

**Slovenská technická univerzita**

Fakulta informatiky a informačných technológií

Ilkovičova 3, 842 16 Bratislava 4

---

# **Správa študentských projektov na fakulte**

**Tímový projekt**

**Dokumentácia k inžinierskemu dielu**

**Tím č. 3 - Reyon**

Bc. František Ivanko

Bc. Róbert Móro

Bc. Samuel Snopko

Bc. Marek Sobôtka

Bc. Pavol Sopko

Bc. Ivan Srba

# Obsah

---

Úvod.....	ii
Účel a rozsah dokumentu.....	ii
Slovník pojmov problémovej oblasti .....	ii
Skratky .....	iii
1 Prvý šprint – „Paleolit“ .....	1-1
1.1 Návrh šablóny používateľského rozhrania.....	1-1
1.1.1 Analýza .....	1-1
1.1.2 Návrh.....	1-4
1.1.3 Implementácia .....	1-4
1.2 Prihlasovanie používateľa.....	1-5
1.2.1 Analýza .....	1-5
1.2.2 Návrh.....	1-7
1.2.3 Implementácia .....	1-7
1.2.4 Testovanie .....	1-7
2 Druhý šprint – „Neolit“ .....	2-1
2.1 Pridanie nového projektu.....	2-1
2.1.1 Analýza .....	2-1
2.1.2 Návrh.....	2-3
2.1.3 Implementácia .....	2-4
2.1.4 Testovanie .....	2-5
2.2 Migrácia údajov.....	2-7
2.3 Zobrazenie zoznamu projektov pedagóga.....	2-8
2.3.1 Analýza .....	2-8
2.3.2 Návrh.....	2-10
2.3.3 Implementácia .....	2-10
2.3.4 Testovanie .....	2-10
2.4 Editovanie zadania a oprava revidovaného zadania .....	2-11
2.4.1 Analýza .....	2-12
2.4.2 Návrh.....	2-16
2.4.3 Implementácia .....	2-16
2.4.4 Testovanie .....	2-17
3 Prototyp.....	3-1
3.1 Architektúra systému.....	3-1
3.2 Dátový model .....	3-1
3.2.1 Opis entít .....	3-4
3.2.2 Návrh tabuliek .....	3-4

# Úvod

---

## Účel a rozsah dokumentu

Témou nášho tímového projektu je Správa študentských projektov na fakulte. V súčasnosti sa na fakulte používajú dva systémy na správu projektov – YonBan a Promasus, ktoré vznikli ako výsledok tímových projektov pred ôsmimi rokmi.

*YonBan* slúži na tvorbu tém bakalárskych, diplomových a tímových projektov pedagógmi a ich pridelenie študentom. Podporuje celý proces od vytvorenia zadania projektu pedagógom, cez schválenie zadania garantom a jeho vypísanie a sprístupnenie študentom, až po registráciu študentov na témy a ich pridelenie.

*Promasus* v súčasnosti slúži na nahrávanie prác a abstraktov študentmi a vkladanie posudkov vedúcich a oponentov.

Cieľom nášho tímového projektu je teda vytvoriť nový systém na správu školských projektov, ktorý by oba systémy nahradil, pretože tieto už nevyhovujú potrebám fakulty. Dôraz je kladený na jeho stabilitu, robustnosť a jednoduchú údržbu.

Implementačné prostredie bolo podmienené požiadavkou „zákazníka“, ktorý požadoval použitie programovacieho jazyka PHP a databázového prostredia PostgreSQL. Ako aplikačný rámec sme si pre jeho jednoduchosť zvolili CodeIgniter, ktorý nám uľahčí implementáciu a zároveň v budúcnosti nebude prekážkou pri údržbe systému.

Projekt riešime agilnou vývojovou metodikou Scrum, v ktorej je vývoj rozdelený na dvojtyždňové úseky, tzv. šprinty. V rámci každého šprintu robíme analýzu, návrh, implementáciu a testovanie zvolených funkcionalít, ktoré sú opísané tzv. príbehmi používateľa. Je zvykom šprinty pre ľahšiu identifikáciu pomenúvať podľa nejakého kľúča, my sme si zvolili historické obdobia.

Predkladaný dokument, ktorý predstavuje dokumentáciu k inžinierskemu dielu, odráža toto rozdelenie a zachytáva prvé dva šprinty a celkový pohľad na vytvorený prototyp.

## Slovník pojmov problémovej oblasti

### Vedúci projektu

Je to pedagóg, ktorý vedie projekt. Väčšinou je to tá istá osoba, ako zadávateľ projektu, výnimkou je prípad, keď je zadávateľom niekto mimo fakulty.

### Garant

Je to pedagogický pracovník fakulty, ktorý vedie a zastrešuje (t.j. garantuje) niektorý študijný program. Pravidlá, kto môže byť garantom, upravuje príslušný zákon.

### Projekt

V danej doméne ide o bakalársky, diplomový alebo tímový projekt.

### Zadanie

Téma projektu, vo všeobecnosti platí, že projekt môže mať viacero zadaní (v prípade, že sa zadanie mení počas riešenia projektu).

# **Skratky**

## **BP**

bakalársky projekt

## **DP(1, 2, 3)**

diplomový projekt (1, 2, 3)

## **TP**

tímový projekt

## **AIS**

Akademický informačný systém

# 1 Prvý šprint – *Paleolit*

---

Prvý šprint sme podľa zvoleného kľúča nazvali paleolit, čiže staršia doba kamenná. V rámci neho sme riešili nasledovné úlohy:

1. Analýza existujúcej databázy a návrh nového riešenia
2. Návrh šablóny používateľského rozhrania
3. Prihlasovanie používateľa pomocou údajov do AIS cez LDAP
4. *Vytvorenie nového projektu pedagógom*

Výsledkom prvej úlohy je návrh nového modelu údajov. Čiastkové modely uvádzame v návrhu jednotlivých funkcionalít, celkový pohľad je v časti popisujúcej prototyp. Posledná úloha bola tzv. *stretch task*, čiže navyše pre prípad časovej rezervy v šprinte. Túto úlohu sme však nestihli dokončiť, takže sme ju presunuli do druhého šprintu.

## 1.1 Návrh šablóny používateľského rozhrania

Ako *používateľ* chcem *vidieť* používateľské rozhranie, pomocou ktorého budem môcť *pracovať* so systémom.

### 1.1.1 Analýza

Návrh má obsahovať minimálne:

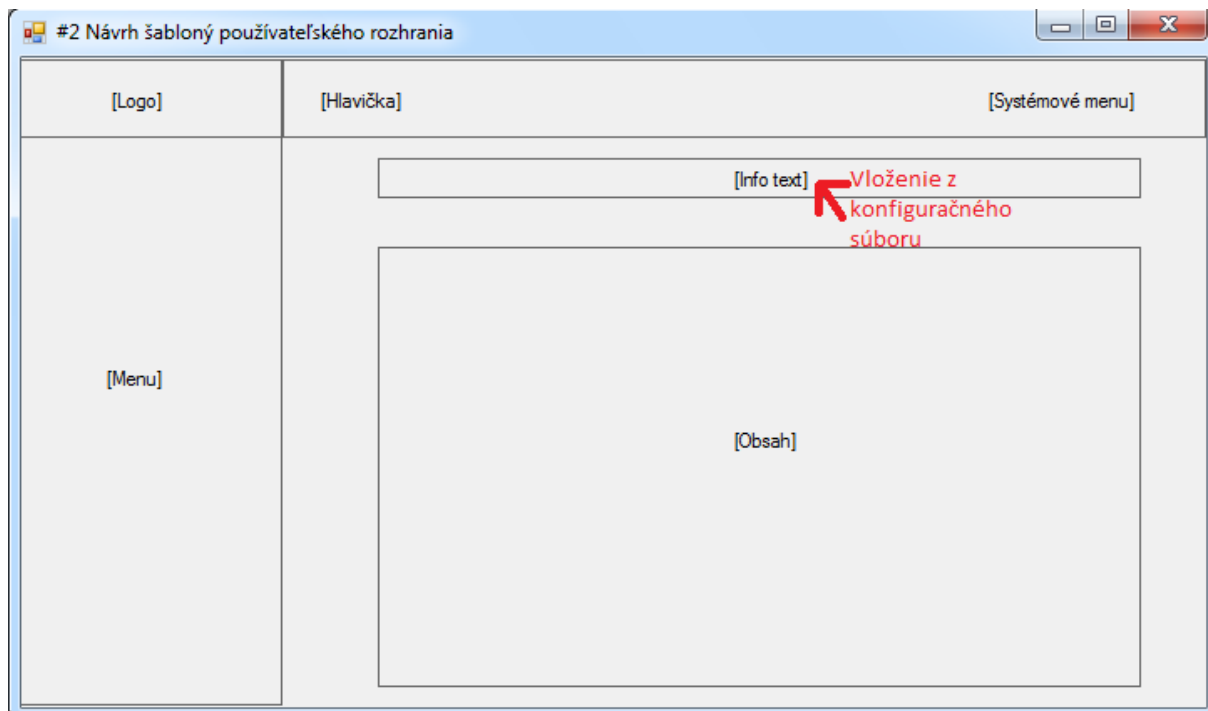
- formulár (obr.1.1, 1.2)
- filtre
- tabuľku údajov
- navigáciu

Na Obr. 1.1 je znázornená hlavná obrazovka systému. Na ľavej strane je menu, ktoré umožňuje navigáciu v systéme, v hornej časti obrazovky je logo, hlavička a systémové menu, ktoré obsahuje položky:

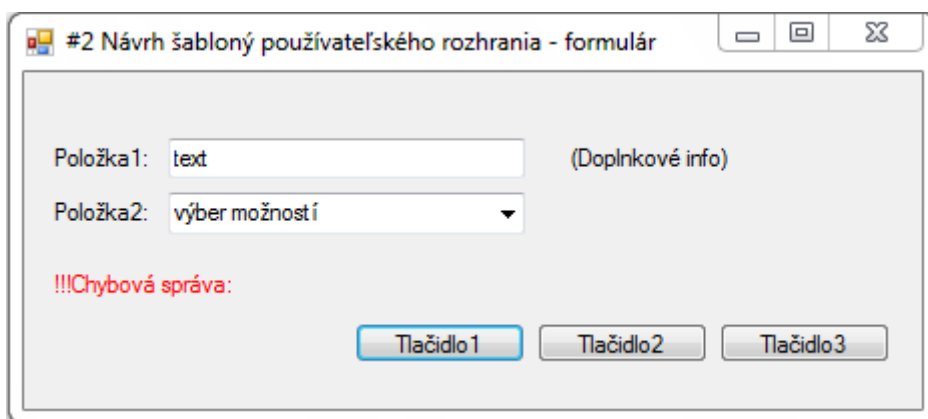
- odkazy na iné systémy
- prihlásený používateľ
- nastavenia
- odhlásenie zo systému

Informačný text sa vkladá z konfiguračného súboru.

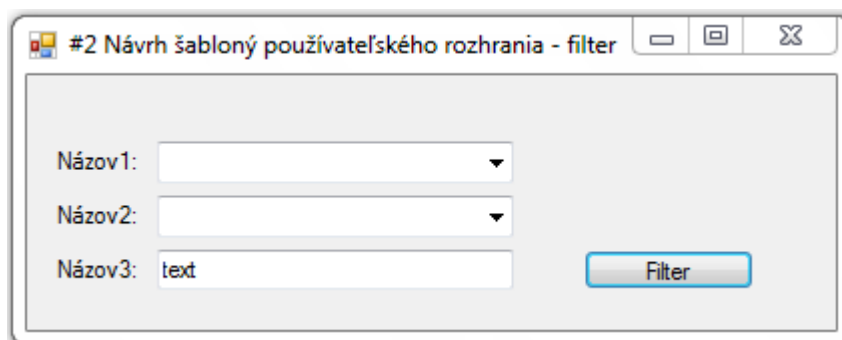
Na Obr. 1.2 je znázornený príklad formulára, na Obr. 1.3 príklad filtra. Na Obr. 1.4 je príklad tabuľky. Údaje v tabuľke je možné usporiadať podľa zvoleného stĺpca.



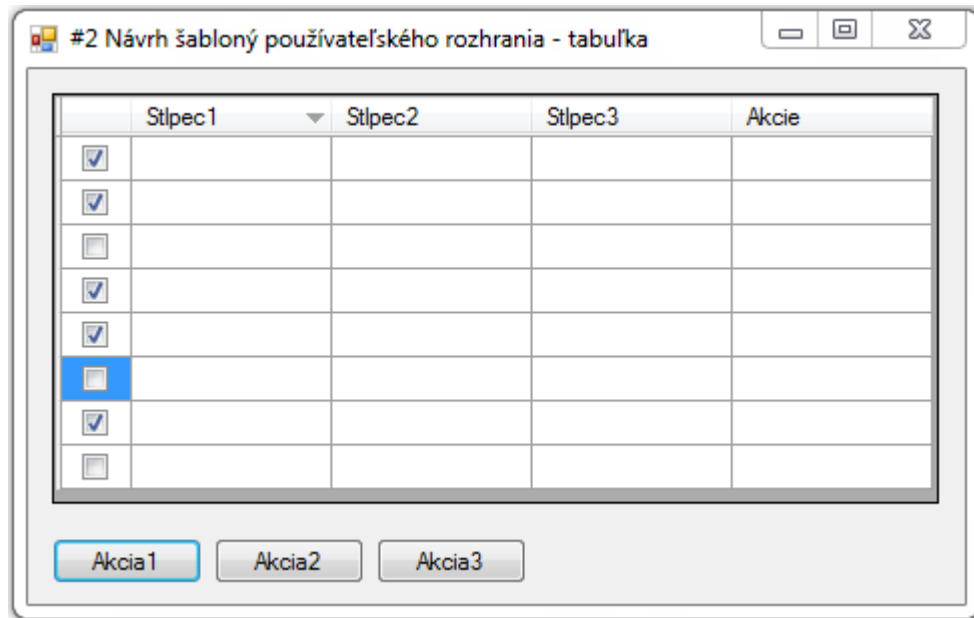
Obr. 1.1 Hlavná obrazovka.



Obr. 1.2 Formulár.



Obr. 1.3 Filter.



Obr. 1.4 Tabuľka.

## Pôvodný systém

Momentálne funkčný systém YonBan (Obr. 1.5) je už niekoľko rokov starý, čo sa neprejavuje len po jeho funkčnej stránke, kde dochádza k výrazným chybám, ale výrazne to je vidieť aj na jeho grafickej/prezentačnej stránke.

Starý systém sa nesie v pomerne smutných modro-sivých farbách. Jeho interakcia s užívateľom je minimálna a môžeme len konštatovať, že niektoré „tlačidlá“ sú aj dosť nešťastne pomenované, kedy nie je hneď na prvý pohľad jasné čo majú vykonať.

Navigácia sa delí na ľavý stĺpec a hornú lištu, kde sú hypertextové odkazy na pravej strane obrazovky, čo nie je v momentálnej dobe štandardné. Taktiež nikde nie je signalizované, na ktorej záložke z tejto hornej lišty sa nachádzame. Celý systém je graficky jednoduchý a pomerne nepríťažlivý.

Název projektu	Vedúci	Pedagogický vedúci	Poč. reg. Akcie
Normalizácia databázy automatizovaným spôsobom	RNDr. Jana Parížková		Pridelený []
Normalizácia databázy automatizovaným spôsobom	RNDr. Jana Parížková		Pridelený []
Algoritmus segmentácie obrazu s použitím metód distribuovaných agentov pre získavanie informácie z lekárskeho snímkov	Ing. Vanda Benešová PhD.		Pridelený []
Aspektovo-orientovaná realizácia zmien	Ing. Valentino Vranič PhD.		Pridelený []
Aspektovo-orientovaná realizácia zmien	Ing. Valentino Vranič PhD.		Pridelený []
Aspektovo-orientované modelovanie a roly	Ing. Valentino Vranič PhD.		Pridelený []
Aspektovo-orientované návrhové vzory	Ing. Valentino Vranič PhD.		Pridelený []
Aspektovo-orientované návrhové vzory	Ing. Valentino Vranič PhD.		0 [Registruj]
Aspektový prístup, syntéza metód a grafická podpora pri refaktoringu	Ing. Ivan Polášek PhD.		2 [Odregistruj]
Automatizované plánovanie	Ing. Marián Lekavý PhD.		Pridelený []
Automatizované plánovanie	Ing. Marián Lekavý PhD.		1 [Registruj]
Automatizácia návrhu integračných riešení	Mgr. Pavol Mederly		1 [Registruj]
Bezpečnostná analýza virtualizácie počítačov	doc. Ing. Ladislav Hudec CSc.		2 [Registruj]
Decentralizované modelovanie používateľa a personalizácia	Ing. Michal Barla		4 [Registruj]
Digitálne knižnice v kultúrnom dedičstve	Ing. Viera Rozinajová PhD.		Pridelený []
Digitálne knižnice v kultúrnom dedičstve	Ing. Viera Rozinajová PhD.		Pridelený []
Dolovanie v (mikro)blogoch	Ing. Marián Šimko		Pridelený []
Dolovanie v (mikro)blogoch	Ing. Marián Šimko		Pridelený []
Emergencia stratégie hry piškvočky pomocou učenia s odmenou a trestom	prof. Ing. Vladimír Kvasnička DrSc.		Pridelený []

Obr. 1.5 Rozhranie pôvodného systému YonBan.

## Požiadavky zákazníka

V prvom rade treba povedať, že nevytvárame nový systém, a teda zákazník aj používatelia systému už prišli do kontaktu so starým systémom. Toto treba brať do úvahy, aby prechod na nový systém nebol komplikovaný pre starých používateľov. Výhodou je, že zákazník vie pomerne presne, ako by mal tento nami vyvíjaný systém vyzerat'.

Po predbežnej diskusii sa ukázalo, že prioritné je zachovať v čo najväčšej miere názvy a rozloženie pôvodného systému. Toto rozloženie je spomenuté vyššie. Pre zákazníka je teda v prvom rade dôležitá prehľadnosť a jasnosť, kde sa nachádzame, a čo môže kde vykonať.

V danej fáze sme ešte neriešili farebnú schému, preto sme použili nami zvolené farby, o ktorých si myslíme, že až na málo zmien by mohli byť použité aj vo finálnej verzii. Táto problematika sa bude riešiť až vo finálnych fázach projektu, keďže primárne ide o funkcionálnosť, no z dôvodu prezentácie je potrebné vytvoriť aspoň nejaké základné grafické rozhranie.

### 1.1.2 Návrh

Z pôvodného systému navrhujeme zachovať rozloženie stránky na ľavý navigačný panel, horný navigačný panel a telo stránky ohraničené týmito panelmi.

Horný navigačný panel bude prioritný. Z tohto dôvodu sa bude niest' v tmavšej farbe, aby bol jasne čitateľný a prehľadný. Jednotlivé odkazy sa na tomto navigačnom paneli budú zobrazovať ako záložky, ktoré budú v prípade aktivity farebne prepojené s telom, ku ktorému patria. Odkazy na tomto paneli reprezentujú nejaký väčší celok ako napríklad „Prehľad“ , „Môj Profil a podobne. Podľa týchto odkazov sa môže meniť aj ľavá navigácia pre jednotlivé odkazy. V tomto navigačnom paneli sa ešte zobrazí aj prihlásený užívateľ a možnosť odhlásiť sa. Prihlásený používateľ je zobrazený na pravej strane a záložky začínajú na ľavej strane od loga systému.

Ľavé menu sa bude niest' v svetlejšej farbe, ktorá netvorí takú ostrú hranicu s telom stránky. Čiže toto menu nie je také výrazné ako hlavné horné menu. Položky na tomto menu sa spresnia počas tvorby systému, ale nebudú ovplyvňovať horné menu, ale iba zobrazený obsah stránky.

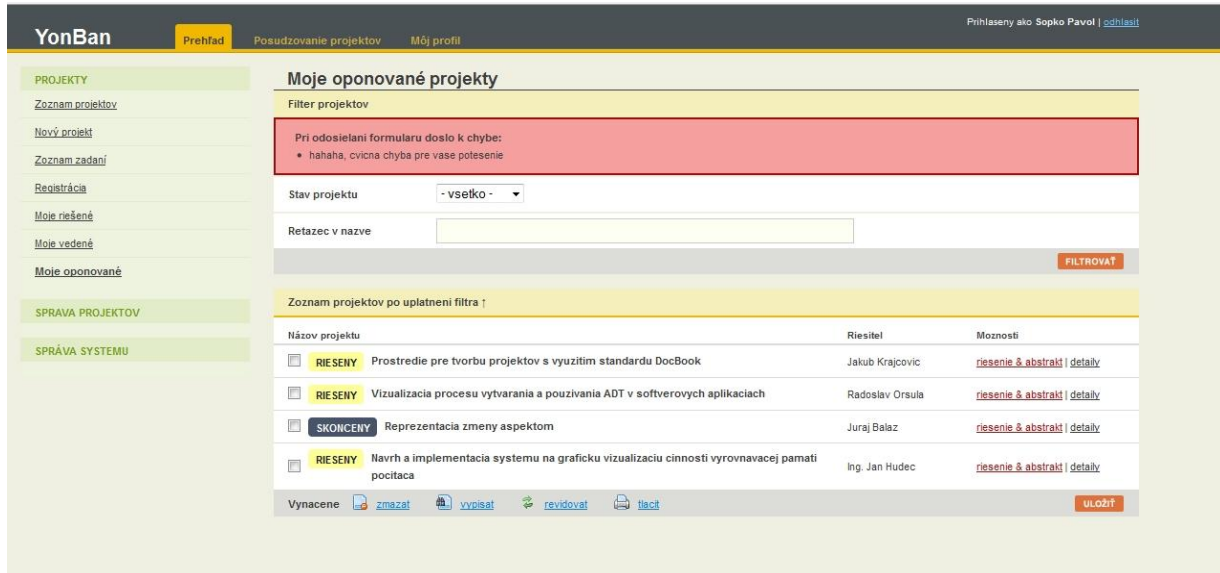
Telo, priestor kde sa zobrazujú jednotlivé formuláre a tabuľky systému, je centrálnou časťou. Oproti pôvodnému systému určite navrhujeme zmeniť používané farby na také, aby bolo telo jasne a dobre čitateľné a nepôsobilo fádne ako terajšie telo. Lepšie je náš zámer vidieť na Obr. 1.6.

Na tomto obrázku je vidieť aj novo navrhnutú tlačidlá („Filtrovat'", „Uložit'"), tabuľky a farbené zobrazenie. Tieto návrhy ešte nie sú finálne, boli len predbežne ukázané zákazníkovi a počas tvorby sa budú konzultovať aj s ostatnými používateľmi systému.

### 1.1.3 Implementácia

Šablóna rozhrania bola implementovaná pomocou HTML5, CSS a JavaScriptu.



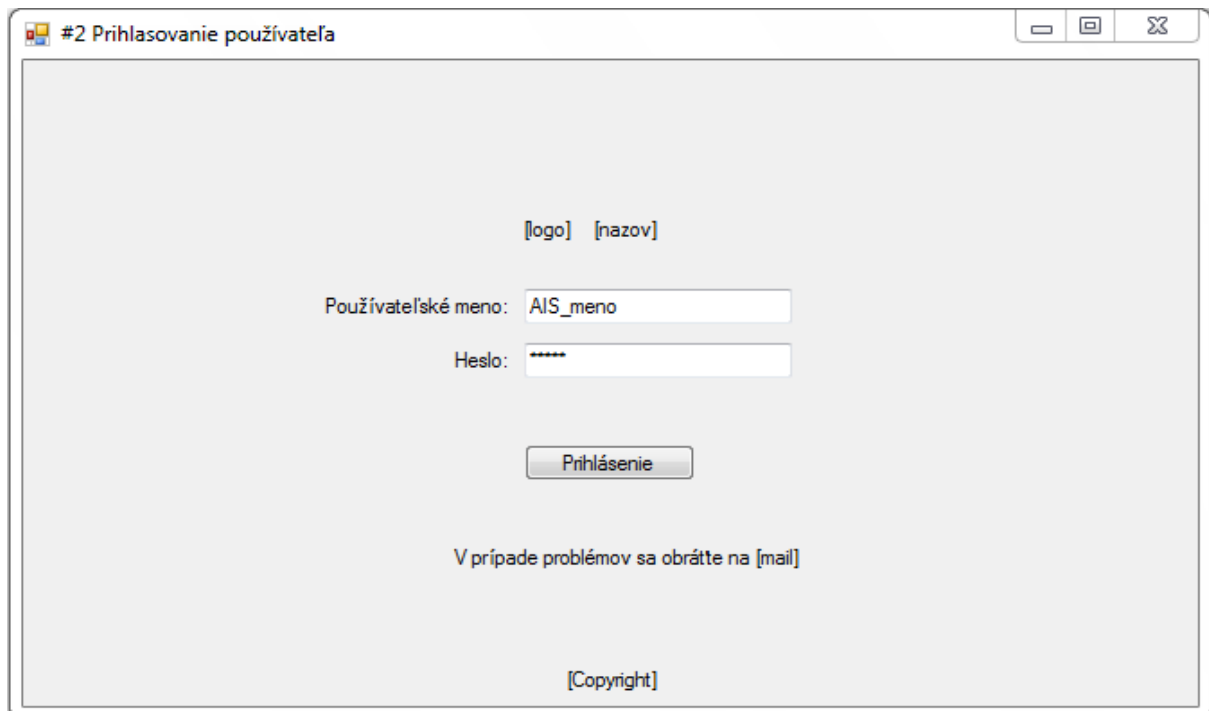


Obr. 1.6 Rozhranie nového systému.

## 1.2 Prihlasovanie používateľa

Ako *používateľ* sa chcem *prihlásiť* do systému, aby som v ňom mohol *pracovať*. Aby som sa nemusel znovu registrovať, chcem použiť prihlasovacie údaje do systému AIS.

### 1.2.1 Analýza



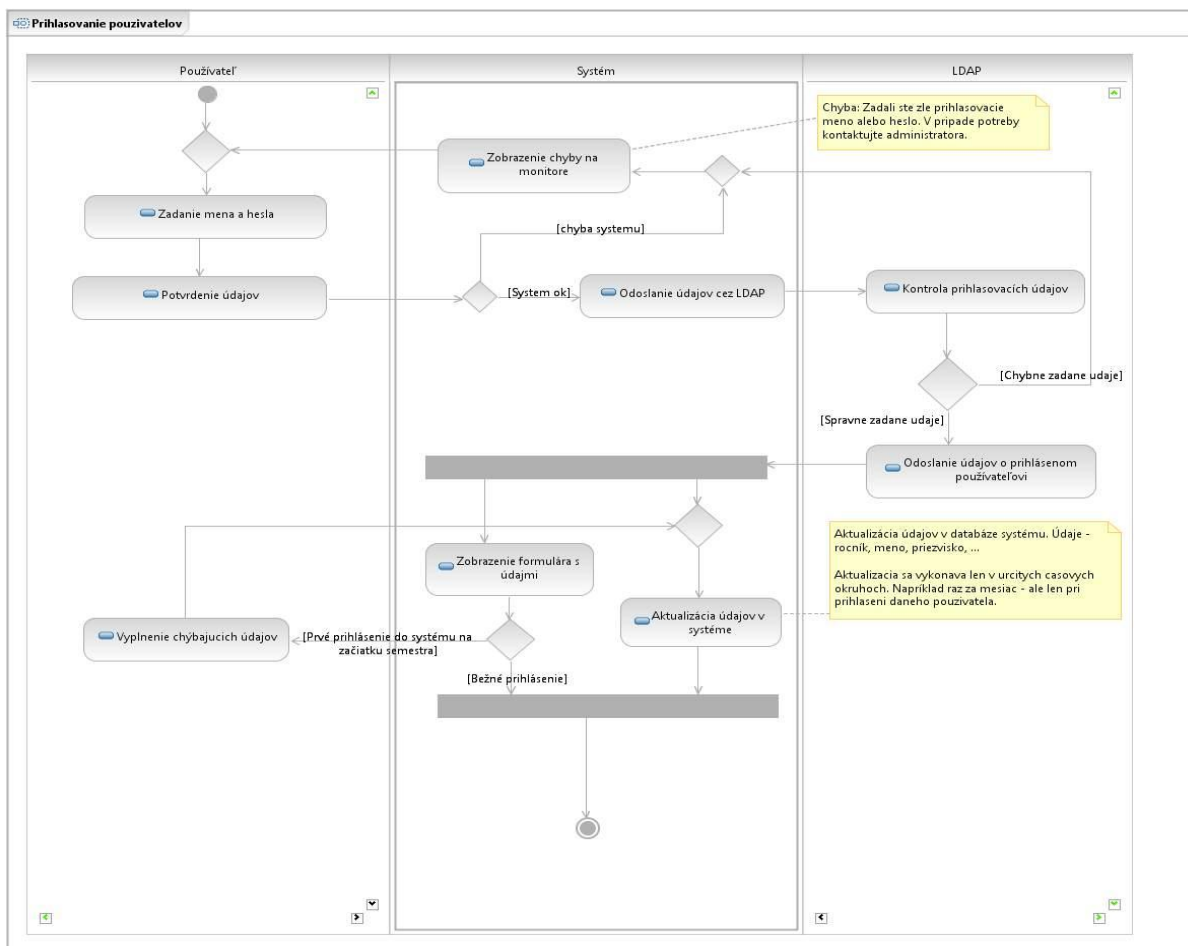
Obr. 1.7 Prihlasovanie používateľa.

Prihlásenie z pohľadu používateľa teda vyzerá nasledovne (Obr. 1.7):

1. Používateľ zadá meno a heslo, ktorým sa prihlasuje do AIS.

2. Systém overí správnosť údajov cez LDAP.
3. V prípade:
  - a. *úspechu*
    - Systém prihlási používateľa do systému.
    - Systém zobrazí všetky informácie, ktoré o používateľovi zistil pomocou LDAP. (Slúži len na otestovanie možností LDAP.)
  - b. *neúspechu* systém zobrazí chybovú správu
    - Chybné meno alebo heslo. (Ak používateľ zadal nesprávne meno alebo heslo.)
    - Zadané meno a heslo sú správne, ale nemáte oprávnenie na prihlásenie do systému. (Ak používateľ nie je importovaný do systému, napr. študent prvého ročníka bakalárskeho štúdia.)
    - Nepodarilo sa prihlásiť, skúste neskôr. (V prípade inej nepredvídanej chyby.)

Názorne je prihlasovanie používateľa zachytené diagramom aktivít (Obr. 1.8). Je v ňom naznačená aj aktualizácia údajov o používateľovi, resp. vyplnenie chýbajúcich údajov používateľom. Toto však nerieši tento prípad použitia a nakoniec sa to možno v budúcnosti zmení. Uvádzame to len preto, aby bolo zrejmé, že bude potrebné nejakým spôsobom zabezpečiť vloženie tých údajov o používateľoch systému, ktoré nie sú prístupné cez LDAP.



Obr. 1.8 Diagram aktivít.

## 1.2.2 Návrh

Návrh je založený na vzore MVC (Model – View – Controller). Prihlasovanie používateľa sa realizuje pomocou knižnice LDAP, ktorá overuje zadané prihlasovacie údaje, preto nie je potrebné uchovávať prihlasovacie údaje v dátovom modeli.

## 1.2.3 Implementácia

Najdôležitejšiu časť prihlasovania tvorí overenie pomocou LDAP.

```
public function authenticate($username, $password)
{
    $ds = @ldap_connect('ldap://ldap.stuba.sk', 636);

    @ldap_set_option($ds, LDAP_OPT_PROTOCOL_VERSION, 3);
    $ldapBindResult = @ldap_bind($ds,
    'uid=.'.$username.',ou=People,dc=stuba,dc=sk', $password);

    if (!$ldapBindResult)
    {
        @ldap_close($ds);
        return NULL;
    }

    $ldapFilter = array("uid", "userPassword", "employeetype", "uisid",
    "cn", "sn", "givenname");

    $ldapSearchResult = @ldap_search($ds, $this->baseDn,
    'uid=.'.$username, $ldapFilter);

    if ($ldapSearchResult)
    {
        $result = @ldap_get_entries($ds, $ldapSearchResult);
    }
    else
    {
        @ldap_close($ds);
        return NULL;
    }
    @ldap_close($ds);
    return $result;
}
```

Tento skript sa volá z príslušného controllera a overí prihlasovacie meno a heslo zadané používateľom na prihlasovacej obrazovke (view).

## 1.2.4 Testovanie

Pre testovanie prihlasovania používateľa sme navrhli nasledovné testovacie scenáre, podľa ktorých sme otestovali funkčnosť implementácie.

<b>Názov</b>	Úspešné prihlásenie	ID Testu	43-01
<b>Rozhranie</b>	Stránka prihlásenia	ID UC	01
<b>Účel</b>	Prihlásenie používateľa		
<b>Vstupné podmienky</b>	Žiadne		
<b>Výstupné podmienky</b>	Žiadne		
<b>Krok</b>	<b>Akcia</b>	<b>Očakávaná akcia</b>	<b>Skutočná reakcia</b>
1.	Zadanie prihlasovacieho mena	Zobrazenie mena v príslušnej kolónke	V kolónke „Používateľské meno“ sa zobrazí zadané meno
2.	Zadanie prihlasovacieho hesla	Zobrazenie hviezdíčiek v kolónke „Heslo“	V kolónke „Heslo“ sa zobrazia hviezdíčky
3.	Kliknutie na tlačidlo „prihlásenie“	Zobrazenie údajov o prihlásenom používateľovi	Používateľ je prihlásený do systému a v pravom hornom rohu sa zobrazí jeho meno, priezvisko a titul

<b>Názov</b>	Neúspešné prihlásenie 1	ID Testu	43-02
<b>Rozhranie</b>	Stránka prihlásenia	ID UC	01
<b>Účel</b>	Zobrazenie chybovej vety		
<b>Vstupné podmienky</b>	Žiadne		
<b>Výstupné podmienky</b>	Žiadne		
<b>Krok</b>	<b>Akcia</b>	<b>Očakávaná akcia</b>	<b>Skutočná reakcia</b>
1.	Zadanie prihlasovacieho mena	Zobrazenie mena v príslušnej kolónke	V kolónke „Používateľské meno“ sa zobrazí zadané meno
2.	Kliknutie na tlačidlo „prihlásenie“	Zobrazenie stránky s chybovou vetou, že nebolo zadané heslo	Zobrazí sa chybová správa „Nezadali ste prihlasovacie meno alebo heslo“

<b>Názov</b>	Neúspešné prihlásenie 2	ID Testu	43-03
<b>Rozhranie</b>	Stránka prihlásenia	ID UC	01
<b>Účel</b>	Zobrazenie chybovej vety		
<b>Vstupné podmienky</b>	Žiadne		
<b>Výstupné podmienky</b>	Žiadne		
<b>Krok</b>	<b>Akcia</b>	<b>Očakávaná akcia</b>	<b>Skutočná reakcia</b>
1.	Zadanie prihlasovacieho hesla	Zobrazenie hviezdíčiek v kolónke „Heslo“	V kolónke „Heslo“ sa zobrazia hviezdíčky
2.	Kliknutie na tlačidlo „prihlásenie“	Zobrazenie stránky s chybovou vetou, že nebolo zadané meno	Zobrazí sa chybová správa „Nezadali ste prihlasovacie meno alebo heslo“

<b>Názov</b>	Neúspešné prihlásenie 3	ID Testu	43-04
<b>Rozhranie</b>	Stránka prihlásenia	ID UC	01
<b>Účel</b>	Zobrazenie chybovej vety		
<b>Vstupné podmienky</b>	Chybne zadané meno alebo heslo		
<b>Výstupné podmienky</b>	Žiadne		
<b>Krok</b>	<b>Akcia</b>	<b>Očakávaná akcia</b>	<b>Skutočná reakcia</b>
1.	Zadanie prihlasovacieho mena	Zobrazenie mena v príslušnej kolónke	V kolónke „Používateľské meno“ sa zobrazí zadané meno
2.	Zadanie prihlasovacieho hesla	Zobrazenie hviezdíčiek v kolónke „Heslo“	V kolónke „Heslo“ sa zobrazia hviezdíčky
3.	Kliknutie na tlačidlo „prihlásenie“	Zobrazenie stránky s chybovou vetou, že meno alebo heslo nie je správne	Zobrazí sa chybová správa „Nesprávne meno alebo heslo“

## 2 Druhý šprint – *Neolit*

Druhý šprint sme nazvali neolit, čiže mladšia doba kamenná. V rámci šprintu sme riešili nasledovné úlohy:

1. Pridanie nového projektu
2. Migrácia údajov
3. Zobrazenie zoznamu zadání pedagóga
4. Editovanie zadania a oprava revidovaného zadania
5. *Tlač projektov*
6. *Zobrazenie zoznamu zadání garantom*

Prvú úlohu sme začali riešiť už v paleolite (prvom šprinte) ako *stretch task*, ale dokončili sme ju až v tomto šprinte, a preto ju uvádzame na tomto mieste. Posledné dve úlohy boli *stretch tasky* pre tento šprint. Nepodarilo sa nám ich však úplne dokončiť, preto sme ich presunuli do nasledujúceho šprintu.

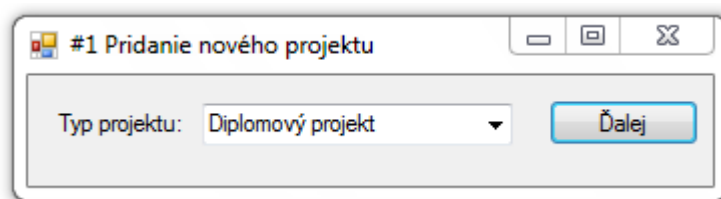
### 2.1 Pridanie nového projektu

Ako *pedagóg* chcem *pridať* nový projekt, aby sa naň mohli *registovať* študenti. Chcem mať možnosť vytváraný projekt *uložiť* a dopracovať ho neskôr, alebo ho rovno *odovzdať* na posúdenie.

#### 2.1.1 Analýza

Najprv si pedagóg zvolí typ projektu (Obr. 2.1). Môže zvoliť jednu z možností:

- Bakalársky projekt (BP)
- Diplomový projekt (DP)
- Tímový projekt (TP)



Obr. 2.1 Typ projektu.

Na základe zvoleného typu projektu sa načíta šablóna príslušného typu zadania a zobrazí sa formulár pre tvorbu nového zadania (Obr. 2.2). Formulár obsahuje dva typy položiek:

- Povinné – vyplňajú sa pri všetkých typoch projektov
- Voliteľné – závisia podľa zvolenej šablóny, resp. typu projektu

Voliteľné položky podľa typu projektu sú nasledovné:

- BP – text
- DP – text, literatúra, poznámka pre študenta
- TP – nemá žiadne voliteľné položky

Položky *typ projektu*, *vedúci projektu* a *rok* sa prednastavia podľa zvoleného typu projektu, mena používateľa, ktorý vytvára projekt a aktuálneho roku a nie je možné ich meniť. Pri položke *študijný*

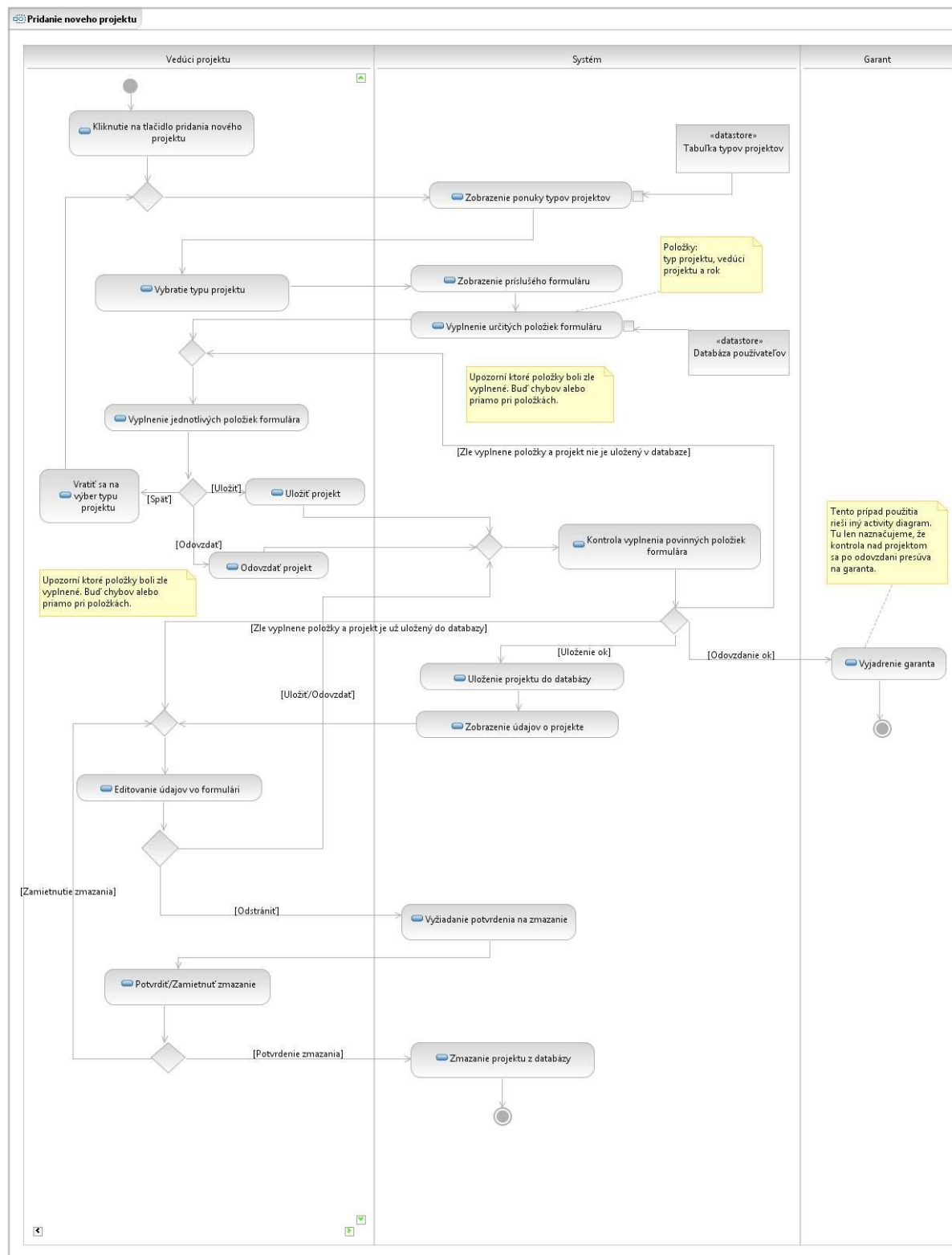
*program* je možné zvoliť viacero možností, pre ktoré sa téma projektu vypisuje. *Pedagogický vedúci* sa vyberá len v prípade, že zadávateľ projektu je mimo fakulty, pričom sa vyberá zo zoznamu všetkých pedagógov.

Pedagóg môže vykonať tri akcie:

1. *Uložiť* rozpracované zadanie. Projekt so zadaním sa zaevidujú v stave *vytváraný*, ale ešte sa neodovzdajú na schválenie a pedagóg môže neskôr vykonať zmeny.
2. *Odovzdať* zadanie. Zadanie sa pred odovzdaním skontroluje, či sú vyplnené všetky povinné položky podľa šablóny:
  - a. Ak áno, projekt so zadaním sa zaevidujú a odovzdajú na posúdenie. Pedagóg ho už ďalej nemôže meniť. Projekt bude mať stav *vytváraný*, zadanie bude *dané na schválenie*.
  - b. Ak nie, označia sa chýbajúce položky a projekt nie je odovzdaný. Pedagóg môže tieto chýbajúce položky doplniť.
3. *Vrátiť sa späť* na výber typu projektu.

Obr. 2.2 Nový projekt.

Na Obr. 2.3 je znázornené vytvorenie nového projektu pomocou diagramu aktivít. V diagrame je naznačená aj možnosť editovať zadanie projektu, čo je riešené v časti 2.4.



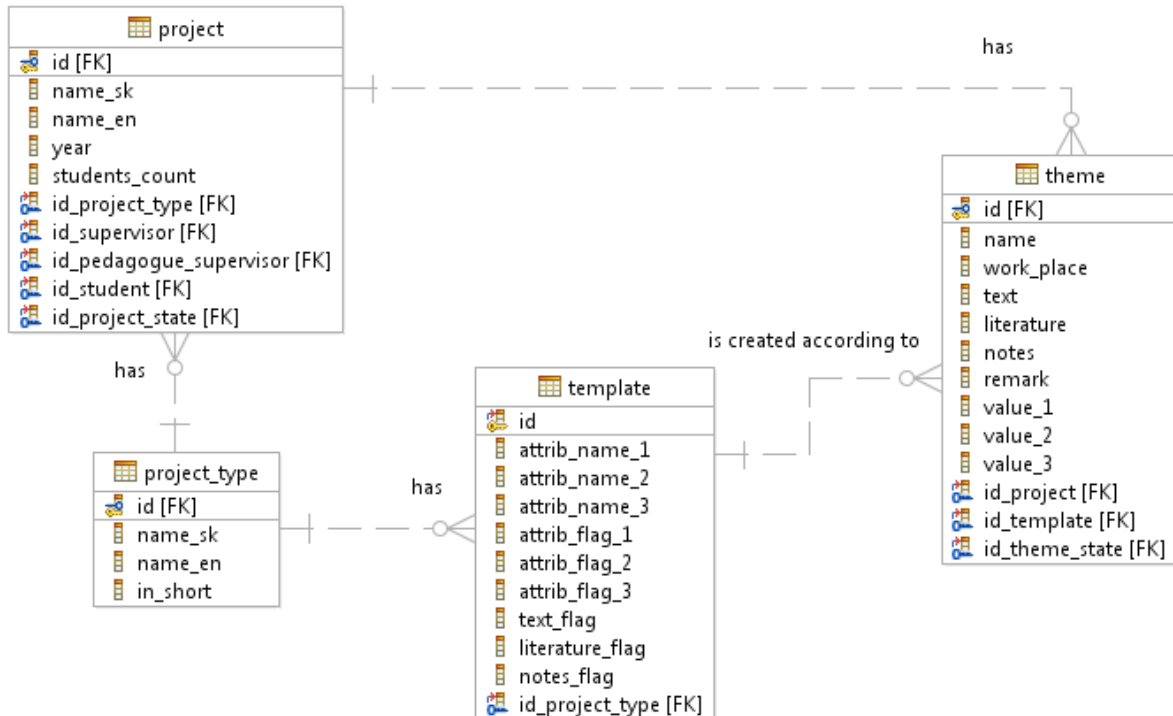
Obr. 2.3 Diagram aktivít.

## 2.1.2 Návrh

Návrh využíva vzor MVC (Model – View – Controller).



Na Obr. 2.4 sú zachytené najdôležitejšie dátové entity a vzťahy medzi nimi. Existuje viacero typov projektov (BP, DP, TP) a pre každý typ projektu je definovaná šablóna zadania. Každý vytváraný projekt je niektorého typu a pre jeden projekt môže byť vo všeobecnosti ľubovoľný počet zadaní (pre prípad úpravy zadania v niektorej fáze projektu). Každé zadanie zodpovedá schéme danej typom projektu.



Obr. 2.4 Čiastkový model údajov.

## 2.1.3 Implementácia

Pridanie nového projektu je implementované nasledovne:

```

public function add()
{
    $this->data['heading'] = 'Vytvorenie nového projektu';
    $this->load->model('Project_type');

    // Zobrazenie prveho kroku
    if (!$this->input->post('id_project_type'))
    {
        $this->data['project_types'] = $this->db->get('project_type')->result();
    }
    // Zobrazenie druhého kroku
    else
    {
        $this->load->model('Teacher');
        $this->load->model('Template');
        $this->load->model('Programme');

        $this->data['id_project_type'] = $this->input->post('id_project_type');
    }
}
    
```

```

        $this->data['project_type_name'] = $this->Project_type-
        >get_name_by_id($this->input->post('id_project_type'));
        $this->data['supervisor_name']      = $this->session-
        >userdata('user_name');
        $this->data['pedagogue_supervisors'] = $this->Teacher-
        >get_teachers();
        $this->data['year'] = date('Y').'/'.date('Y', strtotime('+ 1
        years'));
        $this->data['optional_items'] = $this->Template-
        >get_optional_items($this->input->post('id_project_type'));
        $this->data['programmes']      = $this->Programme-
        >get_programmes();

        $this->data['programme_projects'] = array();
    }

    $this->render();
}

```

Funkcia `save` uloží projekt do databázy a nastaví mu stav *vytváraný*. Vytvorí k nemu tiež príslušné zadanie, buď v stave *vytvárané* alebo *dané na schválenie*, v závislosti na akcii pedagóga.

## 2.1.4 Testovanie

Pre testovanie pridania nového projektu sme navrhli nasledovné testovacie scenáre, podľa ktorých sme otestovali funkčnosť implementácie.

<b>Názov</b>	Úspešné uloženie projektu do databázy	ID Testu	50-01
<b>Rozhranie</b>	Modul správy projektov	ID UC	02
<b>Účel</b>	Vloženie nového projektu		
<b>Vstupné podmienky</b>	Prihlásený používateľ s právami vedúceho projektu		
<b>Výstupné podmienky</b>	Žiadna		
<b>Krok</b>	<b>Akcia</b>	<b>Očakávaná akcia</b>	<b>Skutočná reakcia</b>
1.	Kliknutie v ľavom menu na odkaz „vloženie nového projektu“	Zobrazenie formuláru pre výber typu projektu	Zobrazí sa formulár s výberom typu projektu
2.	Zvolenie typu projektu	Zobrazenie formuláru pre vloženie konkrétneho typu projektu s vyplnenými údajmi o vedúcom	Zobrazí sa formulár pre vypísanie údajov pre nový projekt s vyplnenými údajmi o vedúcom
3.	Vyplnenie formuláru a kliknutie na „uložiť“	Zobrazí sa tabuľka projektov vedúceho aj s týmto projektom	Projekt sa uloží do databázy a zobrazí sa tabuľka projektov vedúceho spolu s novovytvoreným projektom

<b>Názov</b>	Neúspešné uloženie projektu do databázy	ID Testu	50-02
<b>Rozhranie</b>	Modul správy projektov	ID UC	02
<b>Účel</b>	Upozornenie na nevyplnené položky		
<b>Vstupné podmienky</b>	Prihlásený používateľ s právami vedúceho projektu		
<b>Výstupné podmienky</b>	Žiadna		
<b>Krok</b>	<b>Akcia</b>	<b>Očakávaná akcia</b>	<b>Skutočná reakcia</b>
1.	Kliknutie v ľavom menu na odkaz „vloženie nového projektu“	Zobrazenie formuláru pre výber typu projektu	Zobrazí sa formulár s výberom typu projektu
2.	Zvolenie typu projektu	Zobrazenie formuláru pre vloženie konkrétneho typu projektu s vyplnenými údajmi o vedúcom	Zobrazí sa formulár pre vypísanie údajov pre nový projekt s vyplnenými údajmi o vedúcom
3.	Chybné vyplnenie formuláru a kliknutie na „uložiť“ alebo „odovzdať“	Zobrazenie formuláru s údajmi o projekte s výpisom chyby, ktorá nastala. (Pri položkách napísané „Treba vyplniť“)	Zobrazí sa výpis chýb, ktoré je nutné opraviť

<b>Názov</b>	Úspešné odovzdanie projektu garantovi	ID Testu	50-03
<b>Rozhranie</b>	Modul správy projektov	ID UC	02
<b>Účel</b>	Odovzdanie nového projektu		
<b>Vstupné podmienky</b>	Prihlásený používateľ s právami vedúceho projektu		
<b>Výstupné podmienky</b>	Žiadna		
<b>Krok</b>	<b>Akcia</b>	<b>Očakávaná akcia</b>	<b>Skutočná reakcia</b>
1.	Kliknutie v ľavom menu na odkaz „vloženie nového projektu“	Zobrazenie formuláru pre výber typu projektu	Zobrazí sa formulár s výberom typu projektu
2.	Zvolenie typu projektu	Zobrazenie formuláru pre vloženie konkrétneho typu projektu s vyplnenými údajmi o vedúcom	Zobrazí sa formulár pre vypísanie údajov pre nový projekt s vyplnenými údajmi o vedúcom
3.	Vyplnenie formuláru a kliknutie na „odovzdať“	Zobrazí sa tabuľka projektov vedúceho aj s týmto projektom bez možnosti editovať so statusom, že bol projekt poslaný garantovi na potvrdenie	Projekt sa uloží do databázy a zobrazí sa tabuľka projektov vedúceho spolu s novovytvoreným projektom, ktorý už nemožno editovať

<b>Názov</b>	Návrat na výber typu projektu	ID Testu	50-04
<b>Rozhranie</b>	Modul správy projektov	ID UC	02
<b>Účel</b>	Zmena typu projektu		
<b>Vstupné podmienky</b>	Otvorený formulár určitého typu projektu		
<b>Výstupné podmienky</b>	Žiadna (Nový projekt v databáze)		
<b>Krok</b>	<b>Akcia</b>	<b>Očakávaná akcia</b>	<b>Skutočná reakcia</b>
1.	Kliknutie v ľavom menu na odkaz „vloženie nového projektu“	Zobrazenie formuláru pre výber typu projektu	Zobrazí sa formulár s výberom typu projektu
2.	Zvolenie typu projektu	Zobrazenie formuláru pre vloženie konkrétneho typu projektu s vyplnenými údajmi o vedúcom	Zobrazí sa formulár pre vypísanie údajov pre nový projekt s vyplnenými údajmi o vedúcom
3.	Stlačenie tlačidla „späť“	Zobrazenie formuláru pre výber typu projektu	Zobrazí sa formulár s výberom typu projektu

## 2.2 Migrácia údajov

Ako bolo spomenuté vyššie, v súčasnosti už existujú na fakulte dva systémy na správu študentských projektov – *YonBan* a *Promasus*. Oba systémy pracujú nad spoločnou databázou, ktorá predstavuje historickú bázu dát. Túto bázu bolo nutné zachovať, preto sme museli vykonať migráciu starých údajov do nového, nami navrhnutého dátového modelu. Nový dátový model je podrobne popísaný v časti opisujúcej prototyp, preto ho tu neuvádzame.

V tejto fáze sme riešili len migráciu časti databázy. Bolo potrebné zabezpečiť konzistentnosť údajov a vyriešiť konflikty, ktoré vznikli rozdelením niektorých entít zo starého dátového modelu v novom modeli. Napríklad entita *Používateľ* sa rozdelila na entity *Študent* a *Pedagóg*. Konflikt nastal v prípade, že bývalý študent sa neskôr stal pedagógom. Toto sme vyriešili premigrovaním daného používateľa do oboch entít.

Na účel migrácie sme vytvorili nasledovný skript, ktorý však neuvádzame celý:

```
function migrate()
{
    //Vyprazdni novu databazu
    $this->delete();
    //Premigruje pouzivatelov - studentov a pedagogov
    $this->people();
    //Premigruje projekty
    $this->projects();
    //Premigruje zadania projektov
    $this->themes();
    //Premigruje registracie studentov
    $this->registration();
    //Premigruje mapovania projektov na studijne programy
    $this->programme_project();
}
```

```
//Riesi konflikty, vola sa z metody people
public function people_collisions()
{
    $sql = 'SELECT DISTINCT pouzivatel.* FROM pouzivatel
          JOIN projekt ON pouzivatel.id = projekt.id_riesitel
          WHERE pouzivatel.skupina != 2 ';

    $people_query = $this->db_old->query($sql);

    foreach ($people_query->result('array') as $member)
    {
        $student = new Student;
        $student->id = $member['id'];
        $student->personal_number = $member['osobne_cislo'];
        $student->name = $member['meno'];
        $student->surname = $member['priezvisko'];
        $student->title_before =
            ((isset($title_before[$member['titul_pred']])) ?
            $title_before[$member['titul_pred']] : '');
        $student->title_after =
            ((isset($title_after[$member['titul_za']])) ?
            $title_after[$member['titul_za']] : '');
        $student->email = $member['email'];
        $student->telephone = $member['telefon'];
        $student->contact = $member['kontakt'];
        $student->id_programme = $member['id_zameranie'] + 1;

        $this->db->insert('student', $student);
    }
}
```

## 2.3 Zobrazenie zoznamu projektov pedagóga

Ako *pedagóg* si chcem *zobraziť* zoznam svojich projektov, aby som ich mohol *editovať* alebo *odoslať* na schválenie.

### 2.3.1 Analýza

Pedagóg zvolí zobrazenie projektov. Systém zobrazí projekty pedagóga do tabuľky.

V prvom stĺpci tabuľky v každom riadku bude checkbox na zakliknutie/označenie projektu pre viacnásobný výber a v prvom riadku tohto stĺpca checkbox na výber všetkých projektov. Tabuľka nemusí byť sortovateľná/usporadúvateľná podľa stĺpcov. Neskôr v prípade potreby vznikne filter, ale počet projektov pedagóga je minimálny, preto nie je potreba usporadúvania.

Štruktúra tabuľky je podrobne opísaná nižšie, na Obr. 2.5 je znázornená graficky.

#### 1. Tabuľka

<i>Typ</i>	-	Typ projektu (jedna z možností BP, DP, TP).
<i>Názov</i>	-	Reťazec znakov – rôznej dĺžky.
<i>Študijný program</i>	-	Pevne stanovené (PSS, ING, IS, SI)
<i>Vedúci</i>	-	Pedagogický vedúci projektu, ak projekt vypísal niekto mimo fakulty.
<i>Riešiteľ</i>	-	Meno riešiteľa. Môže ostať aj prázdne.
<i>Stav</i>	-	Aktuálny stav projektu.

Akcie - 3. Obrázkové tlačidlá

**1. Tabuľka**

Checkbox all	Typ	Názov	Študijný program	Vedúci	Riešiteľ	Stav	Akcie
ChB							
ChB							
ChB							

**2. Tlačidlá pre prácu**

Tlačidlá pre prácu nad viacnásobným označením projektov.

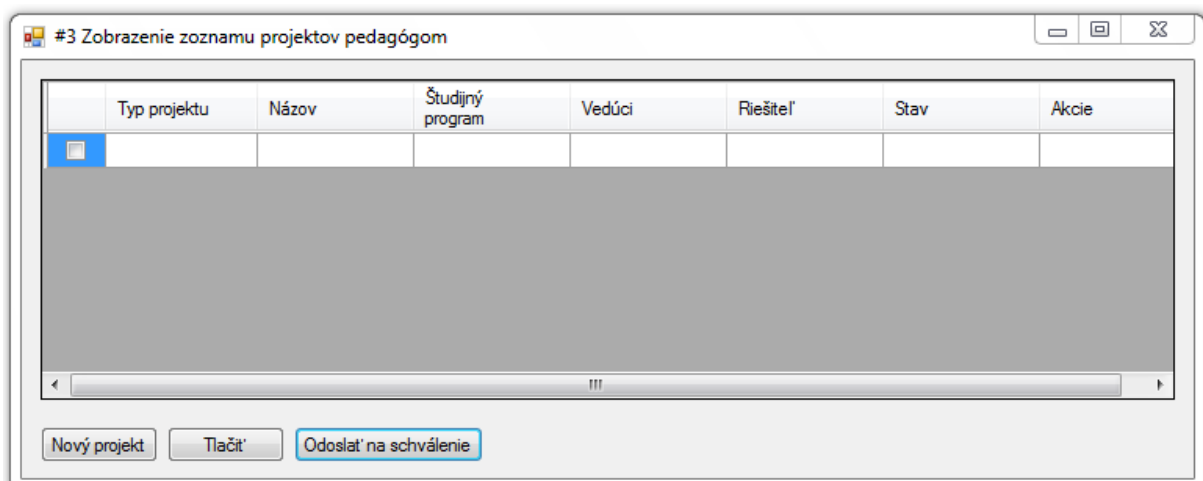
**2. Tlačidlá pre prácu**

- Nový projekt* - Otvorí formulár na vytváranie nových projektov.
- Odoslať na schválenie* - Odošle projekty na schválenie garantovi.
- Zobraz na tlač* - Zobrazí projekty na tlač.

**3. Obrázkové tlačidlá**

- Odoslať na schválenie* - Odošle projekt na schválenie garantovi.
- Detail* - Zobrazí formulár projektu s jeho detailmi
- Editovať* - V prípade, že zadanie projektu je v stave *vytvárané* alebo *revidované*, umožní editovať položky zadania.

Obrázky týchto tlačidiel musia byť výstižné a neskôr sa môžu meniť podľa požiadaviek používateľa. Každý obrázok musí mať vysvetľujúci popis, ktorý sa zobrazí pri nájazde myšou (tooltip).

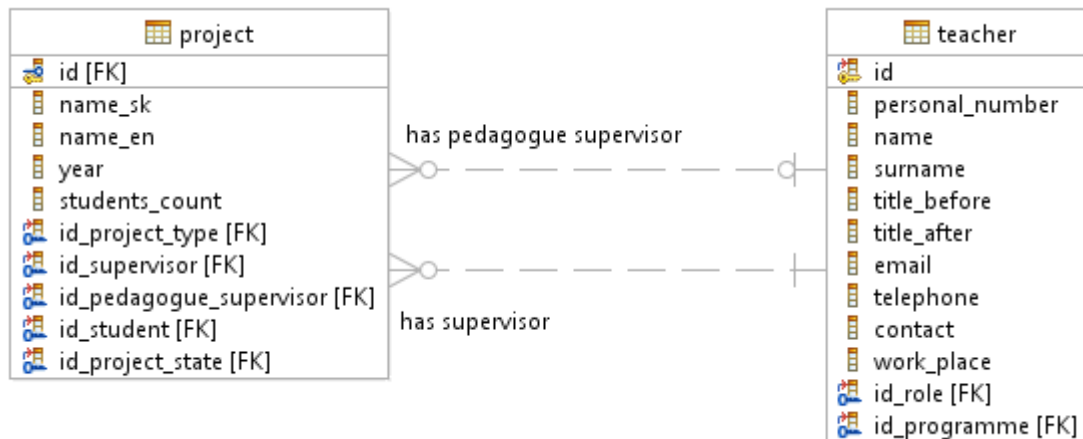


**Obr. 2.5** Zobrazenie zoznamu projektov pedagógom.

## 2.3.2 Návrh

Návrh využíva vzor MVC (Model – View – Controller).

Na Obr. 2.6 sú zachytené najdôležitejšie dátové entity a vzťahy medzi nimi. Projekt má práve jedného vedúceho, zatiaľ čo pedagóg môže viesť viacero projektov. Projekt môže mať nanajvýš jedeného pedagogického vedúceho, ak vedúci je mimo fakulty. Opäť platí, že pedagóg môže byť pedagogický vedúci viacerých projektov.



Obr. 2.6 Čiastkový dátový model.

## 2.3.3 Implementácia

Zobrazenie projektov je implementované nasledovne:

```
public function index()
{
    $this->data['heading'] = 'Zoznam projektov pedagóga';
    $this->load->model('Project');

    $project = new Project();
    $this->data['projects'] = $project->get_projects_by_teacher($this->
    >session->userdata('user_id'));

    $this->data['uri_self'] = 'projects/index';
    $this->render();
}
```

## 2.3.4 Testovanie

Pre testovanie zobrazenia zoznamu projektov pedagóga sme navrhli nasledovné testovacie scenáre, podľa ktorých sme otestovali funkčnosť implementácie.

<b>Názov</b>	Zobrazenie zoznamu projektov	ID Testu	75-01
<b>Rozhranie</b>	Základná obrazovka pedagóga	ID UC	03
<b>Účel</b>	Zobrazenie projektov pedagóga		
<b>Vstupné podmienky</b>	Žiadne		
<b>Výstupné podmienky</b>	Žiadne		
<b>Krok</b>	<b>Akcia</b>	<b>Očakávaná akcia</b>	<b>Skutočná reakcia</b>

1.	V ľavom menu klikneme na tlačidlo pre zobrazenie projektov pedagóga	Zobrazenie tabuľky projektov	Zobrazí sa tabuľka projektov pedagóga
----	---	------------------------------	---------------------------------------

<b>Názov</b>	Označenie projektov	ID Testu	75-02
<b>Rozhranie</b>	Formulár tabuľky projektov pedagóga	ID UC	03
<b>Účel</b>	Označenie projektov		
<b>Vstupné podmienky</b>	Neoznačené všetky projekty		
<b>Výstupné podmienky</b>	Označené všetky projekty		
<b>Krok</b>	<b>Akcia</b>	<b>Očakávaná akcia</b>	<b>Skutočná reakcia</b>
1.	V prvom stĺpci tabuľky klikneme na checkbox prvého projektu	Checkbox ostane zaškrtnutý	Checkbox ostal zaškrtnutý
2.	Klikneme na checkbox pre zaškrtnutie všetkých checkboxov	Zaškrtnú sa všetky checkboxy	Zatiaľ nebolo implementované

<b>Názov</b>	Odškrtnutie projektov	ID Testu	75-03
<b>Rozhranie</b>	Formulár tabuľky projektov pedagóga	ID UC	03
<b>Účel</b>	Odškrtnutie projektov		
<b>Vstupné podmienky</b>	Označené všetky projekty		
<b>Výstupné podmienky</b>	Neoznačené všetky projekty		
<b>Krok</b>	<b>Akcia</b>	<b>Očakávaná akcia</b>	<b>Skutočná reakcia</b>
1.	V prvom stĺpci tabuľky klikneme na checkbox prvého projektu	Checkbox sa odškrtnie	Checkbox sa odškrtol
2.	Klikneme na checkbox pre zaškrtnutie všetkých checkboxov	Odškrtnú sa všetky checkboxy	Zatiaľ nebolo implementované
3.	Klikneme na checkbox pre zaškrtnutie všetkých checkboxov v prípade, že sú v tabuľke projekty zaškrtnuté aj odškrtnuté	Všetky projekty budú odškrtnuté	Zatiaľ nebolo implementované

## 2.4 Editovanie zadania a oprava revidovaného zadania

Ako pedagóg chcem revidovať zadanie, ktoré pripomenkoval garant študijného programu, aby mohlo byť schválené a mohli sa naň registrovať študenti. Takisto chcem ako pedagóg mať možnosť upraviť (editovať) uložený vytváraný projekt, ktorý som zatiaľ nedal na schválenie.



## 2.4.1 Analýza

Ide v zásade o dva rôzne prípady použitia *Oprava revidovaného zadania a Editovanie uloženého zadania projektu*, ktoré sú špecializáciou jedného prípadu použitia. Oba prípady zahŕňajú *Zobrazenie projektov pedagóga (príbeh 2.3)*, pomocou ktorého si pedagóg zobrazí svoje projekty.

V prípade, že pedagóg zadanie projektu uložil, ale ešte ho nedal na schválenie, môže ho *editovať*:

1. Pedagóg zvolí možnosť editovať zadanie projektu.
2. Systém zobrazí rovnaký formulár ako pri vytváraní nového projektu (*príbeh 2.1*), obsah jednotlivých položiek sa načíta podľa toho, ako ich vyplnil pedagóg.
3. Pedagóg edituje položky zadania.
4. Nakoniec má pedagóg tri možnosti:
  - a. *Uložiť* rozpracované zadanie. Uložia sa zmeny, ktoré pedagóg urobil v zadaní, ale zadanie sa ešte sa neodovzdá (zostane v stave *vytvárané*) a pedagóg môže neskôr vykonať zmeny.
  - b. *Odovzdať* zadanie. Zadanie sa pred odovzdaním skontroluje, či sú vyplnené všetky povinné položky podľa šablóny:
    - i. Ak áno odovzdá sa na posúdenie (zmení stav na *dané na schválenie*). Pedagóg ho už ďalej nemôže meniť.
    - ii. Ak nie, označia sa chýbajúce položky a zadanie nie je odovzdané. Pedagóg môže tieto chýbajúce položky doplniť.
  - c. *Vrátiť sa späť* na zoznam projektov pedagóga. Zmeny, ktoré vykonal pedagóg, sa neuložia, projekt zostane v nezmenenom stave. Zadanie zostane neodovzdané a pedagóg môže neskôr vykonať zmeny.

Keď pedagóg odovzdá zadanie projektu na schválenie, je na garantovi daného študijného programu, pre ktorý sa vypisuje, aby zadanie projektu schválil, zamietol, alebo ho dal revidovať (*príbeh 2.5*). V prípade, že garant dá zadanie projektu *revidovať*, môže pedagóg opraviť zadanie tak, aby splnilo pripomienky garanta:

1. Pedagóg zvolí možnosť opraviť zadanie projektu.
2. Systém zobrazí rovnaký formulár ako pri vytváraní nového projektu (*príbeh 2.1*), obsah jednotlivých položiek sa načíta podľa toho, ako ich vyplnil pedagóg. Navrchu formulára sa zobrazí poznámka od garanta v červenom rámečku (Obr. 2.7).
3. Pedagóg edituje položky zadania.
4. Nakoniec má pedagóg tri možnosti:
  - a. *Uložiť* opravované zadanie. Uložia sa zmeny, ktoré pedagóg urobil v zadaní, ale zadanie sa ešte sa neodovzdá, zostane v stave *revidované* a pedagóg ešte môže neskôr vykonať zmeny.
  - b. *Odovzdať* zadanie. Zadanie sa pred odovzdaním skontroluje, či sú vyplnené všetky povinné položky podľa šablóny:
    - i. Ak áno, odovzdá sa na posúdenie (stav zadania sa zmení na *dané na schválenie*). Pedagóg ho už ďalej nemôže meniť.
    - ii. Ak nie, označia sa chýbajúce položky a zadanie nie je odovzdané. Pedagóg môže tieto chýbajúce položky doplniť.
  - c. *Vrátiť sa späť* na zoznam projektov pedagóga. Zmeny, ktoré vykonal pedagóg, sa neuložia, zadanie projektu zostane v stave *revidované* a pedagóg ešte môže neskôr vykonať zmeny.

#4 Oprava revidovaného zadania projektu

[Pripomienky garanta]

Typ projektu: Diplomový projekt

Vedúci projektu: <moje meno>

Pedagogický vedúci: [dropdown menu]

Názov projektu: <Názov projektu>

Študijný program: Infomatika  
PSS

Počet študentov: 0

Miesto vypracovania: <Miesto vypracovania>

Rok: aktuálny/nasledujúci

Text: <Text zadania>

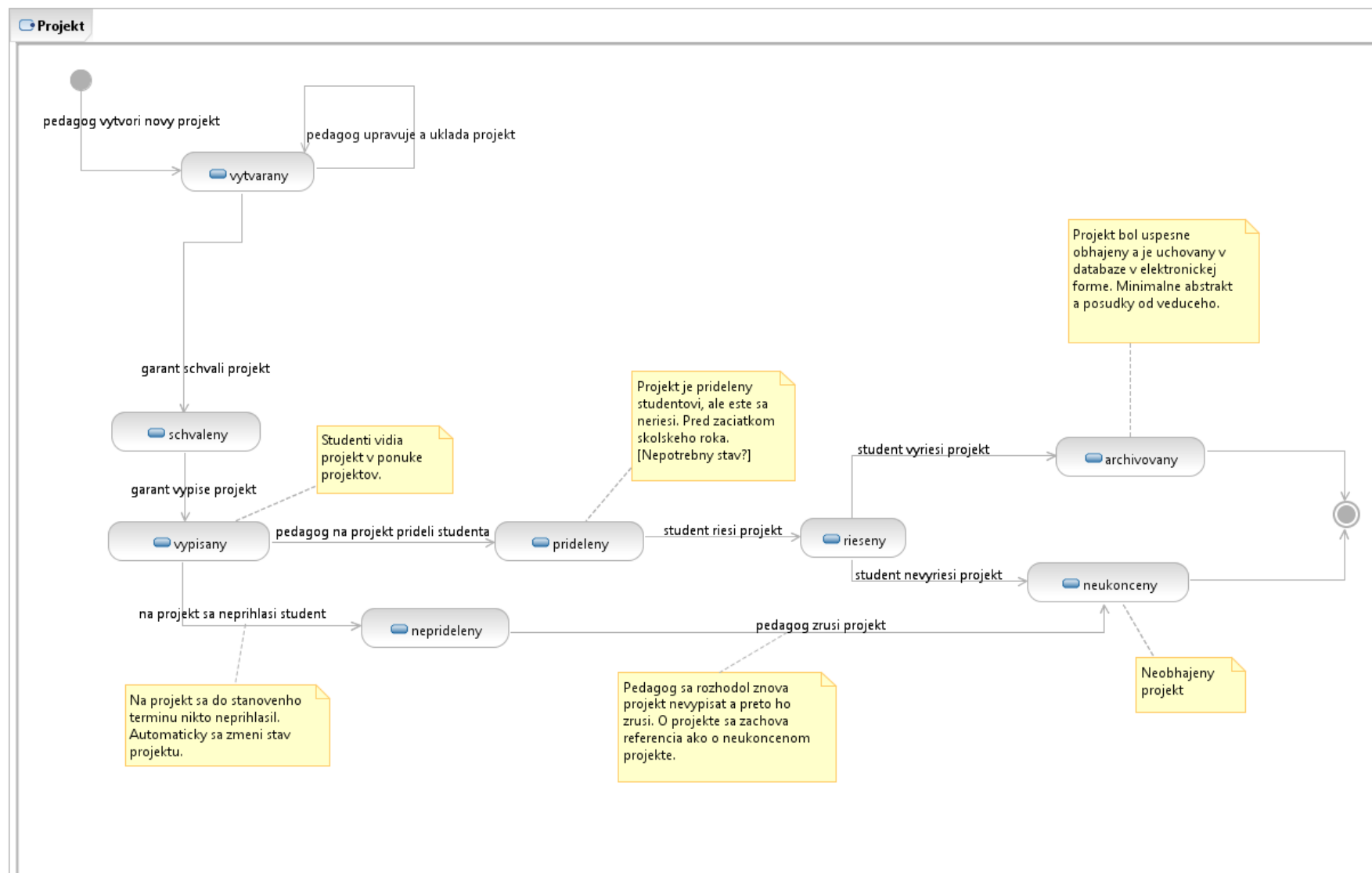
Literatúra: <Literatúra>

Poznámka pre študenta:

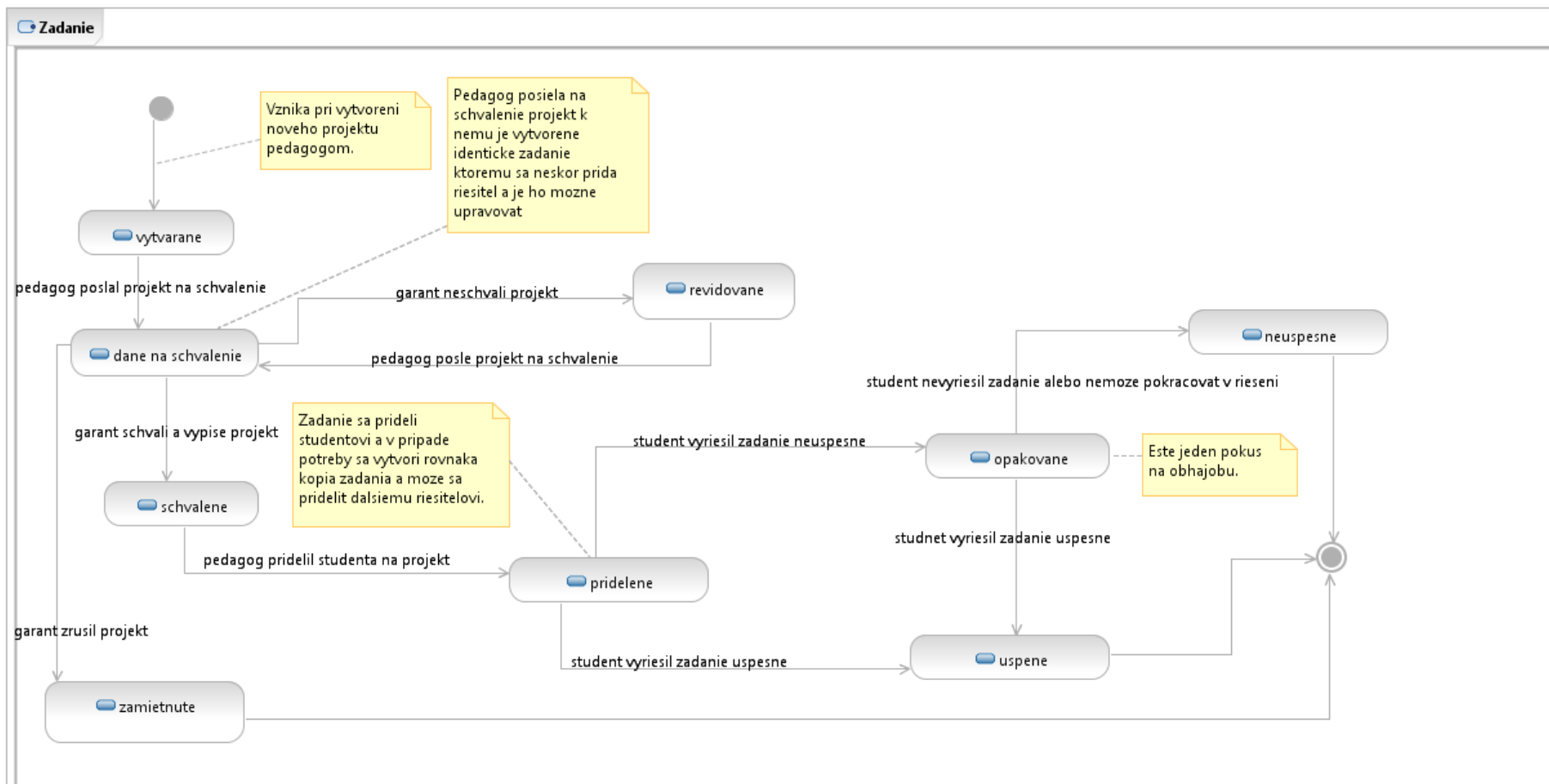
Uložiť Odovzdať Späť

Obr. 2.7 Oprava revidovaného zadania projektu.

Na Obr. 2.8 je znázornený celý životný cyklus projektu pomocou stavového diagramu, na Obr. 2.9 celý životný cyklus zadania. Z hľadiska tohto prípadu použitia je pre projekt dôležitý stav *vytváraný* a pre zadanie stavy *vytvárané*, *dané na schválenie* a *revidované*. Ako môžeme vidieť porovnaním oboch diagramov, projekt svoj stav pri schvaľovaní zadania nemení a zostáva v stave *vytváraný* až do schválenia zadania garantom, kedy sa zmení na *schválený*.



Obr. 2.8 Stavový diagram životného cyklu projektu.

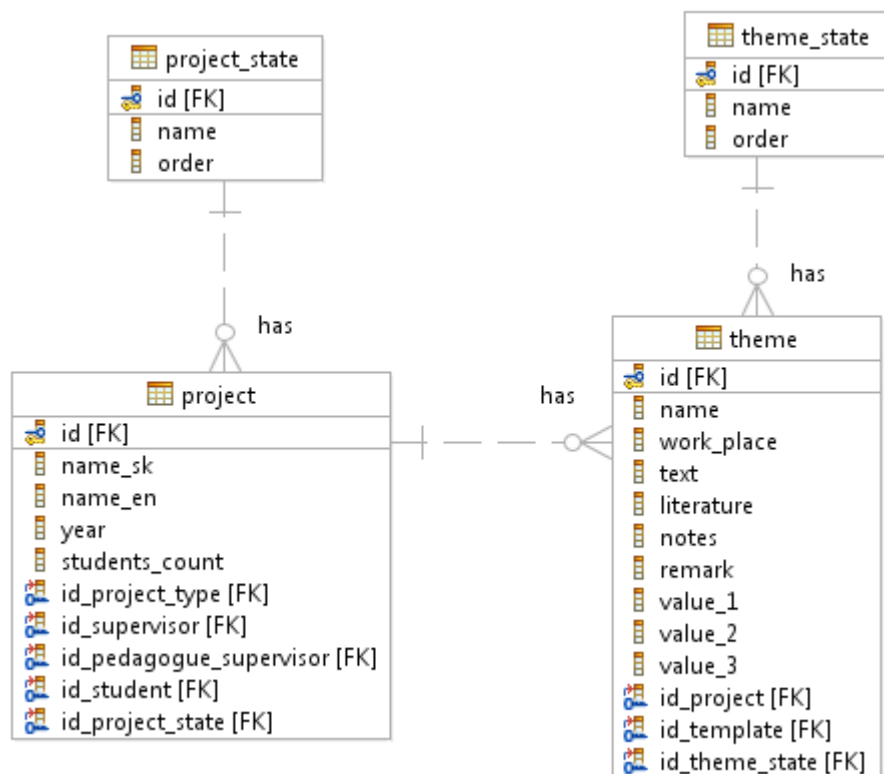


Obr. 2.9 Stavový diagram životného cyklu zadania.

## 2.4.2 Návrh

Návrh vychádza zo vzoru MVC (Model - View - Controller).

Na Obr. 2.10 je sú znázornené najdôležitejšie dátové entity a vzťahy medzi nimi pre prípad použitia oprava revidovaného zadania projektu. Projekt môže mať vo všeobecnosti viacero zadanií. Zadanie aj projekt majú stav. Stav projektu od vytvorenia až po schválenie garantom je *vytváraný*, potom sa zmení na *schválený*. Stav zadania sa mení z *vytváraného* na *dané na schválenie*. Z tohto stavu sa môže dostať do stavu *revidované*, *schválené* alebo *zamietnuté*.



Obr. 2.10 Čiastkový dátový model.

## 2.4.3 Implementácia

Editovanie zadania je implementovaná funkciou `edit`, ktorá sa podobá na funkciu `add` zabezpečujúcu vytváranie nového projektu, kedy sa použije ten istý formulár, len sa načítajú údaje o projekte z databázy. Dôležitá je rovnako ako pri vytváraní projektu funkcia `save`, z ktorej časť tu uvádzame.

```

public function save()
{
    $this->load->model('Project');
    $this->load->model('Theme');
    $this->load->model('Template');

    $project_id = $this->input->post('project_id');

    // Vytvorenie instancie projektu
    $project->id_project_type = $this->input->post('id_project_type');
}
    
```

```

$project->id_supervisor = $this->session->userdata('user_id');
$project->id_pedagogue_supervisor = (($this->input-
>post('id_pedagogue_supervisor')) ? $this->input-
>post('id_pedagogue_supervisor') : NULL);
$project->name_sk = $this->input->post('name_sk');
$project->name_en = $this->input->post('name_en');
$project->year = date('Y').'/'.date('Y', strtotime('+ 1 years'));
$project->state = (($this->input->post('handed_over')) ? '2' : '1' );
$project->students_count = $this->input->post('students_count');

//Uloz, resp. updatuj projekt
...

// Vytvorenie instance zadania
$theme = new Theme();
$theme->id_project = $project_id;
$theme->id_template = $this->Template-
>get_id_template_by_project_type_id($project->id_project_type);
$theme->name = $project->name_sk;
$theme->state = (($this->input->post('handed_over')) ? '2' : '1' );
$theme->work_place = $this->input->post('work_place');

// Spracuj volitelne polozky
foreach ($this->Template->get_optional_items($project-
>id_project_type) as $optional_item)
{
    if ($this->input->post($optional_item['name']))
    {
        $theme->$optional_item['name'] = $this->input-
>post($optional_item['name']);
    }
}

//Uloz, resp. updatuj zadanie
...

...
}

```

## 2.4.4 Testovanie

Pre testovanie editovania zadania a opravy revidovaného zadania sme navrhli nasledovné testovacie scenáre, podľa ktorých sme otestovali funkčnosť implementácie.

<b>Názov</b>	Editovanie zadania projektu	ID Testu	81-01
<b>Rozhranie</b>	Formulár prehľadu projektov pedagóga	ID UC	04
<b>Účel</b>	Zmena a uloženie údajov zadania projektu		
<b>Vstupné podmienky</b>	Vytvorený a uložený projekt v databáze		
<b>Výstupné podmienky</b>	Žiadne		
<b>Krok</b>	<b>Akcia</b>	<b>Očakávaná akcia</b>	<b>Skutočná reakcia</b>
1.	V tabuľke klikneme na „editácia“ v riadku projektu, ktorý chceme upravovať.	Zobrazí sa formulár pre vytváranie projektu s vyplnenými údajmi.	Zobrazí sa formulár pre vytváranie projektu s vyplnenými údajmi
2.	Postupne klikneme a zmeníme každú položku.	Zmena údajov na novo zadané.	V príslušných kolónkach sa zobrazia novo zadané

			údaje
3.	Klikneme na tlačidlo „uložiť“	Uložia sa zmeny a na obrazovke ostane zobrazený aktuálne údaje k projektu.	Uložia sa zmeny a zobrazí sa prehľad projektov pedagóga

<b>Názov</b>	Neúspešné uloženie projektu do databázy	ID Testu	81-02
<b>Rozhranie</b>	Formulár prehľadu projektov pedagóga	ID UC	04
<b>Účel</b>	Upozornenie na nevyplnené položky		
<b>Vstupné podmienky</b>	Vytvorený a uložený projekt v databáze		
<b>Výstupné podmienky</b>	Žiadne		
<b>Krok</b>	<b>Akcia</b>	<b>Očakávaná akcia</b>	<b>Skutočná reakcia</b>
1.	V tabuľke klikneme na „editácia“ v riadku projektu, ktorý chceme upravovať.	Zobrazí sa formulár pre vytváranie projektu s vyplnenými údajmi.	Zobrazí sa formulár pre vytváranie projektu s vyplnenými údajmi
2.	Postupne po jednom zmažeme všetky povinné údaje a pokúsime sa uložiť projekt.	Systém upozorní pedagóga na nevyplnené povinné údaje.	Zobrazí sa prehľad chýb s upozornením na nevyplnené povinné údaje

<b>Názov</b>	Úspešné odovzdanie projektu garantovi	ID Testu	81-03
<b>Rozhranie</b>	Formulár prehľadu projektov pedagóga	ID UC	04
<b>Účel</b>	Odovzdanie upraveného projektu		
<b>Vstupné podmienky</b>	Vytvorený a uložený projekt v databáze		
<b>Výstupné podmienky</b>	Žiadna		
<b>Krok</b>	<b>Akcia</b>	<b>Očakávaná akcia</b>	<b>Skutočná reakcia</b>
1.	V tabuľke klikneme na „editácia“ v riadku projektu, ktorý chceme upravovať.	Zobrazí sa formulár pre vytváranie projektu s vyplnenými údajmi.	Zobrazí sa formulár pre vytváranie projektu s vyplnenými údajmi
2.	Správne upravíme údaje a vyplníme všetky povinné položky a stlačíme odoslať.	Systém odošle projekt garantovi na posúdenie. Zadanie sa prepne do stavu „dané na schválenie“. Následne sa zobrazí formulár prehľadu projektov pedagóga so zmeneným stavom odoslaného projektu.	Projekt sa prepne do stavu „daný na schválenie“ a zobrazí sa prehľad projektov pedagóga so zmeneným stavom odoslaného projektu

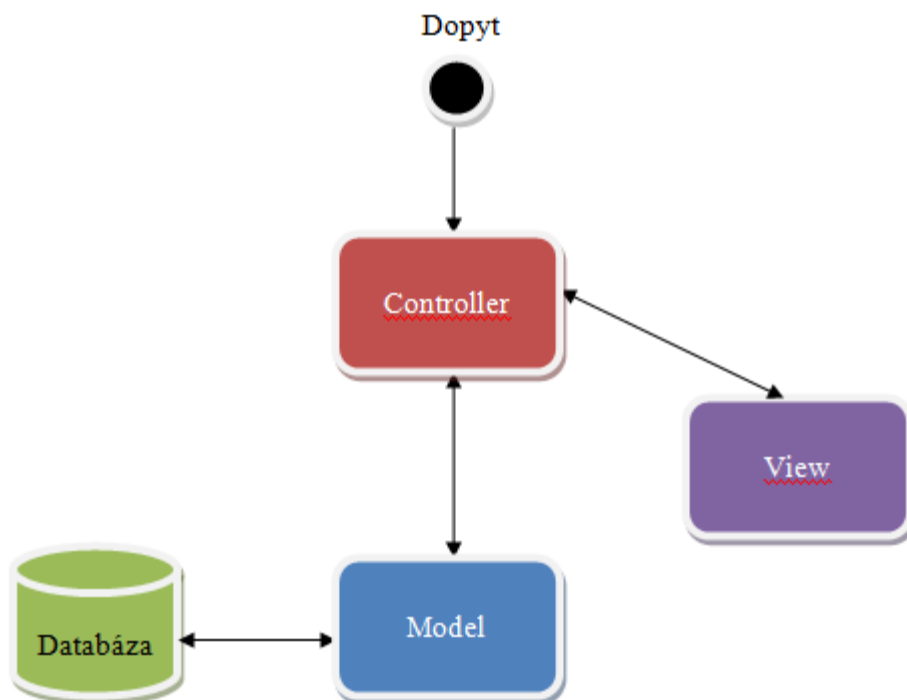
## 3 Prototyp

Keďže postupujeme metodikou Scrum, ktorá je založená na skorom prototypovaní a krátkych vývojových cykloch, vyvinuli sme po prvých dvoch šprintoch prvý prototyp. V tejto kapitole preto uvádzame celkový pohľad na prototyp, jeho architektúru a dátový model.

### 3.1 Architektúra systému

Nami navrhovaný a vyvíjaný systém na správu študentských projektov na fakulte ma klasickú *klient – server* architektúru s tenkým klientom. Na strane servera je databáza a riadiaca vrstva, prezentačná vrstva sa renderuje na strane servera a zobrazuje sa v prehliadači na strane klienta.

Architektúra systému je založená na architektonickom vzore *MVC (Model – View – Controller)*, Obr. 3.1. *Model* predstavuje rozhranie medzi databázou a riadiacou vrstvou. *Controller* (riadiaca vrstva) predstavuje hlavnú funkčnú časť systému, riadi vykonávanie systému, spracúva prichádzajúce dopyty, interaguje s modelom a renderuje pohľad (*view*), čiže prezentačnú časť, ktorá sa zobrazuje používateľovi. *View* teda potom predstavuje rozhranie medzi používateľom a systémom.



Obr. 3.1 MVC.

### 3.2 Dátový model

V prvých dvoch šprintoch sme sa zamerali na vytvorenie dátového modelu, v ktorom by sme vedeli zabezpečiť najmä nasledovné činnosti:

- získať údaje o prihlásenom používateľovi
- pridať alebo uložiť nový projekt



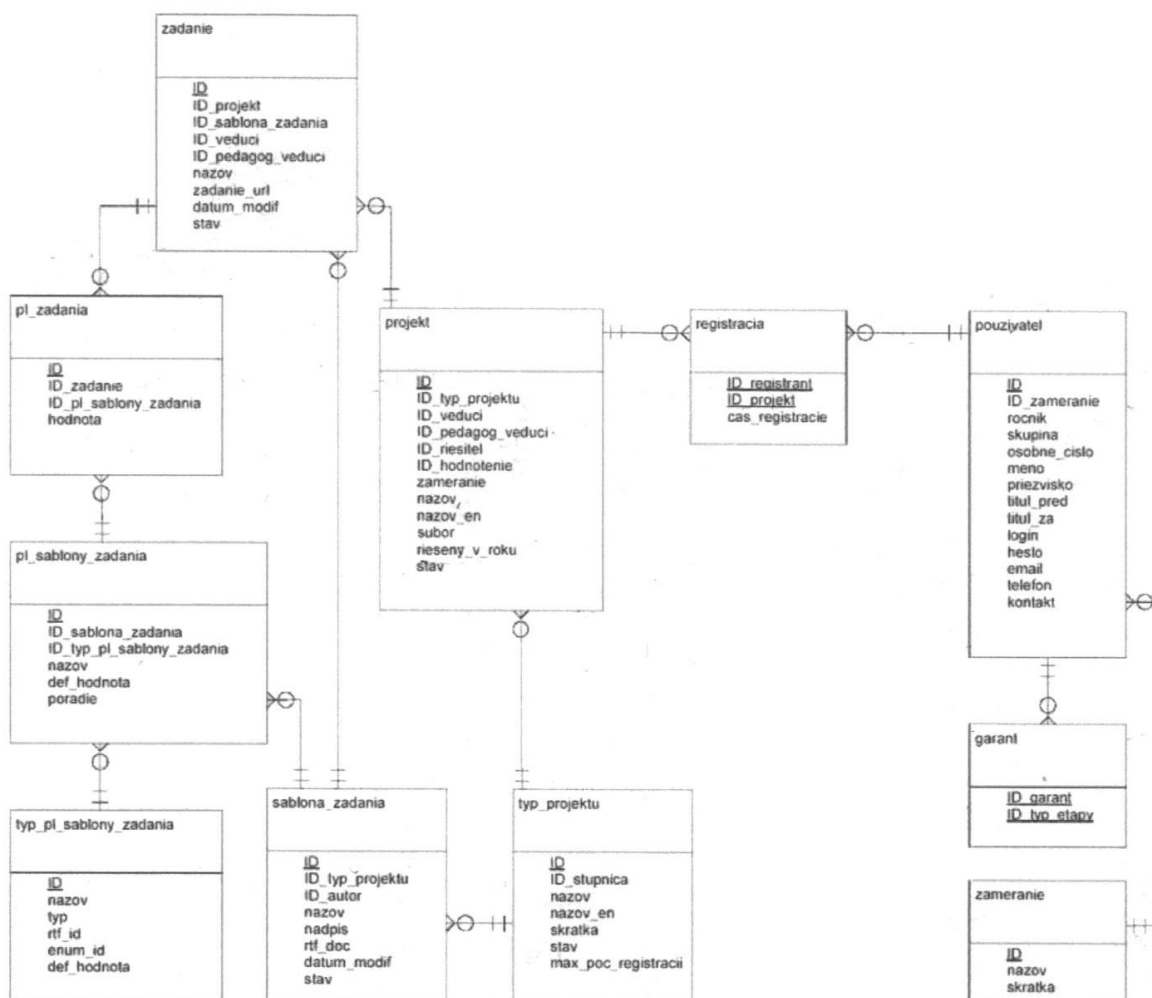
- registrovať študenta na projekt

Pri návrhu nového dátového modelu sme vychádzali z poskytnutého návrhu dátového modelu pôvodného systému *YonBan*. Na Obr. 3.2 je zobrazená časť spomínaného modelu, kde pre názornosť sú zobrazené len tie entity, nad ktorými sme momentálne uvažovali.

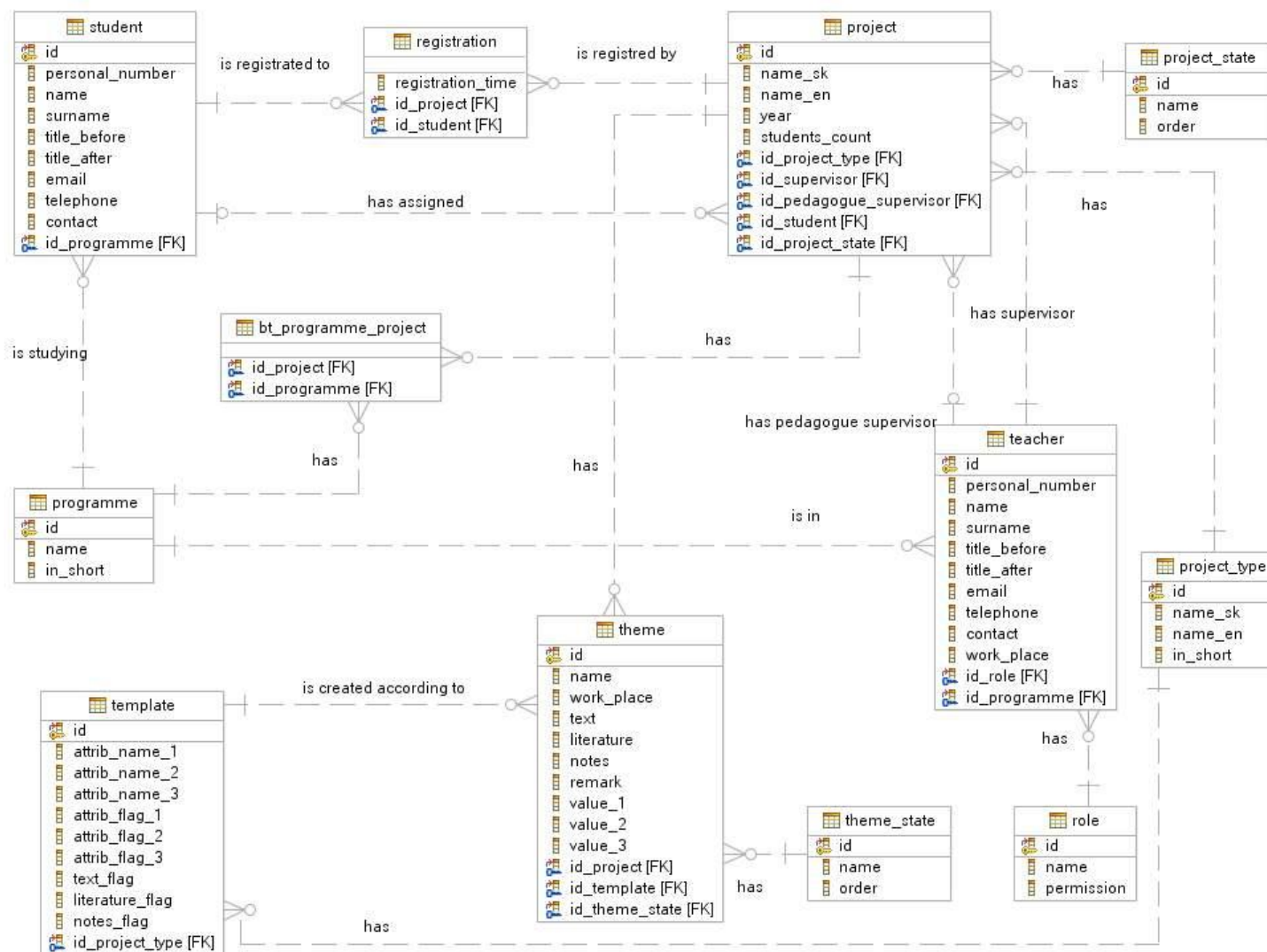
Cieľom pri návrhu nového dátového modelu bolo celkové zjednodušenie, odstránenie zbytočných entít, ktoré boli pôvodne vytvorené s ohľadom na univerzálnosť riešenia. Táto univerzálnosť sa ukázala byť neopodstatnená. Preto sme sa zamerali na minimalizáciu tabuliek. Okrem toho vznikli určité požiadavky, ktoré sme tiež zapracovali:

- každý projekt môže byť vypísaný pre viacero študijných programov (typickým príkladom je SI-IS)
- oddelenie študentov od zvyšných osôb, kvôli lepšej manipulácii s entitami
- uchovanie maximálneho počtu študentov, ktorým bude projekt priradený
- spresnenie atribútov niektorých entít

Na základe týchto nových požiadaviek vychádzajúc zo starého modelu sme navrhli nový dátový model, zobrazený na Obr. 3.3.



Obr. 3.2 Časť dátového modelu pôvodného systému YonBan.



Obr. 3.3 Časť nového dátového modelu systému.

### 3.2.1 Opis entít

- *student* – predstavuje študenta, ktorý si registruje projekty
- *registration* – reprezentuje samotnú registráciu študenta na zvolený projekt
- *project* – predstavuje projekt, ktorý vypíše vedúci
- *programme* – reprezentuje študijný program na fakulte
- *bt\_programme\_project* – väzobná tabuľka medzi projektom a študijným programom, ktorá zabezpečí, že projekt môže byť vypísaný pre viacero študijných programov
- *teacher* – predstavuje pedagóga, ktorý vypisuje projekty, ale môže mať pridelené aj iné úlohy, v závislosti od pridenej roly
- *project\_type* – reprezentuje typ projektu (bakalársky, diplomový, tímový)
- *template* – predstavuje šablónu zadania, ktorá obsahuje položky, ktoré zadanie má v závislosti od zvoleného typu projektu
- *theme* – reprezentuje konkrétne zadanie projektu
- *role* – prideluje pedagógom ďalšie oprávnenia
- *project\_state* – tabuľka enumerácií stavu projektu
- *theme\_state* – tabuľka enumerácií stavu zadania

### 3.2.2 Návrh tabuliek

<b>Názov entity:</b>	student		
<b>Popis entity:</b>	Zoznam študentov		
<b>Príbuzné entity:</b>	registration, project, programme		
Názov	Dátový typ	Popis	Vyžadované
id	int4	identifikačné číslo študenta	áno
id_programme	int4	cudzí kľúč študijného programu	áno
personal_number	varchar	osobné číslo študenta z AIS	áno
name	varchar	meno študenta	áno
surname	varchar	priezvisko študenta	áno
title_before	varchar	títul pred menom	nie
title_after	varchar	títul za menom	nie
email	varchar	e-mailová adresa študenta	nie
telephone	varchar	telefónne číslo študenta	nie
contact	varchar	iné kontaktné údaje študenta	nie

<b>Názov entity:</b>	registration		
<b>Popis entity:</b>	Zoznam registrácií študentov na projekty		
<b>Príbuzné entity:</b>	student, project		
Názov	Dátový typ	Popis	Vyžadované
id_student	int4	cudzí kľúč študenta	áno
id_project	int4	cudzí kľúč projektu	áno
registration_time	timestamp	čas registrácie študenta na projekt	áno

<b>Názov entity:</b>	project		
<b>Popis entity:</b>	Zoznam projektov		
<b>Príbuzné entity:</b>	registration, theme, student, bt_programme_project, teacher, project_type		
Názov	Dátový typ	Popis	Vyžadované

id	int4	identifikačné číslo projektu	áno
id_project_type	int4	cudzí kľúč typu projektu	áno
id_supervisor	int4	cudzí kľúč vedúceho projektu	áno
id_pedagogue_supervisor	int4	cudzí kľúč predagogického vedúceho	nie
id_student	int4	cudzí kľúč študenta	nie
name_sk	varchar	slovenský názov projektu	áno
name_en	varchar	anglický názov projektu	áno
year	varchar	rok riešenia projektu	áno
id_project_state	int2	cudzí kľúč stavu projektu	áno
students_count	int2	maximálny počet študentov, ktorým sa projekt prideli	nie

<b>Názov entity:</b>	programme		
<b>Popis entity:</b>	Zoznam študijných programov		
<b>Príbuzné entity:</b>	student, bt_programme_project, teacher		
<b>Názov</b>	<b>Dátový typ</b>	<b>Popis</b>	<b>Vyžadované</b>
id	int4	identifikačné číslo študijného programu	áno
name	varchar	názov študijného programu	áno
in_short	varchar	skratka študijného programu	nie

<b>Názov entity:</b>	bt_programme_project		
<b>Popis entity:</b>	Väzobná tabuľka medzi študijným programom a projektom		
<b>Príbuzné entity:</b>	programme, project		
<b>Názov</b>	<b>Dátový typ</b>	<b>Popis</b>	<b>Vyžadované</b>
id_project	int4	cudzí kľúč projektu	áno
in_programme	int4	cudzí kľúč študijného programu	áno

<b>Názov entity:</b>	teacher		
<b>Popis entity:</b>	Zoznam pedagógov		
<b>Príbuzné entity:</b>	role, project, programme		
<b>Názov</b>	<b>Dátový typ</b>	<b>Popis</b>	<b>Vyžadované</b>
id	int4	identifikačné číslo pedagóga	áno
id_programme	int4	cudzí kľúč študijného programu	áno
id_role	int4	cudzí kľúč roly	áno
personal_number	varchar	osobné číslo pedagóga z AIS	áno
name	varchar	meno pedagóga	áno
surname	varchar	priezvisko pedagóga	áno
title_before	varchar	titul pred menom	nie
title_after	varchar	titul za menom	nie
email	varchar	e-mailová adresa pedagóga	nie
telephone	varchar	telefónne číslo pedagóga	nie
contact	varchar	iné kontaktné údaje pedagóga	nie
work_place	varchar	pracovné miesto pedagóga	nie

<b>Názov entity:</b>	project_type		
<b>Popis entity:</b>	Zoznam typov projektov		
<b>Príbuzné entity:</b>	theme, project		
<b>Názov</b>	<b>Dátový typ</b>	<b>Popis</b>	<b>Vyžadované</b>
id	int4	identifikačné číslo	áno
name_sk	varchar	slovenský názov typu projektu	áno

name_en	varchar	anglický názov typu projektu	áno
in_short	varchar	skratka typu projektu	áno

<b>Názov entity:</b>	template		
<b>Popis entity:</b>	Zoznam šablón projektov		
<b>Príbuzné entity:</b>	theme, project_type		
Názov	Dátový typ	Popis	Vyžadované
id	int4	identifikačné číslo šablóny	áno
id_project_type	int4	cudzí kľúč typu projektu	áno
attrib_name_1	varchar	názov prvého atribútu	nie
attrib_name_2	varchar	názov druhého atribútu	nie
attrib_name_3	varchar	názov tretieho atribútu	nie
attrib_flag_1	bit	príznak, či sa má použiť prvý atribút	áno
attrib_flag_2	bit	príznak, či sa má použiť druhý atribút	áno
attrib_flag_3	bit	príznak, či sa má použiť tretí atribút	áno
text_flag	bit	príznak, či sa má použiť text zadania	áno
literature_flag	bit	príznak, či sa má použiť literatúra	áno
notes_flag	bit	príznak, či sa majú použiť poznámky	áno

<b>Názov entity:</b>	theme		
<b>Popis entity:</b>	Zoznam zadaní		
<b>Príbuzné entity:</b>	project, template		
Názov	Dátový typ	Popis	Vyžadované
id	int4	identifikačné číslo zadania	áno
id_project	int4	cudzí kľúč projektu	áno
id_template	int4	cudzí kľúč šablóny	áno
name	varchar	názov zadania	áno
id_theme_state	int2	cudzí kľúč stavu zadania	áno
work_place	varchar	miesto vypracovania	áno
text	text	text zadania	nie
literature	text	literatúra zadania	nie
notes	text	poznámky zadania	nie
value_1	varchar	hodnota prvého atribútu zo šablóny	nie
value_2	varchar	hodnota druhého atribútu zo šablóny	nie
value_3	varchar	hodnota tretieho atribútu zo šablóny	nie
remark	text	poznámka osoby, ktorá schvaľuje zadanie	nie

<b>Názov entity:</b>	role		
<b>Popis entity:</b>	Zoznam používateľských rolí v systéme		
<b>Príbuzné entity:</b>	teacher		
Názov	Dátový typ	Popis	Vyžadované
id	int4	identifikačné číslo roly	áno
name	varchar	názov roly	áno
permission	varchar	režazec identifikujúci rolu	áno

<b>Názov entity:</b>	project_state		
<b>Popis entity:</b>	Zoznam enumerácií stavu projektu		
<b>Príbuzné entity:</b>	project		
Názov	Dátový typ	Popis	Vyžadované
id	int4	identifikačné číslo stavu projektu	áno
name	varchar	názov stavu projektu	áno

order	int2	poradie zobrazenia stavu projektu v rozbaľovacom menu	áno
-------	------	--	-----

<b>Názov entity:</b>	theme_state		
<b>Popis entity:</b>	Zoznam enumerácií stavu zadania		
<b>Príbuzné entity:</b>	theme		
<b>Názov</b>	<b>Dátový typ</b>	<b>Popis</b>	<b>Vyžadované</b>
id	int4	identifikačné číslo stavu zadania	áno
name	varchar	názov stavu zadania	áno
order	int2	poradie zobrazenia stavu zadania v rozbaľovacom menu	áno