

Katedra informatiky
Přírodovědecká fakulta
Univerzita Palackého v Olomouci

BAKALÁŘSKÁ PRÁCE

Tvorba rozvrhu



2017

Vedoucí práce: Mgr. Tomáš Kühr,
Ph.D.

Kristýna Hanková

Studijní obor: Aplikovaná informatika,
prezenční forma

Bibliografické údaje

Autor: Kristýna Hanková
Název práce: Tvorba rozvrhu
Typ práce: bakalářská práce
Pracoviště: Katedra informatiky, Přírodovědecká fakulta, Univerzita Palackého v Olomouci
Rok obhajoby: 2017
Studijní obor: Aplikovaná informatika, prezenční forma
Vedoucí práce: Mgr. Tomáš Kühn, Ph.D.
Počet stran: 42
Přílohy: 1 CD/DVD
Jazyk práce: český

Bibliographic info

Author: Kristýna Hanková
Title: Timetabling
Thesis type: bachelor thesis
Department: Department of Computer Science, Faculty of Science, Palacký University Olomouc
Year of defense: 2017
Study field: Applied Computer Science, full-time form
Supervisor: Mgr. Tomáš Kühn, Ph.D.
Page count: 42
Supplements: 1 CD/DVD
Thesis language: Czech

Anotace

K této práci byl vytvořen interaktivní softwarový nástroj pro usnadnění tvorby rozvrhů, vázaný především k univerzitnímu prostředí. Aplikace je určena pro osobu, která bude na univerzitě, konkrétně na určité katedře, rozvrh vytvářet, a měla by uživateli svou možností vizualizace tuto práci usnadnit. Při realizaci aplikace jsem použila programovací jazyky C# a SQL.

Synopsis

As a part of this thesis, an interactive software tool was developed to help with creation of schedules in the university environment. This software is intended to be of help for employees who are in charge of assembling time schedules for specific departments and is supposed to facilitate this task via its visualization options. To realize the application the programming languages C# a SQL were used.

Klíčová slova: tvorba rozvrhu; C#; SQL; Windows Forms Application

Keywords: timetabling; C#; SQL; Windows Forms Application

Poděkovat bych chtěla panu Mgr. Tomáši Kührovi, Ph.D., který vedl mou bakalářskou práci a byl vždy ochoten najít si čas, poradit a pomoci.

Místopřísežně prohlašuji, že jsem celou práci včetně příloh vypracoval/a samostatně a za použití pouze zdrojů citovaných v textu práce a uvedených v seznamu literatury.

datum odevzdání práce

podpis autora

Obsah

1 Úvod	7
1.1 Problematika rozvrhování a plánování	7
1.2 Úvod k aplikaci	7
1.3 Požadavky	8
2 Programátorská dokumentace	9
2.1 Použité technologické prostředky	9
2.1.1 Programovací jazyk C#	9
2.1.2 Visual Studio	9
2.1.3 .NET	9
2.1.4 Microsoft Access	9
2.1.5 SQL	10
2.2 Diagram užití	10
2.3 Databáze	11
2.4 Program	11
2.4.1 Timetabling	11
2.4.2 Classes	12
2.4.3 Tables	21
2.4.4 Administration	22
2.4.5 Forms	22
3 Uživatelská dokumentace	25
3.1 Systémové požadavky	25
3.2 Instalace aplikace	25
3.3 Spuštění aplikace	25
3.4 Základní nabídka funkcí	26
3.5 Tvorba rozvrhu	27
3.6 Úprava databáze	34
Závěr	39
Conclusions	40
Literatura	41
A Obsah přiloženého CD/DVD	42

Seznam obrázků

1	Diagram užití	10
2	Databáze - relace	11
3	Úvodní okno	25
4	Tvorba rozvrhu	26
5	Úprava databáze	27
6	Akce - okno učené pro správu akcí v rozvrzích	28
7	Akce - příklad	29
8	Akce - příklad	29
9	Akce - Omezení	30
10	Prohlížení	31
11	Prohlížení - příklad samostatného zobrazení	32
12	Prohlížení - příklad spojení tabulek	32
13	Vyhledávání	33
14	Vyhledávání - příklad	34
15	Vyhledávání - příklad, kliknutí	34
16	Editace - budova, fakulta, katedra	35
17	Editace - typ místnosti, místnost	35
18	Editace - forma studia, typ studia	36
19	Editace - třída	36
20	Editace - studijní program, obor, předmět	37
21	Editace - vyučující	37

1 Úvod

1.1 Problematika rozvrhování a plánování

Problematika rozvrhování a plánování (více informací v knihách [1], [2]) se v případě tvorby rozvrhů zabývá tím, jak nejefektivněji přiřadit rozvrhové akci časový prostor.

Sestavování rozvrhů se nevztahuje jenom na školství, ale tento problém můžeme nalézt i v jiných oblastech jako je tvorba plánů směn pro zaměstnance nebo výrobních plánů a podobně.

Naplánované akce pak musí dbát různých kritérií, aby později nedocházelo k nežádoucím kolizím. Tyto podmínky budou různé v závislosti na požadovaném výsledku. Obecně je třeba dbát na to, jaká omezení mají zdroje, se kterými pracujeme, a jaké jsou mezi nimi vazby.

1.2 Úvod k aplikaci

Cílem této bakalářské práce bylo vytvořit aplikaci určenou pro tvorbu a následnou úpravu rozvrhů, která má za cíl usnadnit práci osobě, jež bude rozvrhy na univerzitní půdě vytvářet. Jelikož je tvorba rozvrhů na univerzitách obecně nedílnou a podstatnou součástí pro správné fungování nejen jednotlivých kateder, ale i celých fakult, tak právě na vytváření rozvrhů je kladen vysoký nárok, při kterém je nutné skloubit několik faktorů.

Při samotné tvorbě je potřeba ohlídat, aby nedošlo hned k několika kolizím. Mezi hlavní z nich patří:

1. vyučující může vyučovat v daný den a hodinu pouze jeden předmět (nemůže být na dvou místech zároveň a vyučovat v jednu chvíli více předmětů),
2. v učebně může v daný den a hodinu probíhat jen jedna akce.

Co se týče oboru a ročníku (na tuto kombinaci bychom mohli nahlížet jako na obdobu "třídy" na SŠ), předmětů v jeden den a čas může mít více, neboť se pohybujeme v univerzitním prostředí a studenti si mohou rozvrh v rámci možností navolit sami. V mnoha případech je dokonce nutné vypsát více akcí pro totožný předmět a typ akce, neboť vzhledem ke kapacitě místnosti a celkovému počtu studentů v daném oboru a ročníku (především 1. ročníků) by se spousta studentů nemohla na předmět zapsat. Např. cvičení k jednomu předmětu může být vypsáno více, než jaká je dotace, a student si nakonec zvolí jen jednu z několika dostupných variant. Při zobrazení rozvrhu pro konkrétní obor a ročník tedy běžně může dojít k tomu, že se akce budou překrývat.

Pokud bychom ale uvažovali předmět, který bude mít v dotaci přednášku a cvičení, a vypíšeme jenom jedno cvičení a to ještě ve stejnou dobu jako bude probíhat přednáška tohoto předmětu, nebude to zrovna vhodná varianta i když se akce překrývat můžou. Na tuto situaci by měl uživatel dávat pozor.

Každá akce nese také informaci o datovém intervalu, ve kterém bude probíhat (počáteční a koncové datum). Je zcela v rukou uživatele, aby tyto položky vhodně navolil (aby odpovídaly danému semestru). V rámci vyvarování se vzniku kolize je pak kontrolován u vkládání akcí průnik dnu, zvolené hodiny a také tento datový interval.

1.3 Požadavky

Aplikace má podle zadání splňovat následující:

- možnost vkládat rozvrhové akce,
- zobrazení rozvrhu pro místnost, vyučujícího nebo obor a ročník,
- možnost vkládat nerozvrhová omezení pro místnost, vyučujícího nebo obor a ročník,
- vyhledání společného volného času.

2 Programátorská dokumentace

2.1 Použité technologické prostředky

Rozvrhová aplikace byla naprogramována ve vývojovém prostředí Visual Studio 2013 [3] a to v jazyce C# [4]. Pro provázání programu a databáze byl využit standardizovaný strukturovaný dotazovací jazyk SQL [5].

V následující části jsou stručně popsány a přiblíženy využití technologické prostředky.

2.1.1 Programovací jazyk C#

Programovací jazyk C# [4] vznikl na základě jazyků C++ [6] a Java [7] a zařazuje se mezi vysokoúrovňové objektově orientované programovací jazyky. Spolu s platformou .NET Framework [4] byl vyvinut firmou Microsoft. Implementace jazyka C# podporuje principy softwarového inženýrství, mezi které například patří detekování použitých neinicializovaných proměnných či automatický garbage collector (způsob automatické správy paměti, který uvolňuje ty úseky paměti, které již nejsou využívány). Uplatnění nachází při tvorbě webových aplikací, webových stránek a služeb, databázových aplikací apod.

2.1.2 Visual Studio

Microsoft Visual Studio je oficiálním vývojovým prostředím od firmy Microsoft, které umožňuje tvorbu formulářových, konzolových, webových aplikací a mnoha dalších. Obsahuje editor kódu, debugger, designer pro návrh formulářů a GUI. Dalšími podporovanými jazyky jsou například C/C++ [6], HTML/XHTML [8], CSS [8], VB.NET, Python, Ruby, JavaScript [9].

2.1.3 .NET

Standardizovanou specifikací jádra .NET [4] je Common Language Infrastructure, popisující vlastnosti kódu a prostředí vhodného pro běh aplikací. Zmíněné prostředí Microsoft .NET Framework nabízí spouštěcí rozhraní i knihovny. Aplikace se po spuštění nejprve přeloží do mezijazyka Common Intermediate Language (platformně i procesorově nezávislý soubor instrukcí). Více informací o platformě .NET a jazyku C# na: [4].

2.1.4 Microsoft Access

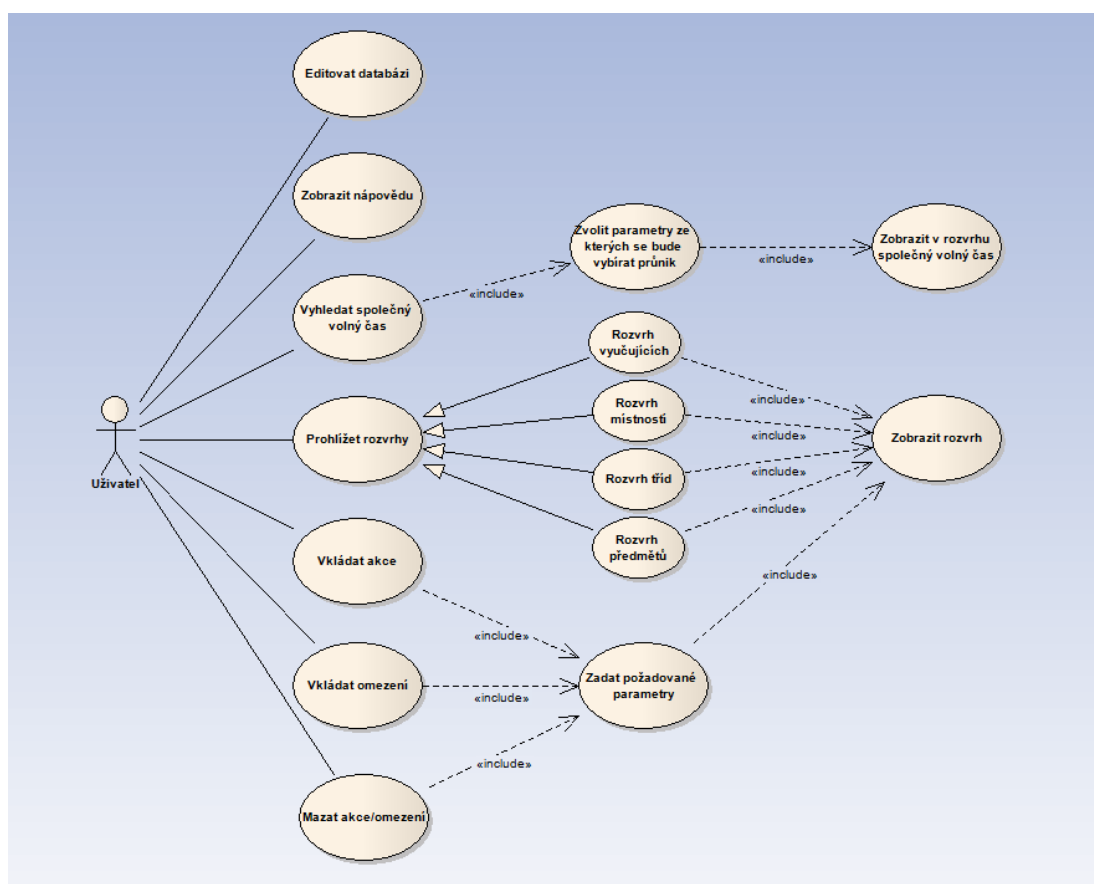
Microsoft Access [10] je nástroj pro správu databází, umožňující vytvářet databázové aplikace. Data jsou ukládána do SQL databáze. Tento software je typickou součástí balíčku Microsoft Office, který je použitelný pro jednoduché i pro složitější projekty.

2.1.5 SQL

Jazyk SQL [12] se používá při práci s relačními databázemi jakožto standardizovaný strukturovaný dotazovací jazyk. Tento jazyk byl vyvinut pro produkt společnosti IBM s názvem DB2 na konci 70. let minulého století ve vědeckých laboratořích nacházejících se v Kalifornii - přesněji v San Jose. Hlavní záměr DB2 bylo vytvoření relačního databázového systému pro různá prostředí a současně i pro různé platformy. SQL se uplatňuje při získávání dat z relačních databází a pro následnou manipulaci s nimi. S daty je pracováno jako s množinami.

2.2 Diagram užití

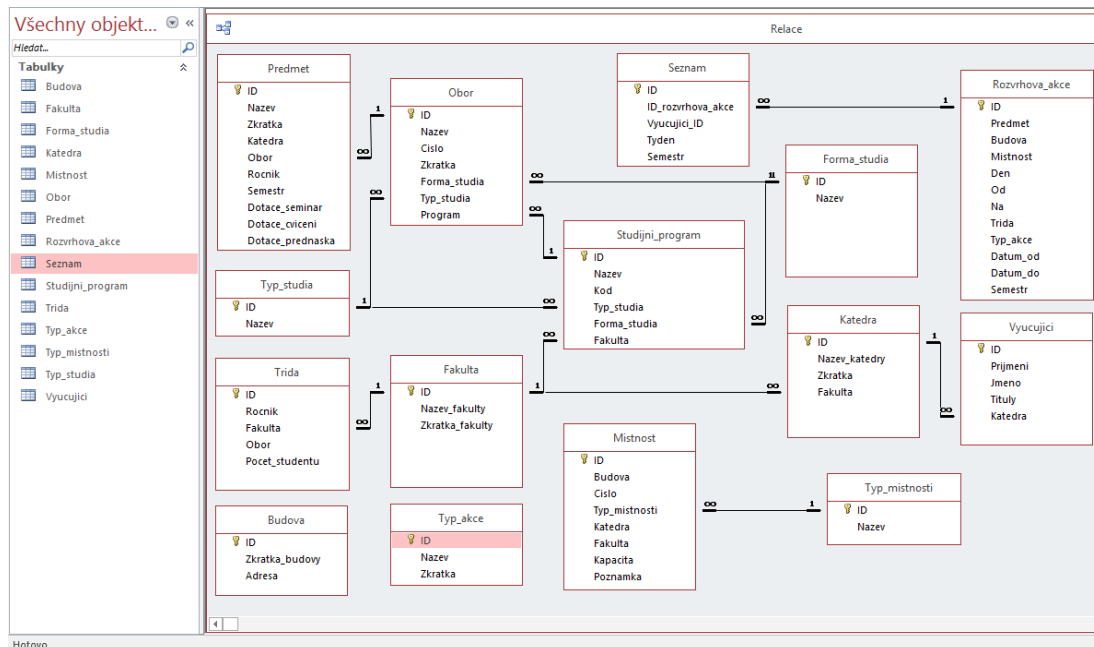
Před samotnou tvorbou aplikace byl navržen podle požadavků diagram užití, který zachycuje vnější pohled na systém. Hlavním účelem je vyobrazit vztahy mezi uživatelem a funkcionalitou programu. Tyto vztahy zachycuje následující obrázek (Obrázek 1).



Obrázek 1: Diagram užití

2.3 Databáze

Databáze vytvořená aplikací obsahuje již vytvořené tabulky a relace (Obrázek 2).



Obrázek 2: Databáze - relace

Tabulka s názvem Rozvrhova_akce přitom slouží jak k uložení akcí, které se vztahují k předmětům, tak i k ukládání omezení.

2.4 Program

Základní kostru celé aplikace tvoří pět tématicky oddělených projektů, které společně tvoří kompletní aplikaci.

2.4.1 Timetabling

Jedná se o hlavní spouštěcí projekt. Základní projektovou částí je třída Timetabling a jedná se o hlavní formulářové okno, které se zobrazí po spuštění programu. Hlavní metody třídy, které se vztahují k funkcionalitě programu:

- `public Timetabling();`

Metoda inicializuje hlavní formulář, spustí kontrolu, zda existuje v adresáři databáze Timetabling.accdb. Pokud soubor neexistuje, vytvoří se.

- `private void controlDatabase();`

Metoda provede kontrolu existence souboru s databází v adresáři, soubor vytvoří a zavolá metodu na vytvoření tabulek a relací.

- `private void createTableAndRelation();`

Metoda pro vytvoření tabulek v souboru `Timetabling.accdb` a jejich relací.

- `private void settingOpeningPanel();`

Nastavení úvodního panelu (tlačítek).

- `private void settingCreationTimetablePanel();`

Nastavení panelu (tlačítek) po kliknutí na tlačítko "Tvorba rozvrhu".

- `private void settingEditingDatabasePanel();`

Nastavení panelu (tlačítek) po kliknutí na tlačítko "Úprava databáze".

- `protected override void OnFormClosing(FormClosingEventArgs e);`

Metoda vyvolaná po kliknutí na "křížek" při ukončování aplikace, nebo při kliknutí na tlačítko "Ukončit". Zobrazí `MessageBox` s dotazem, zda chce uživatel skutečně aplikaci ukončit.

- Ostatní metody slouží k nastavení událostí po kliknutí na konkrétní tlačítka v nabídce okna. Buďto změni nabídku v aktuálním okně a nebo otevřou formulář určený ke konkrétní volbě. Případně metody slouží k nastavení některých GUI prvků.

2.4.2 Classes

Dílčí projekt, obsahující třídy vztahující se k databázi v souboru `Timetabling.accdb`. Každou tabulku realizuje třída.

- Třída `ActionClass` odpovídající tabulce v souboru `Timetabling.accdb` s názvem `Rozvrhova_akce`.

Vlastnosti (položky v závorce odpovídají názvům sloupců v konkrétní tabulce):

1. `public int id { get; set; }`

Nastavuje a vrací vlastnost `id` (ID).

2. `public string subjectName { get; set; }`

Nastavuje a vrací vlastnost `subjectName` (Predmet).

3. `public string buildingAbbreviation { get; set; }`

Nastavuje a vrací vlastnost `buildingAbbreviation` (Budova).

4. `public string roomNumber { get; set; }`
Nastavuje a vrací vlastnost `roomNumber` (Mistnost).
5. `public int dayNumber { get; set; }`
Nastavuje a vrací vlastnost `dayNumber` (Den).
6. `public int from { get; set; }`
Nastavuje a vrací vlastnost `from` (Od).
7. `public int till { get; set; }`
Nastavuje a vrací vlastnost `till` (Na).
8. `public int classId { get; set; }`
Nastavuje a vrací vlastnost `classId` (Trida).
9. `public string typeOfActionName { get; set; }`
Nastavuje a vrací vlastnost `typeOfActionName` (Typ_akce).
10. `public string dateFrom { get; set; }`
Nastavuje a vrací vlastnost `dateFrom` (Datum_od).
11. `public string dateTo { get; set; }`
Nastavuje a vrací vlastnost `dateTo` (Datum_do).
12. `public string semesterAbbreviation { get; set; }`
Nastavuje a vrací vlastnost `semesterAbbreviation` (Semestr).

Metody:

1. `public override string ToString()`

Metoda pro výpis v řetězci.

- Třída `Building` odpovídající tabulce v souboru `Timetabling.accdb` s názvem `Budova`.

Vlastnosti:

1. `public int id { get; set; }`
Nastavuje a vrací vlastnost `id` (ID).
2. `public string abbreviation { get; set; }`
Nastavuje a vrací vlastnost `abbreviation` (Zkratka_budovy).

```
3. public string address { get; set; }
```

Nastavuje a vrací vlastnost address (Adresa).

Metody:

```
1. public override string ToString()
```

Metoda pro výpis v řetězci.

- Třída Department odpovídající tabulce v souboru Timetabling.accdb s názvem Katedra.

Vlastnosti:

```
1. public int id { get; set; }
```

Nastavuje a vrací vlastnost id (ID).

```
2. public string name { get; set; }
```

Nastavuje a vrací vlastnost name (Nazev_katedry).

```
3. public string abbreviation { get; set; }
```

Nastavuje a vrací vlastnost abbreviation (Zkratka).

```
4. public int facultyId { get; set; }
```

Nastavuje a vrací vlastnost facultyId (Fakulta).

Metody:

```
1. public override string ToString()
```

Metoda pro výpis v řetězci.

- Třída Faculty odpovídající tabulce v souboru Timetabling.accdb s názvem Fakulta.

Vlastnosti:

```
1. public int id { get; set; }
```

Nastavuje a vrací vlastnost id (ID).

```
2. public string name { get; set; }
```

Nastavuje a vrací vlastnost name (Nazev_fakulty).

```
3. public string abbreviation { get; set; }
```

Nastavuje a vrací vlastnost abbreviation (Zkratka_fakulty).

Metody:

1. `public override string ToString()`

Metoda pro výpis v řetězci.

- Třída `FieldOfStudy` odpovídající tabulce v souboru `Timetabling.accdb` s názvem `Obor`.

Vlastnosti:

1. `public int id { get; set; }`

Nastavuje a vrací vlastnost `id` (ID).

2. `public string name { get; set; }`

Nastavuje a vrací vlastnost `name` (Název).

3. `public string number { get; set; }`

Nastavuje a vrací vlastnost `number` (Číslo).

4. `public string abbreviation { get; set; }`

Nastavuje a vrací vlastnost `abbreviation` (Zkratka).

5. `public int formOfStudyId { get; set; }`

Nastavuje a vrací vlastnost `formOfStudyId` (Forma_studia).

6. `public int typeOfStudyId { get; set; }`

Nastavuje a vrací vlastnost `typeOfStudyId` (Typ_studia).

7. `public int programId { get; set; }`

Nastavuje a vrací vlastnost `programId` (Program).

Metody:

1. `public override string ToString()`

Metoda pro výpis v řetězci.

- Třída `FormOfStudy` odpovídající tabulce v souboru `Timetabling.accdb` s názvem `Forma_studia`.

Vlastnosti:

1. `public int id { get; set; }`

Nastavuje a vrací vlastnost `id` (ID).

2. `public string name { get; set; }`

Nastavuje a vrací vlastnost name (Nazev).

Metody:

1. `public override string ToString()`

Metoda pro výpis v řetězci.

- Třída `ListAction` odpovídající tabulce v souboru `Timetabling.accdb` s názvem `Seznam`.

Vlastnosti:

1. `public int id { get; set; }`

Nastavuje a vrací vlastnost id (ID).

2. `public int actionClassId { get; set; }`

Nastavuje a vrací vlastnost `actionClassId` (`ID_rozvrhova_akce`).

3. `public int teacherId { get; set; }`

Nastavuje a vrací vlastnost `teacherId` (`Vyucujici_ID`).

4. `public string week { get; set; }`

Nastavuje a vrací vlastnost `week` (Tyden).

5. `public string semester { get; set; }`

Nastavuje a vrací vlastnost `semester` (Semestr).

- Třída `NewClass` odpovídající tabulce v souboru `Timetabling.accdb` s názvem `Trida`.

Vlastnosti:

1. `public int id { get; set; }`

Nastavuje a vrací vlastnost id (ID).

2. `public int year { get; set; }`

Nastavuje a vrací vlastnost `year` (Rocnik).

3. `public int facultyId { get; set; }`

Nastavuje a vrací vlastnost `facultyId` (Fakulta).

4. `public string fieldAbbreviation { get; set; }`

Nastavuje a vrací vlastnost `fieldAbbreviation` (Obor).

5. `public int countStudent { get; set; }`

Nastavuje a vrací vlastnost `countStudent` (`Pocet_studentu`).

Metody:

1. `public override string ToString()`

Metoda pro výpis v řetězci.

- Třída `Room` odpovídající tabulce v souboru `Timetabling.acddb` s názvem `Mistnost`.

Vlastnosti:

1. `public int id { get; set; }`

Nastavuje a vrací vlastnost `id` (`ID`).

2. `public string buildingAbbreviation { get; set; }`

Nastavuje a vrací vlastnost `buildingAbbreviation` (`Budova`).

3. `public string number { get; set; }`

Nastavuje a vrací vlastnost `number` (`Cislo`).

4. `public int typeIdOfRoomId { get; set; }`

Nastavuje a vrací vlastnost `typeOfRoomId` (`Typ_mistnosti`).

5. `public int departmentId { get; set; }`

Nastavuje a vrací vlastnost `departmentId` (`Katedra`).

6. `public int facultyId { get; set; }`

Nastavuje a vrací vlastnost `facultyId` (`Fakulta`).

7. `public int capacity { get; set; }`

Nastavuje a vrací vlastnost `capacity` (`Kapacita`).

8. `public string note { get; set; }`

Nastavuje a vrací vlastnost `note` (`Poznamka`).

Metody:

1. `public override string ToString()`

Metoda pro výpis v řetězci.

- Třída StudyProgram odpovídající tabulce v souboru Timetabling.accdb s názvem Studijni_program.

Vlastnosti:

1. `public int id { get; set; }`
Nastavuje a vrací vlastnost id (ID).
2. `public string name { get; set; }`
Nastavuje a vrací vlastnost name (Nazev).
3. `public string code { get; set; }`
Nastavuje a vrací vlastnost code (Kod).
4. `public int typeOfStudyId { get; set; }`
Nastavuje a vrací vlastnost typeOfStudyId (Typ_studia).
5. `public int formOfStudyId { get; set; }`
Nastavuje a vrací vlastnost formOfStudy (Forma_studia).
6. `public int facultyId { get; set; }`
Nastavuje a vrací vlastnost facultyId (Fakulta).

Metody:

1. `public override string ToString()`
Metoda pro výpis v řetězci.

- Třída Subject odpovídající tabulce v souboru Timetabling.accdb s názvem Predmet.

Vlastnosti:

1. `public int id { get; set; }`
Nastavuje a vrací vlastnost id (ID).
2. `public string name { get; set; }`
Nastavuje a vrací vlastnost name (Nazev).
3. `public string abbreviation { get; set; }`
Nastavuje a vrací vlastnost abbreviation (Zkratka).
4. `public string departmentAbbreviation { get; set; ~ }`

Nastavuje a vrací vlastnost departmentAbbreviation (Katedra).

5. `public int fieldId { get; set; }`

Nastavuje a vrací vlastnost fieldId (Obor).

6. `public int yearId { get; set; }`

Nastavuje a vrací vlastnost yearId (Rocnik).

7. `public int seminar { get; set; }`

Nastavuje a vrací vlastnost seminar (Dotace_seminar).

8. `public int exercise { get; set; }`

Nastavuje a vrací vlastnost exercise (Dotace_cviceni).

9. `public int lecture { get; set; }`

Nastavuje a vrací vlastnost lecture (Dotace_prednaska).

10. `public string semesterAbbreviation { get; set; }`

Nastavuje a vrací vlastnost semesterAbbreviation (Semestr).

Metody:

1. `public override string ToString()`

Metoda pro výpis v řetězci.

- Třída Teacher odpovídající tabulce v souboru Timetabling.accdb s názvem Vyucujici.

Vlastnosti:

1. `public int id { get; set; }`

Nastavuje a vrací vlastnost id (ID).

2. `public string surname { get; set; }`

Nastavuje a vrací vlastnost surname (Prijmeni).

3. `public string firstname { get; set; }`

Nastavuje a vrací vlastnost firstname (Jmeno).

4. `public string titles { get; set; }`

Nastavuje a vrací vlastnost titles (Tituly).

5. `public int departmentId { get; set; }`

Nastavuje a vrací vlastnost departmentId (Katedra).

Metody:

1. `public override string ToString()`

Metoda pro výpis v řetězci.

- Třída `TypeOfAction` odpovídající tabulce v souboru `Timetabling.accdb` s názvem `Typ_akce`.

Vlastnosti:

1. `public int id { get; set; }`

Nastavuje a vrací vlastnost `id` (ID).

2. `public string name { get; set; }`

Nastavuje a vrací vlastnost `name` (Název).

3. `public string abbreviation { get; set; }`

Nastavuje a vrací vlastnost `abbreviation` (Zkratka).

Metody:

1. `public override string ToString()`

Metoda pro výpis v řetězci.

- Třída `TypeOfRoom` odpovídající tabulce v souboru `Timetabling.accdb` s názvem `Typ_mistnosti`.

Vlastnosti:

1. `public int id { get; set; }`

Nastavuje a vrací vlastnost `id` (ID).

2. `public string name { get; set; }`

Nastavuje a vrací vlastnost `name` (Název).

Metody:

1. `public override string ToString()`

Metoda pro výpis v řetězci.

- Třída `TypeOfStudy` odpovídající tabulce v souboru `Timetabling.accdb` s názvem `Typ_studia`.

Vlastnosti:

```
1. public int id { get; set; }
```

Nastavuje a vrací vlastnost id (ID).

```
2. public string name { get; set; }
```

Nastavuje a vrací vlastnost name (Nazev).

Metody:

```
1. public override string ToString()
```

Metoda pro výpis v řetězci.

2.4.3 Tables

Projekt obsahuje grafickou část. Vykreslení jednotlivých akcí i celé rozvrhové tabulky. Třídy využívají pro grafické znázornění System.Drawing Namespace.

- Třída FieldForOne slouží k vytvoření pole pro akci. Toto pole má atributy pro nastavení textu, nastavení velikosti, obrysu, pozice apod.

Metody:

```
1. public void setText(string newName);
```

Metoda nastavuje do pole pro akci text.

```
2. public void Draw(Graphics g, int x, int y);
```

Metoda podle předaných parametrů nastaví šířku a výšku pole. Pokud má instance třídy nastavený text, tak vykreslí políčko i s příslušným textem, jinak bude prázdné.

```
3. public void Click(int x, int y);
```

Metoda slouží pro využití po kliknutí na políčko. Změní barvu obrysu pro zvýraznění.

- Třída FieldForX slouží k vytvoření několika polí pod sebou. Má podobné funkce jako třída FieldForOne výše.
- Třída Table slouží k vizualizaci tabulky s rozvrhem. Tato třída obsahuje různé metody pro jejich vykreslení, dle toho jaké informace uživatel momentálně potřebuje.

Je možné například zobrazit rozvrh jednotlivých studentů, vyučujících, místností, rozvrhů či průmětu volných časů a mnohé další. Nejprve nastavíme třídě atribut přijímající seznam akcí. Při zavolání příslušné funkce dojde k nastavení dalších parametrů tabulky. Poté se nastaví velikost tabulky v závislosti na dnu a času konaných akcí, které zjistíme z vlastností

o dané akci. Následně je volána další funkce pro vykreslení rozvrhu společně se všemi akcemi.

Metody:

1. `public void setting();`

Podle zaslání seznamu akcí nastaví požadované velikosti tabulce. Využívá se při vykreslení klasického zobrazení rozvrhu nebo při sloučení vícero rozvrhů do jednoho.

2. `public void Draw(Graphics g, int x, int y);`

Metoda podle předaných parametrů vykreslí tabulku s rozvrhem na tyto souřadnice.

3. `public void setting2();`

Metoda nastaví velikosti tabulce. Jedná se o metodu, která je využita při vykreslení společného volného času.

4. `public void Draw2(Graphics g, int x, int y);`

Metoda podle předaných parametrů vykreslí tabulku pro znázornění společného volného času s rozvrhem na tyto souřadnice.

2.4.4 Administration

Tento projekt slouží k přímé práci s databází. Obsahuje dvě třídy. Třída SQL obsahuje metody, z nichž každá vrací řetězec, představující SQL dotaz. Dotazy jsou poté využívány ve třídě Administration.

Metody třídy Administration se díky těmto dotazům starají o vkládání, úpravu a mazání dat v databázi Timetabling.acddb. Jelikož je dotazů mnoho, byly odděleny od třídy Administration, pro zvýšení přehlednosti a snadnějšímu vyhledání.

2.4.5 Forms

Projekt slouží k oddělení formulářů od hlavního projektu. Každý formulář nabízí jinou funkcionalitu.

Formuláře určené pro úpravu a editaci databáze (mimo tabulku pro rozvrhové akce):

- Buildings_Faculties_Departments - Obsahuje tři záložky pro editaci budov, fakult a kateder.
- FormsOfStudy_TypesOfStudy - Obsahuje dvě záložky pro editaci forem a typů studia.
- NewClass_Form - Formulář pro editaci třídy (oboru a ročníku).

- StudyPrograms_FieldsOfStudy_Subjects - Obsahuje tři záložky pro editaci studijních programů, oborů a předmětů.
- Teachers - Formulář pro editaci vyučujících.
- TypesOfRooms_Rooms - Obsahuje dvě záložky pro editaci místností a jejich typů.

Formuláře s další funkcionalitou:

- Actions - Formulář slouží především pro vkládání rozvrhových akcí a pro vkládání omezení.

Důležité metody:

1. `private void insert_Click(object sender, EventArgs e);`

Jedná se o metodu, která se vyvolá po stisknutí tlačítka určeného k vkládání akcí. Dojde k vytvoření nové třídy `ActionClass` a požadované položky z formuláře (pokud by nebyly vyplněny, nelze tlačítku předat událost), jsou nastaveny této třídě jako atributy. Dojde k ověření, zda už v tuto dobu vyučující nebo místnost nemá v databázi vloženou akci. Pokud ne, dojde k uložení, pokud ano, uživateli je zaslán `MessageBox` se zprávou, že nebylo možno akci vložit.

Po úspěšném uložení akce se aktualizuje část formuláře určená pro vykreslování rozvrhů a část, která nese informace o počtech studentů a dostupnosti k akci.

2. `private void deleteBySemester_Click(object sender, MouseEventArgs e);`

Umožňuje smazat všechny akce ve zvoleném semestru.

3. `private void insertLimitTeacher_Click(object sender, EventArgs e);`

Pracuje na obdobném principu jako předchozí metoda určená k vkládání akcí. Umožňuje vložit omezení pro vyučujícího.

4. `private void insertLimitRoom_Click(object sender, EventArgs e);`

Umožňuje vložit omezení pro místnost.

5. `private void insertLimitClass_Click(object sender, EventArgs e);`

Umožňuje vložit omezení pro třídu (obor a ročník).

Zobrazené rozvrhy reagují na klikání myši. Po stisknutí akce v rozvrhovém zobrazení pravým tlačítkem se zobrazí menu.

Tímto můžeme zjistit více informací o dané akci, případně můžeme vyvolat její smazání z databáze.

- Browsing - Funkcí tohoto formuláře je prohlížení rozvrhů.

Uživatel si zvolí, jaký rozvrh si chce prohlédnout (rozvrh vyučujícího, místnosti, předmětu či třídy). Tyto parametry může zvyšovat. Může si např. prohlédnout rozvrh dvou vyučujících najednou. Je ponecháno na rozhodnutí uživatele, zda se rozvrhy zobrazí zvlášť, nebo zda si je sloučí do jedné tabulky.

Hlavní metody:

1. `private void display_Click(object sender, EventArgs e);`

Událost vyvolaná po stisknutí tlačítka určeného k zobrazení rozvrhu vyučujícího nebo vyučujících. Pokud je nastaveno, aby se tabulky sloučily, budou akce vykresleny v jedné tabulce. Jinak se vykreslí pro každého vyučujícího samostatný rozvrh.

Metoda vytvoří třídu Table a pošle jí seznam akcí, příslušný pro zvolené vyučující. Dále je zavolána metoda pro nastavení této třídy.

2. `private void pictureBox1_Paint(object sender, PaintEventArgs e);`

Metoda vykreslí rozvrh pro zvolené položky sloučeně.

3. `private void pictureBox2_Paint(object sender, PaintEventArgs e);`

Metoda vykreslí rozvrh pro zvolené položky zvlášť.

- Searching - Funkcí tohoto formuláře je vyhledávání společného volného času mezi vyučujícími, místnostmi a třídami (obor a ročník).

Funkce pro zobrazení rozvrhu pošle seznam akcí třídě Table a na jeho základě je poté zobrazena interaktivní tabulka. Tabulka po kliknutí nabízí seznam akcí (zobrazeny červeně), kterých se v danou dobu účastní uživatelem zvolené položky.

3 Uživatelská dokumentace

3.1 Systémové požadavky

Aplikace je určena pro operační systém Microsoft Windows a jedná se konkrétně o 64-bitovou verzi. Běží na základě databáze, která je vytvořena v databázovém softwaru Microsoft Access. Je tedy nutné mít program Microsoft Access nainstalovaný v počítači, na kterém bude aplikace spuštěna. Pokud chybí, je třeba nainstalovat Microsoft Office balíček.

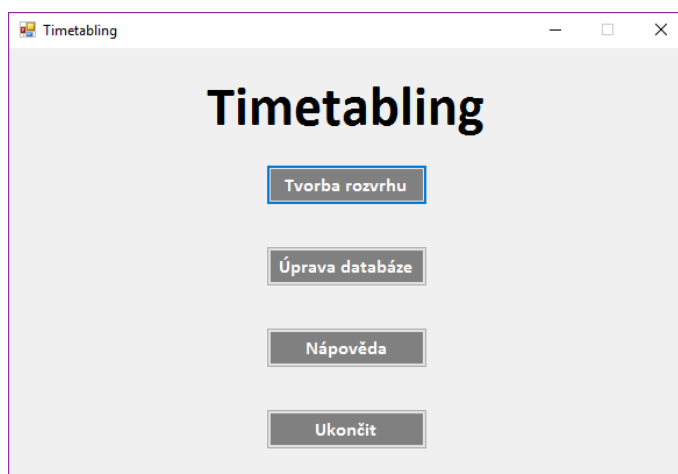
Spojení se zmiňovanou databází umožňuje zprostředkovatel Microsoft Access Database Engine 2010 Redistributable. K dispozici je na příloženém CD, jinak lze doinstalovat ze stránek [11].

3.2 Instalace aplikace

Aplikace se spouští pomocí souboru Timetabling.exe a nevyžaduje instalaci. Tento soubor je k nalezení na příloženém CD a lze jej spustit přímo z tohoto disku nebo lze soubor zkopírovat na libovolné místo v počítači a dále s ním pracovat.

3.3 Spuštění aplikace

Po spuštění aplikace se provede kontrola, zda na stejném místě v adresáři jako je uložen soubor Timetabling.exe existuje také soubor Timetabling.accdb, pokud ne, vytvoří se spolu s potřebnými tabulkami, pokud ano, program má vše, co k chodu potřebuje a zobrazí se úvodní navigační okno (Obrázek 3).



Obrázek 3: Úvodní okno

Co se týče potřebné a již zmiňované databáze k programu, jsou k dispozici 3 varianty.

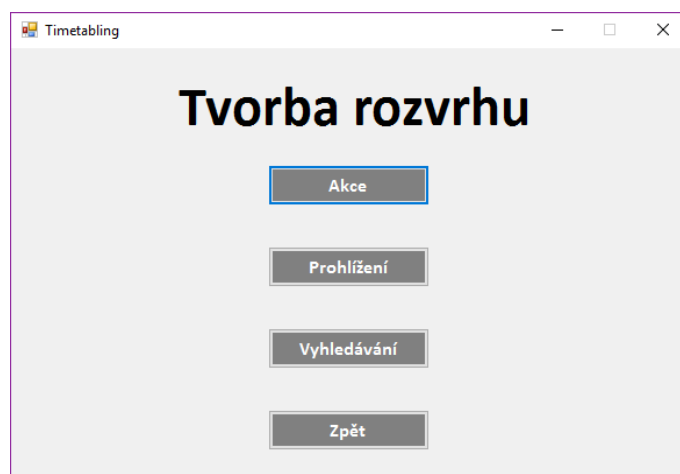
1. Při spuštění aplikace neexistuje soubor Timetabling.accdb v adresáři. V tomto případě je vytvořen soubor nový. Před jakoukoliv prací s hlavní funkcionalitou programu je tedy třeba nejprve hlavní položky databáze vyplnit. K tomuto účelu je vytvořena v programu část pro úpravu databáze (viz níže).
2. Na přiloženém CD je k dispozici soubor Timetabling.accdb s již předvyplněnými základními, ovšem ne zcela kompletními údaji. Vyplnění rozvrhových akcí a omezení zde chybí. Jsou zde vyplněny pouze částečné informace vztahující se k fakultám, katedrám, učebnám, vyučujícím apod. Všechny tyto informace lze opět editovat a provádět s nimi další operace.
3. Poslední variantou je ještě jeden soubor s názvem Timetabling2.accdb, který slouží pro otestování správnosti, aby nebylo třeba se zdržovat zdoluhavým vyplňováním dat. Od předchozího bodu se liší tím, že je zde již vyplněno několik rozvrhových akcí i omezení pro vyučující, učebnu či obor a ročník.

Je třeba dodat, že při využití posledních dvou bodů, tedy použití již vytvořených a naplněných souborů s databází, je třeba zkopírovat konkrétní soubor z CD do adresáře, ve kterém je uložen soubor Timetabling.exe, a při použití souboru z posledního bodu, je ještě třeba soubor přejmenovat na Timetabling.accdb.

3.4 Základní nabídka funkcí

Úvodní okno (Obrázek 3 po spuštění aplikace) nabízí čtyři tlačítka. Každé z nich vyčleňuje určitou část funkcionality programu.

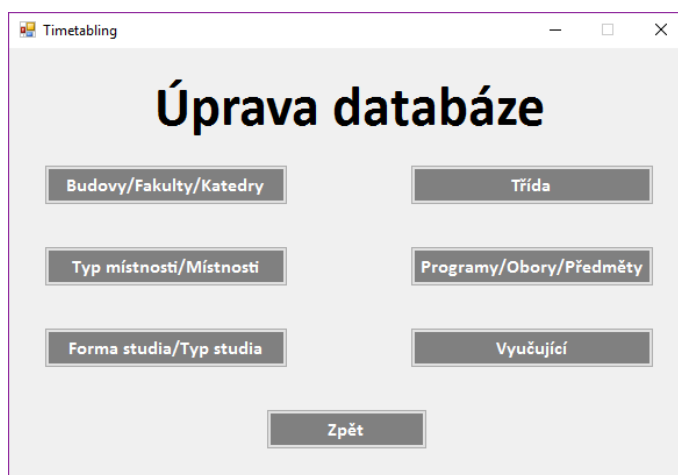
1. Tvorba rozvrhu - V aktuálním okně se zobrazí nabídka určená pro vytváření rozvrhů (Obrázek 4).



Obrázek 4: Tvorba rozvrhu

Jsou zde možnosti pro vkládání rozvrhových akcí, jejich mazání, prohlížení rozvrhů a vyhledávání společného volného času (viz níže).

2. Úprava databáze - V aktuálním okně se zobrazí nabídka pro správu databáze v souboru Microsoft Access (Obrázek 5). Tato část umožňuje vkládat, upravovat a mazat data v databázi. Nezahrnuje však správu dat, která nesou informace o uložených akcích (předmětech a omezeních). Tato správa je možná v části určené pro vkládání akcí.



Obrázek 5: Úprava databáze

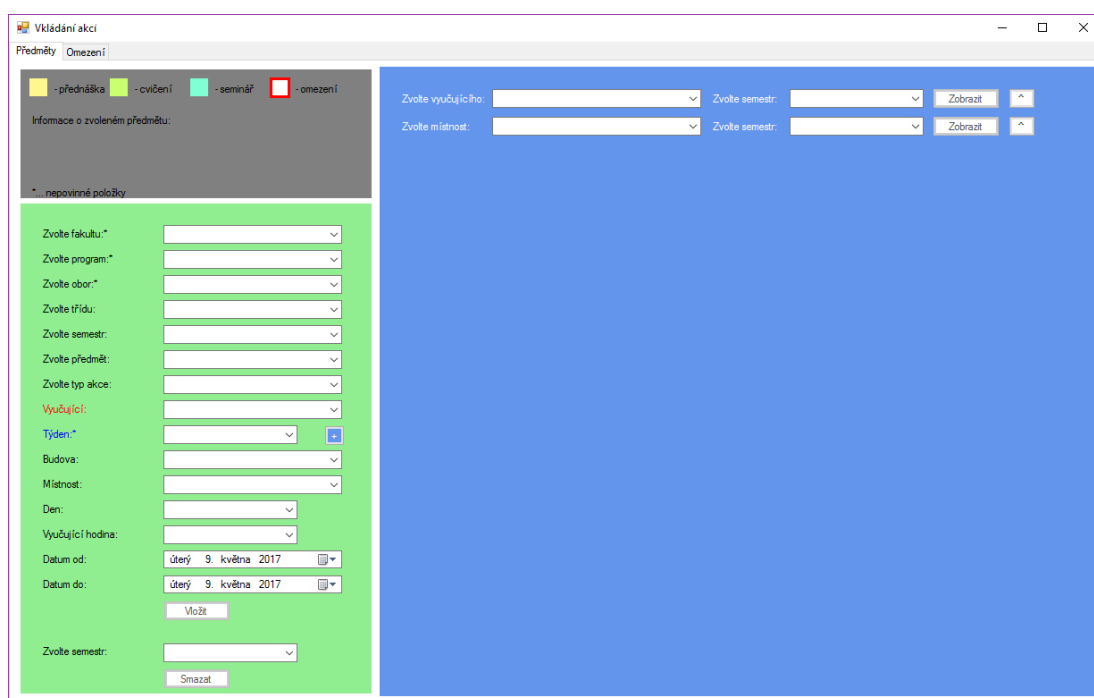
3. Nápověda - Otevře nové okno, které obsahuje nápovědu k obsluze programu.
4. Ukončit - Slouží k ukončení programu stejně jako křížek vpravo nahoře. Po jejich stisknutí se zobrazí dialogové okno pro ověření, zda chceme aplikaci skutečně zavřít. Úvodní okno slouží jako hlavní nabídka při práci s programem a pokud ho budeme chtít zavřít, zavřou se i všechna ostatní případně otevřená okna. Slouží tedy k zajištění bezpečnosti.

3.5 Tvorba rozvrhu

Tato část menu se zobrazí po kliknutí na tlačítko "Tvorba rozvrhu" v úvodní nabídce a nabízí opět čtyři tlačítka.

1. Akce - Otevřené okno se skládá ze čtyř částí (Obrázek 6). Hlavní funkcionalitou je vkládání předmětů do databáze a následné zobrazení vložené akce. Dále vkládání omezení pro vyučujícího, místnost nebo obor a ročník, případně prohlížení rozvrhů a mazání akcí podle zvoleného semestru nebo přímo konkrétně zvolené akce ve vykresleném rozvrhu.

Možnost prohlížení je zde dodána pro zvýšení pohodlí při tvorbě a návrhu akcí, neboť uživatel si před samotným vložením akce může prohlédnout, jaký je aktuální stav konkrétního rozvrhu, který bude změněn.



Obrázek 6: Akce - okno učené pro správu akcí v rozvrzích

Vlevo nahoře se nachází dvě záložky:

- (a) Předměty - Na této záložce je možné vkládat pouze akce týkající se předmětů. Pod záložkami se okno skládá ze tří panelů. Šedý slouží jako informační panel. Obsahuje vysvětlivky pro barevné znázornění, které se bude vyskytovat v rozvrhových tabulkách. Poté, co zvolíme v zeleném panelu třídu, semestr a konkrétní předmět, obsahuje taky informace o tomto zvoleném předmětu (Obrázek 7).

Nese tedy údaj o celkovém počtu studentů, kteří studují zvolený obor ve zvoleném ročníku. Dále vypíše jak je předmět dotovaný a podle vypsaných akcí a kapacit místností, ve kterých budou akce realizovány, přepočítá a vypíše, kolik studentů se nebude moci zapsat případně kolik studentů se bude moci zapsat mimo doporučený obor a ročník.

Pod informačním panelem se nachází zelený panel, který obsahuje vyplňovací formulář. Po navolení třídy, semestru a předmětu se v pravé části okna (v modrém panelu) objeví dvě rozvrhové tabulky (Obrázek 7), stejně jako se aktualizuje informační panel.

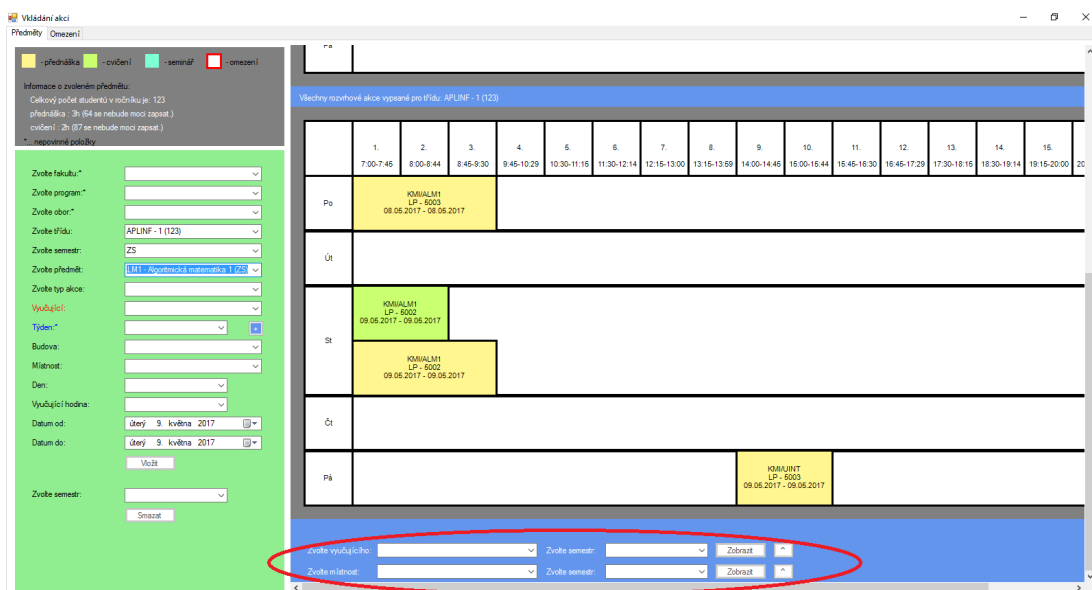
První z nich se vztahuje ke zvolenému předmětu. Zobrazuje všechny akce, které jsou k tomuto předmětu vypsané. Druhá tabulka se vztahuje k samotnému oboru a ročníku. Obě tabulky jsou zde umístěny pro přehlednost a snadný a rychlý přístup k informacím, které jsou zrovna potřeba.

Ve spodní části zeleného panelu je k dispozici ještě možnost pro zvo-



Obrázek 7: Akce - příklad

lení semestru, čímž se povolí tlačítko určené ke smazání akcí. Pokud uživatel zvolí semestr a stiskne toto tlačítko, program se pro ověření správnosti zeptá, zda chce uživatel všechny akce vázající se ke zvolenému semestru smazat, a po potvrzení zmíněné akce smaže z databáze. Uživatel také může kliknout na libovolnou akci vykreslenou v rozvrhu. Konkrétně po stisknutí pravého tlačítka se zobrazí menu s možností zobrazení bližších informací ke zvolené akci a dále s možností smazání akce.

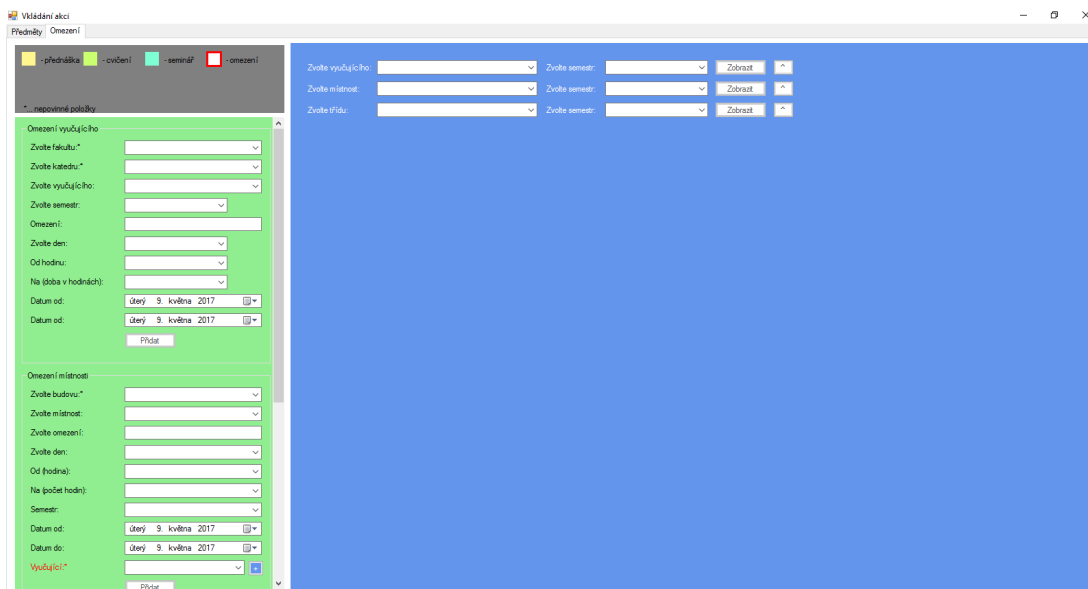


Obrázek 8: Akce - příklad

Při vyplnění povinných položek formuláře (nepovinné jsou označeny symbolem "*"), se povolí stisk tlačítka pro vložení akce. Po stisku tohoto tlačítka se buď aktualizuje modrá plocha a překreslí se již zobrazené rozvrhy na aktuální verzi nebo se teprve vykreslí obdobně, jak bylo zmíněno výše.

Pod oběma tabulkami se nachází panel, sloužící k rozšíření zobrazených rozvrhů (Obrázek 8). V případě potřeby si zde může uživatel zobrazit rozvrh libovolné místnosti, například takové, do které chce budoucí akci vložít, nebo rozvrh vyučujícího, který má budoucí akci vyučovat. Díky tomu si může předem ověřit, zda je zvolená místnost v danou dobu a den volná nebo zda vyučující v danou dobu již nevyučuje, případně zda nemá v rozvrhu nějaké omezení.

- (b) Omezení - Záložka slouží k vkládání nerozvrhových akcí, jinými slovy k vkládání omezení, a to jak pro vyučujícího, místnost, tak i pro obor a ročník (Obrázek 9).



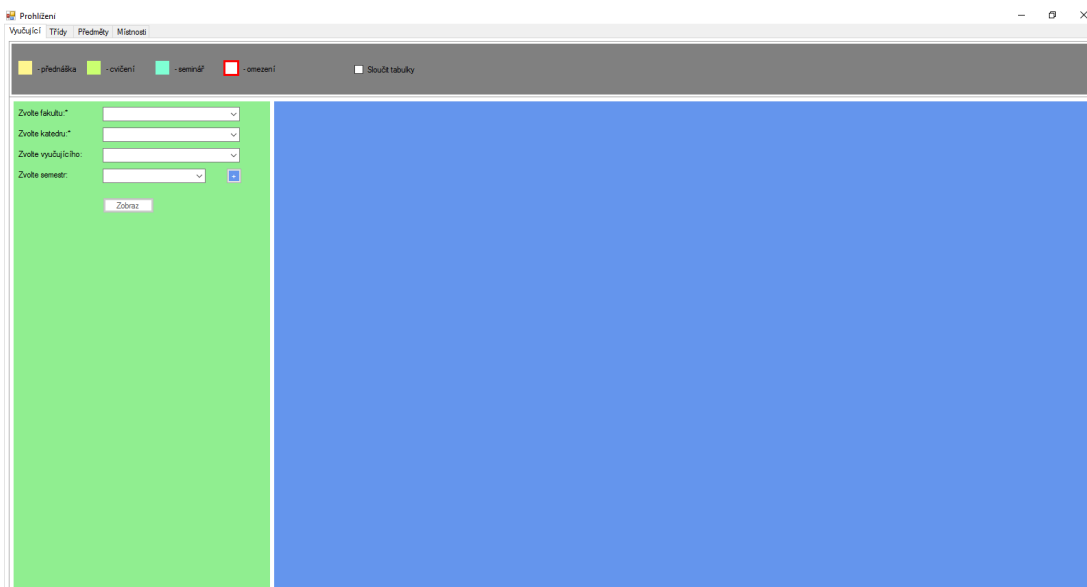
Obrázek 9: Akce - Omezení

Okno je situování velmi obdobně jako u záložky pro vkládání předmětů. V informačním panelu nejsou informace o zvoleném předmětu ani o počtech studentů. Nabídka v zeleném panelu nabízí jiné možnosti. Obsahuje tři formuláře. První umožňuje vkládání omezení pro vyučujícího, druhý pro místnost a třetí pro třídu (třída je brána jako spojení oboru a ročníku, jak již bylo zmíněno). Vložená omezení jsou poté v rozvrhu znázorněná červeným orámováním, jak je zmíněno v informačním panelu. V modrém panelu vpravo je opět panel pro zobrazení a prohlížení rozvrhů, které bude chtít uživatel případně měnit.

Jedná o různá omezení ze strany vyučujícího, omezení pro místnost

či pro obor a ročník. Je to například stanovení, že určitý vyučující nebude moci učit v pátek.

2. Prohlížení - Otevře nové okno určené pro prohlížení rozvrhů (Obrázek 10).



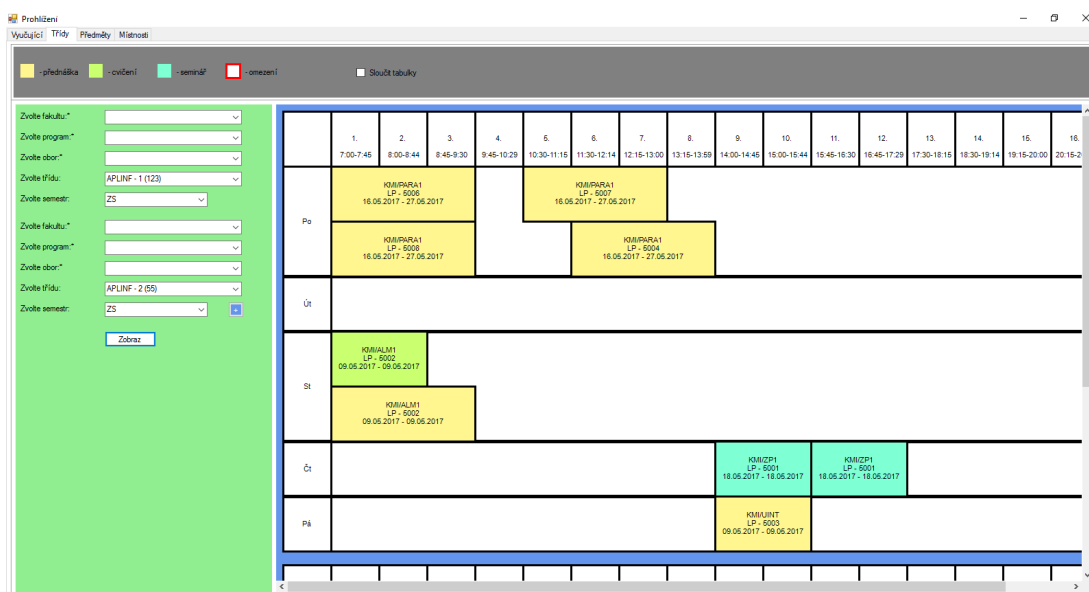
Obrázek 10: Prohlížení

Okno můžeme rozdělit na několik částí:

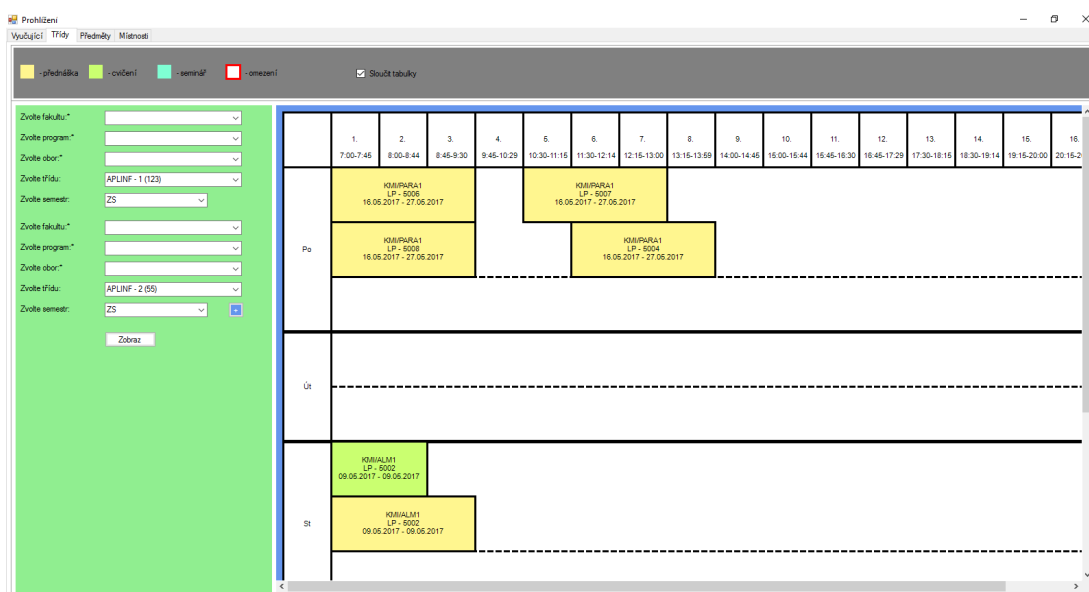
- V levé horní části okna se nachází záložky (Vyučující, Třídy, Předměty, Místnosti). Označují a vymezují část, kterou chceme prohlížet.
- Pod záložkami se nachází šedý panel s vysvětlivkami pro barvy, které se budou v zobrazené rozvrhové tabulce vyskytovat.
- Pod šedým panelem vlevo se nachází zelený panel obsahující formulář pro vyplnění vyučujícího/třídy/místnosti/předmětu. Na základě této volby bude vygenerován rozvrh. Tlačítko "+" umožňuje přidání více položek a tlačítko "Zobrazit" odešle požadavek na zobrazení rozvrhu.
- Modrá část je vyhrazena pro zobrazení rozvrhu.

V této části aplikace si může uživatel prohlížet rozvrhy, podle navolených parametrů z již vložených akcí do databáze. Na obrázku 11 jsem uvedla příklad, při kterém se pohybujeme na záložce "Třídy" a jsou navoleny dvě různé třídy (myšleno jako spojení oboru a ročníku).

V tomto případě se tabulky jednotlivých tříd zobrazí postupně pod sebe. Pokud ale v šedém panelu zvolíme možnost "Sloučit tabulky", dostaneme spojený vzhled tabulek (Obrázek 12). Oba zvolené rozvrhy jsou spojeny do jednoho a rozděleny přerušovanou čarou. Toto zobrazení může být v mnohých případech přehlednější.



Obrázek 11: Prohlížení - příklad samostatného zobrazení



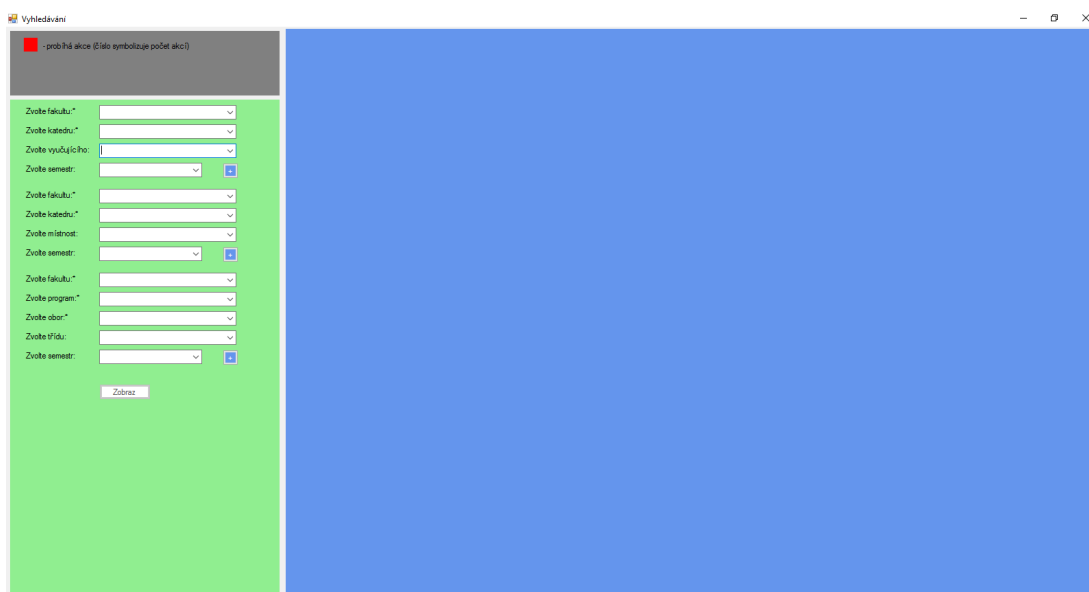
Obrázek 12: Prohlížení - příklad spojení tabulek

U obou variant zobrazení je možné z obrázků vidět, že se vpravo i dole zobrazí scrollbar pro zobrazení celé tabulky s rozvrhem.

3. Vyhledávání - Otevře nové okno sloužící pro vyhledávání společného volného času (Obrázek 13).

Okno se skládá ze tří částí:

- Vlevo nahoře se nachází šedě zbarvený panel. Obsahuje vysvětlivku



Obrázek 13: Vyhledávání

k barvě, která se bude vyskytovat v zobrazeném rozvrhu. V tomto případě se jedná o barvu červenou.

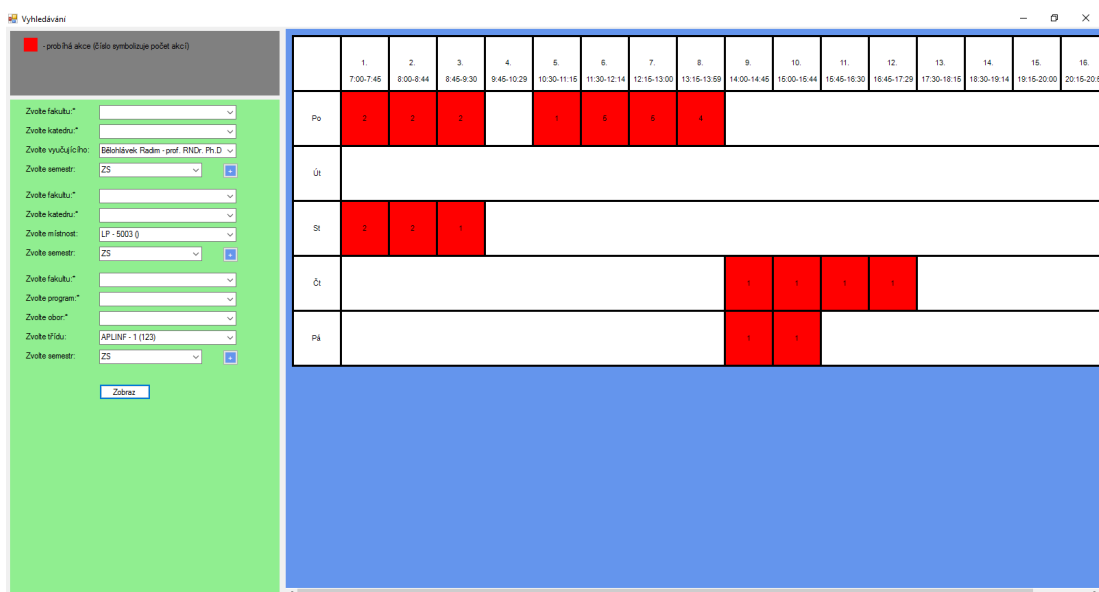
- Pod šedým panelem se nachází zelený panel, v němž můžeme volit položky, které poté chceme zobrazit. Jedná se tedy o vyučujícího, místnost a třídu, která je zde chápána jako spojení ročníku a příslušného oboru. Každou z položek můžeme přidat pomocí tlačítka "+".
- Třetí částí je modrá plocha vpravo, na níž bude následně po stisknutí tlačítka "Zobrazit" vykreslena rozvrhová tabulka přizpůsobená k nalezení společného volného času.

Na Obrázku 14 je zobrazen příklad, kdy je zvolen vyučující, místnost a jedna třída.

Kombinace lze zvolit různě a nemusí být vyplněny všechny položky (například pouze jeden vyučující a místnost apod.). Na zmíněném příkladu můžeme vidět vzhled tabulky. Jak je zmíněno ve vysvětlivce v šedém panelu, červená políčka symbolizují, že v uvedený den a hodinu probíhá nějaká akce. Čísla v políčku udávají počet akcí, které v danou dobu u zvolených položek probíhají. Na jednotlivá políčka lze kliknout pro bližší informace o akcích (Obrázek 15).

Tato část aplikace je vhodná v situaci, kdy chceme zjistit kdy má libovolná kombinace zvoleného vyučujícího, místnosti či oboru a jeho ročníku společně volný čas. K tomuto účelu je i upravená rozvrhová tabulka.

4. Zpět - Volba tohoto tlačítka vrátí uživatele do úvodní nabídky.



Obrázek 14: Vyhledávání - příklad



Obrázek 15: Vyhledávání - příklad, kliknutí

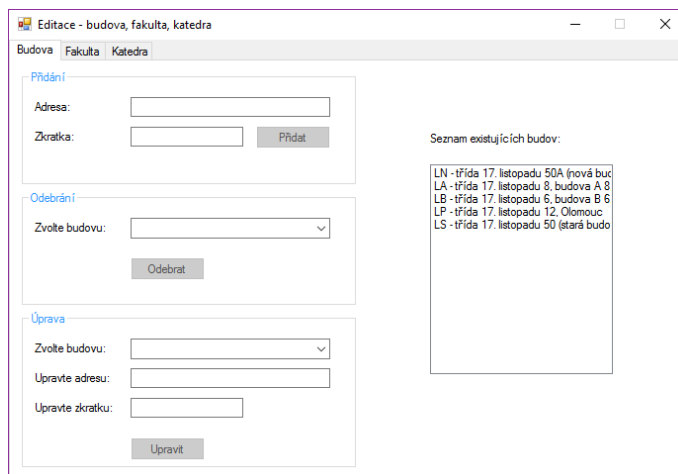
3.6 Úprava databáze

Obsahuje sedm tlačítek. Každé tlačítko kromě tlačítka "Zpět" otevře nové okno a umožňuje editaci vybrané části databáze. Základní správací funkce jsou vkládání dat, mazání či jejich úprava. Tato část neobsahuje správu pro akce (předměty a omezení), které jsou uloženy v databázi. Pro tuto úpravu slouží část pro samotné vkládání akcí, jak již bylo zmíněno.

- Budovy/Fakulty/Katedry - Obrázek 16. Slouží pro editaci budov, fakult

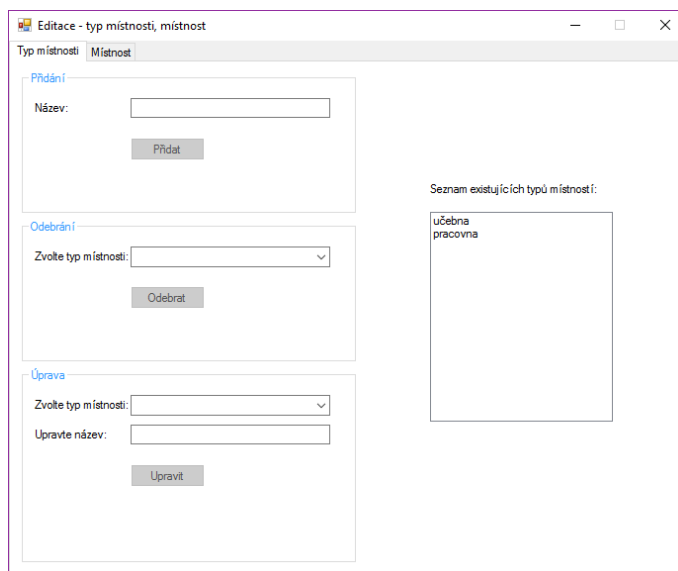
a kateder, které jsou součástí univerzity.

Okno obsahuje tři záložky - Budova, Fakulta a Katedra. Po kliknutí na libovolnou z nich se dostaneme do příslušné správy. V pravé části okna je k nahlédnutí seznam již uložených dat.



Obrázek 16: Editace - budova, fakulta, katedra

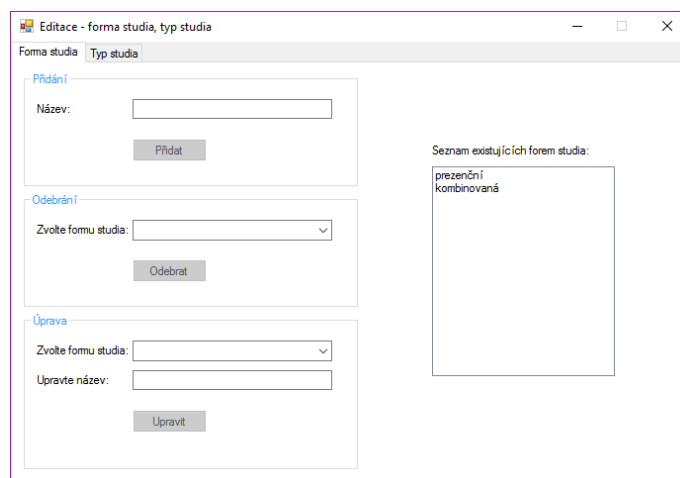
- Typ místnosti/Místnosti - Obrázek 17. Slouží pro editaci typů místností a místností. Okno obsahuje dvě záložky - Typ místnosti, Místnost. Po kliknutí na libovolnou z nich se dostaneme do příslušné správy. V pravé části okna je k nahlédnutí seznam již uložených dat pro lepší přehlednost.



Obrázek 17: Editace - typ místnosti, místnost

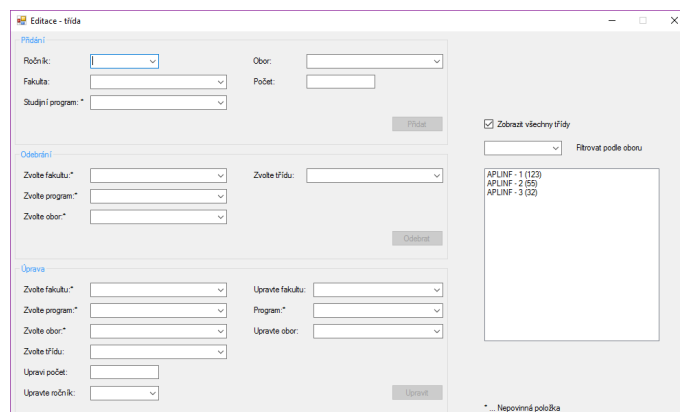
- Forma studia/Typ studia - Obrázek 18. Slouží pro editaci forem a typů studia. Okno obsahuje dvě záložky - Forma studia, Typ studia. Po kliknutí

na libovolnou z nich se dostaneme do příslušné správy. V pravé části okna je k nahlédnutí seznam již uložených dat pro lepší přehlednost.



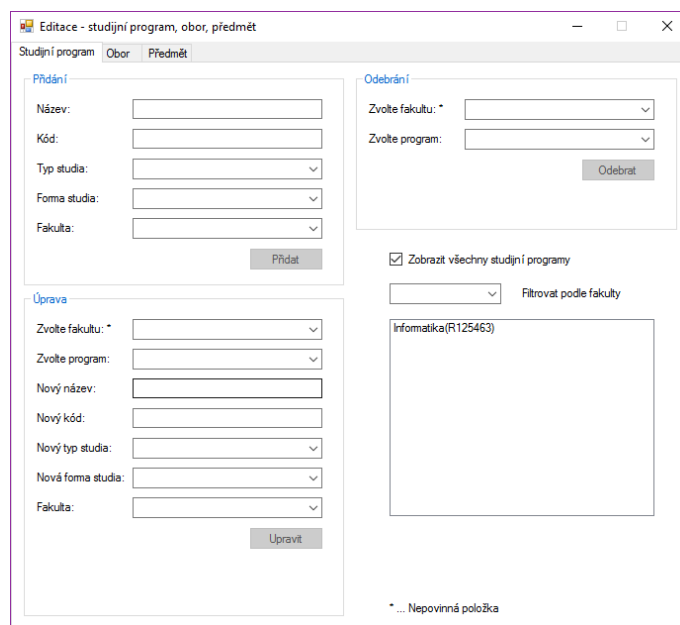
Obrázek 18: Editace - forma studia, typ studia

- Třída - Obrázek 19. Slouží pro editaci tříd. Jak již bylo zmíněno jde o spojení oboru s ročníkem plus přidání informace o počtu studentů. V levé části lze třídy přidávat, odebírat či upravovat a v pravé části okna je k nahlédnutí seznam již uložených tříd. Zde už může být více dat a mohlo by docházet k nepřehlednosti, proto je tento seznam opatřen i filtrací, kdy můžeme buďto nechat zobrazit všechny třídy a nebo jenom ty, podle zvoleného oboru.



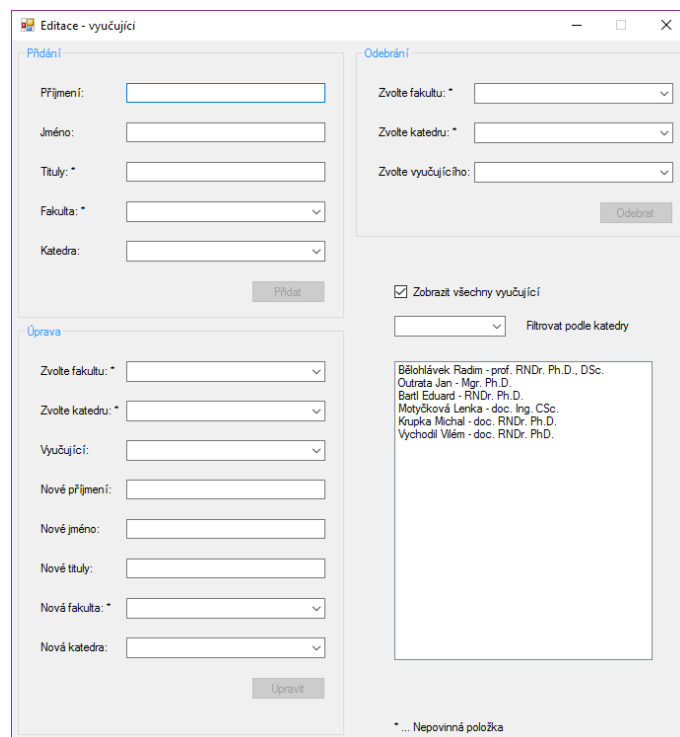
Obrázek 19: Editace - třída

- Programy/Obory/Předměty - Obrázek 20. Slouží pro editaci studijních programů, oborů a předmětů. Okno obsahuje tři záložky - Studijní program, Obor, Předmět. Po kliknutí na libovolnou z nich se dostaneme do příslušné správy. V pravé části okna je k nahlédnutí seznam již uložených dat opět doplněn filtrací ve všech třech záložkách.



Obrázek 20: Editace - studijní program, obor, předmět

- Vyučující - Obrázek 21. Slouží pro editaci vyučujících. Funkcionalita stejná jako v předchozích. Seznam již evidovaných vyučujících je opět opatřen filtrací.



Obrázek 21: Editace - vyučující

- Zpět - Volba tohoto tlačítka vrátí uživatele do úvodní nabídky (na úvodní stránku).

Závěr

Cílem této bakalářské práce bylo vytvoření aplikace pro tvorbu a editaci rozvrhů. Tato aplikace by měla ulehčit rozvrháři jeho činnost.

Program podporuje vkládání rozvrhových akcí a různých omezení. Během práce s aplikací lze průběžně zobrazovat vytvářené rozvrhy. Pomůckou pro usnadnění tvorby je možnost si zobrazit i další rozvrhy pro orientaci. Se zobrazenými rozvrhy můžeme dále manipulovat ve smyslu vkládání či mazání akcí. Dále aplikace splňuje požadavek na vyhledávání a vizualizaci společných volných času. Průnik společných časů se zobrazí po zadání položek, které si uživatel zvolí. Mimo jiné dokáže uživatel prostřednictvím aplikace editovat databázi, na které je aplikace postavena.

V budoucnu by bylo možné rozšířit funkcionalitu programu o schopnost vytvořit ke zvolenému předmětu a jeho typu (přednáška, cvičení nebo seminář) více akcí v návaznosti na hodinovou dotaci. Tato dotace by zůstala nadále stejná, ale akcí by bylo více.

Například cvičení bude mít celkovou dotaci 5 hodin týdně. S pomocí zmínované funkcionality by bylo možné rozdělit cvičení např. na dvě akce, z nichž jedno cvičení bude mít 3 hodiny a druhé 2 hodiny.

V konečném výsledku, ale půjde o jakousi skupinu akcí. V součtu budou odpovídat dotaci, kterou má daný předmět a typ akce přidělen.

Conclusions

The goal of this bachelor thesis was to create an application for creating and editing time tables. This application should ease work for time tables maker.

The program supports inserting time table actions and various restrictions. You can continuously view the generated schedules while working with the application. There is also a possibility to view any other schedule which could help with creating a new one. With displayed schedules we can also manipulate by inserting or deleting actions. Application also fulfils demand for searching and visualisation of common spare times. Intersection of common times is displayed after entering elements chosen by user. Among other things, via application the user can edit database on which the application is built.

In the future, there would be a possibility to extend the functionality of the program by creating more actions in relation to the time requirements of the subject and its type (lecture, exercise or seminar). Time requirements of the subject would be the same, but there would be more actions to use.

For example, the exercise will have a total time requirements 5 hours a week. Using the mentioned functionality, it would be possible to divide this exercise into two actions. One of them would take 3 hours and the other 2 hours.

In the end, it will be some group of action. Their sum will be equal with the total time requirements of the subject and its type.

Literatura

- [1] BRUCKER, Peter. Scheduling Algorithms. 5rd ed. Berlin: Springer, 2007. ISBN 978-3-540-69515-8.
- [2] BUTTAZZO, Giorgio C. Hard Real-Time Computing Systems: Predictable Scheduling Algorithms and Applications. 3rd ed. New York: Springer, 2011. ISBN 978-1-4614-0675-4.
- [3] Visual Studio 2013. Microsoft [online]. ©2017 [cit. 2017-05-17]. Dostupné z: [https://msdn.microsoft.com/cs-cz/library/dd831853\(v=vs.120\).aspx](https://msdn.microsoft.com/cs-cz/library/dd831853(v=vs.120).aspx)
- [4] TROELSEN, Andrew. C# a .NET 2.0 profesionálně. 3. vyd. Brno: ZONER software, 2006. ISBN 80-86815-42-0.
- [5] BEAULIEU, Alan. Learning SQL. 2rd ed. the United States of America: O'Reilly, 2009. ISBN 978-0-596-52083-0.
- [6] STROUSTRUP, Bjarne. The C ++ Programming Language. 4th ed. New Jersey: Addison-Wesley, 2013. ISBN 978-0321563842.
- [7] HEROUT, Pavel. Učebnice jazyka Java. 5. vyd. České Budějovice: Kopp, 2010. ISBN 9788072323982.
- [8] SCHAFER, Steven M. HTML, XHTML a CSS: Bible pro tvorbu WWW stránek. 4th ed. Praha: Grada, 2009. ISBN 978-80-247-2850-6.
- [9] JavaScript Tutorial. W3schools [online]. ©1999-2017 [cit. 2017-05-10]. Dostupné z: <https://www.w3schools.com/js/>
- [10] Databázový software a aplikace | Microsoft Access. Microsoft [online]. ©2017 [cit. 2017-05-10]. Dostupné z: <https://products.office.com/cs-cz/access>
- [11] Download Microsoft Access Database Engine 2010 Redistributable from Official Microsoft Download Center. Microsoft [online]. ©2017 [cit. 2017-05-10]. Dostupné z: <https://www.microsoft.com/en-us/download/details.aspx?id=13255>
- [12] SQL Tutorial. W3schools [online]. ©1999-2017 [cit. 2017-05-10]. Dostupné z: <https://www.w3schools.com/sql/>

A Obsah přiloženého CD/DVD

Na samotném konci textu práce je uveden stručný popis obsahu přiloženého CD/DVD, tj. jeho závazné adresářové struktury, důležitých souborů apod.

bin/

Obsahuje spustitelný soubor Timetabling.exe.

doc/

Obsahuje text práce ve formátu PDF a dále v archivu ZIP zdrojové texty.

src/

Obsahuje kompletní zdrojové texty programu TIMETABLING v archivu Timetabling.zip.

readme.txt

Obsahuje instrukce pro spuštění programu.

Navíc CD/DVD obsahuje:

data/

Obsahuje dvě verze databáze Timetabling.accdb, na kterou je program napojen.

install/

Obsahuje instalátor zprostředkovatele bezproblémového připojení k uvedené databázi.