

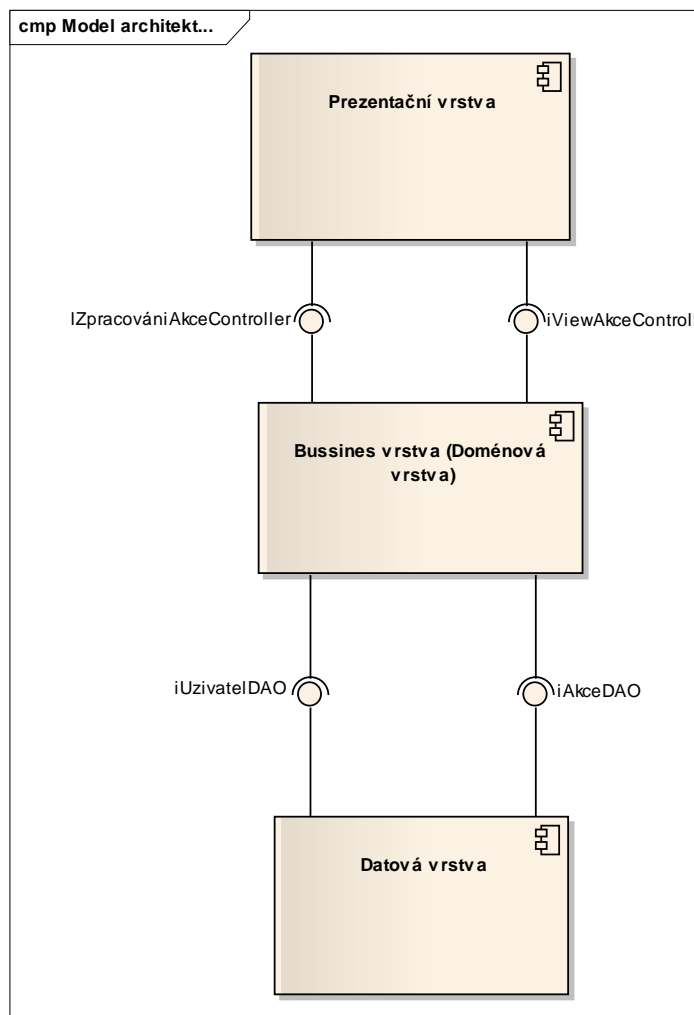
Informační systém pro dětský oddíl
DT
Návrhová dokumentace

Obsah

Model architektury	2
Prezentační vrstva	2
Business vrstva (Doménová vrstva)	2
Datová vrstva	2
Relační datový model	3
Evidence akcí	3
Evidence uživatelů	5
Balíček "Návrhový model"	7
Balíček "Tábor"	7
Model komunikace	17
Model nasazení	19
Instalace MySQL 5.x	20
Konfigurace Hibernate	20
Připojení k aplikaci	20

Model architektury

Balíček obsahuje popis logického členění navrhované aplikace na vrstvy, které umožňují snadnou rozšiřitelnost a udržitelnost aplikace. Jednotlivé vrstvy obsahují další balíčky, do kterých jsou seskupeny třídy, podle jejich logických souvislostí.



Obrázek č. 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...) a předávají je dále.

Business vrstva (Doménová vrstva)

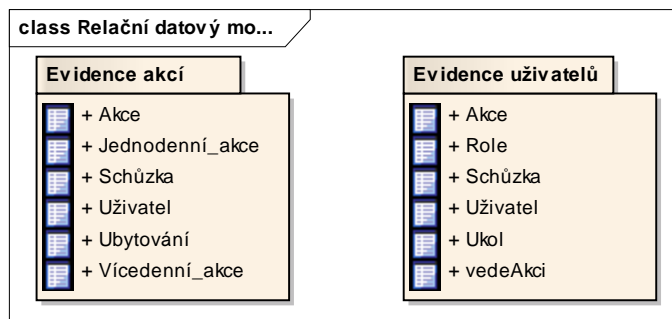
Komponenta obsahuje třídy a komponenty, které realizují požadované chování aplikace z pohledu obchodní logiky.

Datová vrstva

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

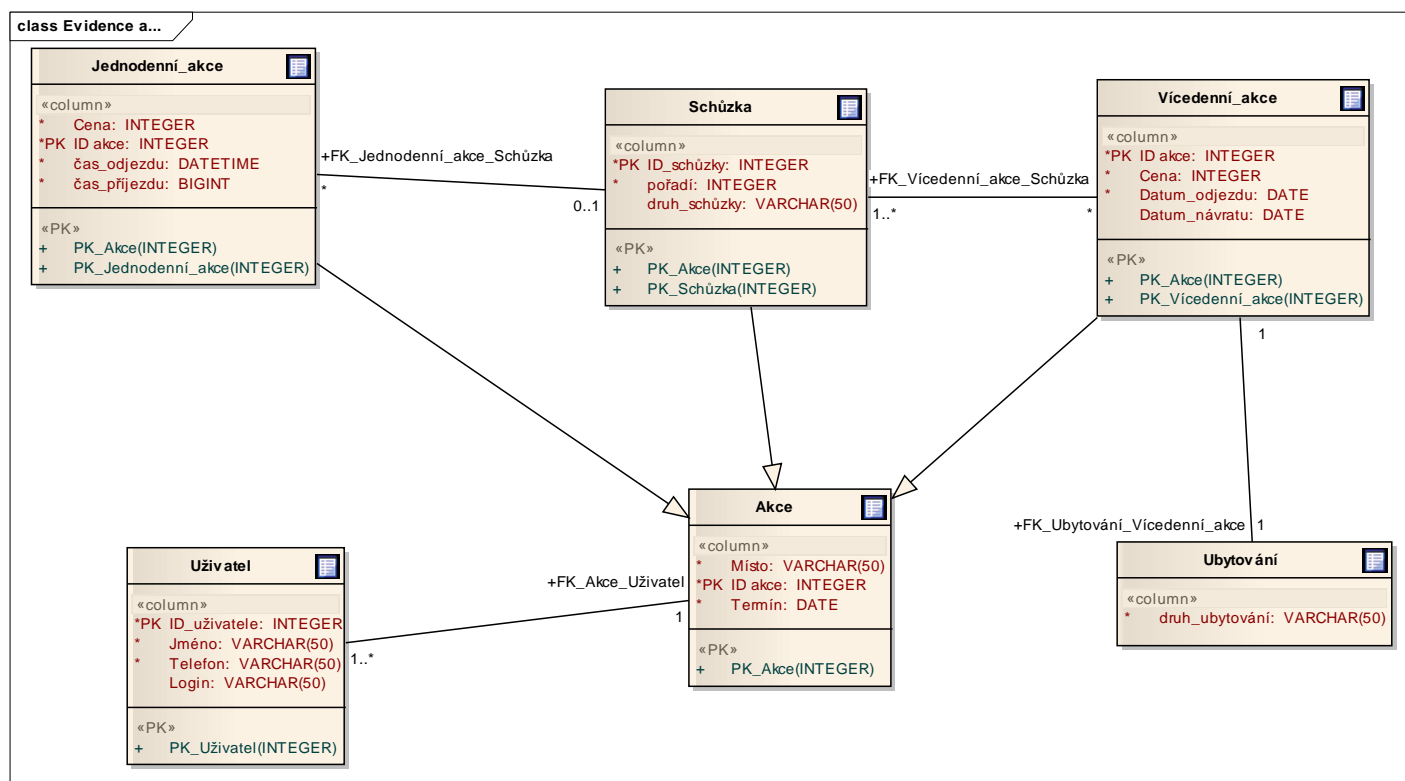
Relační datový model

Obsahuje způsob navržení ukládání dat do relační databáze.



Obrázek č. 2

Evidence akcí



Obrázek č. 3

Tabulka "Akce"

Třída reprezentující akci (výlet, pobyt na horách atd.)

Atributy	Datový typ	Poznámky
Místo	VARCHAR	místo konání akce.
ID akce	INTEGER	unikátní ID akce.
Termín	DATE	termín pořádání akce.

Tabulka "Jednodenní_akce"

Výlety pouze na jeden den.

Atributy	Datový typ	Poznámky
Cena	INTEGER	cena akce
ID akce	INTEGER	unikátní ID akce.
čas_odjezdu	DATETIME	čas kdy akce začíná
čas_příjezdu	BIGINT	kdy akce končí

Tabulka "Schůzka"

Třída reprezentující schůzky vedení oddílu.

Atributy	Datový typ	Poznámky
ID_schůzky	INTEGER	unikátní Id schůzky
pořadí	INTEGER	pořadí schůzky
druh_schůzky	VARCHAR	zda je schůzka organizační nebo programová

Tabulka "Ubytování"

Místo ubytování pro vícedenní akci.

Atributy	Datový typ	Poznámky
druh_ubytování	VARCHAR	Chatka, hotel, stan...

Tabulka "Uživatel"

Třída reprezentující přihlášeného uživatele.

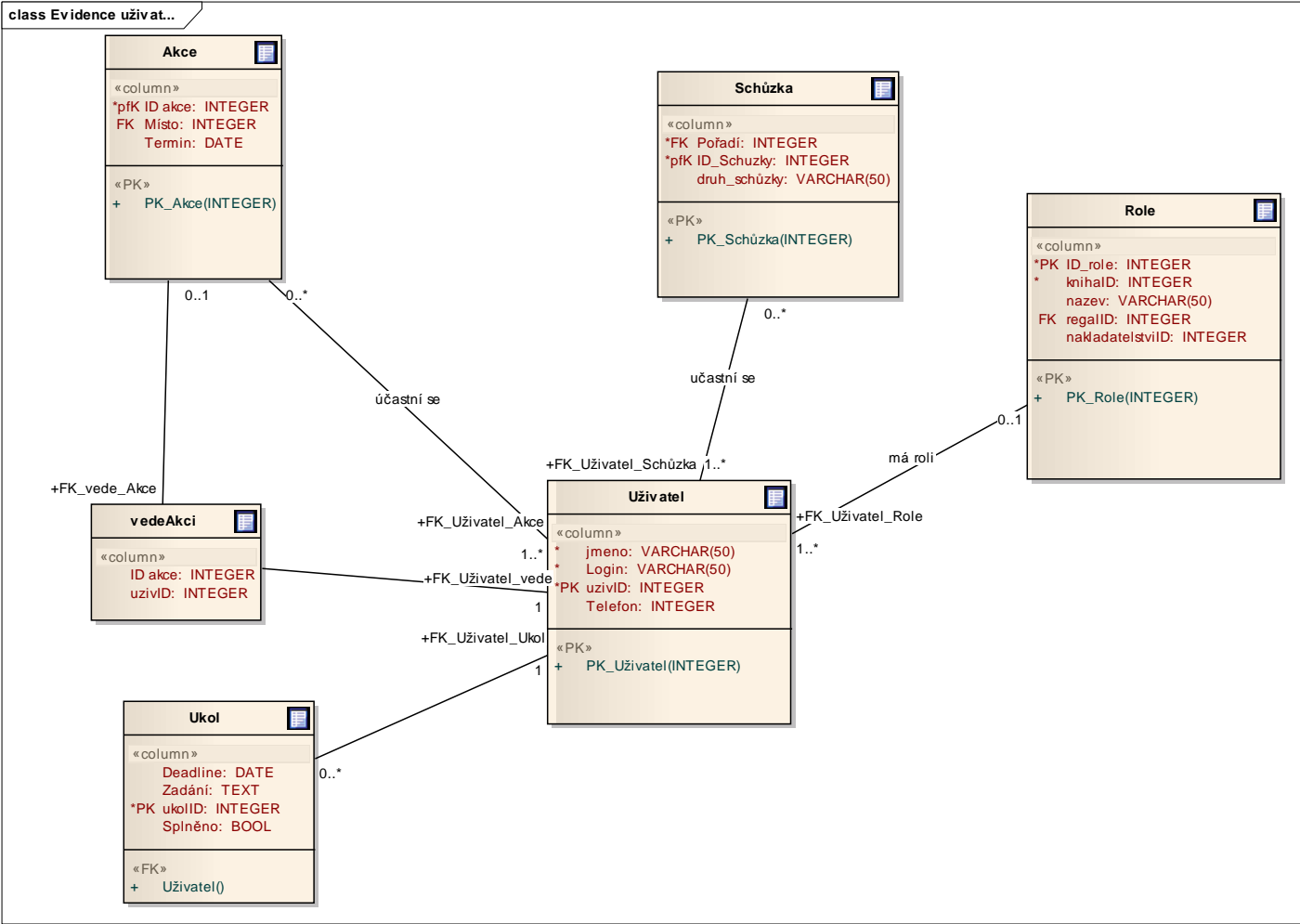
Atributy	Datový typ	Poznámky
ID_uživatele	INTEGER	unikátní ID uživatele
Jméno	VARCHAR	jméno a přímení uživatele
Telefon	VARCHAR	telefon uživatele
Login	VARCHAR	login uživatele do systému

Tabulka "Vícedenní_akce"

Akce trvající více než 1 den.

Atributy	Datový typ	Poznámky
ID akce	INTEGER	unikatní ID akce.
Cena	INTEGER	Cena akce
Datum_odjezdu	DATE	Kdy akce začíná.
Datum_návratu	DATE	Kdy akce končí.

Evidence uživatelů



Obrázek č. 4 - diagram evidence uživatelů

Tabulka "Akce"

Třída reprezentující pořádané akce.

Atributy	Datový typ	Poznámky
ID akce	INTEGER	ID akce
Místo	INTEGER	Místo, kde se akce bude konat.
Termin	DATE	Termin kdy se akce bude konat.

Tabulka "Role"

Uživatel může zastávat nějakou roli.

Atributy	Datový typ	Poznámky
ID_role	INTEGER	ID role
knihaID	INTEGER	
nazev	VARCHAR	Název role, kterou uživatel zastává.
regalID	INTEGER	
nakladatelstviID	INTEGER	

Tabulka "Schůzka"

Třída reprezentující schůzky vedení.

Atributy	Datový typ	Poznámky
Pořadí	INTEGER	
ID_Schuzky	INTEGER	
druh_schůzky	VARCHAR	Zda je schůzka organizační nebo programová

Tabulka "Ukol"

Evidence úkolů, které může uživatel dostat.

Atributy	Datový typ	Poznámky
Deadline	DATE	Deadline úkolu.
Zadání	TEXT	stručné zadání úkolu.
ukolID	INTEGER	
Splněno	BOOL	Zda je úkol splněn.

Tabulka "Uživatel"

Třída představuje osobu, která je přihlášena v systému.

Atributy	Datový typ	Poznámky
jmeno	VARCHAR	Jméno uživatele.
Login	VARCHAR	Login uživatele.
uzivID	INTEGER	ID uživatele.
Telefon	INTEGER	Telefon uživatele.

Tabulka "vedeAkci"

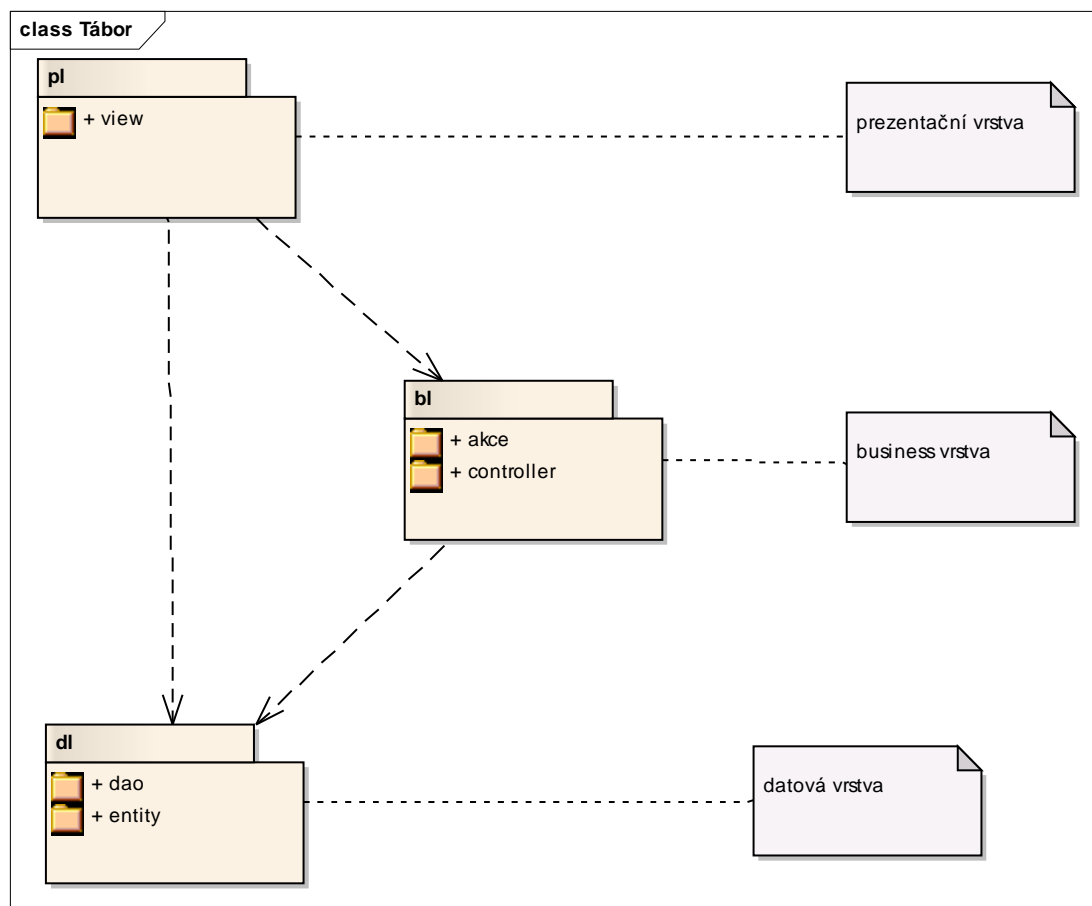
Uživatel může vést akci.

Atributy	Datový typ	Poznámky
ID akce	INTEGER	
uzivID	INTEGER	

Balíček "Návrhový model"

Balíček obsahuje popis návrhových tříd, jejich vzájemných vztahů, přiřazení zodpovědností a popis spolupráce pro realizaci požadovaných funkcí. Pro grafickou reprezentaci statického modelu využívá UML diagram tříd, kde jsou jednotlivé třídy doplněny o metody, atributy o datové typy (dle zvoleného implementačního jazyka nebo s využitím obecných datových typů), konce asociací jsou doplněny o popisy. Pro grafickou reprezentaci dynamického chování využívá UML sekvenční diagramy, UML diagramy spolupráce, UML diagramy aktivit a UML stavové diagramy.

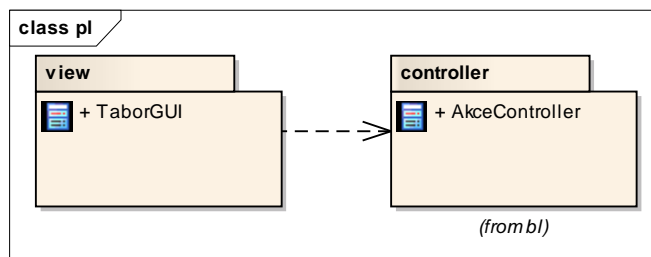
Balíček "Tábor"



Obrázek č. 5

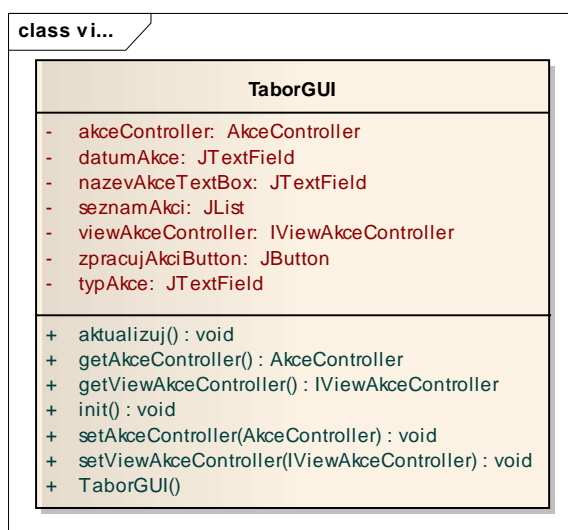
Balíček "pl"

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



Obrázek č. 6

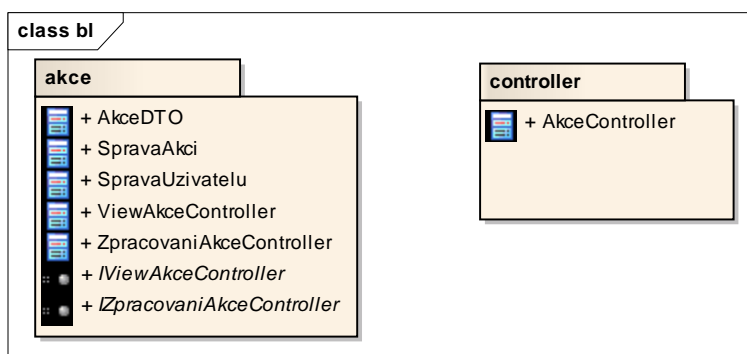
Balíček "view"



Obrázek č. 7

Balíček "bl"

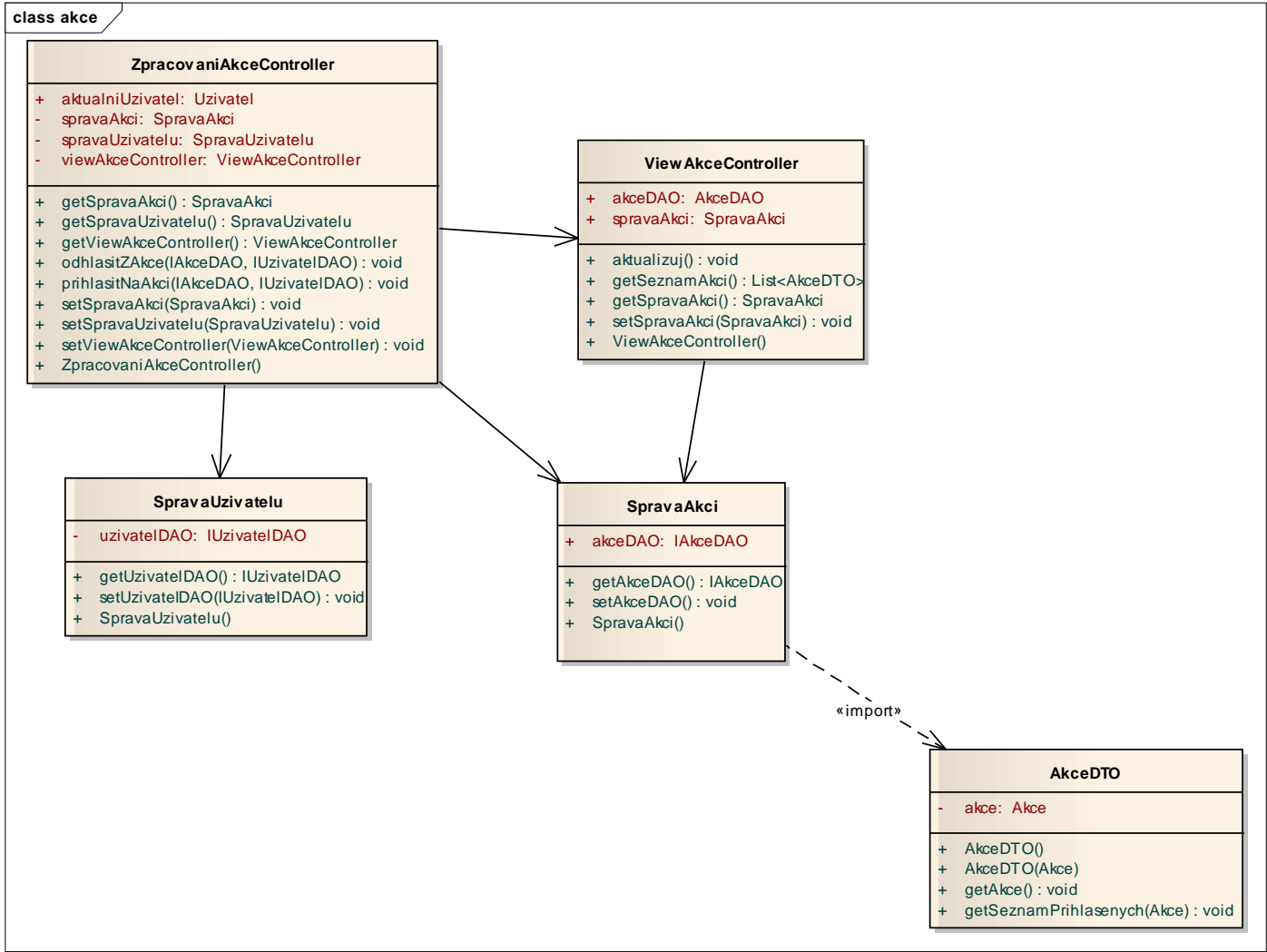
Balíček obsahuje třídy business vrstvy, které implementují vlastní chování celé aplikace. Obsahuje třídy, které realizují veškeré chování aplikace související s obchodní logikou.



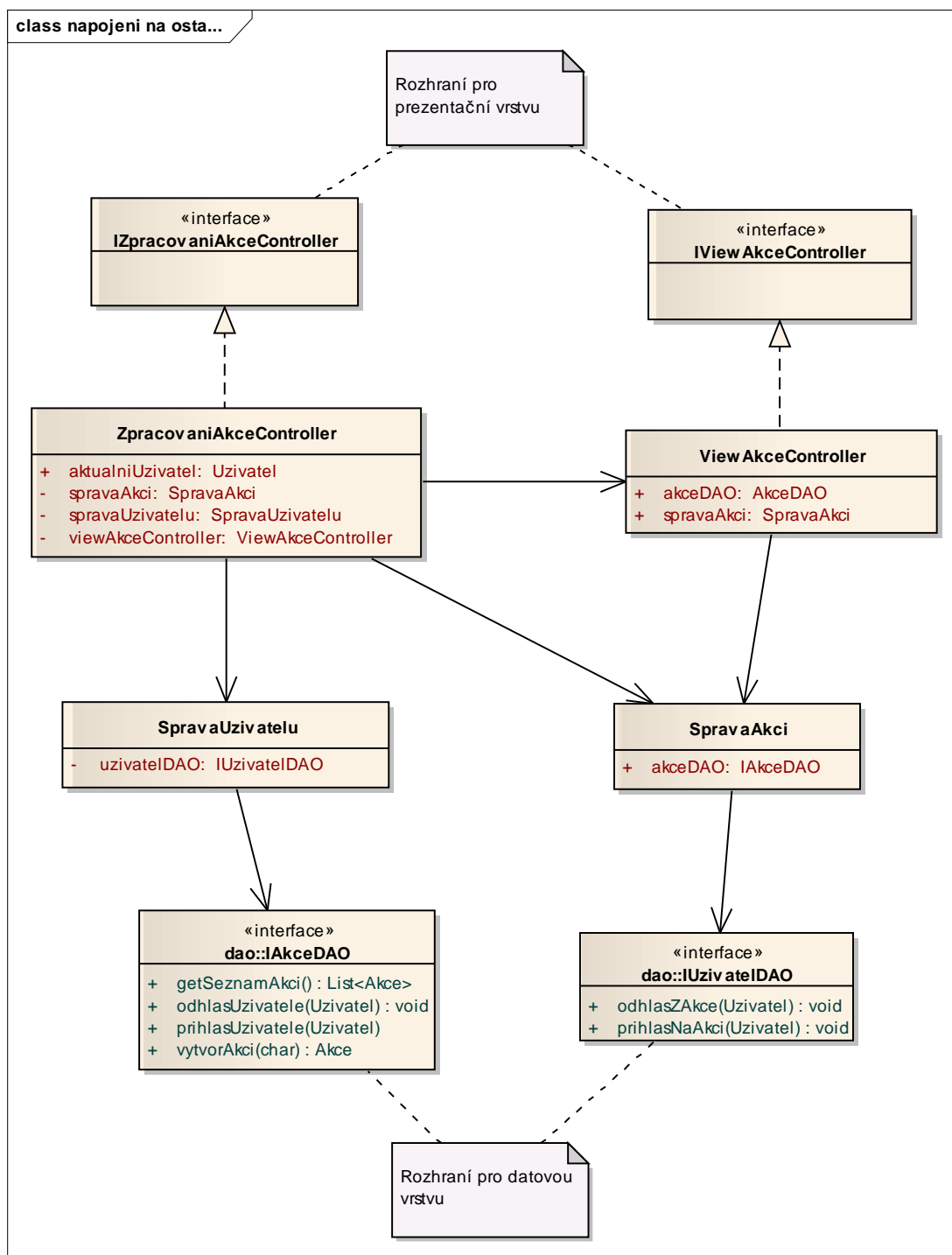
Obrázek č. 8

Balíček "akce"

Balíček obsahující třídy, které umožňují správu akcí.



Obrázek č. 9



Obrázek č. 10

Třída "AkceDTO"

Jeden konkrétní akce, na kterou se mohou uživatelé přihlásit. Obsahuje obchodní logiku související se zpracováním jednotlivých akcí.

Atributy	Datový typ	Poznámky
akce	Akce	

Metody	Návratový typ	Poznámky
AkceDTO	False	Parametry:
AkceDTO	False	Parametry:
getAkce	False	Parametry:
getSeznamPrihlasenych	False	Parametry:

Třída "SpravaAkci"

Třída zapouzdřující logiku související s evidencí akcí, jejich vytváření a editaci.

Atributy	Datový typ	Poznámky
akceDAO	IAkceDAO	

Metody	Návratový typ	Poznámky
getAkceDAO	False	Parametry:
setAkceDAO	False	Parametry:
SpravaAkci	False	Parametry:

Třída "SpravaUzivatelu"

Třída zapouzdřující logiku související s evidencí uživatelů, jejich přidáváním a editací.

Atributy	Datový typ	Poznámky
uzivatelDAO	IUzivatelDAO	

Metody	Návratový typ	Poznámky
getUzivatelDAO	False	Parametry:
setUzivatelDAO	False	Parametry:
SpravaUzivatelu	False	Parametry:

Třída "ViewAkceController"

Třída implementující rozhraní (interface) umožňující připojení uživatelského rozhraní k doménové vrstvě. Třída je součástí vzoru "Observer".

Atributy	Datový typ	Poznámky
akceDAO	AkceDAO	
spravaAkci	SpravaAkci	

Metody	Návratový typ	Poznámky
aktualizuj	False	Parametry:
getSeznamAkci	False	Parametry:
getSpravaAkci	False	Parametry:
setSpravaAkci	False	Parametry:
ViewAkceController	False	Parametry:

Třída "ZpracovaniAkceController"

Třída implementující rozhraní pro uživatelské rozhraní umožňující předávat doménové vrstvě požadavky z prezentační vrstvy.

Atributy	Datový typ	Poznámky
aktualniUzivatel	Uzivatel	
spravaAkci	SpravaAkci	
spravaUzivatelu	SpravaUzivatelu	
viewAkceController	ViewAkceController	

Metody	Návratový typ	Poznámky
getSpravaAkci	False	Parametry:
getSpravaUzivatelu	False	Parametry:
getViewAkceController	False	Parametry:
odhlasitZAkce	False	Parametry:
prihlasitNaAkci	False	Parametry:
setSpravaAkci	False	Parametry:

setSpravaUzivatelu	False	Parametry:
setViewAkceController	False	Parametry:
ZpracovaniAkceController	False	Parametry:

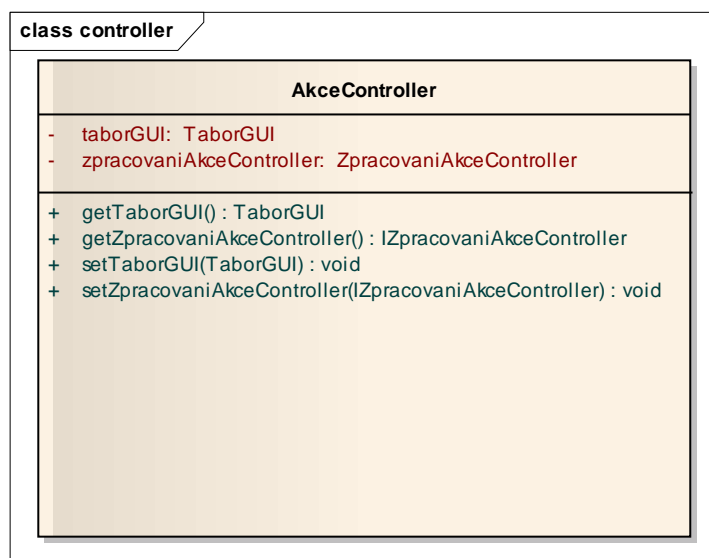
Třída "IViewAkceController"

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

Třída "IZpracovaniAkceController"

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

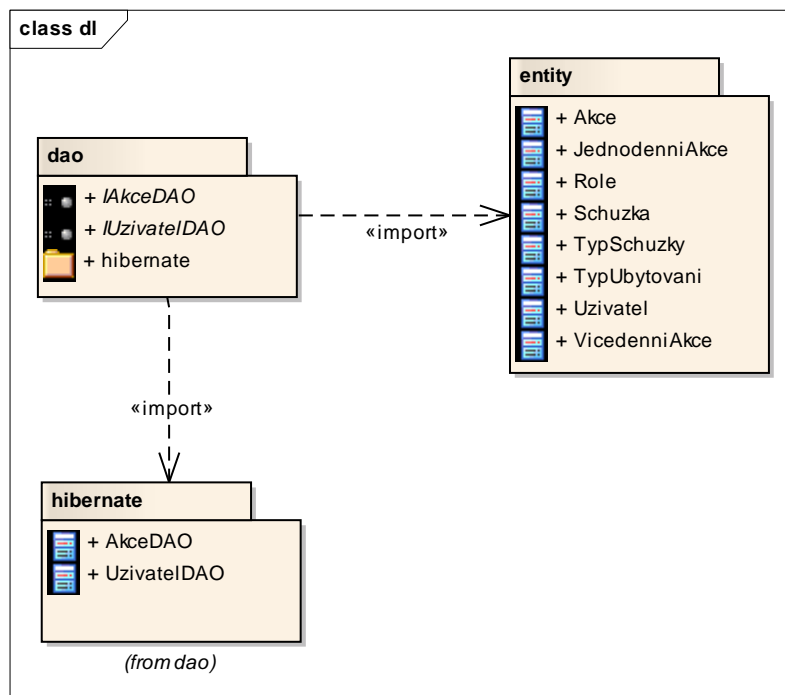
Balíček "controller"



Obrázek č. 11

Balíček "dl"

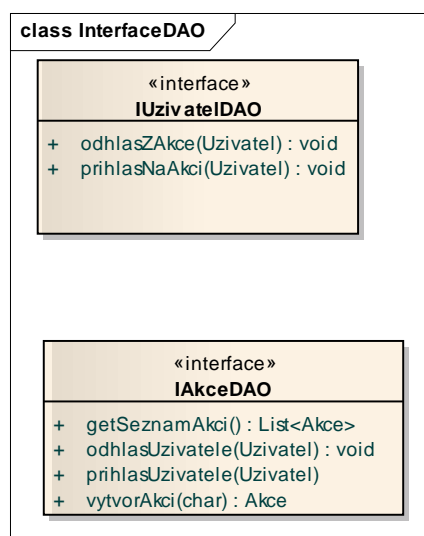
Balíček obsahuje třídy datové vrstvy, které nabízejí technickou podporu pro třídy obchodní vrstvy. Jedná se především o persistentní ukládání informací.



Obrázek č. 12

Balíček "dao"

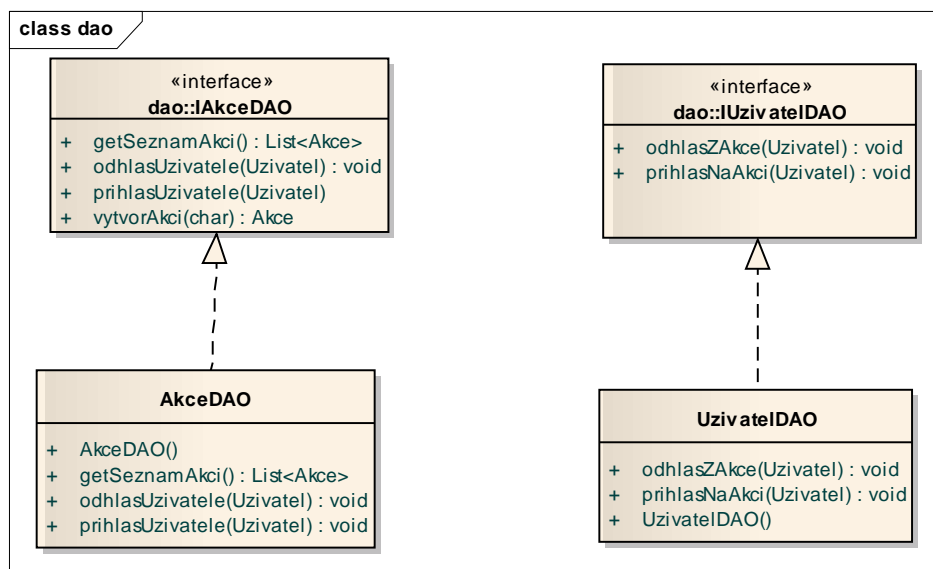
Balíček obsahuje třídy sloužící pro ukládání a načítání persistentních objektů z/do persistentního úložiště. Obsahuje definici rozhraní persistentní vrstvy a jejich implementaci.



Obrázek č. 13

Balíček "hibernate"

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



Obrázek č. 14

Třída "AkceDAO"

Třída umožňuje ukládání a načítání záznamů o akcích.

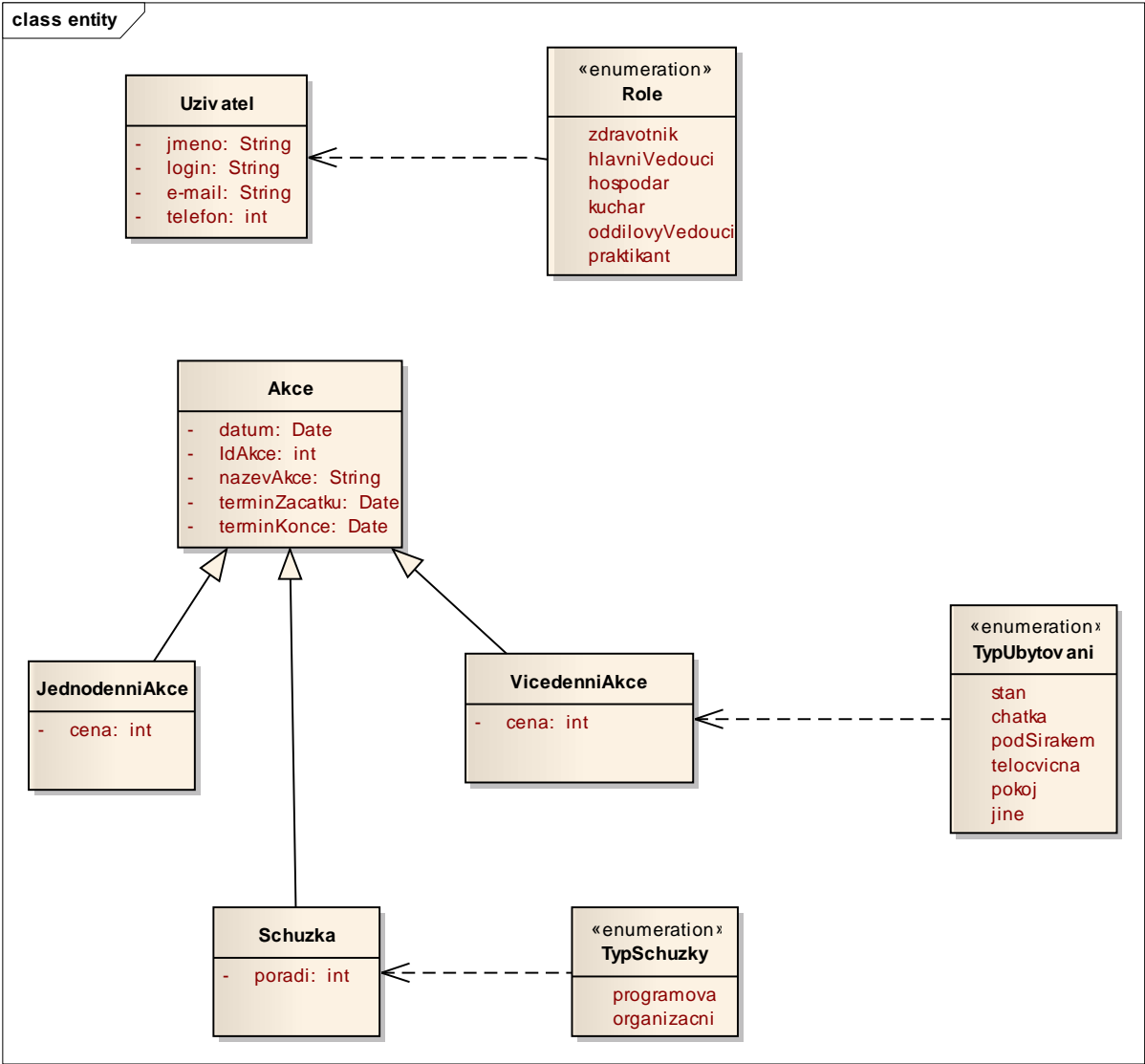
Metody	Návratový typ	Poznámky
AkceDAO	False	Parametry:
getSeznamAkci	False	Parametry:
odhlasUzivatele	False	Parametry:
prihlasUzivatele	False	Parametry:

Třída "UzivatelDAO"

Třída umožňuje ukládání a načítání informací o jednotlivých uživateli.

Metody	Návratový typ	Poznámky
odhlasZAkce	False	Parametry:
prihlasNaAkci	False	Parametry:
UzivatelDAO	False	Parametry:

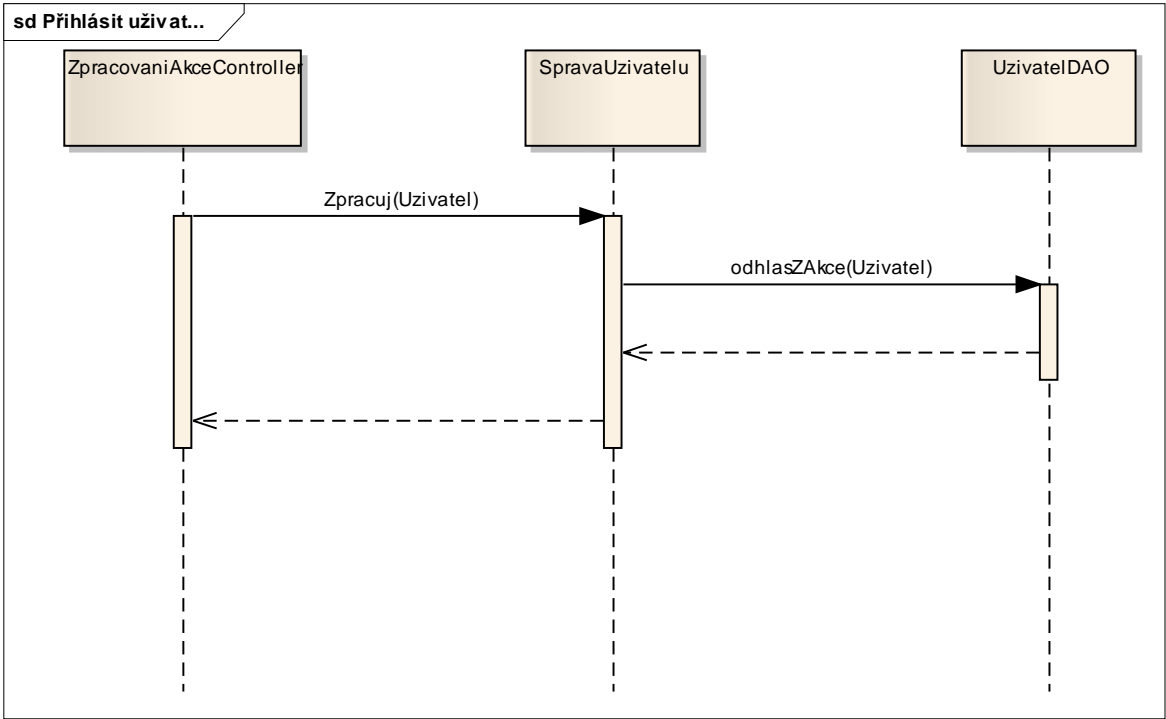
Balíček "entity"



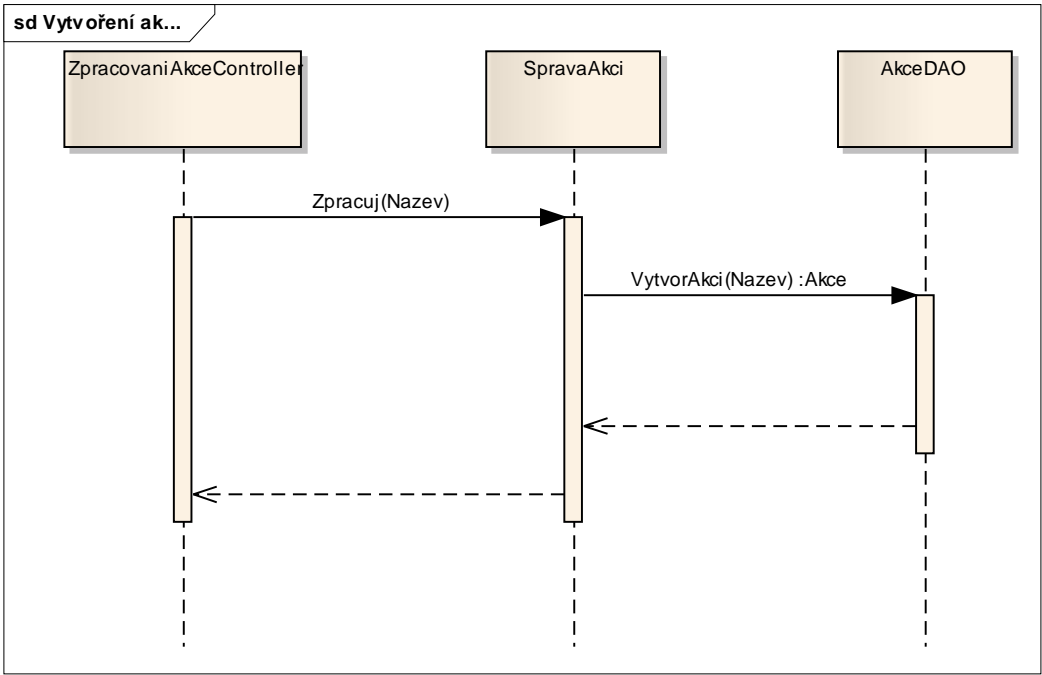
Obrázek č. 15

Model komunikace

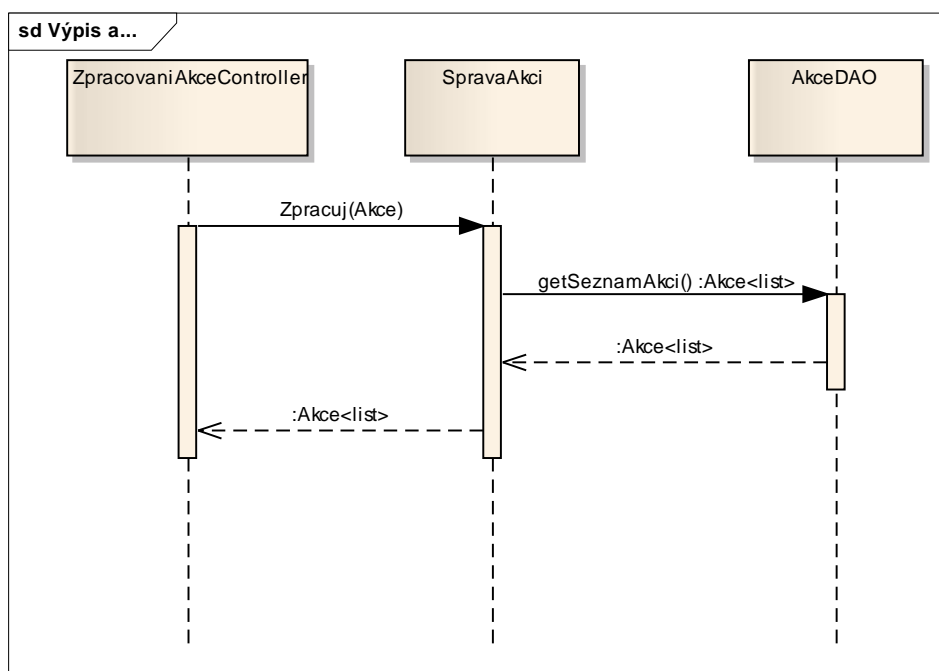
V této kapitole naleznete popis přiřazení zodpovědnosti nejdůležitějších tříd při realizaci žádných funkcností. Popis jednotlivých tříd si můžete dohledat v předchozí kapitole.



Obrázek č. 16



Obrázek č. 17



Obrázek č. 18

Model nasazení

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

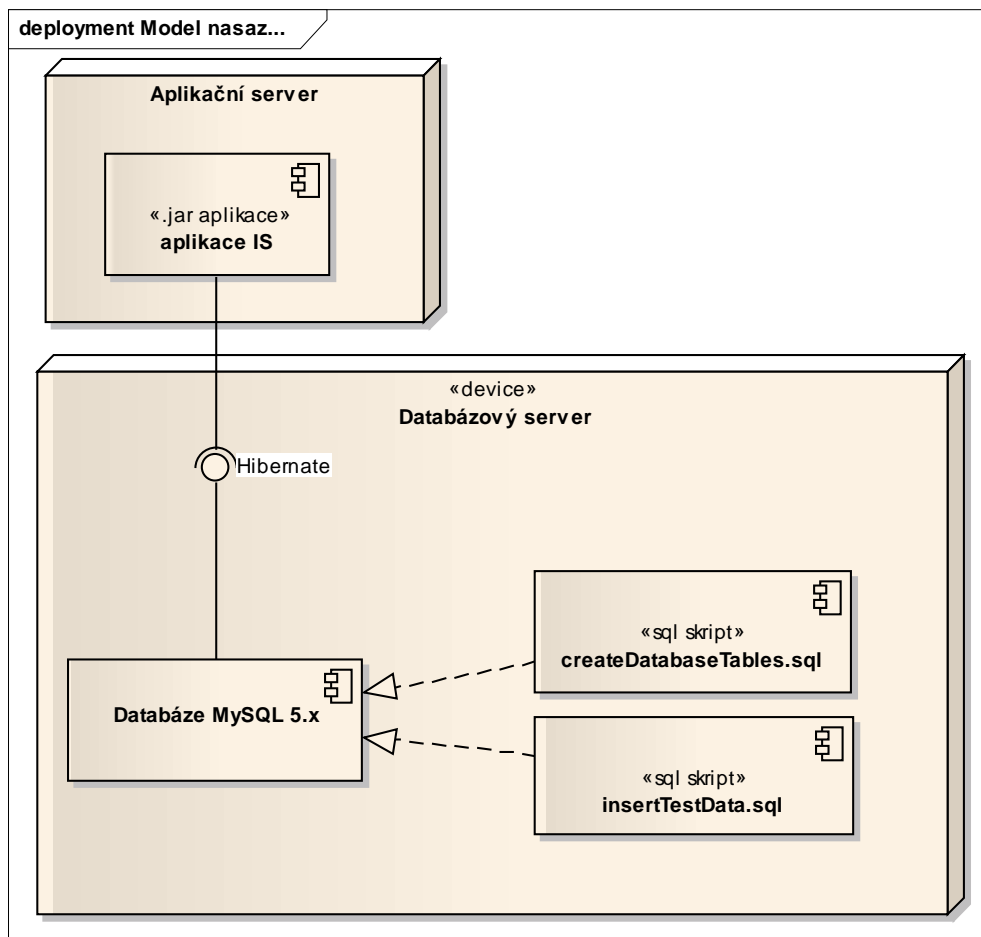
Celá aplikace je vytvořena jako serverová aplikace napsaná v jazyce Java s pomocí frameworku Spring, ke které je možné přistupovat pomocí standartních webových prohlížečů. K přístupu tedy není třeba nějaké specializované aplikace. Společně s touto aplikací je třeba mít na serveru instalovanou i databázi MySQL 5.x a mít zajištěnu podporu pro perzistentní vrstvu Hibernate.

V této databázi musí být vytvořeny tabulky a mohou být vložena i data pro otestování správného nastavení aplikace a databáze. Vytvoření těchto tabulek a vložení dat zajistí spuštění souborů `createDatabaseTables.sql` a `insertTestData.sql`.

Celý postup nasazení aplikace se skládá z těchto 4 kroků:

1. Instalace databázového systému MySQL 5.x a spuštění databázových skriptů pro vytvoření struktury tabulek a naplnění testovacími daty.
2. Konfigurace perzistentní vrstvy Hibernate.
3. Připojení k aplikaci pomocí internetového prohlížeče.

Tyto kroky jsou detailněji popsány v následujících kapitolách.



Obrázek č. 19

Instalace MySQL 5.x

Databáze MySQL je natolik základní databází, že je velmi obtížné nalézt hosting který ji nepodporuje. Ideální je se na toto informovat přímo u poskytovatele Vašeho hostingu. Pokud databáze MySQL nainstalována je, je třeba v ní vytvořit požadovanou strukturu tabulek. To lze udělat pomocí aplikace phpMyAdmin či v ideálním případě spuštěním skriptu createDatabaseTables.sql, který naleznete jako součást aplikace. Ten jednoduše vytvoří celou strukturu databáze za Vás.

Pokud hostujete aplikaci na Vašem vlastním serveru, bude třeba MySQL nainstalovat. Ideální je pro tyto účely instalační balíček MySQL community server, který je zdarma a poskytuje dostatečnou rychlost a komfort práce. Stáhnout si jej můžete z webu <http://dev.mysql.com/downloads/mysql/>. Při tvorbě samotné databázové struktury postupujte stejně jako v předchozím případě.

Nyní máte databázi MySQL nainstalovanou a připravenou k použití.

Konfigurace Hibernate

Knihovny potřebné k běhu perzistentní vrstvy Hibernate jsou zahrnuty ve zdrojových souborech aplikace, není tedy nutné stahovat a nikam umisťovat jakékoli soubory. Je ovšem třeba editovat konfigurační soubor Hibernate a uvést v něm údaje potřebné pro připojení k databázi: adresu databáze, jméno databáze, uživatelské jméno a heslo pro připojení k databázi.

Tyto informace je třeba zanést do souboru hibernate.cfg.xml. Tento soubor je možno editovat jakýmkoli textovým editorem, jen je třeba dát si pozor aby byl uložen zas jako xml soubor. Od tohoto okamžiku bude aplikace schopna se připojit k databázi a pracovat s uloženými daty.

Pokud bude aplikace vykazovat chyby, zkontrolujte, zda jste správně naimportovali strukturu databáze z předchozího kroku.

Připojení k aplikaci

Abychom zkontrolovali, zda proběhla instalace aplikace v pořádku, je třeba se k ní připojit pomocí webového rozhraní. Pokud aplikace neposkytuje nějakou chybovou hlášku, je vše v pořádku.

Gratulujeme, právě jste úspěšně nainstalovali Informační systém pro dětský oddíl.