



**VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ**

BRNO UNIVERSITY OF TECHNOLOGY

**FAKULTA INFORMAČNÍCH TECHNOLOGIÍ**

FACULTY OF INFORMATION TECHNOLOGY

**ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ**

DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA

**PROGRAM PRO PODPORU PLÁNOVÁNÍ ROZVRHŮ**

TIMETABLE PLANNING SUPPORT SOFTWARE

**BAKALÁŘSKÁ PRÁCE**

BACHELOR'S THESIS

**AUTOR PRÁCE**

AUTHOR

**ALENA TESAŘOVÁ**

**VEDOUcí PRÁCE**

SUPERVISOR

**Ing. JAROSLAV DYTRYCH, Ph.D.**

BRNO 2019

## Zadání bakalářské práce



21894

Studentka: **Tesařová Alena**  
Program: Informační technologie  
Název: **Program pro podporu plánování rozvrhů**  
**Timetable Planning Support Software**  
Kategorie: Web

Zadání:

1. Seznamte se s problematikou plánování rozvrhů zkoušek a výuky na FIT VUT v Brně a s programovacími jazyky využitými v nástrojích pro jeho podporu.
2. Prostudujte dostupné programy pro podporu této činnosti a vstupy a výstupy plánování z předchozích semestrů.
3. Navrhněte nový program pro plánování rozvrhů, který poskytne vhodnou podporu pro přípravu podkladů pro plánování a kompletaci požadavků vyučujících. Program musí umožnit export dat ve formátech vhodných pro import do informačního systému FIT i do dříve využívané aplikace.
4. Implementujte navržené řešení.
5. Zhodnoťte dosažené výsledky a vytvořte stručný plakát prezentující výsledky práce.

Literatura:

- Dle doporučení vedoucího

Pro udělení zápočtu za první semestr je požadováno:

- Body 1, 2 a 3.

Podrobné závazné pokyny pro vypracování práce viz <http://www.fit.vutbr.cz/info/szz/>

Vedoucí práce: **Dytrych Jaroslav, Ing., Ph.D.**

Vedoucí ústavu: Černocký Jan, doc. Dr. Ing.

Datum zadání: 1. listopadu 2018

Datum odevzdání: 15. května 2019

Datum schválení: 6. listopadu 2018

## Abstrakt

Tato práce se zabývá návrhem a realizací programu pro plánování rozvrhů výuky a zkoušek na FIT VUT v Brně. Výsledný program poskytne podporu pro přípravu vstupních dat, kompletaci požadavků vyučujících a dále umožní samotné plánování a následný export dat ve formátech vhodných pro import do informačního systému FIT i do dříve využívané aplikace. Teoretická část se věnuje obecné problematice plánování rozvrhů a specifikům plánování na FIT. Realizační část obsahuje návrh systému a popis implementace aplikace. Výsledkem práce je software, který je aktuálně využíván pro podporu plánování rozvrhů na FIT VUT v Brně.

## Abstract

This thesis deals with the design and implementation of a schedule planning software for lectures and exams at FIT VUT. The final result will provide support for input data preparation, completing teacher's requirements and allowing the planning and subsequent data export in formats suitable for importing into the FIT information system and into the previously used application. The theoretical part deals with general issues of schedule planning and specifics of planning at FIT. The implementation part contains the system design and application implementation description. The result of the work is a software that is currently used to support timetable scheduling at FIT VUT.

## Klíčová slova

Plánování rozvrhů, výuka, zkoušky, kompletace požadavků, export, import, REST, FIT, Nette, PHP, MySQL, Doctrine 2, Bootstrap 4

## Keywords

Schedule planning, lectures, exams, completing requirements, export, import, REST, FIT, Nette, PHP, MySQL, Doctrine 2, Bootstrap 4

## Citace

TESAŘOVÁ, Alena. *Program pro podporu plánování rozvrhů*. Brno, 2019. Bakalářská práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce Ing. Jaroslav Dytrych, Ph.D.

# Program pro podporu plánování rozvrhů

## Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracovala samostatně pod vedením pana Ing. Jaroslava Dytrycha, Ph.D. Uvedla jsem všechny literární prameny a publikace, ze kterých jsem čerpala.

.....  
Alena Tesařová  
5. května 2019

## Poděkování

Děkuji vedoucímu bakalářské práce Ing. Jaroslavu Dytrychovi, Ph.D. za účinnou metodickou, pedagogickou a odbornou pomoc a další cenné rady při zpracování mé bakalářské práce. Dále bych chtěla poděkovat Ing. Petrovi Veigendovi, který mi pomáhal s testováním aplikace a při implementaci mi poskytoval zpětnou vazbu.

# Obsah

<b>1</b>	<b>Úvod</b>	<b>4</b>
<b>2</b>	<b>Plánování rozvrhů výuky a zkoušek</b>	<b>6</b>
2.1	Plánování rozvrhů . . . . .	6
2.2	Průběh plánování na FIT . . . . .	14
2.2.1	Plánování výuky . . . . .	16
2.2.2	Plánování zkoušek . . . . .	23
2.3	REST API . . . . .	28
<b>3</b>	<b>Návrh aplikace</b>	<b>30</b>
3.1	Návrh funkcionality aplikace . . . . .	30
3.1.1	Stahování vstupních dat . . . . .	30
3.1.2	Kompletace požadavků . . . . .	34
3.1.3	Plánování rozvrhu . . . . .	35
3.1.4	Navrhované funkce při plánování . . . . .	37
3.1.5	Automatické rozmístění oken u zkoušek . . . . .	38
3.1.6	Příznaky u předmětů . . . . .	38
3.1.7	Exporty . . . . .	40
3.1.8	Import na server FIT . . . . .	41
3.2	Návrh databáze . . . . .	41
3.3	Architektura aplikace . . . . .	45
<b>4</b>	<b>Nástroje a technologie použité pro vývoj</b>	<b>46</b>
<b>5</b>	<b>Implementace</b>	<b>50</b>
5.1	Hlavní strana aplikace . . . . .	51
5.2	Kompletace požadavků . . . . .	53
5.3	Plánování rozvrhů výuky a zkoušek . . . . .	54
5.4	Evidence předmětů . . . . .	56
5.5	Počty studentů na zkoušku . . . . .	56
5.6	Příznaky . . . . .	57
5.7	Export dat . . . . .	57
<b>6</b>	<b>Testování a nasazení aplikace</b>	<b>61</b>
<b>7</b>	<b>Závěr</b>	<b>65</b>
	<b>Literatura</b>	<b>67</b>

# Seznam obrázků

2.1	Tabule pro fakultu, kde dochází k plánování pro jednotlivé dny na MENDELU. Zdroj: Ing. Michal Karhánek. . . . .	9
2.2	Ukázka uživatelského rozhraní rozvrhu UniTime používaného na Masarykově univerzitě. . . . .	11
2.3	Generování rozvrhu v aplikaci Roger. Červeně je zvýrazněna kolize u předmětu STAT1. . . . .	13
2.4	Aktuální proces plánování na FIT . . . . .	15
2.5	Ukázka současné aplikace na plánování rozvrhu výuky . . . . .	16
2.6	Kompletace požadavků probíhá v programu Microsoft Excel . . . . .	19
2.7	Průběh současného exportování stránek na web . . . . .	21
2.8	Ukázka exportování používaná v původní aplikaci . . . . .	22
2.9	Kompletace požadavků v programu Microsoft Excel pro zkoušky . . . . .	25
2.10	Tabulka časového plánu termínů zkoušek pro přednáškovou skupinu 3BIT . . . . .	27
3.1	Diagram případů užití navrhované aplikace. . . . .	31
3.2	Stránka s kolizemi dostupná v IS FIT . . . . .	32
3.3	Stránka s počty na zkoušku dostupná v IS FIT . . . . .	33
3.4	Návrh hlavní stránky aplikace vytvořený v grafickém programu Figma . . . . .	33
3.5	Návrh tabulky sloužící ke kompletaci požadavků . . . . .	34
3.6	Přechody stavů rozvrhu. U hrany je vždy zkratka role uživatele, která může operaci provést: A – správce, S – student. . . . .	36
3.7	ER diagram databáze . . . . .	44
4.1	Životní cyklus aplikace s Frameworkem Nette [27] . . . . .	47
5.1	Hlavní strana aplikace . . . . .	52
5.2	Přihlášení do REST API pro import dat . . . . .	53
5.3	Kompletace požadavků u výuky . . . . .	54
5.4	Prostředí pro práci s rozvrhem . . . . .	55
5.5	Evidence předmětů, jejich povinností, studentů a vyučujících . . . . .	56
5.6	Editace počtů studentů k danému termínu zkoušky . . . . .	57
5.7	Přiřazení příznaků k předmětu a přednáškové skupině . . . . .	57
5.8	Možnosti exportu u výuky . . . . .	58
5.9	Editace tabulky časového plánu zkoušek . . . . .	59
5.10	Konfigurace textů zobrazených pod tabulkou časového plánu . . . . .	60
5.11	Možnosti exportu u zkoušek . . . . .	60
6.1	Tabulka povinností . . . . .	63

# Seznam tabulek

2.1	Tabulka důležitých místností a jejich kapacit . . . . .	17
2.2	Kombinace místností a jejich výsledná kapacita . . . . .	18
2.3	Popis nástrojů současné aplikace, které se nachází v horním rohu aplikace .	20
2.4	Příklad formátu dat rozvrhové akce vrácených přes REST API . . . . .	28
3.1	Rozhraní REST, prostřednictvím kterého budu získávat data. Všechny metody vyžadují autentizaci a autorizaci uživatele. . . . .	32
3.2	Období požadavků, která se používají pro kompletaci. Zkratky: min. – minulý, akt. – aktuální, sem – semestr, LS – letní semestr, ZS – zimní semestr, zk – zkouška. . . . .	35
3.3	Vysvětlení formátu buňky rozvrhového okna . . . . .	36
3.4	Stavy rozvrhu . . . . .	37
3.5	Tabulka příznaků a jejich vlastností . . . . .	38
3.6	Pravidla pro generování příznaků. Zkratka lib. – libovolný ročník . . . . .	39

# Kapitola 1

## Úvod

Tvorba rozvrhu je problém, se kterým se škola pravidelně setkává několikrát ročně. Na prvním stupni základní školy bylo řešení nejjednodušší, každá třída měla svého třídního učitele, se kterým byla celý den v jedné třídě. S druhým stupněm náročnost stoupá, nicméně se stále jedná o rozdělování tříd do učeben a neřeší se zde jednotlivci. Pro tyto potřeby se dají využít k řešení klasické tabulkové editory (např. Microsoft Office Excel) anebo papír a tužka.

Na vysokoškolském stupni jde o zcela jinou úroveň složitosti a dostupné programy přestávají být vhodné, jelikož má každá fakulta své specifické požadavky. Plánování se dělí na dvě části – plánování výuky a zkoušek. Vždy je potřeba myslet na to, aby učitel neučil v jeden čas na dvou místech, aby nebyly naplánovány dvě různé akce do stejné místnosti, anebo aby se studentovi nepřekrývaly dva povinné předměty. Zároveň musíme počítat s tím, že je výuka často organizována i na jiných fakultách, které se mohou nacházet na druhém konci města, proto je třeba myslet na dostatek času pro přechod z fakulty na fakultu. Porušení těchto podmínek by znamenalo nutnost rozvrh přeplánovat. Vedle těchto omezení existují i kritéria, která se snaží vyjít vstříc jak učitelům, tak studentům. Jedná se například o bezkoliznost volitelných přednášek, soustředění výuky do určitých dní, preferovat určité časy a vyhovět požadavkům od učitelů. Z pohledu plánování je nutné myslet na všechny tyto podmínky a snažit se seskládat pokud možno co nejlepší rozvrh.

Problémů u plánování je mnoho a existuje spousta různých řešení, která na problém nahlíží z více stran. Některé programy nabízejí pouze asistenci při vytváření rozvrhů a mohou velmi zjednodušit práci při přesouvání oken do rozvrhů, kontrole kolizí a exportu dat. Existují ale i programy, které nabízejí možnost automatického generování rozvrhu. Tyto programy vyhodnotí všechna kritéria automaticky. Z celé škály softwarových produktů si ale často škola nedokáže vybrat kvůli svým specifickým požadavkům. *Fakulta informačních technologií Vysokého učení technického v Brně* (dále jen *FIT*) je právě jednou z těchto škol.

Na FIT se jedná o problém nekompletnosti dat a nutnost jejich ručního doplnění, dále je to možnost promítání přednášek z jedné místnosti do druhé a umožnění učitelů takto učit až ve třech místnostech zároveň. Neméně důležité jsou také specifické požadavky na výsledný export rozvrhu a import do informačního systému. Současně využívaná aplikace neumí automaticky stahovat data, kompletace dat probíhá na jiném místě ručním způsobem a pro export je třeba složitě spouštět několik skriptů.

Cílem mé práce je prostudování plánování rozvrhů výuky a zkoušek se zaměřením se na *Fakultu informačních technologií*, pro kterou navrhnu a implementuji software pro podporu plánování. Jeho hlavní funkcí bude příprava podkladů, bez kterých není možné zahájit proces plánování, dále bude umožňovat zkompletovat veškerá data včetně požadavků vyu-



čujících, poskytne vhodné rozhraní pro plánování a v poslední řadě umožní export náhledu rozvrhu, který je možno zveřejnit na webu, a vytvoří soubor pro nahrání rozvrhu do informačního systému *FIT*.

V sekci 2.1 se zaměřuji obecně na problém plánování rozvrhů a analyzuji existující řešení, u kterých mě zajímá hlavně nasaditelnost pro *FIT*. V sekci 2.2 je pak podrobně popsán průběh plánování rozvrhů na *FIT*. Na základě znalostí současného stavu je pak možné sestavit návrh aplikace na podporu plánování, kterým se zabývá kapitola 3. Nástrojem použitým při vývoji, implementaci a testování aplikace jsou pak věnované samostatné kapitoly 4, 5 a 6. Na závěr sumarizuji dosažené výsledky a navrhuji další funkce aplikace pro pokračující vývoj.

## Kapitola 2

# Plánování rozvrhů výuky a zkoušek

Tato kapitola se nejprve zabývá obecně plánováním rozvrhů na školách. Dále uvádí různé problémy, na které je potřeba během plánování myslet, a zaměřuje se na specifikace problémů a kritérií na FIT. Obecných řešení a existujících aplikací lze najít velké množství, já jsem si vybrala pouze několik řešení, která popisují v kapitole 2.1.

Jedním z řešení je také aplikace, která se v současnosti na FIT používá. V sekci 2.2 je podrobně rozepsaný současný průběh plánování rozvrhů zkoušek a výuky včetně získání a kompletace vstupních dat, exportu na web a importu do informačního systému.

### 2.1 Plánování rozvrhů

Plánování rozvrhů je jedním z klasických problémů, který se řeší po několik let různými výzkumnými skupinami [17]. Cílem je přiřazení jednotlivých úloh na zdroje v čase tak, aby byl maximalizován celkový zisk z provedení naplánovaných úloh. Hledá se tedy nejlepší řešení z pohledu nákladů na výkon stroje, času programu, lidského pohodlí a podobně. Neexistuje deterministický algoritmus, který by uměl najít optimální řešení v polynomiálním čase. Jedná se o známý NP problém. Různými postupy se však můžeme přiblížit k optimálnímu řešení, které můžeme považovat za dostatečně dobré a ukončit další hledání [9].

Plánování rozvrhů spadá do kategorie optimalizačních úloh s omezujícími podmínkami. Takové úlohy se nazývají CSPs (Constraint Satisfaction Problems) a jsou obvykle definovány množinou proměnných, kterým se přiřazují hodnoty z množin přípustných hodnot pro tyto proměnné [25].

Pro zajímavost lze uvést, že od roku 1995 se svolává mezinárodní konference Practice and Theory of Automated Timetabling (PATAT) [19], kde se schází výzkumníci z celého světa a diskutuje se o počítačově založeném automatickém vytváření rozvrhů. Příští konference se bude konat v roce 2020 v Belgii. Tento rok PATAT sponzoruje soutěž *International Timetabling Competition 2019* (dále jen ITC) organizovanou částečně Masarykovou univerzitou a již je přihlášených 91 účastníků z 34 zemí [21]. Soutěž má za cíl vytvořit bohatý soubor dat s různými charakteristikami, která obohatí další výzkum [10]. Výhru poskytne další sponzor EURO working group on Automated Timetabling<sup>1</sup> (EWG PATAT) v hodnotě 500 EUR.

Z pohledu této práce je nejzajímavější právě plánování rozvrhů výuky a zkoušek na vysokých školách. Plánování je nelehký úkol, který si řeší každý škola sama a neexistuje proto

<sup>1</sup><https://www.euro-online.org/web/ewg/14/ewg-patat-euro-working-group-on-automated-timetabling>

jednoznačné a správné řešení. Existuje řada obecných omezujících podmínek a požadavků charakteristických pro jednotlivé vysoké školy.

## Kritéria pro plánování

Jak bylo zmíněno v předchozí kapitole, každá škola řešící rozvrhy se setkává s různými problémy. Některé problémy jsou avšak pro všechny školy stejné. Obecně můžeme rozdělit omezení na [8]:

- **Tvrdá omezení** (označovaná i jako silná omezení nebo angl. *hard constraints*) rozdělují rozvrhový prostor na dvě disjunktí množiny – množinu přípustných a množinu nepřípustných řešení. Jde klasicky o situace z reálného světa, kdy nemůže být učitel ve dvou místnostech zároveň anebo nemohou v jedné místnosti probíhat dvě rozvrhové události, pokud je učí jiný vyučující.
- **Měkká omezení** (označovaná i jako slabá omezení nebo angl. *soft constraints*) definují optimalizační kritéria na kvalitu rozvrhu. Těmito kritériím se snažíme vyhovět co nejvíce a dosáhnout tak co nejlepšího rozvrhu. V případě školních rozvrhů se jedná například o požadavky vyučujících, aby student neměl dvě zkoušky v jednom dni anebo aby nezačínala výuka v nevhodnou dobu.

Tvrdá omezení představují vlastnost rozvrhu, která musí být splněna, aby byl rozvrh použitelný v praxi. Nedá se o nich diskutovat, jelikož musí být splněna a rozvrhář je musí respektovat.

V mé práci se soustřeďuji na FIT, která musí při rozvrhování řešit jak tvrdá kritéria, tak i kritéria měkká. Z pohledu FIT se jedná o následující **silná omezení** [26]:

- **Student nesmí mít přednášky z povinných nebo povinně volitelných předmětů v kolizi.** Pokud by došlo ke kolizi a student by nemohl chodit na povinné předměty, ztrácela by povinnost smysl. V případě opakujících se přednášek pro jednotlivé přednáškové skupiny je potřeba, aby nebyla v kolizi alespoň jedna z nich.
- **Bezkoliznost přednášek a cvičení z pohledu učitele.** Pokud vyučující přednáší více předmětů, je třeba dbát na to, aby se předměty nepřekrývaly.
- **Přednášky i cvičení musí být naplánovány do místností s dostatečnou kapacitou.** Dostatečným počtem je počet větší nebo roven 70 % z celkového množství studentů, kteří mají tento předmět zapsaný.
- **V dané místnosti v daném čase může být naplánována maximálně jedna rozvrhová akce.**
- Pokud je vyžadováno určité **pořadí u dvojice přednášek nebo přednášky a cvičení**, je potřeba ho zachovat. Typicky potřebujeme mít přednášku v daném týdnu před příslušným cvičením, proto plánování musí toto kritérium splňovat.
- **Dostatek času mezi přednáškami z jiných fakult.** Výuka často probíhá v některých místnostech na Fakultě elektrotechniky a komunikačních technologií Vysokého učení technického v Brně (dále jen FEKT) a nelze ji proto naplánovat hned po předmětu, který je vyučován na FIT, jelikož by studenti nestihli přesun z FIT na FEKT.

- Pokud je použit **streaming**, jsou povoleny pouze dovolené kombinace místností znázorněné v tabulce 2.2.
- Zohlednění časů, kdy vyučující **nemohou učit**. Pokud nemůže vyučující učit v daný čas, nemůže mu v tento konkrétní čas rozvrhář nic naplánovat.

V případě místností můžeme ještě uvažovat o vhodném umístění předmětu do místností vzhledem k jejich vybavení. Jelikož je vybavení jednotlivých místností na FIT téměř stejné, lze proto toto kritérium považovat za téměř vždy splněné. Pokud by měl vyučující zvláštní požadavky na místnosti, může si je dát do preferencí.

Když jsou splněna všechna vyjmenovaná silná kritéria, vznikne rozvrh, který je použitelný v praxi. Takový rozvrh ale nemusí být vždy ideální a ve většině případů nevyhovuje, jelikož silná kritéria neříkají nic o pohodlnosti rozvrhu z pohledu učitele nebo studenta. O kvalitě vypovídají měkká omezení, která se snaží co nejvíce uspokojit požadavky ze strany studenta a vyučujícího. Na FIT se jedná o následující **měkká kritéria** [26]:

- **Bezkoliznost přednášek** u všech typů předmětů. Jedná se o podobný bod jako první bod seznamu tvrdých kritérií, ale navíc se berou všechny typy předmětů – kromě povinných a povinně volitelných i volitelné předměty.
- Co nejméně **dlouhých bloků** výuky. Blokem výuky se myslí několik přednášek jdoucích za sebou. Pokud je blok příliš dlouhý, student ztrácí soustředěnost, je více unavený a nevěnuje tolik pozornost přednášce. Za horní hranici přijatelnosti je možné považovat 6 hodin (2 × 3 hod.). Pokud bude blok delší, budeme ho považovat za porušení kritéria.
- Pauza na **oběd**. Vždy je potřeba při plánování myslet na to, aby jak studenti, tak i vyučující měli čas si zajít na oběd.
- **Soustředit výuku pouze do určitých dní**. Pokud má student 5 přednášek týdně, je pro něj určitě lepší všech 5 přednášek rozmístit do dvou dní než mít jednu přednášku každý den. Navíc je na FIT velké množství studentů, kteří dojíždějí a určitě by ocenili zůstat doma o den déle anebo přijet o den dříve.
- Nezačínat výuku v **sedm ráno**. Plánování v tuto hodinu je možné a někdy se s tím u rozvrhů setkáme. Nutíme pak jak studenty, tak vyučující k brzkému vstávání a to není velmi oblíbené, proto se toto nedoporučuje.
- Nekončit výuku **příliš pozdě**. K plánování je využíván rozsah časových oken až do 20:50. Prakticky ani vyučující ani student není spokojený, že končí tak pozdě ve škole. Výuka, která je naplánována po 19:00, se považuje za nesplnění kritéria.
- Nepoužívat k plánování pátek odpoledne. Studenti často odjíždějí domů, a proto chtějí končit co nejdříve. Extrémně negativně je pak vyhodnocována výuka v pátek večer.
- Předměty, které mají přednášky pro více přednáškových skupin, by je pro různé skupiny měly mít v rozdílných dnech.
- **Preference vyučujících**, kdy by chtěli učit. Učitel si může zvolit preferované časy, místnosti, případně dny, kdy by chtěl učit. Na FIT požadavky vyučujících hrají velmi významnou roli a je k nim vždy přihlíženo, aby se snížil počet reklamací rozvrhů.

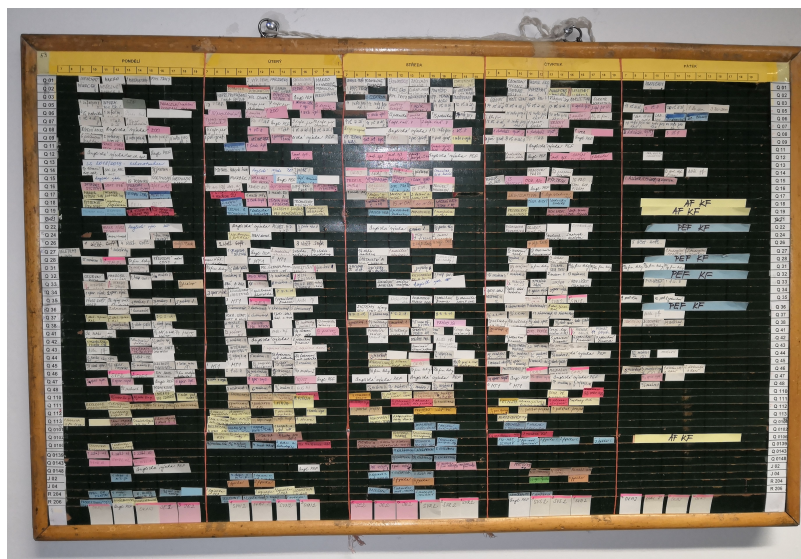
Pomocí slabých kritérií jsme schopni ohodnotit kvalitu rozvrhu, která nám udává, jak velká část studentů a vyučujících bude s rozvrhem spokojena. Někdy ale není možné splnit dvě slabá kritéria. Příkladem je, když vyučující požaduje výuku v sedm hodin ráno, čímž je ve sporu s jedním kritériem o neplánování výuky na sedmou ráno. Rozvrh se pak většinou stává kompromisem obou stran. Téměř vždy se najde někdo, komu rozvrh nevyhovuje, ale cílem rozvrháře je minimalizovat jejich počet, samozřejmě s ohledem na tvrdá omezení.

## Metody vytváření rozvrhů a existující řešení

Problémů u plánování rozvrhů je více a existuje spousta metod, které se tyto problémy snaží vyřešit. Malým školám často stačí zkopírovat rozvrhy z minulého roku, přidat nové předměty a upravit detaily. V tomto případě se plánování může výrazně zkrátit a není potřeba téměř žádné výpočetní techniky. Počítač pak můžeme použít v případě vyhodnocení kvality rozvrhu, který jsme vytvořili [8]. Příkladem by byla například Technická univerzita ve Zvoleně (informace pochází od Ing. Jany Námešné, integrátorky univerzity), která vždy zkopíruje rozvrh z minulého roku, upraví detaily a po vložení dat do systému systém zhodnotí kolize.

### Metoda tužka a papír

Dalším příkladem, kdy není potřeba téměř žádné techniky, je metoda stylem tužka a papír. Někdo by mohl tvrdit, že se jedná o zastaralou a nemoderní techniku, která je velice zdoluhavá. Tvrzení nicméně nemusí být pravdivé, jelikož školy s tímto zaběhnutým způsobem plánování rozvrhů již mají naučené způsoby, jak rychle a efektivně poskládat na tabuli jednotlivé rozvrhové papírky, aby vzniklo co nejméně kolizí. Příkladem používání takového plánování je Mendelova univerzita v Brně (dále jen MENDELU), kde se k plánování používá pět tabulí podle fakult. Jednu z pěti tabulí lze vidět na obrázku 2.1.



Obrázek 2.1: Tabule pro fakultu, kde dochází k plánování pro jednotlivé dny na MENDELU. Zdroj: Ing. Michal Karhánek.

Každý předmět je vytisknutý na samostatném papíru, jehož šířka je ovlivněna délkou rozvrhové akce. Typ akce se pak řeší zvýrazněním. Například dvouhodinová přednáška předmětu XZ bude znázorněna papírem o velikosti dva centimetry bez zvýraznění. Papíry s rozvrhovými okny jsou rozděleny na bílé povinné, modré PV, červené jsou jazyky a žluté volitelné. Přednášky jsou pak zvýrazněny rámečkem. Plánování probíhá tak, že se nejprve rozdělí povinné předměty do největších místností. V dalším kroku pak řeší fakultní předměty. Po celou dobu se pracuje s požadavky vyučujících, které jsou vytisknuté na požadavkových listech pro každý předmět a obsahují mimo požadavků vyučujících také informace o registrovaných studentech, počtu a velikosti studijních skupin, limitech registrací a zápisu, periodicitě výuky a obecné informace o předmětu.

Proces trvá zhruba týden a pak následně 2 – 4 dny se zadávají informace do systému, kdy jeden diktuje a druhý zadává. Tato metoda byla již na FIT několikrát používána, ale plánování zabíralo příliš mnoho času a úsilí rozvrháře, jelikož bylo potřeba si řádně připravit vstupní data, požadavky vyučujících a pak vše vytisknout a rozstříhat. Informace k plánovacímu procesu MENDELU jsou platné ke dni 18.03.2019 a byly poskytnuty integrátorem školy MENDELU Ing. Michalem Karhánkem.

## Metody s asistencí počítače

Vedle plánování stylem tužka a papír existují programy pro usnadnění tvorby rozvrhů, které můžeme rozdělit na dva druhy. První z nich má za cíl asistovat u procesu ručního plánování a co nejvíce ulehčit práci. Jedná se o programy, které kladou význam hlavně na vizualizaci, snadné nahrání dat a často upozorňují uživatele na vzniklé kolize. Některé z nich mohou uživateli doporučit vhodnou pozici pro zvolený předmět. Jejich výhodou oproti ručnímu plánování je kontrola kolizí, snadné kopírování rozvrhů z minulých let a přehledná vizualizace oken rozvrhu např. ve formátu HTML pro vložení na web. Nevýhodou může být zdlouhavé nahrávání dat do softwaru. Vytvoření takového softwaru bude cílem mé bakalářské práce.

Druhým typem jsou většinou složitější programy, jejichž součástí je automatické generování rozvrhu. Před spuštěním výpočtu je vždy potřeba zadat právě množinu měkkých kritérií a parametry algoritmu (například maximální počet generací nebo minimální hodnotu celkové přijatelnosti). Podle zadaných požadavků se pak vygeneruje rozvrh, který je často možné ještě upravit. Příkladem je například slovenský software *aSc Rozvrhy* nebo úspěšný software *UniTime* a *Roger*.

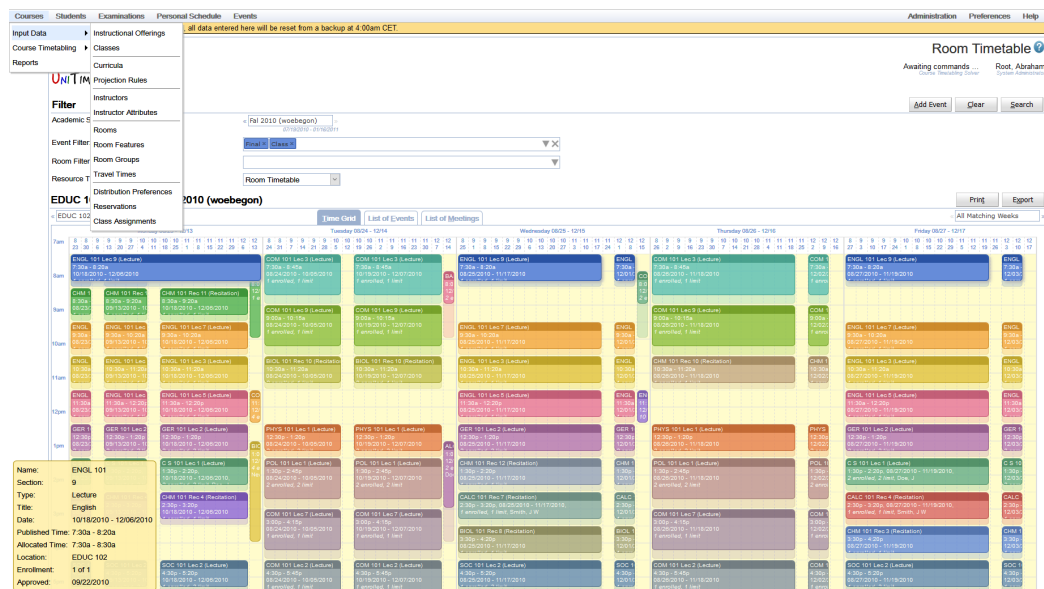
## UniTime

Software UniTime je komplexním řešením s otevřeným zdrojovým kódem (licence GNU GPL<sup>2</sup>) určeným pro rozvrhování na vysokých školách a používaným na univerzitách po celém světě. Systém byl vyvinut na Perdue university v USA a jedna z univerzit, která se podílí na výzkumu, je Masarykova Univerzita, která je od roku 2010 jedním z prvních uživatelů tohoto softwaru. Systém není závislý na žádné platformě, implementace systému je v programovacím jazyce Java, používá SQL databáze a využívá technologie jako jsou: Java Server Pages, Hibernate, Spring a Google Web Toolkit [18]. K aplikaci je možný přístup z webového prohlížeče Firefox, Chrome, Internet Explorer nebo Safari. Import a export dat je možný ve formátu XML [18]. Je v něm implementováno automatické plánování na

---

<sup>2</sup>GNU General Public License je dostupná na adrese <http://www.gnu.org/licenses/gpl-3.0.txt>.

základě algoritmu prohledávání stavového prostoru s omezujícími podmínkami a lokálního prohledávání. Jedná se o výherce soutěže ITC<sup>3</sup> 2007.



Obrázek 2.2: Ukázka uživatelského rozhraní rozvrhu UniTime používaného na Masarykově univerzitě.

## Výhody

- Upozornění na kolize u všech typů předmětů.
- Nastavení preferencí na čas, buď obecně anebo podle konkrétních požadavků vyučujících na dobu jejich výuky. Na základě požadavků se nastaví preference na předměty, které učí.
- Možnost spojovat více předmětů, kdy například jeden předmět má být v rozvrhu hned po druhém předmětu. Pro FIT se jedná o výhodnou funkci, když potřebujeme například umístit demonstrační cvičení hned za přednášku.
- Vyhodnocování **studentských konfliktů**, které mohou nastat ve dvou případech. První případ nastává, když má student zapsané dva předměty, jejichž časy se překrývají, a druhý je v případě, kdy není dostatek času na přesun mezi předměty. Na FIT často dochází k přesunům na FEKT a čas přesunu je minimálně 15 minut.
- Student se může přihlásit a zvolit si své preference, kterým může a nemusí být vyhoveno. Tato funkce by mohla na FIT vést k větší spokojenosti studentů.
- Téměř všechna data se dají nakonfigurovat.

## Nevýhody

- Dokumentace i software je celý v angličtině. Pro někoho může být složité vyznat se v aplikaci, které nerozumí.

<sup>3</sup>International Timetabling Competition 2007 dostupné na stránkách <http://www.cs.qub.ac.uk/itc2007/>.

- Neintuitivní uživatelské rozhraní. Pro nezaškoleného uživatele je velmi těžké se v aplikaci zorientovat, jelikož obsahuje velké množství konfigurací a aplikace uživatele nikam nenavádí. Nezaškolený uživatel pak neví, co má udělat jako první krok.

Pro potřeby FIT není software ideální, jelikož v rámci fakulty chybí data k vložení. Učitelé často požadavky nevyplňují, jelikož si myslí, že se s nimi počítá z minulých let anebo že si je již rozvrhář pamatuje. Prvním řešením by bylo data zkompletovat pomocí jiného softwaru a pak je vložit do aplikace. To by ale vyžadovalo navržení nového softwaru, který by byl stejně pracný jako samotné plánování. Druhým řešením by bylo zaškolit přes 200 učitelů, aby si požadavky vkládali do aplikace sami. Jelikož často učitelé ignorují vložení požadavků přes jednoduchý formulář v IS, není moc pravděpodobné, že by i přes školení software používali. Požadavky by pak nebyly úplné a generátor by neměl z čeho vycházet.

Dalším problémem je příprava dat pro aplikaci ve vhodném formátu (XML). Vstupní data se nachází na různých místech IS, navíc některá nejsou úplná a je potřeba je manuálně doplnit (například přednáškové skupiny). Bylo by potřeba zaměstnat jednoho člověka pro naprogramování skriptu pro vytvoření XML a dalšího člověka pro kontrolu a doplnění dat.

A i kdyby se softwaru podařilo vytvořit zajímavý rozvrh, je třeba počítat s tím, že s jeho podobou může mít některý vyučující problém a pak je třeba rozvrh lehce upravit. Úprava rozvrhu nesmí porušit tvrdá omezení a měla by respektovat slabá omezení. Z tohoto důvodu přesun jednoho okna může přeskládat celý rozvrh, se kterým pak mohou mít problém jiní vyučující. Pokud bychom si nepřáli hýbat s předměty po přesunutí okna, je možné je zamrazit, pak už ale generátor nemá dost prostoru na lepší umístění nevyhovujícího rozvrhového okna.

Export by pak byl ještě větší komplikací. Na web se nyní vkládají rozvrhy v HTML ještě s dalšími informacemi, jako jsou například tabulky kolizí, povinnosti předmětů, počty studentů u oborů a podobně. Software by nám vrátil rozvrh v XML, který by bylo třeba složitě převést na HTML a ostatní informace doplnit ručně. Zároveň nám ani neulehčí importování nového rozvrhu do informačního systému.

## Roger

Roger [28] je systém pro plánování vyvinutým firmou IS4U v letech 2011-2015, který je volně stažitelný pro omezený počet nahraných dat na URL <https://www.rozvrhy.eu/cs/index>. Jedná se o desktopovou aplikaci, využívající databázi SQL. Pro počet lekcí větší než 500 se ceny pohybují od 75 000 Kč ročně. Automatické nahrání dat je možné pro univerzity mající smlouvu s IS4U, dále je to možné z externí aplikace na plánování aSc Timetables, přes soubor v XML anebo je možné zadat data ručně. Výstup lze pak exportovat do formátu XML nebo XLS.

## Výhody

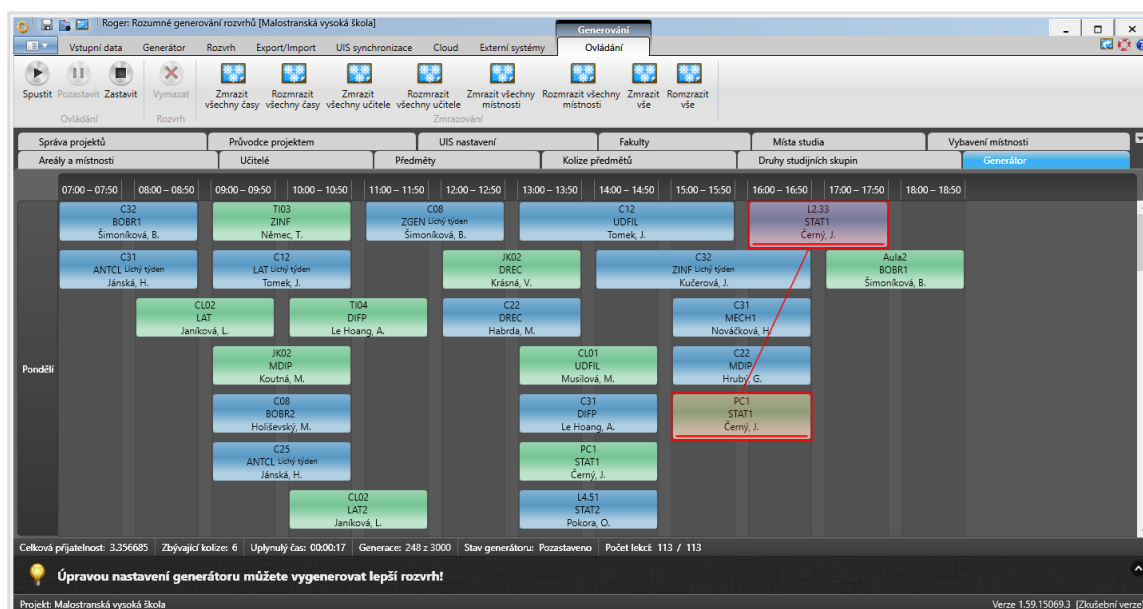
1. Prioritizace konkrétních časových úseků. Výhodné pro nastavení nepreferovaných raných a pozdních hodin z pohledu FIT.
2. Nastavení časových omezení učitelů. Je možné u učitele zadat, kdy může být na fakultě a kdy ne.
3. Automatické generování rozvrhů. Generátor je založen na genetickém algoritmu zobrazujícím průběh generování a číslo generace, která se právě provádí.



4. Zobrazení kolizí spojením předmětů, u kterých kolize nastala.
5. Možnost zamrazení konkrétní akce a pokračování generování.
6. Zobrazení rozvrhu z pohledu místností, dnů a učitelů.
7. Intuitivní uživatelské rozhraní.
8. Program i dokumentace jsou česky.

## Nevýhody

- Nelze zadat požadavek na učebnu u předmětu. V požadavcích učitele může tento požadavek hrát roli pro správné přiřazení do místnosti.
- Nepodporuje plánování rozvrhů zkoušek.
- Nejedná se o levný software.



Obrázek 2.3: Generování rozvrhu v aplikaci Roger. Červeně je zvýrazněna kolize u předmětu STAT1.

Softwaru chybí podpora pro plánování zkoušek, která by pro FIT byla potřeba. Navíc se nejedná o levný software a jeho údržba by stála každý rok školu peníze. Když si spočítáme, že plánování zabere dvěma lidmi na FIT zhruba jeden měsíc na semestr (pro zjednodušení budeme počítat výuku a zkoušky dohromady), přičemž je hrubý měsíční průměrný plat vysokoškolského pedagoga 28 508 Kč [24], zaplatí za ně škola 57 016 Kč na semestr. Rok má dva semestry, což dává 114 032 Kč. Musíme však brát v úvahu, že software neudělá celou práci za ně. Platí zde stejné nevýhody jako u UniTime popsaného v předchozí sekci 2.1. Samotnou přípravu dat, kompletaci požadavků, export a import do IS by musel někdo připravit a čas bychom si proto moc neušetřili. Ve výsledku bychom zaplatili software plus měsíční práci rozvrháře a řešení by vyšlo školu minimálně na 132 016 Kč, a to počítáme

i s možností, že by software uměl plánovat zkoušky. Software by tedy neušetřil práci ani peníze, a proto není vhodný pro potřeby FIT.

## aSc Timetables

Tento program je primárně určený pro základní a střední školy. Jeho demoverze je volně stažitelná na adrese [https://www.asctimetables.com/timetables\\_cz.html](https://www.asctimetables.com/timetables_cz.html). Zakoupení verze PRO s možností individuálního rozvrhu dle potřeb pro každého studenta na dva roky s uživatelskou podporou stojí 23 900 Kč. Všechna data lze importovat ze souboru ve formátu XML nebo ze schránky (clipboard) pomocí Excelu. Výstupy jsou možné ve formátu PDF, HTML, XML nebo XLS. Systém nabízí automatické plánování na základě zadaných preferencí, opravy jsou kdykoli možné, stejně jako zamykání rozvrhových akcí. Systém podrobněji analyzuje Monika Kubalcová ve své práci Porovnání programů pro plánování rozvrhů a zkoušek [13]. Z její analýzy vyplývá, že systém není ideální pro potřeby FIT, jelikož nelze zadat všechny podmínky FIT jako je například propojení místností. Na druhou stranu vyšel z její analýzy jako jediný kandidát pro možné nasazení FIT.

## Metro NG

Metro NG je software od firmy ERUDIO s.r.o. Dodává se jako jeden spustitelný soubor (.exe) určený pro operační systémy Windows Vista, Windows 7 a Windows 8. Neexistuje žádná demoverze ke stažení, ale online dokumentace je dostupná na webových stránkách [http://www.erudio.cz/doc/?apl=rozvrhng\\_metro](http://www.erudio.cz/doc/?apl=rozvrhng_metro). Program neumí automatické plánování rozvrhů, ale pouze poskytuje podporu pro plánování. V současnosti používá rozvrh například Česká zemědělská univerzita v Praze (informaci poskytl Ing. Jiří Mach, technickohospodářský pracovník ČZU), která má systém, propojený s informačním systémem. Jejich princip generování nového rozvrhu spočívá v tom, že vždy dělají pouze změny oproti minulému roku a rozvrh se pouze drobně upravuje na základě požadavků pedagogů a změn ohledně počtů studentů. Další školou využívající tento software je Univerzita Karlova, z jejíhož Výpočetního centra společnost ERUDIO s.r.o. v roce 1994 vznikla [6].

Systém umožňuje podporu rozvrhování pravidelné i nepravidelné výuky a nahlížet na rozvrhová okna z různých úhlů pohledu jako je například pohled výuky na celý semestr, týdenní pohled, zobrazení rozvrhu konkrétních učeben a podobně. Umí vyhodnocovat kolize, které dělí do několika skupin – přejezdové kolize, kolize učitele a kolize studentů. Co se týká požadavků učitelů, software zobrazuje, jak moc jim bylo vyhověno.

Jelikož není možné stažení softwaru, můžeme čerpat pouze z informací poskytnutých v dokumentaci. Na první pohled se zdá být jedním z největších problémů předpoklad vyplnění všech rozvrhových oken pro instalaci produktu. Jelikož se na FITu délka okna řídí požadavkem vyučujícího, nebo se počítá na základě celkového počtu rozvrhových akcí za semestr, není možné tato data poskytnout. Zkompletování a import dat by pak byly celkově náročnější než plánování za současného stavu, proto není systém vhodný pro požadavky FIT.

## 2.2 Průběh plánování na FIT

Fakulta informačních technologií Vysokého učení technického (dále jen FIT) je založena dle Boloňské deklarace<sup>4</sup> na trojstupňovém vysokoškolském studiu. Základním stupněm je baka-

<sup>4</sup><http://www.msmt.cz/vzdelavani/vysoke-skolstvi/bolonsky-proces-2>



Obrázek 2.4: Aktuální proces plánování na FIT

lářské studium s délkou 3 roky, na které navazuje dvouroční magisterské studium a vynikající studenti pak mohou dále pokračovat třetím stupněm doktorského studijního programu. Tyto studijní programy splňují podmínky ECTS (European Credit Transfer System<sup>5</sup>). Popis jednotlivých programů lze najít na webových stránkách fakulty<sup>6</sup> [22].

Předměty jsou rozděleny do tří skupin. První skupinou jsou předměty *povinné*, které je nutné splnit všechny. Druhou skupinou jsou předměty *povinně-volitelné*. Často jsou tyto předměty rozloženy do skupin a je potřeba si z nich vybrat alespoň jeden. Poslední skupinou jsou *volitelné* předměty, které dovolují studentovi zaměřit se na oblast, která ho zajímá. Pro absolvování studia je vždy potřeba si vybrat i některé volitelné předměty pro získání požadovaného počtu kreditů, který je u bakalářského programu stanoven na 180 kreditů, u magisterského studia stačí pouze 120 kreditů. Doktorského studijního programu se dané členění netýká, jelikož je zde pouze jedna skupina předmětů, ze které si student vybírá a následně se pro něj sestavuje individuální studijní plán.

Průběh plánování je zobrazený na obrázku 2.4. Plánování probíhá v několika fázích. V první fázi je potřeba sesbírat veškerá vstupní data, která budou při plánování potřeba. Jelikož není možné přímo přistupovat k fakultní databázi, musí se stahování dat dělat přes různé aplikace v IS FIT, které obsahují pohledy na tabulky.

Současná aplikace zobrazená na obrázku 2.5 potřebuje ke správnému vyhodnocení každý semestr aktuální data. Stahují se proto kolize předmětů, počty studentů u oborů a v případě, že se plánuje zkouška, stahují se i počty studentů na termíny z minulého roku. Data

<sup>5</sup>Více o ECTS na webových stránkách: [https://ec.europa.eu/education/resources-and-tools/european-credit-transfer-and-accumulation-system-ects\\_en](https://ec.europa.eu/education/resources-and-tools/european-credit-transfer-and-accumulation-system-ects_en)

<sup>6</sup><http://www.fit.vutbr.cz/study/programs/index.php.cs>

jako jsou předměty, povinnosti předmětů, obory, programy a přednáškové skupiny jsou zakódována přímo v aplikaci. Je proto potřeba data řádně zkontrolovat, jestli se nezměnila oproti minulému roku.

V dalším kroku se data kompletují, jelikož je není možné všechna stáhnout. Typickým příkladem jsou požadavky vyučujících, které nejsou úplné a vždy je potřeba rozvrháře, aby si potřebná data dohledal z minulých let.

Po kompletaci dat je možné přijít k samotnému plánování, ve kterém se vychází z tvrdých a měkkých omezení popsaných v sekci 2.1 a hledá se kompromis mezi požadavky vyučujícího, studenta a dostupných kapacit místností. Plánování obvykle trvá 2 – 4 dny.

The image shows a screenshot of a course scheduling application. The main part of the screen is a grid representing a weekly timetable. The columns represent time slots from 07:50 to 20:50 in 15-minute increments. The rows represent days of the week, with two weeks shown: Po (Monday) and Út (Tuesday). Each cell in the grid contains a course code (e.g., DM (a), DA (b), ZP (a), etc.) or is empty. To the right of the grid, there are columns for course details, including 'Společná studium' and various course codes like BIT, MBI, MBS, etc. Above the grid, there are several filter buttons: 'Kolize', 'Bez kolize', 'Kolize s okolím', 'Kapacity', 'Naplněnost', 'Tabulka', and 'FR'. The interface is color-coded, with green cells indicating scheduled courses and other colors for different statuses.

Obrázek 2.5: Ukázka současné aplikace na plánování rozvrhu výuky

Když je rozvrh hotový, přichází fáze připomínkování Studenské unie FIT a studenty na sociálních sítích (1 – 3 dny). V tento moment je prostor na různé změny. Nikdy nebudou všichni spokojeni, ale je možné, že sto studentů může vymyslet kombinaci, kterou jeden člověk s podporou počítače nemusí být schopen objevit [5]. Po připomínkování studentů dochází k připomínkování ze strany vyučujících, které trvá zhruba 2 týdny. V poslední fázi se rozvrh vloží do IS FIT, kde jsou již změny náročnější [5]. Jednotlivé kroky budou blíže popsány v následujících kapitolách zvláště pro výuku (sekce 2.2.1) a zkoušky (sekce 2.2.2).

## 2.2.1 Plánování výuky

Plánování výuky začíná v akademickém roce pro zimní semestr koncem měsíce dubna až začátkem května a pro letní semestr se řeší rozvrhy o 4 týdny později. Délka výuky je stanovena na **13 kalendářních týdnů** podle Rozhodnutí č. 24/2017 platného od 1.5.2017 [29]. Vyučování probíhá v čase od **7:00 do 20:50**. Plánuje se jak **pravidelná** výuka přednášek a cvičení, tak **nepravidelné** akce jako je například školení k provoznímu řádu CVT.

## Příprava dat pro plánování

K plánování je potřeba velké množství vstupních dat, která přichází z různých zdrojů.

- Nejdůležitějším zdrojem je **seznam předmětů**. U každého předmětu potřebujeme informaci o počtu hodin na jednotlivé aktivity, typu zakončení předmětu (např. Zk pro označení ukončení předmětu zkouškou), kapacitě, počtu kreditů. Dále je potřeba znát obecné informace jako je název, zkratka, typ předmětu a počet registrovaných studentů. Rozvrhové akce můžeme rozdělit na dva typy: přednášky a cvičení, která se ještě dělí na demonstrační, numerická, laboratorní a počítačová. Přednášky i demonstrační cvičení probíhají v přednáškových místnostech a jsou určené pro větší počet studentů. Existují i další typy cvičení, které nespadají do procesu plánování, ale termíny si určují jednotliví učitelé.
- Předmět má vždy jednoho **garanta** a jednoho nebo více **přednášejících** a **cvičících**. Z hlediska plánování je potřeba dát pozor na to, aby jeden vyučující neučil v jeden okamžik více než jeden předmět. Každý vyučující je veden na daném ústavu, kterému přísluší daná budova na fakultě. Z hlediska plánování je pak potřeba přihlédnout i na vzdálenost budov od sebe, aby bylo vždy možné se v časové pauze mezi výukou dostat na přednášku či cvičení.
- Společně s předměty se stahují i **povinnosti předmětů**. Vždy se jedná o čtveřici: předmět, typ povinnosti, obor a ročník, na základě kterých se pak generuje výstupní graf povinností v současné aplikaci na plánování (popis grafu se nachází v sekci 2.2.2). Tato informace je důležitá i z pohledu práce s tvrdými a měkkými omezeními, kdy například kolizi mezi volitelnými předměty nepovažujeme za vážný nedostatek rozvrhu.

Tabulka 2.1: Tabulka důležitých místností a jejich kapacit

Místnost	Kapacita	Kapacita ob 1	Kapacita ob 2
D105	300	150	112
D0206	154	77	59
D0207	90	45	36
E112	156	78	54
E104	72	37	28
E105	72	37	28
G202	80	40	32
A112	64	32	24
A113	64	32	24

- Abychom věděli, kam předměty umístit do rozvrhu, musíme znát **seznam místností** na fakultě a jejich kapacity. Pro potřeby zkoušek je třeba evidovat i kapacity místností pro obsazení přes jedno nebo dvě místa. Aktuálně se jedná o 9 nejdůležitějších místností, které se používají pro plánování. V tabulce 2.1 naleznete seznam těchto nejdůležitějších místností společně s jejich kapacitami v zařitém pořadí.

Na FIT je možnost přednášky *streamovat* do jiných místností a tímto dosáhnout větší kapacity místnosti. Všechna možná propojení jsou zobrazena v tabulce 2.2.

Tabulka 2.2: Kombinace místností a jejich výsledná kapacita

Kombinace	Kapacita
D105 + D0206 + D0207	544
D105 + D0206	454
D105 + D0207	390
D0206 + D0207	244
E112 + E104 + E105	300
E112 + E104	228
E104 + E105	144
A112 + A113	128

- Pro správné umístění předmětu je třeba znát **kolize** s jinými předměty. Jedná se o počty společných studentů u dvojice předmětů. Je to jeden z hlavních problémů, který se řeší u plánování.
- Při plánování bereme v úvahu **požadavky** od vyučujících. Evidujeme u nich počet hodin, ze kterého vycházíme při plánování, preferovanou místnost, která se vyplňuje jako textové pole a informace je pouze orientační. Dále jsou to informace jako je četnost a typ výuky (např. přednáška, cvičení), požadovaný počet skupin a časy od kdy, do kdy nemůže či může vyučující učit. Může se stát, že učitel nevyplní požadavek na výuku, pak je potřeba požadavek dohledat z minulého roku a na základě svých zkušeností posoudit, jestli se požadavek doplní do aktuálního akademického roku.

Obvyklý počet přijímaných studentů do bakalářského programu je cca 650. Při nahlédnutí do tabulky kombinací místností 2.2 lze vidět, že neexistuje spojení místností, které by dohromady dávalo alespoň 650 míst, proto je potřeba studenty rozdělit do skupin. V prvních dvou ročnících jsou studenti rozděleni podle abecedy do dvou skupin *BIA* a *BIB*, které mají oddělené přednášky a cvičení. Z hlediska plánování je důležité plánovat přednášky dvakrát pro každou skupinu a myslet na to, že pokud existuje událost pro jednu skupinu, která má nenulový počet společných studentů s jinou událostí, nemusí se jednat o problém. Student z jedné skupiny může kdykoli přijít na přednášku skupiny druhé a kolizi se vyhnout.

### Kompletace požadavků vyučujících

Proces začíná sběrem požadavků od učitelů, který trvá asi 2 týdny. Každý učitel vyplňuje dny a časy od kdy do kdy může učit, poznámku a jakou tomuto požadavku dává prioritu. Dále jsou vyplněny důležité informace jako je požadovaná kapacita místností, počet hodin, jak často bude výuka probíhat, jaké preferuje místnosti a poznámka učitele. Z těchto informací se pak vychází při plánování.

Požadavky se zadávají před každým plánováním a obvykle jim vyučující nevěnují takovou pozornost, jakou by měli. Jelikož se některé požadavky opakují každý rok, nepříjde vyučujícímu potřebné je znovu vložit do systému. Pak vznikne problém nedostatku potřebných dat nutných pro naplánování výuky a zkoušek. Problém se řeší nakopírováním některých požadavků z minulých let. Požadavky se kompletují v programu Microsoft Excel, jak je zobrazeno na obrázku 2.6. Využívá se zde několika barev:

- Zeleně jsou obarveny předměty, které jsou již naplánované.
- Žlutá slouží pro předměty, které se neplánují.

- Červená slouží pro buňky, které byly během plánování porušeny.
- Sytě fialová je určená pro preference vyučujících, ze kterých slevili.
- Světle žlutá barva rozlišuje požadavky nakopírované z výuky z minulého roku stejného semestru.
- Tmavě zelenou barvou oddělujeme požadavky z minulého semestru.
- Modrá slouží pro naposledy přidávané požadavky na zkoušky.

Předmět	P pro	PV pro	Počet	Max.	Garant	Přednášející	Vyučující	Typ	Skupina	Kapacita	Hodiny	Týden	Místnost	Poznámka	Vyhovoval by mi termín				Nemohu učit				
															Den	Od	Do	Prio	Poznámka	Den	Od	Do	Poznámka
BIUX		MGM, MIS	30	64	Lampa Petr, Ing.	Lampa Petr, Ing.	Lampa Petr, Ing.	přednáška	1	60	2	každý			středa	8:00	9:50	low		úterý	13:00	16:50	
HSC		MGM, MPV, MSK 1	14	200	Fučík Otto, doc. Dr. Ing.	Fučík Otto, doc. Dr. Ing.	Fučík Otto, doc. Dr. Ing.	přednáška	1	150	5	každý								-	7:00	7:50	dříve studenti nechodí
		MBI, MBS, MIN, MMI																	pondělí	7:00	20:50	schůzka projekat.	
																			úterý	12:00	20:50	senát	
																			čtvrtek	7:00	11:50	kolegium děkana	
																			pátek	13:00	20:50	fakultní semináře	

Obrázek 2.6: Kompletace požadavků probíhá v programu Microsoft Excel

## Plánování

Proces začíná vložení pevných rozvrhových akcí, které FIT neovlivní. Jedná se o akce plánované na jiných fakultách anebo jednorázové akce organizované školou. Samotné plánování začíná umístěním předmětů, které mají nejvíce požadavků od vyučujících. Tyto předměty by pak bylo velice obtížné do rozvrhu zařadit, aby byla splněna přání vyučujících na výuku.

V druhé fázi se umísťují předměty s největším počtem přihlášených studentů, kterými jsou typicky předměty pro první ročník bakalářského studia. Na rozdíl od zkoušek se pak pokračuje s plánováním předmětů sestupně podle počtu studentů. Dochází proto ke střídání umísťování bakalářských a magisterských předmětů.

Kvůli omezenému počtu místností na fakultě je potřeba v čase 8 – 17 hodin plánovat pro většinu lidí a není možné si dovolit vynechat volný jediný časový úsek. Při kontrolách je potřeba nejprve zjistit, zda se nepřekrývají přednášky, které vyučuje stejný vyučující, a následně se kontroluje, zda-li se některému studentovi nepřekrývají dva povinné předměty. U magisterských předmětů se jedná o celkem složitou kontrolu, jelikož učitel vidí v aplikaci pouze počet společných studentů u předmětu, ale nedokáže zjistit, kteří z nich mají předmět povinný. Například se zobrazí *FLP EVO 12*, což znamená, že předměty se zkratkou FLP a EVO mají 12 společných studentů. Pokud by se jednalo o 12 studentů, kteří mají oba tyto předměty volitelné nebo alespoň jeden z nich je volitelný, tak to není dobré, ale nejedná se o fatální problém. Problém nastává v případě, kdy by se našli studenti, kteří by tyto dva předměty měli povinné. K usnadnění vyhledávání povinností pro jednotlivé obory slouží v plánovací aplikaci graf povinností (popis grafu se nachází v sekci 2.2.2). Další nástroje sloužící k ulehčení práce rozvrháře se nachází v pravém horním rohu aplikace (obr. 2.5) a jsou podrobněji vysvětleny v tabulce 2.3.

Tabulka 2.3: Popis nástrojů současné aplikace, které se nachází v horním rohu aplikace

<b>Tlačítko</b>	<b>Popis funkce</b>	<b>Spolehlivost</b>
<i>Kolize</i>	Funkce umožní zadat dvojici předmětů, u kterých následně zobrazí počet společných studentů.	ano
<i>Bez kolize</i>	Po zadání zkratky předmětu se vypíše seznam předmětů, které s daným předmětem nemají žádného společného studenta.	ano
<i>Kolize s okolím</i>	Podle aktuální pozice kurzoru se zobrazí seznam společných studentů pro každou dvojici předmětů, které jsou naplánovány dva dny před a dva dny po aktuálním dni. V seznamu nejsou ošetřené duplicity a pokud přesahuje jednu stránku, není možné se dostat na konec.	částečně
<i>Kapacity</i>	Po zadání zkratky předmětu se vypíše počet studentů pro příslušný termín. Zaškrtnutím jednotlivých místností hledáme takovou kombinaci, která nám poskytne dostatečnou kapacitu pro požadované rozsazení studentů.	ano
<i>Naplánovat</i>	Funkce zobrazí předměty, které ještě nejsou naplánovány. Data nezaručují správnost.	ne
<i>Tabulka</i>	Funkce zobrazí možnost vložení nových řádků do tabulky.	ano
<i>FR</i>	Funkce filtruje předměty v rozvrhu pouze pro daný ročník. Filtrování funguje pouze 1x, pokud chceme filtrovat podruhé, je potřeba aktualizovat stránku.	částečně

## Export rozvrhu

Export je poslední fází vytváření rozvrhů, ke které dochází v momentě, kdy je rozvrh připraven na zveřejnění. Může k němu docházet i opakovaně, a to v případě, že se ukáže nějaká chyba, anebo přijde reklamace ze strany vyučujícího nebo studenta. V případě reklamace od studenta musí osoba odpovědná za vytváření rozvrhu rozhodnout, jestli je důvod změny vážný a jestli změna nezpůsobí větší problémy než vyřeší.

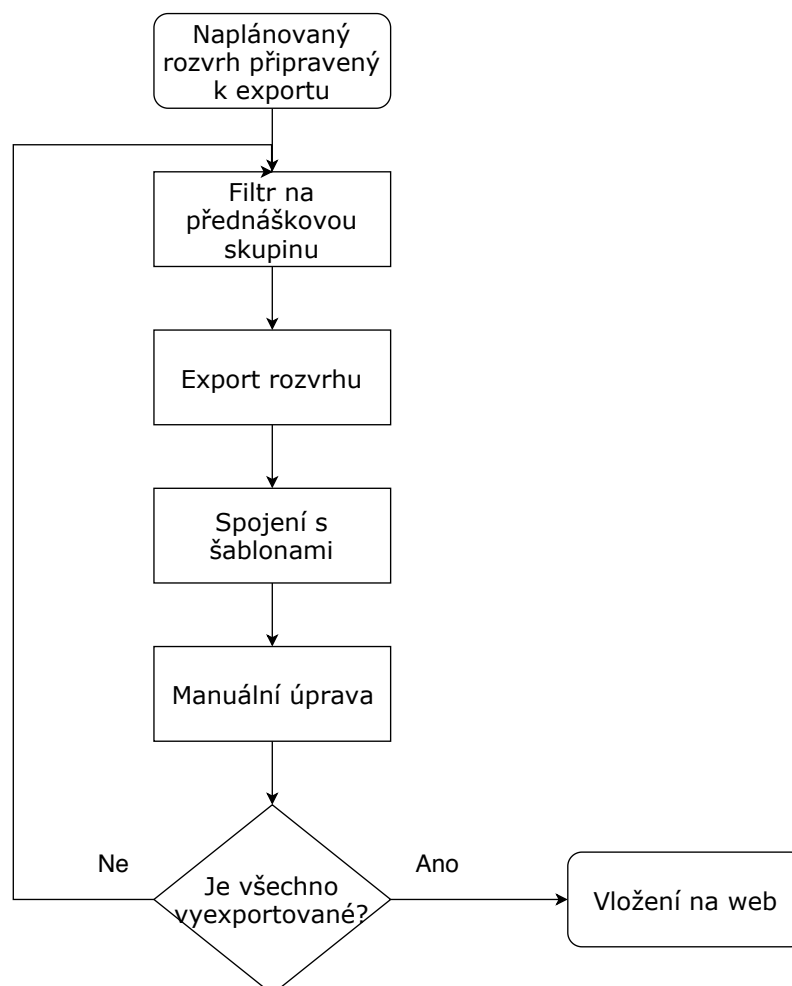
Export probíhá ve více fázích zobrazených na obrázku 2.7. Každá přednášková skupina nebo jejich kombinace se exportuje zvlášť. Využívá se funkce FR (vysvětlení v tabulce 2.3) zabudované v aplikaci. Vždy je potřeba nejprve vyfiltrovat rozvrhová okna pro přednáškovou skupinu a následně je vyexportovat (tlačítko *Export všech souborů*, které lze vidět na obrázku 2.8). Výsledkem bude soubor exportovaných dat (většinou fragmentů stránek v HTML), který bude kromě rozvrhu obsahovat:

- tabulku společných studentů – dvojice předmětů s počtem společných studentů,
- tabulku s počty studentů u oborů – počty studentů u konkrétního předmětu a oboru,
- tabulku místností – důležité místnosti s kapacitami, nachází se zde i kapacita ob jedno a ob dvě místa,
- přehled předmětů a jejich povinností u oboru,



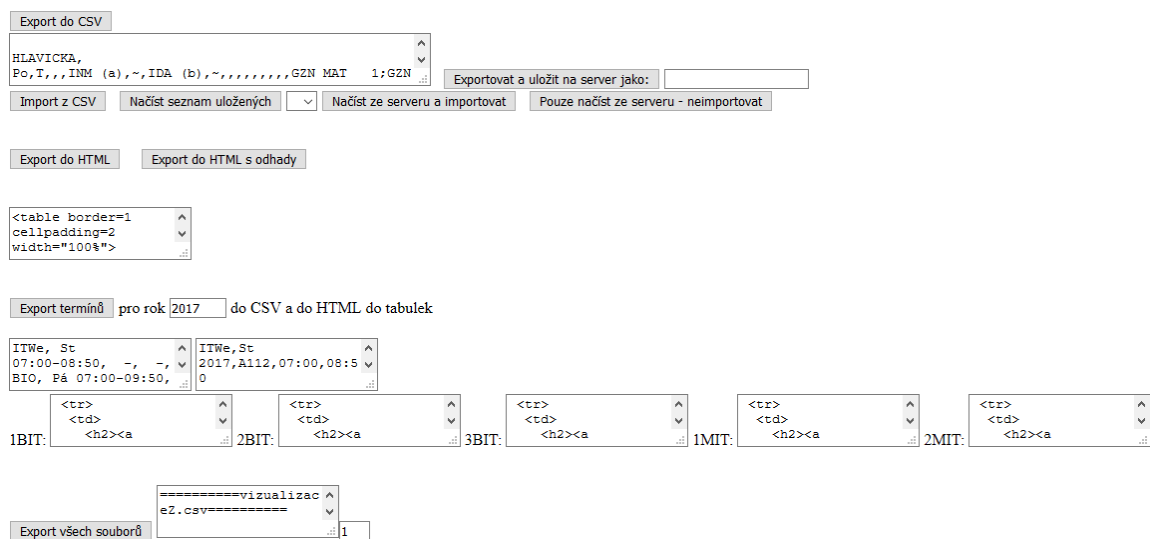
- rozvrh ve formátu CSV.

Tento výsledný textový soubor se pak musí rozdělit na další pomocné soubory. Rozdělení probíhá podle zarážek vložených ve vyexportovaném souboru pomocí skriptu *rozděl.php*. Po rozdělení se vzniklé soubory musí **spojit s již předdefinovanými šablonami** (pomocí *Makefile*). Nakonec se přidá aktuální datum a čas (skript *nastav\_data.sh*), upraví se nastavení kódování (skript *hackni\_pro\_Terezu.sh*) a rozvrh je připraven pro vložení na web.



Obrázek 2.7: Průběh současného exportování stránek na web

Proces se provádí pro minimálně deset přednáškových skupin (*1BIA*, *1BIB*, *2BIA*, *2BIB*, *3BIT*, *1MIT*, *2MIT*, *1EIT*, *2EIT* a *INTE*). Pokud se později ukáže nějaká chyba v rozvrhu, je potřeba celý proces exportování opakovat. Kromě toho se exportuje kompletní rozvrh pro všechny skupiny a stránka, která obsahuje odkazy na vytvořené stránky, jméno osoby, která rozvrh připravila, a datum poslední změny rozvrhu.



Obrázek 2.8: Ukázka exportování používaná v původní aplikaci

## Import do IS

Stránky se umístí na web a přichází fáze připomínkování od studentů a následně připomínkování od učitelů. V případě, že již nejsou žádné připomínky k návrhu rozvrhu, dochází k vložení dat do IS FIT. U každého rozvrhového okna se vyplňuje: čas od kdy, do kdy, den v týdnu, semestr, rok, typ výuky, týden semestru (sudý, lichý), místnost, předmět, přednášková skupina a v některých případech skupina od a skupina do. Jedná se o několik položek, které musí ručně vyplnit osoba odpovědná za zadávání dat do IS pomocí zadávacího formuláře. Tato osoba musí znát přesně pravidla pro zadávání a musí být velice pozorná, aby během zadávání neudělala chybu. Nejtěžšími položkami k přidávání jsou právě přednášková skupina a rozsah skupin od, do. Následující seznam pravidel udává, jakým způsobem se rozvrhová okna vkládají do IS:

1. Pokud je v ročníku **více přednáškových skupin**, rozvrhová akce se vkládá pro každou přednáškovou skupinu v případě, že není u okna přímo specifikována konkrétní skupina. Například v prvním ročníku bakalářského studia se nachází dvě přednáškové skupiny: *1BIA* a *1BIB*, všechny rozvrhové akce pro první ročník se proto budou vkládat do systému minimálně 2×, pokud nebude v rozvrhovém okně upřesnění (*a*) pro BIA nebo (*b*) pro BIB.
2. Pokud je předmět **povinný** v X-tém ročníku, musí se rozvrhové okno zadat i pro následující ročník. Pokud student nezvládne předmět jeden rok, bude ho mít automaticky zapsaný následující rok. Takové rozvrhové akce se značí skupinou od: *xx* a skupinou do: *xx*.
3. **Upřesnění přednáškové skupiny** u rozvrhové akce se vztahuje pouze na ročníky, kde je předmět **povinný**. V případě, že bude například u okna upřesnění *skupina A* a předmět bude povinný pro 1. ročník, vloží se jeden záznam pro *1BIA* a dva záznamy s rozsahem *xx* pro *2BIA* a *2BIB* podle druhého pravidla tzn. *skupina A* neovlivní vložení přednáškové skupiny *2BIB* do rozvrhu.

4. Pokud je předmět **povinný v bakalářském programu** a jedná se o **přednášku**, tak se nevkládá rozsah skupin, pro které je akce určena, jelikož jsou implicitně zahrnuty všechny skupiny. Mějme například předmět IDA, který je povinný v prvním ročníku bakalářského studia. Pro jeho přednášku se vloží celkem 4 záznamy do IS. V prvním záznamu se nachází skupina *1BIA* bez specifikovaného rozsahu skupin od a do. Druhý záznam je stejný, ale se skupinou *1BIB*. Ostatní dva záznamy se přidávají podle pravidla číslo dvě.
5. Pokud je předmět **povinný v bakalářském programu** a jedná se o **demonstrační cvičení**, vkládá se buď rozsah skupin specifikovaný u rozvrhového okna anebo se vezme výchozí rozsah skupin u přednáškové skupiny, který je závislý od počtu studentů v ročníku. Pokud je například rozsah *1BIA* 10 – 24 a *1BIB* 30 – 44, vloží se pro cvičení předmětu IDA čtyři záznamy do systému. Jeden záznam bude kromě obecných informací o rozvrhové akci obsahovat skupinu *1BIA*, skupina od: 10 a skupina do: 24 a druhý záznam bude obsahovat kromě obecných informací skupinu *1BIB*, skupina od: 30 a skupina do 44. Pokud by u rozvrhového okna bylo specifikováno, že se jedná pouze o *1BIA* a pro rozsah 10 – 11, vložil by se pouze jeden záznam se skupinou *1BIA*, skupinou od 10: a skupinou do: 11. Ostatní dva záznamy se přidávají podle pravidla číslo dvě.
6. Pokud je předmět **povinný v bakalářském nebo magisterském studijním programu** a jedná se o **jiný druh výuky než přednáška a demonstrační cvičení**, vkládá se buď rozsah skupin specifikovaný u rozvrhového okna, nebo se vloží záznam se skupinou od: *xx* a skupinou do: *xx*.
7. Pokud je předmět **povinný v magisterském programu** a jedná se o **přednášku nebo demonstrační cvičení**, je potřeba zjistit všechny obory, ve kterých je předmět povinný, a číslo oboru s názvem oboru vložit do rozsahu skupin. Například předmět MAT je povinný v prvním ročníku magisterského studia pro obory: 11 MBS, 12 MGM, 13 MIN, 14 MIS, 15 MMI, 16 MMM, 17 MPV, 18 MSK. Aby se nevkládalo osm záznamů pro každý obor, vloží se pouze jeden záznam s celým rozsahem, kde *skupina od* bude 11 MBS a *skupina do* 18 MSK pro přednáškovou skupinu 1MIT.

### 2.2.2 Plánování zkoušek

Předmět může být zakončený zkouškou, zápočtem nebo kolokviem. Kolokviem se rozumí ukončení předmětu rozpravou o problematice předmětu, případně vypracováním písemné práce, která se zabývá dílčí tematikou předmětu. Zkouška může být písemná, ústní nebo kombinovaná, slouží k ověření vědomostí studenta. Výsledek zkoušky klasifikuje zkoušející bodovým hodnocením. Pro úspěšné absolvování zkoušky může garant předmětu stanovit minimální bodové hodnocení, které musí student dosáhnout (jinak za ni nezískává bodové hodnocení). Student může konat zkoušku ze zapsaného předmětu nejvýše třikrát, má tedy maximálně dva opravné termíny, přičemž předmět zakončený zkouškou má buď tři termíny, nebo pět (a více) termínů. V případě, že má zkouška pouze tři termíny, musí kapacita termínu uspokojit všechny registrované studenty. V případě pěti termínů platí pravidlo, že celková kapacita musí být minimálně 1,5 násobek počtu registrovaných studentů předmětu. Podmínky skládání zkoušek definuje *Studijní a zkušební řád Vysokého učení technického*

v Brně platný k 1. 9. 2018<sup>7</sup>, který je doplněný dodatkem č. 1<sup>8</sup>. Je zde definována délka zkuškového období na **5 týdnů**. Zkoušku většinou předchází zápočet, který je udělován za splnění daných požadavků určených garantem pro příslušný předmět. Uděluje jej vyučující předmětu.

## Vstupní data potřebná pro plánování

Stejně jako u výuky i u zkoušek je potřeba celá škála vstupních dat potřebná k plánování. Z většiny se jedná o stejná data jako jsou uvedena v sekci 2.2.1, u zkoušek navíc přibudou počty studentů na zkoušku z minulého roku a pevně dané termíny.

- **Seznam předmětů** společně s **povinnostmi** a informacemi, kdo jsou **vyučující** a **cvičící** předmětu. Z pohledu plánování zkoušek je zajímavé rozdělení předmětů na dva typy:
  - **centrálně plánované** – centrální plánování fakultou,
  - **necentrálně plánované** – plánuje učitel nebo daná fakulta podle sebe, z pohledu FIT se jedná o rozvrhové akce, které mají **přesně daný čas**, který je potřeba v rozvrhu dodržet.
- **Seznam místností** a jejich kapacit, ve kterých se mohou zkoušky konat.
- **Kolize** s jinými předměty udávající počet společných studentů u dvojice předmětů.
- **Požadavky** od vyučujících na rozdíl od požadavků na výuku obsahují:
  - Požadavek na **rozsazení** během zkoušky, kde má učitel na výběr ze tří možností. Buď mohou studenti sedět vedle sebe, přes jedno místo a nebo přes dvě místa.
  - Požadavek na **počet kol**, který určuje, kolikrát se zkouška bude psát v jednom termínu. Pokud je vyplněno pouze jedno kolo, píše všichni studenti zkoušku naráz, pokud se jedná o dvě kola, studenti jsou rozděleni na poloviny (či menší skupiny) a vykonávají zkoušku v jiný čas. Při určování délky rozvrhového okna se počtem kol násobí doba trvání zkoušky.
  - **Doba trvání zkoušky** je různá a je třeba počítat i s úvodem, kdy vyučující vysvětluje pokyny, a s dobou, kdy se rozdávají písemky, anebo probíhá přesazování studentů.

Jinak se evidují stejné údaje, kde místo typu výuky se logicky vyplňuje termín (předtermín, řádný termín, první oprava apod.) a nezadávají se pouze časy, ale také data od, do, kdy by vyučující chtěl, anebo nemůže zkoušet.

- U zkoušek je potřeba znát **počty studentů**, kteří přišli na určitý termín minulý rok. Na základě této tabulky a počtu registrovaných studentů aktuálního roku se odhaduje, kolik přijde zhruba studentů na jednotlivé termíny. Informace slouží pro odhad kapacity určitého termínu zkoušky, pro učitele, kteří si připravují zadání, i pro kuchařky, aby věděly, kolik připravit obědů v daný den.

<sup>7</sup>Úplné znění studijního a zkušebního řádu VUT lze najít na <https://www.vutbr.cz/uredni-deska/vnitri-predpisy-a-dokumenty/-d149085/uplne-zneni-studijniho-a-zkusebniho-radu-od-1-9-2018-p163172>

<sup>8</sup>Dodatek č. 1 ke studijnímu a zkušebnímu řádu VUT lze najít na <https://www.vutbr.cz/uredni-deska/vnitri-predpisy-a-dokumenty/dodatek-c-1-d174049/dodatek-c-1-p163171?armsgt=1U4azkk6bB>



v nichž jsou vypsané pouze tři termíny. K této kontrole nám velmi poslouží používaný graf v plánovací aplikaci, u kterého cílíme na to, aby byl co nejplošší a aby se v něm neobjevovaly žádné ostré vrcholy. Podrobněji je graf popsán v následující podkapitole.

## Graf s povinnostmi

Na pravé straně od sloupce *Společní studenti* v aplikaci lze vidět graf povinností předmětů (obr. 2.5). Na horizontální ose jsou zobrazeny zkratky oborů a pod nimi jednotlivé ročníky. Vybarvený obdélník v grafu značí, že se v daném dni vyskytuje předmět s povinností pro konkrétní ročník a obor. Povinnosti jsou od sebe odděleny barvami. Ze zkušeností rozvrháře se osvědčilo v grafu zobrazovat **počet povinných předmětů** (zelená barva jako je u rozvrhových oken), **počet povinně volitelných předmětů** (tmavě zelená) a **počet povinných předmětů pro všechny ročníky v oboru** (světle zelená barva). Oranžová barva znamená **cvičení**. Povinnost povinně volitelná pro všechny obory se nezobrazuje, stejně tak jako počty u volitelných předmětů. Graf by byl potom nepřehledný. Políčka se dynamicky mění podle toho, jak posouváme okna.

## Export zkoušek

Export probíhá stejným způsobem, jak je popsáno u výuky (2.2.1). U zkoušek se jedná o export minimálně devíti webových stránek. Pět stránek obsahuje tabulky časových plánů zkoušek pro hlavní přednáškové skupiny dle vzoru Ing. Eysselta, kterými jsou *1BIT (1BIA + 1BIB)*, *2BIT (2BIA + 2BIB)*, *3BIT*, *1MIT*, *2MIT*. Ukázka časového plánu zkoušek pro přednáškovou skupinu 3BIT je zobrazená na obrázku 2.10. V prvním sloupci se nachází termíny půlsestrální zkoušky, které nejsou vyplněny. Jelikož tabulka vzniká až po termínech půlsestrálních zkoušek, je tento sloupec zbytečný. Na posledním řádku jsou vždy uvedeny odkazy do časových plánů jiných přednáškových skupin, jelikož se nacházejí na jiných URL. Výběr předmětů, který se v tabulce nachází, je historicky zavedený. Předměty jsou v tabulce seřazeny podle termínů řádné zkoušky. Pod tabulkou můžeme najít texty s obecnými informacemi ke zkouškám pro daný rok. Jedná se o informace, jako je upozornění pro studenty, že nevyužití termíny propadají bez náhrady.

Kromě tabulek termínů zkoušek se exportují i rozvrhy ve třech pohledech:

- stránka s kompletním rozvrhem zkoušek bez místností (*index.html*),
- stránka s kompletním rozvrhem zkoušek s místnostmi (*mistnosti.html*),
- stránka s kompletním rozvrhem zkoušek s místnostmi a odhady počtů obědů pro menzu (*menza.html*) – jedná se o export sloužící kuchařkám, aby věděly, kolik připravit na daný den obědů.

Poslední vyexportovaná stránka obsahuje **seznam předmětů s odhady počtů studentů** na řádný termín, první opravný, druhý opravný apod. Tento export není veřejný pro studenty. Pokud by byl soubor veřejný, mohlo by se stát, že si studenti budou vybírat předměty na základě počtu studentů na druhém a třetím termínu, nebo se na jednotlivé termíny podle těchto počtů budou připravovat. Pokud je tento počet malý, student z něj pak nemá strach a zapíše si jej, případně se nepřipraví na zkoušku. V tom případě si i velmi těžký předmět zapíše průměrní studenti, čehož pak třeba litují. O rok později pak lze vidět nízkou úspěšnost u předmětu, proto si ho zapíše motivovaní studenti, kteří předmětu zvednou úspěšnost, a tak by se to mohlo střídát každý rok.

### Časový plán zkoušek především z povinných předmětů

pro **3. ročník v IT-BC-3 na FIT**

2018/2019 - ZIMNÍ SEMESTR, **3BIT**

<i>Předmět</i>	<i>Půlsestrální zkouška</i>	<i>Řádná Zkouška</i>	<i>První opravná zkouška</i>	<i>Druhá opravná zkouška</i>
<a href="#">ISA</a> Síťové aplikace a správa sítí	stanoví učitel	2019-01-02 13.00 – 15.50	2019-01-14 12.00 – 14.50	stanoví učitel
<a href="#">IMS</a> Modelování a simulace	stanoví učitel	2019-01-04 14.00 – 17.50	2019-01-18 13.00 – 14.50	2019-01-30 15.00 – 16.50
<a href="#">IMP</a> Mikroprocesorové a vestavěné systémy	stanoví učitel	2019-01-08 09.00 – 10.50	2019-01-16 09.00 – 10.50	2019-01-29 12.00 – 13.50
<a href="#">IPZ</a> Periferní zařízení	stanoví učitel	2019-01-08 14.00 – 16.50	2019-01-21 14.00 – 16.50	2019-01-28 15.00 – 17.50
<a href="#">IIS</a> Informační systémy	stanoví učitel	2019-01-10 14.00 – 15.50	2019-01-24 14.00 – 15.50	2019-01-31 14.00 – 15.50
<a href="#">ISP</a> Semestrální projekt	není	Zápočet získat do 2019-02-01	není	není
<a href="#">ITU</a> Tvorba Uživatelských rozhraní	není	Klas. zápočet získat do 2019-02-01	není	není
<a href="#">IPA</a> Pokročilé asemblery	stanoví učitel	termíny stanoví učitel <==		
Zde neuvedené	předměty	hledejte	u <b>2BIT</b>	nebo <b>zde</b>

Obrázek 2.10: Tabulka časového plánu termínů zkoušek pro přednáškovou skupinu 3BIT

Každá stránka s kompletním rozvrhem (kromě menzy) zahrnuje kromě rozvrhu pro jednotlivé dny:

- soupis **pevných termínů** a **omezení** před rozvrhem,
- **odkazy** na tabulky časových termínů zkoušek pro jednotlivé přednáškové skupiny,
- **tabulku společných studentů** – stejná jako je u výuky,
- **tabulku s počty studentů u oboru** – stejná jako je u výuky,
- **tabulku místností** – stejná jako je u výuky,
- **požadavky vyučujících** na termíny zkoušek, rozsazení, délku zkoušky a další informace popsané blíže v sekci **2.2.2**,
- **text za rozvrhem**, který vyjmenovává členy malého kolegia děkana,
- **přehled předmětů a jejich povinností u oboru** – stejný jako je u výuky.

## Import do IS

Vkládání rozvrhových akcí je podobné jako u výuky. Největší rozdíl spočívá v tom, že se neposílají rozsahy skupin. Vyplňované údaje jsou: začátek rozvrhové události, datum události, den v týdnu, zkratka semestru, akademický rok, konec rozvrhové události, místnost, předmět, přednášková skupina a informace, o jaký termín zkoušky se jedná. Místností je většinou u zkoušky přiřazeno více, a proto je třeba si definovat pravidla pro správné umístění předmětu do dané přednáškové skupiny:

1. Pokud by se mělo do rozsahu skupin **vložit xx** (tzn. pravidlo číslo dvě u výuky), pak je přednášková skupina umístěna do **největší místnosti**. Studenti opakující předmět se tedy budou nacházet v místnosti s největší kapacitou.
2. Pokud je předmět **povinný**, vložíme ho do **všech místností** rozvrhové akce.
3. Pokud je specifikováno **upřesnění skupiny u předmětu** u konkrétní místnosti, **nemůže** se v ní vyskytovat skupina s jiným upřesněním. Mějme například v místnosti D105 předmět IDA s upřesněním *skupina A*. Příznak určuje, že se v D105 nemůže nacházet přednášková skupina *1BIB* (jelikož její upřesnění je *skupina B*). Pravidlo neplatí pro skupiny s rozsahem *xx*, které se vkládají do největších místností podle pravidla číslo jedna.

## 2.3 REST API

Do tohoto roku bylo získávání dat z IS FIT velmi složité a dělalo se buď přes dostupné pohledy v IS FIT anebo parsováním jednotlivých webových stránek, které obsahovaly chtěná data, jelikož nebylo možné přistupovat přímo k databázi IS FIT. Tento rok však vzniklo fakultní REST API pro snadnější přístup k datům. Pro export je možný přístup z URL <https://wis.fit.vutbr.cz/FIT/export/vyuka/rest/>. Pomocí metody GET je možné se takto snadno dostat k datům jako jsou předměty, povinnosti předmětů, obory, programy, místnosti, vyučující, požadavky, typy povinností a podobně. Data jsou dostupná ve formátu JSON. V tabulce 2.4 je zobrazený formát dat jednoho záznamu rozvrhové akce vložené v IS.

Označení	Hodnota	Popis
year	2018	akademický rok rozvrhu
sem	L	zkratka semestru (L Z)
day	1	ID dne v týdnu
start	9:00	začátek rozvrhové události
end	10:50	konec rozvrhové události
date	2018-05-17	datum události
type	exam	typ vkládaného záznamu
week	s	týden, vkládá se „s“ pro jednorázovou činnost
room	1442	ID místnosti rozvrhové události
room_abbrev	D105	zkratka místnosti rozvrhové události
course	12876	ID předmětu
course_abbrev	PDS	zkratka předmětu
lg	1MIT	přednášková skupina
info	řádná	typ termínu zkoušky

Tabulka 2.4: Příklad formátu dat rozvrhové akce vrácených přes REST API

## Shrnutí

Je evidentní, že současný stav není ideální a je potřeba spousta manuální práce. Největší nevýhodu vidím na začátku plánování, kdy je potřeba manuálně opravovat předměty zakódované ve skriptu a jejich povinnosti, a na konci plánování, kdy dochází k dlouhému ručnímu exportu stránek ve formátu HTML. Navíc aplikace neposkytuje žádnou podporu



pro vkládání rozvrhových oken do IS FIT a nepracuje s požadavky vyučujících, jelikož se nachází v jiném programu. Funkce, které obsahuje, nejsou všechny spolehlivé a je kladen velký důraz na zkušenosti rozvrháře s plánováním. Na druhou stranu oproti manuálnímu plánování ulehčuje uživateli práci s distribucí rozvrhových oken, zobrazuje kolize studentů a povinnosti pro obor při každém posunutí okna a obsahuje funkce pro umístění předmětu do místnosti, filtrování a jednoduchý export.

O vylepšení aplikace pojednávají dvě diplomové práce. První je práce Aleše Horkého [8], který se zabýval automatickým generováním rozvrhů ve své práci. Hlavním problémem tohoto řešení je komplikovaná konfigurace rozvrhu a dlouhý čas generování, který se pohybuje v jednotkách až desítkách hodin. Druhou prací je práce od Vladimíra Čilla [26], který vyvíjel aplikaci pro podporu manuálního vytváření rozvrhu v jazyce Python. Jeho největším problémem bylo pouhé spuštění aplikace. Pro spuštění bylo potřeba doinstalovat různé nástroje a knihovny ve stejné verzi používané aplikací. Dále bylo složité zpracování vstupních dat, přihlášení prvního uživatele přes příkazový řádek a nepřívětivé uživatelské rozhraní. Jeho aplikace nezobrazovala chyby ani svůj současný stav. Když uživatel udělal chybu, celá aplikace se bez hlášení zasekla a uživatel nevěděl, co má dělat. Řešení nebylo ideální, a proto je potřeba vytvořit novou aplikaci se snadno udržovatelným a uživatelsky přívětivým rozhraním. Mým úkolem tedy bude navrhnout a vytvořit aplikaci s vhodným uživatelským rozhraním pro podporu plánování podobně jako dělal V. Čillo, která uživatele provede celým procesem plánování od importu vstupních dat až po export na web a import rozvrhu do informačního systému.

## Kapitola 3

# Návrh aplikace

Na základě analýzy procesu tvorby rozvrhu provedené v kapitole 2.2 představuji návrh aplikace, která má za cíl přípravu vstupních dat pro plánování výuky i zkoušek, kde se soustředím hlavně na požadavky vyučujících. Dále bude aplikace poskytovat prostředí pro plánování a umožní hromadný export dat. Následující kapitola rozebírá návrh řešení, který je nutný před samotnou implementací aplikace. V sekci 3.1 rozebírám jednotlivé funkce aplikace, na základě kterých je pak možné sestavit databázi popsanou v sekci 3.2. Poslední sekce 3.3 obsahuje popis navrhované architektury aplikace.

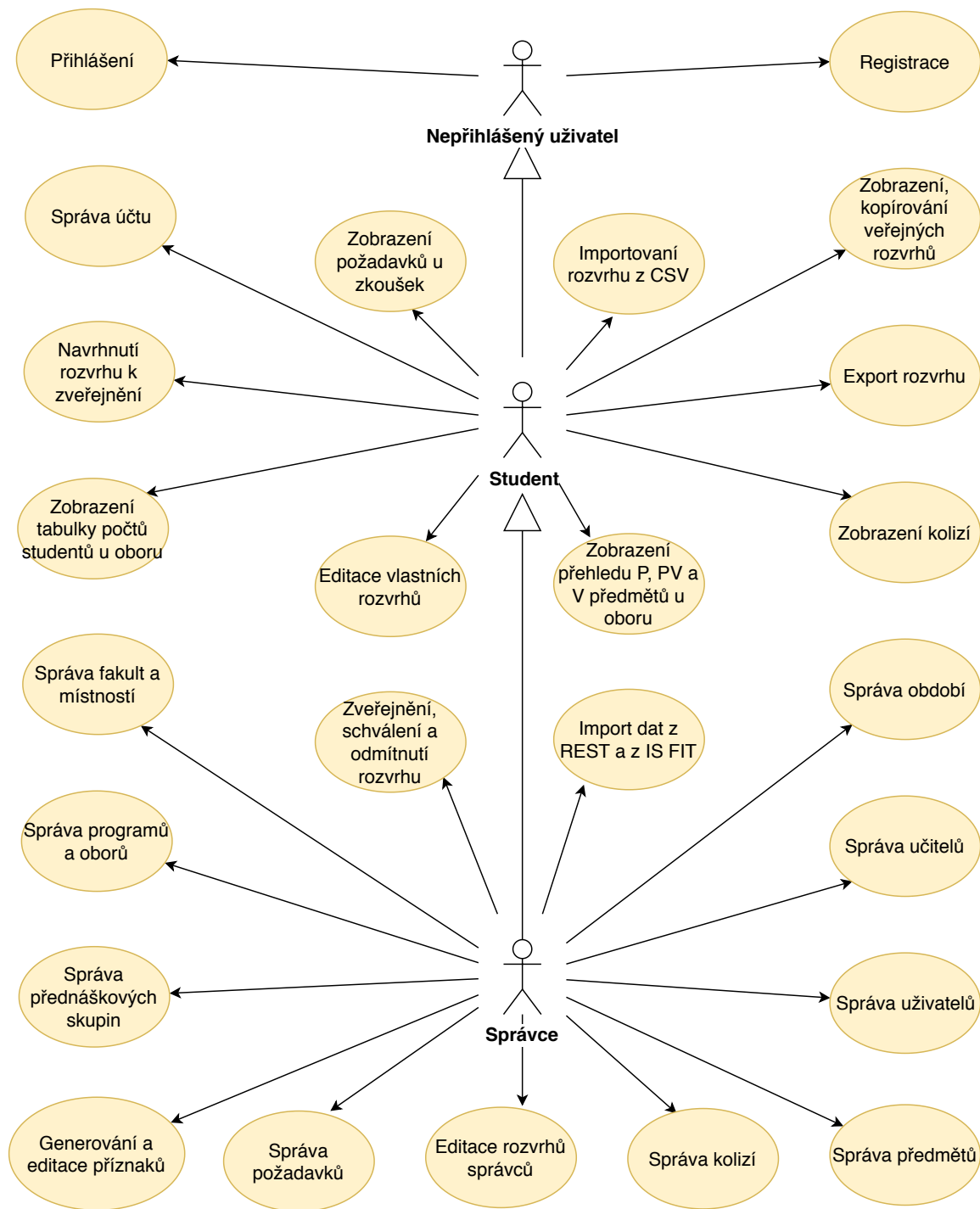
### 3.1 Návrh funkcionality aplikace

Aplikace bude navržena pro tři typy uživatelů. První pohled bude pro nepřihlášeného uživatele, druhý pro studenta, který má zájem zapojit se do tvorby rozvrhu, a třetí pro správce aplikace. Celkový diagram případu použití lze pak vidět na obrázku 3.1, kde jsou vidět jednotlivé funkce uživatelů v aplikaci. Student má oproti správci velmi omezené funkce. Nemůže například editovat ani stahovat vstupní data, což je práce pro správce.

#### 3.1.1 Stahování vstupních dat

Základem pro vytváření rozvrhů je zpracování vstupních dat. Současná aplikace na plánování řeší jen částečně import vstupních dat, jelikož má většinu dat zakódovaných přímo v aplikaci. Data je pak potřeba před každým plánováním zkontrolovat a doplnit ručně o nová. Tento způsob není ideální, jelikož je ruční zadávání zdlouhavé a může vzniknout spousta chyb. Proto do nové aplikace bude zakomponované automatické importování dat z IS FIT s využíváním REST API. Manuální vkládání a kopírování dat z minulých let musí být samozřejmě umožněno také pro případné doplnění dat, která se nestáhnou z REST API.

Stahování dat bude rozděleno na tři fáze. V první fázi se budou importovat předměty, jejich povinnosti, učitelé s vazbami na předmět a požadavky vyučujících na výuku. Jedná se o primární informace spojené s předmětem. Přes REST API se k takovým datům dostaneme přes URL: <https://wis.fit.vutbr.cz/FIT/export/vyuka/rest/courses>, kde nám rozhraní vrátí seznam všech předmětů. Podrobněji je REST rozhraní popsáno v tabulce 3.1. Rozhraní je dostupné pro přihlášeného uživatele pod jménem *rest*. Většinou je dostupná pouze metoda GET, metody POST, PUT a DELETE jsou implementovány pouze tam, kde jsou používány. Jazyk poskytovaných textů se volí parametrem *lang* (cs, en). V seznamu lze některá data filtrovat, např. *year* (akademický rok) a *sem* (semestr).



Obrázek 3.1: Diagram případů užití navrhované aplikace.

Tabulka 3.1: Rozhraní REST, prostřednictvím kterého budu získávat data. Všechny metody vyžadují autentizaci a autorizaci uživatele.

URL	Metoda	Popis
<i>courses</i>	GET	Seznam předmětů
<i>courses/id</i>	GET	Informace k předmětu se zadaným id
<i>course/id/teachers</i>	GET	Seznam učitelů a garantů předmětu
<i>course/id/fields</i>	GET	Seznam zařazení do oborů-programů
<i>course/id/students</i>	GET	Seznam zapsaných studentů
<i>course/id/sched_req</i>	GET	Seznam požadavků na výuku
<i>course/id/exam_req</i>	GET	Seznam požadavků na zkoušku
<i>course/id/schedule</i>	GET, POST	Seznam oken rozvrhu předmětu id
<i>programs</i>	GET	Seznam programů
<i>programs/id/fields</i>	GET	Seznam oborů/specializací programu se zadaným id
<i>rooms</i>	GET	Seznam místností

Ve druhé fázi se bude jednat o počty společných studentů v předmětech. Tento údaj REST neobsahuje a bude potřeba data stáhnout z IS FIT. Data pro export jsou dostupná pro přihlášené uživatele na adrese <https://wis.fit.vutbr.cz/FIT/export/vyuka/course-x.php>. Stránka je zobrazená na obrázku 3.2. Je zde vidět několik filtrů:

- Program se zkratkami: B – bakalářský a N – magisterský,
- Akademický rok,
- Semestr se zkratkami: L – letní a Z – zimní,
- Obory – zobrazí se počty studentů u konkrétních oborů,
- Plánované – zobrazí se pouze centrálně plánované předměty,
- Vč. bez Zk – zobrazí se i předměty, které nejsou zakončeny zkouškou,
- Se studenty – vyfiltruje předměty, kde je zapsaný alespoň jeden student,
- JSON – výstup dat se zobrazí ve formátu JSON.

Dokumente	Provoz	Výuka	VaV	Osoby a org.	Majetek								
<a href="#">IS FIT</a> > <b>Studenti v předmětech</b>					<a href="#">Zpět na i</a>								
Program: <input type="text" value="B"/> Ak.r.: <input type="text" value="2018"/> Sem: <input type="text" value="L"/> Obory: <input type="checkbox"/> Plánované: <input checked="" type="checkbox"/> Vč. bez Zk: <input type="checkbox"/> Se studenty: <input type="checkbox"/> JSON: <input type="checkbox"/> Vypsat													
	<b>Stud</b>	<b>IDS</b>	<b>IJC</b>	<b>IKR</b>	<b>IMA</b>	<b>IOS</b>	<b>IPK</b>	<b>IPP</b>	<b>ISU</b>	<b>ITS</b>	<b>IZA</b>	<b>IZG</b>	<b>IZU</b>
		12737	12753	12754	12756	12769	12772	12776	12721	12797	12814	12815	12818
<b>IDS</b>	446	-	2	36	56	110	413	406	36	46	15	259	409
<b>IJC</b>	148	2	-	0	140	142	3	2	141	0	0	80	4
<b>IKR</b>	48	36	0	-	2	2	37	36	1	5	4	17	35
<b>IMA</b>	543	56	140	2	-	525	46	44	500	0	0	281	44
<b>IOS</b>	614	110	142	2	525	-	102	97	510	0	3	327	97
<b>IPK</b>	461	413	3	37	46	102	-	414	28	53	17	256	413
<b>IPP</b>	449	406	2	36	44	97	414	-	25	50	16	256	412

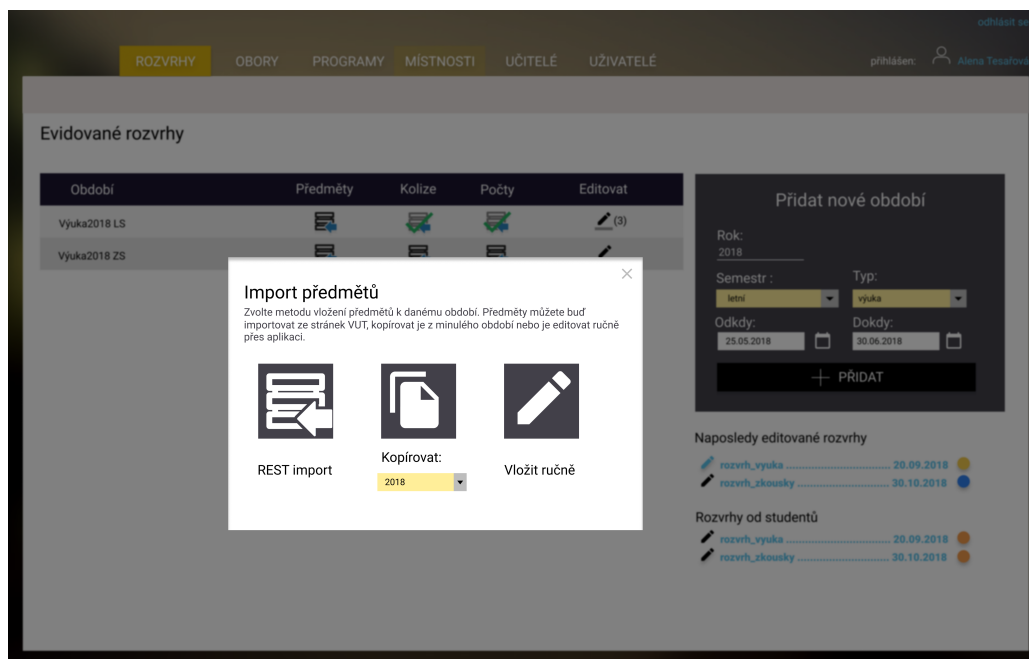
Obrázek 3.2: Stránka s kolizemi dostupná v IS FIT

Z dostupných filtrů je vidět, že pro stažení magisterských a bakalářských předmětů bude potřeba stránku načíst dvakrát vždy s vybráním vhodného filtru. Dále je důležité zjistit, které předměty jsou centrálně plánované pro konkrétní akademický rok. Opět musí stahování proběhnout dvakrát pro oba programy. Finálně se budou stahovat informace o počtu registrovaných studentů v oboru. Při každém stahování dat nás budou zajímat všechny předměty, které mají zapsaného alespoň jednoho studenta (filtr *Se studenty*).

Dokumenty		Provoz		Výuka		VaV		Osoby a org.		Majete						
IS FIT > Vypsání termínů																
Rok:	2018	Semestr:	Z	Ústav:	vše	JSON:	<input type="checkbox"/>	Vybrat								
Zkr	Sem	Ústav	Zak	Stud	#term	Za	Zk	#1	#2	#3	#4	#5	#6	ZkB	Max	Garant
ACH	Z	UPSY	ZaZk	42	6	20	4	19	14	4	2			60	100	Jaroš Jiří, doc. Ing., Ph.D.
AIS	Z	UIFS	ZaZk	67	5	24	3	65	16	2				51	100	Zendulka Jaroslav, doc. Ing., CSc.
BIO	Z	UIFS	ZaZk	81	8	15	4	57	15	8	3			55	100	Drahanský Martin, prof. Ing., Dipl.-Ing., Ph.D.
BIS	Z	UIFS	ZaZk	160	4	1	3	154	36	14				70	100	Hanáček Petr, doc. Dr. Ing.
BMS	Z	UIFS	ZaZk	101	5	2	3	86	18	9				70	100	Hanáček Petr, doc. Dr. Ing.
CCS	Z	UIFS	Klz	5	4	50										Matoušek Petr, Ing., Ph.D., M.A.

Obrázek 3.3: Stránka s počty na zkoušku dostupná v IS FIT

Třetí fáze importu je platná pouze pro plánování zkoušek, jelikož se v ní budou importovat počty studentů na termíny z minulého roku. Je dostupná pro export na adrese <https://wis.fit.vutbr.cz/FIT/export/vyuka/course-items-zk.php>. Stránka je zobrazena na obrázku 3.3. V tomto případě nás budou zajímat pouze počty studentů u konkrétních termínů zkoušek, jelikož ostatní informace již známe z první fáze. Termíny jsou zde označeny #1 – #6. Počet studentů pod sloupcem #1 ale nemusí nutně znamenat 1. termín zkoušky, jelikož se může jednat i o předtermín. Tuto informaci z tabulky nepoznáme, ale můžeme ji odhadnout. Pro lepší odhad budeme v aplikaci předpokládat, že výskyt čtyř nebo šesti termínů znamená existenci předtermínu.



Obrázek 3.4: Návrh hlavní stránky aplikace vytvořený v grafickém programu Figma



Návrh tabulky je zobrazen na obrázku 3.5. Ikona u jména vyučujícího slouží k přidání jeho preferovaných časů z minulých let. Konkrétně se bude jednat o tři období, která budou rozlišena barvami. Obdobím budeme v tomto smyslu chápat trojici: akademický rok, semestr a typ požadavku (výuka nebo zkouška). V tabulce 3.2 jsou zobrazené všechny kombinace, které mohou nastat, když je známý typ požadavku a semestr. Takto přidané preferované časy z minulých let se nebudou ukládat hned, ale rozvrhář si vybere, které chce přidat kliknutím na špendlík u vloženého záznamu. Barvy jsou závislé na období, ze kterého pochází daný požadavek. Důležité je, že při špendlíkováním se přidá jak záznam, tak i barva. Skrytí řádků je možné kliknutím na × anebo znovu načtením stránky. Další důležitá ikona + bude hned u předmětu a bude přidávat všechny požadavky na typ výuky z minulého roku pro konkrétní předmět.

Tabulka 3.2: Období požadavků, která se používají pro kompletaci. Zkratky: min. – minulý, akt. – aktuální, sem – semestr, LS – letní semestr, ZS – zimní semestr, zk – zkouška.

Typ	Sem.	1. období (modrá)			2. období (fialová)			3. období (žlutá)		
		Sem.	Rok	Typ	Sem.	Rok	Typ	Sem.	Rok	Typ
Výuka	ZS	ZS	min.	výuka	LS	min.	výuka	LS	min.	zk
	LS	LS	min.	výuka	ZS	akt.	výuka	LS	min.	zk
Zk	ZS	ZS	min.	zk	LS	min.	zk	ZS	akt.	výuka
	LS	LS	min.	zk	ZS	akt.	zk	LS	akt.	výuka

Systém barvení buněk jsem navrhla tak, aby byl co nejjednodušší. Nad tabulkou se bude nacházet paleta barev. Nanesení barvy bude možné po kliknutí na danou barvu. Pokud si budeme přát buňku odbarvit, bude pro tento účel sloužit ikona s gumou u palety barev. Každá barva má svůj jednoznačný význam, který se zobrazí po najetí myši na barvu.

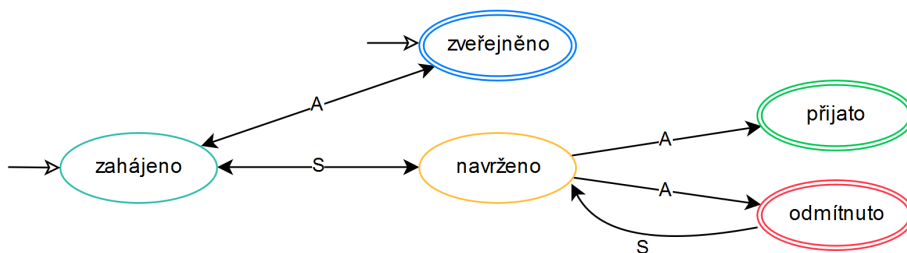
Jelikož rozvrh může plánovat i více lidí, je potřeba režim obarvování rozlišit na globální a na privátní. V globálním režimu vidí zbarvení oken všichni. Pokud ale uživatel chce obarvovat pouze pro sebe, použije privátní režim obarvování. Přepínání bude realizovatelné kliknutím na zámeček umístěný vedle gumy.

### 3.1.3 Plánování rozvrhu

Rozhraní pro plánování rozvrhů bude podobné jako v současné aplikaci, jelikož se za léta plánování již osvědčilo. V záhlaví tabulky se budou nacházet časové úseky, v levém sloupci den a místnost. Na pravé straně tabulky pak budou zobrazeny počty společných studentů a graf povinností předmětů popisovaný blíže u zkoušek v sekci 2.2.2. Rozvrhové akce se budou vepisovat do buněk se zaběhnutou syntaxí a sémantikou používanou v původní aplikaci, rozšířenou o identifikace týdne (lichý, sudý) a ostatní typy cvičení (laboratorní, počítačová atd.). Možnosti formátování rozvrhového okna jsou zobrazeny v tabulce 3.3 i s příklady použití.

Verze rozvrhu se bude moci nacházet v pěti stavech. Diagram přechodů stavů se nachází na obrázku 3.6 a jednotlivé stavy jsou vysvětleny v tabulce 3.4. Diagram znázorňuje možné přechody mezi dostupnými stavy verze a kdo přechod může realizovat. Verze může mít dva počáteční stavy: *zahájeno* a *zveřejněno*. Ve stavu *zahájeno* je verze vždy po založení nové čisté verze. Do stavu *zveřejněno* je možné se dostat dvěma způsoby. Buď správce **zveřejní** zahájenou verzi, nebo **schválí** navrženou verzi studenta. Schválením verze studenta se tato verze zkopíruje do nové verze, která již patří správci a je ve stavu *zveřejněno*. Studentova verze se pak přepne do stavu *přijato společně s komentářem*, který přidal správce. Stav

přijato je konečný a není možné verzi znovu navrhnout na schválení. Pokud by přece jenom student chtěl verzi zlepšit, lze ji samozřejmě zkopírovat, upravit a poslat znova na schválení.



Obrázek 3.6: Přejechy stavů rozvrhu. U hrany je vždy zkratka role uživatele, která může operaci provést: A – správce, S – student.

Tabulka 3.3: Vysvětlení formátu buňky rozvrhového okna

Ozn.	Umístění	Význam u zk.	Význam u výuky	Příklad
!	před zkratkou před.	omezení	omezení	<i>!IDA</i>
0	před zkratkou před.	předtermín	-	<i>0IDA</i>
2	před zkratkou před.	2. termín	num. cvičení	<i>2IDA</i>
3	před zkratkou před.	3. termín	lab. cvičení	<i>3IDA</i>
4	před zkratkou před.	4. termín	poč. cvičení	<i>4IDA</i>
5	před zkratkou před.	5. termín	jiné cvičení	<i>5IDA</i>
6	před zkratkou před.	6. termín	demo. cvičení	<i>6IDA</i>
(a)	za zkratkou před., v závorce	před. skupina A, význam pouze u místností	před. skup. A	<i>IDA (a)</i>
(b)	za zkratkou před., v závorce	před. skupina B, význam pouze u místností	před. skup. B	<i>2IDA (b)</i>
(l)	za zkratkou před., v závorce	-	lichý týden	<i>IDA (l)</i>
(s)	za zkratkou před., v závorce	-	sudý týden	<i>IDA (as)</i>
<i>+Int</i>	za zkratkou před.	přidání/odebrání hodin u rozvrhové akce	přidání/odebrání hodin u rozvrhové akce	<i>IDA +3</i>
<i>[Int-Int]</i>	za zkratkou před., v hranatých závorkách	-	rozsah skupin	<i>IDA [10-20]</i>

Studenti uvidí pouze zveřejněné rozvrhy, které jim zveřejnil správce. Pokud chtějí rozvrh editovat, je potřeba vytvořit jeho kopii. Takto vzniklý rozvrh lze editovat a následně ho navrhnout správci na schválení. Všechny navržené rozvrhy se zobrazí správci na hlavní straně společně s komentářem, který student přidal. V případě, že se mu rozvrh líbí, může ho přijmout. V ten moment se vytvoří kopie rozvrhu ve stavu zveřejněno, kde je již majitelem správce a rozvrh může dále upravovat. Správci vidí a editují své rozvrhy a rozvrhy



ostatních správců. Abychom zajistili, že si verze vzájemně nepřepíší, bude se rozvrh automaticky **zamykat** po editaci rozvrhového okna. Po dokončení změn může správce rozvrh znovu odemknout pro ostatní. Je potřeba myslet i na situace, kdy správce zapomene rozvrh odemknout. V tom případě by pak nesměl jiný správce rozvrh editovat a mohl by vzniknout problém. Tlačítko *Odemknout rozvrh* proto bude dostupné pro všechny správce, a tím se předejde danému problému.

Tabulka 3.4: Stavy rozvrhu

Stav	Zkr.	Rozvrh vidí	Poznámka
zahájeno	B	uživatel, který rozvrh založil nebo správce	počáteční stav
navrženo	S	správce + student, který rozvrh založil	rozvrh určený správci k posouzení
přijato	A	student, který rozvrh založil	přijatý rozvrh
odmítnuto	R	student, který rozvrh založil	odmítnutý rozvrh
zveřejněno	P	všichni	zveřejněný rozvrh pro všechny

### 3.1.4 Navrhované funkce při plánování

Následující seznam obsahuje navrhované funkce při plánování rozvrhu. Nachází se v něm jak funkce obsažené v současně používané aplikaci, tak další funkce, které budou nápomocné při plánování.

- Dynamické ukládání dat při přidávání, editaci a mazání rozvrhových oken
- Dynamická změna počtu společných studentů po editaci rozvrhového okna
- Dynamická změna grafu povinností po editaci rozvrhového okna
- Dynamické přidávání a skrývání řádků
- Editace místností na řádku
- Přepnutí typu týdne (sudý, lichý)
- Filtr na přednáškové skupiny a menzu (u zkoušek)
- Kopírování verze rozvrhu
- Odstranění verze rozvrhu
- Přejmenování verze rozvrhu
- Přepínání sudého a lichého týdne u výuky
- Zobrazení počtů studentů pro dvojici předmětů u daného dne
- Zobrazení počtu povinných a povinně volitelných předmětů ve sloupcovém grafu
- Přidělení předmětů do místností podle jejich kapacity s ohledem na požadavek na rozsazení u daného termínu zkoušky

- Přepínání pohledů s místnostmi a bez místností u plánování zkoušek
- Dílčí exporty do HTML, export do CSV a hromadný export

Hromadný export se bude lehce lišit pro studenta, jelikož by student neměl mít právo ke stažení souboru pro import do IS FIT a souboru s odhadovanými počty studentů na zkoušku. Proto budou tyto soubory při exportování vynechány.

### 3.1.5 Automatické rozmístění oken u zkoušek

U plánování zkoušek se postupuje nejprve tak, že se přidělí časy k jednotlivým událostem. V tomto kroku se bere ohled na kolize, požadavky a na další kritéria. V druhém kroku se přiřazují rozvrhová okna do místností. Současná aplikace sice obsahovala dva pohledy na rozvrh, ale po přepnutí do stavu s místnostmi se již nedalo dostat zpět do pohledu bez místností. Uživatel pak musel všechna přiřazená okna posunovat zvlášť. Navíc byla okna umístěna do místností v pořadí, v jakém šly místnosti za sebou v aplikaci. Tento způsob nebyl ideální. V rámci navrhované aplikace bude implementováno chytřejší rozmístění do místností, přičemž bude respektována potřebná kapacita jednotlivých termínů a požadavky na rozsazení. Algoritmus bude umisťovat rozvrhová okna od největších předmětů po nejmenší a vkládat je do místností od těch s nejmenší kapacitou po největší místnosti (respektive kombinace místností s největší celkovou kapacitou). Důvodem je předejití problému, kdy nemůžeme vložit předmět do rozvrhu kvůli jeho kapacitě, jelikož si místnost již zabral předmět s menší kapacitou. Zároveň nechceme umisťovat předměty s menšími kapacitami do velkých místností, aby se neplýtvalo energiemi na klimatizaci a osvětlení.

### 3.1.6 Příznaky u předmětů

V kapitolách 2.2.1 a 2.2.2 byla provedena analýza pravidel pro ruční vkládání rozvrhových akcí do IS FIT. Pravidla jsou důležitá nejen u vkládání do IS, ale také pro informaci, jaké předměty patří k přednáškové skupině. Právě přiřazení přednášková skupina-předmět je potřeba při filtraci rozvrhových oken u rozvrhu a exportu rozvrhů na web. První možností realizace je funkce, která projde všechna pravidla a vrátí přednáškové skupiny pro konkrétní předmět. Existuje ale pár výjimek – předmětů, které mají jiné chování. Příkladem je předmět *IZG*, který si je možné zapsat i v prvním ročníku, i když je povinný až ve druhém. Podle pravidel by pak nevzniklo žádné spojení předmětu *IZG* se skupinami *1BIA* nebo *1BIB* a nejednalo by se o správné chování. Bude proto potřeba přijít s jiným mechanismem, který bude počítat i s těmito výjimkami.

Tabulka 3.5: Tabulka příznaků a jejich vlastností

Druh příznaku	Studium	Povinnost předmětu	Příklad	Generování
Rozsah skupin	Bakalářské	Povinný	10-24	úplné
Obory	Magisterské	Povinný	10 MBI; 17 MPV	úplné
xx	Bakalářské, magisterské	Volitelný, povinně volitelný, doporučený, pro opakující	xx	částečné

Tabulka 3.6: Pravidla pro generování příznaků. Zkratka lib. – libovolný ročník

Prog.	Povin.	Roč.	Dodatečná podmínka	Vygenerovaný příznak	Před. skup.
Bc.	P	1	-	Rozsah skupin	1BIA, 1BIB
				xx	2BIA, 2BIB
Bc.	P	2	-	Rozsah skupin	2BIA, 2BIB
				xx	3BIT
Bc.	P	3	-	Rozsah skupin	3BIT
Bc.	PV, V	1	-	xx	1BIA, 1BIB, 2BIA, 2BIB, 3BIT
Bc.	PV, V	2	-	xx	2BIA, 2BIB, 3BIT
Bc.	PV, V	3	-	xx	3BIT
Bc.	PV, V	lib.	Není povinný pro žádný obor.	xx	1BIA, 1BIB, 2BIA, 2BIB, 3BIT
NMgr.	P	1	-	Obory	1MIT
				xx	2MIT
			Je v angličtině.	Obory	1EIT
				xx	2EIT, INTE
NMgr.	P	2	-	Obory	2MIT
				xx	1MIT
			Je v angličtině.	Obory	2EIT
				xx	1EIT, INTE
NMgr.	P	lib.	-	Obory	1MIT, 2MIT
			Je v angličtině.	Obory	1EIT, 2EIT
				xx	INTE
NMgr.	PV, V	lib.	Není povinný pro žádný obor.	xx	1MIT, 2MIT
			Je v angličtině.	xx	1EIT, 2EIT, INTE

Návrh řešení spočívá v definování a použití příznaků, které bude možné přiřadit dvojici předmět-přednášková skupina. Příznak se bude přiřazovat podle povinnosti předmětu u konkrétního oboru a ročníku. Budeme evidovat tři druhy příznaků zobrazené v tabulce 3.5.

Příznak **Rozsah skupin** se bude používat pouze pro bakalářské povinné předměty. Jeho existence u předmětu a přednáškové skupiny bude znamenat vkládání rozsahu přednáškové skupiny, který je buď specifikovaný u rozvrhového okna, anebo se použije celý rozsah pro konkrétní období. Výjimkou jsou přednášky s tímto příznakem, u kterých se rozsah nekládá. Další příznak je označen **Obory**, jelikož jsou to právě zkratky oborů, které se při existenci tohoto příznaku vkládají do rozsahů skupin. Oba příznaky lze správně vygenerovat. Příznak **xx** se používá pro volitelné, povinně volitelné, doporučené předměty a předměty studentů, kteří měli předmět povinný minulý rok. Tímto příznakem se označují i výjimky, které je potřeba zadat ručně. Příklady výjimek jsou například předměty IZG a ITU, které si je možné zapsat o rok dříve.

## Pravidla pro generování příznaků

Aplikace bude umožňovat generování příznaků pro přednáškové skupiny: 1BIA, 1BIB, 2BIA, 2BIB, 3BIT, 1MIT, 2MIT, 1EIT, 2EIT, INTE. Pro doplnění je přednášková skupina EIT určená pro magisterský program v angličtině a skupina INTE je určena pro zahraniční studenty. Seznam pravidel pro generování příznaků je znázorněn v tabulce 3.6 a byl vytvořen na základě konzultací s Ing. Jaroslavem Dytrychem, který je v současnosti společně s Ing. Petrem Veigendem zodpovědný za vytváření rozvrhu na FIT. První čtyři sloupce udávají vstupní podmínky, za kterých se vygeneruje příznak, jehož označení je v pátém sloupci. Poslední sloupec pak udává, ke kterým přednáškovým skupinám se příznak přiřadí.

### 3.1.7 Exporty

Nejdůležitějším vylepšením je práce s exporty. Exporty musí obsahovat všechna data popisovaná v sekci 2.2.1. Vzhledem k množství informací je třeba myslet na export již při importu dat. Některé informace však nelze stáhnout a je potřeba je přidat manuálně. Tento případ se týká například tabulky časového plánu zkoušek (obr. 2.10).

Jedná se o tabulku, která se exportuje pro jednotlivé přednáškové skupiny a obsahuje přehledný soupis termínů na zkoušky. Bude se nacházet v hromadném exportu a zobrazí se jak správci tak studentovi. Jak je vidět na obrázku 2.10, některé řádky není možné vygenerovat. Jedná se například o termíny státních závěrečných zkoušek nebo datum zápočtu semestrálního projektu. Pro pohodlnost bude editace tabulky obsahovat možnost kopírování textů z minulých let, jelikož se texty opakují každý rok a mění se pouze termíny. Dále je třeba označit, jaké texty se budou aktualizovat a jaké budou statické. Aktualizování dat je vhodné v případě, že se jedná o rozvrhovaný předmět, kterému se mění termíny zkoušek. Aby rozvrhář nemusel všechny předměty k dané přednáškové skupině přidávat ručně, bude přidána možnost automatického přidání termínů předmětů k přednáškové skupině. Je potřeba dodat, že se data do tabulky zadávají ve formátu HTML, aby je bylo možné jakkoli formátovat. Jelikož není čistý text HTML moc přehledný, bude přidána funkce náhledu tabulky. Postup rozvrháře pro přípravu exportu tabulky pak bude následující:

1. Nechá si předgenerovat předměty pro danou přednáškovou skupinu
2. Odstraní předměty, které generátor vygeneroval navíc
3. Přidá předměty, které chybí v tabulce (typicky volitelné předměty, které generátor ignoruje)
4. Přidá texty buněk tabulky z minulého roku
5. Upraví termíny u negenerovaných textů
6. Zobrazí si návrh tabulky a zkontroluje, jestli je všechno v pořádku
7. Tabulka je připravená na hromadný export

Jakmile je připravená tabulka, stačí pouze dodělat texty s informacemi u zkoušek, které se zobrazují pod tabulkou. Texty jsou téměř každý rok stejné a bude stačit pouze označit, které texty patří ke které přednáškové skupině. Nesmíme v návrhu zapomenout i na pořadí jednotlivých položek u textů i u tabulky. Pořadí musí být možné v jakýkoli okamžik měnit.

### 3.1.8 Import na server FIT

Pravidla pro import do IS jsou rozepsána v sekcích 2.2.1 a 2.2.2, na základě kterých umíme vygenerovat příznaky podle tabulky 3.6. Vytváření souboru pro import se nám proto velmi zjednodušilo. Postupně budeme procházet předměty rozvrhových akcí a podle příznaku je budeme umísťovat do přednáškových skupin následujícím způsobem:

- Pokud má předmět příznak **Rozsah skupin** u přednáškové skupiny YZ a jedná se o cvičení, pak se vloží přednášková skupina YZ a rozsah skupin od, do buď podle definovaného rozsahu skupin v rozvrhovém okně, anebo podle globálního rozsahu přednáškové skupiny YZ pro aktuální období, např. od: 10, do: 30, skupina: 1BIA.
- Pokud má předmět příznak **Rozsah skupin** u přednáškové skupiny YZ a jedná se o přednášku, nevkládá se žádný rozsah skupin, vkládá se pouze zkratka a ročník přednáškové skupiny YZ.
- Pokud má předmět příznak **Obory** u přednáškové skupiny YZ, je potřeba zjistit všechny obory, pro které je předmět povinný, a do rozsahu skupin vložit číslo skupiny oboru se zkratkou oboru, např. od: 18 MSK, do: 18 MSK, skupina: 1MIT.
- Pokud má předmět příznak **xx** u přednáškové skupiny YZ, vloží se rozsah skupiny od, do s hodnotou „xx“ a zkratka a ročník přednáškové skupiny YZ, např. od: xx, do: xx, skupina: 2BIB.

## 3.2 Návrh databáze

V současné chvíli již známe veškerá vstupní data a je vytvořený návrh aplikace společně s případy použití. Na základě těchto dat je proto možné sestavit schéma databáze, které je přiložené za následujícím textem. Databáze obsahuje **50 tabulek**, které lze na první pohled rozdělit do několika skupin:

- Tabulky obsahující informace o **přednáškových skupinách** (např. *Pred\_skup*, *Pred\_skup\_rozvrh*, *Pred\_skup\_text* atd.),
- Tabulky s **požadavky** vyučujících (např. *Pozadavky\_na\_vyuku*, *Pozadavky\_na\_zkousku*, *Pozad\_na\_term\_zk*),
- Tabulky pro ukládání **rozvrhu** (např. *Rozvrhy*, *Rozvrh\_verze*, *R\_zmena\_stavu*, *R\_zkousky\_okno*, *R\_vyukove\_okno*),
- Tabulky s informacemi o **uživatelích** (např. *Uzivatele*, *Barvy\_na\_pozadavky*, *Rest\_import*),
- Tabulky pro práci s **předměty** (*Predmety*, *Povinnosti\_predmetu*, *Ucitel\_uci*, *Ma\_registrovane*),
- Tabulky s počty studentů potřebné pro **export** (např. *Vuka\_pocty*, *Zkousky\_pocty*),
- Obecné **číselníky** (např. *Mistnosti*, *Fakulty*, *Ustavy*, *Semestry*, *Typy\_vyuky*).

Není potřeba popisovat všechny tabulky, jelikož jsou jejich názvy většinou samopopisné. Méně zřejmý je možná význam tabulek s přednáškovými skupinami. Například tabulka

*Pred\_skup\_rozvrh\_polozka* představuje tabulku časového plánu zkoušek podle vzoru Ing. Eyselta 2.10. Je možné do ní zadat buď předmět anebo nějaký text, který se bude zobrazovat místo zkratky předmětu. Pokud je předmět označený ke generování, bude se jeho obsah, tzn. sloupce řádná zkouška, první opravná zkouška a druhá opravná zkouška aktualizovat podle dat verze rozvrhu, se kterou aktuálně pracujeme. Tabulka se vytváří každé období pro jednotlivé přednáškové skupiny.

Obdobím se databázově myslí akademický rok a semestr. Pokud potřebujeme konkrétnizovat ještě období pro zkoušky a výuku, bude se jednat o podobdobí. Předmět vždy patří ke konkrétnímu období, proto při importování dat u rozvrhu bude stačit data importovat 1x pro období rozvrhu. Zajímavý je ještě atribut u předmětu *zmena\_priznaku*, který udává, jestli došlo k manuální úpravě jeho příznaku u nějaké přednáškové skupiny. Tato změna znamená, že u něho generátor nedokázal přiřadit příznaky správně a je proto potřeba si tento předmět pohlídat. Ve stejné tabulce se také nachází atribut *exportovat*. Pokud je předmět označený, že se neexportuje, nezobrazí se v souboru určeném pro import do IS.

### Přiřazení místností k rozvrhovým akcím

Rozvrh je možné importovat ze souboru ve formátu CSV, který však nemusí obsahovat správné zkratky místností. Často se stává, že rozvrhář ví, že bude daná výuka probíhat na FEKT, ale neví přesně v jaké učebně. Vloží pak do okna pro místnost např. T8, i když správná zkratka místnosti je T8/010. Z tohoto důvodu je třeba evidovat u výuky společně s místností i *text\_mistnosti*, aby bylo možné text dohledat a aplikačně ho upravit na správnou místnost. U místností plánovaných pro zkoušky je řešení trochu komplikovanější. Místnosti zkoušek se řeší v tabulce *R\_akce\_zk\_mistnost*, ve které je možné přetížít délku rozvrhové akce pro konkrétní místnost a zároveň specifikovat, pro jakou přednáškovou skupinu bude primárně určena. Řešení je založeno na skutečnosti, že při plánování dochází k přepínání režimů s místnostmi a bez místností. Pokud by nebyla místnost samostatně v tabulce a nacházeli jsme se v režimu bez místností, nebylo by jasné, které okno zobrazit, jelikož by jich mohlo být více.

### Obarvování požadavků

Pro uložení obarvených buněk požadavků slouží tabulka *Barvy\_na\_pozadavky*. Nachází se v ní *uzivatel*, který se vkládá pouze v případě, že se jedná o privátní režim barvení buňky. Do atributu *identifikator* se vkládá ID záznamu tabulky určené v *nazev\_tabulky*. Buňka je jednoznačně identifikována pouze v případě, že nevkládáme tabulku *Predmetry*. Jelikož se stejná tabulka používá jak u zkoušek, tak u výuky, je potřeba specifikovat, k jaké buňce přesně barvu vkládáme, proto je zde příznak *vyuka*.

### Menza

Zajímavý je ještě atribut *menza* u tabulky s přednáškovými skupinami *pred\_skup*. Jedná se o příznak, který určuje, že se jedná o menzu a nejedná se tedy o přednáškovou skupinu. Tento záznam bude typicky v tabulce pouze jeden. Zjednoduší se pak práce s filtrováním, jelikož je menza jedním z druhů filtrů, který přidává počty studentů na oběd k jednotlivým přednáškovým akcím a bude se hodit i v exportech u zkoušek.

## Ukládání konfigurace exportu

Při exportu výuky se podle návrhu budou vždy označovat přednáškové skupiny k exportu. Jelikož je výběr přednáškových skupin téměř vždy stejný, je výhodné si konfiguraci uložit, abychom ji rovnou mohli uživateli nabízet. Jedná se o tabulky *E\_export\_skupiny* a *E\_export\_default*.

## Rozlišení místností CVT

U rozvrhů výuky rozlišujeme dva pohledy na rozvrh. V prvním pohledu jsou důležité přednáškové místnosti a ve druhém místnosti CVT. Proto byla potřeba přidat tabulku *Místnosti\_typy*, která obsahuje různé typy místností pro možné přepnutí režimu rozvrhu výuky.





### 3.3 Architektura aplikace

Při návrhu aplikace je potřeba se zamyslet nad tím, kolik lidí ji bude používat, jestli bude potřebovat připojení k síti, jaké budou požadavky na rychlost aplikace a technické informace, jako je složitost implementace, znovupoužitelnost, podpora vývoje vybraného jazyka a podobně.

Nejprve stojí za zvážení, o jakou aplikaci se bude jednat. Na výběr máme buď desktopové nebo webové aplikace. Jelikož se bude jednat o software pro studenty a učitele, je v mém zájmu udělat webovou aplikaci. V případě desktopové by totiž byla potřeba instalace anebo fyzicky přenášet počítač s nainstalovanou aplikací.

Technologií pro webové aplikace je mnoho a každý rok se vyvíjí nové. Problém nově vyvíjených jazyků je obecné pravidlo, že čím rychleji se vyvíjí framework, tím dříve bude aplikace nepoužitelná. S přidávajícími verzemi se pak může stát, že aplikace napsaná v určité verzi již nebude za pár let podporována a případné rozšíření aplikace bude o to složitější. Dalším problémem může být úplné ukončení podpory vývoje. Zvolila jsem proto pro vývoj jazyk PHP, který má jednoduchou syntaxi a jedná se o nejrozšířenější jazyk pro web. Běží na něm zhruba 80 % všech webů [23]. Je multiplatformní, zdarma dostupný na webových stránkách [www.php.net](http://www.php.net), kde se nachází kompletní dokumentace, a je podporován webovým serverem Apache, který je nejpoblárnějším webovým serverem na Internetu [1]. PHP má k dispozici mnoho frameworků, z nichž jsem vybrala český framework Nette, jelikož je poměrně stabilní. Závislosti se řeší přes Composer a dokumentace jasně popisuje upgrade mezi verzemi. Více o frameworku je popsáno v kapitole 4.

V projektu budu používat databázi MySQL, která bude plně vyhovovat mým požadavkům. V návrhu databáze je 50 tabulek, které obsahují vazby mezi sebou. Hledala jsem proto nějakou technologii, pomocí které můžu s tabulkami jednodušeji pracovat. Byla proto vybrána knihovna Doctrine 2, která mapuje relační schéma na objekty a obráceně. Potom je možné přistupovat k tabulce jako k objektu.

## Kapitola 4

# Nástroje a technologie použité pro vývoj

Tato kapitola obsahuje popis technologií a knihoven použitých při implementaci aplikace navrhnuté v kapitole 3. Úvod kapitoly je věnován použitému frameworku pro PHP, na kterém je založena celá aplikace. Následuje popis knihovny Doctrine 2, která je využívána pro snadnější přístup k datům uloženým v relační databázi, a popis rozhraní REST, pomocí kterého je získávána většina vstupních dat aplikace. Na závěr jsou rozepsány další technologie v kontextu architektury aplikace.

## Nette

Nette je kompletní framework pro PHP. Dostupný je pod licencí BSD, která je jedna z nejvolnějších. Díky tomu je ho možné používat zdarma i pro komerční účely. Jeho autorem je český vývojář David Grudl, proto i celá dokumentace a fórum je v češtině.

Nette je možné rozšiřovat pomocí doplňků z portálu Componette<sup>1</sup>, proto není nutné časté problémy vlastnoručně implementovat. Nejjednodušším způsobem jejich instalace je pomocí nástroje Composer<sup>2</sup>. Díky němu navíc není potřeba při přenášení projektu přikládat potřebné knihovny, jelikož se postará o vložení knihoven do struktury projektu. V projektu je použita knihovna Kdyby/Doctrine, jejímž úkolem je integrace ORM frameworku Doctrine 2 do Nette. Více o Doctrine 2 je popsáno v nadcházející sekci.

Nette je postaven na návrhovém vzoru MVC<sup>3</sup>, tedy stojí na komponentách tří typů, které se v aplikaci dělí o tři základní úlohy: řízení (**kontrolery**), logiku (**moduly**) a výstupy (**pohledy**). Kontroler je spojení mezi modelem a pohledem. Zpracovává požadavky od uživatele a na jejich základě pak volá model. O vykreslení stránky pak požádá pohled. V Nette jsou obdobou kontroleru presentery.

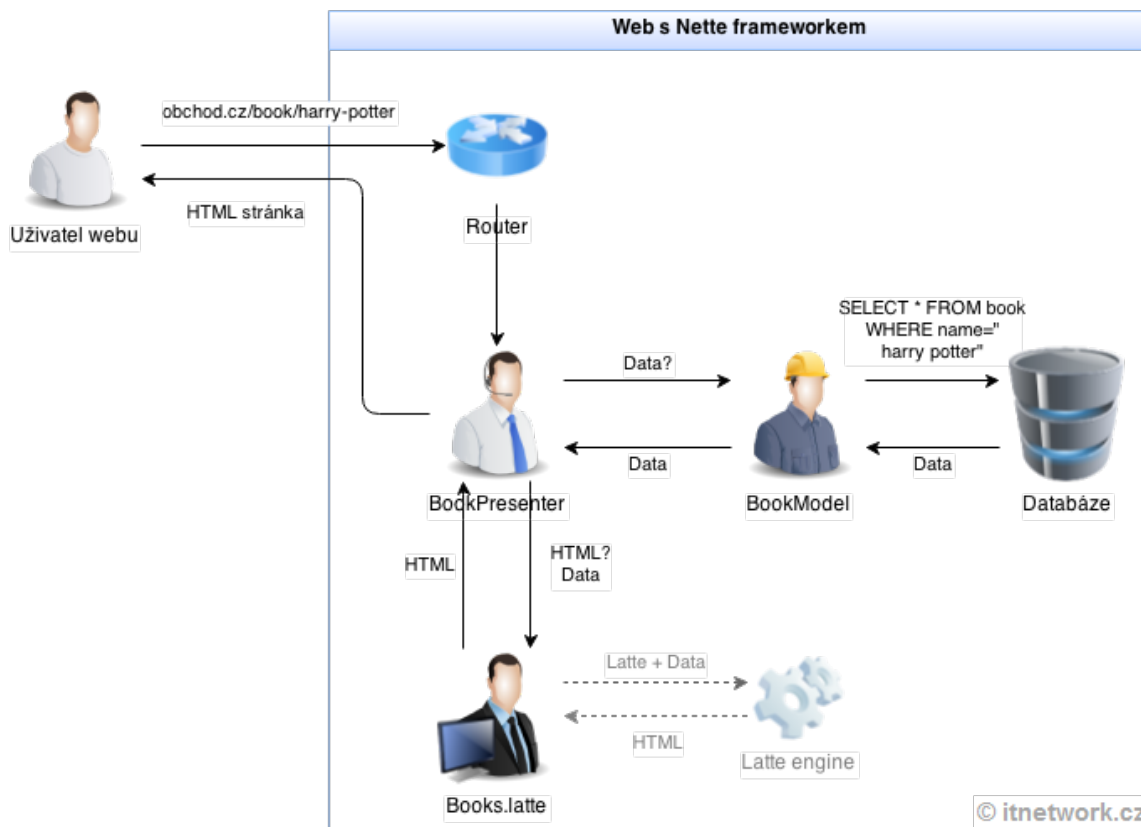
Životní cyklus aplikace po zadání URL `http://obchod.cz/book/harry-potter` do adresního řádku je znázorněn na obrázku 4.1 [27].

---

<sup>1</sup><https://componette.com/>

<sup>2</sup><https://getcomposer.org/>

<sup>3</sup><https://www.itnetwork.cz/php/mvc/objektovy-mvc-redakcni-system-v-php-popis-architektury>



Obrázek 4.1: Životní cyklus aplikace s Frameworkem Nette [27]

1. Nejprve se požadavek dostane k routeru, který podle adresy zjistí, že uživatel chce něco s knihami, a proto zavolá BookPresenter a předá mu zbytek URL.
2. BookPresenter se podívá do parametrů a zjistí, že uživatel chce knihu Harry Potter. Získá proto BookModel, kterému sdělí, že chce tuto knihu.
3. BookModel dostane v parametrech název knihy a vrátí data z databáze.
4. BookPresenter předá data pohledu (šabloně).
5. Šablona již obsahuje stránku v HTML a jsou v ní značky Latte<sup>4</sup>, do kterých vloží data. Vložení dat obstarává automaticky Latte engine.
6. BookPresenteru přijde z šablony výsledné HTML, které pošle uživateli.
7. Uživateli se zobrazí stránka.

Pohledy ve frameworku Nette tvoří **šablonovací systém Latte**. Šablony jsou tvořeny značkami HTML a speciálními makry ve složených závorkách. Dále šablony obsahují filtry, které slouží například k přeformátování vypisovaných dat. Kromě filtrů a maker zde můžeme použít komponenty, které se starají například o vykreslení formuláře. Taková šablona je přeložena pouze při její změně a její překlad je uložen do dočasného souboru. Jakmile se opět použije šablona, využije se již přeložený kód.

<sup>4</sup>Soupis značek Latte je k dispozici na adrese <https://latte.nette.org/cs/guide>

Nette má také svůj ladící nástroj pod názvem **Laděňka**, který slouží k rychlému odhalení chyb. Jedná se o knihovnu Tracy, kterou je možné instalovat i do projektů vytvářených v PHP bez použití frameworku Nette. Užitečnou funkcí této knihovny je tzn. **Debugger Bar**. Jedná se o plovoucí panel, který se zobrazí na okraji stránky. Nacházejí se v něm například informace jako jsou databázové dotazy a doba jejich trvání, přesměrování mezi stránkami anebo doba načítání stránky. Dále Tracy umožňuje přehledně zobrazit chyby a výjimky při běhu aplikace. Pak je velice snadné hledat v kódu chyby.

Při implementaci byla použita nejaktuálnější verze **Nette 2.4**. Informace vychází ze stránek Nette, které jsou dostupné na adrese <https://nette.org/cs/>.

## Doctrine 2

Jedná se o knihovnu pro ORM<sup>5</sup> pro PHP, jejímž cílem je mapovat objekty na relační databázi. Potom veškerá práce s daty se týká objektů a příkazy v SQL vůbec nejsou potřeba, jelikož se o ně ORM postará samo. Tento Framework má i vlastní dotazovací jazyk DQL, neboli **Doctrine Query Language**, jehož syntaxe se velmi podobá SQL. Největší rozdíl je v tom, že DQL pracuje s objekty a jejich atributy.

V implementaci byla použita knihovna pro Nette Kdyby/Doctrine ve verzi **3.1**.

## REST

REST (Representational State Transfer) je architektura rozhraní navržena pro distribuované prostředí. Tento pojem byl představen v roce 2000 v disertační práci Roye Fieldinga, který je jedním z tvůrců HTTP. Rozhraní REST se používá pro snadný přístup k zdrojům, což mohou být data nebo stavy aplikace. Každý zdroj má vlastní identifikátor URI a REST definuje čtyři základní operace k přístupu k nim (CRUD<sup>6</sup>). Jedná se o operace:

1. GET – čtení dat (select),
2. POST – vložení dat (insert),
3. PUT – uložení dat (update),
4. DELETE – odstranění dat (delete).

V rámci POST, PUT, DELETE lze odeslat libovolná data, je však nutno specifikovat typ (MIME<sup>7</sup>). Výsledkem požadavku je obvykle odpověď obsahující serializovaná data. Formát serializace je indikován hlavičkou Content-type v odpovědi. Obvykle opět *application/json* nebo *application/xml* [15] [4].

V práci bylo použito rozhraní REST IS FIT pro získání většiny vstupních dat k plánování rozvrhu.

## JavaScript

JavaScript je multiplatformní, objektově orientovaný, událostmi řízený jazyk. Patří do rodiny jazyků C/C++/Java, ale s Javou nemá nic společného. Kód je interpretovaný na

<sup>5</sup>Objektově Relační Mapování [https://en.wikipedia.org/wiki/Object-relational\\_mapping](https://en.wikipedia.org/wiki/Object-relational_mapping)

<sup>6</sup>CRUD (Create, Read, Update, Delete)

<sup>7</sup>MIME je označení pro typ internetového média spravovaný organizací IANA. Seznam registrovaných typů lze najít na adrese: <https://www.iana.org/assignments/media-types/media-types.xhtml>

straně klienta, proto poskytuje dynamickou funkcionalitu bez nutnosti čekat na načtení stránky serverem. Využívá se například při kontrolách formulářových vstupů, k tvoření animací a efektů obrázku anebo při otevírání dialogových oken. Při implementaci byla použita aktuální verze 9, známá i jako ECMAScript 2018 [12] [11].

## JQuery

Jquery je javascriptová knihovna, která klade důraz na interakci mezi JavaScriptem a HTML. Obsahuje funkce pro procházení a změnu DOM<sup>8</sup>, manipulaci s CSS, správu událostí, efekty a animace. Při implementaci byla použita knihovna JQuery verze 3.3.1 [2].

## HTML

Hypertext Markup Language (zkratka HTML) je značkovací jazyk pro tvorbu webových stránek. Pomocí značek (tzv. tagů) a jejich atributů dává částem dokumentu různý význam [2]. V implementaci se využívá verze 5.

## CSS

Cascading Style Sheets (CSS) je jazykem kaskádových stylopisů, který se používá k popisu grafické reprezentace dokumentu napsaného v HTML nebo XML. Určuje, jak bude vypadat element zobrazený na monitoru. Je jedním z klíčových jazyků na webu a je standardizován specifikací W3C [3] [14].

## Material Design & Bootstrap

Bootstrap je světově nejpopulárnější CSS framework s otevřeným zdrojovým kódem pro tvorbu responzivních webových aplikací. Obsahuje šablony pro tabulky, formuláře, tlačítka, navigace, dialogová okna, sestavené komponenty pro použití a zásuvné moduly založené na JQuery [7]. Při implementaci byla použita verze 4.

Na bootstrapu je založen i další CSS framework Material Design for Bootstrap, který je používán v aplikaci. Material Design je designový jazyk<sup>9</sup>, který byl vyvinut Googlem v roce 2014. Seskupuje v sobě klasické principy dobrého designu s vědeckými a technickými inovacemi. Inspiruje se texturami, včetně toho, jak odrážejí světlo a vrhají stíny. Je navržen tak, aby byl jednoduše rozšiřovatelný a umožňoval bezproblémovou implementaci komponent, zásuvných modulů a konstrukčních prvků. Je možné ho použít na více platformách (např. Android) [16] [20].

---

<sup>8</sup>[https://cs.wikipedia.org/wiki/Document\\_Object\\_Model](https://cs.wikipedia.org/wiki/Document_Object_Model)

<sup>9</sup>[https://en.wikipedia.org/wiki/Design\\_language](https://en.wikipedia.org/wiki/Design_language)

## Kapitola 5

# Implementace

Na základě návrhu představeného v předchozí kapitole byla implementována webová aplikace na podporu manuální tvorby rozvrhů výuky a zkoušek pro potřeby FIT. V této kapitole jsou popsány nejdůležitější implementované části aplikace a změny provedené oproti návrhu. Struktura aplikace vychází z doporučené struktury frameworku Nette popsané v kapitole 4.

app	.....	adresář s aplikací
├─ config	.....	konfigurační soubory
│ └─ config.local.neon	.....	přihlašovací údaje do DB
│ └─ config.neon	.....	konfigurační soubor
├─ forms	.....	třídy formulářů
├─ lang	.....	adresář pro překlady
├─ model	.....	modelová vrstva
│ └─ entities	.....	třídy entit
│ └─ facades	.....	třídy nad entitami
│ └─ queries	.....	třídy pro skládání dotazů do DB
│ └─ services	.....	třídy pro opakující se služby
├─ presenters	.....	třídy presenterů
│ └─ templates	.....	adresář se šablonami
│ └─ @layout.latte	.....	šablona společného rozložení stránky
├─ router	.....	třídy routerů
└─ bootstrap.php	.....	zaváděcí soubor aplikace
log	.....	obsahuje logy, chyby apod.
temp	.....	pro dočasné soubory
tests	.....	testy
vendor	.....	adresář na knihovny např. třetích stran
├─ nette	.....	knihovny Nette
└─ autoload.php	.....	soubor, který se stará o načítání tříd nainstalovaných balíčků
www	.....	veřejný adresář, document root projektu
├─ images	.....	obrázky
├─ css	.....	soubory CSS
├─ js	.....	skripty v JavaScriptu
├─ webfonts	.....	fonty
├─ .htaccess	.....	pravidla pro mod_rewrite
└─ index.php	.....	vstupní bod aplikace

Veškeré požadavky se posílají přes jeden soubor (*index.php*) umístěný ve složce *www*, který předává řízení do aplikace zavádějícímu souboru *bootstrap.php*, kde se konfiguruje a vytváří **DI kontejner**, který je srdcem celé aplikace. Nachází se v něm registrace všech služeb a parametry pro běh aplikace. Neprogramuje se ručně, ale je generován z konfiguračních souborů, které se obvykle zapisují ve formátu **NEON**. Ve vývojářském režimu se kontejner automaticky aktualizuje při každé změně kódu nebo konfiguračních souborů. V produkčním režimu se generuje pouze jednou, změny v souborech se nekontrolují, proto je vždy potřeba pro znovu načtení kontejneru vymazat obsah ve složce *cache*. DI kontejner vytváří službu, kterou následně předá do *index.php* a ten spustí webovou aplikaci.

Jednotlivé složky a soubory jsou ještě blíže popsány v dokumentaci Nette na adrese <https://doc.nette.org/cs/2.4/presenters>.

## 5.1 Hlavní strana aplikace

Hlavní strana aplikace je zobrazena na obrázku 5.1 a jedná se o výchozí místo pro uživatele. Oproti návrhu zde byly provedeny menší změny, které souvisí s technickými detaily aplikace a s pohodlnějším přístupem pro uživatele.

Byla sem přesunuta možnost vložení nového rozvrhu a nového období, aby ji nemusel uživatel nikde složitě hledat. Přidávací formuláře se dynamicky rozbalí v momentě kliknutí na tlačítko *Nový rozvrh* nebo *Nové období*. Nejdůležitějším prvkem na stránce zůstává evidence rozvrhů, kde má od sebe uživatel barevně rozlišené zkoušky (ružové) a výuku (černě). Oproti návrhu je zde místo jedné ikony v každém sloupci ještě jedna ikona pro přímý vstup do editační aplikace, kde je možné data manuálně opravit. Jelikož se v prvním sloupci jedná o stahování dat předmětů a požadavků na předměty, musí zde být dvě ikony – jedna pro vstup do předmětů a druhá pro vstup do požadavků. Barva ikony závisí na existenci dat u daného období. Pokud se například nacházíme v rozvrhu *LS 2018* a jsou ve sloupci *Kolize* všechny ikony zelené, znamená to, že existuje alespoň jedna kolize u období daného rozvrhu.

Oproti návrhu se lehce změnilo i vyskakovací okno pro stažení požadavků, které lze vidět na obrázku 5.2. Místo tří možností stažení dat se zde nachází pouze hlavní preferovaná možnost, kterou je stažení dat z REST. Je zde potřeba vyplnit základ URL pro REST API, který se liší pro export, ke kterému mohou přistupovat i vybraní studenti, a pro rozvrháře, kteří mají vyšší oprávnění. Dále je potřeba vyplnit přihlašovací údaje do REST a vybrat jednu z možností vložení dat. Data je možné buď vymazat a nahrát znovu, což vymaže všechna původní data aplikace a následně vloží nová, nebo je možné pouze aktualizovat data. Doporučená je druhá možnost.

Po stažení všech vstupních dat je možné přejít k vytváření samotného rozvrhu. Jak bylo popsáno v návrhu, rozvrh může mít více verzí, ale každý si edituje vlastní rozvrh kromě správců, kteří si mohou vzájemně upravovat rozvrhy. Nejrychlejší vložení verze rozvrhu je vložit CSV rozvrhu již vytvořeného původní aplikací. Možnost vložení CSV rozvrhu je na hlavní straně ve sloupci *Rozvrh* po kliknutí na ikonu s nápisem CSV.

Pod evidovanými rozvrhy se nachází naposledy editované verze přihlášeného uživatele pro rychlý přístup do rozvrhu. Vedle názvu každé verze je zobrazeno kolečko se zkratkou stavu uprostřed. Stav je vysvětlen v tabulce 3.4. Se změnou stavu rozvrhu je vždy možné přidat komentář, který se zobrazí pod názvem verze společně s datem změny stavu rozvrhu. Po kliknutí na kolečko je možné stav změnit. Přechody stavů se realizují podle grafu na obrázku 3.6. Zobrazí se vždy maximálně deset naposledy editovaných rozvrhů.

☰ FIT rozvrhy
👤

Rozvrhy

## Evidované rozvrhy

+ NOVÝ ROZVRH
+ NOVÉ OBDOBÍ

V tabulce jsou zobrazeny všechny Vaše rozvrhy. Růžovou barvou jsou označeny **zkoušky**, černou barvou výuka. Rozvrh je možné upravovat v evidenci rozvrhů nebo smazat kliknutím na křížek hned vedle názvu rozvrhu. U každého rozvrhu je možný import z FIT REST API. V případě předmětů se stahují do databáze také (v tomto pořadí): fakulty, typy zakončení předmětů, předměty, programy, obory, povinnosti předmětů, učitelé, výuka učitelů a pořadavky učitelů (na výuku/zkoušku podle rozvrhu). U kolizí se stahují: společní studenti (*kolize*), jaké předměty jsou centrálně plánované (*kolize\_planovane\_predmet*) a dále se stahují počty studentů na obor (*kolize\_obory*). Vždy proběhne napřed stáhnutí dat pro bakalářský program, pak pro magisterský.

Novou verzi je možné založit v evidenci verzí rozvrhu kliknutím na šipku ve sloupci Rozvrh nebo vložení CSV souboru po kliknutí na ikonu s CSV ve sloupci Rozvrh.

Zobrazeno  záznamů

Hledat:

Název rozvrhu	Předměty   Preference	Kolize	Počty na zk.	Rozvrh
2018 LS výuka (vyuka 2018 LS)				
2018 ZS výuka (vyuka 2018 ZS)				
2018 LS zkoušky_kopie (zkousky 2018 LS)				
zkouskaLS2017 (zkousky 2017 LS)				

Zobrazeno 1 až 4 z 4 záznamů
Předchozí
1
Další

## Moje naposledy editované rozvrhy

Pro rychlý přístup k naposledy editované verzi rozvrhu. Pokud si přejete rozvrh zveřejnit, klikněte na kolečko u názvu verze rozvrhu. Ke změně stavu je pak možné přidat komentář. Pro vstup do rozvrhu stačí kliknout na název verze.

P

2018 LS zkoušky\_20190404224816  
Komentář: ok je to  
13.04.2019 18:13:52

B

2018vyukaLS\_20190326222933

B

zkouskaLS2017\_20190407203824

B

2018 LS zkoušky\_20190404190919

B

2018 LS zkoušky\_01

B

2018 LS zkoušky\_20190401205145

B

2018 LS zkoušky\_20190326203130

B

2018 LS zkoušky\_20190327213749

## Rozvrhy od studentů

S rolí správce máte právo prohlížení rozvrhů od studentů. Verze se může nacházet v následujících stavech:

- Zahájeno* - výchozí stav
- Navrženo* - verze je navržena správcem na posouzení
- Přijato* - správce přijal rozvrh a vytvořila se kopie ve stavu zveřejněno, která již patří správci
- Odmítnuto* - správce odmítl verzi od studenta
- Zveřejněno* - verze je zveřejněna studentům

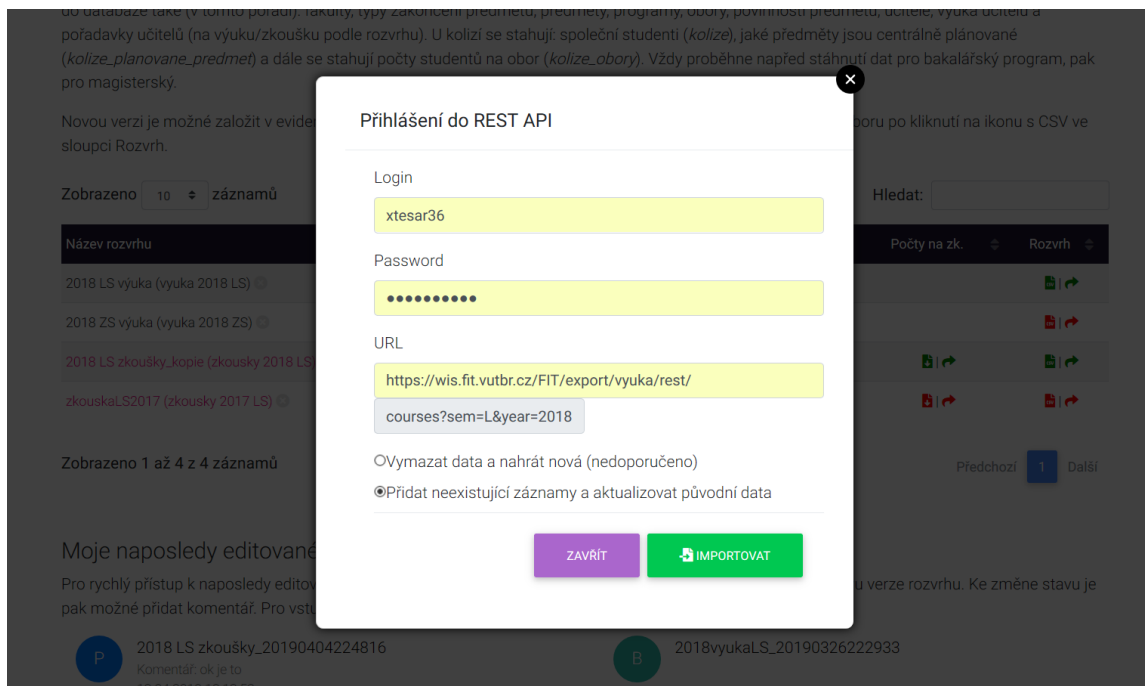
S

2018 LS zkoušky\_20190404224816\_kopie\_kopie  
(student student)  
Komentář: Lepší návrh.  
06.04.2019 18:48:31

© 2018 FIT, All rights reserved

Obrázek 5.1: Hlavní strana aplikace





Obrázek 5.2: Přihlášení do REST API pro import dat

Poslední částí hlavní strany je seznam navrhovaných verzí od studentů. Tuto sekci vidí pouze správce, který si je může zobrazit a vybrat si, jestli je přijme a zveřejní, nebo odmítne. Ke každé změně je vždy možné přidat komentář, aby bylo jasné, proč například byla verze odmítnuta.

## 5.2 Kompletace požadavků

Tato část je plně implementována podle návrhu. Náhled tabulky požadavků je zobrazený na obrázku 5.3. Zobrazuje se zde  $X$  předmětů  $\times$   $Y$  požadavků na výuku nebo termín zkoušky, u kterých je evidováno  $Z$  požadavků vyučujících na preferované časy. Dynamické přidávání a skrývání řádků musí fungovat na každé úrovni. Oproti návrhu se trochu změnila podoba ikon, ale funkcionality zůstávají stejné. Ikonou u předmětu se přidávají požadavky z minulého roku na typ výuky (u výuky) nebo typ termínu (u zkoušek) a ikonou s panáčkem u vyučujícího se přidávají požadavky na preferované časy. Jaká období se používají pro kompletaci je zobrazeno u návrhu v tabulce 3.2. Koš je celkem nebezpečná funkce, která záznam smaže z databáze. Oproti tomu špendlík přidává nový záznam do databáze. Ikony s plus a mínus slouží k přidání a skrývání řádků.

Upravování již existujících buněk funguje jednoduše kliknutím do dané buňky a přepsáním hodnoty. Obarvování bylo oproti návrhu lehce upraveno. Ze zobrazené palety barev je vždy možné kliknout na barvu a tím ji nanést na kurzor. Pro obarvení buňky pak stačí kliknout na buňku tabulky. Ve stejný moment se obarvení uloží do databáze podle režimu, ve kterém se zrovna nacházíme. Oproti návrhu barva zůstává nanesená na kurzoru do chvíle, kdy uživatel provede dvojklik, pak je možné pokračovat v editaci tabulky požadavků. Pokud jsme v privátním režimu, obarvení neuvidí ostatní uživatelé. V případě globálního režimu obarvujeme buňky pro všechny.

Před.	P	PV pro	Počet	Max	Garant	Přednáší	Vyučující	Typ	Poc	Kap	Hod	Týden	Místnost	Poznámka	Vyhovoval by mi termín	Nemohu učít	
ACH		2MPV/2MSK	0	90	Jaroslav Jirák, doc. Ing., Ph.D.	Jaroslav Jirák, doc. Ing., Ph.D.	Jaroslav Jirák, doc. Ing., Ph.D.	exc	1	20	2	vše		Pro cvičení bych potřeboval jednu konkrétní učebnu 0204, kde jsou GPU, které jsme nakoupili právě pro tento kurz.	Bylo by super mít cvičení hned po přednášce, tedy pondělí odpoledne		
															Pa 09:00 - 11:50 low	Út 12:50 - 12:50 Akademický senát VUT nebo různé komise senátu	
															Po 08:00 - 16:50 med	Pa 13:00 - 14:50 Vědu ústavní semináře	
																Pa 13:00 - 13:50 Vědu semináře ústavu	
																Pa 08:00 - 09:00 Zkouška	
								lect	1	80	2	vše					
								pc	1	20	2	vše	0204				

Obrázek 5.3: Kompletace požadavků u výuky

## Postup kompletace dat

1. Stažení požadavků z REST na hlavní straně
2. Stažení a uložení obecných požadavků vyučujících na termín zkoušek (u zkoušek) a typ výuky (u výuky) z minulých let v kompletaci dat (ikona u předmětu)
3. Stažení a uložení požadavků na termíny vyučujících (ikona u konkrétního vyučujícího)
4. Ruční přidání požadavků z poznámek

V momentě, kdy je kompletace hotová, je možné přejít k samotnému plánování rozvrhů, ve kterém budou hrát preference významnou roli. Z pohledu plánování budou nejdůležitější požadované délky termínů zkoušek (u zkoušek) a požadavky na délku výuky (u výuky). Časy budou sloužit jako výchozí délka rozvrhového okna u předmětu.

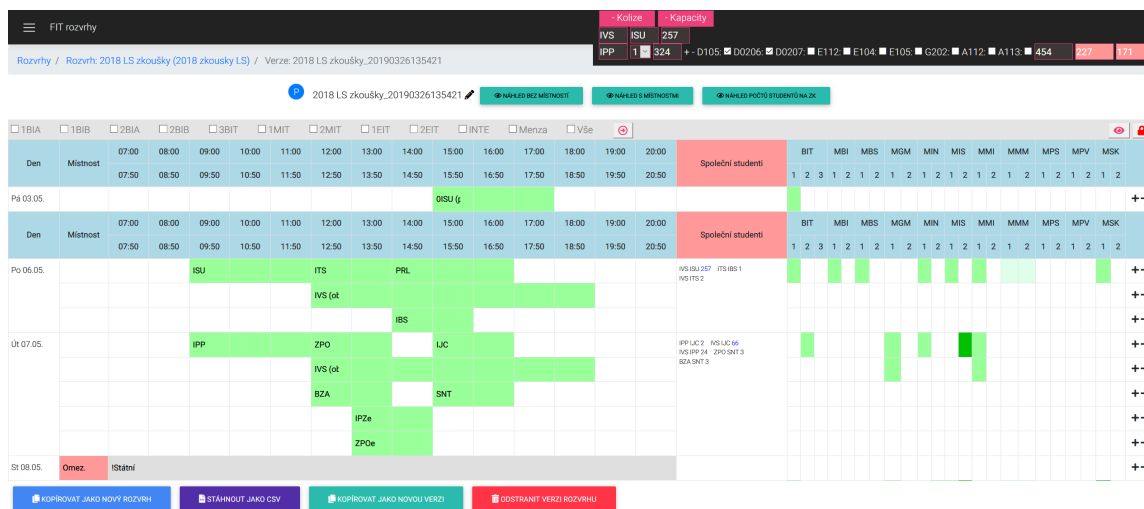
## 5.3 Plánování rozvrhů výuky a zkoušek

V části plánování byly implementovány všechny požadované funkce popsané v sekci 3.1.3. Výsledné prostředí pro práci s rozvrhem lze vidět na obrázku 5.4. U názvu verze se nachází opět kolečko pro zobrazení a změnu stavu rozvrhu, jako to bylo na hlavní straně. Zároveň je zde i možnost změnit název verze kliknutím na tužku vedle názvu. Navíc oproti návrhu jsou zde zobrazená tlačítka pro zobrazení náhledů exportů. Po kliknutí na tlačítko dojde k přesměrování na novou stránku, kde bude přesně zobrazené, jak daný export bude vypadat. Více o exportech bude popsáno v sekci 5.7.

Další rozšíření aplikace jsou nástroje v pravém horním okraji, které plní stejnou funkci jako tyto nástroje používané v současné aplikaci na plánování. Funkce *Kolize* umožní zadat zkratky dvou předmětů a vrátí pro ně počet společných studentů a funkce *Kapacity* vybírá vhodnou kombinaci místností, která bude kapacitně postačující pro předmět. Zobrazené počty studentů pro první až pátý termín u zkoušek vychází z počtů minulého roku. Vezme se vždy poměr počtů studentů z minulého roku, kteří přišli na daný termín, a počtu registrovaných studentů u předmětu. Tento poměr je následně vynásobený současným počtem registrovaných studentů.

Další implementovanou funkcí je přepínání pohledů s místnostmi a bez místností u zkoušek po kliknutí na tlačítko s okem vpravo od filtrů na přednáškové skupiny. Pokud by se stalo, že by rozvrhová akce nebyla přiřazena žádné místnosti, aplikace umí automaticky

přiřadit předmět podle jeho odhadované kapacity k místnosti anebo kombinaci místností (kombinace místností se zadávají v evidenci místností).



Obrázek 5.4: Prostředí pro práci s rozvrhem

Vedle ikony s okem se nachází ikona se zámkem, který je buď otevřený nebo uzavřený podle toho, jestli je rozvrh uzavřený anebo otevřený. Uzavřený rozvrh nemůže editovat jiný uživatel než ten, který ho uzavřel. Rozvrh se uzavře ve chvíli, kdy ho začne někdo editovat. V ten moment ho nesmí nikdo jiný upravovat, aby nedošlo k přepsání změn. Otevření může provést pouze správce kliknutím na ikonu se zámkem (i jiný správce než ten, který rozvrh uzavřel).

V tabulce s rozvrhovými akcemi jsou od sebe barevně odděleny různé termíny u zkoušek a různé typy výuky u plánování výuky. Druhý sloupec zobrazuje místnosti. U výuky a pohledu na zkoušky s místnostmi se vždy zobrazují výchozí místnosti uložené v databázi s atributem *dulezity* s hodnotou *true*. Místnost na řádku lze editovat a uložení proběhne hned po kliknutí jinam na stránku. Mohou pak nastat tři situace. První situací je, že na řádku s místností není žádná akce a místnost existuje. Pak se v pořádku uloží a zobrazí se informace o její kapacitě. Druhá možnost nastane, když místnost neexistuje. Takové místnosti jsou podbarveny červenou barvou (jako omezení) s cílem upozornit uživatele na chybu, aby si místnost opravil, jelikož neexistující místnosti nelze vložit do IS. Poslední možnost nastává, když existují nějaké akce na řádku, pro který chceme místnost změnit. Aplikace pak upozorní uživatele, že je vždy nejprve potřeba zadat místnost a pak rozvrhová okna, jelikož aplikace neumí automaticky změnit místnost u všech rozvrhových oken na řádku. Jednalo by se o nebezpečnou akci, kdyby uživatel náhodou přepsal místnost, která je již evidovaná v rozvrhu, protože by mohlo dojít k překrývání oken.

Syntaxe rozvrhového okna odpovídá pravidlům zobrazeným v tabulce 3.3 u návrhu. Okno se uloží na událost *onchange* a zobrazí vlevo dole informaci o tom, jestli proběhlo uložení buňky správně. Pokud u pozměněného okna jsou evidované nějaké požadavky, zobrazí se zároveň i požadavky.

Důležité akce se pak nacházejí ještě pod tabulkou. Kliknutím na tlačítko *Kopírovat jako nový rozvrh* se vytvoří nový rozvrh, který má právě jednu verzi (kopie verze, se kterou pracujeme). Pomocí tlačítka *Stáhnout rozvrh jako CSV* se pomocí JavaScriptu projdou řádky tabulky s okny a vytvoří se soubor CSV kompatibilní se současnou aplikací na plánování.

Funkce *Kopírovat jako novou verzi* vytvoří kopii verze a přiřadí ji k současnému rozvrhu a funkce *Odstranit verzi rozvrhu* verzi vymaže z databáze (může vykonat pouze zakladatel verze nebo správce).

## 5.4 Evidence předmětů

Evidence předmětů slouží nejen k zobrazení, editaci, přidávání a odstraňování předmětů, ale nacházejí se zde i povinnosti předmětů, registrovaní studenti a učitelé učící daný předmět. Náhled této části je zobrazen na obrázku 5.5. Vedle nadpisu *Předměty k danému období* se nachází ikona pro filtrování předmětů, které byly minulý rok centrálně plánované a tento rok již nejsou. Sloupce s počtem kreditů, počtem přihlášených studentů, počtem hodin rozvrhových akcí, fakultou a typem zakončení lze dynamicky měnit po kliknutí do příslušné buňky tabulky. Stejně tak lze kdykoli označit předmět za centrálně plánovaný, nebo ho naopak odznačit.

Po kliknutí na ikonu ve sloupci učitelé se zobrazí seznam učitelů a jejich rolí (např. cvičící, garant) s možností přidat nového učitele a jeho roli v předmětu. V povinnostech předmětů se nacházejí vždy kombinace: **obor**, **ročník** a **povinnost**, které jsou důležité pro generování příznaků. Pokud je povinnost platná pro všechny ročníky v oboru, ročník je označený nulou. Posledním zobrazením u předmětů jsou studenti, kteří mají předmět registrovaný (evidují se loginy a ID studentů). Tato informace poslouží do budoucna k vyhodnocování kolizí u více jako dvou předmětů.

Předměty k danému období

Kromě základních údajů u předmětů evidujeme také povinnosti, učitele a registrované studenty. Pokud je předmět obarvený růžovou barvu, jedná se o předmět, který byl minulý rok centrálně plánovaný a tento rok již není. Tyto předměty je možné filtrovat kliknutím na ikonu s filtrem vedle nadpisu. Všechny údaje v inputech lze dynamicky měnit.

Zobrazeno 10 z 4 záznamů

ID	Zkratka	Název	Počet kreditů	Kapacita	Přihlášených	Před	Sem./Cv.	Lab. cv.	Poc. cv.	Jiná cv.	Fakulta	Typ zakon.	Jazyk	Centrálně	Učitelé	Povinnosti	Studenti	Odstranit
#12635	AGS	Agentiní a multiagentní systémy	5	90	32	26	0	0	13	13	FIT VI FIT VUT	Zk Zk	cs	<input type="checkbox"/>				
#12853	MUL	Multimédia	5	90	27	26	0	0	13	13	FIT VI FIT VUT	Zk Zk	cs	<input checked="" type="checkbox"/>				
#12854	MULe	Multimédia (v angličtině)	5	40	0	26	0	0	26	0	FIT VI FIT VUT	Zk Zk	en	<input type="checkbox"/>				
#12927	SNT	Simulační nástroje a techniky	5	90	41	39	0	0	0	13	FIT VI FIT VUT	ZaZk ZaZk	cs	<input checked="" type="checkbox"/>				

Zobrazeno 1 až 4 z 4 záznamů (filtrováno z 151 celkem záznamů)

První Předchozí 1 Další Poslední

© 2018 FIT. All rights reserved

Obrázek 5.5: Evidence předmětů, jejich povinností, studentů a vyučujících

## 5.5 Počty studentů na zkoušku

Počty studentů na zkoušku se stahují z minulého roku a vychází z nich odhady počtu studentů pro aktuální období. Jak již bylo zmíněno v návrhu u stahování vstupních dat v sekci 3.1.1, nelze jednoznačně určit, k jakému termínu počet patří. Vycházíme z odhadu,

že pokud jsou zadane čtyři nebo šest termínů, ten první je předtermín, ale jinak se jedná o 1. termín zkoušky. V sekci *Počty na zkoušku* zobrazené na obrázku 5.6 je jednoduché šipkami doleva a doprava termíny posunout, pokud přiřazení k termínu neodpovídá. Zároveň je možné chybějící počty přidat kliknutím do dané buňky a zadáním počtu studentů.

### Počty na zkoušku

Zobrazeno 10 záznamů Hledat: AGS

Předmět	Sem	Garant	Zakončení	Stud	Předtermín	1. termín	2. termín	3. termín	4. termín	5. termín	6. termín	Doleva	Doprava
AGS	LS	Zbořil František doc. Ing., Ph.D.	Zk	32	11	29	8	2				←	→

Obrázek 5.6: Editace počtů studentů k danému termínu zkoušky

## 5.6 Příznaky

Chování a význam příznaků byly vysvětleny v kapitole 3.1.6. Aplikace je pak zobrazena na obrázku 5.7. V záhlaví jsou vidět všechny evidované přednáškové skupiny (s atributem *Pro export*). Funkce **Generování příznaků** se nachází v nápovědě, která se zobrazí po kliknutí na ikonu s otazníkem vedle nadpisu. Jsou v ní shrnuta všechna pravidla popsána v návrhu v kapitole 3.1.6. Zároveň je možné najít vedle ikony s nápovědou i ikonu pro vyfiltrování předmětů, u kterých proběhla změna příznaku v minulém roce. Pokud proběhla změna minulý rok, je velice pravděpodobné, že proběhne stejná změna i tento rok. Filtr je proto vhodný, abychom na žádný předmět nezapomněli. Takové předměty jsou zvýrazněné sytě růžovou barvou.

Přiřazení příznaku k předmětu a dané přednáškové skupiny

Příznaky vycházejí z povinnosti předmětů v daném ročníku a oboru. Většinu z nich je možné pomocí pravidel generovat automaticky. Seznam pravidel s možností automatického generování naleznete v nápovědě kliknutím na modrý otazník. Existují však výjimky, které je nutné upravit ručně. Pokud v následujícím formuláři změníte příznak, uloží se příznak změny do databáze a příští rok bude tento předmět označen růžovou barvou. Takto označené předměty je možné filtrovat kliknutím na růžový filtr hned vedle nadpisu.

Skupina xx znamená něco jako 'ostatní'. Jsou v ní zařazeny přednášky z volitelných předmětů, cvičení, kde si lze vybrat 1 z N termínů, apod. V xx jsou zařazeny také předměty pro předchozí ročník (pro opakující) a v 1BIT také IZG a ve 2BIT ITU, které si studenti s lepšími výsledky mohou zapost o rok dříve.

Zobrazeno 10 záznamů Hledat: IZ

Zkratka	1BIA	1BIB	2BIA	2BIB	3BIT	1MIT	2MIT	1DVI	1EIT	2EIT	INTE	
IZ	xx	+	-	+	xx	+	-	+	-	+	-	+
IZA	xx	+	-	+	xx	+	-	+	-	+	-	+
IZG	-	+	-	+	Od Do	+	Od Do	+	xx	+	-	+
IZU	-	+	-	+	Od Do	+	Od Do	+	xx	+	-	+
VIZa	-	+	-	+	-	+	-	+	xx	+	xx	+
IZFI	xx	+	-	+	xx	+	-	+	-	+	-	+
IZMA	xx	+	-	+	xx	+	-	+	-	+	-	+

Obrázek 5.7: Přiřazení příznaků k předmětu a přednáškové skupině

## 5.7 Export dat

Důležitou částí aplikace je export dat, který se nachází u plánování pod tabulkou s rozvrhovými okny. Možnosti se liší u výuky a zkoušek. Zatímco u zkoušek jsou v hromadném exportu vždy stejné soubory, u výuky jsou různorodé. Rozdíl je v tom, že se u výuky vybírají

kombinace přednáškových skupin, které mají být v exportu rozvrhu zastoupeny. U zkoušek se exportuje rozvrh pro všechny přednáškové skupiny, ale pokud je na konci zkratky písmeno *A* nebo *B*, skupiny se spojí (tzn. spojí se přednáškové skupiny *1BIA* a *1BIB*).

## Výuka

Jak je vidět na obrázku 5.8, je možné přesně nakonfigurovat, jaké kombinace přednáškových skupin budeme v hromadném exportu potřebovat. Zvolenou kombinaci je nutné pojmenovat, jelikož se název bude zobrazovat jako odkaz do příslušného rozvrhu s danou kombinací přednáškových skupin. Stáhnout lze i samostatnou stránku po vybrání žádané kombinace skupin a kliknutí na tlačítko *HTML* pod tabulkou s výběrem. Výsledný kód stránky v HTML se vloží do textového boxu a je možné ho stáhnout ve formátu HTML po kliknutí na tlačítko *Stáhnout*.

Hromadný export bude obsahovat kromě zvolených kombinací přednáškových skupin také soubor *index.html*. Tento soubor obsahuje odkazy do rozvrhů vybraných přednáškových skupin, je zde datum poslední editace rozvrhu a jméno uživatele, který rozvrh vyexportoval. Posledním souborem, který se exportuje, je soubor *importFile.json*, ve kterém se nachází data ve formátu JSON naformátovaná k vložení do IS FIT. Pravidla pro vygenerování tohoto souboru jsou detailně popsána v kapitole 3.1.8.

The screenshot displays the 'Export rozvrhu' interface. It features two main configuration sections, each with a table of course group combinations and a corresponding HTML preview window.

**Export rozvrhu 1BIA + 1BIB**

<input checked="" type="checkbox"/> 1BIA	<input checked="" type="checkbox"/> 1BIB	<input type="checkbox"/> 2BIA
<input type="checkbox"/> 2BIB	<input type="checkbox"/> 3BIT	<input type="checkbox"/> 1MIT
<input type="checkbox"/> 2MIT	<input type="checkbox"/> 1EIT	<input type="checkbox"/> 2EIT
<input type="checkbox"/> INTE		

HTML

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
  </head>
  <body>
```

STÁHNOUT

**Export rozvrhu BIT**

<input checked="" type="checkbox"/> 1BIA	<input checked="" type="checkbox"/> 1BIB	<input checked="" type="checkbox"/> 2BIA
<input checked="" type="checkbox"/> 2BIB	<input checked="" type="checkbox"/> 3BIT	<input type="checkbox"/> 1MIT
<input type="checkbox"/> 2MIT	<input type="checkbox"/> 1EIT	<input type="checkbox"/> 2EIT
<input type="checkbox"/> INTE		

HTML

STÁHNOUT

Připravil  
Alena Tesařová

HROMADNÉ EXPORTOVAT

Obrázek 5.8: Možnosti exportu u výuky

## Zkoušky

U zkoušek není potřebná konfigurace přednáškových skupin, které se budou exportovat. Na druhou stranu přibyla ve filtrech v tabulce položka *Menza*, jak lze vidět na obrázku 5.11. Pokud je zaškrtnutá tato možnost, přidá se ke každému rozvrhovému oknu informace o počtu studentů, se kterými se bude počítat na oběd. Hodnota se vypočítá jako 2/3 z odhadnutého počtu lidí, který má přijít na danou zkoušku. Kromě menzy zde přibyla možnost exportu do HTML bez místností.

Velmi důležitým odkazem je *Přejít na časový plán zkoušek především z povinných předmětů*, který se nachází hned pod nadpisem **Časový plán zkoušek** a je určený pouze pro správce. Dojde k přesměrování do aplikace, kde dochází k editaci tabulky časového plánu zkoušek, která je blíže popsána v 2.2.1. Byly implementovány všechny funkce popsané v návrhu. Finální podoba aplikace je zobrazená na obrázku 5.9.

1BIT	2BIT	3BIT	1MIT	2MIT	1EIT	2EIT	INTE
<b>ISU</b> Programování na strojové úrovni	<b>06.05.2019</b>  16<sup>00</sup></sup> 18<sup>50</sup></sup>	<b>21.05.2019</b>  09<sup>00</sup></sup> 11<sup>50</sup></sup>	<b>06.06.2019</b>  09<sup>00</sup></sup> 11<sup>50</sup></sup>	<input checked="" type="checkbox"/>	169		
<b>INC</b> Návrh číslicových systémů	<b>18.05.2019</b>  15<sup>00</sup></sup> 16<sup>50</sup></sup>	<b>29.05.2019</b>  09<sup>00</sup></sup> 10<sup>50</sup></sup>	<b>07.06.2019</b>  14<sup>00</sup></sup> 15<sup>50</sup></sup>	<input checked="" type="checkbox"/>	170		
<b>IMA</b> Matematická analýza	<b>10.05.2019</b>  11<sup>00</sup></sup> 13<sup>50</sup></sup>	<b>30.05.2019</b>  11<sup>00</sup></sup> 13<sup>50</sup></sup>	<b>08.06.2019</b>  11<sup>00</sup></sup> 13<sup>50</sup></sup>	<input checked="" type="checkbox"/>	171		
<b>IOS</b> Operační systémy	<b>15.05.2019</b>  12<sup>00</sup></sup> 16<sup>50</sup></sup>	<b>24.05.2019</b>  13<sup>00</sup></sup> 17<sup>50</sup></sup>	<b>04.06.2019</b>  09<sup>00</sup></sup> 11<sup>50</sup></sup>	<input checked="" type="checkbox"/>	172		

+ NOVÁ POLOŽKA
👁️ NÁHLED TABULKY
🔄 VYČISTIT TABULKU A VYGENEROVAT ZNOVU
🕒 PŘIDAT TEXTY Z MINULÉHO ROKU

🕒 PŘIDAT PŘEDMĚTY Z MINULÉHO ROKU

Obrázek 5.9: Editace tabulky časového plánu zkoušek

V horním menu lze přepínat mezi přednáškovými skupinami. Po každém načtení stránky dojde ke zkontrolování a případně aktualizaci termínů zkoušek u zobrazených předmětů, u kterých je zaškrtnuto, že se mají generovat (sloupec Generovat). Je samozřejmě možné přidat i jiné záznamy kliknutím na tlačítko *Nová položka*. Jelikož jsou buňky ve formátu HTML, přidala jsem tlačítko pro náhled tabulky. Funkce *Vyčistit a vygenerovat znovu* odstraní všechny záznamy tabulky a přidá předměty, které jsou pro danou přednáškovou skupinu a ročník **povinné**, **povinně volitelné** nebo **doporučené**. Po kliknutí na tlačítko *Přidat texty z minulého roku* se přidají do tabulky záznamy, které minulý rok nebyly označeny ke generování. Podobnou funkcí je i *Přidat předměty z minulého roku*, která doplní

tabulku o předměty z minulého roku, které byly označeny ke generování. Celkový náhled exportu všech přednáškových skupin si lze prohlédnout po kliknutí na *Náhled exportu*, který se nachází vedle nadpisu nad tabulkou (na obrázku není vidět).

U časového plánu se vždy navíc zobrazují texty, které informují studenty o obecných informacích a omezeních ke zkouškám. Tyto texty je možné editovat hned pod tabulkou s časovým plánem. Současně s editací je zde možné text přiřadit přednáškové skupině, pro kterou bude text určený. Příklad přiřazení je vidět na obrázku 5.10, kde je text přiřazený všem skupinám s prioritou 10. Obecně platí, že čím větší priorita, tím se bude text nacházet výše v seznamu.

Text	Priorita	1BIT	2BIT	3BIT	1MIT	2MIT	1EIT	2EIT	INTE	Nový	Odstranit
< ><b><span style=color:red>Nevyužité zkušební termíny propadají bez náhrady, a nezkoumá se důvod nedostavení se studenta na zkušební termín.</span></b></td><td>10</td><td><input checked="" type=											

Obrázek 5.10: Konfigurace textů zobrazených pod tabulkou časového plánu

V momentě, kdy je tabulka kompletní a exporty jsou zkontrolovány v náhledech, je možné přejít k hromadnému exportu. Kromě odkazů u tabulky s časovým plánem bude export obsahovat soubor *místnosti.html*, ve kterém se nachází rozvrh s místnostmi, *index.html* pro rozvrh bez místností a nesmí chybět ani *menza.html*, ve kterém je rozvrh bez místností s počty obědů. Poslední dva soubory jsou určeny pouze pro správce. Jedná se o tabulku odhadů počtů na zkoušky a soubor *importFile.json*, ve kterém se nachází data ve formátu JSON určená k vložení do IS FIT. Pravidla pro vygenerování tohoto souboru jsou detailně popsána v kapitole 3.1.8.

#### Časový plán zkoušek

[→ Přejít na časový plán zkoušek především z povinných předmětů](#)

#### Export rozvrhu

<input checked="" type="checkbox"/> 1BIA	<input checked="" type="checkbox"/> 1BIB	<input checked="" type="checkbox"/> 2BIA
<input checked="" type="checkbox"/> 2BIB	<input checked="" type="checkbox"/> 3BIT	<input checked="" type="checkbox"/> 1MIT
<input checked="" type="checkbox"/> 2MIT	<input type="checkbox"/> 1EIT	<input type="checkbox"/> 2EIT
<input type="checkbox"/> INTE	<input type="checkbox"/> Menza	

```
HTML
<!DOCTYPE html>
<html>
<head>
  <meta charset="UTF-8">
</head>
STÁHNOUT
```

#### Hromadný export

Hromadný export zahrnuje soubory: *index.html*, *místnosti.html*, *menza.html*, *1bitXX - 3bitXX*, kde XX je zkratka semestru. Dále je obsažen soubor ve formátu json určený pro import na server IS.

Připravil  
Alena Tesařová

Obrázek 5.11: Možnosti exportu u zkoušek



## Kapitola 6

# Testování a nasazení aplikace

Testování aplikace díky vzniku nového fakultního REST API probíhalo na reálných datech FIT. Jako studentka jsem měla přístup ke všem datům kromě seznamů studentů registrovaných u předmětu. Aby byla data různorodá, stažení dat probíhalo z akademických let 2018/2019, 2017/2018 a 2016/2017. Po importu docházelo vždy k manuální kontrole vložených dat podle referenčního REST rozhraní. Tímto se částečně ověřila funkčnost stahování z REST (zatím nebyl ověřen seznam registrovaných studentů). Po vložení všech vstupních dat jsem testovala kompletnost požadavků u výuky a u zkoušek. Během testování byly nalezeny chyby související se skrýváním prvního řádku v předmětu nebo také špatně zvolené období, které se používalo pro doplnění dat. Všechny chyby byly následně opraveny. Rozvrh byl při testování vložen ve formátu CSV ze současné aplikace na plánování a data se porovnávala. Testovalo se správné umístění rozvrhových oken do místností, správné zobrazení grafu povinností na pravé straně od kolizí u plánování a u exportů se testovalo, aby všechna data a formáty dat odpovídaly současné aplikaci. Kontrolou se ověřilo jak správné stažení povinností předmětů, tak i správné vygenerování příznaků u přednáškových skupin. Aplikace byla nasazena v rámci testování na fakultní server **pcknot4** (PHP v. 7.2.15) a společně s ní byla nahrána i databáze přes aplikaci phpmyadmin. Proces probíhal následujícím způsobem:

1. Zdrojové kódy byly vloženy na server
2. Chybějící knihovny byly staženy pomocí nástroje *composer*
3. Bylo potřeba nastavit práva zápisu složkám *www*, *temp* a *log*
4. Databáze byla nahrána na server
5. Bylo pozměněno heslo a název databáze v souboru *config.local.neon*

Po vykonání všech kroků bylo možné se do aplikace přihlásit přes přihlašovací jméno *admin* a heslo *admin1234*. Dne 26. 3. 2019 proběhlo první testování aplikace Ing. Jaroslavem Dytrychem, Ph.D. a Ing. Petrem Veigendem. Učitelé měli za cíl vyexportovat soubor pro vložení do IS pro rozvrh zkoušek, který již byl vytvořený v aplikaci, která se současně používá pro plánování. V databázi byly uloženy pouze základní číselníky jako jsou dny v týdnu, časové úseky, semestry, práva, stavy rozvrhů, typy termínů zkoušek, přednáškové skupiny a podobně. Postup byl následující:

1. Založení **nového období** a **rozvrhu** (hlavní strana)

2. Stažení **předmětů** a **preferencí** (hlavní strana)
3. Stažení **kolizí** (hlavní strana)
4. Stažení **počtů na zkoušku** z minulých let (hlavní strana)
5. Doimportování **oborů** a vybrání oborů, které se budou zobrazovat v grafu vedle rozvrhu (správa oborů)
6. Doimportování **místností**, vybrání místností, které se budou zobrazovat v rozvrhu, a doplnění kapacit ob jedno a ob dvě místa (správa místností)
7. Vygenerování **příznaků** a doplnění příznaků u výjimek (přiřazení příznaků u období)
8. **Vyexportování** rozvrhu (plánovací strana aplikace)

Jelikož stahování prováděl vyučující, který má více práv, bylo otestováno i stažení registrovaných studentů, kde došlo ke dvěma problémům. Prvním problémem bylo zjištění, že *login* nemusí odpovídat e-mailové adrese. Řešením bylo přidat do rozhraní REST atribut *login*. Druhý problém způsobila neznámá povinnost se zkratkou *O*. Jedná se o povinnost s označením *Ostatní* a nachází se u předmětů, které si student platí. Řešením problému bylo upravení aplikace tak, aby se neznámé povinnosti ukládaly rovnou do tabulky *Povinnosti*.

Ve vyexportovaném souboru pro import do IS pak došlo pouze k jedné úpravě. Jednalo se o přání učitele, aby se u předmětů v angličtině posílal i termín zkoušky v angličtině. Přání bylo doplněno do aplikace. Vytvořený soubor byl následně důkladně zkontrolován Ing. Petrem Veigendem a importován do IS FIT pro zkoušky na letní semestr 2018.

Dne 26. 4. 2019 proběhlo druhé kolo testování aplikace se stejným složením uživatelů. Tentokrát bylo testování důkladnější, jelikož v aplikaci probíhalo reálné plánování a ne jenom export a import dat. Cílem bylo vytvořit rozvrh **výuky** na zimní semestr pro akademický rok 2019. Kroky 1 – 7 byly stejné jako u zkoušek. Navíc bylo potřeba:

- doplnění výchozích rozsahů přednáškových skupin (editace rozsahů přednáškových skupin ve správě období),
- přidání identifikace místností (správa místností),
- doplnění povinností předmětů u neakreditovaných specializací, jelikož tato data nebylo možné stáhnout z REST (správa předmětů),
- doplnění preferencí na výuku z minulých let (správa požadavků výuky).

Při plánování bylo nalezeno a následně opraveno několik chyb. Prvním nalezeným problémem byl **nulový počet zapsaných studentů**, jelikož v momentě plánování nebyli žádní studenti zapsaní, ale pouze registrovaní. Proto bylo vytvořeno nové rozhraní REST *registered*, pomocí kterého je možné takové studenty zpracovat a tímto byl problém vyřešen.

Dalším problémem bylo plánování rozvrhu pro **Centrum výpočetní techniky (CVT)**, se kterým nejprve aplikace nepočítala. Jedná se o centrum, které zajišťuje provoz počítačových laboratoří na celé fakultě, fakultní počítačové síť a je využíváno jak pro rozvrhovanou výuku, tak pro řešení různých projektů. Z pohledu exportu a plánování je pak potřeba zobrazení pohledu pouze s místnostmi patřícími CVT. Proto byla nově vytvořena tabulka s typy místností. Filtr na tento typ byl následně implementován k plánování výuky a k exportu.

Poslední zásadnější úprava proběhla kvůli požadavku na **pohled na rozvrh bez povinných předmětů z jiných ročníků**. Pokud je například vyfiltrovaná přednášková skupina *2BIA*, rozvrhář chce mít možnost odstranit z pohledu předměty, které jsou povinné v prvním ročníku a v rozvrhu pro 2. ročník jsou kvůli opakujícím studentům. Důvodem je lepší přehlednost pro studenty, kteří neopakují žádný předmět z minulého roku. Tato možnost byla přidána vedle filtrování přednáškových skupin k plánování výuky a k exportu dat.

Ostatní chyby nebyly zásadního charakteru a oprava byla provedena ihned. Výčet některých chyb:

- místo zkratky specializace se zobrazoval název programu u povinností předmětů (export),
- chybělo upozornění na překročení délky časového okna v rozvrhu (plánovací strana aplikace),
- nebylo možné přidat nový údaj o počtu studentů, kteří přišli na daný termín zkoušky (správa počtů studentů na zkoušku),
- nebylo možné smazat povinnost předmětu (správa předmětů),
- špatně se aktualizoval počet cvičení u předmětu (správa předmětů).

Zobrazeno 10 záznamů Hledat:

	BIT	MMI	MBI	MSK	MMM	MBS	MPV	MIS	MIN	MGM
AEU	0	0	0	0	0	0	0	0	0	0
AGS		0	0	0	0	0	0	0	1	0
AIS		0	0	2	0	0	0	2	0	0
AIT	2									
AVS			0	2	0	0	2	0	0	2
BAN2	2									
BAN3	2									
BAN4	2									
BAYa										
BIO		0	1	0	0	2	0	0	0	0

Zobrazeno 1 až 10 z 127 záznamů Předchozí 1 2 3 4 5 ... 13 Další

Obrázek 6.1: Tabulka povinností

Vedle oprav chyb vzniklo i několik vylepšení, která byla hned implementována. Přibyla například **tabulka povinností**, která se bude nově evidovat u konkrétního období. Jak je vidět na obrázku 6.1, jedná se o matici, která zobrazuje povinnosti předmětů pro konkrétní

obor nebo specializaci. Barva rozlišuje typy povinností, kde tmavě červeně jsou povinné předměty, světle červeně pak povinně volitelné a nejsvětleji červeně jsou označeny volitelné předměty. Číslo znamená ročník. Pokud platí povinnost pro všechny ročníky, je v tabulce uvedena nula. Tabulka je potřebná pro závěrečnou kontrolu rozvrhu, aby nedošlo k porušení některých z tvrdých kritérií (např. překrývání povinných předmětů v rozvrhu). Pro jednodušší vyhledávání více předmětů je v tabulce možné vyhledávat i pomocí regulárních výrazů.

Druhým vylepšením je **pohled pro menzu**. Jelikož slouží export rozvrhu pro menzu pouze kuchařkám na FIT, je zbytečné, aby se do kompletního počtu obědů za den započítávali studenti, kteří nemají zkoušku v nějaké místnosti na FIT. Mohou nastat celkem tři možnosti:

1. **Všechny místnosti** pro termín zkoušky jsou na FIT. V tomto případě se do celkového počtu započítávají všichni studenti, kteří na termín mají přijít.
2. **Existuje** místnost, která **není** na FIT a zároveň alespoň jedna místnost termínu zkoušky na FITu je. Jelikož se většinou jedná o případ, že skupina A má zkoušku na FIT a skupina B na FEKT, budeme do odhadu počítat právě jednu polovinu studentů, kteří mají přijít na daný termín zkoušky.
3. **Všechny místnosti** na termín zkoušky jsou **mimo** FIT. Pokud není žádná místnost na FIT, je zbytečné do celkového počtu studenty počítat.

Po třech dnech plánování byla nakonec dne 28. 4. 2019 vytvořena první verze rozvrhu pro zimní semestr 2019/2020, kterou si lze prohlédnout na stránkách studijního poradenství FIT na následující adrese: [http://www.fit.vutbr.cz/study/advisor/20192020/priprava\\_ZS/](http://www.fit.vutbr.cz/study/advisor/20192020/priprava_ZS/). Všechny vytvořené stránky byly vyexportované z méj aplikace a zkontrolované P. Veigendem a J. Dytrychem.

Testování probíhalo v prohlížečích Google Chrome verze 73 a Mozilla Firefox verze 66.

# Kapitola 7

## Závěr

V rámci této bakalářské práce jsem prostudovala problematiku plánování výuky a zkoušek na *Fakultě informačních technologií Vysokého učení technického v Brně*. Z analýzy vyplynuly největší problémy současné plánovací aplikace, kterými jsou stahování vstupních dat, kompletace požadavků a export výstupních dat.

Předměty, se kterými současná aplikace pracuje, se nestahují. Dochází pouze k vizuální kontrole, jestli se data nezměnila, a případně se ručně doplní předměty a jejich povinnosti do aplikace. Stahují se pouze některé informace, jako jsou například kolize, ale je potřeba manuálního spuštění skriptu. Skript data uloží do souboru, se kterým pak aplikace umí pracovat. Kompletace požadavků probíhá v programu Microsoft Excel a kopírování dat je možné pouze manuálním zkopírováním vybraných požadavků z minulého roku a jejich vložení do souboru. Data pak nejsou žádným způsobem propojená s plánovací aplikací. Pro export existuje několik vytvořených skriptů, které umí seskládat stránky v HTML pro vložení na web. Jedná se o celkem složitý postup, který se musí vykonat jednotlivě pro přednáškové skupiny, pro které rozvrh exportujeme. Chybí automatické nahrání rozvrhu do IS FIT, a proto se dělá vkládání manuálně přes formuláře v IS.

Nově navržená a implementovaná aplikace obsahuje všechny kroky plánování na jednom místě. Je možné automaticky stáhnout data, zkompletovat požadavky a případně upravit data jako jsou předměty, povinnosti, učitelé, přednáškové skupiny, místnosti a podobně. Pro snadnější přiřazení předmětů k přednáškovým skupinám bylo využito mechanismu přiřazování příznaků, podle kterých pak předměty u skupin filtrujeme a exportujeme. Nejdůležitější jsou příznaky pro vytváření souboru pro import do IS. Všechny exporty si v aplikaci lze zobrazit a případně upravit podle potřeb.

Aplikace podporuje režimy zobrazování rozvrhů zkoušek s místnostmi, bez místnosti a s počty obědů. Všechny změny rozvrhových oken se ukládají dynamicky a zároveň jsou poskytovány informace o požadavcích vyučujících pro daný předmět. Je zajištěno, aby nedošlo k přepsání dat u rozvrhu, systémem uzavírání a odemykání rozvrhu. Implementované jsou nástroje zobrazení kolizí dvou předmětů a nástroj pro hledání kombinace místností, která poskytne dostatečnou kapacitu pro daný předmět. Navíc aplikace umožňuje poloautomatizované přiřazení předmětů do místností na základě požadavků u předmětů a odhadu počtu studentů, kteří mohou přijít na daný termín.

Do aplikace se může registrovat kdokoli jako student. Studenti vidí pouze zveřejněné rozvrhy správců a svoje rozvrhy. Nemají přístup ke konfiguračním aplikacím, ke stahování dat a exportování je omezeno o soubory pro import do IS fakulty a odhady počtů studentů u zkoušek. Mohou vytvářet nové verze ze zveřejněného rozvrhu a posílat je na schválení

správci, který může jejich rozvrh schválit, nebo odmítnout. Vždy je možné ke změně přidat komentář.

V rámci dalšího vývoje bude do aplikace přidáno tlačítko zpět, nástroje pro zobrazení počtu společných studentů u více jak dvou předmětů, kolize jednoho předmětu v rámci časového okna  $\pm 2$  dní, upozornění, pokud učitel učí paralelně ve stejný čas, a kontroly, jestli nebyla porušena další z tvrdých či měkkých kritérií.

# Literatura

- [1] Atkinson, L.; Suraski, Z.: *Core PHP programming*. Pearson Education, Inc., třetí vydání, 2004, ISBN 0-13-046346-9.
- [2] Beran, V.: *Tvorba uživatelských rozhraní*. [Online; navštíveno 12.04.2019].  
URL <https://www.fit.vutbr.cz/study/courses/ITU/private/lectures/Web/itu-webgui-cs.pdf>
- [3] Bos, B.: *Cascading Style Sheets*. [Online; navštíveno 1.5.2019].  
URL <https://www.w3.org/Style/CSS/>
- [4] Burget, R.: *Ajax a REST*. [Online; navštíveno 1.5.2019].  
URL <https://wis.fit.vutbr.cz/FIT/st/cfs.php?file=%2Fcourse%2FIIIS-IT%2Flectures%2Fp09 AJAX.pdf&cid=12750>
- [5] Dytrych, J.; Veigend, P.; Nunvářová, S.: *Studijní poradenství*. Březen 2019, [Online; navštíveno 1.5.2019].  
URL <http://www.fit.vutbr.cz/study/advisor/.css>
- [6] *ERUDIO s.r.o.* [Online; navštíveno 03.04.2019].  
URL <https://www.erudio.cz/>
- [7] Galante, A.; Harborow, B.; Sears, C.; aj.: *Bootstrap 4*. [Online; navštíveno 12.04.2019].  
URL <https://getbootstrap.com/>
- [8] Horký, A.: *Systém pro pokročilé plánování*. Diplomová práce, Vysoké učení technické v Brně, 2015, 5 s.
- [9] Hsiao-Lan, F.: *Genetic Algorithms in Timetabling and Scheduling*. Dizertační práce, University of Edinburgh, 1994, 1 s.
- [10] *ITC 2019: International Timetabling Competition*. [Online; navštíveno 12.04.2019].  
URL <https://www.itc2019.org/home>
- [11] *JavaScript Versions*. [Online; navštíveno 22.04.2019].  
URL [https://www.w3schools.com/js/js\\_versions.asp](https://www.w3schools.com/js/js_versions.asp)
- [12] Kacerovsky, M.; Scholz, F.; Grňo, M.; aj.: *JavaScript*. [Online; navštíveno 12.04.2019].  
URL <https://developer.mozilla.org/cs/docs/Web/JavaScript>
- [13] Kubalcová, M.: *Porovnání programů pro plánování rozvrhů a zkoušek*. Bakalářská práce, Vysoké učení technické v Brně, 2012.

- [14] Makovický, M.; Scholz, F.; aj.: *CSS*. Březen 2019, [Online; navštíveno 22.04.2019].  
URL <https://developer.mozilla.org/cs/docs/Web/CSS>
- [15] Malý, M.: *REST: architektura pro webové API*. Srpen 2009, [Online; navštíveno 1.5.2019].  
URL <https://www.zdrojak.cz/clanky/rest-architektura-pro-webove-api/>
- [16] *Material Design*. [Online; navštíveno 12.04.2019].  
URL <https://material.io/design/>
- [17] Müller, T.: *UniTime: University timetabling – Comprehensive academic scheduling solutions*. [Online; navštíveno 03.04.2019].  
URL <https://www.unitime.org/>
- [18] Müller, T.; Rudová, H.; Müllerová, Z.: Rozvrhování předmětů na Masarykově univerzitě. In *Jan Rychlík, EUNIS-CZ – Sborník příspěvků ke konferenci Studijní a informační systémy a IS na podporu výzkumu a vývoje*, 2015, ISBN 978-80-261-0507-7, s. 51–57, 7 s.
- [19] *PATAT Conferences*. [Online; navštíveno 03.04.2019].  
URL <http://www.patatconference.org/>
- [20] Simoes, A.: *Material Design*. Březen 2019, [Online; navštíveno 22.04.2019].  
URL <https://www.interaction-design.org/literature/article/google-s-material-design-android-design-language>
- [21] *Soutěž ITC 2019 má již 91 registrací z 34 zemí. Stále se lze registrovat*. [Online; navštíveno 03.04.2019].  
URL <https://www.muni.cz/kalendar/4891-soutez-itc-2019-ma-jiz-91-registraci-z-34-zemi-stale-se-lze-registrovat>
- [22] *Studijní programy*. [Online; navštíveno 03.04.2019].  
URL <http://www.fit.vutbr.cz/study/programs/index.php.cs>
- [23] *Usage statistics and market share of PHP for websites*. [Online; navštíveno 22.04.2019].  
URL <https://w3techs.com/technologies/details/pl-php/all/all>
- [24] *Vysokoškolský pedagog*. [Online; navštíveno 09.04.2019].  
URL <https://www.platy.cz/platy/skolstvi-vzdelavani-veda-vyzkum/vysokoskolsky-pedagog>
- [25] Zbořil, F.: *Úvod, definice umělé inteligence (UI), typy UI úloh, přehled metod řešení těchto úloh*. [Online; navštíveno 03.04.2019].  
URL [https://www.fit.vutbr.cz/study/courses/IZU/private/1819izu\\_1.pdf](https://www.fit.vutbr.cz/study/courses/IZU/private/1819izu_1.pdf)
- [26] Čillo, V.: *Program pro plánování rozvrhů*. Diplomová práce, Vysoké učení technické v Brně, 2017.
- [27] Čápka, D.: *Úvod do Nette frameworku pro PHP*. Červenec 2018, [Online; navštíveno 1.5.2018].  
URL <https://www.itnetwork.cz/php/nette/zaklady/uvod-do-php-frameworku-nette/>



- [28] Šorm, M.; Lukač, M.; aj.: *Roger*. [Online; navštíveno 03.04.2019].  
URL <https://www.rozvrhy.eu/cs/index>
- [29] Štěpánek, P.: *Rozhodnutí č. 24/2017 - Časová struktura výukové části akademického roku*. Květen 2017.  
URL  
<https://www.vutbr.cz/uredni-deska/vnitrni-predpisy-a-dokumenty/d143516>