Univerzita Komenského v Bratislave

Fakulta matematiky, fyziky a informatiky

Návrh systému

Kurzy

Andrea Spišáková

Patrícia Marmanová

Dana Škorvánková

Michal Brcko November 2017

Obsah

1 Úvod 3

1.1 Cieľ 3

1.2 Rozsah 3

1.3 Definície, akronymy a skratky 3

1.4 Prehľad zvyšku dokumentu 3

2 Popis dekompozície 4

2.1 Dekompozícia modulov 5

2.1.1 Kontroléry 5

2.2 Dekompozícia dát 6

2.2.1 Databázový model 6

3 Popis závislostí 8

3.1 Technologické závislosti 8

3.2 Prístup do webovej aplikácie 8

4 Popis rozhraní 9

4.1 Funkčné rozhrania. 9

4.1.1 Metódy podľa kontrolérov 9

4.1.2 Triedny diagram 11

5 Návrh použivateľského rozhrania 12

5.1 Prihlásenie 12

5.2 Vylistovanie kurzov 13

5.3 Prehľad kurzu 15

5.4 Prehľad kurzov používateľa 16

5.5 Prehľad kurzov školiteľa 17

5.6 Vytvorenie nového termínu kurzu garantom 18

5.7 Vytvorenie nového kurzu subadminom 18

5.8 Priradenie subadmina súčasti 19

# 1 Úvod

## 1.1 Cieľ

Tento dokument popisuje návrh systému webovej aplikácie Kurzy, ktorá bude slúžiť Centru informačných technológií na správu školení a používateľom na prihlasovanie na jednotlivé školenia. Cieľom vývoja opísanej aplikácie je nahradiť a rozšíriť aktuálne existujúci univerzitný softvér.

## 

## 1.2 Rozsah

Aplikácia bude slúžiť pre správu školení organizovaných univerzitnými pracoviskami. Systém má za úlohu umožniť používateľom vytvárať jednotlivé kurzy a prihlasovať sa na ne. Implementovaných bude 5 rôznych používateľských rolí, ktoré popisujú práva používateľov, od základného návštevníka, až po administrátora. Po prihlásení do systému môžu všetci používatelia filtrovať a prezerať kurzy. V prípade, že je naplnená kapacita kurzu, systém poskytuje používateľovi možnosť hlásiť sa na kurz ako náhradník, resp. prejaviť o kurz záujem. To znamená, že používateľ bude notifikovaný v prípade, že by sa miesto uvoľnilo, alebo bola navýšená maximálna kapacita kurzu.

## 1.3 Definície, akronymy a skratky

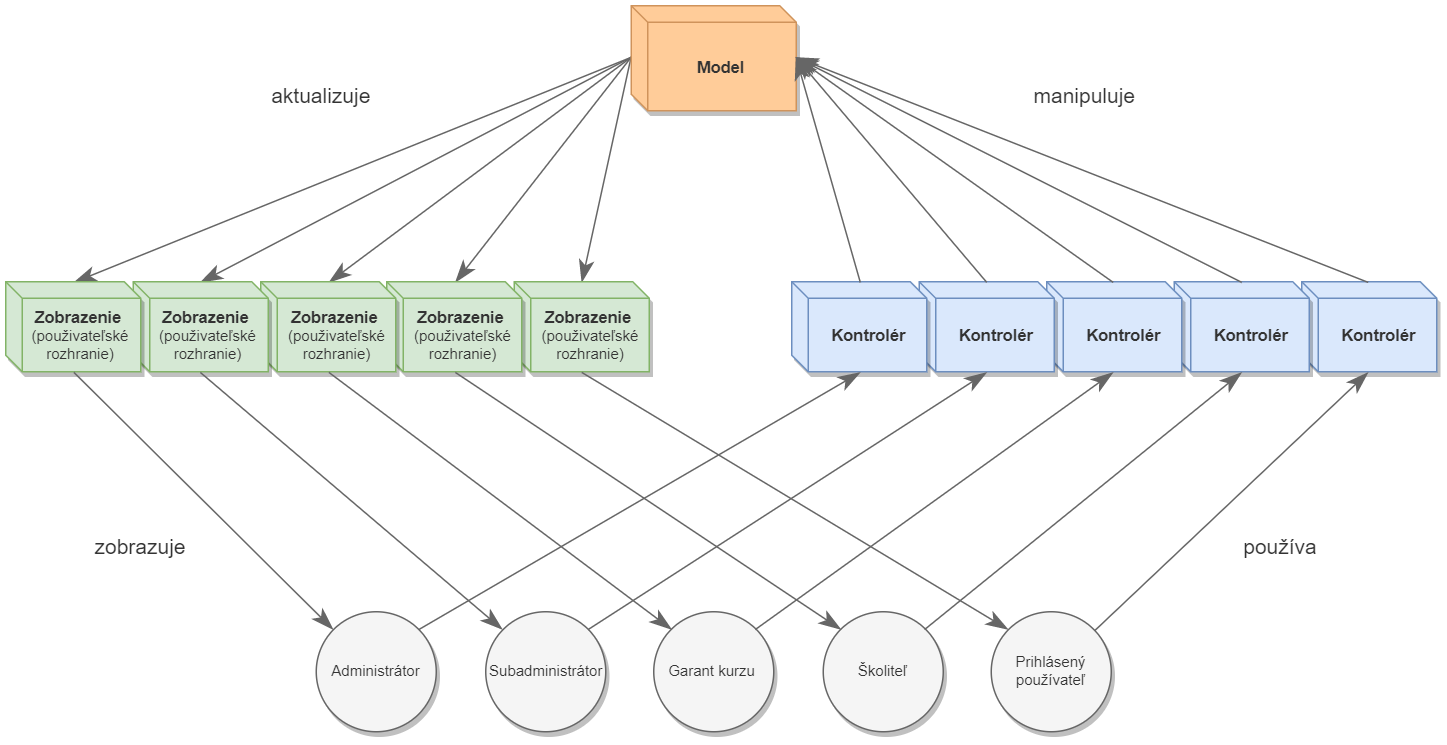
|  |  |
| --- | --- |
| **CIT** | Centrum informačných technológií |
| **súčasť** | univerzita sa delí na menšie celky (katedry, fakulty) |
| **prerekvizita** | povinnosť, ktorá musí byť splnená pred absolvovaním kurzu |
| **id** | identifikácia konkrétneho používateľa |
| **PHP** | PHP: Hypertext Preprocessor (programovací jazyk používaný najmä pre vývoj webových aplikácií) |
| **CSS3** | Cascading Style Sheet (technológia na jednotné štýlovanie obsahu HTML dokumentu) |
| **HTML5** | Hyper Text Markup Language (značkový jazyk určený na tvorbu webových dokumentov a iných informácií zobraziteľných vo webovom prehliadači) |
| **MySQL** | My Structured Query Language (relačný databázový server) |
| **Symfony** | framework pre jazyk PHP |
| **framework** | softvérová štruktúra, ktorá slúži ako podpora pri programovaní |

## 1.4 Prehľad zvyšku dokumentu

V druhej sekcii sa nachádzajú referencie. V tretej sekcii je obsiahnutý popis dekompozície, teda dekompozícia modulov, konkurentných procesov a dát. Štvrtá sekcia obsahuje popis závislostí systému. Piata sekcia popisuje rozhranie modulov a procesov. V šiestej sekcii je načrtnutý detailný dizajn aplikácie.

# 2 Popis dekompozície

Aplikácia implementuje tzv. *Model-view-controller* softwarovú architektúru, ktorá rozdeľuje dátový model aplikácie, používateľské rozhranie a riadiacu logiku do troch nezávislých komponentov tak, že modifikácia niektorého z nich má len minimálny vplyv na ostatné.



*Obrázok 1 - Mode-view-controller softwarová* architektúra

## 2.1 Dekompozícia modulov

Aplikácia bude rozdelená na niekoľko modulov. Dve základné časti budú administrácia kurzov a prihlasovanie na kurzy.

### 2.1.1 Kontroléry

V závislosti od používateľských rolí bude systém implementovať 5 rôznych kontrolérov (ktoré budú sprostredkúvať prenos externých udalostí zvonka do vnútra systému a naopak).

**Prihlásený používateľ**

Koncový používateľ, ktorý si môže prezerať a filtrovať všetky kurzy. Môže prejaviť záujem o absolvovanie niektorého kurzu v budúcnosti. Prihlasuje sa na vypísané kurzy, alebo sa odhlasuje z kurzov, na ktoré sa zapísal. Môže kontaktovať školiteľa prostredníctvom e-mailu. Nemá možnosť vidieť žiadne informácie o ostatných koncových používateľoch a ani nemôže meniť informácie o kurzoch, ktoré už absolvoval.

**Školiteľ**

Používateľ, ktorý je zodpovedný za termín kurzu, ktorý mu bol pridelený garantom kurzu a bude ho viesť. Môže prihlasovať a odhlasovať používateľov na kurz, alebo ich kontaktovať. Tiež môže tlačiť automaticky generované dokumenty (zoznam prihlásených na kurz a certifikáty o absolvovaní kurzu) vo formáte pdf.

**Garant kurzu**

Používateľ, ktorý je zodpovedný za jemu pridelený kurz, prípadne viac kurzov, subadministrátorom. Jeho úlohou je vypisovať termíny školení a prideľovať im jednotlivých školiteľov (má možnosť priradiť ako školiteľa aj samého seba) a vypisovať základné informácie o kurze – anotácia, požiadavky, prerekvizity.

**Subadministrátor**

Používateľ, ktorého právomoci sa pohybujú len v rámci jednej súčasti a nemôže zasahovať do správy kurzov a používateľov, ktorí patria do inej súčasti. Jeho úlohou je určovať garantov kurzov, vytvárať nové kurzy a upravovať už existujúce kurzy.

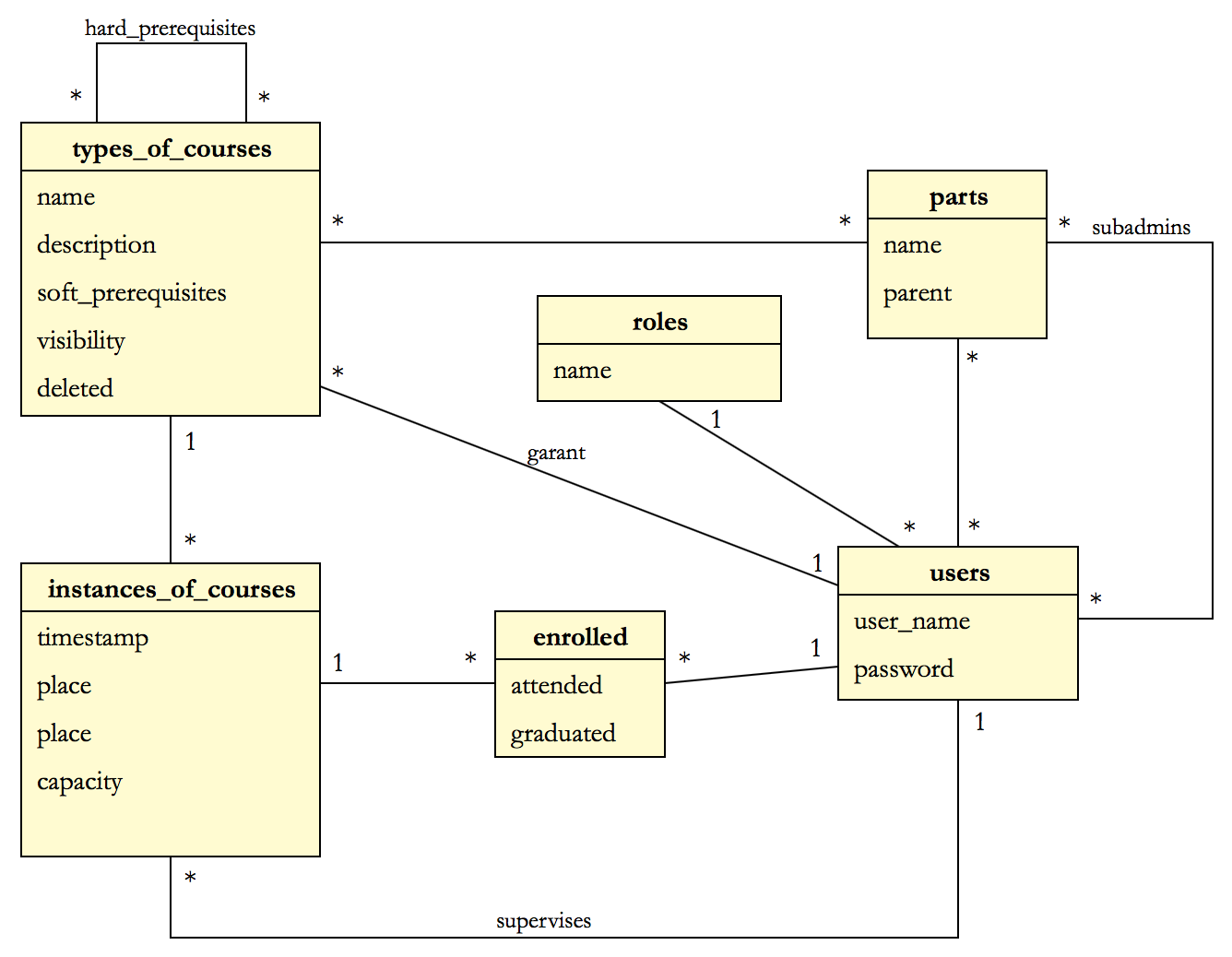
**Administrátor**

Hierarchicky najvyššia používateľská rola, ktorá má na starosti chod celého systému. Má všetky právomoci na správu používateľov a kurzov. Jeho úlohou je určovať subadministrátorov, dohliadať na aktuálnosť a správnosť informácií a pomáhať riešiť prípadné problémy.

## 

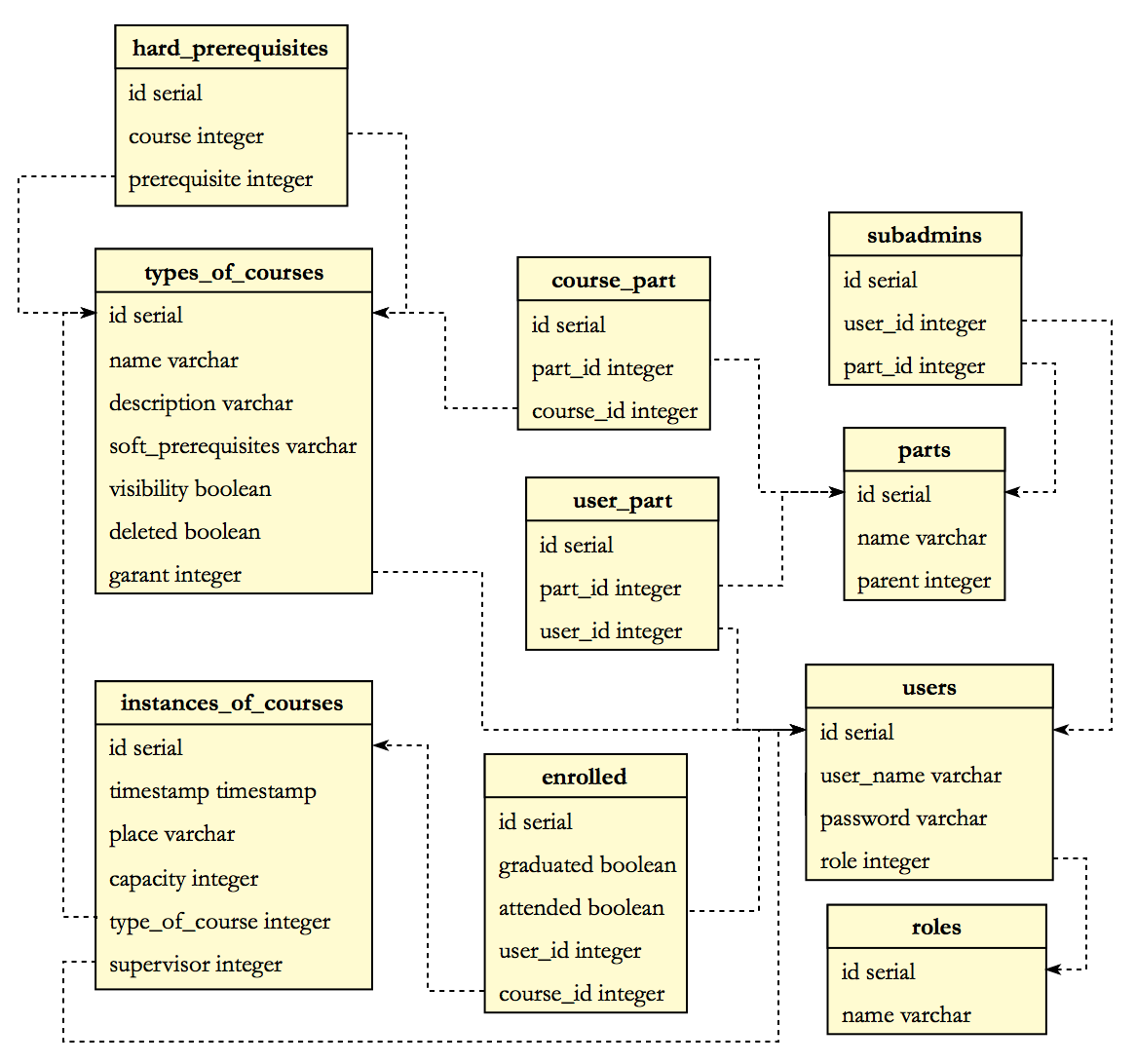
## 2.2 Dekompozícia dát

## 2.2.1 Databázový model



Obrázok 2 - Entitno-relačný model databázy

Obrázok č. 2 zobrazuje entitno-relačný model databázy pre aplikáciu Kurzy. Kľúčovými sú tabuľka „users“ (používatelia), „types\_of\_courses“ (typy kurzov) a „instances\_of\_courses“ (konkrétne inštancie kurzov, respekt. termíny kurzov). Každý typ kurzu má prideleného garanta, môže obsahovať prerekvizity, ktorými sú iné typy kurzov a patrí jednej alebo viacerým súčastiam. Súčasti sú v tabuľke „parts“ a každá súčasť má prideleného jedného alebo viac subadministrátorov. Každá konkrétna inštancia kurzu má práve jedného školiteľa a môže sa na ňu prihlásiť toľko používateľov, aká je jej kapacita. Používatelia majú v systéme rozličné roly. Tieto roly sú v tabuľke „roles“. Každý používateľ tiež spadá pod súčasť. Tabuľka „enrolled“ uchováva informácie o tom, či sa používatelia prihlásení na konkrétne inštancie kurzov aj reálne zúčastnili kurzov a úspešne ich absolvovali.



Obrázok 3 - Relačný model databázy

Obrázok 3 predstavuje relačný model databázy, ktorý vznikol transformáciou entitno-relačného modelu. Zobrazuje ako sú dáta uchovávané v jednotlivých tabuľkách.

# 3 Popis závislostí

## 3.1 Technologické závislosti

Aplikácia je vyvíjaná v programovacích jazykoch PHP, HTML5 a CSS3. Bude bežať na serveri s operačným systémom Debian a využívať relačný databázový server MySQL.

Pri vývoji sa budú tiež využívať:

**Symfony** – framework pre vývoj webových aplikácií v jazyku PHP,

**Twig** – šablónovací systém, ktorý umožňuje oddeliť aplikačnú vrstvu od prezentačnej vrstvy,

**Doctrine** – framework pre jazyk PHP, ktorý umožňuje abstrahovať databázovú vrstvu a pracovať s dátami ako s objektmi.

## 3.2 Prístup do webovej aplikácie

Prístup do tejto webovej aplikácie je možný len po prihlásení sa do Integrovaného informačného a komunikačného systému (IIKS) Univerzity Komenského.

# 4 Popis rozhraní

## 4.1 Funkčné rozhrania

V tejto sekcii dokumentu je popísaná funkcionalita jednotlivých kontrolérov (ich metódy a vstupné a výstupné parametre týchto metód). Obsahuje tiež triedny diagram detailne zobrazujúci atribúty, metódy a vzťahy medzi jednotlivými triedami.  
  
  
4.1.1 Metódy podľa kontrolérov

**Bežný používateľ (trieda User)**

|  |  |
| --- | --- |
| Názov metódy | Funkcionalita |
| enrollInCourse(Course course): boolean | prihlásenie na konkrétny termín kurzu  vracia úspešnosť vykonanej akcie |
| disenrollFromCourse(Course course): boolean | odhlásenie z termínu kurzu  vracia úspešnosť vykonanej akcie |
| listMyCourses(): ArrayList<Enrollment> | vylistovanie všetkých zapísaných a absolvovaných kurzov  vracia pole objektov triedy Enrollment (obsahuje referenciu na kurz, používateľa a informáciu o absolvovaní resp. neabsolvovaní kurzu) |
| listAvailableCourses(): ArrayList<Course> | vylistovanie všetkých dostupných kurzov, na ktoré sa daný používateľ môže prihlásiť  vracia pole objektov triedy Course |
| sendEmail(Supervisor s, String message): void | kontaktovanie školiteľa prostredníctvom e-mailu |

|  |  |
| --- | --- |
| Názov metódy | Funkcionalita |
| supervisedCourses(): ArrayList<Course> | zobrazenie všetkých kurzov, ktoré daný školiteľ vedie  vracia pole objektov triedy Course |
| enrolledUsers(Course c): ArrayList<User> | zobrazenie všetkých používateľov prihlásených na kurz  vracia pole objektov triedy User |
| printCertificate(User u, Course c): void | tlač certifikátu o absolvovaní kurzu používateľom |
| setAttendance(User u, Course c, boolean attended): void | potvrdenie účasti resp. neúčasti používateľa na kurze |
| enrollUser(User u, Course c): void | zápis používateľa na kurz (dá sa využiť napr. ak má používateľ splnené prerekvizity externe) |

**Školiteľ (trieda Supervisor)**

**Garant kurzu (trieda Master)**

|  |  |
| --- | --- |
| Názov metódy | Funkcionalita |
| listCoursesByType(CourseType coursetype): ArrayList<Course> | zobrazenie všetkých termínov daného (typu) kurzu  vracia pole objektov triedy Course |
| setSupervisor(Supervisor supervisor, Course course): void | priradenie školiteľa k termínu kurzu |
| setVisibility(CourseType coursetype, boolean visible): void | nastavenie viditeľnosti kurzu |
| createCourse(Timestamp t, String place, int capacity): Course | vytvorenie termínu kurzu (parametre – dátum, čas, miesto, kapacita)  vracia vytvorený objekt triedy Course |
| deleteCourse(Course course): void | zrušenie termínu kurzu |

**Subadministrátor (trieda Subadmin)**

|  |  |
| --- | --- |
| Názov metódy | Funkcionalita |
| createCourseType(String name, String desc, String prereqs, boolean visible): CourseType | vytvorenie nového kurzu (parametre – názov, popis, prerekvizity, viditeľnosť)  vracia objekt vytvoreného kurzu |
| editCourse(CourseType ctype, String name, String desc, String prereqs, boolean visible): void | upravenie existujúceho kurzu |
| deleteCourseType(CourseType ctype): void | zrušenie kurzu |
| setMaster(Master master, CourseType ctype): void | priradenie garanta ku kurzu |

**Administrátor (trieda Admin)**

|  |  |
| --- | --- |
| Názov metódy | Funkcionalita |
| setSubadmin(Department d, Subadmin subadmin): void | priradenie subadministrátora k súčasti |
| listCourses(Department d): ArrayList<Course> | zobrazenie termínov všetkých kurzov |

### 

### 4.1.2 Triedny diagram

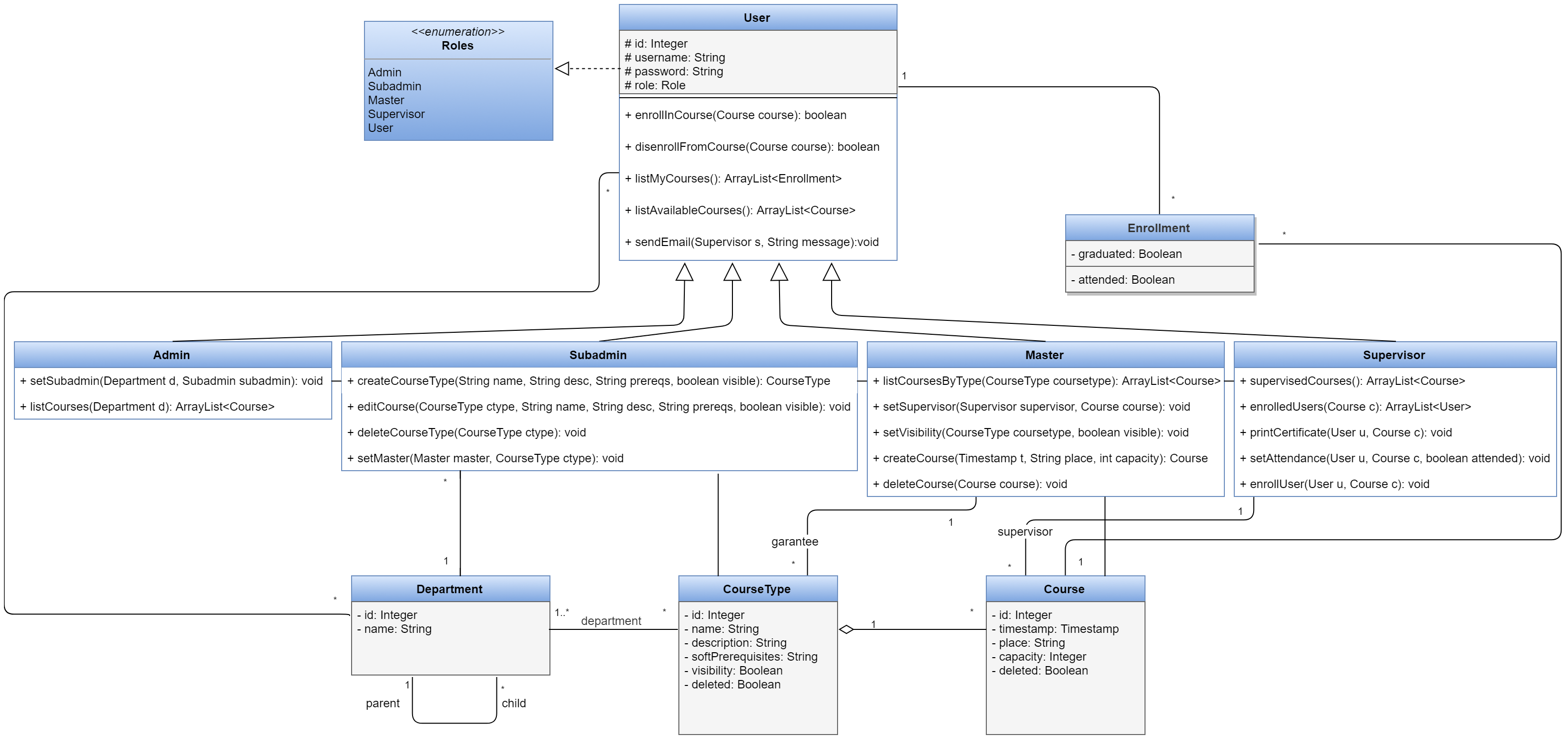
Triedny diagram zobrazuje atribúty, metódy tried a vzťahy medzi triedami. Nadtrieda User pokrýva základnú funkcionalitu prihláseného používateľa. Táto trieda má štyri podtriedy (Admin, Subadmin, Master, Supervisor), ktoré od nej dedia atribúty a metódy a každá pridáva nejakú vlastnú špecifickú funkcionalitu v závislosti od právomocí konkrétnej používateľskej role.

Trieda CourseType predstavuje typ kurzu, zatiaľ čo trieda Course predstavuje konkrétny termín nejakého typu kurzu. Medzi triedami Course a CourseType je vzťah agregácie (pričom CourseType je celok a Course je časť). Jeden typ kurzu môže mať ľubovoľne veľa termínov, ale jeden termín kurzu patrí vždy práve k jednému typu kurzu.

Trieda Department predstavuje súčasť univerzity. Súčasti sú hierarchicky usporiadané – každá súčasť má atribút rodič (zrejme koreňová súčasť – samotná univerzita – tam bude mať null), kde je referencia na nadradenú súčasť a atribút deti, kde sú referencie na podradené súčasti.

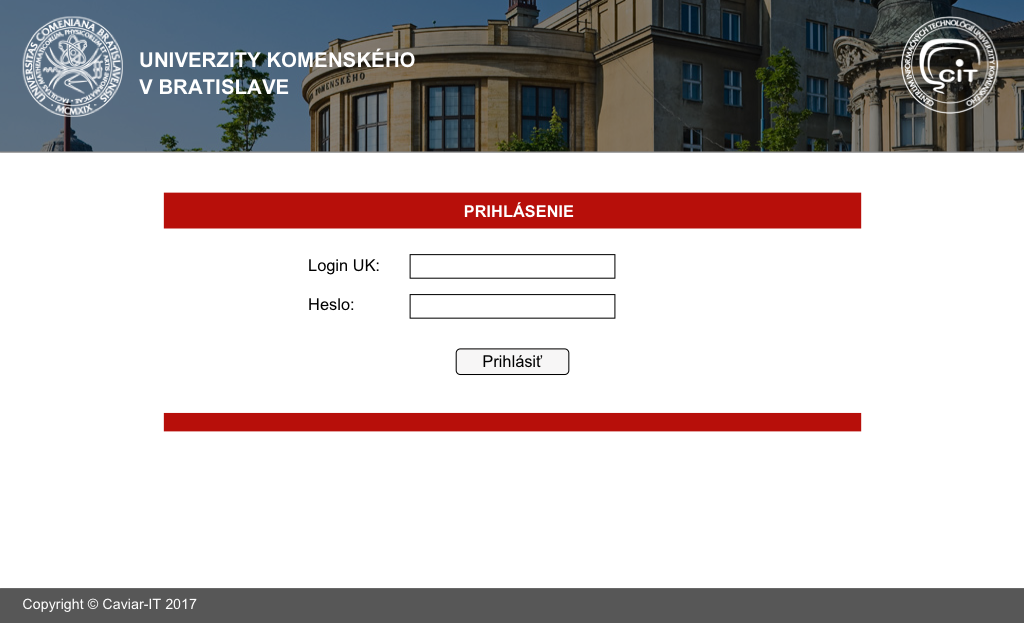
Trieda Enrollment je pomocná trieda, ktorá obsahuje atribút s referenciou na používateľa a atribút s referenciou na termín kurzu. K týmto referenciám pridáva ešte vlastné atribúty graduated a attended, ktoré poskytujú informácie o tom, či používateľ kurz úspešne absolvoval a či sa zúčastnil.

Obrázok 4 - Triedny diagram



# 5 Návrh použivateľského rozhrania

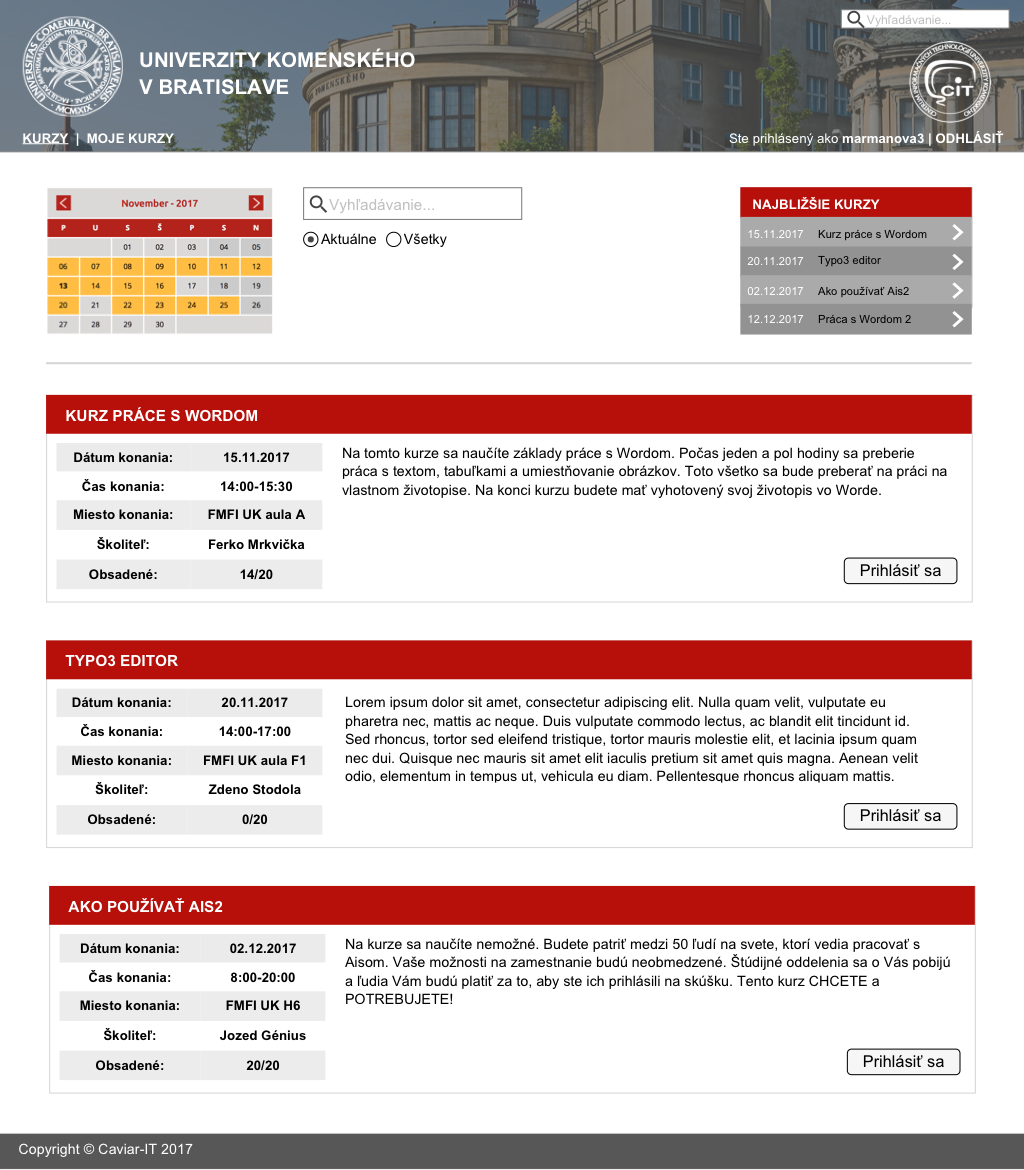
## 5.1 Prihlásenie



Obrázok 5 - Návrh rozhrania pre prihlásenie

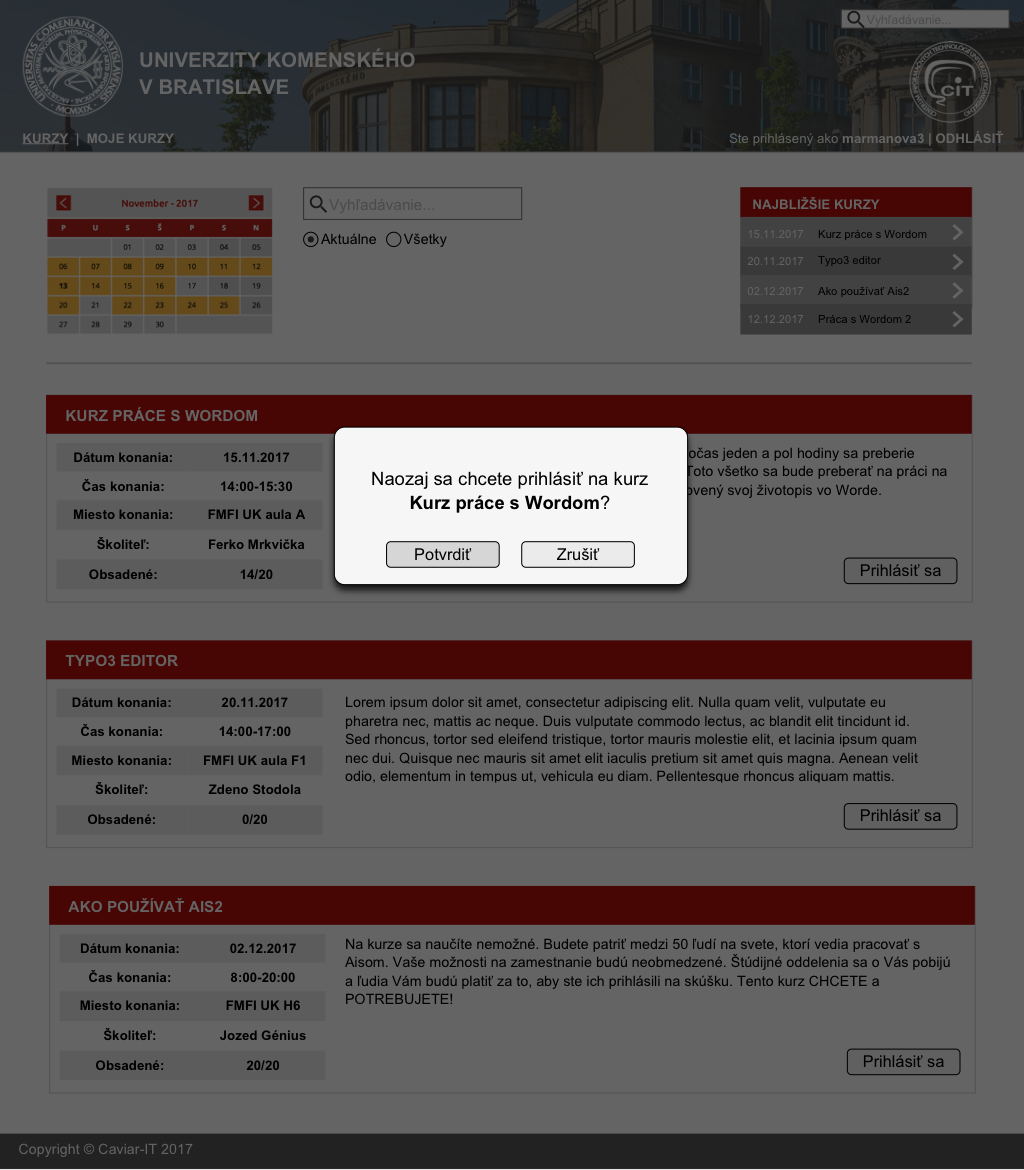
Na akúkoľvek prácu so systémom je vyžadované prihlásenie. Bez prihlásenia nie je možné ani prezerať kurzy. Návrh rozhrania pre prihlásenie do systému je na obrázku 5.

## 5.2 Vylistovanie kurzov



Obrázok 6.1 - Návrh rozhrania pre vylistovanie kurzov

Základnou funkcionalitou pre každého používateľa je prezeranie kurzov. Systém umožňuje kurzy vylistovať. Používateľ je na túto stránku presmerovaný hneď po prihlásení, alebo po kliknutí na "KURZY" v navigácii. Návrh rozhrania pre prehľad kurzov je na obrázku 6.1. Pod navigáciou sa nachádzajú možnosti filtrovania kurzov. Kurzy sa dajú filtrovať podľa dátumu (viď kalendár), podľa toho či chce používateľ zobraziť aktuálne vyhlásené termíny kurzov, alebo všetky druhy kurzov a podľa vyhľadávania konkrétneho výrazu ako napr. celý alebo časť názvu alebo meno školiteľa. Kurzy sú ďalej vypísané podľa filtru. Pri každom kurze je jeho unikátny názov, sylabus a základné údaje o dátume a čase konania, mieste konania, školiteľovi a obsadenosti kurzu. Na každý aktuálne vyhlásený kurz sa môže používateľ prihlásiť pomocou tlačidla "Prihlásiť sa".



Obrázok 6.2 - Návrh rozhrania pre dvojkrokové prihlásenie

Po kliknutí na tlačidlo "Prihlásiť sa" vyskočí používateľovi okno s potvrdením prihlásenia. Požívateľ má možnosť buď potvrdiť prihlásenie alebo zrušiť. Návrh tohto rozhrania je na obrázku 6.2.

## 5.3 Prehľad kurzu



Obrázok 7 - Návrh rozhrania pre prehľad kurzu

Po kliknutí na ľubovoľný kurz v prehľade kurzov sa otvorí prehľad daného kurzu, viď obr. 7. V ňom sú zobrazené všeobecné informácie o kurze - názov, sylabus a prerekvizity - ako tabuľka konkrétnych inštancií kurzu, ktorá obsahuje dátum a čas konania, miesto, školiteľa a obsadenosť. Každý konkrétny termín školenia má tlačidlo "Prihlásiť", ktoré spustí dvojkrokové prihlásenie na daný termín s vyskakovacím oknom na potvrdenie.

## 5.4 Prehľad kurzov používateľa

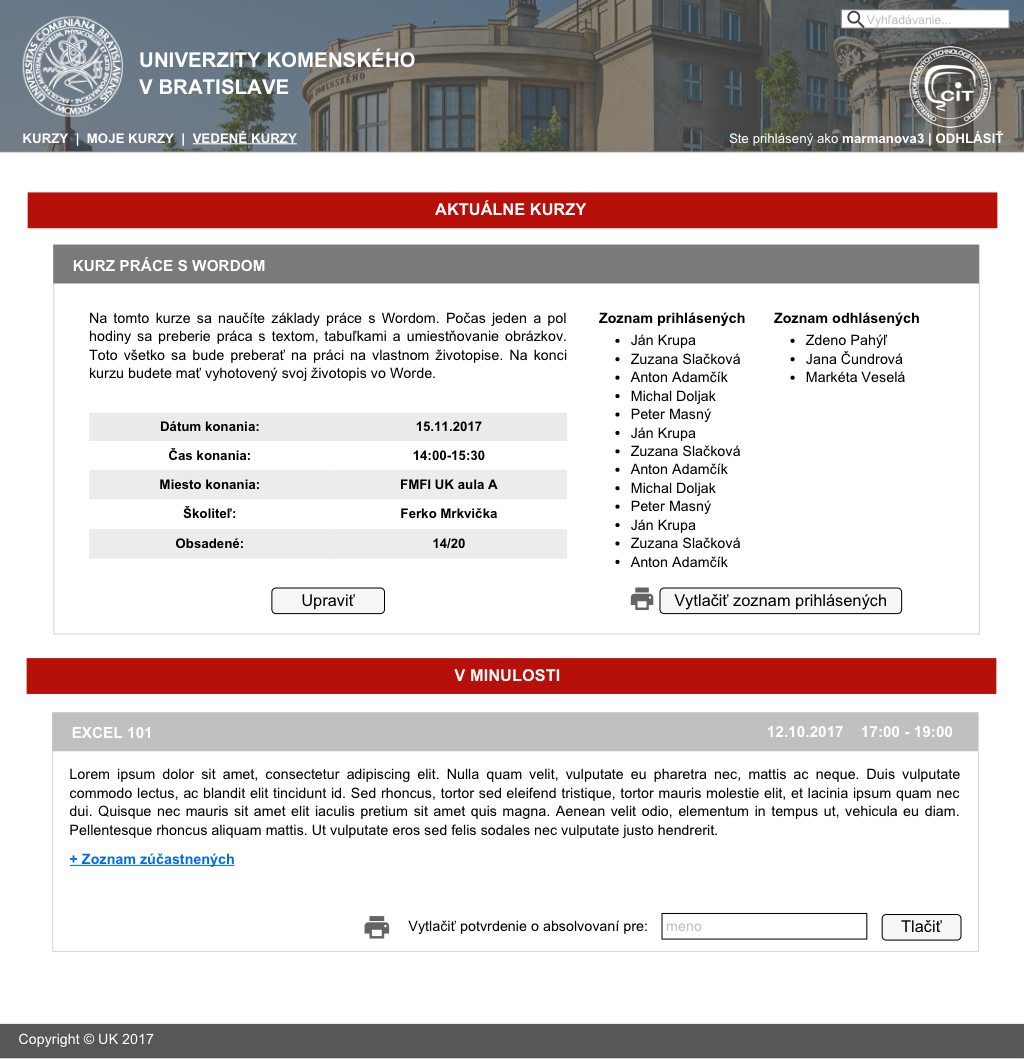


Obrázok 8 - Návrh rozhrania pre prehľad kurzov používateľa

Návrh rozhranie pre prehľad kurzov používateľa je na obrázku 8. Každý používateľ má právo prezerať kurzy ktoré absolvoval, pričom tieto kurzy nemôže meniť, a takisto prezerať kurzy, na ktoré je prihlásený. Ak ešte neuplynul termín, do ktorého je možné sa odhlásiť, môže sa používateľ z daného kurzu odhlásiť tlačidlo "Odhlásiť sa".

Prehľad kurzu má k dispozícii rovnako ako pri vylistovaní všetkých kurzov a kliknutím na konkrétny kurz sa dostane k jeho prehľadu.

## 5.5 Prehľad kurzov školiteľa



Obrázok 9.1 - Návrh rozhrania pre prehľad kurzov školiteľa

Návrh rozhrania pre školiteľa na obrázku 9.1. Každý používateľ, ktorý sa stal školiteľom tým, že ho garant priradil na nejaký kurz ako školiteľa, má k dispozícii prehľad kurzov, ktoré školil v minulosti alebo školí aktuálne. K tomuto prehľadu sa môže dostať cez navigáciu kliknutím na "VEDENÉ KURZY". Tu si môže prezerať a upravovať aktuálne školené kurzy a tlačiť zoznam prihlásených vyexportovaných do PDF. Takisto môže vytlačiť potvrdenie pre účastníkov, ktorí absolvovali jeho kurzy v minulosti. Po kliknutí na "+ Zoznam zúčastnených" sa k danému kurzu rozrolluje prehľad všetkých účastníkov kurzu, kde môže školiteľ následne editovať, že či sa daného kurzu zúčastnili a či ho úspešne absolvovali - obrázok 9.2.

../../../../../../../var/folders/v0/3x_6jxrn2d98ckrgp5fgy09h0000gn/T/com.apple.Preview/com.apple.Preview.PasteboardItems

Obrázok 9.2 - Návrh rozhrania pre rozrollovanie kurzu pri prehľade kurzov školiteľa

## 5.6 Vytvorenie nového termínu kurzu garantom



Obrázok 10 - Návrh rozhrania pre vytvorenie nového temínu kurzu garantom

Na obrázku 10 je zobrazený návrh rozhrania pre vytvorenie novej inštancie (termínu) kurzu garantom, ktorému bol daný kurz pridelený.

Garant sa sem dostane cez navigáciu kliknutím na "NOVÝ TERMÍN KURZU", a môže vyberať druh kurzu z kurzov, ktoré mu boli pridelené subadminom. Následne vyplní dátum, čas, miesto, kapaitu a priradí školiteľa pomocou univerzitného loginu.

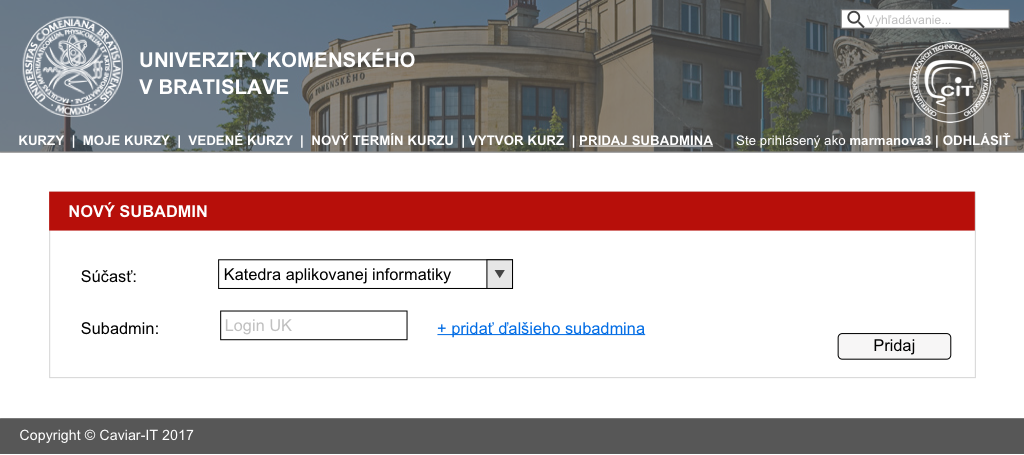
## 5.7 Vytvorenie nového kurzu subadminom



Obrázok 12 - Návrh rozhrania pre vytvorenie nového kurzu subadminom

Používateľ, ktorému boli adminom pridelené práva subadmina k nejakej súčasti má právo vytvárať nové kurzy, ku ktorým môže priraďovať garanta. Na obrázku 12 môžeme vidieť návrh tohto rozhrania. Subadmin vyplní názov kurzu, sylabus vypíše soft prerekvizity a vyberie kurzy, ktoré sú hard prerekvizitami daného kurzu. Stlačením "+ pridať prerekvizitu" môže pridávať ďalšie. Napokon priradí garanta na základe univerzitného loginu.

## 5.8 Priradenie subadmina súčasti



Obrázok 13 - Návrh rozhrania pre admina na pridávanie subadminov k súčasti

Admin má právo pridávať subadminov k jednotlivým súčastiam. To môže urobiť kliknutím na "PRIDAJ SUBADMINA" v navigácii. Ďalej zvolí súčasť a vyplní jedného alebo viac subadminov. Návrh tohto rozhrania je na obrázku 13.