



**FAKULTA
INFORMAČNÍCH
TECHNOLOGIÍ
ČVUT V PRAZE**

BijákIS

Návrh architektury a databázový model

Dokument vytvořen pro potřeby předmětu BI-SI1

Autoři:
Denis Drda
Kryštof Šádek
Ladislav Kougl
Mark Sobolev
Martin Kostrubanič

Obsah

1. Návrh architektury	3
1.1 business_tier	3
1.1.1 enums.....	4
1.1.2 managers.....	4
1.2 data_tier	5
1.2.1 data_mappers	5
1.2.2 models.....	5
1.3 presentation_tier.....	5
1.3.1 Class urls.py.....	6
1.3.2 forms.....	6
1.3.3 templates	6
1.3.4 views.....	6
2. Relační datový model.....	6
2.1 Evidence lístků	7
2.1.1 Lístek «table»	8
2.1.2 Platba «table»	8
2.2 Evidence představení.....	8
2.2.1 Film «table»	9
2.2.2 Představení «table».....	9
2.2.3 Sal «table».....	9
2.2.4 Sedadlo «table»	9
2.3 Základ.....	9
2.3.1 Role «table»	10
2.3.2 Zamestnanec «table».....	11



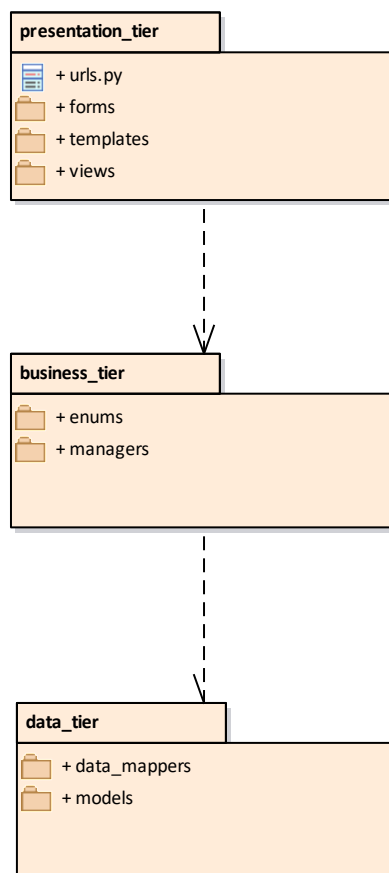
1. Návrh architektury

Aplikace je navržena jako třívrstvá architektura. Pro implementaci celé aplikace je použití frameworku Django. Rozhraní (interface) mezi jednotlivými vrstvami zajišťuje přímo Django. Používaná databáze aplikace je PostgreSQL. Pomocí jazyku Jinja2, který je moderní šablonovací systém, prezentujeme data uživateli.

Rozhraní umožní snadnou výměnu jednotlivých vrstev. Tato výměna může být požadována např. z důvodu změny uživatelského rozhraní na webovou aplikaci, výměny datové úložiště nebo pro potřeby otestování funkčnosti jednotlivých vrstev odděleně. Architektura také usnadňuje spolupráci více osob na projektu, kdy každý implementuje svojí část aplikace proti tomuto rozhraní.

Použité knihovny a frameworky:

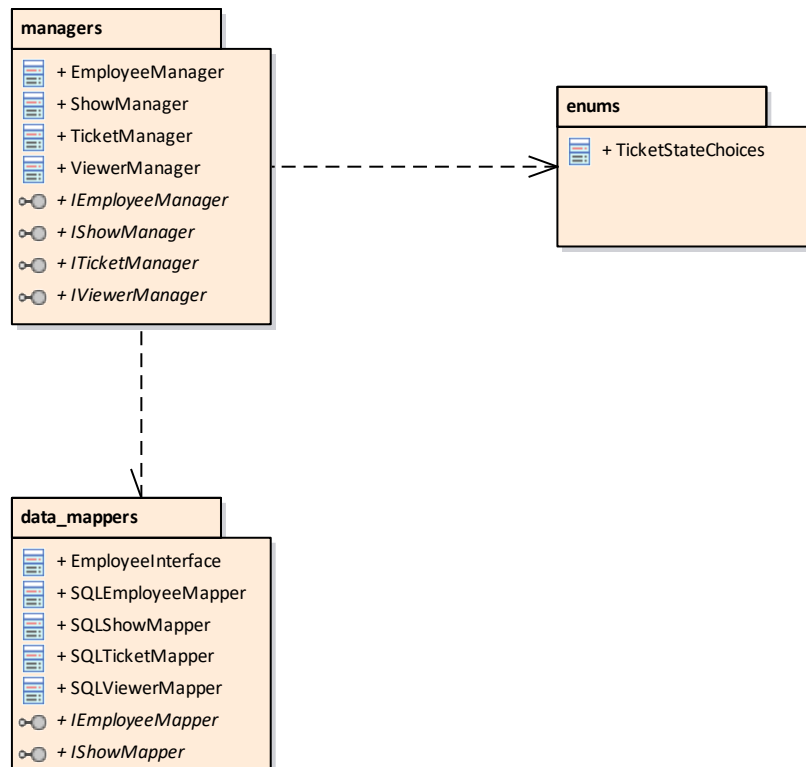
- Django (<https://www.djangoproject.com/>)
- Jinja2 (<https://jinja.palletsprojects.com/en/2.10.x/>)
- PostgreSQL (<https://www.postgresql.org/>)



Obrázek 1 - Návrh architektury

1.1 business_tier

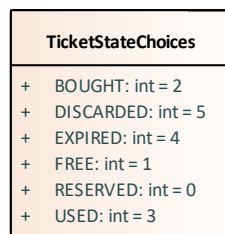
Veškerá business logika, kterou je nutné ze zadání implementovat, bude obsažena v balíčcích v této vrstvě. Spojení s datovou vrstvou zajišťují třídy v models.py.



Obrázek 2 - business_tier

Balíček, který implementuje logiku byznys vrstvy.

1.1.1 enums

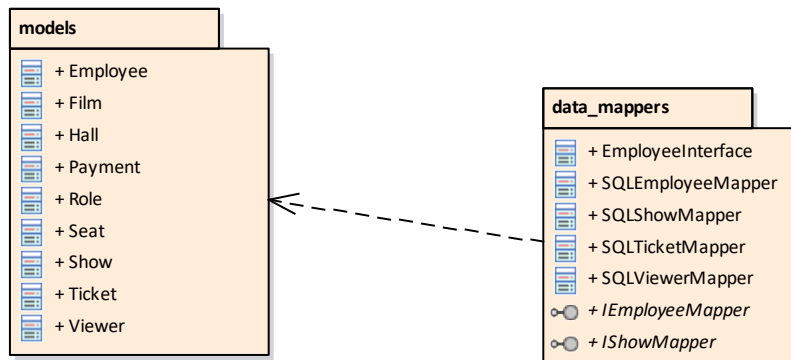


Obrázek 3 - enums

1.1.2 managers

1.2 data_tier

Datová vrstva aplikace, která bude obsahovat třídu umožňující persistentně ukládat /načítat objekty z datového úložiště.



Obrázek 4 - data_tier

1.2.1 data_mappers

Balíček, který poskytuje interface mezi business a datovou vrstvou

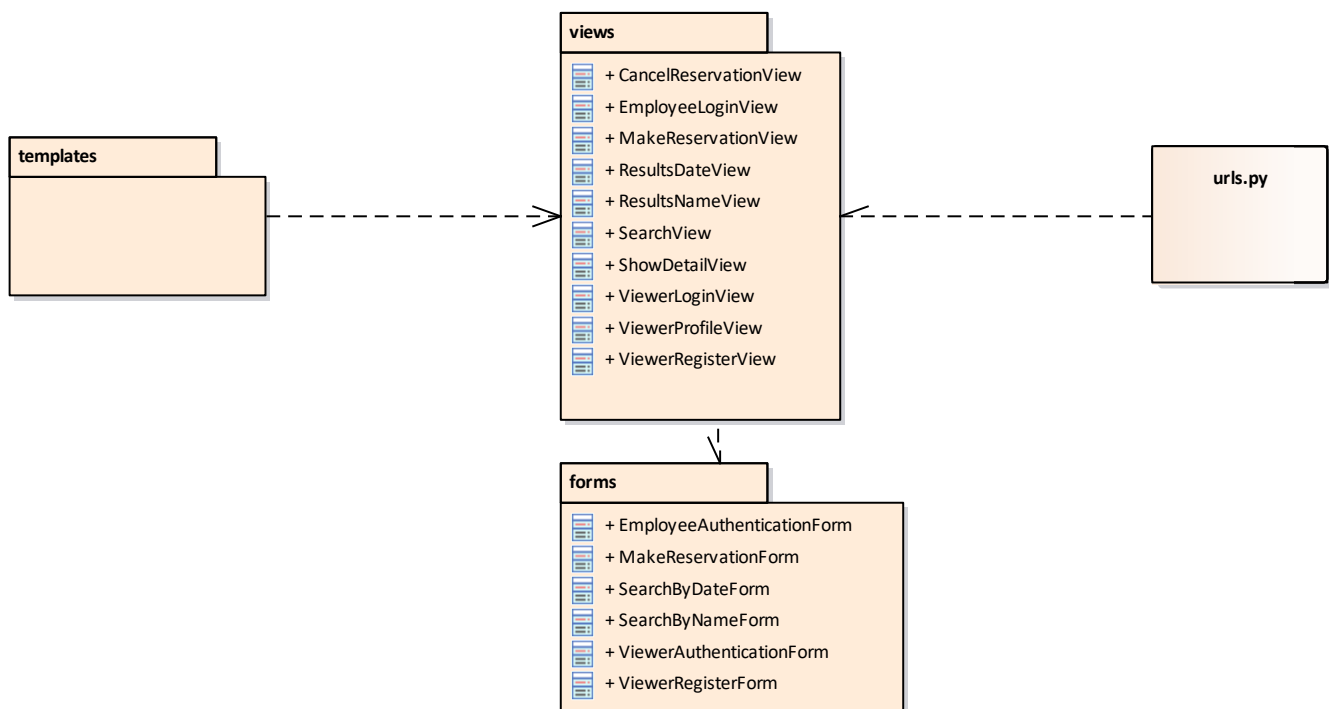
1.2.2 models

Balíček, který pracuje s datovými zdroji, je určen pro třídy, které poskytují interface do databáze.

1.3 presentation_tier

Prezentační vrstva aplikace umožňuje zobrazování stavu aplikace uživateli a předávání uživatelských požadavků do business vrstvy. Tyto požadavky třídy v prezentační vrstvě transformují tak, aby odstranily závislost na použité technologii (desktopová aplikace, http request).

Oddělení prezentační vrstvy se skládá ze tří základních částí templates, view a urlpatterns které reprezentují stejnojmenné balíčky. Třídy view z tohoto balíčku komunikují s třídami v business vrstvě pomocí rozhraní



Obrázek 5 - presentation_tier

1.3.1 Class urls.py

Django parsuje obsah souborů `urls.py` a používá ho k překladu URL na volání konkrétních tříd z `views.py`.

1.3.2 forms

Balíček je určen pro třídy, slouží pro validaci dat, který zajišťuje způsob přidávání a upravy příspěvků a dále také správu webových stránek.

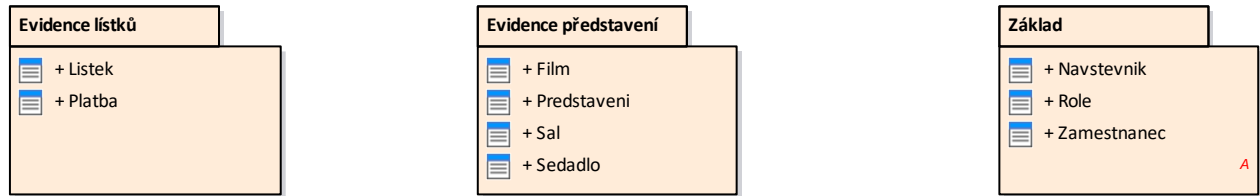
1.3.3 templates

Templates obsahují HTML šablony s podporou jazyka `jinja2`, které prezentují uživateli data doplněná z přiřazených tříd v souborech `views.py`.

1.3.4 views

Balíček je určen pro třídy, které přejímají uživatelský vstup předávajího do bysnys vrstvy, přistupuje k bysnys vrstvě prostřednictvím `models.py` a následně generuje výslednou stránku do HTML vzoru z `templates`.

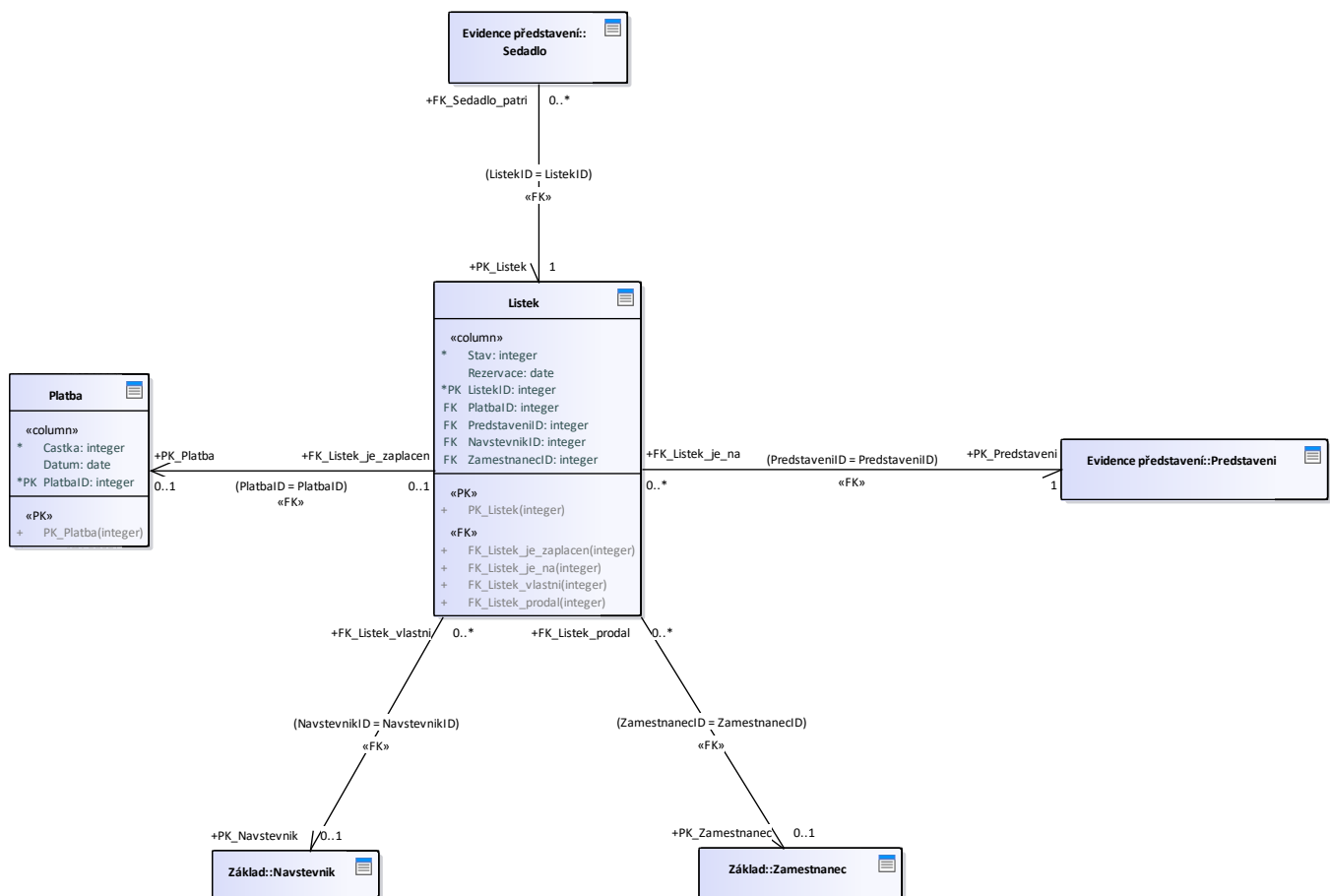
2. Relační datový model



Obrázek 6 - Relační datový model

Zde je popsán návrh ukládání dat. Forma uložení je v podobě objektově-relační databáze PostgreSQL (<https://www.postgresql.org/>). Aplikace je rozdělena do tří provázaných balíčků, aby byla lépe viditelná zodpovědnost jednotlivých částí, ovšem pro všechny balíčky je použita jediná databáze, jelikož použitý framework registruje projekt jako celek.

2.1 Evidence lístků



Obrázek 7 - Evidence lístků

2.1.1 Lístek «table»

Prostředek, jak se návštěvník může dostat do kina na film.

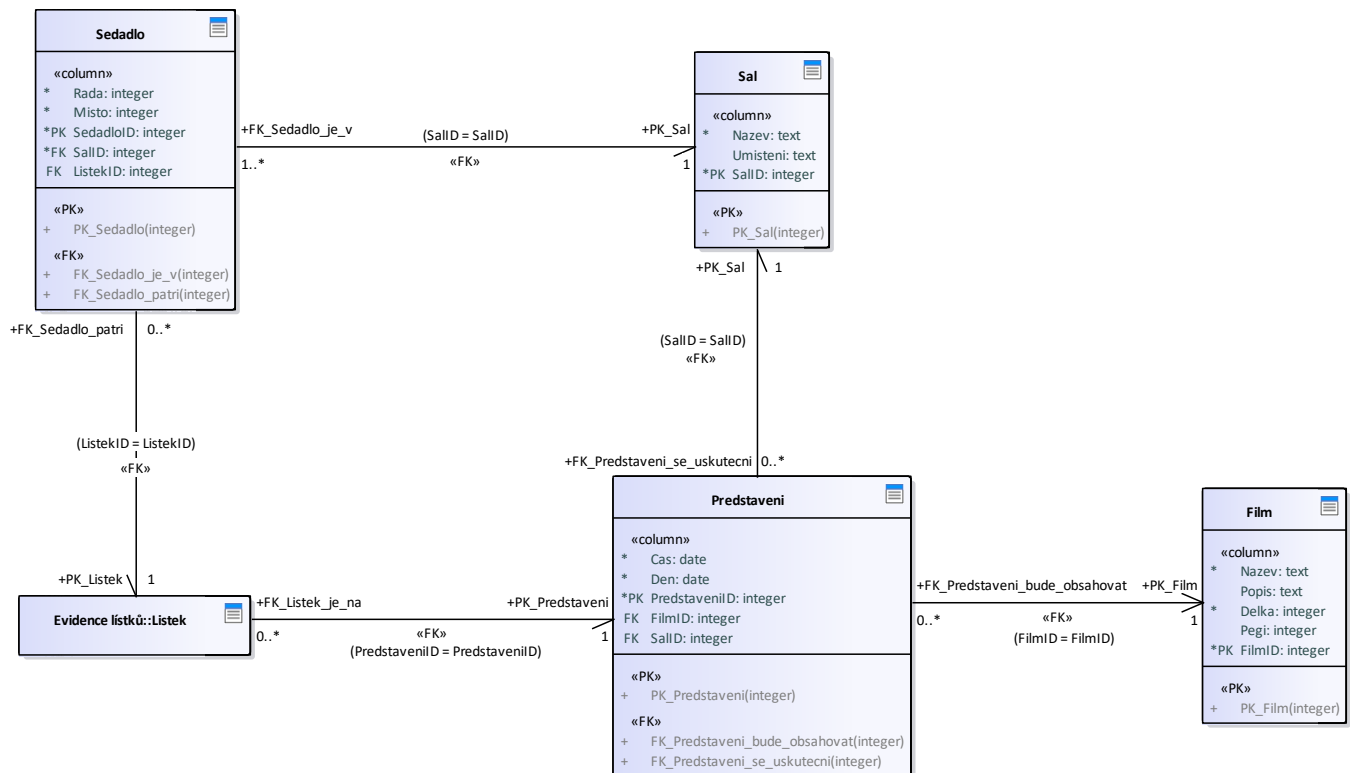
Název atributu	Datový typ	Not null	Popis
Stav	integer	True	Určuje stav lístku: nový, rezervovaný, zaplacený, využitý, propadlý,...
Rezervace	date	False	Datum, do kdy je lístek rezervovaný
ListekID	integer	True	
PlatbaID	integer	False	
PredstaveniID	integer	False	
NavstevnikID	integer	False	
ZamestnanecID	integer	False	

2.1.2 Platba «table»

Zaplacení za lístek.

Název atributu	Datový typ	Not null	Popis
Castka	integer	True	Kolik peněz stál lístek
Datum	date	False	Datum platby za lístek
PlatbaID	integer	True	

2.2 Evidence představení



Obrázek 8 - Evidence představení



2.2.1 Film «table»

Souvislá řada snímků určená k promítání v kině.

Název atributu	Datový typ	Not null	Popis
Nazev	text	True	Název filmu
Popis	text	False	Stručný popis filmu
Delka	integer	True	Doba trvání filmu
Pegi	integer	False	Systém věkového hodnocení
FilmID	integer	True	

2.2.2 Predstaveni «table»

Produkce v promítání filmů.

Název atributu	Datový typ	Not null	Popis
Cas	date	True	Časový údaj, kdy začíná představení
Den	date	True	Den, ve kterém se koná představení
PredstaveniID	integer	True	
FilmID	integer	False	
SalID	integer	False	

2.2.3 Sal «table»

Konkrétní místo, kde se promítá představení.

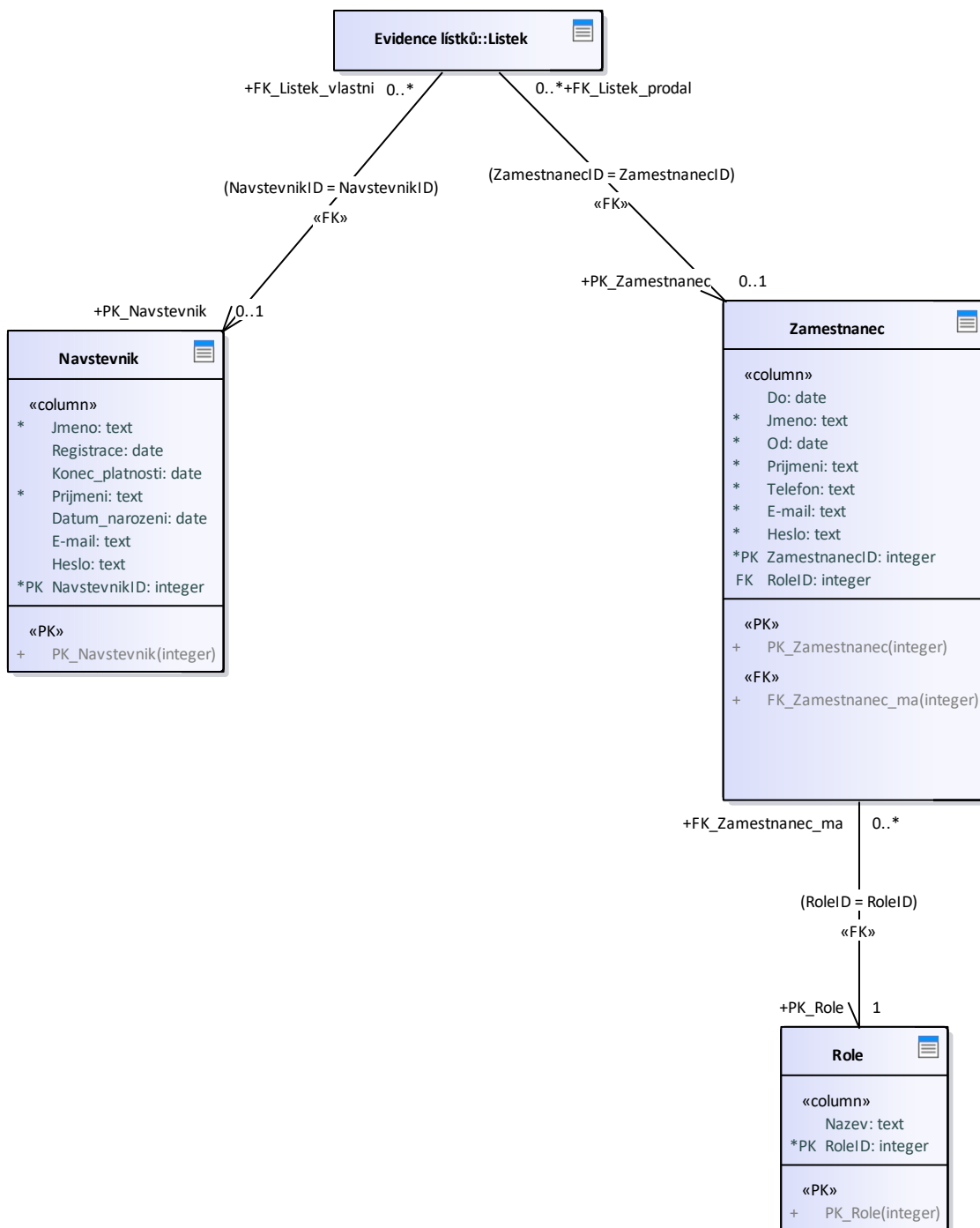
Název atributu	Datový typ	Not null	Popis
Nazev	text	True	Pojmenování sálu
Umisteni	text	False	Bližší informace o umístění sálu
SalID	integer	True	

2.2.4 Sedadlo «table»

Konkrétní sedadlo, kam si návštěvník sedne na promítání filmu.

Název atributu	Datový typ	Not null	Popis
Rada	integer	True	Řada, ve které bude návštěvník sedět
Misto	integer	True	Místo, ve kterém návštěvník bude sedět
SedadloID	integer	True	
SalID	integer	True	
ListekID	integer	False	

2.3 Základ



Obrázek 9 - Základ

2.3.1 Role «table»

Třída, na kterou se váží oprávnění v systému. Každý zaměstnanec má právě jednu roli, která se může měnit.



Název atributu	Datový typ	Not null	Popis
Nazev	text	False	Pojmenování role
RoleID	integer	True	

2.3.2 Zamestnanec «table»

Osoba, která pracuje ve společnosti kina.

Název atributu	Datový typ	Not null	Popis
Do	date	False	Do jakého data je osoba zaměstnána.
Jmeno	text	True	Jméno zaměstnance.
Od	date	True	Od jakého data je osoba zaměstnána.
Prijmeni	text	True	Příjmení zaměstnance.
Telefon	text	True	Kontakní telefonní číslo na zaměstnance.
E-mail	text	True	Kontakt na zaměstnance. Současně slouží k přihlašování do systému.
Heslo	text	True	Tajné heslo sloužící k přihlášení do systému.
ZamestnanecID	integer	True	
RoleID	integer	False	



**FAKULTA
INFORMAČNÍCH
TECHNOLOGIÍ
ČVUT V PRAZE**