

České vysoké učení technické v Praze  
Fakulta jaderná a fyzikálně inženýrská

Katedra softwarového inženýrství  
Obor: Aplikace softwarového inženýrství



# Webová aplikace pro správu rozvrhů hodin

## Web Application for Management of Timetables

BAKALÁŘSKÁ PRÁCE

Vypracoval: Michal Čaha  
Vedoucí práce: Mgr. Dana Majerová, Ph.D.  
Rok: 2021



# České vysoké učení technické v Praze

## Fakulta jaderná a fyzikálně inženýrská

Katedra softwarového inženýrství

Akademický rok 2019/2020

## ZADÁNÍ BAKALÁŘSKÉ PRÁCE

<b>Student:</b>	Michal Caha
<b>Studijní program:</b>	Aplikace přírodních věd
<b>Obor:</b>	Aplikace softwarového inženýrství
<b>Název práce česky:</b>	Webová aplikace pro správu rozvrhů hodin
<b>Název práce anglicky:</b>	Web Application for Management of Timetables

### Pokyny pro vypracování:

1. Prozkoumejte trh, popište hlavní rysy vybraných aplikací pro správu rozvrhu hodin a uveďte jejich výhody a nevýhody.
2. Proveďte analýzu problému a stanovte základní cíle webové aplikace pro správu rozvrhů hodin (např. jaké funkce bude aplikace nabízet, jaké role budou uživatelé mít).
3. Navrhněte vzhled webové aplikace tak, aby odpovídal jednotlivým funkcím a uživatelským rolím.
4. Stanovte integritní omezení pro data a adekvátně tomu navrhněte databázi (ERA diagram).
5. Zvolte vhodné technologie pro realizaci webové databázové aplikace.
6. Vytvořte webovou aplikaci pro správu rozvrhů hodin a ověřte její funkčnost vytvořením demonstračního rozvrhu hodin (jeden semestr na VŠ a 3 ročníky vybraného bakalářského studijního oboru, respektovat časové požadavky vyučujících).
7. Popište implementaci aplikace s využitím prostředků softwarového inženýrství.
8. Napište uživatelskou příručku (např. z pohledu jednotlivých uživatelských rolí) včetně softwarových požadavků a popisu instalace webové aplikace.

## Doporučená literatura:

- [1] *JavaScript Tutorial (w3schools)* [online]. Sandnes (Norsko): Refsnes Data, 2019 [cit. 2019-10-11]. Dostupné z: <https://www.w3schools.com/js/default.asp>.
- [2] WELLING, L. and THOMSON, L. *Mistrovství PHP a MySQL*. Překlad O. Baše. 1. vydání. Brno: Computer Press, 2017. ISBN 978-80-251-4892-1.
- [3] HUSEBY, S. H. *Zranitelný kód*. Brno: Computer Press, 2006. ISBN 80-251-1180-6.
- [4] aplikace pro generování rozvrhu hodin, například:
  - aSc Rozvrhy (<https://www.asctimetables.com/>),
  - BAKALÁŘI Rozvrh hodin (<https://www.bakalari.cz/Timetable/>),
  - dm Software, modul *Rozvrh a suplování* (<https://portal.dmssoftware.cz/>),
  - Docendo – School Scheduling software (<https://docendo.co>),
  - Timetabling Software For Schools – TimeTabler (<https://www.timetabler.com/>),
  - Untis Express 2019 (<http://www.school-timetabling.com/index.php?lang=CZ>).

## Jméno a pracoviště vedoucího práce:

Mgr. Dana Majerová, Ph.D.

Katedra softwarového inženýrství, Fakulta jaderná a fyzikálně inženýrská, ČVUT v Praze

## Jméno a pracoviště konzultanta:

–

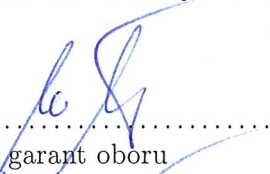


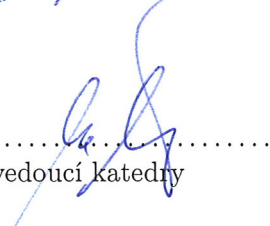
vedoucí práce

Datum zadání bakalářské práce: 11. 10. 2019

Termín odevzdání bakalářské práce: 7. 7. 2020

Doba platnosti zadání je dva roky od data zadání.

  
.....  
garant oboru

  
.....  
vedoucí katedry



  
.....  
děkan

V Praze dne 11. 10. 2019

## **Prohlášení**

Prohlašuji, že jsem svou bakalářskou práci vypracoval samostatně a použil jsem pouze podklady (literaturu, projekty, SW atd.) uvedené v příloženém seznamu.

V Děčíně dne .....

.....

Michal Čaha

## **Poděkování**

Děkuji Mgr. Daně Majerové, Ph.D za vedení mé bakalářské práce a za neocenitelné rady.

Michal Čaha

*Název práce:*

## **Webová aplikace pro správu rozvrhů hodin**

*Autor:* Michal Caha

*Studijní program:* Aplikace přírodních věd

*Obor:* Aplikace softwarového inženýrství

*Druh práce:* Bakalářská práce

*Vedoucí práce:* Mgr. Dana Majerová, Ph.D.

Katedra softwarového inženýrství, Fakulta jaderná a fyzikálně inženýrská, České vysoké učení technické v Praze

*Konzultant:* –

*Abstrakt:* Cílem této bakalářské práce je navrhnout a implementovat webovou aplikaci pro tvorbu rozvrhu pro školu, která umožní uživateli sestavit rozvrh a zobrazit rozvrh na webové stránce. Text bakalářské práce obsahuje nejprve porovnání existujících aplikací pro tvorbu rozvrhu. Dále je v práci popsán návrh databáze a webové aplikace, použité technologie a implementace vybraných funkcí systému. Příloha práce je věnována uživatelské příručce včetně popisu instalace.

*Klíčová slova:* Rozvrh hodin, webová aplikace, databáze

*Title:*

## **Web Application for Management of Timetables**

*Author:* Michal Caha

*Abstract:* The aim of this bachelor thesis is to create a draft and implementation of a web application that allows its users to create and view school timetables on a web page. The thesis compares already existing tools for creating school timetables, proposes a draft of a database, web application, and lists technologies together with the implementation of selected system features. The user manual and installation process are enclosed with the thesis.

*Key words:* Timetable, Web application, database





# Obsah

<b>Úvod</b>	<b>11</b>
<b>1 Přehled vybraných existujících aplikací</b>	<b>13</b>
1.1 UniTime . . . . .	13
1.2 B-INTIME . . . . .	14
1.3 aSc Timetables . . . . .	14
1.4 Skolaris . . . . .	15
1.5 Porovnání vybraných programů . . . . .	16
<b>2 Použité technologie</b>	<b>17</b>
2.1 HTML . . . . .	17
2.2 CSS . . . . .	17
2.3 PHP . . . . .	17
2.4 Fat-Free Framework . . . . .	18
2.5 Firebird . . . . .	18
2.6 PhpStorm . . . . .	19
2.7 Bootstrap . . . . .	19
2.8 AJAX . . . . .	19
<b>3 Implementace webové aplikace pro tvorbu rozvrhu hodin</b>	<b>21</b>
3.1 Databáze . . . . .	21
3.1.1 Integritní omezení a popis tabulek . . . . .	21
3.1.2 Domény . . . . .	24
3.1.3 Pohledy . . . . .	24
3.1.4 ERA diagram databáze . . . . .	25
3.2 Implementace webové aplikace . . . . .	25
3.2.1 Fat-Free Framework . . . . .	25
3.2.2 Třídy a jejich funkce . . . . .	27
3.2.3 Diagram tříd webové aplikace . . . . .	34
3.3 JavaScript . . . . .	34
3.3.1 AJAX . . . . .	34
3.3.2 Diagram Javascriptu . . . . .	34
<b>Závěr</b>	<b>37</b>
<b>Literatura</b>	<b>38</b>
<b>Přílohy</b>	<b>41</b>

<b>A</b>	<b>Uživatelská příručka</b>	<b>41</b>
A.1	Popis instalace . . . . .	41
A.1.1	Instalace databáze . . . . .	41
A.1.2	Instalace webové aplikace . . . . .	41
A.2	Uživatelská příručka pro nepřihlášeného uživatele (student) . . . . .	41
A.3	Uživatelská příručka pro přihlášeného uživatele bez administrátorských práv (učitel) . . . . .	42
A.4	Uživatelská příručka pro přihlášeného uživatele s administrátorskými právy (administrátor) . . . . .	42
A.4.1	Uživatelé . . . . .	43
A.4.2	Akademické roky . . . . .	43
A.4.3	Typ učebny . . . . .	43
A.4.4	Typ výuky . . . . .	43
A.4.5	Učebna . . . . .	43
A.4.6	Předmět . . . . .	43
A.4.7	Studium . . . . .	44
A.4.8	Kruh . . . . .	44
A.4.9	Semestr . . . . .	44
A.4.10	Bílá kniha . . . . .	44
A.4.11	Skupina . . . . .	45
A.4.12	Rozvrh . . . . .	45
<b>B</b>	<b>Obsah CD</b>	<b>47</b>

# Úvod

Během svého studia jsem se setkal s mnoha rozvrhy hodin, ať už šlo rozvrhy školní nebo rozvrh třídní služby. Vybral jsem si tuto bakalářskou práci, protože si myslím, že na trhu není mnoho programů na tvorbu rozvrhů ve webové aplikaci.

V první kapitole je uveden průzkum trhu. Našel jsem na internetu čtyři aplikace pro tvorbu rozvrhů hodin, které jsem stručně popsal a porovnal mezi sebou.

Ve druhé kapitole jsou uvedeny technologie, které jsem použil pro tuto bakalářskou práci. Jelikož se jedná o webovou aplikaci, tak základem jsou: značkovací jazyk HTML, skriptovací jazyk PHP a databázový systém Firebird.

Ve třetí kapitole je uveden návrh aplikace, popis databáze a webové aplikace.

V závěru jsou shrnuty funkce vytvořené aplikace.

V příloze je uživatelská příručka pro všechny uživatele.



# Kapitola 1

## Přehled vybraných existujících aplikací

V této části jsou stručně popsány čtyři aplikace, které řeší problematiku vytváření rozvrhu hodin pro školy a jsou dostupné na internetu a porovnal je. Vybral jsem si aplikace UniTime, B-INTIME, aSc Timetables, Skolaris.

### 1.1 UniTime



Obrázek 1.1: Logo webové aplikace UniTime

Zpracováno na základě [9].

UniTime (obrázek 1.1) je komplexní webový systém, který podporuje vývoj rozvrhů, kurzů a zkoušek, správu změn rozvrhů, sdílení místností a plánování studentů do jednotlivých tříd. Jedná se o distribuovaný systém, který umožňuje více manažerům univerzitních a resortních rozvrhů koordinovat sestavení a úpravu rozvrhu, který odpovídá jejich různorodým organizačním potřebám.

Lze jej použít samostatně k vytvoření a udržování školního rozvrhu hodin či zkoušek, nebo k propojení s existujícím informačním systémem pro studenty. Systém byl vyvinut za pomoci studentů a zaměstnanců univerzit v Severní Americe a Evropě.

Tento software je distribuován zdarma na základě licence s otevřeným zdrojovým kódem v naději, že jiné vysoké školy a univerzity mohou svým studentům prospět k lepšímu plánování nebo chtějí přispět k probíhajícímu výzkumu v této oblasti.

## 1.2 B-INTIME



Obrázek 1.2: Logo aplikace B-INTIME

Zpracováno na základě [3].

B-INTIME (obrázek 1.2) je desktopová aplikace na tvorbu rozvrhů hodin, která se používá ve školském webovém informačním systému Bakaláři. Nedá se pořídít samostatně bez systému Bakaláři.

Způsob vytváření rozvrhu je pomocí lístků, které se vytvoří automaticky, na základě údajů zadaných v aplikaci. Lístky (v grafické podobě) lze nasazovat do rozvrhu, přesouvat na jiné místo rozvrhu, vyměňovat s jinými. To vše za pohotového barevného označování hodin, v nichž by nasazením lístku vznikla kolize. Hlídána je i vzájemná kolize vyučovacích skupin.

Je zde možnost blokovat hodiny ve všech rozvrzích (například učebna rezervovaná pro jinou školu, studijní volno nebo jiný důvod nepřítomnosti učitele apod.). Nespornou výhodou počítačového zpracování rozvrhu je, vedle zmíněného hlídání vzniku kolizí, možnost tisku rozvrhů tříd, učitelů i místností. Lze tisknout i přehled rozvrhu celé školy a další sestavy. Výstupy rozvrhů lze směřovat i do souborů formátu HTML.

Na monitoru můžeme současně zobrazit více rozvrhů. Při nasazování hodin tělesné výchovy lze například sledovat obsazení tělocvičen a rozvrhy tělocvikářů. Lístky lze nasazovat jak v rozvrhu třídy, tak v rozvrhu učitele. Při využití generátoru může dále program hlídat volno na oběd pro žáky i učitele, nasazení hodin do vyhovujících učeben a podobně.

## 1.3 aSc Timetables



Obrázek 1.3: Logo aplikace aSc Timetables

Zpracováno na základě [2].

ASc Timetables (obrázek 1.3) je desktopová aplikace, která je jedna z nejlepších placených aplikací na tvorbu rozvrhů hodin. Ceny jsou uvedeny na obrázku 1.4 na straně 15. Aplikace umožňuje velmi snadné a rychlé zadání dat potřebných pro tvorbu rozvrhu (ročníků, učeben, dělení ročníků atd.). Vytvoří rozvrh automaticky.

Hotový rozvrh lze vytisknout nebo lze uložit ve formátu xls, PDF, HTML nebo ho lze sdílet on-line. Aplikace je již na trhu používána a vylepšována po dobu 20 let. Lze i stáhnout zkušební verzi.

	Základní školy Výhodná cena pro základní školy	Standard Neomezené možnosti rozvrhování pro školy	Premium Náš tým Vám s vytvořením rozvrhu pomůže	PRO Individuální rozvrhy dle potřeb každého studenta
	Kč 4400 one time payment	Kč 6000 one time payment	Kč 11900 one time payment	Kč 23900 one time payment
	OBJEDNAT	OBJEDNAT	OBJEDNAT	OBJEDNAT
Automatický návrh rozvrhu	✓	✓	✓	✓
Multilicence pro všechny počítače ve škole	✓	✓	✓	✓
aSc Suplování	✓	✓	✓	✓
Mobilní rozvrhy Studenti a učitelé mají přístup k aktuálním rozvrhům přes telefon či tablet	✓	✓	✓	✓
Neomezená podpora Podpora a aktualizace po dobu dvou let	✓ after 2nd year: Kč 1250	✓ after 2nd year: Kč 1250	✓ after 2nd year: Kč 1250	✓ after 2nd year: Kč 1250
Podpora Premium Analýza, testování, kontrola a návrhy změn rozvrhů			✓	✓
Automatický návrh pro studenty Vytváření individuálních rozvrhů pro studenty				✓
	OBJEDNAT	OBJEDNAT	OBJEDNAT	OBJEDNAT

+ Koupit pouze doplňky

Obrázek 1.4: Ceny ASC Timetables

## 1.4 Skolaris

Zpracováno na základě [8].

Skolaris (obrázek 1.5) je placený webový nástroj pro tvorbu rozvrhů a správu suplování pro základní a střední školy. Umí vytvořit rozvrh od začátku i optimalizovat



Obrázek 1.5: Logo webové aplikace Skolaris

rozvrh, který již existuje. Je plně nastavitelný a poskytuje kontrolu vážených podmínek. Aplikace umožňuje víceuživatelský přístup s ochranou dat. Je podporována ve všech webových prohlížečích na operačních systémech Linux, MacOS a Windows.

## 1.5 Porovnání vybraných programů

V tabulce 1.1 je porovnání aplikací, které jsem si vybral. Z hlediska kritérií, která jsou:

- je-li aplikace placená,
- možné výstupy,
- generuje rozvrh automaticky,
- je-li možnost ruční úpravy.

Název	Placená	Generuje automaticky	Výstup	ruční úprava
B-INTIME	ANO	NE	XLS, HTML	ANO
aSc timetables	ANO	ANO	XLS, HTML, PDF	ANO
Skolaris	ANO	ANO	XLS, HTML	ANO
UniTime	NE	NE	HTML	ANO

Tabulka 1.1: Porovnání aplikací

Z této tabulky vidíme, že aSc timetables a Skolaris jsou skoro stejné, akorát aSc Timetables má více výstupů. B-INTIME si nelze koupit bez systému Bakaláři. Pokud nechcete platit za tvorbu rozvrhu, tak doporučuji UniTime.



# Kapitola 2

## Použité technologie

V této části se budu zabývat technologiemi, které jsem použil při tvorbě webové aplikace a databáze.

### 2.1 HTML

Zpracováno na základě [10].

HTML je zkratkou pro HyperText Markup Language, patří mezi značkovací jazyky, které se využívají pro vytvoření kostry webových stránek. Slovo HyperText znamená možnost vzájemně propojovat texty pomocí odkazů, Markup znamená schopnost jazyka HTML dávat významy jednotlivým blokům textu s pomocí speciálních značek nazývaných elementy či tagy.

Jazyk HTML patří do široké rodiny značkovacích jazyků SGML. Vznikl v roce 1990 ve Švýcarsku a postupně se vyvíjel až k současné verzi HTML5.

### 2.2 CSS

Zpracováno na základě [11].

CSS (Cascading Style Sheets), známé také jako kaskádové styly jsou moderním jazykem umožňujícím účinně naformátovat stránky psané ve značkovacích jazycích HTML, XHTML nebo XML.

### 2.3 PHP

Zpracováno na základě [12].

PHP je jedním z nejvíce rozšířených programovacích jazyků používaných k vytváření webových aplikací. PHP se používá na straně serveru a slouží tedy ke generování HTML kódu stránky, jenž pak server odesílá do prohlížeče (na rozdíl od klientského JavaScriptu, který funguje až při zobrazení stránky v prohlížeči).

Hlavní výhodou PHP je jeho nezávislost na platformě (Windows, Linux, Unix...), mezi výhody patří také široké možnosti použití. PHP umí například pracovat se soubory a s mnoha různými databázemi, generovat a upravovat grafiku, umí odesílat a přijímat emaily, vytvářet PDF a mnoho dalších funkcí.

PHP má poměrně volnou syntaxi (způsob zápisu), snadno se učí, zejména pokud již máte zkušenosti s jinými programovacími jazyky.

## 2.4 Fat-Free Framework

Zpracováno na základě [5].

Fat-Free Framework je open-source webový framework, který je určený především pro programování dynamických webových stránek a webových aplikací například ve formátu HTML, XHTML či WML. Při použití PHP pro dynamické stránky jsou skripty prováděny na straně serveru tj. k uživateli je přenášen až výsledek jejich činnosti. Má také objektově-relační mapovač pro abstrakci a modelování dat. Není potřeba žádná konfigurace.

Fat-Free Framework vás také chrání před spamem a DoS útoky prováděním kontrol DNSBL. Může zvýšit provozuschopnost serveru tím, že řídí provoz webového serveru pomocí analýzy profilu a omezení šířky pásma.

## 2.5 Firebird

Zpracováno na základě [6].

Firebird je implementovaná relační databáze, která je multiplatformní (běží na různých unixových systémech včetně Linuxu a MS Windows). Firebird vznikl v roce 2000 jako část z open source databáze InterBase od společnosti Borland.

Firebird jsem si vybral kvůli jeho vlastnostem, které jsou například:

- plná podpora pro uložené procedury a Triggery (spouště),
- plně kompatibilní transakce ACID,
- referenční integrita,
- multigenerační architektura (MVCC),
- podpora externích uživatelsky definovaných funkcí (UDF),

- nástroje třetích stran včetně grafických administračních nástrojů a replikačních nástrojů,
- mnoho přístupových metod jako jsou nativní API, dbExpress ovladače, ODBC, OLEDB, .NET, JDBC ovladač verze 4, modul pro Python, PHP, Perl,
- inkrementální zálohy,
- plná implementace kurzoru pro PSQL.

## 2.6 PhpStorm

Zpracováno na základě [7].

PhpStorm je placené multi-platformní vývojové prostředí (IDE) pro PHP vyvíjený českou společností JetBrains. Celou svoji webovou část aplikace jsem vytvořil v tomto prostředí, kvůli jeho vlastnostem jako jsou například:

- našeptávač doplňuje třídy, metody, jména proměnných, PHP výrazy,
- PHP UML (generování UML diagramů tříd),
- systém sledování chyb,
- SSH přístup na vzdálené servery,
- A mnoho dalších.

## 2.7 Bootstrap

Zpracováno na základě [4].

Bootstrap je open-source sada nástrojů kaskádových stylů pro tvorbu webu a webových aplikací. Obsahuje návrhářské šablony založené na HTML a CSS, sloužící pro úpravu typografie, formulářů, tlačítek, navigace a dalších komponent rozhraní, stejně jako další volitelná rozšíření programovacího jazyka JavaScriptu.

## 2.8 AJAX

Zpracováno na základě [1].

AJAX (Asynchronous JavaScript and XML) je technologie vyvíjející interaktivní webové aplikace, které mění obsah svých stránek bez nutnosti jejich kompletního znovu načítání za pomoci asynchronního zpracování webových stránek pomocí knihovny napsané v JavaScriptu. Na rozdíl od klasických webových aplikací poskytují uživatelsky příjemnější prostředí, ale vyžadují použití moderních webových prohlížečů.

Tyto aplikace jsou vyvíjeny s využitím technologií:

- HTML (nebo XHTML) a CSS pro prezentaci informací,
- DOM a JavaScript pro zobrazování a dynamické změny prezentovaných informací,
- XMLHttpRequest pro asynchronní výměnu dat s webovým serverem (typicky je užíván formát XML, ale je možné použít libovolný jiný formát včetně HTML, prostého textu, JSON či EBML).

# Kapitola 3

## Implementace webové aplikace pro tvorbu rozvrhu hodin

Rozhodl jsem se aplikaci pro tvorbu rozvrhu hodin vytvořit jako webovou. Rozvrh hodin potřebuje ukládat mnoho dat, proto se nejprve budu zabývat databází.

### 3.1 Databáze

#### 3.1.1 Integritní omezení a popis tabulek

Před vytvořením návrhu databáze je nutné stanovit integritní omezení, která budou muset dodržovat data a která budou kontrolována databázovým systémem. Některá integritní omezení bude potřeba kontrolovat pomocí aplikace.

V následujícím textu se budu věnovat entitám, které bude moje aplikace používat.

- Den

Evidujeme název dne a zkratku, bude předvyplněno, pět pracovních dní (od pondělí do pátku).

Den je použitý v 0-N časových omezeních učitelů.

Den je použitý v 0-N rozvrzích.

- Hodina

Evidujeme čas začátku a čas konce vyučovací hodiny. Bude předvyplněno osm hodin (8:00-16:30).

Hodina je použita v 0-N časových omezeních.

Hodina je použita v 0-N rozvrzích.

- Typ semestru

Evidujeme název a zkratku typu semestru. Budou předvyplněny dva typy semestrů (zimní semestr a letní semestr).

Jeden typ semestru je přiřazen 0-N semestrům.

Jeden typ semestru je přiřazen 0-N předmětům.

- Semestr

Evidujeme typ semestru (z tabulky *typ semestru*), kód semestru, akademický rok, jestli je aktivní (tzn. zda je použit pro tvorbu rozvrhu), kdy začínají časová omezení (od kdy může uživatel zadávat časové omezení), kdy končí časová omezení, od kdy je výuka v semestru a kdy končí.

Semestr je použitý v 0-N skupinách.

Semestr je použitý pro 0-N časových omezení.

Semestr je použitý v 0-N studijních plánech.

- Akademický rok

Evidujeme rok od a rok od.

Je přiřazen 0-N (přesněji 0-2) semestrům.

Je přiřazen 0-N kruhům.

- Předmět

Evidujeme název předmětu, kód předmětu, typ semestru (z tabulky *typ semestru*) a zakončení.

Jeden předmět je přiřazen 0-N skupinám.

Jeden předmět je přiřazen 0-N studijním plánům.

- Typ výuky

Evidujeme název a zkratku typu výuky (např. přednáška, cvičení).

Jeden typ výuky je použit v 0-N rozpisech.

Jeden typ výuky je použit v 0-N skupinách.

- Učitel

Evidujeme jméno, příjmení, titul před jménem, titul za jménem, e-mail, zakódované heslo (Hash256) a jestli má administrátorská práva.

Jeden učitel je přiřazen 0-N předmětům (přesněji: ke konkrétnímu typu nějakého předmětu ve studijním plánu).

Jeden učitel zadává 0-N časových omezení (pro 1 semestr).

- Učebna

Evidujeme typ učebny (z tabulky *typ učebny*), označení a kapacitu.

V učebně se vyučuje 0-N předmětů (rozvrh).

- Studium

Evidujeme název a zkratku studia.

Studium má přiřazeno 0-N kruhů (ročníků).

- Kruh

Kruh je konkrétní ročník konkrétního studijního směru (studia), evidujeme studium (z tabulky *studium*), akademický rok (z tabulky *akademický rok*), zkratku a ročník.

Jeden kruh je zařazen do 0-N skupin (pro různé předměty).

Jeden kruh má svůj studijní plán pro daný semestr (ve studijním plánu je zařazeno 0-N předmětů s maximálně dvěma typy výuky).

- Časové omezení

Časové omezení uchovává dny a časy, kdy nemůže daný učitel v daném semestru učit.

Evidujeme učitele (z tabulky *učitel*), den (z tabulky *den*), hodinu (z tabulky *hodina*) a semestr (z tabulky *semestr*).

- Studijní plán

Evidujeme kruh (z tabulky *kruh*), semestr (z tabulky *semestr*), předmět (z tabulky *předmět*) a jestli je předmět povinný nebo volitelný.

Studijní plán obsahuje 0-N rozpisů (přesněji 1-2).

- Rozpis

Evidujeme studijní plán (z tabulky *studijní plán*, typ výuky (z tabulky *typ výuky*), učitele (z tabulky *učitel*), semestr (z tabulky *semestr*), počet hodin týdně a typ učebny (z tabulky *typ učebny*).

- Skupina

Skupina spojuje nebo rozděluje dané studium v rámci jednotlivých rozpisů (předmětů).

Evidujeme předmět (z tabulky *předmět*), semestr (z tabulky *semestr*), typ výuky (z tabulky *typ výuky*), zda je předmět vyučován týdně nebo ob týden a kolik hodin má pro týden (pokud je ob týden, tak má dvojnásobný počet hodin).

Skupina obsahuje 0-N kruhů (realizováno za pomoci tabulky zařazení).

Skupina je v rozvrhu 0-N krát (podle počtu hodin v týdnu).

- Zařazení

Dekomponuje vztah M:N mezi tabulkami *kruh* a *skupina*.

Evidujeme kruh (z tabulky *kruh*), skupinu (z tabulky *skupina*) a jestli je předmět pro daný kruh povinný nebo volitelný.

- Rozvrh

Evidujeme skupinu (z tabulky *skupina*), den (z tabulky *den*), hodinu (z tabulky *hodina*), učitele (z tabulky *učitel*), učebnu (z tabulky *učebna*) a poznámku.

### 3.1.2 Domény

Pro databázi jsem si vytvořil své vlastní domény, které lépe odpovídají požadavkům na data defaultní datové typy:

- *identifikace* je celé číslo, které nesmí být menší než 0. Používá se pro umělé primární klíče.
- *Text5* je text o maximálně 5 znacích v UTF8, nejčastěji používán pro zkratky.
- *Text10* je text o maximálně 10 znacích v UTF8, který musí být povinně vyplněn a nemá mezeru před textem a za textem. Používám nejčastěji pro krátké názvy a zakončení.
- *Text10n* je text o maximálně 10 znacích v UTF8, který může být prázdný. Používám pro titul před jménem a titul za jménem.
- *Text30* je text o maximálně 30 znacích v UTF8, který musí být povinně vyplněn a nemá mezeru před textem a za textem. Používám nejčastěji pro jméno, příjmení atd.
- *Text60* je text o maximálně 60 znacích v UTF8, který musí být povinně vyplněn a nemá mezeru před textem a za textem. Používám jej nejčastěji pro delší názvy atd.
- *Text120* je text o maximálně 120 znacích v UTF8, který musí být povinně vyplněn a nemá mezeru před textem a za textem. Používán pro e-mail.
- *Text120n* je text o maximálně 120 znacích v UTF8, který musí být povinně vyplněn. Používán pro poznámku.
- *Heslohash* je text o maximálně 60 znacích v UTF8, který musí být povinně vyplněn, nemá mezeru před textem a za textem. Používám speciálně pro heslo, které je zakódované.
- *Datum* je povinně zadávaný údaj typu datum. Používám pro všechny entity, které obsahují datum (např. datum ukončení semestru).
- *Bool* je číslo, které může být jenom 0 nebo 1 a který musí být povinně vyplněn. Používám pro entity, které jsou ano/ne jako např. vyučováno týdně (ano/ne).

### 3.1.3 Pohledy

Pro lepší zobrazení dat z databáze jsem si v databázi vytvořil pohledy (views). Vzhledem k potřebě složitějších pohledů v průběhu mé práce jsou některé pohledy podobné.

- *V\_semestry* spojuje tabulky semestr, typ semestru a akademický rok.



- V\_ucebny spojuje tabulku *učebna* a *typ učebny*.
- V\_predmety spojuje tabulky *předmět* a *typ semestru*.
- V\_kruhy spojuje tabulky *kruh* a *studium*.
- V\_bilakniha spojuje tabulky *rozpis*, *studijní plán*, *typ výuky*, *předmět*, *studium* a *kruh*.
- V\_ucebna\_obsazeno spojuje tabulky *rozvrh* a *učebna*.
- V\_kruh\_obsazeno spojuje tabulky *rozvrh*, *skupina* a *zařazení*.
- \_ucitel\_obsazeno spojuje tabulky *rozvrh*, *skupina*, *učitel* a *časové omezení*.
- V\_rozvrhcelek spojuje tabulky *rozvrh*, *učitel*, *učebna*, *skupina*, *předmět*, *zařazení*.
- V\_skupinapridej spojuje tabulky *rozpis*, *studijní plán*, *typ výuky*, *předmět*, *kruh*, *akademický rok*, *studium*.
- V\_prazdny\_rozvrh obsahuje kartézský součin tabulek *den* a *hodina*.

### 3.1.4 ERA diagram databáze

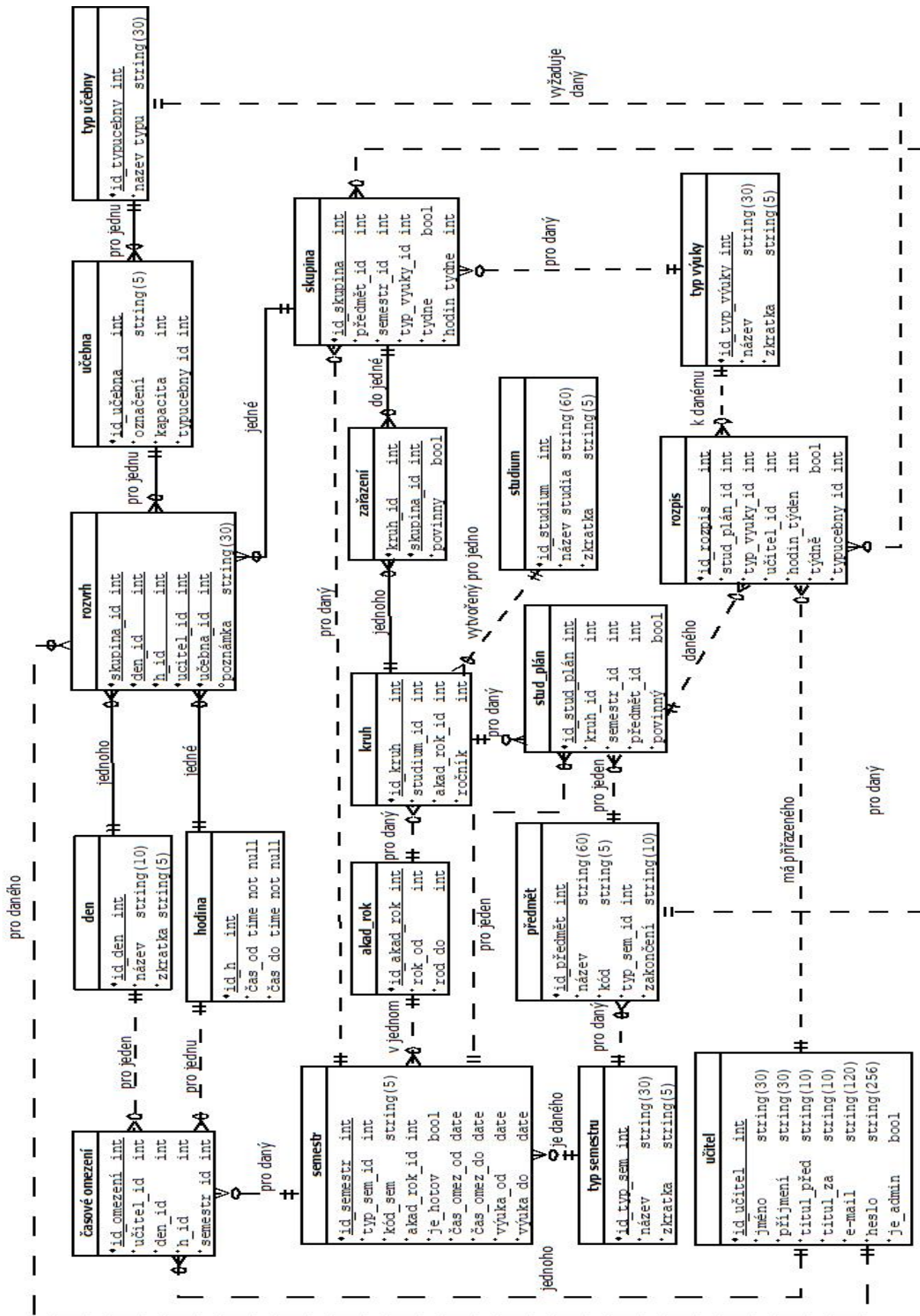
Na straně 26 je ERA diagram databáze, která obsahuje celkem 18 tabulek. Využil jsem k návrhu ERA modelu databáze program Dia, který je pro vytváření technických diagramů.

## 3.2 Implementace webové aplikace

Webová aplikace pro tvorbu rozvrhu hodin využívá jazyk PHP. Aby se mi rychleji pracovalo, použil jsem Fat-Free Framework. Pro práci s databází je využíváno rozhraní PDO a příkazy jazyka SQL. Aplikace funguje na principu Model-view-controller (MVC). Pro lepší vizuální zobrazení (například tabulek) je v aplikaci zahrnut také Bootstrap.

### 3.2.1 Fat-Free Framework

Webová aplikace používá Fat-Free Framework (*f3*). Ve složce *app* je složka *ui*, ve které jsou všechny HTML soubory, které tento framework překládá přes *\_layout.html*. Fat-Free Framework také vyžaduje také na webu zapnutý mod *\_rewrite* na serveru Apache a správně nastavený soubor *Htaccess*. Ve složce *app* je soubor *config.ini*, který obsahuje nastavení aplikace pro Fat-Free Framework, například cestu adresáře pro UI, autoload a spoustu dalších. V souboru *routes.ini* jsou uloženy tzv. *routy*, které se aktivují pro přesměrování. Pokud jdete přes *routy* na jakoukoliv metodu, která je *Změň* nebo *Smaž*, tak jde přes *routu s parametrem* (název parametru je všude *datazpet*).



Obrázek 3.1: ERA diagram databáze

## 3.2.2 Třídy a jejich funkce

U některých metod, které připravují formulář je závorce uveden soubor z UI, který F3 vkládá do `_layout.html`.

### Třída `timetable`

Třída `timetable` je pro nepřihlášeného uživatele.

Její metody jsou:

- `afterroute` je metoda, která sestaví výslednou webovou stránku.
- `__construct` je metoda, která se připojí k databázi.
- `prihlaseniform` je metoda, která připraví formulář k přihlášení uživatele (`prihlaseni.html`).
- `prihlasenfce` je metoda, která obsluhuje a zkontroluje formulář k přihlášení.
- `odhlaseni` je metoda, která odhlásí uživatele.
- `semestryprehled` je metoda, která vrátí data k otevřenému semestru.
- `rozvrhyuiform` je metoda, která zobrazí rozvrhy (`rozvrhy.html`).

### Třída `ucitel`

`Ucitel` je třída pro přihlášeného uživatele bez administrátorských práv (učitel) a která dědí od třídy `timetable`.

Její metody jsou:

- `beforeroute` je metoda, která zkontroluje, zda je uživatel přihlášený jako učitel.
- `__construct` metoda, která připojí třídu k databázi.
- `casomezproucform` je metoda, která zobrazí formulář časových požadavků pro učitele (`casomezprouc.html`).
- `casomezproucfce` je metoda, která kontroluje data z formuláře časových požadavků pro učitele.
- `tabulkaproucform` je metoda, která zobrazí rozvrh pro daného učitele (`tabulkaprouc.html`).
- `ucitelzmenaheslaform` je metoda, která připraví formulář změny hesla pro učitele (`ucitelzmenahesla.html`).
- `ucitelzmenaheslafce` je metoda, která obsluhuje formulář změny hesla pro učitele.

- `uvodprouc` je metoda, která zobrazí úvodní stránku pro učitele (*ucitelhome.html*).
- `casomezano` je metoda, která kontroluje, jestli se mohou přidávat časová omezení.

## Třída `admin`

`Admin` je třída, která dědí od třídy `timetable`. Obsahuje nejvíce metod a proto je rozdělena podle činností:

### 1. základní metody

- `beforeroute` je metoda, která zkontroluje, zda je uživatel přihlášený jako administrátor.
- `__construct` je metoda, která připojí třídu k databázi.
- `adminhome` je metoda, která zobrazí úvodní stránku pro administrátora (*adminhome.html*).
- `tabulky` je obecná metoda, která podle svého vstupu zobrazí data z databáze do tabulky (*vypistabulky.html*).

### 2. správa učitelů

- `ucitelpridejform` je metoda, která zobrazí formulář na přidání učitele (*ucitelpridej.html*).
- `ucitelpridejfce` je metoda, která provádí obsluhu formuláře při přidání učitele.
- `ucitelzmenform` je metoda, která zobrazí formulář pro změnu učitele (*ucitelzmen.html*).
- `ucitelzmenfce` je metoda, která provádí obsluhu formuláře při změně dat pro učitele.
- `ucitelsmazform` je metoda, která zobrazí formulář pro smazání učitele (*ucitelsmaz.html*).
- `ucitelsmazfce` je metoda, která provádí obsluhu formuláře při smazání učitele.
- `ucitelkontrola` je privátní metoda, která kontroluje, jestli daná metoda splňuje dané požadavky s integritními omezeními do databáze. Kontroluje metody `ucitelpridejfce`, `ucitelzmenfce` a `zmenaheslafce`.
- `zmenaheslaform` je metoda, která zobrazí formulář pro změnu hesla z pozice administrátora (*zmenahesla.html*).
- `zmenaheslafce` je metoda, která provádí obsluhu formuláře při změně hesla z pozice administrátora.

### 3. správa učeben

- `ucebnapridejform` je metoda, která zobrazí formulář na přidání učebny (*ucebnapridej.html*).
- `ucebnapridejfce` je metoda, která provádí obsluhu formuláře při přidání učebny.
- `ucebnakontrola` je privátní metoda, která kontroluje, jestli daná metoda splňuje dané požadavky s integritními omezeními do databáze. Kontroluje metody `ucebnapridejfce` a `ucebnazmenfce`.
- `ucebnazmenform` je metoda, která zobrazí formulář pro změnu učebny (*ucebnazmen.html*).
- `ucebnazmenfce` je metoda, která provádí obsluhu formuláře při změně dat pro učebnu.
- `ucebnasmazform` je metoda, která zobrazí formulář pro smazání učebny (*ucitelhome.html*).
- `ucebnasmazfce` je metoda, která provádí obsluhu formuláře při smazání učebny.

#### 4. správa rozvrhu

- `kontrolarozvrhu` je privátní metoda, která kontroluje, jestli daná metoda splňuje dané požadavky s integritními omezeními do databáze. Kontroluje metodu `rozvrhpridejfce`.
- `rozvrhpridejform` je veřejná metoda, která zobrazí formulář na přidání hodin do rozvrhu (*rozvrhpridej.html*).
- `rozvrhpridejfce` je veřejná metoda, která provádí obsluhu formuláře při přidání hodiny do rozvrhu.
- `volnoodeber` je chráněná metoda pro odčítání hodin, která jsou obsazená.
- `predmetzesemestrakruhu` je chráněná metoda, která vrátí předmět z daného semestru.
- `nacticasy` je chráněná metoda, která vrátí hodiny z db (pro funkci `volnoodeber`).
- `rozvrhsmazform` je metoda, která zobrazí formulář pro smazání dat v rozvrhu (*rozvrhsmaz.html*).
- `rozvrhsmazfce` je metoda, která provádí obsluhu formuláře při smazání hodin v rozvrhu.

#### 5. správa akademického roku

- `akadrokpridejform` je metoda, která zobrazí formulář na přidání akademického roku (*akadrokpridej.html*).
- `akadrokpridejfce` je metoda, která provádí obsluhu formuláře při přidání akademického roku.
- `akadrokzmenform` je metoda, která zobrazí formulář pro změnu akademického roku (*akadrokzmen.html*).

- `akadrokkontrola` je privátní metoda, která kontroluje, jestli daná metoda splňuje dané požadavky s integritními omezeními do databáze. Kontroluje metody `akadrokpridejfce` a `akadrokzmenfce`.
- `akadrokzmenfce` je metoda, která provádí obsluhu formuláře při změně dat pro akademický rok.
- `akadroksmazform` je metoda, která zobrazí formulář pro smazání akademického roku (`akadroksmaz.html`).
- `akadroksmazfce` je metoda, která provádí obsluhu formuláře při smazání akademického roku.

## 6. správa semestru

- `semestrpridejform` je metoda, která zobrazí formulář na přidání semestru (`semestrpridej.html`).
- `semestrpridejfce` je metoda, která provádí obsluhu formuláře při přidání semestru.
- `semestrakce` je metoda, která zobrazí všechny semestry do tabulky (`semestrotevri.html`).
- `semestrzavri` je metoda, která zavře semestr.
- `semestrkontrola` je privátní metoda, která kontroluje, jestli daná metoda splňuje dané požadavky s integritními omezeními do databáze. Kontroluje metody `semestrpridejfce` a `semestrzmenfce`.
- `semestrzmenform` je metoda, která zobrazí formulář pro změnu semestru (`semestrzmen.html`).
- `semestrzmenfce` je metoda, která provádí obsluhu formuláře při změně dat pro semestr.
- `semestrsmazform` je metoda, která zobrazí formulář pro smazání semestru (`semestrsmaz.html`).
- `semestrsmazfce` je metoda, která provádí obsluhu formuláře při smazání semestru.

## 7. správa studií

- `studiumpridejform` je metoda, která zobrazí formulář na přidání studia (`studiumpridej.html`).
- `studiumpridejfce` je metoda, která provádí obsluhu formuláře při přidání studia.

## 8. správa kruhů

- `kruhpridejform` je metoda, která zobrazí formulář na přidání kruhu (`kruhpridej.html`).
- `kruhpridejfce` je metoda, která provádí obsluhu formuláře při přidání kruhu.

- `kruhzenform` je metoda, která zobrazí formulář pro změnu kruhu (*kruhzen.html*).
- `kruhzenfce` je metoda, která provádí obsluhu formuláře při změně dat pro kruh.
- `kruhkontrola` je privátní metoda, která kontroluje, jestli daná metoda splňuje dané požadavky s integritními omezeními do databáze. Kontroluje `kruhpridejfce` a `kruhzenfce`.
- `kruhsmazform` je metoda, která zobrazí formulář pro smazání kruhu (*kruhsmaz.html*).
- `kruhsmazfce` je metoda, která provádí obsluhu formuláře při smazání kruhu.

## 9. správa předmětů

- `predmetpridejform` je metoda, která zobrazí formulář na přidání předmětu (*predmetpridej.html*).
- `predmetpridejfce` je metoda, která provádí obsluhu formuláře při přidání předmětu.
- `predmetkontrola` je privátní metoda, která kontroluje, jestli daná metoda splňuje dané požadavky s integritními omezeními do databáze. Kontroluje metody `predmetpridejfce` a `predmetzmenfce`.
- `predmetzmenform` je metoda, která zobrazí formulář pro změnu předmětu (*predmetzmen.html*).
- `predmetzmenfce` je metoda, která provádí obsluhu formuláře při změně dat pro předmět.
- `predmetsmazform` je metoda, která zobrazí formulář pro smazání předmětu (*predmetsmaz.html*).
- `predmetsmazfce` je metoda, která provádí obsluhu formuláře při smazání předmětu.

## 10. správa typů učeben

- `typucebnypridejform` je metoda, která zobrazí formulář na přidání typu učebny (*typucebnypridej.html*).
- `typucebnypridejfce` je metoda, která provádí obsluhu formuláře při přidání typu učebny.
- `typucebnyzmenform` je metoda, která zobrazí formulář pro změnu typu učebny (*typucebnyzmen.html*).
- `typucebnykontrola` je privátní metoda, která kontroluje, jestli daná metoda splňuje dané požadavky s integritními omezeními do databáze. Kontroluje metody `typucebnypridejfce` a `typucebnyzmenfce`.
- `typucebnyzmenfce` je metoda, která provádí obsluhu formuláře při změně dat pro typ učebny.

- `typucebnysmazform` je metoda, která zobrazí formulář pro smazání typu učebny (*typucebnysmaz.html*).
- `typucebnysmazfce` je metoda, která provádí obsluhu formuláře při smazání typ učebny.

#### 11. správa typů výuky

- `typvyukypridejform` je metoda, která zobrazí formulář na přidání typu výuky (*typvyukypridej.html*).
- `typvyukypridejfce` je metoda, která provádí obsluhu formuláře při přidání typu výuky.
- `typvyukysmazform` je metoda, která zobrazí formulář pro smazání typu výuky (*typvyukysmaz.html*).
- `typvyukysmazfce` je metoda, která provádí obsluhu formuláře při smazání typu výuky.
- `typvyukyzmenform` je metoda, která zobrazí formulář pro změnu typu výuky (*typvyukyzmen.html*).
- `typvyukyzmenfce` je metoda, která provádí obsluhu formuláře při změně dat pro typ výuky.
- `typvyukykontrola` je privátní metoda, která kontroluje, jestli daná metoda splňuje dané požadavky s integritními omezeními do databáze. Kontroluje metody `typvyukypridejfce` a `typvyukyzmenfce`.

#### 12. správa studií

- `studiumsmazform` je metoda, která zobrazí formulář pro smazání studium (*studiumsmaz.html*).
- `studiumsmazfce` je metoda, která provádí obsluhu formuláře při smazání studia.
- `studiumkontrola` je privátní metoda, která kontroluje, jestli daná metoda splňuje dané požadavky s integritními omezeními do databáze. Kontroluje metody `studiumpridejfce` a `studiumzmenfce`.
- `studiumzmenform` je metoda, která zobrazí formulář pro změnu studia (*studiumzmen.html*).
- `studiumzmenfce` je metoda, která provádí obsluhu formuláře při změně dat pro studium.

#### 13. správa bílé knihy

- `bilaknihapridejform` je metoda, která zobrazí formulář na přidání dat do bílé knihy tj. rozpis a studijní plán (*bilaknihapridej.html*).
- `bilaknihapridejfce` je metoda, která provádí obsluhu formuláře při přidání dat do bílé knihy tj. rozpis a studijní plán.
- `bilaknihazmenform` je metoda, která zobrazí formulář pro změnu dat v bílé knize tj. rozpis a studijní plán (*bilaknihazmen.html*).



- `bilaknihazmenfce` je metoda, která provádí obsluhu formuláře při změně dat pro bílou knihu tj. rozpis a studijní plán.
- `bilaknihasmazform` je metoda, která zobrazí formulář pro smazání dat v bílé knize tj. rozpis a studijní plán (`bilaknihasmaz.html`).
- `bilaknihasmazfce` je metoda, která provádí obsluhu formuláře při smazání dat v bílé knize tj. rozpis a studijní plán.

#### 14. správa skupin

- `skupinapridejform` je metoda, která zobrazí formulář na přidání skupiny (`skupinapridej.html`).
- `skupinakontrola` je privátní metoda, která kontroluje, jestli daná metoda splňuje dané požadavky s integritními omezení do databáze. Kontroluje metody `skupinapridejfce` a `skupinazmenfce`.
- `skupinapridejfce` je metoda, která provádí obsluhu formuláře při přidání skupiny.
- `skupinazmenform` je metoda, která zobrazí formulář pro změnu skupiny (`skupinazmen.html`).
- `skupinazmenfce` je metoda, která provádí obsluhu formuláře při změně dat pro skupinu.
- `skupinasmazform` je metoda, která zobrazí formulář pro smazání skupiny (`skupinasmaz.html`).
- `skupinasmazfce` je metoda, která provádí obsluhu formuláře při smazání skupiny.

### Třída `ajax`

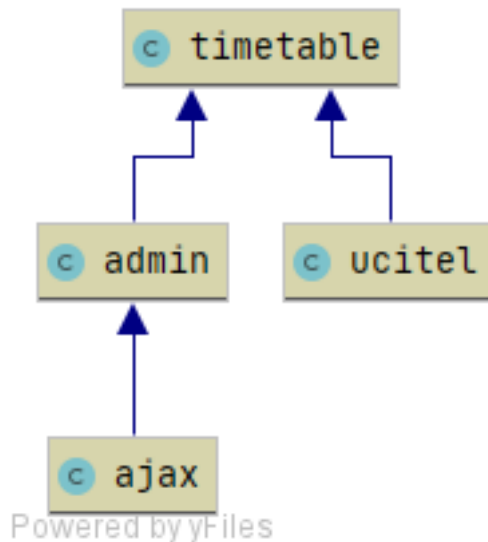
Třída `ajax` která dědí od třídy `admin`. Její metody slouží pro Javascript ze souboru `skupinaAjax.js` a `rozvrhAjax.js`.

Obsahuje tyto metody:

- `skupinapridej1` je metoda, která ve formuláři na přidání skupiny, při změně předmětu změní data pro typ výuky a kruh.
- `skupinapridej2` je metoda, která ve formuláři na přidání skupiny, při změně typu výuky změní data pro kruh.
- `rozvrhpridej1` je metoda, která ve formuláři pro přidání hodin do rozvrhu, při změně předmětu změní data pro možnost výběru hodin.
- `rozvrhpridej2` je metoda, která ve formuláři pro přidání hodin do rozvrhu, při změně učitele změní data pro možnost výběru hodin.
- `rozvrhpridej3` je metoda, která ve formuláři pro přidání hodin do rozvrhu, při změně učebny změní data pro možnost výběru hodin.

### 3.2.3 Diagram tříd webové aplikace

Na straně 34 je diagram tříd webové aplikace bez metod. Tento diagram jsem vytvořil pomocí PhpStormu, který využívá yFiles.



Obrázek 3.2: Diagram tříd (bez metod)

## 3.3 JavaScript

Ve webové aplikaci jsem pro lepší zobrazení webových stránek použil JavaScript.

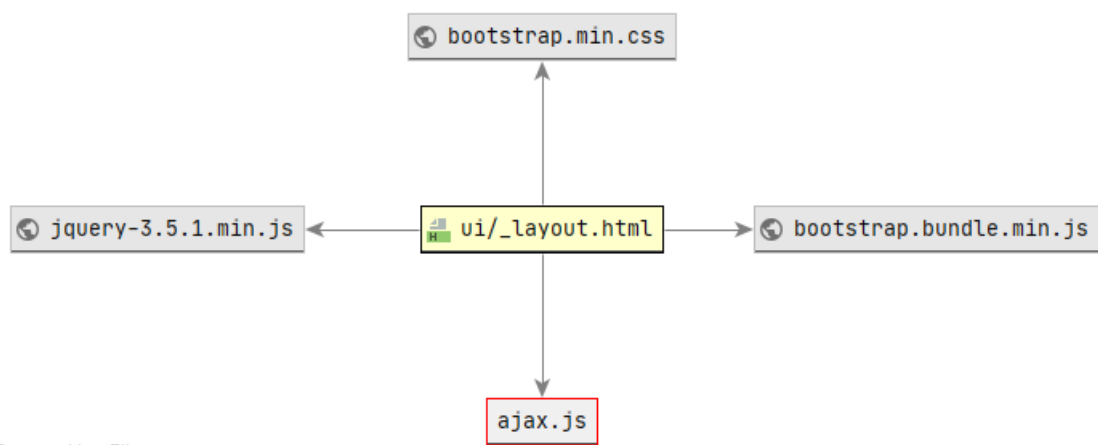
### 3.3.1 AJAX

Webová aplikace používá AJAX, konkrétně metodu `post` z knihovny jQuery. Cílem je aktualizace výběrových menu při přidání nebo změně skupiny, resp. při vkládání do rozvrhu, a to bez nutnosti znovunačtení celé stránky.

Ve složce `js` jsou soubory `rozvrhAjax.js` a `skupinaAjax.js`, které si vkládají jen ty stránky, kde je AJAX použitý.

### 3.3.2 Diagram Javascriptu

Na straně 35 je diagram Javascriptu (obrázek 3.3). Tento diagram jsem vytvořil pomocí PhpStormu, který využívá yFiles.



Powered by yFiles

Obrázek 3.3: Diagram javascriptu



# Závěr

Cílem této bakalářské práce bylo vytvoření webové aplikace pro tvorbu rozvrhu hodin, která by se dala použít pro potřeby vysoké školy. Webová aplikace má 3 typy uživatelů, přičemž administrátor má k dispozici nejvíce funkcí.

Webová aplikace pro tvorbu rozvrhu hodin umožňuje:

1. Přihlášenému uživateli s administrátorskými právy (administrátor):
  - vkládání dat do databáze,
  - má možnost upravit a smazat vložená data,
  - ruční tvorbu rozvrhu hodin.
2. Přihlášenému uživateli bez administrátorských práv (učitel):
  - zadat časová omezení,
  - zobrazit svůj rozvrh.
3. Nepřihlášenému uživateli:
  - přihlásit se,
  - zobrazení rozvrhů.

Možné budoucí úpravy:

- automatické generování rozvrhu hodin,
- hromadný import vstupních dat (z XML),
- lepší výstup dat (možnost tisku).



# Literatura

- [1] AJAX. *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2001-, 2. 4. 2021 [cit. 2021-7-1]. Dostupné z: <https://cs.wikipedia.org/wiki/AJAX>
- [2] ASc Rozvrhy. *ASc rozvrhy* [online]. Bratislava: ASC Applied Software Consultants, 1993 [cit. 2021-7-4]. Dostupné z: [https://www.asctimetables.com/timetables\\_cz.html](https://www.asctimetables.com/timetables_cz.html)
- [3] B-INTIME. *Bakaláři* [online]. Sukova třída 1548, 53002 Pardubice, Zelené Předměstí: Bakaláři, 2018 [cit. 2021-7-5]. Dostupné z: <https://napoveda.bakalari.cz/index.html?indexr.htm>
- [4] Bootstrap. *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2001-, 6. 6. 2021 [cit. 2021-7-5]. Dostupné z: <https://cs.wikipedia.org/wiki/Bootstrap>
- [5] Fat-Free Framework. *Fat-Free Framework* [online]. F3::Community, 2009 [cit. 2021-7-5]. Dostupné z: <https://fatfreeframework.com/3.7/home>
- [6] Firebird. *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2001-, 22. 1. 2018 [cit. 2021-7-5]. Dostupné z: <https://cs.wikipedia.org/wiki/Firebird>
- [7] PhpStorm. *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2001-, 15. 5. 2021 [cit. 2021-7-5]. Dostupné z: <https://cs.wikipedia.org/wiki/PhpStorm>
- [8] Skolaris. *Skolaris* [online]. Sochorova 21, 616 00 Brno-Žabovřesky: Skolaris software, 2015, 05-04-2021 [cit. 2021-7-5]. Dostupné z: <https://skolaris.net/napoveda/wiki/34>
- [9] UniTime. *UniTime* [online]. USA: The Apereo Foundation, 2015, červen 2021 [cit. 2021-7-5]. Dostupné z: <https://www.unitime.org/>
- [10] ŠTRÁFELDA, Jan. HTML. *Kdo je Štráfelda?* [online]. [cit. 2021-7-5]. Dostupné z: <https://www.strafelda.cz/html>
- [11] ŠTRÁFELDA, Jan. Kaskádové styly (CSS). *Kdo je Štráfelda?* [online]. 25-03-2021 [cit. 2021-7-5]. Dostupné z: <https://www.strafelda.cz/kaskadove-styly>

[12] ŠTRÁFELDA, Jan. PHP. *Kdo je Štráfelda?* [online]. [cit. 2021-7-5]. Dostupné z: <https://www.strafelda.cz/php>



# Příloha A

## Uživatelská příručka

K webové aplikaci na tvorbu rozvrhu hodin jsou následující požadavky:

- Firebird verze 3.0+,
- PHP verze 7.3+,
- webový server s nainstalovaným a povoleným `mod_rewrite`.

### A.1 Popis instalace

#### A.1.1 Instalace databáze

Při instalaci musíte začít nejdříve databází. Na přiloženém CD je textový soubor s názvem *dbbakalarka.txt*, který obsahuje definice domén, tabulek a pohledů (včetně některých dat). Obsah souboru je potřeba spustit ve Firebirdu, ve vyšší verzi než 3.0.

#### A.1.2 Instalace webové aplikace

Webová aplikace vyžaduje PHP ve verzi alespoň 7.3 a najdete ji na přiloženém cd, ve složce *webová aplikace*, tato data zkopírujte na webserver, např. přes program WINCSP, který budete mít připojený k serveru.

### A.2 Uživatelská příručka pro nepřihlášeného uživatele (student)

Webová aplikace pro rozvrh hodin je dostupná na adrese <https://inta.fjfi.cvut.cz/~cahamich/bp> ve webovém prohlížeči.

Nepřihlášený uživatel uvidí úvodní stránku s informacemi o aktuálním semestru. V levé nabídce pak má možnost se přihlásit nebo zobrazit rozvrhy (pro aktuální semestr podle ročníků, podle místností nebo podle učitelů).

### A.3 Uživatelská příručka pro přihlášeného uživatele bez administrátorských práv (učitel)

Po vybrání záložky *Přihlásit se* v levém menu a zadání e-mailu a hesla, které dostane od svého administrátora se přihlásí. Oproti nepřihlášenému uživateli se v menu objeví o tři položky navíc: *Osobní rozvrh*, *Změna hesla*, *Časová omezení* a *Přihlásit se* se změní na *Odhlásit*.

Zadání časových omezení: učitel si ve formuláři vybere, kdy nemůže vyučovat. Zaškrtnutí znamená, že nemůže v daný den a čas.

Změna hesla: učitel si může změnit heslo sám sobě, v záložce *Změna hesla* zadá staré heslo a dvakrát nové heslo. Heslo musí mít více jak šest znaků a z toho musí být jedno velké písmeno a jedno číslo, přičemž dvakrát zadané heslo se musí shodovat.

### A.4 Uživatelská příručka pro přihlášeného uživatele s administrátorskými právy (administrátor)

Pro první přihlášení použijte záložku *Přihlásit se* a zadejte e-mail `admin@admin.com` s heslem `Admin12`, poté jděte do záložky *uživatelé* a přidejte sebe jako samostatného administrátora, smažte univerzálního administrátora, poté se znovu přihlaste.

Před tvorbou rozvrhu hodin je nutné vyplnit potřebná data, nejlépe v tomto pořadí: uživatelé, akademický rok, typy učeben, učebny, typy výuky, předměty, semestr (který bude otevřený), studium, kruh(y), bílá kniha, skupina a nakonec samotný rozvrh. Po vytvoření rozvrhu je potřeba semestru uzavřít. Pokud rozvrh souhlasí s bílou knihou, tak se začne zobrazovat nepřihlášeným uživatelům.

Administrátor může přidávat, měnit nebo mazat téměř vše, až na typ semestru, dny v týdnu a vyučovací hodiny (toto lze udělat pouze v databázi nebo při instalaci nutno změnit příkazy `INSERT` v souboru `dbbakalarka.txt`).

Administrátor má v levé části webové stránky menu s mnoha položkami. Většina z nich zobrazí stránku s formulářem pro přidání dat a pod ním přehled aktuálních dat.

Pokud není uvedeno jinak, položka nemůže být prázdná.

### **A.4.1 Uživatelé**

Uživatele musíte zadat sám ručně, není žádná registrace, podle e-mailu a smyšleného hesla, které si poté uživatel sám změní.

Jméno musí obsahovat dva až třicet písmen.

Příjmení musí obsahovat dva až třicet písmen.

Titul před jménem a za jménem musí obsahovat maximálně deset znaků a mohou být prázdné.

Heslo musí obsahovat velké písmeno, číslici a musí mít víc než pět znaků, totéž pro heslo znovu.

E-mail musí mít formát klasického e-mailu (admin@admin.com), musí mít maximálně sto dvacet znaků a musí být unikátní.

### **A.4.2 Akademické roky**

Rok od a rok do musí být čtyři celá čísla, od nuly až do devítky a musí být unikátní.

### **A.4.3 Typ učebny**

Zadáte název typu učebny (např. počítačová učebna).

Název typu učebny musí být maximálně třicet znaků a unikátní.

### **A.4.4 Typ výuky**

Zadáte název typu výuky a zkratku (např. přednáška, př). Název typu výuky musí být maximálně třicet znaků a unikátní.

Zkratka typu výuky musí být maximálně deset znaků a unikátní.

### **A.4.5 Učebna**

Vyberete typ učebny, zadáte označení učebny a kapacitu.

Označení učebny musí být maximálně pět znaků a unikátní.

Kapacita musí být celé číslo větší než nula.

### **A.4.6 Předmět**

Zadáte název předmětu, kód předmětu, zakončení a vyberete pro jaký typ semestru je tento předmět.

Název předmětu musí být maximálně šedesát znaků a unikátní.

Kód předmětu musí být maximálně pět znaků a unikátní.

### **A.4.7 Studium**

Zadáte název nového studia a zkratku.

Název studia musí být maximálně šedesát znaků.

Zkratka studia musí být kratší než šest znaků.

### **A.4.8 Kruh**

Vyberete název studia, akademický rok a zadáte ročník.

Ročník musí být jedno celé číslo (jedna až tři).

### **A.4.9 Semestr**

Vyberete typ nového semestru, akademický rok, zadáte kód nového semestru, začátek výuky semestru, konec výuky semestru, datum začátku omezení pro zadávání časových omezení a konec časových omezení.

Kód semestru musí být kratší než šest znaků a unikátní.

Začátek výuky semestru nesmí být dříve než konec výuky semestru.

Datum začátku omezení pro zadávání časových omezení nesmí být později než konec časových omezení.

Při přidávání je semestr automaticky uzavřený. Pokud chcete přidávat data do bílé knihy, skupiny nebo rozvrhu, musíte mít otevřený právě jeden semestr. Ten otevřete v záložce *semestry* a kliknete na semestr, který chcete změnit a ten zakliknutím otevřete (to samé funguje pro uzavření semestru akorát naopak).

### **A.4.10 Bílá kniha**

Vyberete kruh, předmět, typ výuky, typ učebny, jméno učitele, zadáte počet hodin týdně a jestli je předmět vyučován týdně. Když předmět není týdně tak se nezohledňuje při kontrole rozvrhu (předmět nemusí být zařazen do rozvrhu).

Počet hodin týdně musí být celé číslo. Pokud nezaškrtnete jednu z vybraných možností na vyučování týdně, je předmět automaticky vyučován týdně.

### **A.4.11 Skupina**

Skupina tj. blok který pak vložíte do rozvrhu.

Vyberete předmět, typ výuky předmětu, kruh, zda je vyučován týdně a zadejte kolik hodin bude mít blok v rozvrhu. Při změně výběru předmětu v aplikaci se změní i typ výuky a kruh podle bílé knihy. Pokud nezaškrtnete jednu z vybraných možností na vyučování týdně, je předmět automaticky vyučován týdně.

Počet hodin týdně musí být menší nebo stejný jako je u stejného předmětu a stejného typu výuky v bílé knize.

### **A.4.12 Rozvrh**

Vyberete předmět, kruh, učitele, učebnu, čas a pokud budete mít nějakou poznámku k této části rozvrhu, můžete si jí napsat do poznámek ale není to povinné.

Poznámka musí být kratší než třicet znaků.

Po vyplnění rozvrhu uzavřete semestr a rozvrh se zobrazí ve webové aplikaci všem.



# Příloha B

## Obsah CD

<i>bp_caha.pdf</i>	text bakalářské práce
Složka <i>webová aplikace</i>	složka s webovou aplikací
<i>dbbakalarka.txt</i>	textový soubor pro instalaci databáze