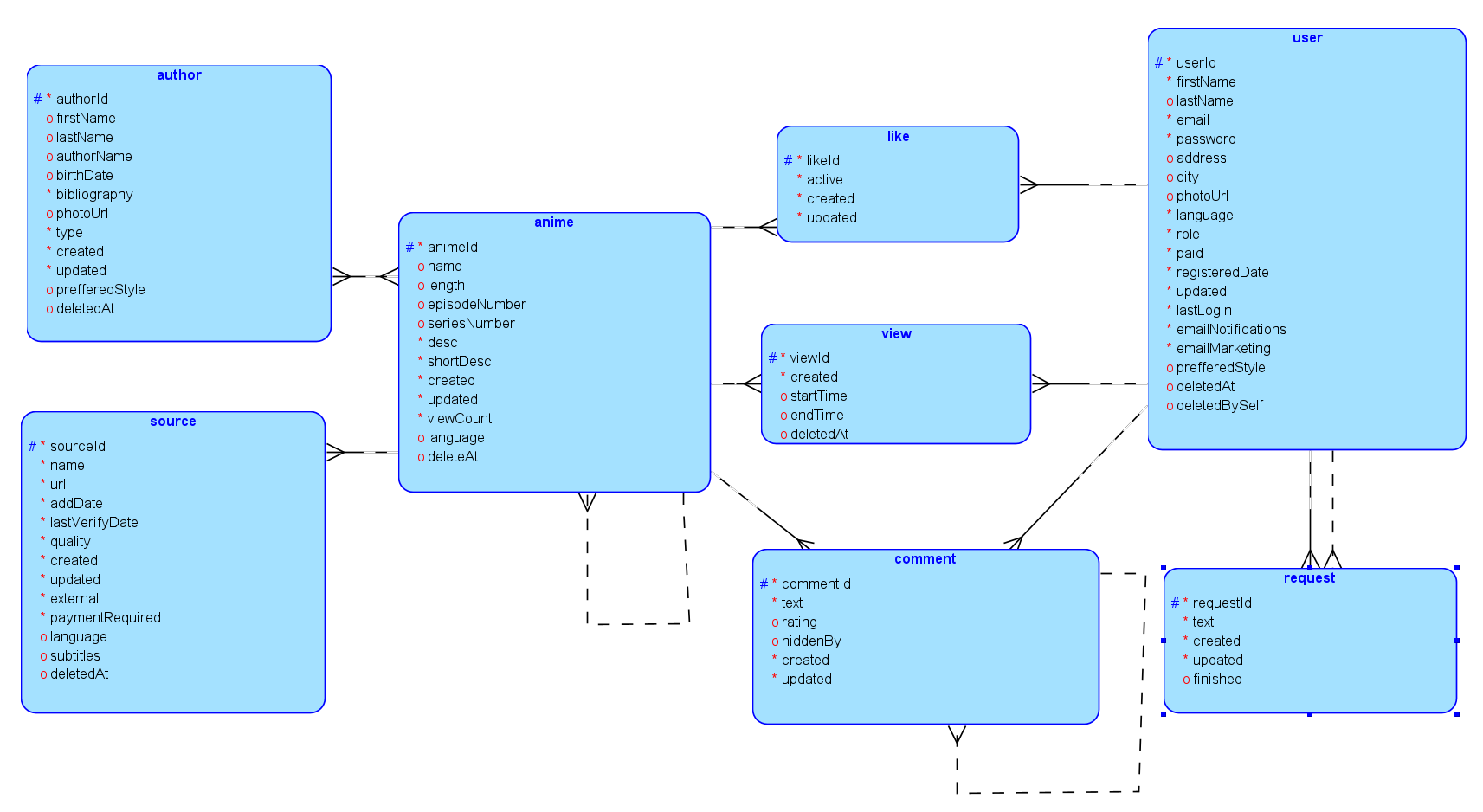
Neméně důležitou součástí prostředí pro administrátory je vytvoření grafů počtu shlédnutí, liků, komentářů pro jednotlivé anime, seriály a epizody. Taktéž budou mít administrátoři možnost zobrazit statistiky moderace (počtu smazaných komentářů) dle jednotlivých administrátorů.

**Funkce**

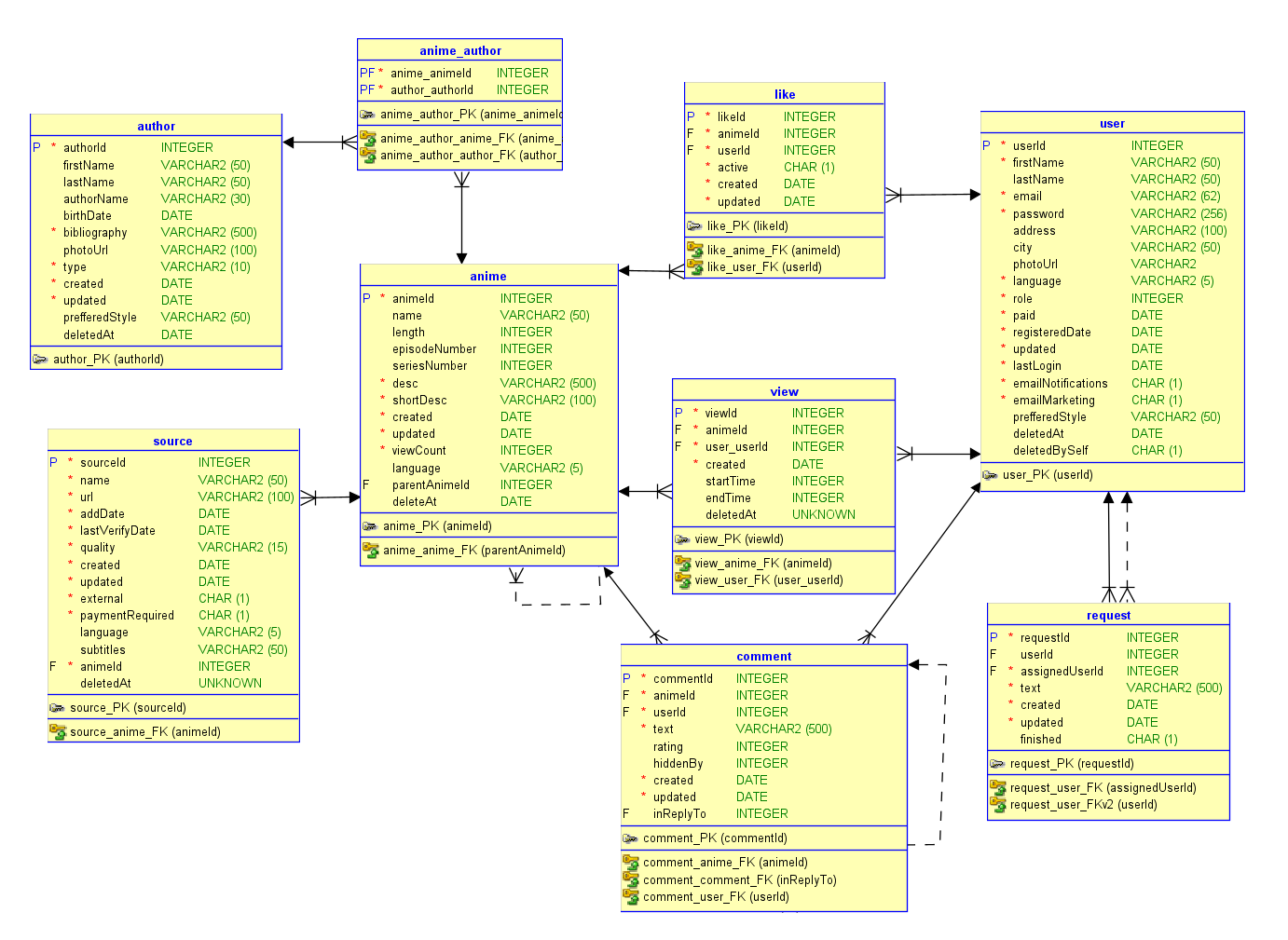
Jednou ze složitějších funkcí systému je optimalizace načítání liků a zhlédnutí. Chceme předejít potřebě načítat několik Gb dat pro zobrazení statistik na hlavní stránce. Tyto data však nemusí být vždy aktuální a stačí je přepočítávat v několika hodinovém intervalu a ukládat přímo k anime.

1. **Datová analýza**

**Konceptuální datový model**



**Relační datový model**



**Datový slovník**

Popis jednotlivých tabulek je uveden v následujícím datovém slovníku.

Tabulka **anime:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Název atributu | Dat. typ | Délka | Klíč | Null | IO | Popis |
| animeId | INTEGER |  | PK | Ne |  | Automaticky inkrementovaný PK |
| parentAnimeId | INTEGER |  | FK(anime) | Ano |  | Anime, kterému epizoda patří |
| name | VARCHAR | 50 |  | Ano |  | Jméno anime |
| length | INTEGER |  |  | Ano |  | Délka anime |
| episodeNumber | INTEGER |  |  | Ano |  | Číslo epizody |
| seriesNumber | INTEGER |  |  | Ano |  | Číslo série |
| desc | VARCHAR | 500 |  | Ne |  | Popis filmu/série/epizody |
| shortDesc | VARCHAR | 100 |  | Ne |  | Kratší verze popisu |
| language | VARCHAR | 5 |  | Ano |  | Zkratka jazyku, ve kterém je originál |
| created | DATE |  |  | Ne | 3 | Datum vytvoření |
| updated | DATE |  |  | Ne | 4 | Datum poslední úpravy |
| viewCount | INTEGER |  |  | Ne | 2 | Přibližný počet shlédnutí |
| deletedAt | DATE |  |  | ANO |  | Datum smazání |

Tabulka **anime\_author:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Název atributu | Dat. typ | Délka | Klíč | Null | IO | Popis |
| animeId | INTEGER |  | FK(anime) | Ne |  | Anime na kterém autor pracoval |
| authorId | INTEGER |  | FK(author) | Ne |  | Autor anime |

Tabulka **author:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Název atributu | Dat. typ | Délka | Klíč | Null | IO | Popis |
| authorId | INTEGER |  | PK | Ne |  | Automaticky inkrementovaný PK |
| firstName | VARCHAR | 50 |  | Ano |  | Jméno |
| lastName | VARCHAR | 50 |  | Ano |  | Příjmení |
| authorName | VARCHAR | 30 |  | Ano |  | Umělecké jméno/přezdívka |
| birthDate | DATE |  |  | Ano |  | Datum narození |
| bibliography | VARCHAR | 500 |  | Ne |  | Popis autora |
| photoUrl | VARCHAR | 100 |  | Ano |  | Url adresa fotky autora |
| type | VARCHAR | 10 |  | Ne | 1 | Typ autora |
| prefferedStyle | VARCHAR | 50 |  | Ano |  | Preferovaný typ kresby |
| created | DATE |  |  | Ne | 3 | Datum vytvoření |
| updated | DATE |  |  | Ne | 4 | Datum poslední úpravy |
| deletedAt | DATE |  |  | ANO |  | Datum smazání |

Tabulka **source:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Název atributu | Dat. typ | Délka | Klíč | Null | IO | Popis |
| sourceId | INTEGER |  | PK | Ne |  | Automaticky inkrementovaný PK |
| name | VARCHAR | 50 |  | Ne |  | Jméno zdroje nebo externí stránky |
| url | VARCHAR | 100 |  | Ne |  | Odkaz na přehrání |
| addDate | DATETIME |  |  | Ne |  | Datum a čas přídání |
| lastVerifyDate | DATETIME |  |  | Ne |  | Datum a čas posledního ověření funkčnosti |
| quality | VARCHAR | 15 |  | Ne | 5 | Identifikátor kvality |
| external | BOOLEAN |  |  | Ne |  | Přehrávač není na našem serveru |
| paymentRequired | BOOLEAN |  |  | Ne |  | Je skryto za platební bránou jakéhokoliv tyku |
| language | VARCHAR | 5 |  | Ano |  | Jazyk audia |
| subtitles | VARCHAR | 50 |  | Ano |  | Jazyky, ve kterých jsou dostupné titulky oddělené čárkou |
| animeId | INTEGER |  | FK (anime) | Ne |  | Odkaz na anime |
| created | DATE |  |  | Ne | 3 | Datum vytvoření |
| updated | DATE |  |  | Ne | 4 | Datum poslední úpravy |
| deletedAt | DATE |  |  | ANO |  | Datum smazání |

Tabulka **like:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Název atributu | Dat. typ | Délka | Klíč | Null | IO | Popis |
| likeId | INTEGER |  | PK | Ne |  | Automaticky inkrementovaný PK |
| animeId | INTEGER |  | FK (anime) | Ne |  | Anime které dostalo like |
| userId | INTEGER |  | FK (user) | Ne |  | Uživatel který dal like |
| active | DATETIME |  |  | Ne |  | Like je stále aktivní |
| created | DATE |  |  | Ne | 3 | Datum vytvoření |
| updated | DATE |  |  | Ne | 4 | Datum poslední úpravy |

Tabulka **view:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Název atributu | Dat. typ | Délka | Klíč | Null | IO | Popis |
| viewId | INTEGER |  | PK | Ne |  | Automaticky inkrementovaný PK |
| animeId | INTEGER |  | FK (anime) | Ne |  | Anime které dostalo like |
| userId | INTEGER |  | FK (user) | Ne |  | Uživatel který dal like |
| startTime | INTEGER |  |  | Ano |  | Moment od kterého je anime shlédnuté |
| endTime | INTEGER |  |  | Ano |  | Moment do kterého je anime shlédnuté |
| created | DATE |  |  | Ne | 3 | Datum shlédnutí |
| deletedAt | DATE |  |  | Ano |  | Datum smazání |

Tabulka **comment:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Název atributu | Dat. typ | Délka | Klíč | Null | IO | Popis |
| commentId | INTEGER |  | PK | Ne |  | Automaticky inkrementovaný PK |
| animeId | INTEGER |  | FK(anime) | Ne |  | Okomentované anime |
| userId | INTEGER |  | FK(user) | Ne |  | Uživatel který komentář napsal |
| text | VARCHAR | 500 |  | Ne |  | Text komentáře |
| rating | INTEGER |  |  | Ano |  | Hodnocení anime |
| hiddenBy | INTEGER |  |  | Ano |  | Administrátor, který komentář skryl |
| inReplyTo | INTEGER |  | FK(comment) | Ano |  | Komentář na který odpovídá |
| created | DATE |  |  | Ne | 3 | Datum vytvoření |
| updated | DATE |  |  | Ne | 4 | Datum poslední úpravy |

Tabulka **user:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Název atributu | Dat. typ | Délka | Klíč | Null | IO | Popis |
| userId | INTEGER |  | PK | Ne |  | Automaticky inkrementovaný PK |
| firstName | VARCHAR | 50 |  | Ne |  | Jméno nebo přezdívka |
| lastName | VARCHAR | 50 |  | Ano |  | Příjmení |
| email | VARCHAR | 62 |  | Ne |  | Email |
| password | VARCHAR | 256 |  | Ne |  | Heslo |
| address | VARCHAR | 100 |  | Ano |  | Adresa |
| city | VARCHAR | 50 |  | Ano |  | Město |
| photoUrl | VARCHAR | 100 |  | Ano |  | Profilová fotka |
| language | VARCHAR | 5 |  | Ne |  | Jazyk |
| role | INTEGER |  |  | Ne |  | Role |
| paid | BOOLEAN | 1 |  | Ne |  | Má zaplacené prémium |
| registeredDate | DATETIME |  |  | Ne | 3 | Datum a čas registrace |
| updated | DATETIME |  |  | Ne |  | Poslední editace profilu |
| lastLogin | DATETIME |  |  | Ne |  | Datum a čas poslednídho přihlášení |
| emailNotifications | BOOLEAN |  |  | Ne |  | Emailové notifikace povoleny |
| emailMarketing | BOOLEAN |  |  | Ne |  | Emailový marketing povolen |
| prefferedStyle | VARCHAR | 50 |  | Ano |  | Preferovaný styl anime |
| deletedAt | DATE |  |  | Ano |  | Datum smazání |
| deletedBySelf | BOOLEAN |  |  | Ne |  | Uživatel si profil smazal sám |

Tabulka **request:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Název atributu | Dat. typ | Délka | Klíč | Null | IO | Popis |
| requestId | INTEGER |  | PK | Ne |  | Automaticky inkrementovaný PK |
| assignedUserId | INTEGER |  | FK (user) | Ano |  | Administrátor, který zpracovává požadavek |
| userId | INTEGER |  | FK(user) | Ne |  | Uživatel který požadavek přidal |
| text | VARCHAR | 500 |  | Ne |  | Text požadavku |
| created | DATETIME |  |  | Ne | 3 | Datum vytvoření |
| updated | DATE |  |  | Ne | 4 | Datum poslední úpravy |

Integritní omezení:

1. **author.type** musí mít jednu z hodnot „Mangaka“, „Editor“, „Producer“, „Script“, „Writer“, „VoiceAct“, „Music“, „Manager“, „Director“
2. **anime.viewCount** výchozí hodnota 0
3. **\*.created** výchozí hodnota aktuální čas
4. **\*.updated** při každé aktualizaci nastaveno na aktuální čas, při vytvoření také
5. **quality** jedna z hodnot „HD“, „FHD“, „4K“, „720p“ a „HDx“, „FHDx“, „4Kx“, „720px“
6. Formální analýza

(Přeskočeno)

1. Stavová analýza

Viditelnost komentáře:

* **Viditelný** - *#comment.hiddenBy* *IS NULL*
* **Skrytý** - *#comment.hiddenBy IS NOT NULL*

Pozn. #comment představuje záznam aukce, u které určujeme stav v tabulce comments.

Platnost zdroje:

* **Validní** - #source.lastVerifyDate není starší než 2 týdny od aktuálního času
* **Nevalidní** - #source.lastVerifyDate je starší než 2 týdny od aktuálního času

Pozn. #source představuje záznam zdroje, u kterého určujeme stav v tabulce source.

Stavy uživatele:

* **Běžný** - #user.deletedAt IS NULL a #user.paid není starší než délka předchozího měsíce ve dnech
* **Prémiový** - #user.deletedAt IS NULL a #user.paid je starší, než délka předchozího měsíce ve dnech, nebo IS NULL
* **Zrušený** - #user.deletedAt IS NOT NULL a deletedBySelf je True
* **Zabanovaný** - #user.deletedAt IS NOT NULL a deletedBySelf je False

Pozn. #user představuje záznam uživatele, u kterého určujeme stav v tabulce user.

1. Funkční analýza

Zodpovědnosti:

Přihlášený uživatel má stejné zodpovědnosti jako anonymní + některé navíc  
Správci databáze mají stejné zodpovědnosti jako přihlášený uživatel + některé navíc

Mazání záznamů operací DELETE je v celé aplikaci zakázáno.

Jediná výjimka je funkce 5.5, která má za úkol snížit počet záznamů jejich spojením

* 1. Seznam funkcí

1. Evidence anime  
   Zodpovědnost: Správce aplikace / Vydavatelské společnosti
   1. Nové anime
   2. Seznam anime  
      Zodpovědnost: Uživatel (anonymní)
   3. Detail anime  
      Zodpovědnost: Uživatel (anonymní)
   4. Seznam epizod anime  
      Zodpovědnost: Uživatel (anonymní)
   5. Aktualizace anime
   6. Smazání anime – dojde k nastavení deletedAt atributu na aktuální čas
   7. Obnovení anime
   8. Předchozí a následující anime
   9. Změna pořadí anime epizod
2. Evidence autorů  
   Zodpovědnost: Uživatel (anonymní)
   1. Nový autor  
      Zodpovědnost: Správce aplikace / Vydavatelské společnosti
   2. Seznam autorů
   3. Detail autora
   4. Seznam autorů podle anime
   5. Aktualizace autora  
      Zodpovědnost: Správce aplikace / Vydavatelské společnosti
   6. Smazání autora – dojde k nastavení deletedAt atributu na aktuální čas
   7. Přidání autora k anime  
      Zodpovědnost: Správce aplikace / Vydavatelské společnosti
   8. Odebrání autora) od anime  
      Zodpovědnost: Správce aplikace / Vydavatelské společnosti
3. Evidence zdrojů  
   Zodpovědnost: Správce aplikace / Vydavatelské společnosti
   1. Nový zdroj
   2. Seznam zdrojů pro anime  
      Zodpovědnost: Uživatel (anonymní)
   3. Aktualizace autora
   4. Smazání zdroje – dojde k nastavení deletedAt atributu na aktuální čas
4. Evidence liků  
   Zodpovědnost: Uživatel (přihlášený)
   1. Nový like
   2. Odebrání liku
5. Evidence shlédnutí  
   Zodpovědnost: Uživatel (přihlášený) [pouze svoje záznamy]
   1. Přidání shlédnutí
   2. Seznam shlédnutých anime
   3. Seznam shlédnutých částí anime (pouze pro externí api)
   4. Odebrání shlédnutí (pouze pro externí api) – dojde k nastavení deletedAt atributu na aktuální čas
   5. Optimalizace shlédnutí – v pravidelném intervalu sjednocení záznamů, které se časově překrývají
6. Evidence komentářů  
   Zodpovědnost: Uživatel (přihlášený)
   1. Přidání komentáře  
      Zodpovědnost: Uživatel (přihlášený) [pouze svůj komentář]
   2. Aktualizace komentáře  
      Zodpovědnost: Uživatel (přihlášený) [pouze svůj komentář]
   3. Skrytí komentáře  
      Zodpovědnost: Uživatel (přihlášený) [pouze svůj komentář] / Správce aplikace
   4. Seznam komentářů k anime  
      Zodpovědnost: Uživatel (anonymní)
   5. Seznam komentářů ke komentáři  
      Zodpovědnost: Uživatel (anonymní)
7. Evidence uživatelů
   1. Nový uživatel  
      Zodpovědnost: Uživatel (anonymní)
   2. Seznam uživatelů  
      Zodpovědnost: Uživatel (anonymní)
   3. Detail uživatele  
      Zodpovědnost: Uživatel (přihlášený) [pouze svůj profil]
   4. Aktualizace uživatele  
      Zodpovědnost: Uživatel (přihlášený) [pouze svůj profil]
   5. Smazání uživatele – dojde k nastavení deletedAt atributu na aktuální čas (pokud uživatel smazal profil sám nastaví se deletedBySelf na 1)  
      Zodpovědnost: Uživatel (přihlášený) [pouze svůj profil] / Správce aplikace
   6. Obnovení uživatele  
      Zodpovědnost: Uživatel (přihlášený) [pouze svůj profil] / Správce aplikace
8. Evidence požadavků

Zodpovědnost: Správce aplikace

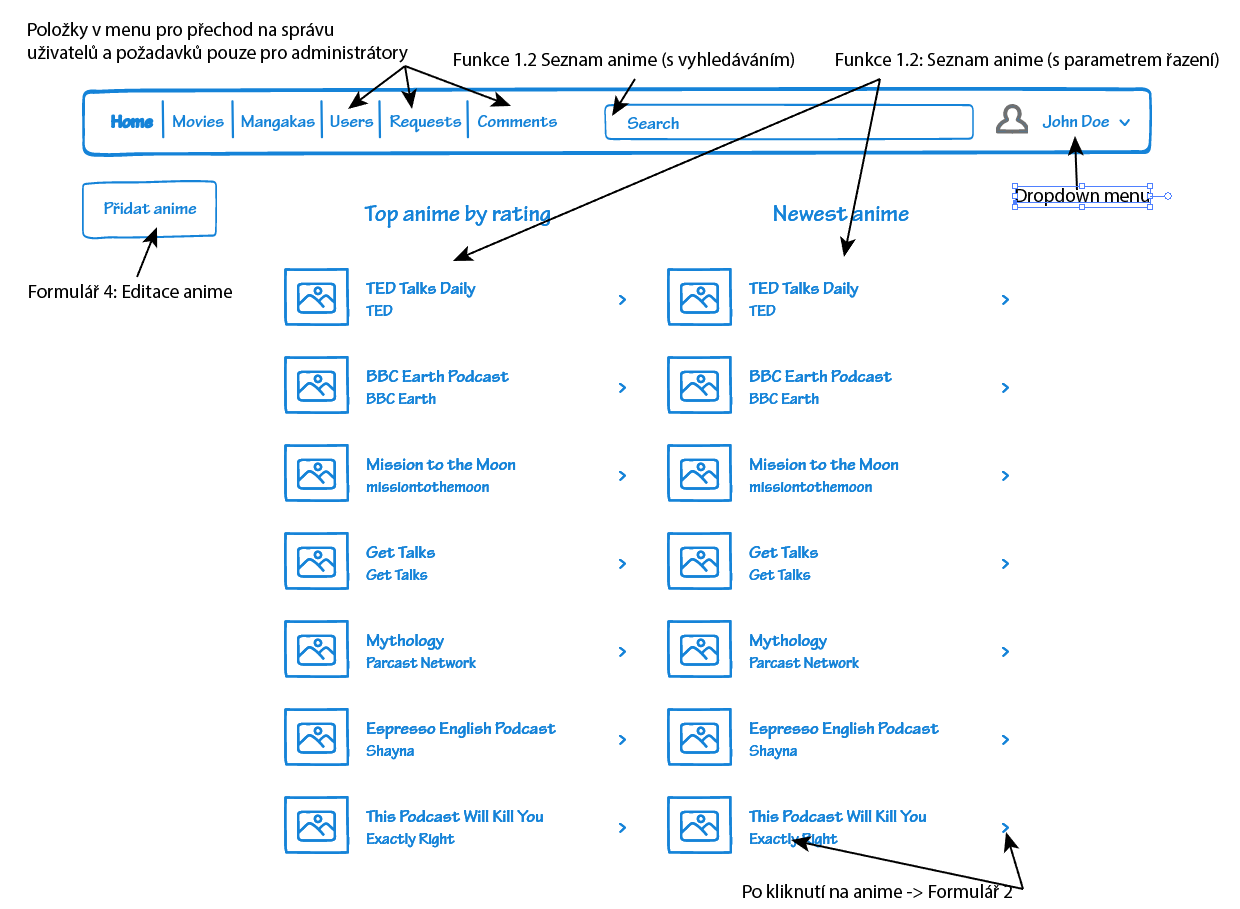
* 1. Nový požadavek  
     Zodpovědnost: Uživatel (anonymní)
  2. Detail požadavku
  3. Aktualizace požadavku
  4. Ukončení požadavku
  5. Detailní popis funkcí

|  |
| --- |
| Funkce 1.9: Změna pořadí epizod  Vstupy:   * #episode\_id – ID epizody, kterou chceme posunout * #direction – směr kam chceme epizodu posunout („down“ pro posun dolů, „up“ pro posun nahoru)   Výstupy:   * #error\_msg – Hlášení o chybě, která mohla vzniknout   Funkce zkontroluje, zda směr je jeden ze stringů „up“, nebo „down“. Poté zkontroluje, zda epizoda, u které se uživatel snaží změnit pořadí je opravdu epizoda. Dále zkontroluje, zda je epizod rodičovského anime více, než jedna a zda je posun vůbec možný (epizoda je poslední a uživatel jí chce posunout níže, nebo epizoda je první a uživatel ji chce posunout výše). Funkce adekvátně změní číslo pořadí u žádané epizody a okolních zasažených epizod. Funkce bude řešena jako transakce   * + - 1. Zkontrolujeme, zda #direction je string „down“. Pokud ano do proměnné #dirInt vložíme 1.       2. Pokud je #direction rovno „up“, do #dirInt vložíme -1.       3. Pokud hodnota #direction není rovna s žádnou předchozí, nastavíme #error\_msg na „Invalid arguments“ a transakci ukončíme.       4. Do proměnných #parent\_anime\_id a #episode\_number načteme id rodičovského anime a pořadové číslo epizody, kterou chceme posunout   SELECT parentAnimeId, episodeNumber  FROM Anime  WHERE animeId = #episode\_id   * + - 1. Pokud krok 2 selhal, nebo nevrátil hodnoty nastavíme #error\_msg na „Episode not found, or is not an episode.“ a transakci ukončíme.       2. Zkontrolujeme, jestli existuje anime s požadovaným pořadovým číslem   SELECT count(\*)  FROM Anime  WHERE parentAnimeId = #parent\_anime\_id AND episodeNumber = #episode\_number + #dirInt   * + - 1. Pokud počet epizod z kroku 6 není rovno 1, nastavíme #error\_msg na „Move out of range, or episode numbers are not sequencial“ a transakci ukončíme.       2. Upravíme pořadové číslo epizody, se kterou je potřeba epizodu prohodit   UPDATE Anime SET episodeNumber = #episode\_number  WHERE parentAnimeId = #parent\_anime\_id AND episodeNumber = #episode\_number + #dirInt   * + - 1. Nastavíme správné číslo pořadové číslo epizodě, kterou se uživatel snaží posunout   UPDATE Anime SET episodeNumber = #episode\_number + #dirInt  WHERE animeId = #episode\_id |

|  |
| --- |
| Funkce 6.4: Seznam komentářů k anime  Rekapitulace ze specifikace:   * Je žádané, aby dotaz vracel i smazané komentáře, pokud mají odpovědi, které nejsou smazané. * Smazání uživatele, neznamená skrytí jeho komentářů.   Vstupy:   * #animeId – id anime pro které hledáme komentáře   SELECT c.commentId,  CASE  WHEN u.deletedAt IS NOT NULL THEN 'Deleted user'  ELSE u.firstName  END as userName,  CASE  WHEN c.hiddenBy IS NOT NULL THEN 'Comment has been deleted'  ELSE c.text  END as text,  c.rating, (  SELECT count(\*)  FROM comment c2  WHERE c2.inReplyTo = c.commentId AND (C2.hiddenBy IS NULL OR (  SELECT count(\*)  FROM comment c3  WHERE c3.inReplyTo = c2.commentId AND c3.hiddenBy IS NULL  ))  ) as replies  FROM Comment c LEFT JOIN User u on c.userId = u.userId  WHERE  animeId = #animeId AND  inReplyTo IS NULL AND  (c.hiddenBy IS NULL OR (  SELECT count(\*)  FROM comment c2  WHERE c2.inReplyTo = c.commentId AND (C2.hiddenBy IS NULL OR (  SELECT count(\*)  FROM comment c4  WHERE c4.inReplyTo = c2.commentId AND c4.hiddenBy IS NULL  ))  ) > 0);  Poznámky:   * Funkce CASE je podle mých znalostí **snad** ve standardu sql. Respektive MS SQL, Oracle, MariaDB i PostgreSQL jí mají definovanou stejně. * Předpokládám, že jelikož klauzule WITH je už DDL/DML příkaz a možná dokonce i mimo specifikaci standardního SQL tak ho zde nemohu použít. Proto jsem rekurzi nahradil jen trojitým zanořením což vrátí trochu přesnější výsledky v replies. Nejpřesnější by byla rekurze, ale to může být v produkčním řešení až moc náročné. * Ověřit funkčnost možno zde: http://sqlfiddle.com/#!9/d235b9/13 |

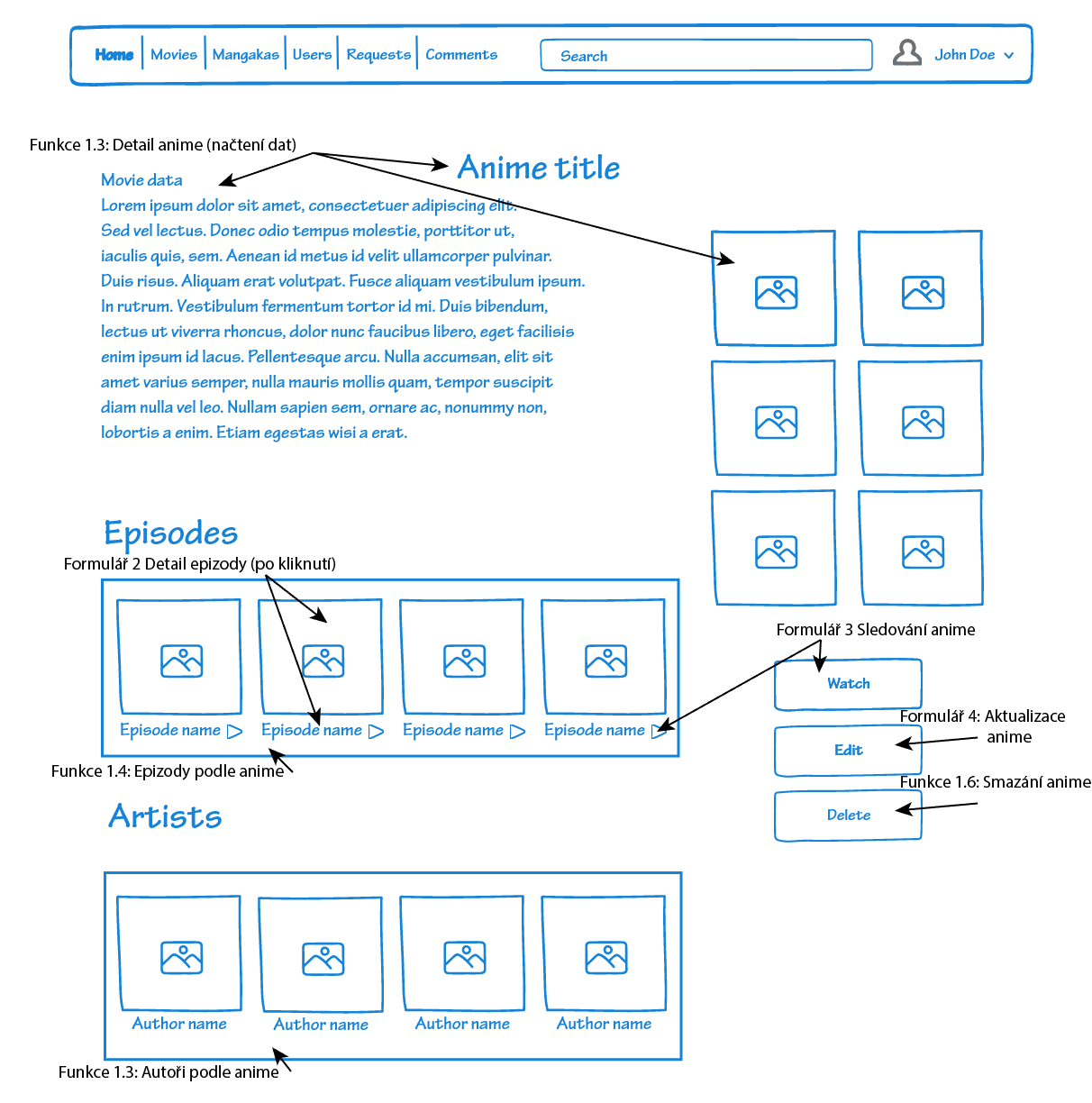
|  |
| --- |
| Funkce 6.5: Seznam komentářů ke komentáři  Rekapitulace ze specifikace:   * Je žádané, aby dotaz vracel i smazané komentáře, pokud mají odpovědi, které nejsou smazané. * Smazání uživatele, neznamená skrytí jeho komentářů.   Vstupy:   * #animeId – id anime pro které hledáme komentáře * #inReplyTo – id komentáře pro který hledáme odpovědi   SELECT c.commentId,  CASE  WHEN u.deletedAt IS NOT NULL THEN 'Deleted user'  ELSE u.firstName  END as userName,  CASE  WHEN c.hiddenBy IS NOT NULL THEN 'Comment has been deleted'  ELSE c.text  END as text,  c.rating, (  SELECT count(\*)  FROM comment c2  WHERE c2.inReplyTo = c.commentId AND (C2.hiddenBy IS NULL OR (  SELECT count(\*)  FROM comment c3  WHERE c3.inReplyTo = c2.commentId AND c3.hiddenBy IS NULL  ))  ) as replies  FROM Comment c LEFT JOIN User u on c.userId = u.userId  WHERE  animeId = #animeId AND  inReplyTo = #inReplyTo AND  (c.hiddenBy IS NULL OR (  SELECT count(\*)  FROM comment c2  WHERE c2.inReplyTo = c.commentId AND (C2.hiddenBy IS NULL OR (  SELECT count(\*)  FROM comment c4  WHERE c4.inReplyTo = c2.commentId AND c4.hiddenBy IS NULL  ))  ) > 0);  Poznámky:   * Funkce CASE je podle mých znalostí **snad** ve standardním sql. Respektive MS SQL, Oracle, MariaDB i PostgreSQL jí mají definovanou stejně. * Předpokládám, že jelikož klauzule WITH je už DDL/DML příkaz a možná dokonce i mimo specifikaci standardního SQL tak ho zde nemohu použít. Proto jsem rekurzi nahradil jen trojitým zanořením což vrátí trochu přesnější výsledky v replies. Nejpřesnější by byla rekurze, ale to může být v produkčním řešení až moc náročné. * Ověřit funkčnost možno zde: http://sqlfiddle.com/#!9/d235b9/13 |

1. Návrh uživatelského rozhraní
   1. Formuláře
      * 1. Hlavní stránka

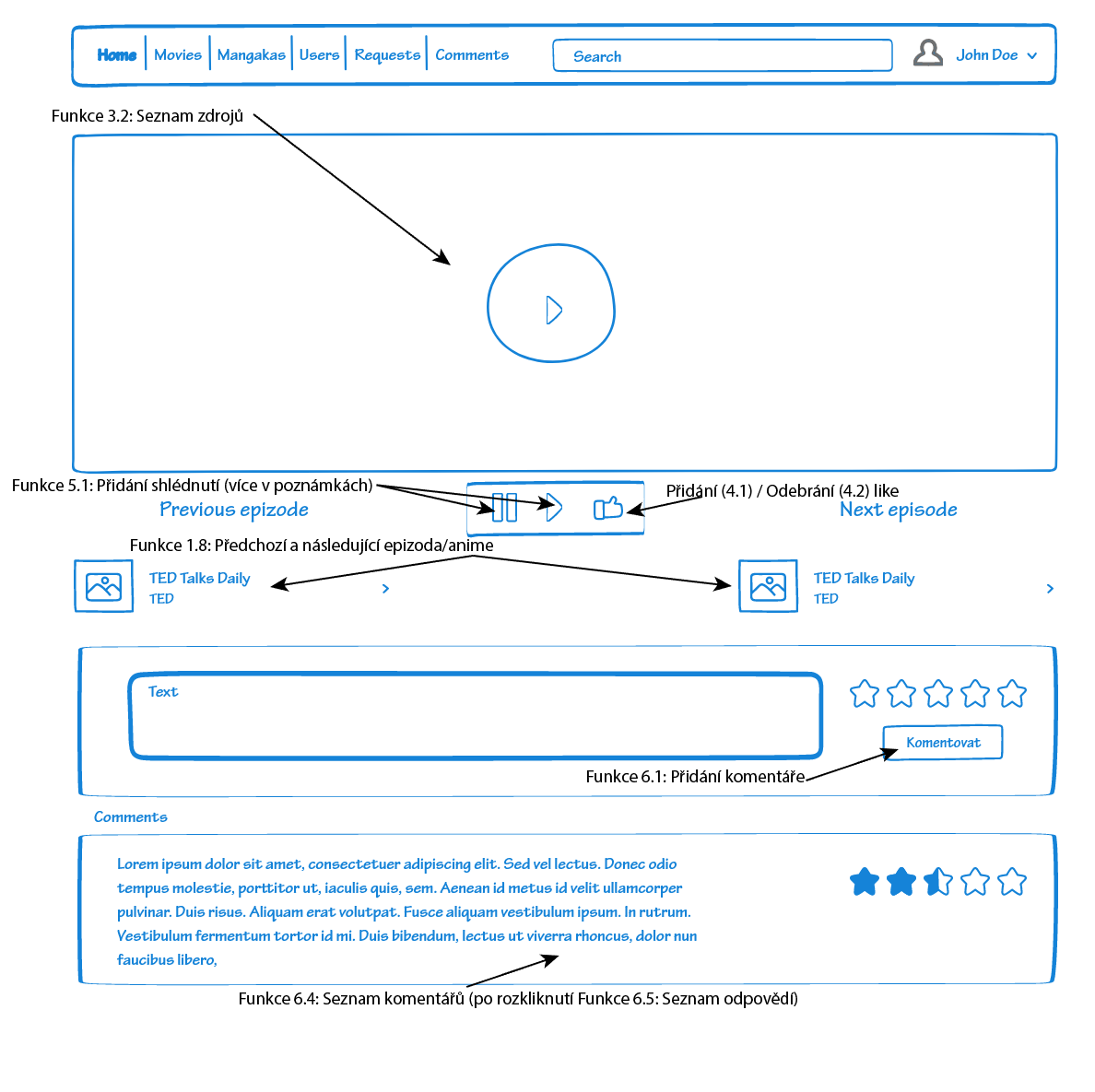


Dodatky k menu

* Dropdown menu
  + Tlačítko pro výpis shlédnutých anime
  + Tlačítko pro úpravu profilu
  + Tlačítko pro odhlášení
    - 1. Detail anime



* + - 1. Přehrávač anime



Poznámky:

* Element přehrávače automaticky použije první zdroj, přímo v přehrávači je možno přepnout na jiný. V případě že existují pouze placené zdroje zobrazí výzvu k předplatnému. V případě že existují pouze externí zdroje (např Netflix) zobrazí logo služby a odkaz.
* V případě že je ikona liku prázdná, po kliknutí se provede funkce přidání, v případě že je ikona plná, provede se funkce odebrání
  + - 1. Editace anime

