

FilmDB - Filmová databáze

Patrik Mintěl - MIN0150

13.12.2024

Vize

Proč?

Zákazník chce vytvořit platformu, kde mohou uživatelé sdílet a získávat informace o filmech, hodnotit je a přispívat recenzemi. Taková aplikace bude sloužit ke shlukování komunity filmových nadšenců, sloužit jako nástroj pro hledání vhodného filmu, na základě žánru a recenzí, propagaci nových filmů filmovými distributory.

Co?

Aplikace bude sloužit/obsahovat:

- ke správě databáze filmů (název, délka, žánry, herci, režisér, rok vydání...)
- k hodnocení uživatelů (uživatelé budou moci filmy hodnotit hvězdičkovým systémem 1-5)
- k psaním recenzí (uživatelé budou moci přidávat textové recenze)
- k vyhledávání a filtrování (filmy bude možno vyhledávat podle názvu, žánru, roku vydání, obsazení...)
- uživatelské účtu (uživatelé si budou moci přidávat filmy do oblíbených, upravovat svůj profil, přidávat komentáře na jiné profily, případně komunikovat s ostatními uživateli)
- api (aplikace bude disponovat API, která bude sloužit uživatelům k použití v jejich aplikacích, případně slouží jako rozhraní pro budoucí implementaci mobilní aplikace, nebo jiných služeb)

Jak?

- Primárně jako webová aplikace dostupná z každého zařízení prostřednictvím webového prohlížeče. Bude napsána ve fullstack frameworku SvelteKit a serverová část poběží na NodeJS. Jako databáze bude využívána MariaDB, nebo JSONDb.
- API bude implementována jako klasická REST API
- Autentifikace bude implementována pomocí cookies a JsonWebTokenu

Kde?

Aplikace bude dostupná od kud koliv, kde má uživatel přístup k internetovému připojení a je na zařízení s webovým prohlížečem na adrese <https://filmdb.cz>. Díky responzivnímu designu bude jednoduše i z mobilního telefonu.

Kdy?

Systém bude spolehlivý a poběží 24/7. Údržby se budou provádět během brzkých ranních hodin, kdy je očekávaná nízký trafic. Systém bude pravidelně zálohován,

Funkční specifikace

Use Case diagram

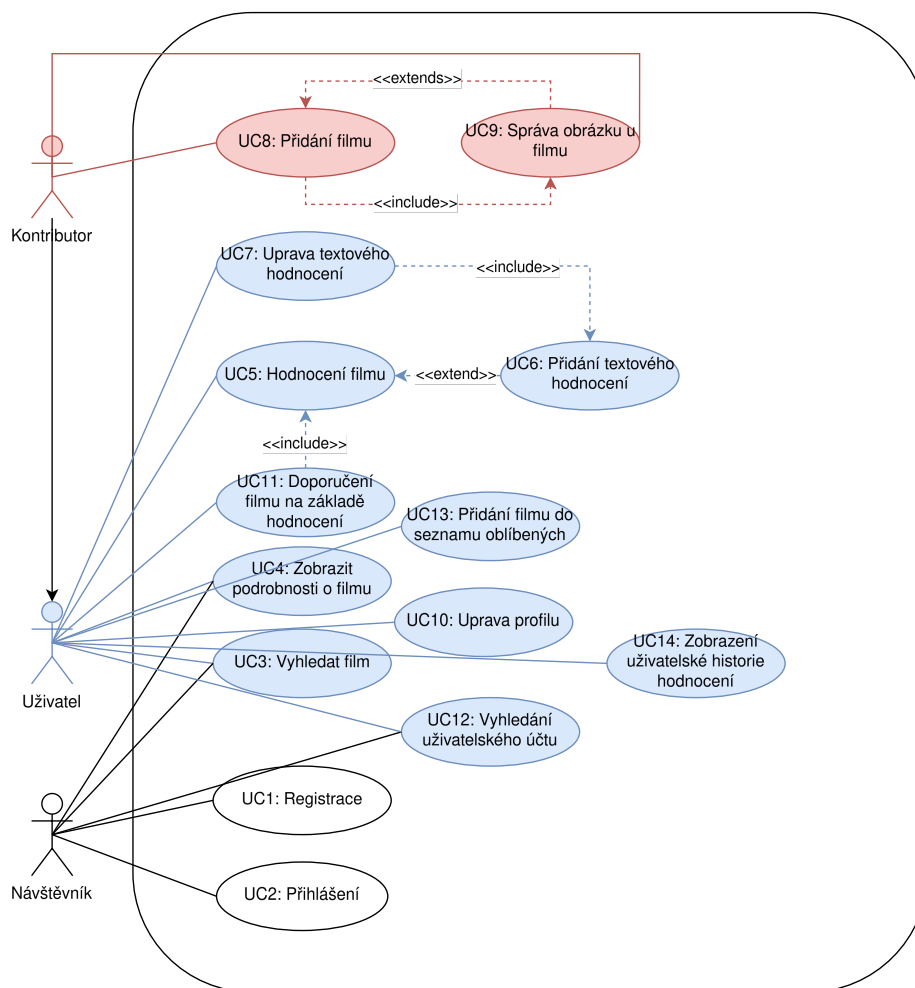


Figure 1: UC

Popis případů užití

UC5: Hodnocení filmu

Parametry:

- Název: Hodnocení filmu
- Kontext: Uživatel právě zhlédl film a chce přidat své hodnocení (například 4 z 5 hvězdiček)
- Primární aktér: Uživatel
- Předpoklad: Uživatel musí být přihlášen
- Minimální záruka: Chybové hlášky při chybě systému
- Záruka úspěchu: Hodnocení filmu je úspěšně uloženo a zobrazeno v profilu filmu i v uživatelské historii hodnocení.
- Trigger: Uživatelská akce

Hlavní scénář:

1. Uživatel naviguje na detail filmu
2. Uživatel klikne na tlačítko pro hodnocení filmu
3. Otevře se menu pro zadání hodnocení
4. Uživatel vybere počet hvězdiček
5. Uživatel potvrdí hodnocení
6. Systém uloží hodnocení do databáze
7. Uživatel obdrží hlášku o úspěšném přidání hodnocení

Alternativní scénář:

1. Uživatel naviguje na detail filmu
2. Uživatel klikne na tlačítko pro hodnocení filmu
3. Otevře se menu pro zadání hodnocení
4. Uživatel vybere počet hvězdiček
5. Uživatel potvrdí hodnocení
6. Systém hlásí chybu při ukládání hodnocení
7. Uživatel je vyzván k opětovnému zadání hodnocení
8. Systém uloží hodnocení do databáze
9. Uživatel obdrží hlášku o úspěšném přidání hodnocení

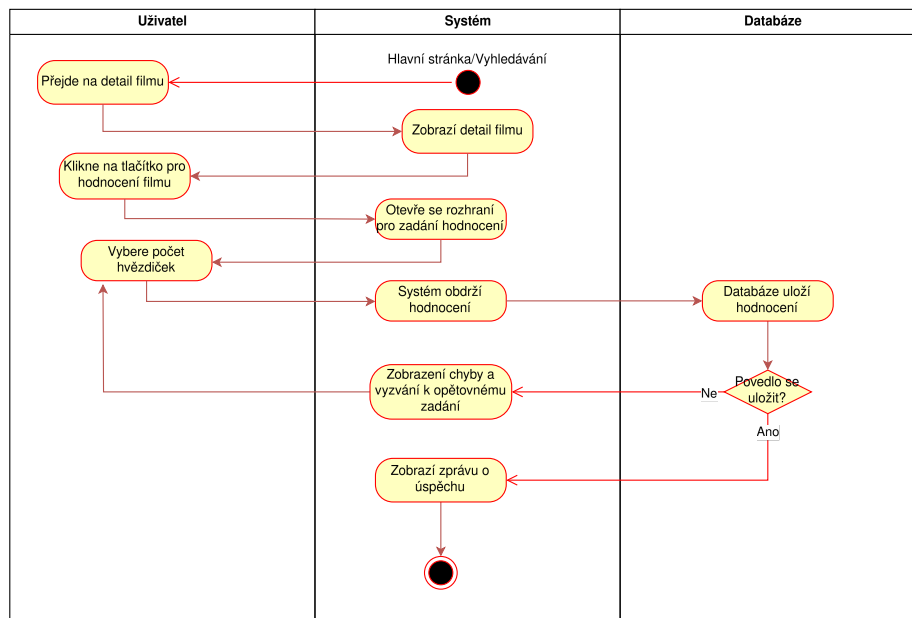


Figure 2: UC5 - Activity diagram

UC6: Přidání textového hodnocení**Parametry:**

- Název: Přidání textového hodnocení
- Kontext: Uživatel chce přidat textovou recenzi k hodnocenému filmu
- Primární aktér: Uživatel
- Předpoklad: Uživatel musí být přihlášen
- Minimální záruka: Chybové hlásky při chybě systému nebo při nevyplnění recenze
- Záruka úspěchu: Textové hodnocení je úspěšně uloženo a zobrazeno na stránce filmu i v uživatelské historii hodnocení
- Trigger: Uživatelská akce

Hlavní scénář:

1. Uživatel naviguje na detail filmu
2. Uživatel klikne na tlačítko pro přidání textového hodnocení
3. Otevře se textové pole pro zadání recenze
4. Uživatel napíše svou recenzi
5. Uživatel odešle recenzi
6. Systém uloží recenzi do databáze
7. Uživatel obdrží hlášku o úspěšném přidání recenze

Alternativní scénář:

1. Uživatel naviguje na detail filmu
2. Uživatel klikne na tlačítko pro přidání textového hodnocení
3. Otevře se textové pole pro zadání recenze
4. Uživatel napíše svou recenzi
5. Uživatel odešle recenzi
6. Uživatel nevyplní textové pole a pokusí se recenzi odeslat
7. Systém zobrazí chybu, že recenze musí obsahovat text

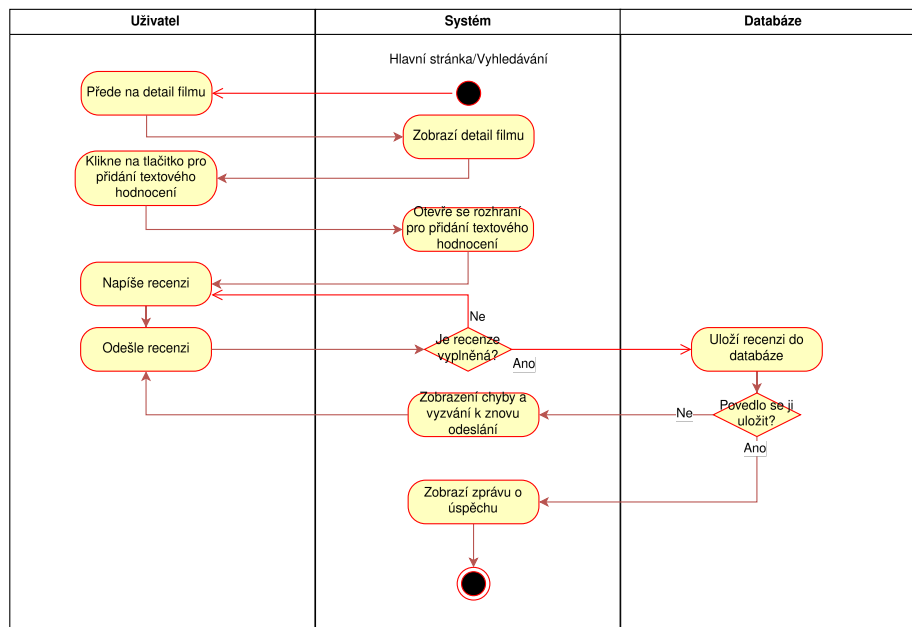


Figure 3: UC6 - Activity diagram

UC13: Přidání filmu do seznamu oblíbených**Parametry:**

- Název: Přidání filmu do seznamu oblíbených
- Kontext: Uživatel narazil na zajímavý film, který si chce uložit do svého seznamu oblíbených
- Primární aktér: Uživatel
- Předpoklad: Uživatel musí být přihlášen
- Minimální záruka: Chybové hlášky při chybě systému
- Záruka úspěchu: Film je úspěšně přidán do seznamu oblíbených a zobrazen v sekci “Oblíbené filmy” v profilu uživatele
- Trigger: Uživatelská akce

Hlavní scénář:

1. Uživatel naviguje na detail filmu
2. Uživatel klikne na tlačítko “Přidat do oblíbených”
3. Systém pošle požadavek na uložení filmu do oblíbených
4. Film je úspěšně přidán
5. Uživatel obdrží hlášku o úspěšném přidání

Alternativní scénář:

1. Uživatel naviguje na detail filmu
2. Uživatel klikne na tlačítko “Přidat do oblíbených”
3. Systém pošle požadavek na uložení filmu do oblíbených
4. Film se nepovede uložit
5. Uživatel je informován o chybě a vyzván k opětovnému pokusu

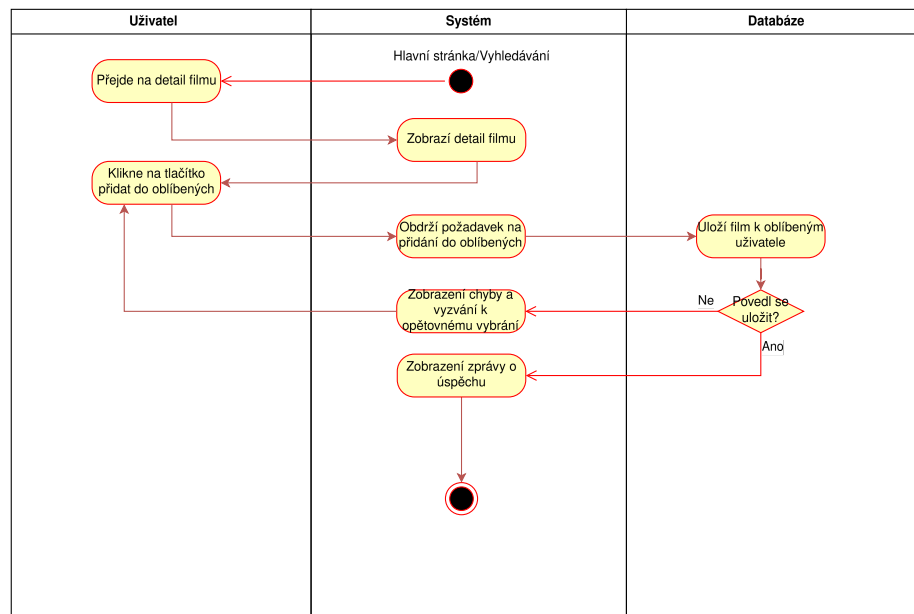


Figure 4: UC13 - Activity diagram

UC8: Přidání filmu**Parametry:**

- Název: Přidání filmu
- Kontext: Kontributor chce přidat nový film, který zrovna vyšel
- Primární aktér: Kontributor
- Předpoklad: Kontributor musí být přihlášen
- Minimální záruka: Chybové hlášky při chybě systému nebo při nedostatečném vyplnění formuláře
- Záruka úspěchu: Film je úspěšně přidán do databáze a zobrazen mezi ostatními filmy
- Trigger: Uživatelská akce

Hlavní scénář:

1. Kontributor naviguje na stránku pro správu filmů
2. Kontributor klikne na tlačítko pro přidání nového filmu
3. Otevře se formulář pro zadání informací o filmu
4. Kontributor vyplní povinné údaje (název, délka, žánr, rok, popis, herci...)
5. Kontributor odešle formulář
6. Systém uloží data do databáze
7. Kontributor obdrží hlášku o úspěšném přidání filmu

Alternativní scénář:

1. Kontributor naviguje na stránku pro správu filmů
2. Kontributor klikne na tlačítko pro přidání nového filmu
3. Otevře se formulář pro zadání informací o filmu
4. Kontributor neplní povinné údaje (název, délka, žánr, rok, popis, herci...)
5. Kontributor odešle formulář
6. Systém zobrazí chybu, že formulář musí být kompletní

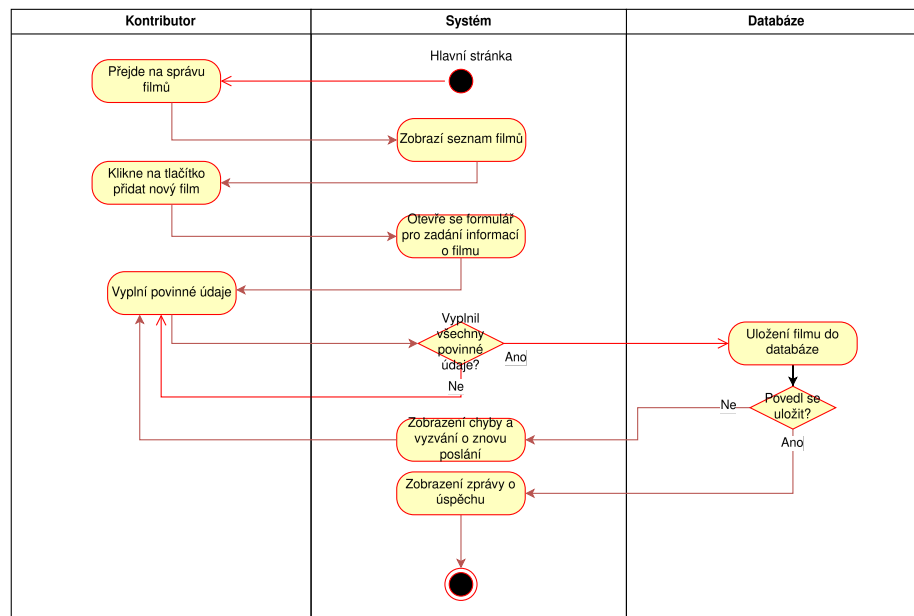


Figure 5: UC8 - Activity diagram

UC1: Registrace**Parametry:**

- Název: Registrace
- Kontext: Návštěvník si chce udělat na FilmDB účet, aby mohl hodnotit a recenzovat filmy
- Primární aktér: Návštěvník
- Předpoklad: -
- Minimální záruka: Chybové hlášky při chybě systému nebo při nedostatečném vyplnění formuláře
- Záruka úspěchu: Návštěvníkovi se úspěšně vytvoří účet a může se přihlásit
- Trigger: Uživatelská akce

Hlavní scénář:

1. Návštěvník naviguje na stránku pro registraci
2. Návštěvník vyplní všechny povinné údaje (uživatelské jméno, email, heslo)
3. Návštěvník klikne na tlačítko registrovat se
4. Systém uloží údaje do databáze
5. Návštěvník obdrží hlášku o úspěšném zaregistrování a je přesměrován na přihlašovací stránku

Alternativní scénář:

1. Návštěvník naviguje na stránku pro registraci
2. Návštěvník vyplní všechny povinné údaje (uživatelské jméno, email, heslo)
3. Návštěvník klikne na tlačítko registrovat se
4. Systém zjistí, že již uživatel se stejným uživatelským jménem/emailem již existuje
5. Návštěvník obdrží hlášku o této informaci a je vyzván vybrat jiné uživatelské jméno/heslo

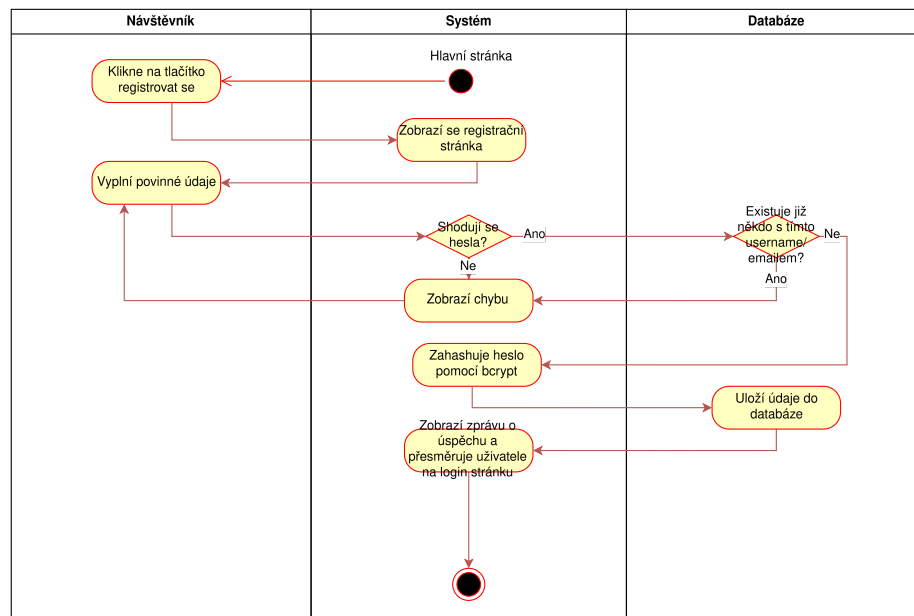


Figure 6: UC1 - Activity diagram

Technická specifikace

Doménový diagram

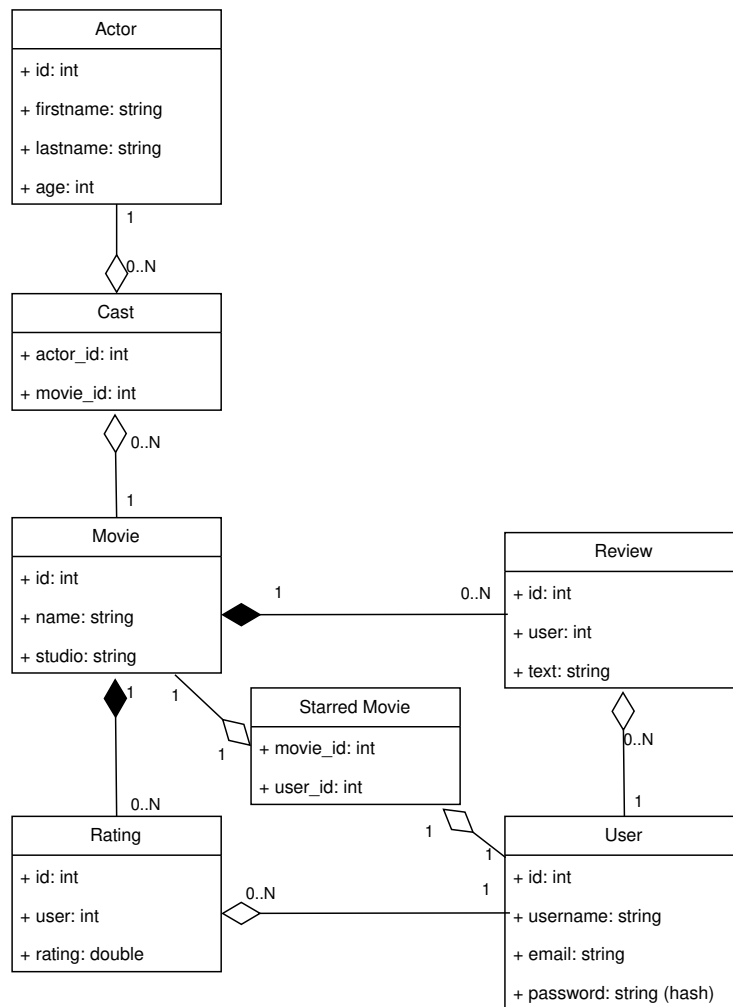


Figure 7: Domain Diagram

Odhady velikostí

Jméno	Počet	Velikost jedné položky	Velikost všech (MB)
Actor	500	518 Bytes	0.25 MB
Movie	1000	514 Bytes	0.51 MB
Cast	10000	8 Bytes	0.08 MB
Rating	1000000	16 Bytes	15.26 MB
Review	100000	508 Bytes	48.44 MB
User	50000	578 Bytes	27.56 MB
Starred Movie	25000	8 Bytes	2.00 MB

Celková velikost za rok: **94.1 MB**

Uložení dat

Všechna data budou uložena v databázi MariaDB.

Interakce Aktérů

- Návštěvník: Přístup přes webový prohlížeč, může vyhledávat a procházet filmy.
- Uživatel: Přihlásí se do aplikace, hodnotí filmy a píše recenze.
- Příspěvatel: Přihlásí se do aplikace, spravuje filmy: přidává nové, upravuje obrázky u stávajících...

Použité technologie

- Databáze: MariaDB
- Frontend: Svelte
- Backend: SvelteKit (NodeJS)
- Hosting: Linuxový server
- Vývoj: Git + Github

Návrh

Hlavní stránka

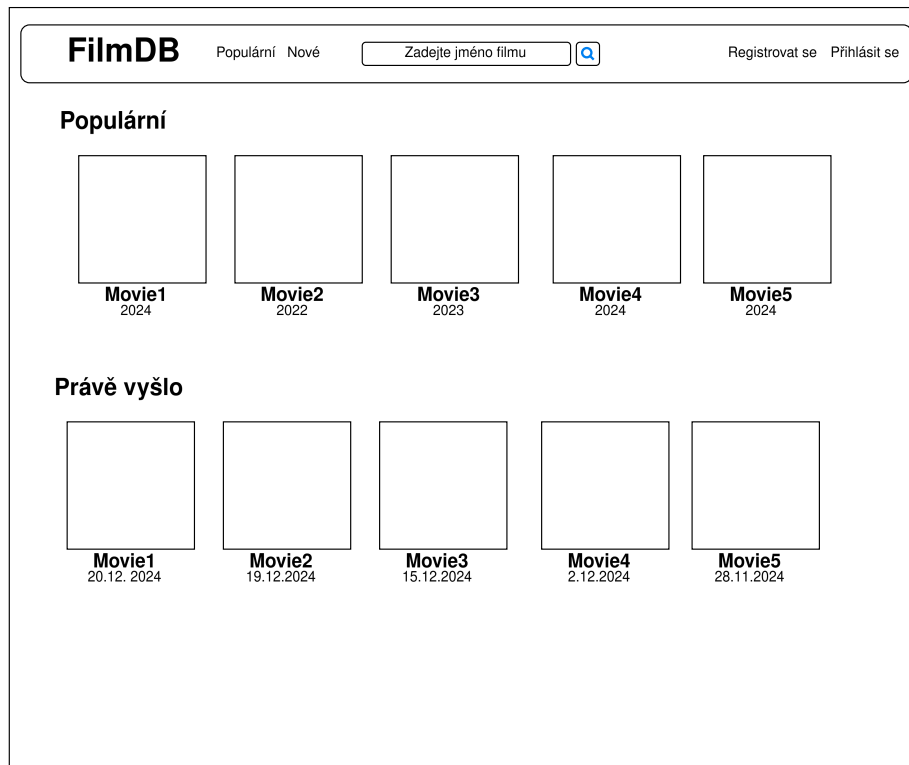


Figure 8: Home Page

[illegible]

Figure 9: Search

Detail filmu

FilmDB

Populární Nové

Registrovat se Přihlásit se

Movie1
Drama, ..., ...

Herci:
aaaaa aaaaa, bbbbbb bbbbbb, cccccc cccccc, ddddd ddddd,
eeeeee eeeee

4.89 / 5.0

Hodnotit

Pepa 2.5
Honza 4.0
Karel 5.0
Lukáš 1.5

Popis
Lorem ipsum, ... gdf.sgsd, sdgmgksdm sdgkkdg sklsdgmgd klsdgmmsdgks md
dgsdg sdgksdg msdg kmddg sksdg mlsdg msdg mklmsdg g sdsdg

gsngosdndsg gsd njksdg jsdg njksdg njkdgs nsdg jknsdg jndg sjksdg
dg sdg sdg sdg sdg dg ssdg


Recenze

Pepa - 11:51 15.12.2024 - 4.5/5.0
Lorem ipsum gisdmgdsd sdg sdnjgs n sdndg nsdgngs nsdg nnsdg j





Honza - 18:36 10.12.2024 - 5.0/5.0
Lorem ipsum gsdgisdmgdsd sdg gsd sgd njsdgnksdg nsdgn sdg nj sdnjgs n sdndg nsdgngs nsdg nnsdg j
sdg gsdgsd sdgg sdgmgklsd msdgkdg msklsdgkldg mgklssdg

Figure 10: Movie detail

Správa filmů

FilmDB Populární Nové  patrick115 (Contributor)

Správa Filmů Přidat

	Movie1	Vydáno	Recenzi	Hodnocení	Přidáno	Poslední úprava	
	4.8/5.0 Drama, Horror,...	15.12.2024	48	101	12:48 1.12.2024	1:11 2.12.2024	Editovat
	4.8/5.0 Drama, Horror,...	15.12.2024	48	101	12:48 1.12.2024	1:11 2.12.2024	Editovat
	4.8/5.0 Drama, Horror,...	15.12.2024	48	101	12:48 1.12.2024	1:11 2.12.2024	Editovat
	4.8/5.0 Drama, Horror,...	15.12.2024	48	101	12:48 1.12.2024	1:11 2.12.2024	Editovat

1 2 3 ... 45

Figure 11: Movie manage

Návrh doménového modelu

Použité návrhové vzory

- Identity field
- Gateway
- Domain model
- Service layer
- Data Transfer Object

Třídni diagram

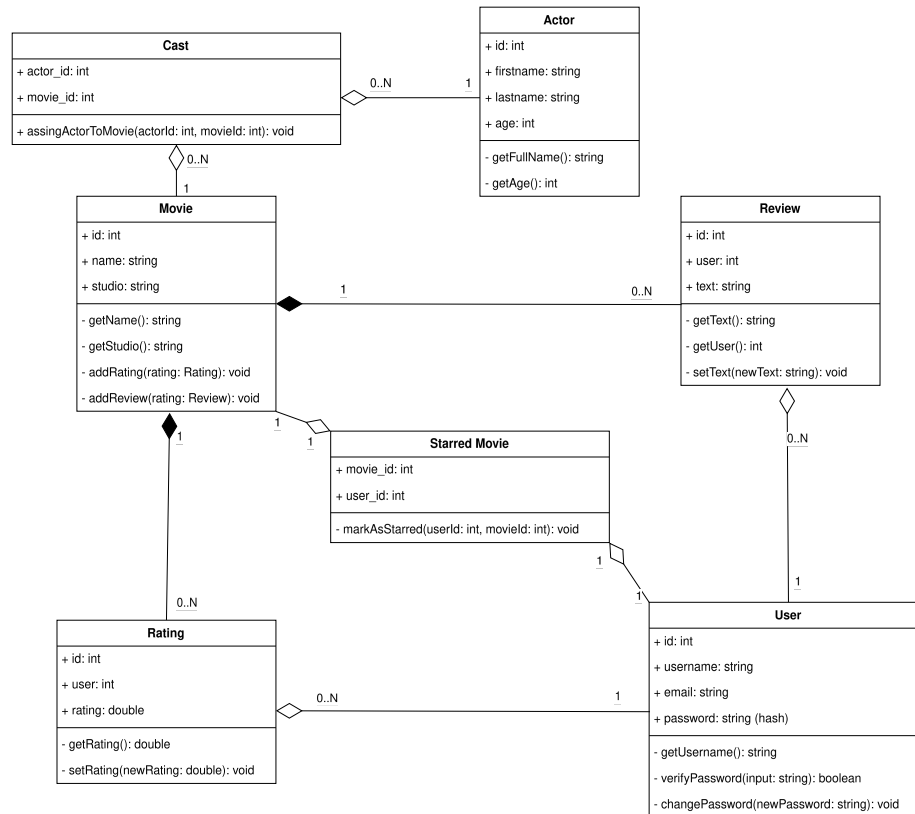


Figure 12: Class Diagram

Sekvenční diagramy

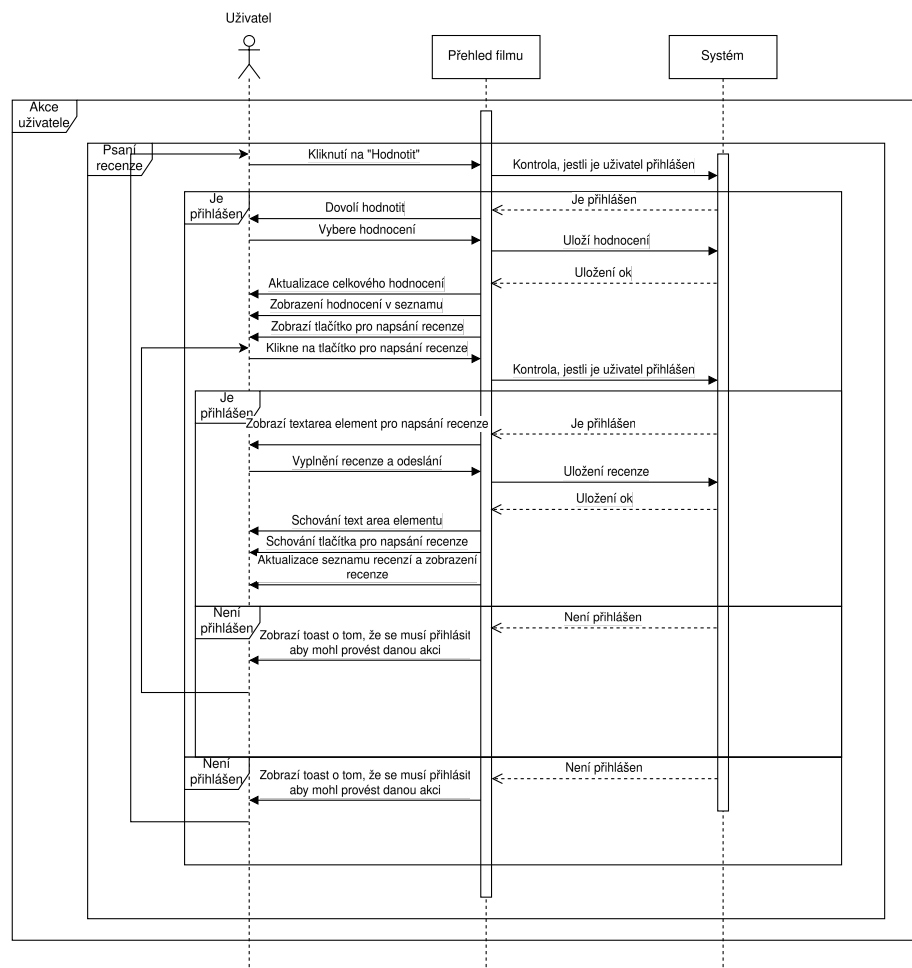


Figure 13: Sequence diagram 1

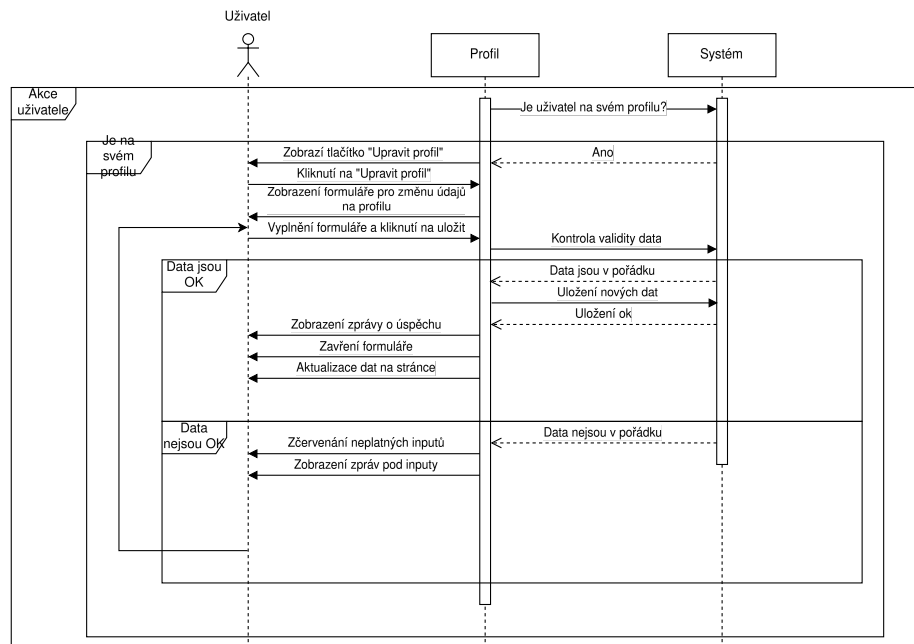


Figure 14: Sequence diagram 2

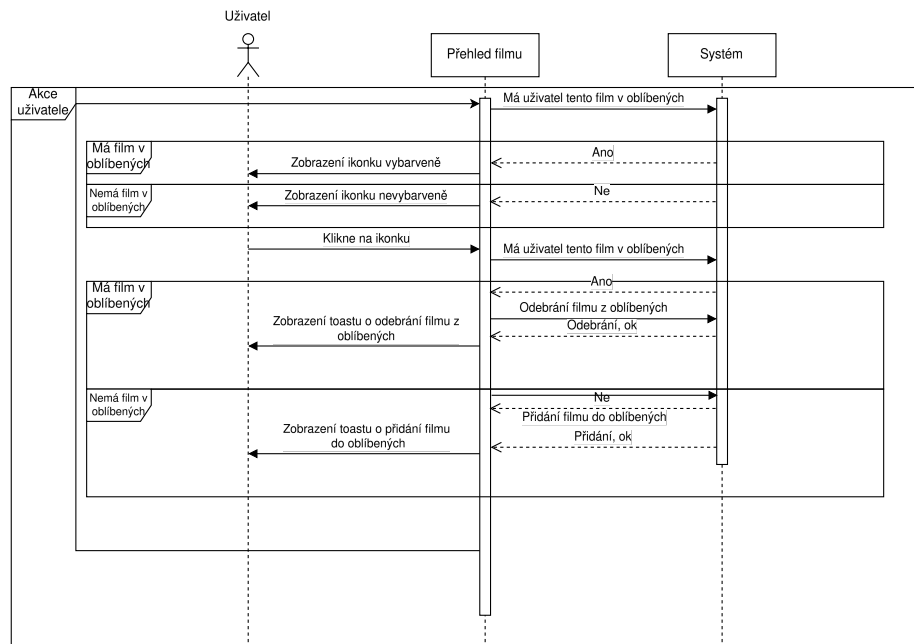


Figure 15: Sequence diagram 3

Popis architektury systému

Rozložení systému

Backend (3-vrstvá architektura)

1. Datová vrstva:
 - Přístup k databázi
 - Třídy: UserDTO, MovieDTO...
 - Technologie: MariaDB(Kysely pro typesafe sql)/SimpleJsonDB
2. Domenová vrstva:
 - Obsahuje doménové modely
 - Třídy: UserDomainModel, MovieDomainModel...
3. Service vrstva:
 - Obsahuje aplikační logiku
 - Třídy: UserService, MovieService...

Frontend (2-vrstvá architektura)

1. Gateway vrstva:
 - Poskytuje propojení mezi frontendem a backendem přes API
 - Třídy: UserGateway, MovieGateway
2. Service vrstva:
 - Zajišťuje logiku na stráně klientské strany (přebírání dat z gateway vrstvy a zpracování dat)
 - Třídy: UserService, MovieService...

Rozložení a propojení fyzických vrstev

- Frontend (UI):
 - Běží v prohlížeči uživatele
 - Technologie: Svelte, Typescript
- Backend (Serverová část):
 - Poběží na aplikačním serveru (SvelteKit - NodeJS adapter)
 - Poskytuje data pomocí API
- Databáze:
 - Běží samostatně na RaspberryPI
 - Technologie: MariaDB
- API:
 - Vlastní knihovna pro routování SvelteKit požadavků a handlování
 - Jednoduchý koncept implementující MiddleWary a procedury které poté komunikují s Service Vrstvou na Backendu

Diagram komponent

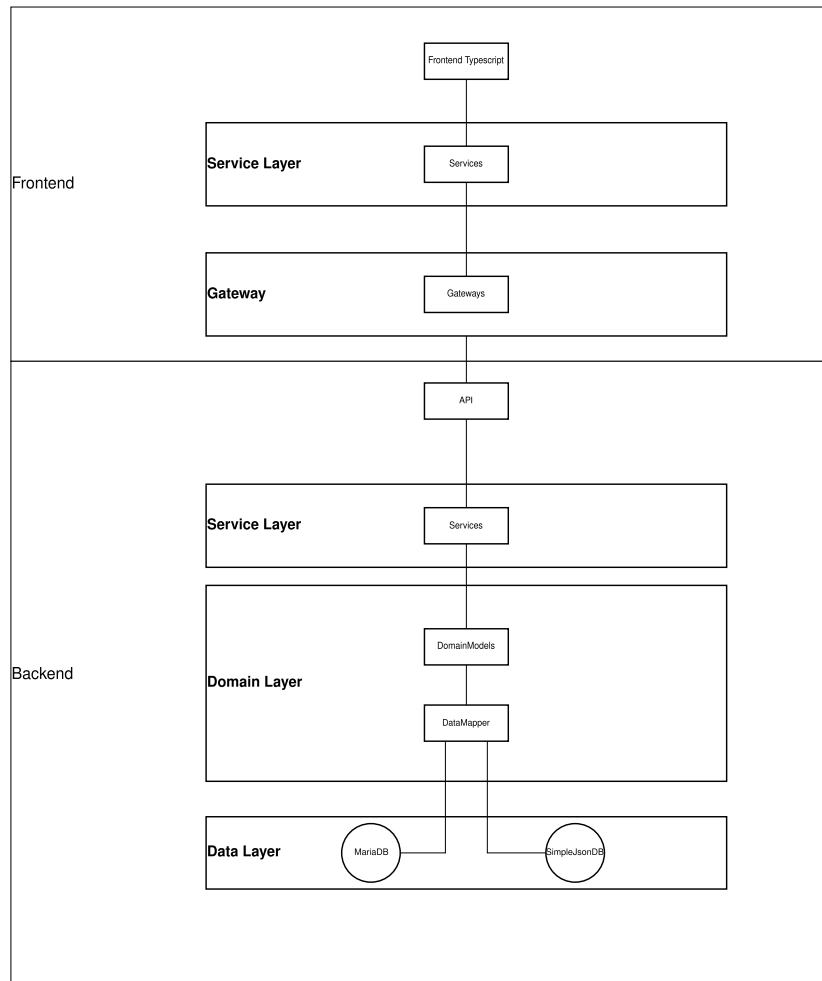


Figure 16: Component diagram