

Informační systém pro knihovnu
Návrhová dokumentace (Design)

Ukázkový projekt pro předmět 36SIN

Vypracoval: Jiří Mlejnek

Obsah

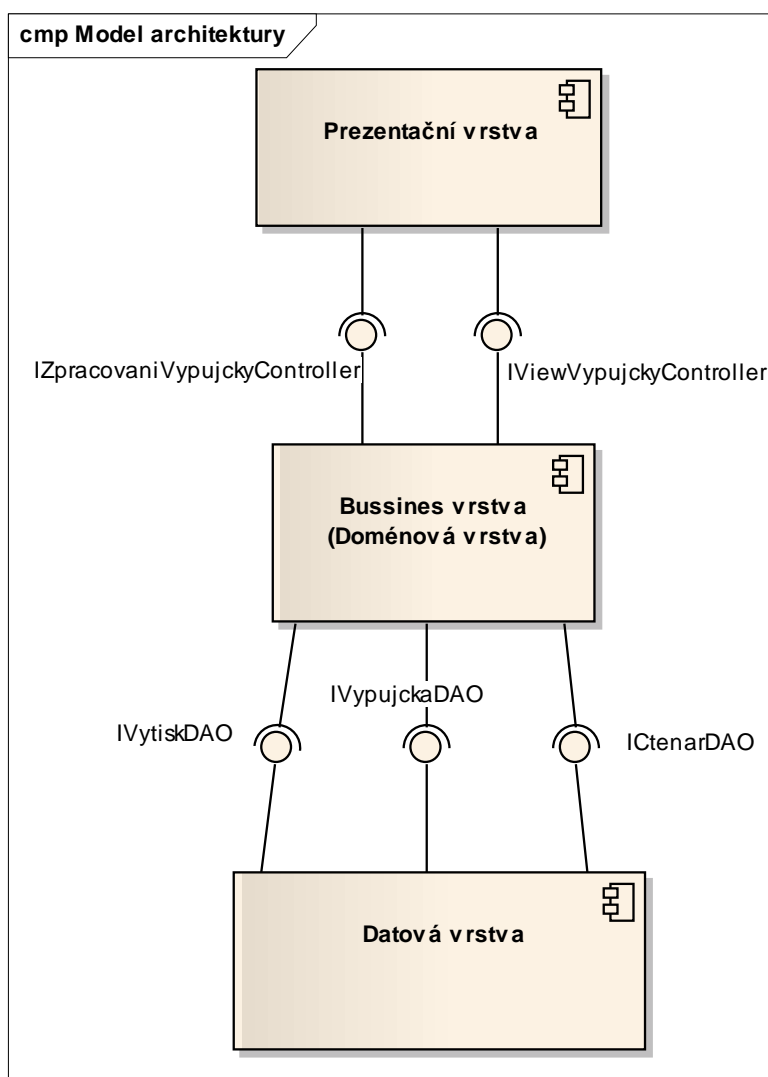
Návrhová dokumentace (Design)	1
Model architektury	4
Prezentační vrstva	4
Bussines vrstva (Doménová vrstva)	5
Datová vrstva	5
Balíček "Návrhový model"	5
Balíček "knihovna"	5
Balíček "pl"	6
Třída "KonstantyUI"	7
Balíček "controller"	7
Balíček "view"	8
Balíček "bl"	8
Balíček "vypujcky"	9
Třída "KnihaVypujcek"	13
Třída "SpravceCtenaru"	13
Třída "SpravceVytisku"	14
Třída "Stav"	14
Třída "StavVolny"	15
Třída "StavVypujcen"	15
Třída "ViewVypujckyController"	15
Třída "VytiskDTO"	16
Třída "ZpracovaniVypujckyController"	17
Třída "IPohled"	18
Třída "IViewVypujckyController"	18
Třída "IZpracovaniVypujckyController"	19
Balíček "dl"	19
Balíček "dao"	20
Třída "IVypujckaDAO"	21
Balíček "hibernate"	21
Třída "CtenarDAO"	22
Třída "VypujckaDAO"	22
Třída "VytiskDAO"	23
Balíček "entity"	23
Model komunikace	24
Výpůjčky	24
Relační datový model	27
Evidence knih	28
Tabulka "Autor"	30
Tabulka "Kniha"	30
Tabulka "Nakladatelstvi"	30
Tabulka "Regal"	30
Tabulka "Zanr"	30
Evidence vypujcek	31

Tabulka "Ctenar"	33
Tabulka "Pokuta"	33
Tabulka "Rezervace"	33
Tabulka "Vypujcka"	33
Tabulka "Vytisk"	34
Model nasazení	34
Instalace databáze	35
Konfigurace připojení	35
Spuštění aplikace	36

Model architektury

Architektura aplikace je navržena jako třívrstvá. Rozhraní mezi jednotlivými vrstvami je definováno pomocí rozhraní (interface). Přesná definice těchto rozhraní umožňuje 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 funkcí jednotlivých vrstev odděleně. Architektura také usnadňuje spolupráci více osob na projektu, kdy každý implementuje svojí část aplikace proti rozhraní bez ohledu na jeho konkrétní implementaci. Architektura aplikace je znázorněna na následujícím obrázku.

Detailní popis metod jednotlivých navržených rozhraní (interfejsů) je možné nalézt v kapitole "Návrhový model".



Obrázek: 1

Prezentační vrstva

Prezentační vrstva obsahuje třídy a komponenty, které se starají o prezentování informací uživateli, a které od něho

zpátky přebírají požadavky. Tyto požadavky transformují tak, aby odstranily závislost na použité technologii (desktopová aplikace, http request) a předávají je dále do doménového modelu.

Bussines vrstva (Doménová vrstva)

Komponenta obsahuje třídy a komponenty, které realizují požadované chování aplikace z pohledu obchodní logiky.

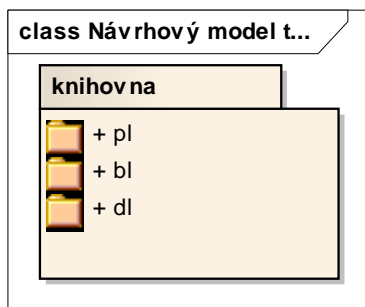
Datová vrstva

Komponenta obsahuje třídy a komponenty poskytující technickou podporu pro doménovou vrstvu v podobě persistentního ukládání dat apod.

Balíček "Návrhový model"

Kapitola obsahuje statický popis systému. Obsahuje detailní popis jednotlivých tříd, rozhraní a balíčků, které realizují tento systém. Kapitoly v této knize přesně odpovídají názvům java packages, které jsou použity pro seskupování zdrojových kódů tříd.

Pro implementaci celé aplikace je navrženo použití frameworku Spring, usnadňující vývoj enterprise aplikací. V tomto projektu je využito především vzoru IoC, který umožňuje provázat jednotlivé komponenty (třídy) deklarativním způsobem v konfiguraci aplikace, což umožňuje jednoduše nahradit, některé části za jiné, bez zásahu do zdrojových kódů, což může být vhodné například pro potřeby testování.

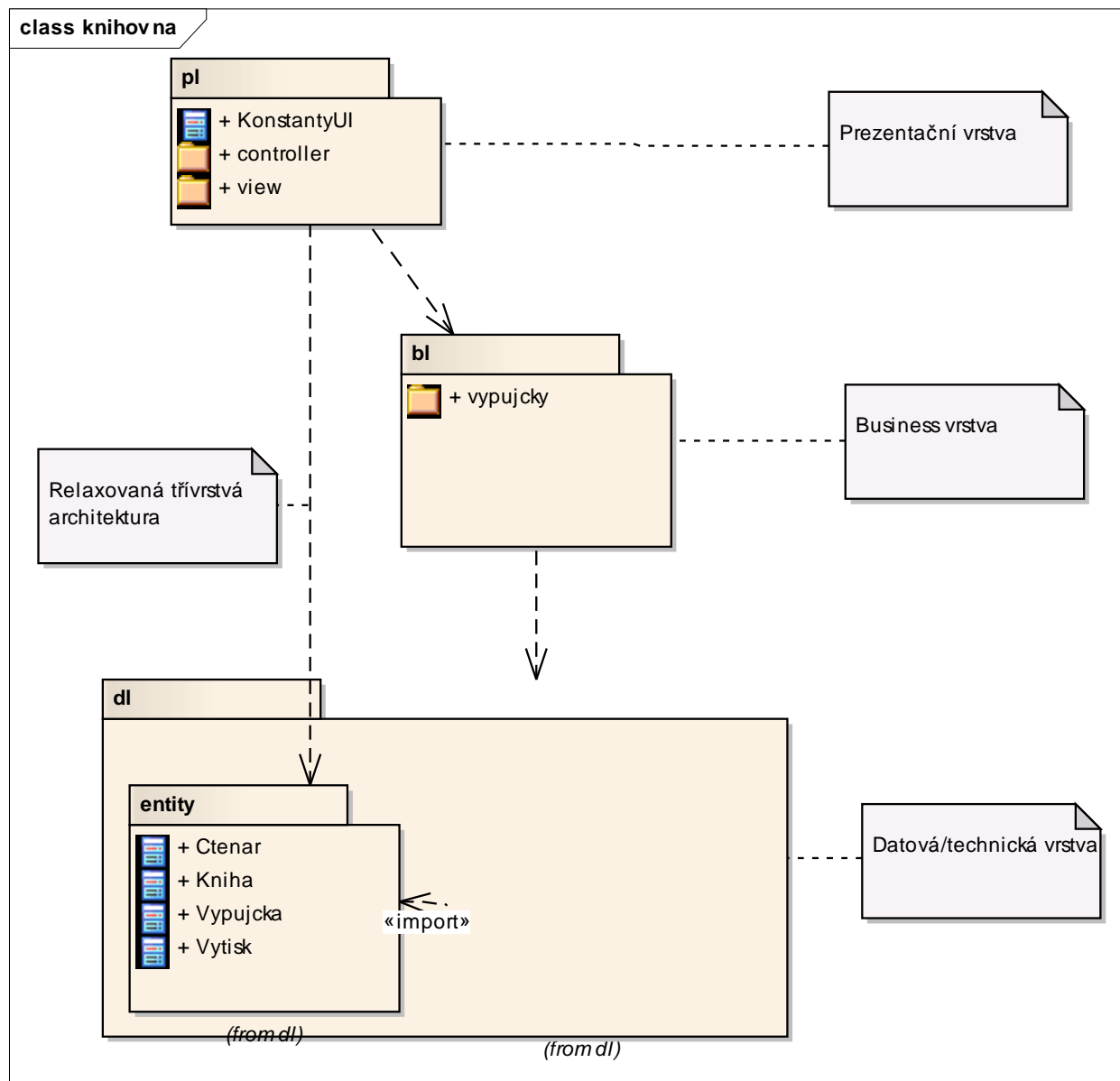


Obrázek: 2

Balíček "knihovna"

Balíček obsahuje třídy realizující ukázkovou implementaci IS knihovny pro potřeby výuky softwarového inženýrství. Celá aplikace je rozdělena do tří základních balíčků, které odpovídají jednotlivým vrstvám třívrstvé architektury. Obrázek dokumentuje závislosti mezi těmito balíčky.

Prezentační vrstva využívá kromě služeb balíčku business vrstvy, také přímo balíček entity, který je součástí datové vrstvy. Tato závislost umožňuje zjednodušení implementace, kdy není nutné na úrovni business vrstvy zapouzdřovat všechny objekty datové vrstvy do pomocných objektů. Jedná se o tzv. relaxovanou třívrstvou architekturu.

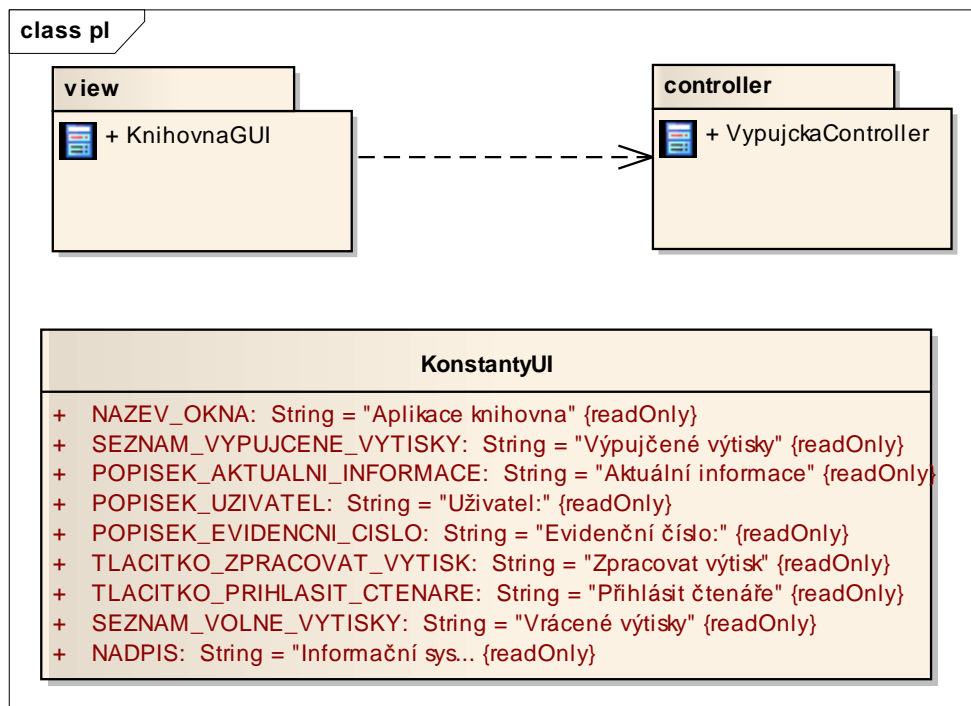


Obrázek: 3

Balíček "pl"

Balíček obsahuje třídy prezentační vrstvy, které umožňují zobrazování stavu aplikace uživateli a předávat uživatelské požadavky do business vrstvy.

Cílem tohoto projektu není vysvětlení principů návrhu uživatelského rozhraní, proto nejsou třídy v tomto balíčku popisovány detailněji.



Obrázek: 4

Třída "KonstantyUI"

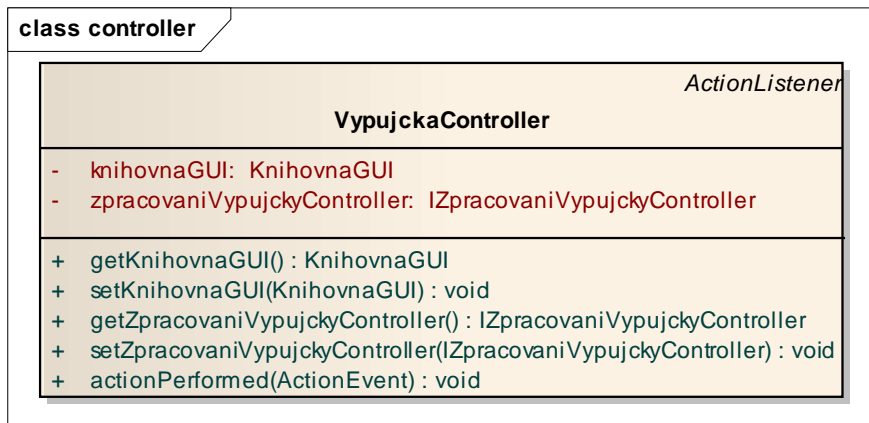
Třída obsahuje popisky komponent pro uživatelské rozhraní.

Atributy

Atributy	Datový typ	Poznámky
NAZEV_OKNA	String	
SEZNAM_VYPUJCENE_VYTISKY	String	
POPISEK_AKTUALNI_INFORMACE	String	
POPISEK_UZIVATEL	String	
POPISEK_EVIDENCNI_CISLO	String	
TLACITKO_ZPRACOVAT_VYTISK	String	
TLACITKO_PRIHLASIT_CTENARE	String	
SEZNAM_VOLNE_VYTISKY	String	
NADPIS	String	

Balíček "controller"

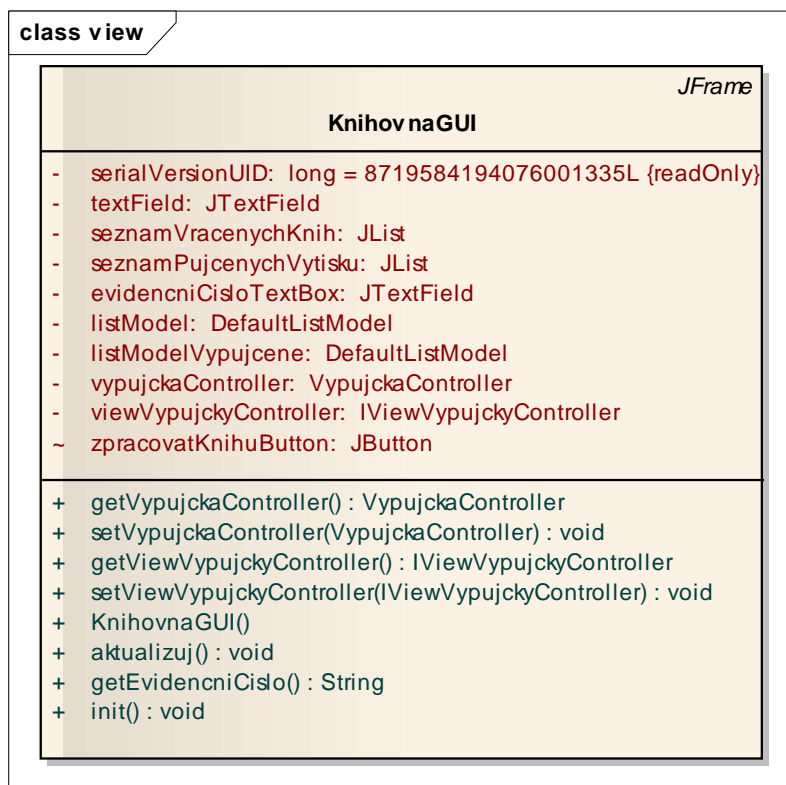
Balíček, který obsahuje třídy, které zprostředkovávají uživatelské vstupy pro třídy z business vrstvy. Jedná se o třídy označované jako controllers ve vzoru MVC.



Obrázek: 5

Balíček "view"

Balíček obsahuje třídy, které se starají o zobrazování informací uživatelům. Jedná se o třídy označované jako view ve vzoru MVC.

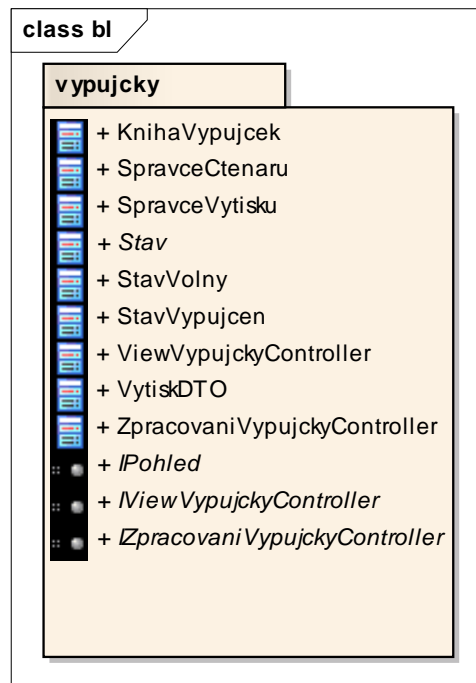


Obrázek: 6

Balíček "bl"

Balíček obsahuje třídy business vrstvy, které implementují vlastní chování celé aplikace. Obsahuje třídy, které

realizují veškeré chování aplikace související s obchodní logikou.

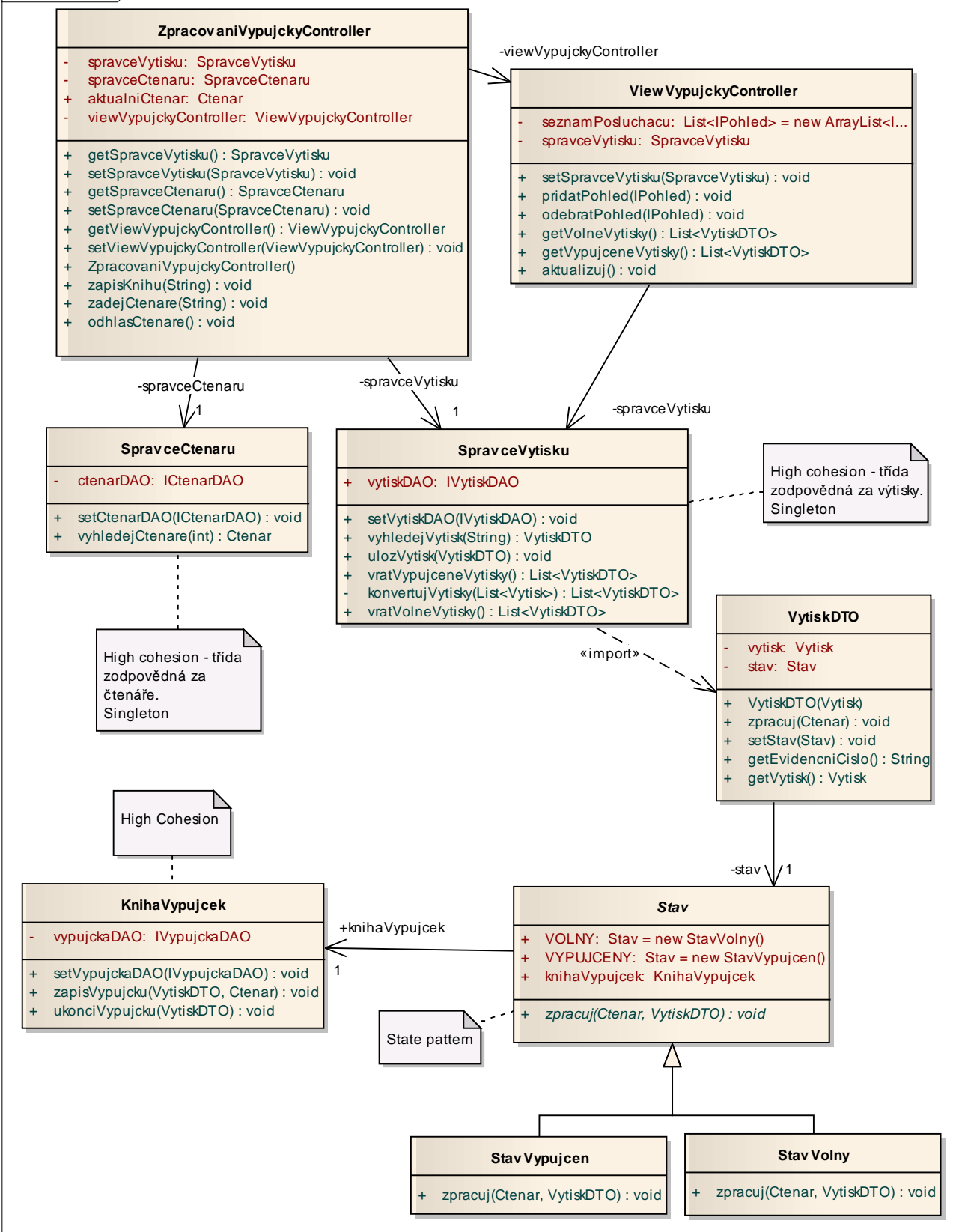


Obrázek: 7

Balíček "vypujcky"

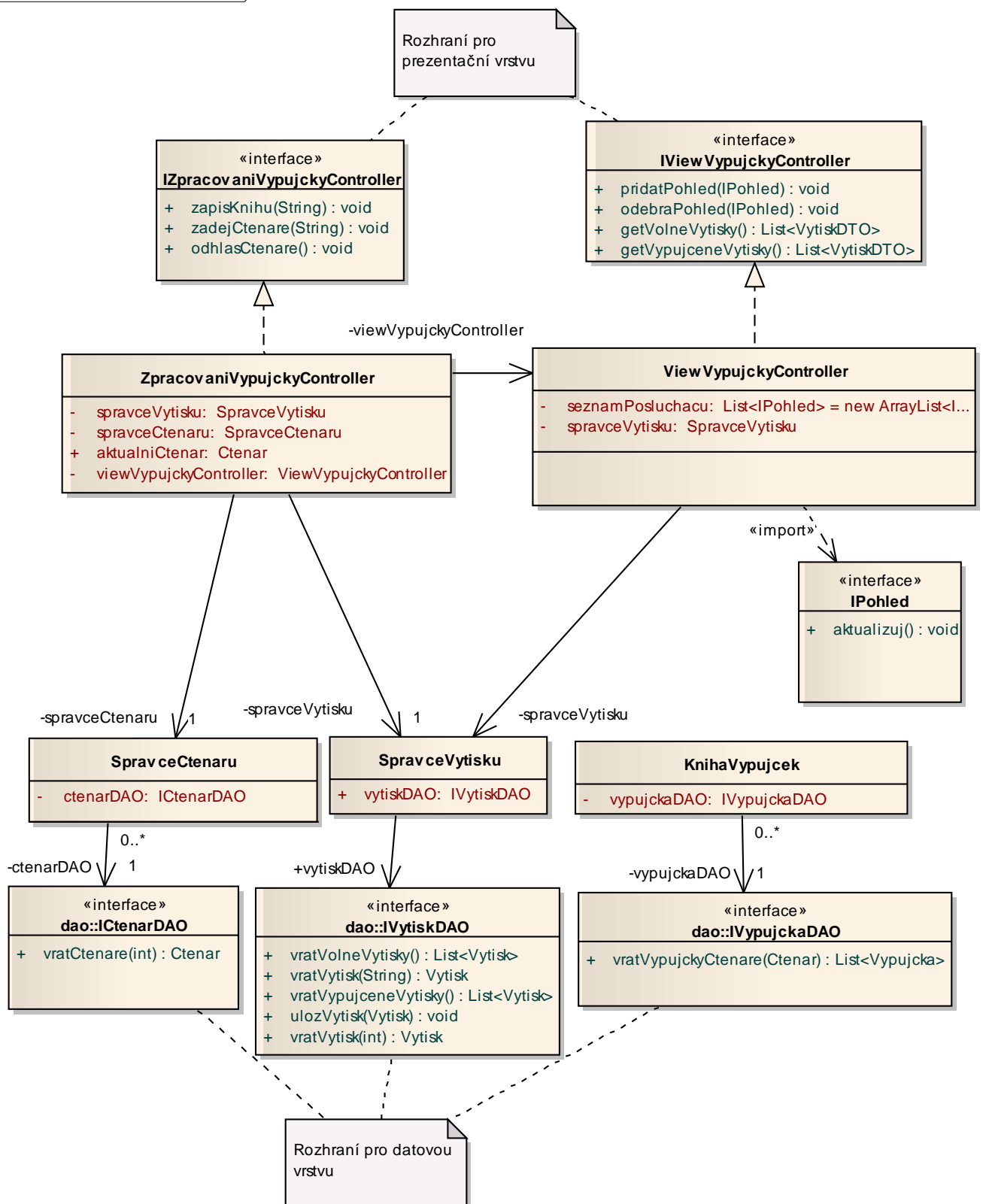
Balíček obsahující třídy, které umožňují evidenci výpůjček a vracení jednotlivých výtisků.

class Výpůjč...



Obrázek: 8

class Napojení na ostatní vrst...



Obrázek: 9

Na obrázku jsou zachyceny třídy, které realizují požadované rozhraní pro prezentační vrstvu a třídy, které vyžadují realizaci zobrazených interfejsů po datové vrstvě.

Třída "KnihaVypujcek"

Třída reprezentující knihu, do které se zapisují všechny výpůjčky čtenářů.

Atributy

Atributy	Datový typ	Poznámky
vypujckaDAO	IVypujckaDAO	Rozhraní umožňující persistentní ukládání záznamů o výpůjčkách.

Metody	Návratový typ	Poznámky
setVypujckaDAO	False	Umožňuje nastavit třídu implementující rozhraní pro persistentní ukládání záznamů o výpůjčkách. Parametry: <ul style="list-style-type: none"> vypujckaDAO: IVypujckaDAO Instance třídy implementující rozhraní IVypujckaDAO.
zapisVypujcku	False	Zapíše záznam o výpůjčení knihy čtenářem. Parametry: <ul style="list-style-type: none"> vytisk: VytiskDTO Výtisk, který si chce čtenář půjčit. ctenar: Ctenar Čtenář, který si půjčuje výtisk.
ukonciVypujcku	False	Zapíše záznam o vrácení výtisku knihy čtenářem. Parametry: <ul style="list-style-type: none"> vytisk: VytiskDTO Výtisk, který chce čtenář vrátit.

Třída "SpravceCtenaru"

Třída zapouzdřující logiku související s evidencí čtenářů.

Atributy

Atributy	Datový typ	Poznámky
ctenarDAO	ICtenarDAO	Rozhraní umožňující persistentní ukládání záznamů o čtenářích.

Metody	Návratový typ	Poznámky
setCtenarDAO	False	Umožňuje nastavit třídu implementující rozhraní ICtenarDAO Parametry: <ul style="list-style-type: none"> ctenarDAO: ICtenarDAO Instance třídy implementující rozhraní ICtenarDAO.
vyhledejCtenare	False	Vyhledá čtenáře, na základě jeho čísla průkazky. Parametry:

Metody	Návratový typ	Poznámky
		<ul style="list-style-type: none"> cisloPrukazky: int Číslo průkazky čtenáře.

Třída "SpravceVytisku"

Třída zapouzdřující logiku související s evidencí výtisků, jejich nákupem a vyřazováním z knihovny.

Atributy

Atributy	Datový typ	Poznámky
vytiskDAO	IVytiskDAO	Rozhraní umožňující persistentní ukládání záznamů o výtiscích.

Metody	Návratový typ	Poznámky
setVytiskDAO	False	Umožňuje nastavit třídu implementující rozhraní IVytiskDAO Parametry: <ul style="list-style-type: none"> vytiskDAO: IVytiskDAO Instance třídy implementující rozhraní IVytiskDAO.
vyhledejVytisk	False	Na základě evidenčního čísla vyhledá konkrétní výtisk Parametry: <ul style="list-style-type: none"> evidencniCislo: String Evidenční číslo výtisku, který má být vyhledán.
ulozVytisk	False	Trvale uloží změny ve výtisku. Parametry: <ul style="list-style-type: none"> vytisk: VytiskDTO Výtisk, který má být uložen.
vratVypujceneVytisky	False	Vrátí seznam výtisků, které jsou aktuálně vypůjčeny. Parametry:
konvertujVytisky	False	Převěde kolekci třídy Vytisk na kolekci třídy VytiskDTO. @return Kolekce převedených výtisků. Parametry: <ul style="list-style-type: none"> volneVytisky: List<Vytisk> Kolekce výtisků, která má být převedena.
vratVolneVytisky	False	Vrátí seznam výtisků, které jsou v knihovně k dispozici. @return Seznam volných výtisků Parametry:

Třída "Stav"

Třída reprezentuje stav, ve kterém se může každý výtisk nacházet. Tato třída je součástí vzoru "State pattern".

Atributy

Atributy	Datový typ	Poznámky
VOLNY	Stav	
VYPUJCENY	Stav	

Atributy	Datový typ	Poznámky
knihavypujcek	Knihavypujcek	

Metody	Návratový typ	Poznámky
zpracuj	False	Zpracuje zadaný výtisk. Zpracováním se rozumí zapsání jeho výpůjčky nebo ukončení výpůjčky, podle toho, zda byl výtisk půjčen nebo nebyl. Parametry: <ul style="list-style-type: none"> ctenar: Ctenar Čtenář, který o zpracování výtisku žádá. vytisk: VytiskDTO Výtisk, který má být zpracován.

Třída "StavVolny"

Potomek třídy Stav, realizuje požadované chování, které odpovídá tomuto stavu. Třída je součástí vzoru: "State pattern". Představuje stav výtisku, kdy je výtisk v knihovně k dispozici pro výpůjčení čtenáři.

Metody	Návratový typ	Poznámky
zpracuj	False	Zpracuje zadaný výtisk. Zapiše do knihy výpůjček nový záznam o výpůjčce a nastaví stav výtisku na výpůjčený. Parametry: <ul style="list-style-type: none"> ctenar: Ctenar Čtenář, který o zpracování výtisku žádá. vytisk: VytiskDTO Výtisk, který má být zpracován.

Třída "StavVypujcen"

Potomek třídy Stav, realizuje požadované chování, které odpovídá tomuto stavu. Třída je součástí vzoru: "State pattern". Představuje stav výtisku, kdy je výtisk výpůjčen některým z čtenářů.

Metody	Návratový typ	Poznámky
zpracuj	False	Zpracuje zadaný výtisk. Zapiše do knihy výpůjček datum ukončení výpůjčky a nastaví stav výtisku na volný. Parametry: <ul style="list-style-type: none"> ctenar: Ctenar Čtenář, který o zpracování výtisku žádá. vytisk: VytiskDTO Výtisk, který má být zpracován.

Třída "ViewVypujckyController"

Třída implementující interfejs umožňující připojení uživatelského rozhraní k doménové vrstvě. Třída je součástí vzoru "Observer".

Atributy

Atributy	Datový typ	Poznámky
----------	------------	----------

Atributy	Datový typ	Poznámky
seznamPosluchacu	List<IPohled>	Seznam objektů, které chtějí být informováni o změnách v doménové vrstvě.
spravceVytisku	SpravceVytisku	Třída zapouzdřující logiku související s evidencí výtisků.

Metody	Návratový typ	Poznámky
setSpravceVytisku	False	Umožňuje nastavit instanci třídy nabízející logiku související s evidencí výtisků. Parametry: <ul style="list-style-type: none"> spravceVytisku: SpravceVytisku spravceVytisku
pridatPohled	False	Umožňuje přidat do seznamu posluchačů nový objekt. Parametry: <ul style="list-style-type: none"> pohled: IPohled Objekt, který má být informován o změnách.
odebratPohled	False	Umožňuje odebrat objekt ze seznamu posluchačů. Parametry: <ul style="list-style-type: none"> pohled: IPohled Objekt, který má být odebrán.
getVolneVytisky	False	Vrátí seznam výtisků, které jsou v knihovně k dispozici. Parametry:
getVypujceneVytisky	False	Vrátí seznam výtisků, které jsou aktuálně vypůjčeny některým z čtenářů. Parametry:
aktualizuj	False	Všem objektům v seznamu posluchačů zašle zprávu aktualizuj. Parametry:

Třída "VytiskDTO"

Jeden konkrétní výtisk knihy, který si mohou čtenáři vypůjčit. Obsahuje obchodní logiku související se zpracováním jednotlivých výtisků podle jeho aktuálního stavu.

Atributy

Atributy	Datový typ	Poznámky
vytisk	Vytisk	Objekt, který umožňuje všechny informace o výtisku uložit do databáze.
stav	Stav	Stav, ve kterém se výtisk nachází.

Metody	Návratový typ	Poznámky
VytiskDTO	False	Vytvoří novou instanci třídy, na základě uloženého záznamu o výtisku. Parametry: <ul style="list-style-type: none"> vytisk: Vytisk Objekt reprezentující výtisk v databázi.

Metody	Návratový typ	Poznámky
zpracuj	False	Zapíše výpůjčku nebo vrácení výtisku podle aktuálního stavu, ve kterém se výtisk nachází. Parametry: <ul style="list-style-type: none"> ctenar: Ctenar Čtenář, kterému má být výpůjčka zaznamenána.
setStav	False	Nastaví nový stav výtisku. Parametry: <ul style="list-style-type: none"> stav: Stav Stav, který má být nastaven.
getEvidencniCislo	False	Vrátí evidenční číslo výtisku. @return Evidenční číslo výtisku. Parametry:
getVytisk	False	Vrátí výtisk pro potřeby jeho persistentního uložení. @return Výtisk pro persistentní uložení. Parametry:

Třída "ZpracovaniVypujckyController"

Třída implementující rozhraní pro uživatelské rozhraní umožňující předávat doménové vrstvě požadavky z prezentační vrstvy. Jedná se o tzv. use case controller.

Atributy

Atributy	Datový typ	Poznámky
spravceVytisku	SpravceVytisku	
spravceCtenaru	SpravceCtenaru	
aktualniCtenar	Ctenar	
viewVypujckyController	ViewVypujckyController	

Metody	Návratový typ	Poznámky
getSpravceVytisku	False	Parametry:
setSpravceVytisku	False	Parametry:
getSpravceCtenaru	False	Parametry:

Metody	Návratový typ	Poznámky
setSpravceCtenaru	False	Parametry:
getViewVypujckyController	False	Parametry:
setViewVypujckyController	False	Parametry:
ZpracovaniVypujckyController	False	Parametry:
zapisKnihu	False	Realizuje případ užití "Výpůjčit/vrátit knihu". Na základě evidenčního čísla dohledá výtisk dané knihy podle stavu výtisku zapíše novou výpůjčku, či k existující výpůjčce zpíše datum vrácení. Parametry: <ul style="list-style-type: none"> evidencniCislo: String Číslo výtisku knihy, odpovídající čárovému kódu na výtisku.
zadejCtenare	False	Zaznamená čtenáře, ke kterému mají být následně všechny výpůjčky zaznamenávány. Typicky probíhá po načtení čtenářské průkazky. Parametry: <ul style="list-style-type: none"> cisloPrukazky: String Číslo čtenářské průkazky na kterou mají být následně všechny výpůjčky zaznamenávány.
odhlasCtenare	False	Odhlásí aktuálního čtenáře. Další výpůjčky již nebudou zapisovány na jeho jméno. Parametry:

Třída "IPohled"

Interfejs definující rozhraní, které musí třída realizovat, aby mohla být doménovou vrstvou informována o změnách v modelu (doménové vrstvě).

Metody	Návratový typ	Poznámky
aktualizuj	False	Metoda, která je volána u všech registrovaných pohledů, když dojde k nějaké změně v modelu (doménové vrstvě). Parametry:

Metody	Návratový typ	Poznámky

Třída "IViewVypujckyController"

Interfejs definující rozhraní mezi prezentační a doménovou vrstvou. Nabízí metody pro registrování posluchačů, kteří chtějí být informováni o změnách v doménové vrstvě.

Metody	Návratový typ	Poznámky
pridatPohled	False	Parametry:
odebraPohled	False	Parametry:
getVolneVytisky	False	Parametry:
getVypujceneVytisky	False	Parametry:

Třída "IZpracovaniVypujckyController"

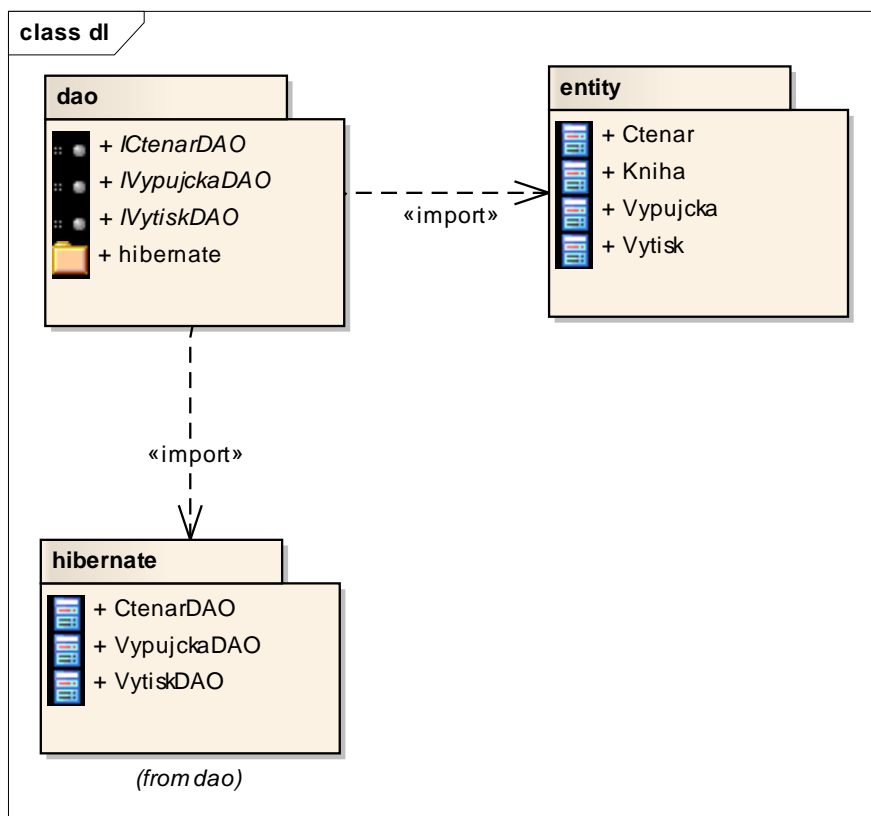
Interfejs definující rozhraní mezi prezentační a doménovou vrstvou umožňující předávat požadavky z prezentační do doménové vrstvy.

Metody	Návratový typ	Poznámky
zapisKnihu	False	Umožňuje zpracovat knihu se zadaným evidenčním číslem. Zpracováním knihy se rozumí zápis jejích vrácení nebo výpůjčení podle jejího aktuálního stavu. Je-li výtisk právě vypůjčen, pak provede zápis o vrácení výtisku a naopak. Parametry: <ul style="list-style-type: none"> evidencniCislo: String Evidenční číslo výtisku, který má být zpracován.
zadejCtenare	False	Zaznamená čtenáře, ke kterému mají být následně všechny výpůjčky zaznamenávány. Typicky probíhá po načtení čtenářské průkazky. Parametry: <ul style="list-style-type: none"> cisloPrukazky: String Číslo čtenářské průkazky na kterou mají být následně všechny výpůjčky zaznamenávány.

Metody	Návratový typ	Poznámky
odhlasCtenare	False	Odhlásí aktuálně zadaného čtenáře. Využívá se pro zpracování všech výtisků, které čtenář předal knihovníkovi ke zpracování. Parametry:

Balíček "dl"

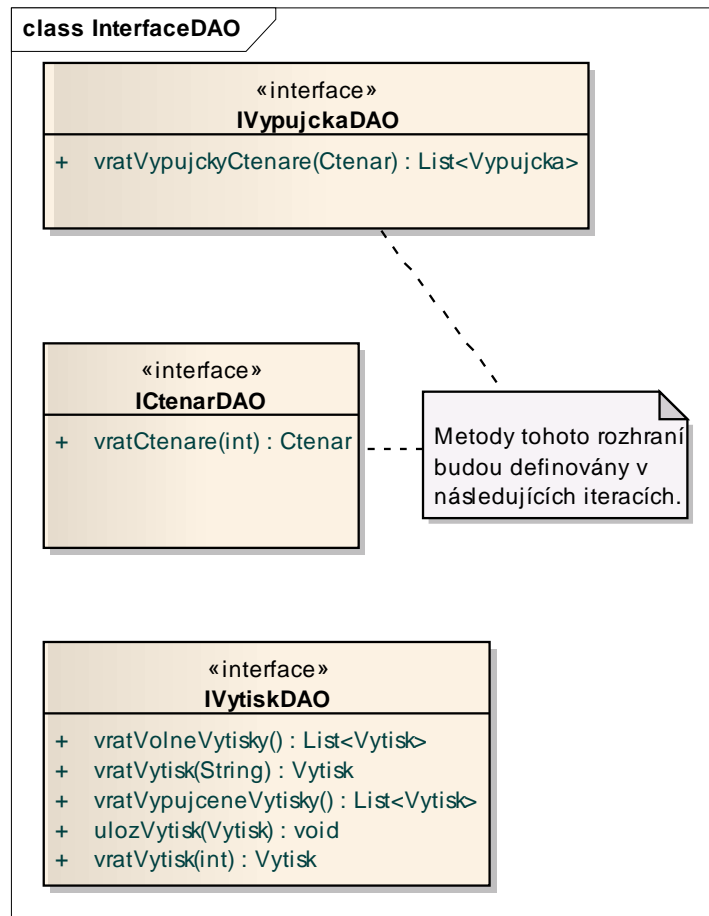
Balíček obsahuje třídy datové vrstvy, které nabízejí technickou podporu pro třídy obchodní vrstvy. Jedná se především o persistentní ukládání informací.



Obrázek: 10

Balíček "dao"

Balíček obsahuje třídy sloužící pro ukládání a načítání persistentních objektů z/do persistentního úložiště. Obsahuje definici rozhraní persistentní vrstvy a jejich implementaci.



Obrázek: 11

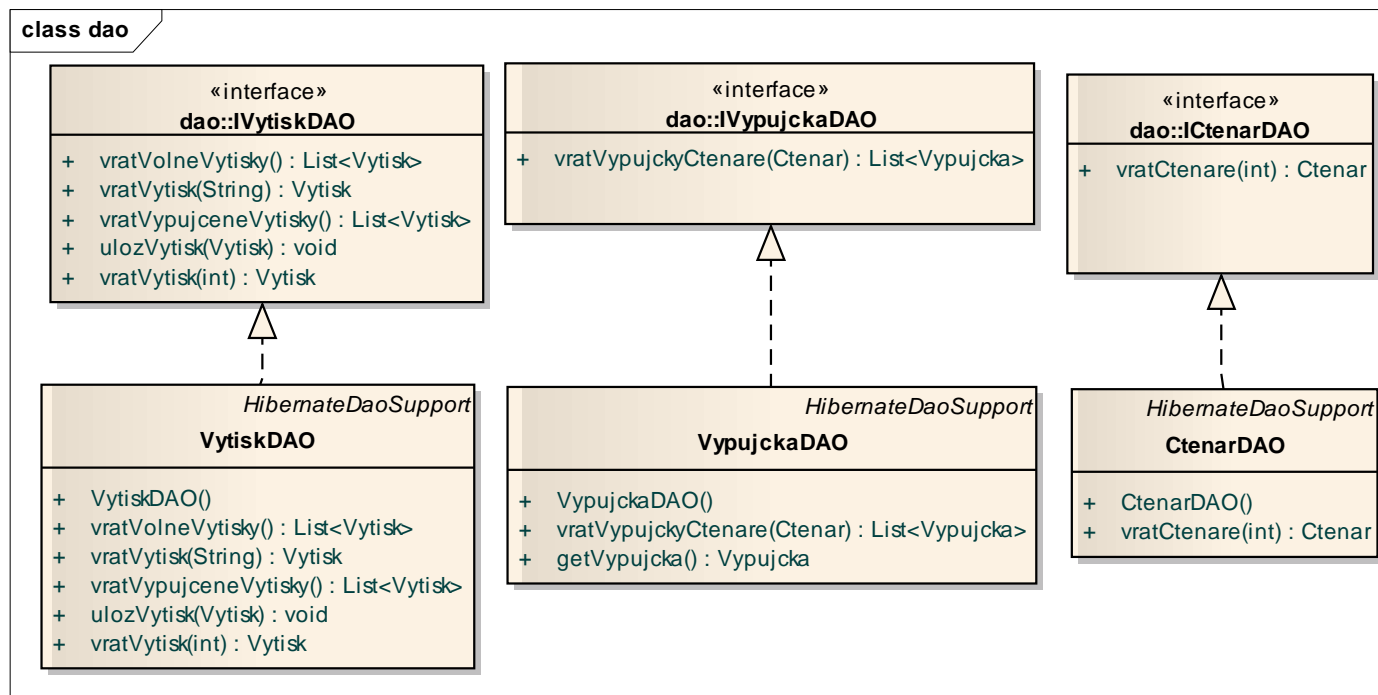
Třída "IVypujckaDAO"

Interface definující rozhraní datové vrstvy pro správu výpůjček.

Metody	Návratový typ	Poznámky
vratVypujckyCtenare	False	Vrátí seznam všech výpůjčených výtisků daného čtenáře. Parametry: <ul style="list-style-type: none"> ctenar: Ctenar Čtenář jehož výpůjčky má metoda vrátit.

Balíček "hibernate"

Balíček obsahuje implementaci rozhraní datové vrstvy s využitím frameworku Hibernate pro persistentní ukládání dat.



Obrázek: 12

Třída "CtenarDAO"

Třída umožňuje ukládání a načítání informací o jednotlivých čtenářích.

Metody	Návratový typ	Poznámky
CtenarDAO	False	Parametry:
vratCtenare	False	Vyhledá čtenáře, na základě jeho čísla průkazky. Parametry: <ul style="list-style-type: none"> cisloPrukazky: int Číslo průkazky čtenáře.

Třída "VypujckaDAO"

Třída umožňuje ukládání a načítání záznamů o provedených výpůjčkách jednotlivých výtisků.

Metody	Návratový typ	Poznámky
VypujckaDAO	False	Parametry:
vratVypujckyCtenare	False	

Metody	Návratový typ	Poznámky
		Parametry:
getVypujcka	False	Parametry:

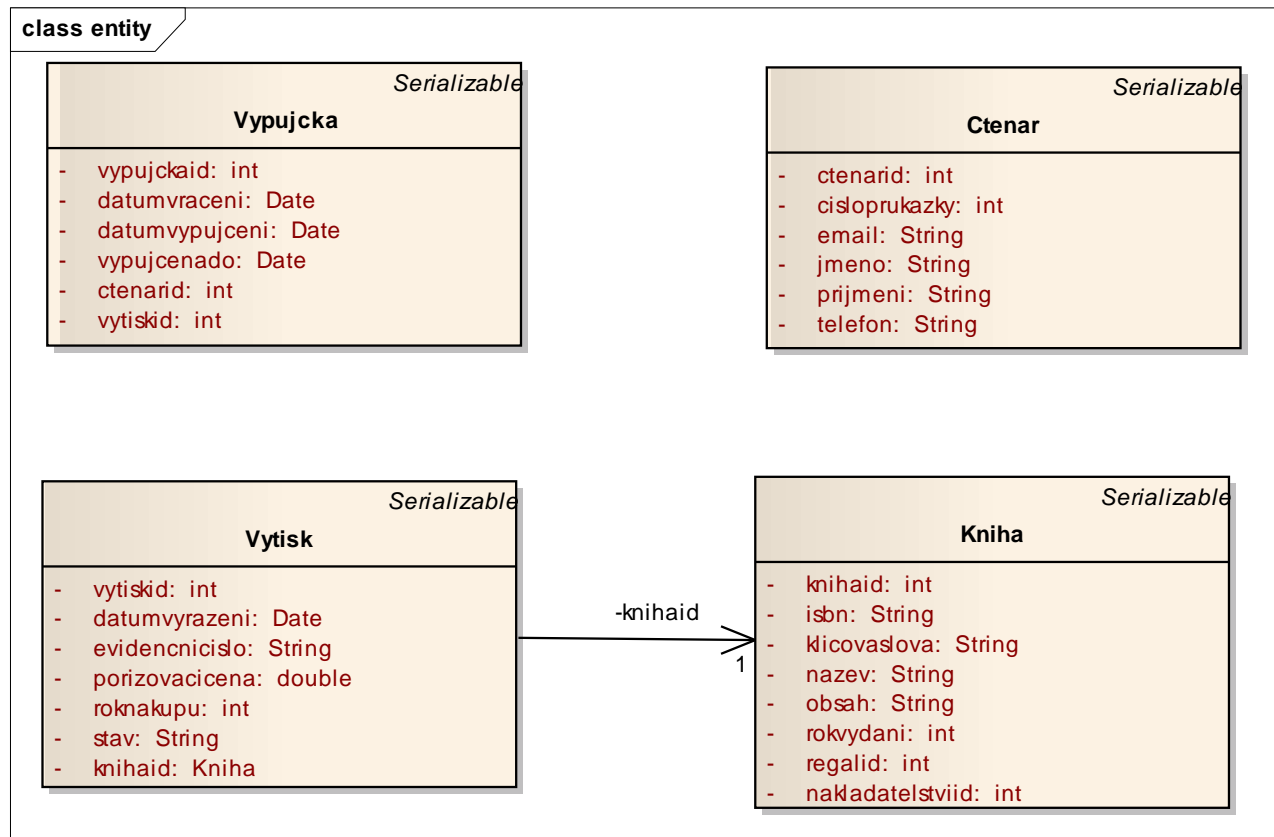
Třída "VytiskDAO"

Třída umožňuje ukládání a načítání výtisků a jejich stavů.

Metody	Návratový typ	Poznámky
VytiskDAO	False	Parametry:
vratVolneVytisky	False	Vrátí seznam výtisků, které jsou v knihovně právě k dispozici. Parametry:
vratVytisk	False	Vrátí výtisk knihy na základě jeho evidenčního čísla. Parametry: <ul style="list-style-type: none"> evidencniCislo: String evidencniCislo
vratVypujceneVytisky	False	Parametry:
ulozVytisk	False	Parametry:
vratVytisk	False	Vrátí výtisk knihy na základě jeho evidenčního čísla. Parametry: <ul style="list-style-type: none"> evidencniCislo: int evidencniCislo

Balíček "entity"

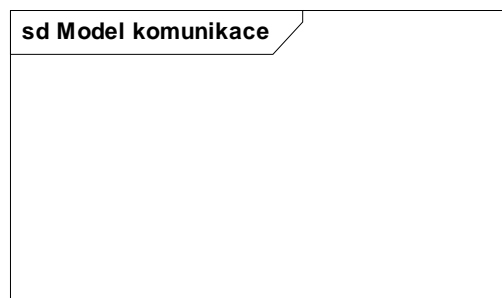
Balíček obsahuje třídy, představují datové objekty, které je nutné persistentně ukládat.



Obrázek: 13

Model komunikace

Kapitola popisuje přiřazení zodpovědnosti tříd při realizaci požadovaných funkcí. Jednotlivé třídy byly popsány v předchozí kapitole. V této kapitole jsou obsaženy pouze popisy spolupráce tříd, které jsou z pohledu objektově orientovaného návrhu zajímavé.

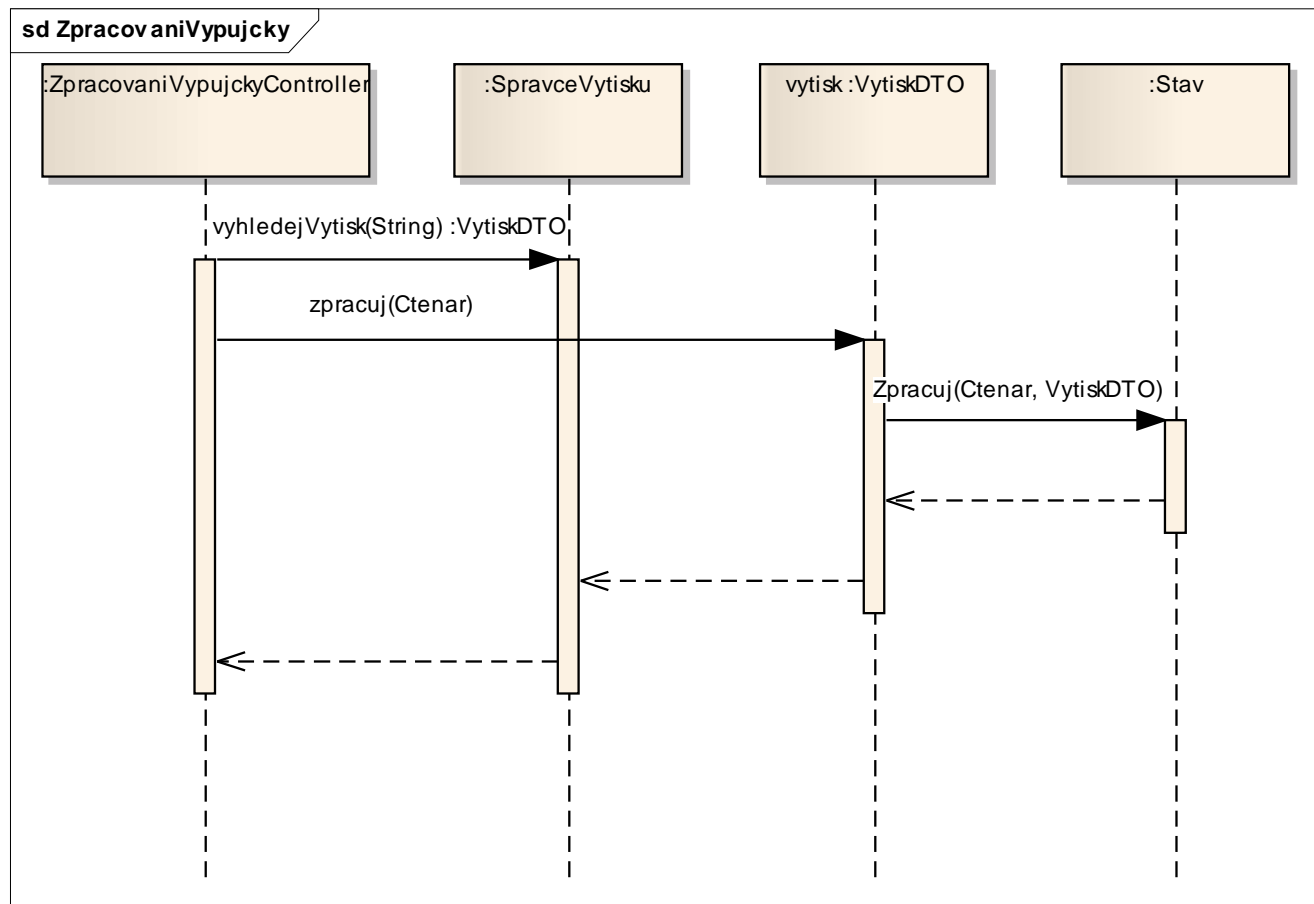


Obrázek: 14

Výpůjčky

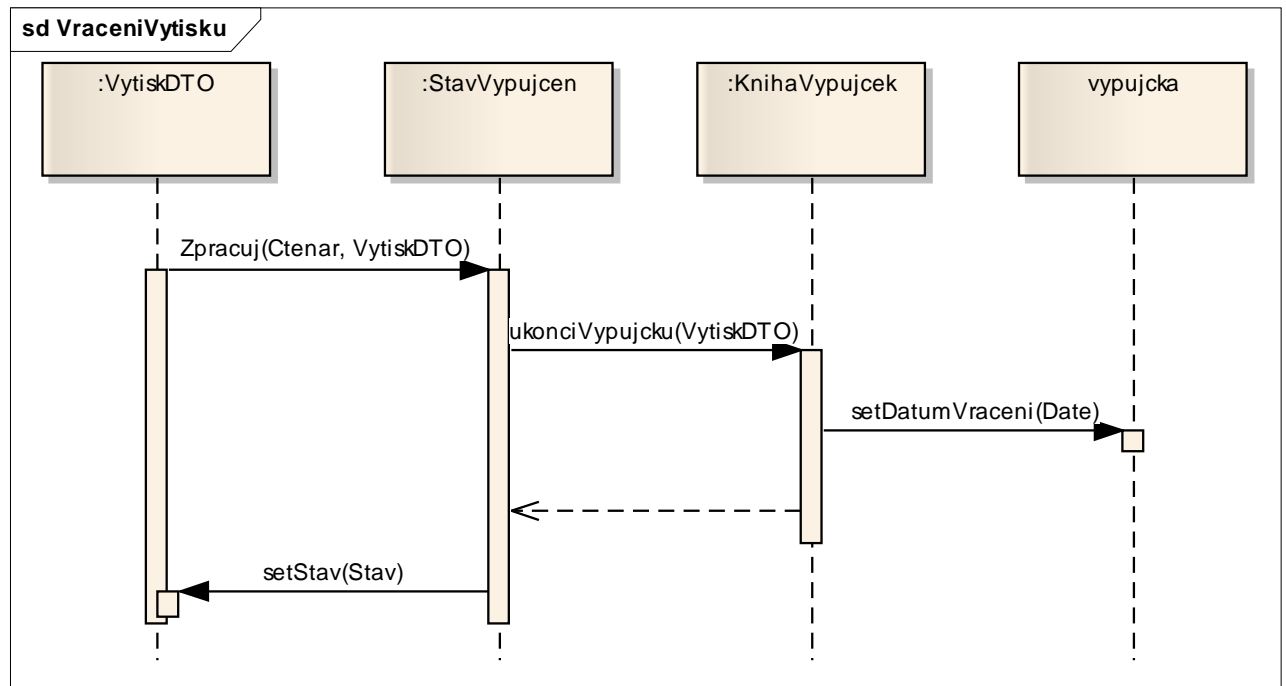
Kapitola obsahuje popis spolupráce tříd pro realizaci funkcí souvisejících s výpůjčováním výtisků čtenáři.

Na obrázcích je zachycena spolupráce tříd při zpracování výtisku. Zpracováním výtisku se rozumí vytvoření zápisu o jeho výpůjčení nebo vrácení podle jeho aktuálního stavu. Popis spolupráce začíná na úrovni doménové vrstvy. prezentační vrstva zde není řešena. V tomto případě se bude pravděpodobně jednat o načtení čárového kódu umístěného na výtisku. Stejným způsobem by vše probíhalo i pokud by knihovník evidenční číslo výtisku přepsal do nějakého formuláře pomocí klávesnice.



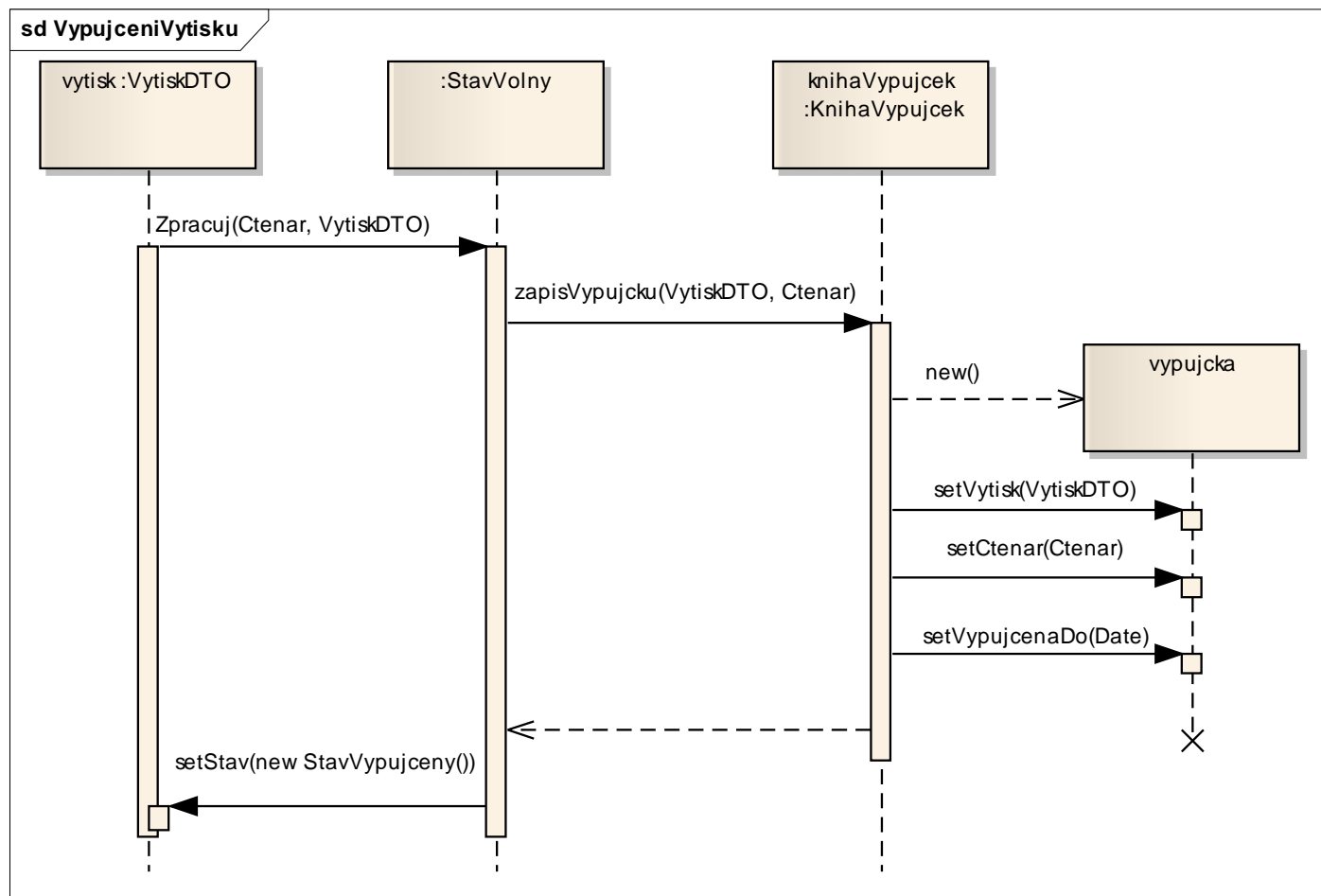
Obrázek: 15

Celá spolupráce začíná zavoláním metody *zpracujKnihu* třídy *ZpracovaniVypujckyController*. Tato třída vyhledá výtisk zadanému evidenčnímu číslu a přenechá zpracování výtisku na třídě *VytiskDTO*, která pro rozhodnutí, zda se jedná o výpůjčený výtisk a má být zapsáno jeho vrácení nebo o volný výtisk a má být zapsáno jeho vrácení, využívá "State pattern", takže toto rozhodnutí přenechá třídě *Stav*, která představuje konkrétní stav v jakém se výtisk nachází.



Obrázek: 16

Obrázek zachycuje pokračování předchozí komunikace, pokud je výtisk ve stavu *Vypujcen*. V tomto případě zavolá třída *StavVypujcen* metodu *ukonciVypujcku* třídy *kniha výpůjček* a tomuto výtisku nastaví nový stav na *StavVolny*. tento princip je blíže popsán v GoF jako State pattern.

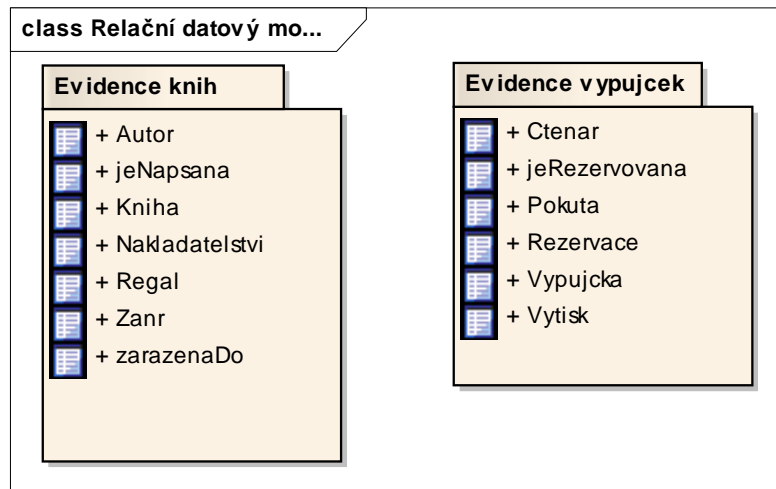


Obrázek: 17

Obrázek zachycuje pokračování komunikace v případě, že daný výtisk je ve stavu *Volny*. V tomto případě třída *StavVolny* zavolá metodu třídy *KnihaVypujcek* *zapisVypujcku*. V této metodě je vytvořen nový záznam o výpůjčce.

Relační datový model

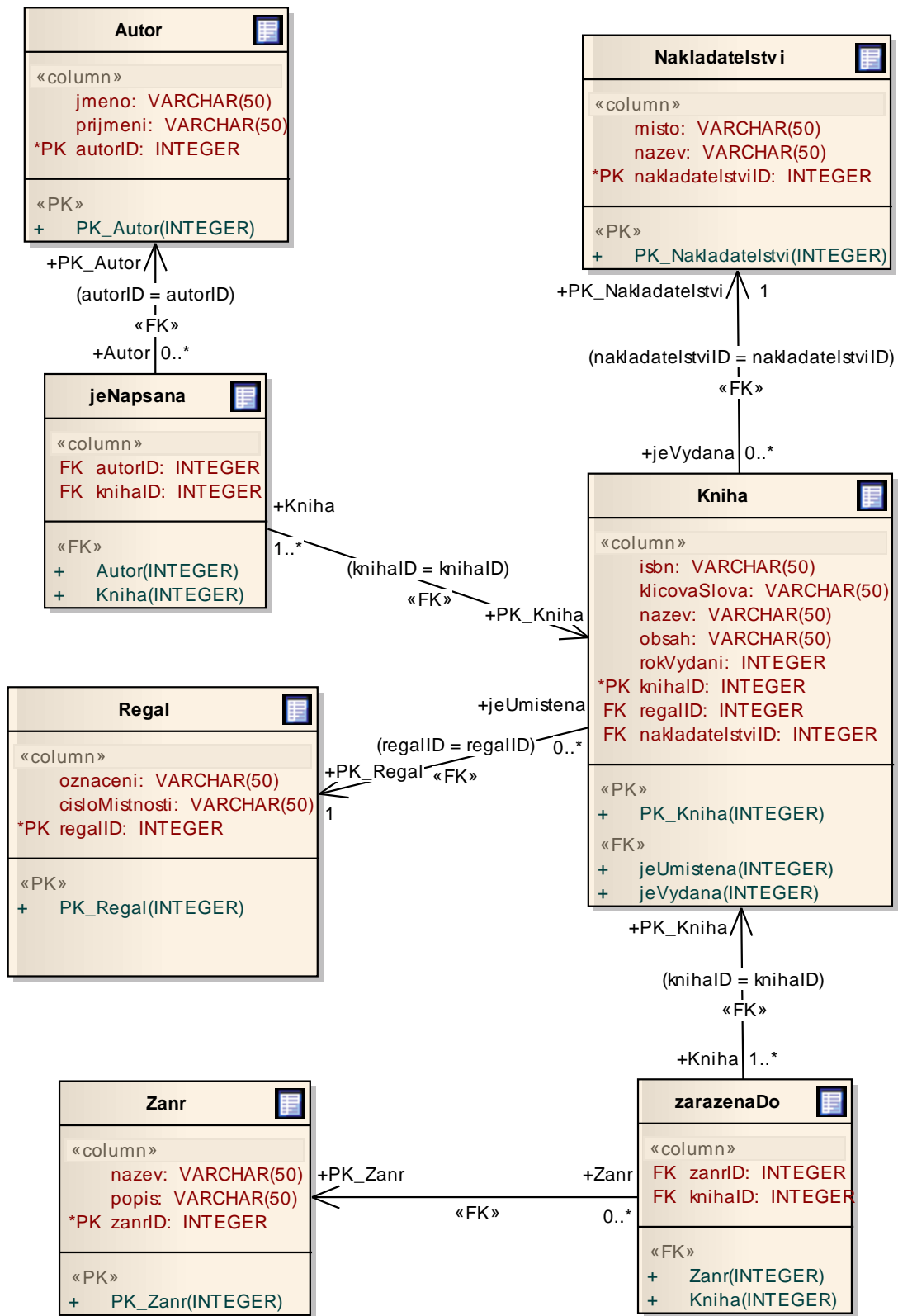
Kapitola obsahuje navržený způsob ukládání dat do relační databáze.



Obrázek: 18

Evidence knih

class Evidence knih



Tabulka "Autor"

Třída představuje osobu, která je autorem nebo spoluautorem knihy.

Atributy

Atributy	Datový typ	Poznámky
jmeno	VARCHAR	Jméno autora knihy.
prijmeni	VARCHAR	Příjmení autora knihy.
autorID	INTEGER	

Tabulka "Kniha"

Třída představující popis skupiny výtisků knih.

Atributy

Atributy	Datový typ	Poznámky
isbn	VARCHAR	Mezinárodní standardizované číslo knihy.
klicovaSlova	VARCHAR	Klíčová slova, se kterými souvisí obsah knihy, které usnadňují rozhodnutí, zda se jedná skutečně o požadovanou knihu nebo pouze o shodu názvů.
nazev	VARCHAR	Název knihy.
obsah	VARCHAR	Stručný obsah, či přehled kapitol, které jsou v knize obsaženy.
rokVydani	INTEGER	Rok, kdy byla daná kniha v nakladatelství vydána.
knihaID	INTEGER	
regalID	INTEGER	
nakladatelstviID	INTEGER	

Tabulka "Nakladatelství"

Nakladatelství, které knihy vydalo.

Atributy

Atributy	Datový typ	Poznámky
misto	VARCHAR	Místo, kde nakladatelství působí.
nazev	VARCHAR	Název nakladatelství.
nakladatelstviID	INTEGER	

Tabulka "Regal"

Regál, do kterého jsou vkládány knihy v knihovně.

Atributy

Atributy	Datový typ	Poznámky
oznaceni	VARCHAR	Jednoznačné označení regálu.
cisloMistnosti	VARCHAR	
regalID	INTEGER	

Tabulka "Zanr"

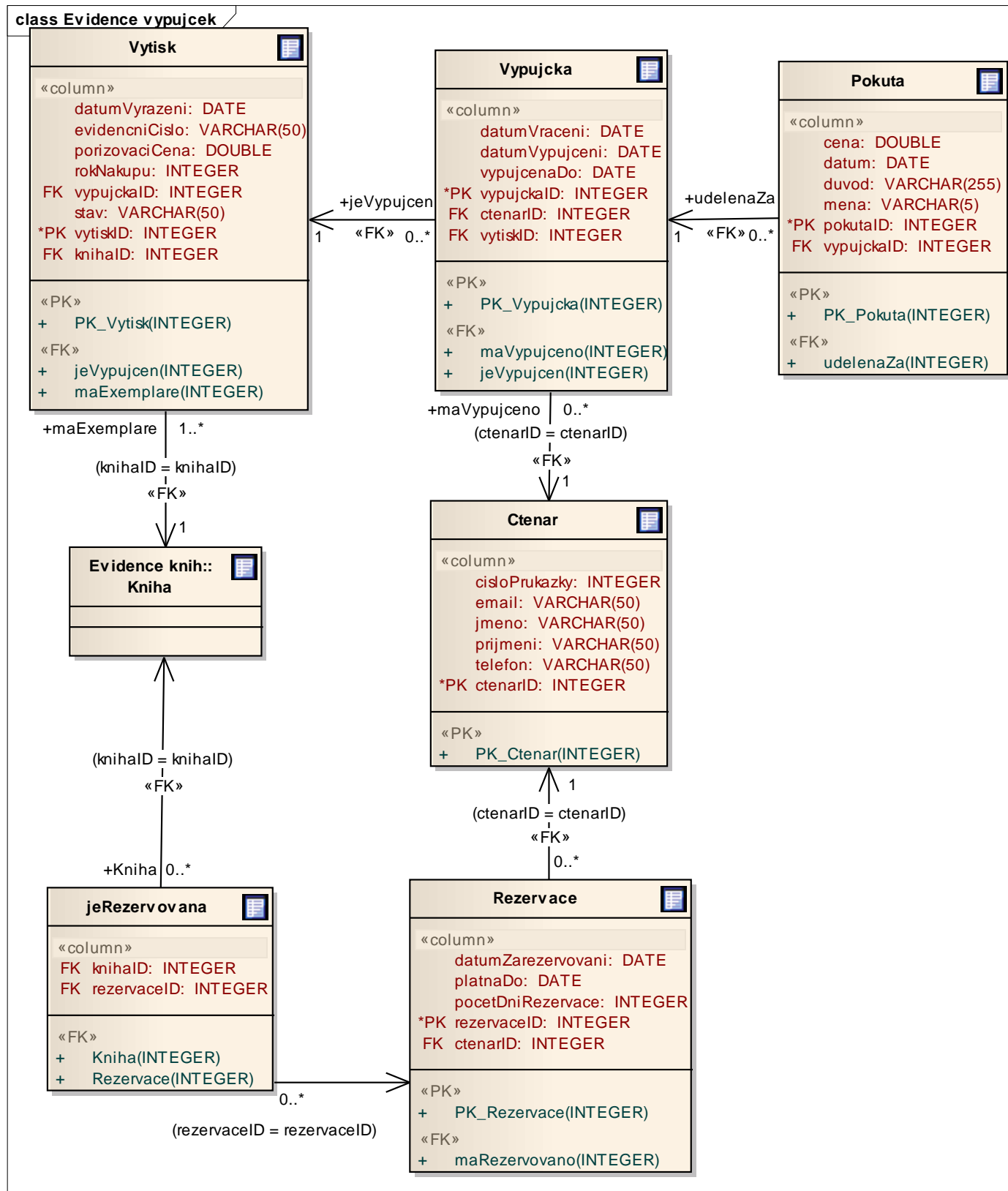
Kategorie, do které kniha spadá (naučná, detektivka, sci-fi, fantasy)

Atributy

Atributy	Datový typ	Poznámky
nazev	VARCHAR	Název žánru knihy.

Atributy	Datový typ	Poznámky
popis	VARCHAR	Stručný popis žánru, vysvětlující jeho význam.
zanrID	INTEGER	

Evidence vypujcek



Obrázek: 20

Tabulka "Ctenar"

Osoba, která navštěvuje knihovnu, aby si zde mohla vyhledat a vypůjčit knihy.

Atributy

Atributy	Datový typ	Poznámky
cisloPrukazky	INTEGER	Jednoznačný identifikátor průkazky, která je čtenářovy vystavována při první návštěvě knihovny.
email	VARCHAR	Kontaktní email čtenáře, na který jsou odesílány různá upozornění z knihovny ohledně opožděného vrácení knih, či informací o dostupnosti rezervované knihy.
jmeno	VARCHAR	Jméno čtenáře.
prijmeni	VARCHAR	Příjmení čtenáře.
telefon	VARCHAR	Telefonní kontakt na čtenáře umožňující rychlé vyřízení některých požadavků, či upozornění v případě, kdy čtenář nereaguje na emailové výzvy.
ctenarID	INTEGER	

Tabulka "Pokuta"

Peníze, které musí čtenář uhradit v případě opožděného vrácení, či ztracení knihy.

Atributy

Atributy	Datový typ	Poznámky
cena	DOUBLE	Výše pokuty.
datum	DATE	Datum, kdy byla pokuta uhrazena.
duvod	VARCHAR	Zdůvodnění proč byla pokuta udělena (ztráta knihy, poškození, opožděné vrácení).
mena	VARCHAR	Měna, ve které byla pokuta uhrazena.
pokutaID	INTEGER	
vypujckaID	INTEGER	

Tabulka "Rezervace"

Požadavek na vypůjčení knihy, kterou má někdo půjčenou.

Atributy

Atributy	Datový typ	Poznámky
datumZarezervovani	DATE	Datum, kdy byla žádost o vypůjčení knihy podána.
platnaDo	DATE	Datum, do kdy je pro čtenáře rezervace vyžadována, po tomto datumu ji čtenář nemá o tuto výpůjčku zájem.
pocetDniRezervace	INTEGER	Počet dní, jak dlouho bude kniha pro čtenáře rezervována od odeslání oznámení.
rezervaceID	INTEGER	
ctenarID	INTEGER	

Tabulka "Vypujcka"

Záznam o vypůjčení knihy čtenářem.

Atributy

Atributy	Datový typ	Poznámky
datumVraceni	DATE	Datum, kdy čtenář knihu vrátil.
datumVypujceni	DATE	Datum, kdy si čtenář knihu vypůjčil.

Atributy	Datový typ	Poznámky
vypujcenaDo	DATE	Datum, do kdy by měla být kniha vrácena.
vypujckaID	INTEGER	
ctenarID	INTEGER	
vytiskID	INTEGER	

Tabulka "Vytisk"

Jeden konkrétní výtisk knihy, který si mohou čtenáři vypůjčit.

Atributy

Atributy	Datový typ	Poznámky
datumVyrazeni	DATE	Datum, kdy byl výtisk vyřazen z knihovny.
evidencniCislo	VARCHAR	Jednoznačný identifikátor výtisku, je dán číslem čárkového kódu, který je na výtisku nalepen.
porizovaciCena	DOUBLE	Pořizovací cena výtisku knihy, slouží ke stanovení výše pokuty při jeho ztracení.
rokNakupu	INTEGER	Rok, kdy knihovna tento výtisk do knihovny nakoupila.
vypujckaID	INTEGER	
stav	VARCHAR	
vytiskID	INTEGER	
knihaID	INTEGER	

Model nasazení

Kapitola popisuje nasazení aplikace a umístění jednotlivých částí na fyzická zařízení.

Celá aplikace je vytvořena jako desktopová aplikace, takže je možné ji spouštět přímo na osobním počítači. Spolu s touto aplikací, která je distribuována jako jar soubor musí být na tomto počítači nainstalována i databáze MySQL 5.0.

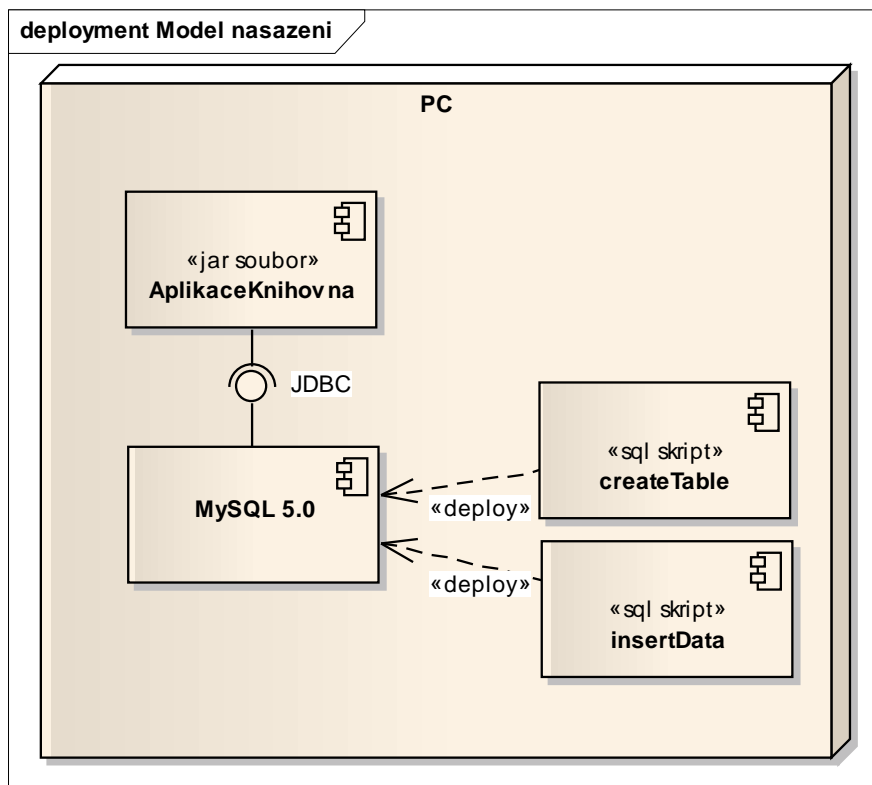
V této databázi musí být vytvořeny tabulky a vloženy počáteční data. Vytvoření těchto tabulek a vložení dat zajistí spuštění souborů createTable.sql a insertData.sql.

Celý postup nasazení aplikace se skládá z těchto kroků:

1. Instalace databázového systému MySQL 5.0 a spuštění databázových skriptů pro vytvoření struktury tabulek a naplnění počátečními testovacími daty.
2. Konfigurace připojení aplikace k nainstalované databázi.
3. Spuštění grafického rozhraní umožňujícího ověření implementované funkčnosti.

Tyto kroky jsou detailněji popsány v následujících kapitolách.

Implementovaný prototyp v souboru aplikaceKnihovna.jar obsahuje pouze funkčnost zpracování výtisku podle jeho aktuálního stavu. V aplikaci je možné zadat evidenční číslo knihy a stiskem tlačítka se daný výtisk přesune do stavu vypůjčený nebo volný, podle toho v jakém stavu se aktuálně nacházel. Připsání výpůjčky konkrétnímu čtenáři bude cílem následující iterace implementace. Zadání evidenčního čísla pomocí klávesnice simuluje jeho zadání použitím čtečky čárových kódů.



Obrázek: 21

Instalace databáze

Nejprve je nutné nainstalovat na Váš počítač databázový systém MySQL 5.0. Instalační balíček pro tento databázový systém je možné stáhnout z webových stránek <http://dev.mysql.com/downloads/>.

Pro studijní účely je doporučeno nainstalovat kromě vlastního databázového systému i další podpůrné nástroje, umožňující jeho snadnější správu. Doporučeno je nainstalovat balíček WampServer, který obsahuje MySQL, Apache a Php. Automaticky se také nainstaluje nástroj pro správu databáze phpMyAdmin, který zjednoduší vytvoření a nasazení celé databáze. Instalační balíček je možné stáhnout na adrese:

<http://www.wampserver.com/dl.php>. Po stažení tohoto instalačního balíčku stačí spustit stažený soubor a provést standardní instalaci. Po instalaci spustě nainstalovaný WampServer.

Po nainstalování databázového systému je nutné vytvořit novou databázi, jednotlivé tabulky a vložit testovací data.

Toto vše lze jednoduše provést přímo v aplikaci phpMyAdmin, kterou spustíte ve Vašem webovém prohlížeči zadáním url <http://localhost/phpmyadmin/>. Na úvodní stránce v této aplikaci můžete rovnou vytvořit novou databázi.

Pro testovací účely doporučuji vytvořit databázi s názvem knihovna, tak aby nebylo nutné měnit v dalším kroku nastavení připojení aplikace k této databázi. V této databázi vytvořte tabulky pomocí záložky import, ve které vyberte pro import soubor createTable.sql, který je dodáváný s aplikací. Stejným postupem importujte také soubor insertData.sql obsahující počáteční testovací data.

Tímto je databáze připraveno k připojení vlastní aplikace.

Konfigurace připojení

V tomto kroku je nutné nakonfigurovat Vaši aplikaci tak, aby se mohla připojit do nově vytvořené databáze. Tato konfigurace se provádí v souboru application-context-dl.xml, který je umístěn v dodávaném souboru knihovna.jar v adresáři knihovna. V tomto souboru je možné nastavit připojení k databázi, jedná se o zadání správného url, na

kterém databázový systém běží, název databáze, uživatelské jméno a heslo. Pokud jste v předchozím kroku provedli standardní instalaci, není nutné pro testovací účely tento konfigurační soubor měnit. Používáte-li Total Commander můžete jar soubor jednoduše rozbalit stisknutím klávesy Ctrl+PageDown a následně konfigurační soubor přímo editovat.

Spuštění aplikace

Posledním krokem při nasazování této aplikace je její spuštění. Toto spuštění lze provést z příkazového řádku zadáním příkazu `java -jar kniovna.jar`.