

VŠB – TECHNICKÁ UNIVERZITA OSTRAVA
FAKULTA STROJNÍ
KATEDRA AUTOMATIZAČNÍ TECHNIKY A ŘÍZENÍ

Automatizovaný import rozvrhu pedagoga do kalendáře MS Outlook

Automated Import of Schedule Teacher into MS Outlook Calendar

Student:

Miroslav Čermák

Vedoucí bakalářské práce:

Ing. David Fojtík, Ph.D.

Ostrava 2013

VŠB - Technická univerzita Ostrava
Fakulta strojní
Katedra automatizační techniky a řízení

Zadání bakalářské práce

Student: **Miroslav Čermák**
Studijní program: B2341 Strojírenství
Studijní obor: 3902R001 Aplikovaná informatika a řízení
Téma: **Automatizovaný import rozvrhu pedagoga do kalendáře MS Outlook
Automated Import of Schedule Teacher into MS Outlook Calendar**

Zásady pro vypracování:

1. Navrhněte varianty automatického vkládání kalendářních dat do aplikace MS Outlook včetně jejich srovnání.
2. Charakterizujte možnosti automatizovaného získávání dat aktuálního rozvrhu pedagoga na VŠB-TU Ostrava.
3. Navrhněte a zrealizujte programové řešení automatizovaného vkládání rozvrhu pedagoga do kalendáře aplikace MS Outlook.
4. Zhodnoťte dosažené výsledky a navrhněte směry dalšího řešení.

Seznam doporučené odborné literatury:

AWSON, F. & STENERSON, D. 1998 Internet Calendaring and Scheduling Core Object Specification (iCalendar). Copyright (C) The Internet Society (1998). November 1998. Dostupný z WWW: <<http://www.ietf.org/rfc/rfc2445.txt>>
FARANA, R., SMUTNÝ, L., VÍTEČEK, A. 1999. Zpracování odborných textů z oblasti automatizace a informatiky. 1. vyd. Ostrava : VŠB-TU Ostrava, 1999. 68 s. ISBN 80-7078-737-6.
KOSEK, J. 1998 HTML - tvorba dokonalých WWW stránek - podrobný průvodce. Grada Publishing 1998. 296 stran. ISBN 80-7169-608-0.
MICROSOFT CORPORATION 2009A. Outlook 2007 [online]. ©2009. Dostupný z WWW: <[http://msdn.microsoft.com/cs-cz/library/bb187387\(en-us\).aspx](http://msdn.microsoft.com/cs-cz/library/bb187387(en-us).aspx)>.
MICROSOFT CORPORATION 2009B. LINQ to XML [online]. ©2009. Dostupný z WWW: <<http://msdn.microsoft.com/en-us/library/bb387098.aspx>>.
SLOVAK, K. 2007 Professional Outlook 2007 Programming. Wrox Press 2007. 454 p. ISBN 0470049944

Formální náležitosti a rozsah bakalářské práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

Vedoucí bakalářské práce: **Ing. David Fojtík, Ph.D.**

Datum zadání: 14.12.2012

Datum odevzdání: 20.05.2013

Tůma

prof. Ing. Jiří Tůma, CSc.
vedoucí katedry



Hlavatý

doc. Ing. Ivo Hlavatý, Ph.D.
děkan fakulty

Místopřísežné prohlášení studenta

Prohlašuji, že jsem celou diplomovou (bakalářskou) práci včetně příloh vypracoval samostatně pod vedením vedoucího diplomové (bakalářské) práce a uvedl jsem všechny použité podklady a literaturu.

V Ostravě20.5.2013.....


.....

podpis studenta

Prohlašuji, že

- jsem byl seznámen s tím, že na moji diplomovou (bakalářskou) práci se plně vztahuje zákon č. 121/2000 Sb., autorský zákon, zejména § 35 – užití díla v rámci občanských a náboženských obřadů, v rámci školních představení a užití díla školního a § 60 – školní dílo.
- beru na vědomí, že Vysoká škola báňská – Technická univerzita Ostrava (dále jen „VŠB-TUO“) má právo nevýdělečně ke své vnitřní potřebě diplomovou (bakalářskou) práci užít (§ 35 odst. 3).
- souhlasím s tím, že diplomová (bakalářská) práce bude v elektronické podobě uložena v Ústřední knihovně VŠB-TUO k nahlédnutí a jeden výtisk bude uložen u vedoucího diplomové (bakalářské) práce. Souhlasím s tím, že údaje kvalifikační práci budou zveřejněny v informačním systému VŠB-TUO.
- bylo sjednáno, že s VŠB-TUO, v případě zájmu z její strany, uzavřu licenční smlouvu s oprávněním užít dílo v rozsahu § 12 odst. 4 autorského zákona.
- bylo sjednáno, že užít své dílo – diplomovou (bakalářskou) práci nebo poskytnout licenci k jejímu využití mohu jen se souhlasem VŠB-TUO, která je oprávněna
- v takovém případě ode mne požadovat přiměřený příspěvek na úhradu nákladů, které byly VŠB-TUO na vytvoření díla vynaloženy (až do jejich skutečné výše).
- beru na vědomí, že odevzdáním své práce souhlasím se zveřejněním své práce
- podle zákona č. 111/1998 Sb., o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších předpisů, bez ohledu na výsledek její obhajoby.

V Ostravě 20. 5. 2013

.....


podpis

Jméno a příjmení autora práce: Miroslav Čermák

Adresa trvalého pobytu autora práce: Mírová 538/6 Kobeřice, 747 27

ANOTACE BAKALÁŘSKÉ PRÁCE

ČERMÁK, M. *Automatizovaný import rozvrhu pedagoga do kalendáře MS Outlook*
Ostrava: katedra automatizační techniky a řízení, Fakulta strojní VŠB –
Technická univerzita Ostrava, 2013, 36 s.
Bakalářská práce, vedoucí FOJTÍK, D.

Cílem této bakalářské práce je navrhnout a následně zrealizovat automatické vkládání rozvrhu pedagoga z rozvrhového systému VŠB-TUO do aplikace MS Outlook.

Teoretická část se zabývá analýzou jednotlivých způsobů vkládání dat a jejich srovnání. Dále popisuje strukturu výměnného formátu iCalendar a některých jeho komponentů, možný způsob získávání rozvrhových dat pedagoga a následnou volbu řešení.

Praktická část obsahuje návrh a realizaci internetové aplikace, která bude potřebná data získávat, zpracovávat, a následně převádět do formátu iCalendar a také importovat do aplikace MS Outlook, z možností stažení objektu iCalendar do souboru na počítači.

ANNOTATION OF THE BACHELOR THESIS

ČERMÁK, M. *Automated Import of Schedule Teacher into MS Outlook Calendar*
Department of Control Systems and Instrumentation, Faculty of
Mechanical Engineering VŠB – Technical University of Ostrava 36 p.
Bachelor Thesis , supervised by: FOJTÍK, D.

The purpose of this thesis is to design and implement automatic insertion of the teacher's schedule from the schedule system of VŠB-TUO into MS Outlook.

The theoretical part analyzes individual methods of the schedule data insertion and their comparison. It also describes the structure of the exchange iCalendar format and some of its components, the possible ways of obtaining schedule data of the teacher and the subsequent choice of the solution.

The practical part includes the design and implementation of web-based application that will acquire required data, process, and subsequently transferred to the iCalendar format and import into MS Outlook, with an option to download iCalendar object to a file on your computer.

Poděkování

Děkuji vedoucímu mé bakalářské práce Ing. Davidu Fojtíkovi, Ph.D. za ochotu při odborných konzultacích a také Ing. Jaroslavu Juráňovi za poskytnutí výstupu z rozvrhu.

OBSAH

Úvod	9
1 MS Outlook	10
1.1 Outlook kalendář	10
1.2 Způsoby automatického vkládání událostí	11
1.2.1 Vkládání programově	12
1.2.2 Import ze souboru	13
1.3 Rozhodnutí	14
2 Výměnné formáty iCalendar a vCalendar	14
2.1 Řádky obsahu souboru	15
2.2 Struktura objektu iCalendar	15
2.2.1 Komponent VTIMEZONE	16
2.2.2 Komponent VEVENT	18
2.2.3 Komponent VALARM	19
3 Získávání dat	20
4 Volba programovacího jazyka a technologie	22
4.1 XSLT procesor	22
4.2 Document Object Model	22
5 Návrh programového řešení	23
5.1 Obecné konstanty	24
5.2 Funkce ics	24
5.2.1 Řešení získávání dat	24
5.2.2 Konstrukce objektu iCalendar	27
5.2.3 Konstanty funkce ics	31
5.3 Funkce servertest	31
5.4 Sdílený soubor ics	31
5.5 Soubor ics ke stažení	32
6 Grafický návrh	33
7 Realizace a používání	34
7.1 Návod k použití	34
8 Závěr	35
9 Seznam použité literatury	36
10 Přílohy	37

Použité zkratky

C#	C Sharp
CRLF	Carriage return + Line feed
HTML	Hypertext Markup Language
HTTP	Hypertext Transfer Protocol
ID	Identifikace ve výpočetní technice
IANA	Internet Assigned Numbers Authority
LAN	Local Area Network
MIME	Multipurpose Internet Mail Extensions
MS	Microsoft
PHP	Hypertext Preprocessor (skriptovací programovací jazyk)
SMTP	Simple Mail Transfer Protocol
URI	Uniform Resource Identifier
VŠB-TUO	Vysoká škola báňská technická univerzita
Wi-Fi	Wireless LAN (wireless fidelity)
XML	Extensible Markup Language - rozšiřitelný značkovací jazyk
XSLT	eXtensible Stylesheet Language Transformations
DOM	Document Object Model

Úvod

Plánování formou stolních kalendářů a papírových diářů se pomalu stává historií. V dnešní přetechnizované době, kdy připojení k internetu a odesílání emailu zvládne téměř každý mobilní telefon, se také plánování a organizování času vydalo tímto směrem.

Vyřizování pošty, plánování schůzek a řízení úkolů se dnes čím dál tím víc vyžaduje téměř ve všech firmách a institucích. K řízení těchto procesů vzniklo mnoho různých softwarů a konkurenčních aplikací, které komplexně řeší všechny tyto požadavky. Jedna z nejrozšířenějších aplikací tohoto druhu je MS Outlook, která je také používána zaměstnanci VŠB-TUO.

K tomu, aby funkce kalendář dala zaměstnanci plně využívat, musí se do ní zadat informace o jednotlivých naplánovaných událostech. To je v některých případech velice zdoluhavé, obzvláště pokud je událostí mnoho.

Velkou částí naplánovaných událostí pedagogů je rozvrh hodin, které vyučují. Plánování rozvrhů na VŠB-TUO je řízeno a uloženo v systému jiném než MS Outlook. Tyto dva systémy fungují naprosto nezávisle na sobě, přesto je třeba převést data ze systému rozvrhů do aplikace MS Outlook, aby se zaměstnancům usnadnila práce, a nemuseli používat na plánování dva systémy. Doposud se tento proces prováděl ručně, přepsáním potřebných dat pomocí klávesnice. Zaměstnanec tak stráví mnoho času zadáváním resp. opisováním údajů z jednoho okna do druhého. Aplikaci tím spíše obsluhuje, než aby její funkce využíval.

Tato bakalářská práce řeší možnosti, jak tento proces co nejvíce zautomatizovat a jeho realizaci.

1 MS Outlook

Microsoft Outlook je e-mailový a organizační klient od společnosti Microsoft, který je standardně součástí Microsoft Office. [1]

Obsahuje moduly

- ▶ klienta elektronické pošty
- ▶ vedení kontaktů
- ▶ organizaci času (kalendář)
- ▶ plánování úkolů
- ▶ poznámky
- ▶ deník

Tento projekt se bude zabývat pouze kalendářem a jeho funkcemi.

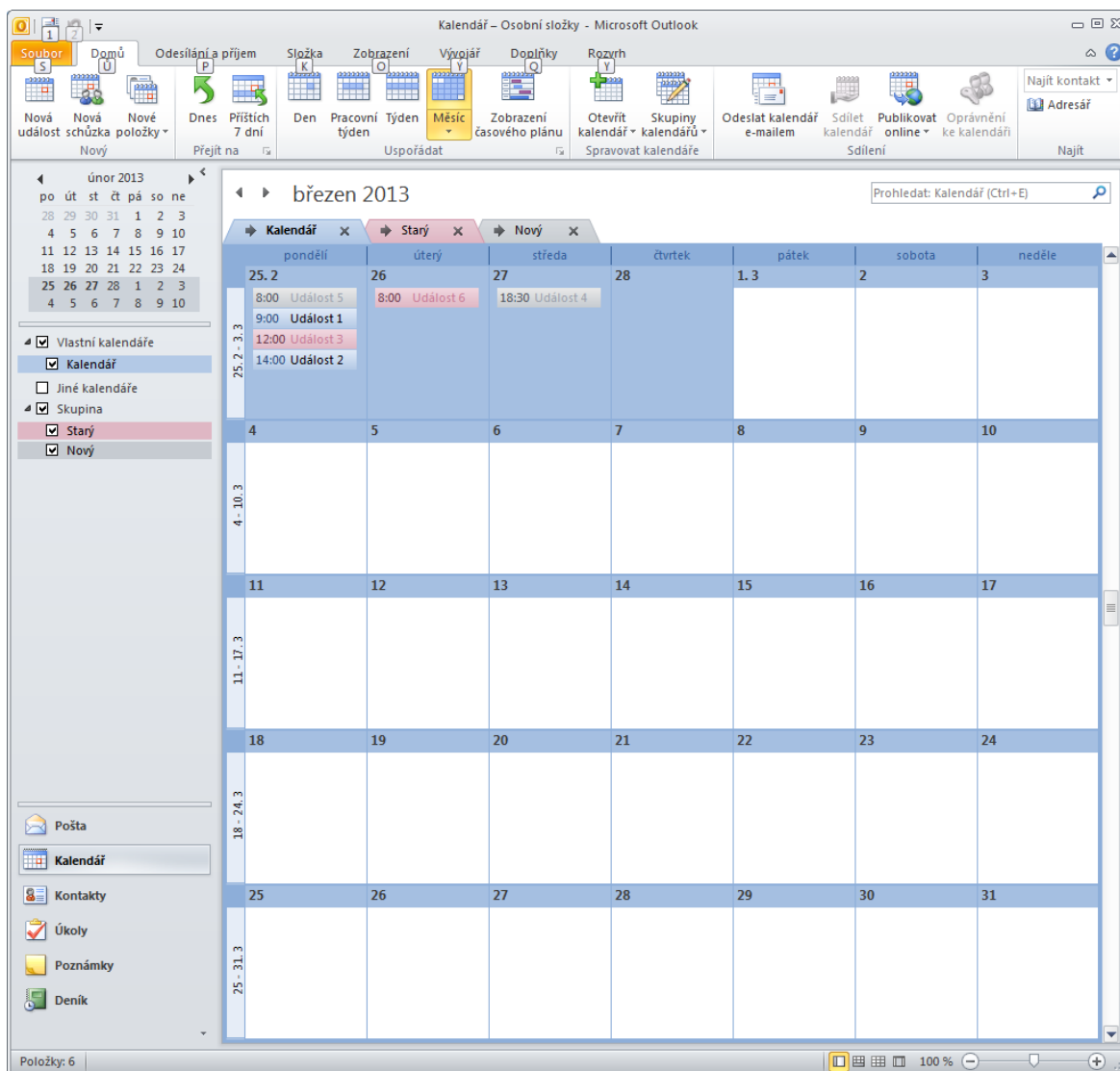
1.1 Outlook kalendář

Outlook kalendář funguje jako správce kalendářů. Na Obr. 1.1 je snímek okna MS Outlook v režimu kalendář. Vlevo vidíme postranní panel a v jeho střední části se nachází stromová struktura složek, podsložek a kalendářů, které má uživatel k dispozici. Tuto strukturu může uživatel měnit. Může vytvářet, mazat, přesouvat a přejmenovat složky podsložky i kalendáře. Složka „vlastní kalendáře“ a kalendář s názvem „kalendář“ jsou výchozí objekty a nejdou smazat ani přejmenovat.

V horní části je skupina záložek (panelů) tzv. „Ribbon“, který se objevuje v MS Office od verze 2007. Záložka „Domů“ obsahuje nejčastěji používaná funkční tlačítka pro práci s kalendářem. Na příklad k vytváření nových událostí nebo schůzek, různé možnosti zobrazení a uspořádání a také otevírání a ukládání kalendářů z jiných souboru nebo vytváření kalendářových skupin.

Vkládat kalendáře můžeme z adresáře, z místnosti, ze souboru uloženého na lokálním disku nebo z internetu.

Kalendáře lze zobrazit jednotlivě, nebo překrýt přes sebe tak, že uvidíme události dvou a více kalendářů zobrazeny v jednom kalendáři, rozlišeny barevně.



Obr. 1.1: Screen shot okna MS Outlook

1.2 Způsoby automatického vkládání událostí

Cílem tohoto projektu je najít řešení, jak vložit rozvrh pedagoga do kalendáře v aplikaci MS Outlook, případně i do jiných kalendářních aplikací tak, aby byl přehledný a aby s tím měl uživatel co nejméně práce.

Existují dvě základní možnosti vkládání:

- ▶ Programově
- ▶ Importem ze souboru

V dalších podkapitolách rozebereme podrobněji oba způsoby a jejich výhody a nevýhody.

1.2.1 Vkládání programově

K tomuto účelu vyhovují aplikace vytvořené v jazyce Visual Basic nebo C# v programu Microsoft Visual Studio anebo přímo ve tvůrci maker aplikace MS Outlook. Tyto programovací jazyky přímo spolupracují s MS Outlookem, dovolují nám tak využít jeho plný potenciál. Dokáží vkládat události do defaultního kalendáře, a také vytvářet nové samostatné kalendáře.

Samostatný kalendář pro rozvrh je přehlednější a lépe se s ním pracuje. Události v něm jsou barevně rozlišeny od událostí v jiných kalendářích, rozvrh lze jednoduše celý skrýt nebo zobrazit pouze rozvrh nebo sloučit s událostmi v jiných kalendářích apod.

Nově vzniklý kalendář nemusí být chráněn proti zápisu, to znamená, že ho může uživatel jednoduše měnit přímo v MS Outlooku. To je pro uživatele příjemnější, protože může mazat nebo přesouvat hodiny, ke každé hodině přidávat své poznámky a připomenutí apod. V případě změny rozvrhu v systému VŠB-TUO pak nelze provést aktualizaci automaticky, protože kalendář s rozvrhem může být přejmenován nebo pozměněn natolik, že aplikace nijak nerozpozná, který kalendář se má aktualizovat. Aktualizaci musí uživatel provést odebráním kalendáře s rozvrhem a novým vložením nebo manuální změnou přímo v MS Outlooku.

Pomocí této aplikace lze také data načítat. Data budou získávána z databáze systému rozvrhů VŠB-TUO načítáním a filtrací HTML kódu, což je v jazyce Visual Basic snadno proveditelné. Celý problém se pak dá vyřešit jedním programem či makrem.

Výhody:

- ▶ Velmi dobrá kompatibilita s MS Outlook
- ▶ Možnost vkládání do nového nebo výchozího kalendáře
- ▶ Dokáže data také načítat
- ▶ Celý problém řešen jedním programem
- ▶ Kalendář s rozvrhem lze měnit a upravovat přímo uživatelem

Nevýhody:

- ▶ Složitější naprogramování
- ▶ Funguje pouze pro MS Outlook
- ▶ Aktualizace pouze opětovným vložením nebo manuálně
- ▶ Program běží na lokálním počítači
- ▶ V případě maker je nutná instalace

1.2.2 Import ze souboru

Rozvrh lze také vložit do aplikace MS Outlook pomocí univerzálního výměnného souboru iCalendar. Importem se události obsažené v tomto souboru vloží do samostatného kalendáře, podobně jako v předchozí kapitole.

Existuje několik možností, jakým způsobem lze kalendář importovat.

- ▶ Jako sdílený (z adresáře, z místnosti, z internetu)
- ▶ Jako nesdílený (z lokálního disku)

Nesdílený kalendář může uživatel libovolně měnit, přidávat nebo mazat události, přidávat upozornění, poznámky k jednotlivým událostem apod. Po importu nevzniká žádná vazba mezi souborem a vloženým kalendářem. Obsah souboru je pouze zkopírován do MS Outlooku a na pozdější změny v souboru už MS Outlook nijak nereaguje. Aktualizaci tedy provedeme tak, že aktualizovaný soubor importujeme znovu jako nový kalendář. [5]

Sdílený kalendář je určen pouze pro čtení. Uživatel nemá možnost jakkoliv do kalendáře zasahovat, může pouze celý kalendář smazat. Soubor, ze kterého byl kalendář importován, je uložen na serveru a při jeho změně se automaticky změní i importovaný kalendář přímo v MS Outlooku (pokud je server přístupný). [5]

K tomu, aby šlo soubor typu iCalendar použít, musí se nejprve vytvořit. K vytvoření souboru lze použít aplikaci napsanou v jazyce, který umí pracovat se soubory a s textovými daty. Postačí, aby aplikace načetla soubor ve formátu HTML ze systému VŠB-TUO, vyfiltrovala a zpracovala získané údaje a převedla je do formátu iCalendar. (Více o souborech typu iCalendar se dozvíme v následující kapitole.)

Aplikace nemusí běžet na lokálním počítači, protože nemusí přímo komunikovat MS Outlookem a může být umístěna na server. Pro tento účel lze použít internetové aplikace, kde jako rozhraní pro komunikaci uživatele s aplikací poslouží jakýkoliv internetový prohlížeč. Internetová aplikace se jednodušeji udržuje, jelikož odpadá instalace na lokální počítač.

Pro usnadnění importu existuje funkce webcal. Tato funkce je psaná v jazyce HTML a funguje jako odkaz na soubor formátu iCalendar. Po kliknutí na tento odkaz se automaticky importuje kalendář, obsažený v souboru formátu iCalendar, přímo do aplikace MS Outlook. Tím se celý proces importu uživatelsky zjednoduší.

Výhody:

- ▶ Lze využít pro všechny aplikace s podporou iCalendar
- ▶ Jednodušší provedení
- ▶ Snazší údržba a aktualizace aplikace
- ▶ Možnost automatické aktualizace kalendáře u sdíleného souboru
- ▶ Aplikace neběží na lokálním systému ale na serveru
- ▶ Lze použít i pro aplikace běžící na jiných operačních systémech, než je Windows
- ▶ Možnost volby sdíleného či nesdíleného kalendáře

Nevýhody:

- ▶ Musí být nainstalovaný internetový prohlížeč
- ▶ Rozvrh se nevkládá do defaultního kalendáře

1.3 Rozhodnutí

Po analýze obou způsobů automatického vkládání a zhodnocení všech výhod a nevýhod, jsem se rozhodl pro variantu řešení importem ze souboru iCalendar. Přesvědčila mě hlavně univerzálnost provedení a větší rozsah využití. Také to, že se aplikace nemusí nijak instalovat na počítač uživatele a tím zjednoduší její dostupnost méně zkušeným uživatelům, bylo silným argumentem.

2 Výměnné formáty iCalendar a vCalendar

iCalendar umožňuje uchování a výměnu informací obvykle uložené uvnitř aplikace pro plánování a kalendář, jako např. Personal Information Manager (PIM) nebo MS Outlook. [2]

Formát iCalendar je vhodný pro výměnu různých formátů kalendářů mezi aplikacemi nebo systémy. Je definovaný ve standardu MIME. To umožní tomuto objektu, aby mohl být odeslán několika různými způsoby, jako např. SMTP, HTTP, systémem souborů, desktopovými interaktivními protokoly LAN, Wi-Fi, atd. [2]

iCalendar v sobě zahrnuje objektové metody pro podporu kalendářů a plánovacích operací jako jsou: požadavky, odpovědi, úpravy a rušení schůzek a meetingů a free/busy time data. Tento plánovací protokol je definován ve „iCalendar Transport-independent Interoperability Protocol“ (iTIP). [2]

2.1 Řádky obsahu souboru

Objekt iCalendar se skládá z jednotlivých řádků textu, nazývané „content lines“. Tyto řádky jsou odděleny oddělovačem konce řádku, tzv. sekvencí CLRF. Řádky textu by neměly být delší než 75 oktetů, nepočítaje oddělovače konce řádků. Delší řádky by měly být rozděleny do více řádků pomocí tzv. zalomení řádků. To znamená, že dlouhý řádek může být rozdělen mezi jakýmkoli dvěma znaky vložení sekvence CRLF následovaný znakem mezerníku (space). Znaky sekvence CLRF následované mezerou, nemají jakýkoliv vliv na obsah. [2]

Například

```
DESCRIPTION: Toto je dlouhý popis, který je napsán v jednom řádku  
Může být zapsáno takto
```

```
DESCRIPTION: Toto je dlou  
hý popis, který je napsán  
v jednom řádku
```

Některé vlastnosti a parametry mohou obsahovat více než jednu hodnotu. V takovém případě se hodnoty oddělují čárkou. Pokud vlastnost obsahuje více parametrů, oddělují se středníkem.

Například

```
RDATE;VALUE=DATE:19970304,19970504,19970704,19970904
```

2.2 Struktura objektu iCalendar

Soubor iCalendar obsahuje sbírku informací o kalendáři a časovém rozvrhu. Tyto informace jsou obvykle obsaženy v jednom objektu iCalendar. Avšak v jednom souboru může být obsaženo více objektů iCalendar postupně řazených za sebou.

Tělo objektu iCalendar se skládá z řady po sobě jdoucích vlastností a jednoho nebo více komponentů. Vlastnosti kalendáře jsou atributy, které platí pro kalendář jako celek. Komponenty obsahují skupinu vlastností, které vyjadřují určitou kalendářní oblast. Komponent může například popisovat událost, úkol, záznam v deníku, upozornění, nebo informaci o dostupnosti uživatele.

Vlastnosti, které jsou definovány v tabulce 1, musí být definovány v každém souboru typu iCalendar.

[2]

Tabulka 1: Struktura objektu iCalendar

Vlastnosti	Komentář
BEGIN:VCALENDAR	Označuje začátek objektu iCalendar
PRODID:Rorvrh	Atribut PRODID je identifikační parametr. Měl by být unikátní a musí být v každém objektu iCalendar
VERSION:2.0	Tento atribut slouží k určení verze, ve které je objekt iCalendar specifikován. V tomto projektu je použita verze 2.0
Komponenty (např. VEVENT VTIMEZONE ...)	Každý objekt iCalendar musí obsahovat alespoň jeden komponent.
END:VCALENDAR	Označuje konec objektu iCalendar

V tomto projektu se soubor iCalendar použije k přenosu informací o vyučovacích hodinách v podobě kalendářních událostí. Každá jednotlivá hodina v rozvrhu pedagoga bude představovat jednu událost v kalendáři. Všechny tyto události budou definovány pomocí komponentů **VEVENT**.

K tomu aby časové informace ve všech definovaných událostech byly jednoznačné, musí být definováno také časové pásmo a přechody mezi letním a standartním (zimním) časem. K tomu slouží komponent **VTIMEZONE**.

S použitím obou těchto komponentů lze všechny potřebné informace o vyučovacích hodinách pedagoga v a aktuálním semestru definovat

2.2.1 Komponent **VTIMEZONE**

Při přenosu kalendářních informací především u opakujících se událostí, je nutné také přenést informaci o datu a čase v jednoznačném formátu. Specifikace aktuálních informací o časovém pásmu je toho nedílnou součástí.

Komponent **VTIMEZONE** slouží k definici sady pravidel pro standartní (zimní) a letní čas pro danou oblast v daném časovém intervalu. V tabulce 2 je uveden příklad struktury komponentu a popis jednotlivých vlastností.

Tabulka 2 Struktura komponentu VTIMEZONE

<i>Vlastnosti</i>	<i>Komentář</i>
BEGIN:VTIMEZONE	Označuje začátek komponentu VTIMEZONE
TZID:Central Europe Time	Atribut TZID má unikátní textovou hodnotu, která slouží k identifikaci časového pásma
BEGIN:STANDARD	Označuje začátek pravidla standartního času.
DTSTART:16011028T030000	Hodnota atributu DTSTART určuje začátek platnosti pravidla.
RRULE:FREQ=YEARLY; BYDAY=-1SU;BYMONTH=10	Pravidlo opakování s roční frekvencí poslední neděle v 10-tém měsíci.
TZOFFSETFROM:+0200	Určuje časovou odchylku od UTC ve chvíli před změnou času.
TZOFFSETTO:+0100	Určuje časovou odchylku od UTC po změně času.
END:STANDARD	Označuje konec pravidla standartního času.
BEGIN:DAYLIGHT	Označuje začátek pravidla letního času
Pravidlo změny na letní čas	Podobně jako u změny na standartní čas (mezi BEGIN:STANDARD a END:STANDARD)
END:DAYLIGHT	Označuje konec pravidla letního času
END:VTIMEZONE	Označuje konec komponentu VTIMEZONE

2.2.2 Komponent VEVENT

Tento komponent obsahuje vlastnosti, které definují událost v objektu iCalendar. Každá událost je definována jedním komponentem **VEVENT**. Událostí, které se pravidelně opakují, mohou být definovány jedním komponentem, pomocí pravidla opakování, události které se neopakují pravidelně, se musí definovat zvlášť. [2] V tabulce 3 je uveden příklad struktury komponentu a popis jednotlivých vlastností.

Tabulka 3 Struktura komponentu VEVENT

<i>Vlastnosti</i>	<i>Komentář</i>
BEGIN:VEVENT	Označuje začátek komponentu VEVENT
CLASS:PUBLIC	Definuje míru zabezpečení události. Nabývá hodnot PUBLIC=veřejné PRIVATE=soukromé CONFIDENTAL=důvěrné
DTSTAMP:20130302T133911Z	Definuje čas vytvoření události.
DTSTART;TZID="Central Europe Time":20130302T08000	Definuje čas začátku události v jednoznačném formátu
DTEND;TZID="Central Europe Time":20130302T083000	Definuje čas začátku události v jednoznačném formátu
SUMMARY;LANGUAGE=cs:Předmět	Hodnota tohoto parametru se zobrazí v poli předmět v MS Outlooku. Definuje také jazyk hodnoty.
LOCATION:Místo	Hodnota tohoto parametru se zobrazí v poli místo v MS Outlooku
DESCRIPTION:Popis události.	Hodnota parametru DESCRIPTION se zobrazí v poli popis v MS Outlooku.
END:VEVENT	Označuje konec komponentu VEVENT

2.2.3 Komponent VALARM

Komponent VALARM se vkládá do komponentu VEVENT. Definuje v něm upozornění na danou událost. Definuje čas a způsob provedení upozornění. [2]

V tabulce 4 je uveden příklad struktury komponentu a popis jednotlivých vlastností.

Tabulka 4 Struktura komponentu VALARM.

Vlastnosti	<i>Komentář</i>
BEGIN:VALARM	Označuje začátek komponentu VALARM
TRIGGER:-PT30M	Atribut TRIGGER je slouží k určení časového intervalu od začátku události do momentu upozornění
ACTION:DISPLAY	Tento atribut slouží určuje, jakým způsobem se má provést upozornění. Hodnota DISPLAY zobrazuje pop-up okno.
DESCRIPTION:Reminder	Označuje text pop-up okna.
END: VALARM	Označuje konec komponentu VALARM

3 Získávání dat

K tomu aby se mohl vkládat rozvrh pedagoga do kalendáře v MS Outlooku, je potřeba získat informace o všech vyučovacích hodinách, které pedagog vyučuje v aktuálním semestru

Potřebné informace

- ▶ Název vyučovací hodiny
- ▶ Datum a čas začátku
- ▶ Doba trvání
- ▶ Četnost hodin v semestru
- ▶ Místo konání (učebna)
- ▶ Studijní skupina

Pro tento projekt byla vytvořena webová stránka, jako výstup z rozvrhu pedagogů. Na adrese:

„<http://rozvrh.vsb.cz/Reporting/Spreadsheet?idtype=id&objectclass=staff&template=staffout&identifier=>“, kde za poslední rovnítko se dosadí ID pedagoga, nalezneme veškeré potřebné informace k vytvoření rozvrhu. Výstupem této stránky je tabulka s rozvrhovými informacemi ve formátu HTML.

Tabulka 5 Datový výstup z rozvrhu

Host Key	Description	Period in Week	Duration of Scheduled Activities	Teaching week pattern	Named Availability	Locations	Student Sets
K352052702P/01	Management počítačových sítí	65	6	00100000000000		PORF233	SN2AUT01K
K352050801P/01	Operační systémy a programování	65	6	00001000000000		PORF233	SN1AUT01K
352052702P/01	Management počítačových sítí - předn.	33	2	11111111111111	.	PORF233	SN2AUT01
352052702C/01	Management počítačových sítí - cvič.	35	3	11111111111111	.	PORF233	SN2AUT01
352050801P/01	Operační systémy a programování - předn.	2	2	11111111111111	.	PORF233	SN1AUT01
352050801C/01	Operační systémy a programování - cvič.	4	3	11111111111111	.	PORF233	SN1AUT01

V tabulce 5 je zobrazena ukázka výstupu webové stránky se zdrojovými daty jak jej zobrazuje internetový prohlížeč. Význam jednotlivých sloupců je vysvětlen v tabulce 6.

Tabulka 6 Vysvětlení Tabulky 5

<i>Název</i>	<i>Vysvětlivky</i>
Host Key	Kód předmětu
Description	Název předmětu
Period in Week	Každý den má 14 vyučovacích hodin a školní týden má 6 dní od pondělí do soboty. Číslo v tomto sloupci určuje pořadí hodiny v týdnu kde 1. hodina v pondělí má číslo 1 a poslední hodina v sobotu má číslo 84.
Duration of Scheduled Activities	Délka výuky resp. počet vyučovacích hodin za sebou jdoucích pro jeden předmět.
Teaching week pattern	Každá cifra znázorňuje jeden týden v semestru. První cifra první týden, poslední cifra poslední týden. 1 znamená, že se v daný týden předmět vyučuje, 0 že ne.
Named Availability	[tečka] znamená, že se předmět vyučuje každý týden v semestru, [mezera] znamená, že se předmět vyučuje pouze v některých týdnech v semestru.
Locations	Učebna kde se předmět vyučuje
Student Sets	Studijní skupina

Potřebná data se dají získat tak, že se načte HTML kód datového výstupu a vyfiltrují se přebytečné informace a znaky. Výsledkem jsou data, která se pak dají použít a dále zpracovat.

4 Volba programovacího jazyka a technologie

Řešení bude realizováno pomocí internetové aplikace, proto je nasnadě vybrat programovací jazyk přímo pro tento druh aplikace. Kritéria programovacího jazyku jsou následující.

Musí zvládnout

- ▶ Načíst textový soubor
- ▶ Odstranit z něj nepotřebné informace a přebytečné znaky
- ▶ Vytvořit soubor nový se strukturou iCalendar
- ▶ Vytvořit uživatelské rozhraní v internetovém prohlížeči

K řešení tohoto problému jsem vybral programovací jazyk PHP, protože splňuje všechny požadavky a mám už s tímto jazykem zkušenosti. V minulosti jsem v jazyce PHP vytvářel internetové aplikace, a některé do dnes fungují a jsou využívány.

4.1 XSLT procesor

V jazyce PHP (také v mnoha jiných) existuje tzv. XSLT procesor. Jeho funkcí je transformovat vstupní XML soubor podle pravidel definovaných v tzv. šabloně XSLT na soubor výstupní, např. iCalendar.

Takové řešení by bylo velice jednoduché, a univerzální. Stačilo by vytvořit šablonu XSLT, definující vztah mezi vstupním HTML souborem a výstupním .ics souborem a XSLT procesor by se postaral o zbytek.

Šablona XSLT má však jeden nedostatek. Nepodporuje cykly s podmínkou, které jsou nezbytné pro zpracování vstupního HTML souboru. Proto tento způsob nebyl realizován.

[12]

4.2 Document Object Model

Document object model (DOM) je podporován mnoha jazyky a ani jazyk PHP není výjimkou. Tento objekt slouží k práci s XML dokumenty.

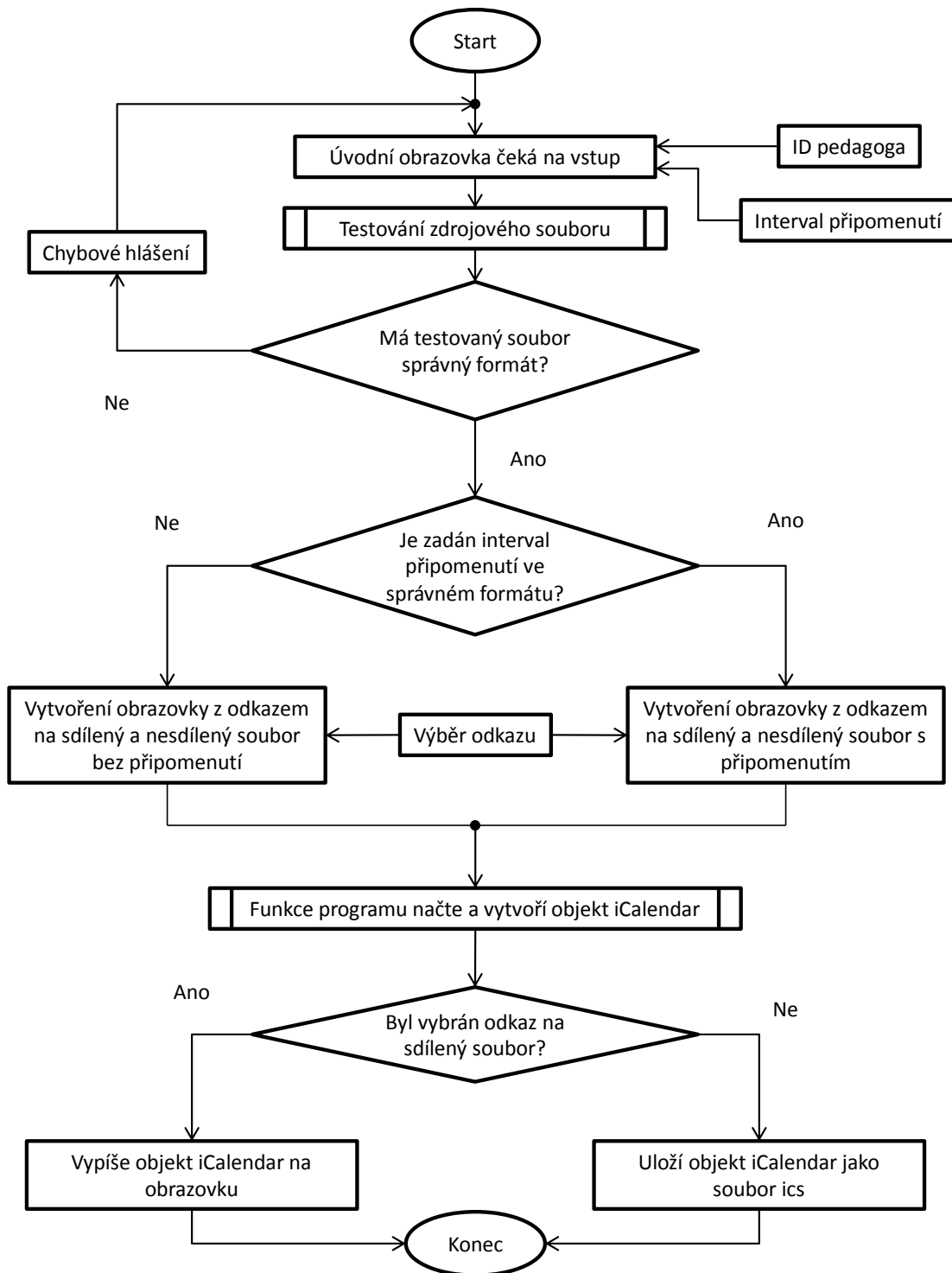
Soubor se zdrojovými daty je ve formátu HTML, což je druh XML, proto na něj lze DOM aplikovat.

Pomocí DOM lze jednoduše soubor HTML načíst a nepotřebné údaje z něj vyfiltrovat.

[8]

5 Návrh programového řešení

Běh programu je znázorněn na Obr. 5.1.



Obr. 5.1 Algoritmus programu

5.1 Obecné konstanty

Obecné konstanty jsou definované v souboru `global_vars.php` a pokud jsou potřeba, jsou načteny pomocí příkazu `require("global_vars.php");`.

Soubor `global_vars.php` je vypsán ve skriptu (5.1).

```
1 $url="http://rozvrh.vsb.cz/Reporting/Spreadsheet?idtype=id&objectclass=
  staff&template=staffout&identifier=";
2 $err1="Zadali jste neplatné ID pedagoga nebo server
  http://rozvrh.vsb.cz/ je dočasně nedostupný.";
3 $err2="Zadejte ID pedagoga";
```

(5.1) Výpis konstant ze souboru `global_vars.php`

Konstanta `$url` obsahuje adresu HTML stránky zdroje dat a konstanty `$err1` a `$err2` obsahují chybové hlášky.

5.2 Funkce `ics`

Funkce `ics` má za úkol načíst rozvrhová data pedagogů z webových stránek k tomu určených, zpracovat je a základě získaných informací vytvořit objekt `iCalendar` ve formě `stringu` a vrátit ho jako výstup. Vstupem funkce je ID pedagoga, které je potřeba k získání odpovídajících informací.

5.2.1 Řešení získávání dat

Potřebné rozvrhové informace jednotlivých pedagogů jsou uvedeny na webových stránkách ve formátu HTML. Data jsou uložena v tabulce. Příklad výstupu a struktury HTML souboru ze kterého se data budou čerpat je uveden v příloze (10.9).

K manipulaci a zpracování souborů typu HTML a XML obecně, existuje v jazyce PHP soubor funkcí `Document Object Model` (zkráceně `DOM`). `DOM` dokáže načíst postupně obsah každé buňky s tabulky se zdrojovými daty a následně s nimi pracovat.

Popis skriptu (5.3)

Řádek 01: Proměnné `$html` se přiřadí nový DOM dokument.

Řádek 02: Provede se načtení HTML souboru z webové adresy uložené v proměnné `$url` zakončené identifikací pedagoga uložené v proměnné `$id`.

Řádek 03: Ze souboru HTML se vyberou postupně všechny elementy s názvem `td`.

Řádek 04: Do proměnných `$j` a `$k` se přiřadí hodnota 1, které se použijí v cyklu.

Řádek 05-14: Zde je definovaný cyklus, který projde všechny hodnoty v buňkách tabulky a postupně je seřadí a uloží do pole `$data`.

V prvním řádku HTML tabulky je hlavička, kterou není potřeba načítat, proto proměnná `$i` v cyklu začíná hodnotou 8. Tím se hlavička vynechá.

Proměnná `$data[$k][\"weeks\"]` obsahuje hodnoty 1 a 0, jednu hodnotu pro jeden týden, seřazeny postupně, první hodnota pro první týden. Hodnota 1 pokud se předmět v týdnu vyskytuje a hodnotu 0 pokud ne. Hodnoty se rozdělí do pole po jedné hodnotě pomocí funkce `str_split`, aby se s ní dalo lépe pracovat.

Další potřebná informace k vytvoření souboru `ics` je datum začátku semestru resp. datum prvního dne výuky v aktuálním semestru. Toto datum je uloženo v patičce tabulky, v elementech s názvem `span` spolu s dalšími údaji. Hodnota atributu `class` v těchto elementech je `footer-0-0-2` pro den, `footer-0-0-4` pro měsíc a `footer-0-0-6` pro rok.

```
1 $load=$html->getElementsByTagName("span");
2 foreach ($load as $load) {
3   if ($load->getAttribute("class")=="footer-0-0-2") $start_array[d]=$load->nodeValue;
4   if ($load->getAttribute("class")=="footer-0-0-4") $start_array[m]=$load->nodeValue;
5   if ($load->getAttribute("class")=="footer-0-0-6") $start_array[y]=$load->nodeValue;
6 }
7 $start_string=$start_array[y]."-".$start_array[m]."-".$start_array[d];
```

(5.4) Část programu, která provádí načítání

Popis skriptu (5.4)

Řádek 1: Načtou se všechny elementy s názvem `span`.

Řádek 2-6: Projdou se všechny načtené elementy a do pole `$start_array` se uloží hodnoty den, měsíc a rok.

Řádek 7: Hodnoty den měsíc a rok se spojí do jednoho stringu ve formátu `d-m-Y` a uloží do proměnné `$start_string`.

5.2.2 Konstrukce objektu iCalendar

Jsou-li všechna potřebná data načtena, může začít vlastní konstrukce objektu iCalendar. Celý objekt iCalendar se bude ukládat jako string do proměnné `$ics`. To umožní pozdější manipulaci s objektem.

Nejprve se provede uložení začátku objektu pomocí skriptu (5.5), ve kterém se definuje

- ▶ Začátek objektu
- ▶ Název objektu
- ▶ Verze objektu
- ▶ Časové pásmo

```
01 $ics.="BEGIN:VCALENDAR\r\n"
02 . "PRODID:Rorvrh ". $id
03 . "\r\nVERSION:2.0\r\n"
04 . "BEGIN:VTIMEZONE\r\n"
05 . "TZID:Central Europe Standard Time\r\n"
06 . "BEGIN:STANDARD\r\n"
07 . "DTSTART:16011028T030000\r\n"
08 . "RRULE:FREQ=YEARLY;BYDAY=-1SU;BYMONTH=10\r\n"
09 . "TZOFFSETFROM:+0200\r\n"
10 . "TZOFFSETTO:+0100\r\n"
11 . "END:STANDARD\r\n"
12 . "BEGIN:DAYLIGHT\r\n"
13 . "DTSTART:16010325T020000\r\n"
14 . "RRULE:FREQ=YEARLY;BYDAY=-1SU;BYMONTH=3\r\n"
15 . "TZOFFSETFROM:+0100\r\n"
16 . "TZOFFSETTO:+0200\r\n"
17 . "END:DAYLIGHT\r\n"
18 . "END:VTIMEZONE\r\n";
```

(5.5) Část programu, která zapisuje začátek objektu iCalendar do stringu

Následně se musí vytvořit komponent `VEVENT` pro každou sadu vyučovacích hodin. To se provede cyklem `foreach`, který projde načtené předměty a dalším vnořeným cyklem `foreach`, který projde všechny týdny od začátku semestru a zjistí, jestli se předmět v daném týdnu vyučuje nebo ne. Třetí cyklus `foreach` vnořený do druhého cyklu zjišťuje, jestli se daný předmět nevyučuje také ve státní svátek. Pokud ano, pro tento den se událost nevytvoří.

```

01 foreach ($data as $data) {
02     $date = new DateTime($start string);
03     $first=true;
04     foreach ($data["weeks"] as $weeks) {
05         if ($weeks=="1") {
06             $break=false;
07             if ($first) {
08                 settype($data["period"], integer);
09                 settype($data["duration"], integer);
10                 $days=floor($data["period"]/14);
11                 $hours=$data["period">%14;
12                 $hours end=$hours+$data["duration"]-1;
13                 $start_int="+". $days." day ". $period[$hours];
14                 $end_int = "+". $days." day ". $period[$hours_end];
15                 $first=false;
16             }
17             $date reset=$date->format($r format);
18             $date->modify($start_int);
19             foreach ($hollydays as $hol {if ($date->format($h format)==$hol) {
    $break=true; break;}}
20             if (!$break) {
21                 $dtstart=$date->format($ics_format);
22                 $date = new DateTime($date_reset);
23                 $date->modify($end_int);
24                 $date->modify($period_int);
25                 $dtend=$date->format($ics_format);
26                 $ics.="BEGIN:VEVENT
27 CLASS:PUBLIC
28 DTSTAMP:".date($ics format,time())."\r\n"
29 ."DTSTART;TZID=\"Central Europe Standard Time\":".$dtstart."\r\n"
30 ."DTEND;TZID=\"Central Europe Standard Time\":".$dtend."\r\n"
31 ."SUMMARY;LANGUAGE=cs: ".$data["description"]."\r\n"
32 ."LOCATION: ".$data["location"]."\r\n"
33 ."DESCRIPTION:Studijní skupina: ".$data["students"]." \n Kód předmětu:
    ".$data["key"]."\r\n"
34 ."END:VEVENT\r\n";
35             }
36             $date = new DateTime($date reset);
37         }
38         $date->modify($week_int);
39     }
40 }

```

(5.6) Část programu, která převádí získaná data do formátu iCalendar a zapisuje je do stringu.

Popis skriptu (5.6)

Řádek 1: Definice cyklu `foreach` pro pole `$data`.

Řádek 2: Převede datum začátku semestru ze stringu na objekt `DateTime` a přiřadí ho proměnné `$date`

Řádek 3: Přiřadí proměnné `$first` hodnotu `true` (nastaví se počáteční hodnota).

Řádek 4: Definice vnořeného cyklu `foreach` pro pole uloženo v `$data["weeks"]`.

Řádek 5: Testuje se, jestli aktuální hodnota z pole `$data["weeks"]` je rovna 1. Pokud ano, předmět se v tom týdnu vyučuje a program pokračuje ve vytváření události definovaný v dalších řádcích, pokud ne program pokračuje řádkem 38.

Řádek 6: Přiřadí proměnné `$break` hodnotu `false` (nastaví se počáteční hodnota).

Řádek 7: Testuje se, jestli se již prováděl výpočet intervalu od začátku týdne do začátku předmětu a do konce předmětu, resp. jestli hodnota proměnné `$first` rovna `true`.

Řádek 8: Změní datový typ proměnné `$data["period"]` na `integer`, aby se s ní dali provádět matematické operace.

Řádek 9: Změní datový typ proměnné `$data["duration"]` na `integer`, aby se s ní dali provádět matematické operace.

Řádek 10: Hodnota proměnné `$data["period"]` se podělí počtem možných hodin v jednom dni tj. 14 a zaokrouhlí se dolů na celé číslo. Výsledkem je číslo dne v týdnu, ve kterém se předmět vyučuje. Hodnota se uloží do proměnné `$days`.

Řádek 11: Zbytek po dělení čísla `$data["period"]` a 14 je číslo hodiny. Hodnota se uloží do proměnné `$hours`.

Řádek 12: Sečtením hodnot proměnných `$hours` a `$data["duration"]` a odečtením jedničky se získá číslo poslední hodiny vyučovaného předmětu. Výsledná hodnota se uloží do proměnné `$hours_end`.

Řádek 13: Do proměnné `$start_int` se uloží interval od začátku týdne do začátku vyučování předmětu ve formě `stringu`, který vznikne kombinací čísla dne, hodin a minut ve správném formátu. Formát hodin a minut je uložen v poli `period` seřazen podle čísla hodin.

Řádek 14: Do proměnné `$end_int` se uloží interval od začátku týdne do začátku poslední hodiny vyučování, podobně jako v předchozím řádku.

Řádek 15: Do proměnné `$first` se uloží hodnota `false`, to znamená, že výpočet intervalů již byl proveden a nebude již probíhat pro aktuální přímět.

Řádek 16: Ukončení podmínky z řádku 07.

Řádek 17: Do proměnné `$date_reset` se uloží datum začátku aktuálního týdne v podobě `stringu` ve formátu definovaného v proměnné `$r_format`.

Řádek 18: K `$date` kde je uložen datum začátku týdne se přičte doba, kdy začíná vyučování předmětu. Výsledkem je přený datum a čas začátku vyučování předmětu.

Řádek 19: Další cyklus `foreach` ověří, jestli datum vyučování předmětu nekoliduje se státním svátkem tak, že porovná data státních svátků uložené v poli `$hollydays`. Pokud nastane shoda, ukončí se cyklus příkazem `break` a do proměnné `$break` se přiřadí hodnota `true`.

Řádek 20: Pokud v den výuky není státní svátek resp. hodnota `$break` je `false`, pokračuje program ve formování komponentu `VEVENT`, pokud ne, pokračuje až řádkem 36.

Řádek 21: Přesný datum a čas začátku vyučovací hodiny, který je uložený v proměnné `$date`, se převede do formátu podporovaného objektem `iCalendar` ve formě `stringu` a uloží do proměnné `$dtstart`.

Řádek 22: Datum uložený v proměnné `$date` se nastaví znovu na začátek týdne.

Řádek 23: K datu začátku týdne, uloženého v proměnné `$date`, se přičte interval do začátku poslední hodiny vyučovaného předmětu.

Řádek 24: K datu a času začátku poslední hodiny vyučovaného předmětu uloženému v proměnné `$date` se přičte doba trvání jedné vyučovací hodiny. Výsledkem je Datum a čas konce vyučovaného předmětu.

Řádek 25: Přesný datum a čas konce vyučovací hodiny, který je uložený v proměnné `$date`, se převede do formátu podporovaného objektem `iCalendar` ve formě `stringu` a uloží do proměnné `$dtend`.

Řádek 26: Pokračování tvorby objektu `iCalendar` vložení začátku komponentu `VEVENT`.

Řádek 27: Definice třídy zabezpečení komponentu.

Řádek 28: Definice data a času vytvoření události. Vloží se aktuální datum a čas.

Řádek 29: Definice data a času začátku události uloženého v proměnné `$dtstart`.

Řádek 30: Definice data a času konce události uloženého v proměnné `$dtend`.

Řádek 31: Vložení názvu předmětu do kolonky „předmět“, uloženého v proměnné `$data["description"]`.

Řádek 32: Vložení názvu učebny do kolonky „místo konání“, uloženého v proměnné `$data["location"]`.

Řádek 33: Vložení označení studijní skupiny a kódu předmětu do kolonky „popis“, uloženého v proměnných `$data["students"]` a `$data["key"]`.

Řádek 34: Vložení konce komponentu `VEVENT`.

Řádek 35: Ukončení podmínky definované na řádku 20.

Řádek 36: Datum uložený v proměnné `$date` se nastaví znovu na začátek týdne.

Řádek 37: Ukončení podmínky definované na řádku 05.

Řádek 38: K proměnné `$date` se přičte přesně jeden týden.

Řádek 39: Ukončení podmínky definované na řádku 04.

Řádek 39: Ukončení podmínky definované na řádku 01.

5.2.3 Konstanty funkce ics

Ve funkci ics jsou také použity konstanty, které jsou definované ve skriptu (5.7). Tyto konstanty se používají pouze ve funkci ics, proto nejsou definovány v souboru `global_vars.php`. Naopak konstanty v souboru `global_vars.php` jsou pro běh funkce nutné a musí být načteny, viz řádek 8 skript (5.7).

```
1 $hollydays=array(1=>"01.05.", "08.05.", "05.07.", "06.07.", "28.09", "28.10.",  
  "17.11.", date("d.m.", strtotime("+1 day", easter date(2013))));  
2 $h_format="d.m.";  
3 $ics_format="Ymd\THis";  
4 $r_format="Y-m-d";  
5 $period=array(1=>"+7 hour +15 minute", "+8 hour", "+9 hour", "+9 hour +45 minute", "+10  
  hour +45 minute", "+11 hour +30 minute", "+12 hour +30 minute", "+13 hour +15 minute",  
  "+14 hour +15 minute", "+15 hour", "+16 hour", "+16 hour +45 minute", "+17 hour +45  
  minute", "+18 hour +30 minute");  
6 $week_int="+1 week";  
7 $period_int="+45 minute";  
8 require("global_vars.php");
```

(5.7)

5.3 Funkce servertest

Tato funkce testuje, jestli soubor HTML s rozvrhovými daty pedagoga má předpokládaný formát.

Vstupem funkce je ID pedagoga a výstupem `true` nebo `false`.

```
1 function servertest($id) {  
2     require("global_vars.php");  
3     $html = new DomDocument;  
4     $html->loadHTMLFile($url.$id);  
5     $load=$html->getElementsByTagName("td");  
6     if ($load->item(0)->nodeValue=="Host Key") return true;  
7     else return false;  
8 }
```

(5.8)

Funkce načte HTML soubor a porovnává první hodnotu v buňce tabulky s předpokládanou hodnotou. Pokud hodnota souhlasí, vrátí `true`, pokud ne `false`.

5.4 Sdílený soubor ics

Sdílený soubor ics je jednoduše soubor uložený na serveru určen, jen pro čtení, ze kterého MS Outlook, nebo jakákoliv jiná aplikace podporující ics soubory, načte objekt `iCalendar` a zobrazí jako kalendář.

Soubor ics nemusí fyzicky existovat, stačí, aby server objekt iCalendar poslal aplikaci jako výstup, ve formě textu. To je řešeno soborem `rozvrh.php`. Obsah tohoto souboru je uveden ve skriptu (5.9).

```
1 require("function.php");
2 header('content-type: text/html; charset=UTF-8');
3 echo ics($_GET["teacherID"]);
```

(5.9) Obsah souboru `rozvrh.php`

5.5 Soubor ics ke stažení

Programové řešení stažení souboru je velmi podobné jako u sdíleného souboru, s tím rozdílem, že do hlavičky vepíšeme skript, který přinutí prohlížeč, aby výstup uložil jako soubor. Tato hlavička je uvedena ve skriptu (5.10) na řádce 06.

V tomto případě je skript rozšířen o ověření správného formátu Souboru HTML se zdrojovými daty.

```
01 <?php
02 require("function.php");
03 require("global_vars.php");
04 header('content-type: text/html; charset=UTF-8');
05 if (servertest($_GET["teacherID"])):
06     header('Content-Disposition: attachment;
07     filename="rozvrh '.$_GET["teacherID"].'.ics"');
08     echo ics($_GET["teacherID"]);
09 else :?>
10 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
11 <"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
12 <html xmlns="http://www.w3.org/1999/xhtml">
13 <head>
14 <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
15 <title>VŠB-TUO schedule exporter</title>
16 </head>
17 <body>
18 <?php echo $err; ?>
19 </body>
20 </html>
21 <?php endif ?>
```

(5.10)

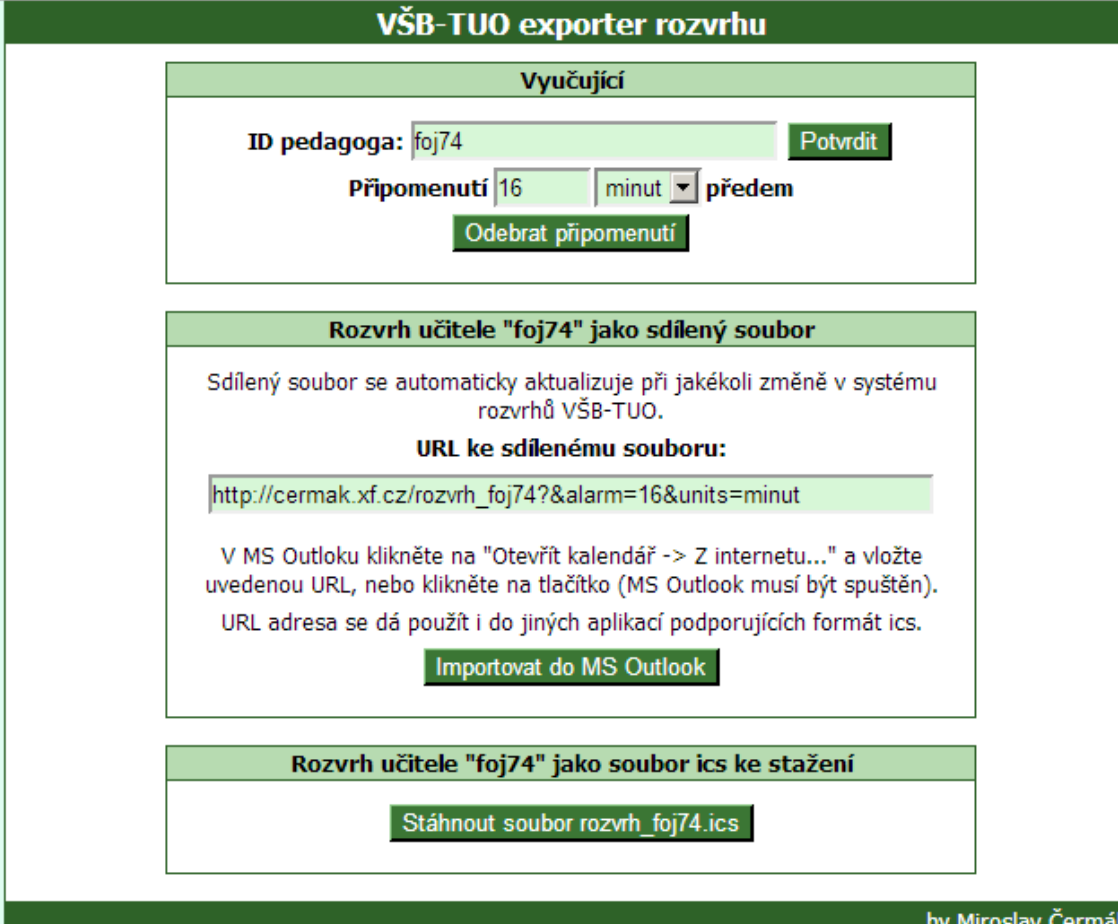
6 Grafický návrh

K vytvoření grafického vzhledu byl použit podobný styl, jaký je přímo na stránkách s rozvrhem na rozvrh.vsb.cz. Návrh je jednoduchý, a uživatel rozvrhu má pocit že stránky rozvrhu vůbec neopustil.



The screenshot shows the title "VŠB-TUO exporter rozvrhu" in a green header. Below it is a section titled "Vyučující" (Teacher). Inside this section, there is a label "ID pedagoga:" followed by a text input field. To the right of the input field is a "Potvrdit" (Confirm) button. Below the input field is a "Přidat připomenutí" (Add reminder) button. At the bottom right of the page, it says "by Miroslav Čermák".

Obr. 6.1 Úvodní obrazovka.



The screenshot shows the same application after the teacher ID "foj74" has been entered. The "ID pedagoga:" field now contains "foj74" and the "Potvrdit" button is visible. Below this, there is a "Připomenutí" (Reminder) section with a numeric input field set to "16", a dropdown menu set to "minut", and the word "předem" (ahead). There is an "Odebrat připomenutí" (Remove reminder) button. Below this is a section titled "Rozvrh učitele 'foj74' jako sdílený soubor" (Teacher's schedule as a shared file). It contains the text: "Sdílený soubor se automaticky aktualizuje při jakékoli změně v systému rozvrhů VŠB-TUO." and "URL ke sdílenému souboru:" followed by a text input field containing the URL "http://cermak.xf.cz/rozvrh_foj74?&alarm=16&units=minut". Below the URL is the text: "V MS Outlooku klikněte na 'Otevřít kalendář -> Z internetu...' a vložte uvedenou URL, nebo klikněte na tlačítko (MS Outlook musí být spuštěn). URL adresa se dá použít i do jiných aplikací podporujících formát ics." and an "Importovat do MS Outlook" (Import to MS Outlook) button. At the bottom is a section titled "Rozvrh učitele 'foj74' jako soubor ics ke stažení" (Teacher's schedule as an ics file for download) with a "Stáhnout soubor rozvrh_foj74.ics" (Download file) button. At the bottom right, it says "by Miroslav Čermák".

Obr. 6.2 Obrazovka po zadání ID pedagoga

7 Realizace a používání

V současné době je kompletně funkční program umístěn na free webovém serveru na adrese: <http://cermak.xf.cz/import.php>

Je plánováno, pokud bude program pracovat správně, že bude umístěn přímo na stránkách rozvrhu. Program tak můžou používat všichni pedagogové na VŠB-TUO.

7.1 Návod k použití

- ▶ Do webového prohlížeče se napíše adresa: <http://cermak.xf.cz/import.php>.
- ▶ Zobrazí se úvodní obrazovka viz. Obr. 6.1.
- ▶ Pomocí tlačítka „přidat upozornění“ a následně zadáním hodnoty lze přidat upozornění před všechny události v rozvrhu.
- ▶ Do prázdného textového pole se napíše ID pedagoga a potvrdí entrem nebo tlačítkem potvrdit.
- ▶ Zobrazí se obrazovka s výběrem možností importu
- ▶ Nesdílený soubor se stáhne po kliknutí na „stáhnout_soubor.ics“
- ▶ Sdílený soubor pro MS Outlook se importuje po stisknutí „importovat do MS Outlook“. MS Outlook musí být spuštěný.
- ▶ Sdílený soubor pro jiné aplikace je na adrese uvedené v textovém poli „URL ke sdílenému souboru.
- ▶ Pro aplikaci připomenutí v MS Outlooku se musí soubor .ics importovat jako nesdílený do defaultního kalendáře pomocí „Soubor – Otevřít – Importovat“ pro verzi 2010 nebo „Soubor – Importovat“ pro verzi 2007.

8 Závěr

Úvodní část bakalářské práce je věnována MS Outlooku, stručnému popisu modulu kalendář a analýze způsobu automatického vkládání rozvrhu do kalendáře. Výsledkem této analýzy bylo zvolení vkládání rozvrhu pomocí univerzálního výměnného souboru iCalendar, z důvodu jeho výhod, především univerzálnosti.

V další části byla rozebrána struktura souboru iCalendar a také popis některých komponentů nezbytných k uchování informací o jednotlivých událostech.

V třetí části byl popsán zdroj rozvrhových dat pedagogů a přístup k němu, seznámení se se způsobem získávání potřebných a filtrací nepotřebných údajů a znaků.

Na konec teoretické části byl zvolen programovací jazyk a technologie, pomocí které se bude vkládání rozvrhu do kalendáře realizovat.

Praktická část byla věnována návrhu a realizaci řešení. Popisuje způsob získávání dat i jejich zpracování a způsob vytváření objektu iCalendar pomocí internetové aplikace napsané v jazyce PHP. Dále popisuje použité funkce, stavbu programu i grafický vzhled webu.

Výsledkem této práce je webová aplikace, která dokáže načíst rozvrhová data pedagoga z výstupu rozvrhového systému VŠB-TUO a vytvořit z nich kalendář ve formátu iCalendar. Uživatel si pak může vybrat, jestli si kalendář uloží do souboru .ics, nebo importuje do MS Outlooku pomocí tlačítka nebo odkazu. Odkaz i soubor .ics může být použit i do jiné aplikace podporující tento formát.

Řešení je nezávislé na operačním systému, kalendář může být importován dokonce i kok do některých mobilních telefonů.

Uživatel si také může zvolit dobu připomenutí před událostmi. Bohužel tato funkce objektu iCalendar není v mnoha aplikacích standardně podporována, např. do aplikace MS Outlook se musí soubor .ics vložit do defaultního kalendáře, aby se připomenutí aplikovala.

Dosud je již plně funkční aplikace umístěna na free PHP serveru, dostupná na adrese: <http://cermak.xf.cz/import.php> kde jí můžou zaměstnanci VŠB-TUO využívat. Dalším krokem bude nainstalování aplikace přímo na školní server a vložení odkazu na stránky, a zajistit tak stálou dostupnost stránek pro všechny.

9 Seznam použité literatury

- [1] Microsoft Outlook. In: *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2001-2013, Stránka byla naposledy editována 10. 3. 2013 ve 02:03 [cit. 2013-05-19]. Dostupné z: <http://cs.wikipedia.org/wiki/Microsoft_Outlook>
- [2] DAWSON, F. & STENERSON, D. 1998 *Internet Calendaring and Scheduling Core Object Specification (iCalendar)*. Copyright (C) The Internet Society (1998). November 1998. [cit. 2013-05-19]. Dostupný z: <http://www.ietf.org/rfc/rfc2445.txt>
- [3] FARANA, R., SMUTNÝ, L., VÍTEČEK, A. 1999. *Zpracování odborných textů z oblasti automatizace a informatiky*. 1. vyd. Ostrava : VŠB-TU Ostrava, 1999. 68 s. ISBN 80-7078-737-6.
- [4] KOSEK, J. 1998 *HTML - tvorba dokonalých WWW stránek - podrobný průvodce*. Grada Publishing 1998. 296 stran. ISBN 80-7169-608-0.
- [5] MICROSOFT CORPORATION 2009A. *Outlook 2007* [online]. ©2009. [cit. 2013-05-19]. Dostupný z: <[http://msdn.microsoft.com/cs-cz/library/bb187387\(en-us\).aspx](http://msdn.microsoft.com/cs-cz/library/bb187387(en-us).aspx)>.
- [6] MICROSOFT CORPORATION 2009B. *LINQ to XML* [online]. ©2009. [cit. 2013-05-19]. Dostupný z: <<http://msdn.microsoft.com/en-us/library/bb387098.aspx>>.
- [7] SLOVAK, K. 2007 *Professional Outlook 2007 Programming*. Wrox Press 2007. 454 p. ISBN 0470049944
- [8] THE PHP GROUP. *Php: Hypertext Processor* [online]. © 2001-2013, Last updated: Sun May 19 06:21:06 2013 PDT. [cit. 2013-05-19]. Dostupné z: <<http://php.net>>
- [9] STACK EXCHANGE INC. *Stack Overflow* [online]. © 2013, rev 2013.5.17.690 [cit. 2013-05-19]. Dostupné z: <<http://stackoverflow.com>>
- [10] JANOVSKÝ D. *Jak psát web: o tvorbě internetových stránek* [online]. [2003], Poslední aktualizace 26. dubna 2013 [cit. 2013-05-19]. ISSN 1801-0458. Dostupné z: <<http://www.jakpsatweb.cz>>
- [11] ZAJÍC, P. PHP. KYSILKA, Pavel. *Linux Software* [online]. 2003-2013 [cit. 2013-05-19]. Dostupné z: <<http://www.linuxsoft.cz/php>>
- [12] REFSNES DATA. *W3Schools Online Web Tutorials* [online]. 1999-2013 [cit. 2013-05-19]. Dostupné z: <<http://www.w3schools.com>>
- [13] AYA S.R.O. *Webzdarma.cz - web, e-mail a databáze zdarma: Diskuzní fora* [online]. 2002 - 2013 [cit. 2013-05-19]. Dostupné z: <<http://www.webzdarma.cz/forum>>
- [14] THE APACHE SOFTWARE FOUNDATION. *The Apache HTTP Server Project* [online]. © 2012 [cit. 2013-05-19]. Dostupné z: <<http://httpd.apache.org>>

10 Přílohy

```

01 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
02 <html xmlns="http://www.w3.org/1999/xhtml">
03 <head>
04 <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
05 <title>VŠB-TUO exporter rozvrhu</title>
06 <link href="style.css" rel="stylesheet" type="text/css" />
07 <script type="text/javascript" src="java.js"></script>
08 </head>
09
10 <body>
11 <?php require("global_vars.php"); ?>
12 <?php require("function.php"); ?>
13 <div class="container" align="center">
14 <h1>VŠB-TUO exporter rozvrhu</h1>
15 <div id="input" class="blok">
16 <h2>Vyučující</h2>
17 <form action="<?php echo $_SERVER["PHP_SELF"]; ?>" method="get" id="form1">
18 <label>ID pedagoga: </label>
19 <input class="input" name="teacherID" type="text" value="<?php echo
    $_GET["teacherID"]; ?>" />
20 <input type="submit" value="Potvrdit" /><br />
21 </form>
22 <input id="add" value="Přidat připomenutí" type="button"
    onclick="javascript:addTB();" />
23 </div>
24 <?php if (isset($_GET["teacherID"]) and $_GET["teacherID"]<>"") : ?>
25 <?php if (servertest($_GET["teacherID"])) : ?>
26 <div class="blok">
27 <h2>Rozvrh učitele "<?php echo $_GET["teacherID"] ?>" jako sdílený soubor</h2>
28 <p>Sdílený soubor se automaticky aktualizuje při jakékoli změně v systému
    rozvrhů VŠB-TUO.</p>
29 <p class="bold">URL ke sdílenému souboru:</p>
30 <form>
31 <input class="output" type="text" value="http://<?php echo
    $_SERVER['SERVER_NAME']; ?>/rozvrh_<?php echo $_GET["teacherID"]."?"; if
    (preg_match("/^[0-9]+$/", $_GET["alarm"]) && ($_GET["units"]=="minut" ||
    $_GET["units"]=="hodin" || $_GET["units"]=="dnů")) echo
    "&alarm=".$_GET["alarm"]."&units=".$_GET["units"]; ?>" readonly="true" />
32 <p>V MS Outlooku klikněte na "Otevřít kalendář -> Z internetu..." a vložte
    uvedenou URL, nebo klikněte na tlačítko (MS Outlook musí být spuštěn).</p>
33 <p>URL adresa se dá použít i do jiných aplikací podporujících formát
    ics.</p>
34 <input type="button" onClick="location.href='webcal://<?php echo
    $_SERVER['SERVER_NAME']; ?>/rozvrh_<?php echo $_GET["teacherID"]."?"; if
    (preg_match("/^[0-9]+$/", $_GET["alarm"]) && ($_GET["units"]=="minut" ||
    $_GET["units"]=="hodin" || $_GET["units"]=="dnů")) echo
    "&alarm=".$_GET["alarm"]."&units=".$_GET["units"]; ?>' value="Importovat do MS Outlook"
    />
35 </form>
36 </div>
37 <div class="blok">
38 <h2>Rozvrh učitele "<?php echo $_GET["teacherID"] ?>" jako soubor ics ke
    stažení</h2>
39 <form>
40 <input type="button" value="Stáhnout soubor rozvrh <?php echo
    $_GET["teacherID"]; ?>.ics" onClick="location.href='ics.php?teacherID=<?php echo
    $_GET["teacherID"]; if (preg_match("/^[0-9]+$/", $_GET["alarm"]) &&
    ($_GET["units"]=="minut" || $_GET["units"]=="hodin" || $_GET["units"]=="dnů")) echo
    "&alarm=".$_GET["alarm"]."&units=".$_GET["units"]; ?>' />
41 </form>
42 </div>
43 <?php else: ?><p class="warning"><?php echo $err1; ?></p>
44 <?php endif; ?>
45 <?php elseif (isset($_GET["teacherID"]) and $_GET["teacherID"]==""): ?><p
    class="warning"><?php echo $err2; ?>
46 <?php endif; ?>
47 <h3>by Miroslav Čermák</h3>
48 </div>
49 <script type="text/javascript">
50 <?php if (preg_match("/^[0-9]+$/", $_GET["alarm"]) && ($_GET["units"]=="minut" ||
    $_GET["units"]=="hodin" || $_GET["units"]=="dnů")) {
51 echo "addTB('".$_GET["alarm"]."', '".$_GET["units"]."');";
52 } ?>
53 </script>
54 </body>
55 </html>

```

(10.1) Výpis ze souboru import.php

```
01 body {
02   background-color:#E0FDF1;
03   font-family:Tahoma, Arial, Helvetica,
    sans-serif;
04   margin:0;
05   padding:0;
06   font-size: small;
07 }
08 form {
09   margin:0px;
10 }
11 input[type="text"], select {
12   background-color: #D7F7D7;
13 }
14 input.input {
15   width:200px;
16 }
17 input.output {
18   width:400px;
19   margin-bottom:10px;
20 }
21 .alarmTB {
22   width:50px;
23 }
24 .alarmSel {
25   border-color: #888 #EEE #EEE #888;
26   border-width: 2px;
27 }
28 input[type="submit"],
    input[type="button"] {
29   background-color: #3B7634;
30   color:white;
31   border-color: #8FCB89 #264F22 #264F22
    #8FCB89;
32   border-width: 1px 2px 2px 1px;
33 }
34 label {
35   font-weight:bold;
36 }
37 div {
38   border: 1px solid #2E6228;
39   background-color: #FFF;
40   margin-right: auto;
41   margin-left: auto;
42 }
43 div.container {
44   width: 630px;
45   margin-top: 0px;
46   margin-bottom: 0px;
47 }
48 div.blok {
49   width: 450px;
50   margin-bottom: 15px;
51   padding-bottom:15px;
52 }
53 h1 {
54   background:#2E6228;
55   color:white;
56   font-size:120%;
57   font-weight:bold;
58   margin-top:0px;
59   padding:2px;
60 }
61 h2 {
62   background:#B7DBB1;
63   margin-top:0px;
64   padding:1px;
65   font-size:100%;
66   font-weight:bold;
67   border-bottom: 1px solid #2E6228;
68 }
69 h3 {
70   background:#2E6228;
71   color:white;
72   font-size:100%;
73   margin-bottom:0px;
74   padding-top:2px;
75   padding-bottom:2px;
76   padding-right:5px;
77   border-top:1px solid #2E6228;
78   font-weight: normal;
79   text-align: right;
80 }
81 p {
82   margin:5px;
83 }
84 p.bold {
85   font-weight:bold;
86 }
87 p.warning {
88   color:#F00;
89   font-weight:bold;
90 }
```

(10.2) Výpis ze souboru style.css

```

01 var intTextBox=0;
02 function addTB(value, sel)
03 {
04   intTextBox = intTextBox + 1;
05   var formID = document.getElementById('form1');
06   var newTB = document.createElement('input');
07   var newList = document.createElement('select');
08   var newLbl = document.createElement('label');
09   var newOpt1 = document.createElement('option');
10   var newOpt2 = document.createElement('option');
11   var newOpt3 = document.createElement('option');
12   var newLbl1 = document.createElement('label');
13   newLbl.innerHTML = "Připomenutí ";
14   newTB.setAttribute('name','alarm');
15   newTB.setAttribute('type','text');
16   newTB.setAttribute('class','alarmTB');
17   newTB.setAttribute('onkeyup','this.value=this.value.replace(/[^\0-9]/g, "")');
18   newList.setAttribute('name','units');
19   newList.setAttribute('id','list');
20   newList.setAttribute('class','alarmSel');
21   if (value!=undefined)
22   {
23     newTB.setAttribute('value',value);
24   }
25   formID.appendChild(newLbl);
26   formID.appendChild(newTB);
27   formID.appendChild(newList);
28   var listID = document.getElementById('list');
29   if (sel == "minut") {newOpt1.setAttribute('selected','selected');}
30   newOpt1.innerHTML = "minut";
31   listID.appendChild(newOpt1);
32   if (sel == "hodin") {newOpt2.setAttribute('selected','selected');}
33   newOpt2.innerHTML = "hodin";
34   listID.appendChild(newOpt2);
35   if (sel == "dnů") {newOpt3.setAttribute('selected','selected');}
36   newOpt3.innerHTML = "dnů";
37   listID.appendChild(newOpt3);
38   newLbl1.innerHTML = "předem";
39   formID.appendChild(newLbl1);
40   formID.appendChild(document.createElement('br'));
41   var addID = document.getElementById('add');
42   addID.setAttribute('value','Odebrat připomenutí');
43   addID.setAttribute('onclick','javascript:removeTB()');
44
45 }
46 function removeTB()
47 {
48   if (intTextBox > 0)
49   {
50     var formID = document.getElementById('form1');
51     formID.removeChild(formID.lastChild);
52     formID.removeChild(formID.lastChild);
53     formID.removeChild(formID.lastChild);
54     formID.removeChild(formID.lastChild);
55     formID.removeChild(formID.lastChild);
56     var addID = document.getElementById('add');
57     addID.setAttribute('value','Přidat připomenutí');
58     addID.setAttribute('onclick','javascript:addTB()');
59     intTextBox = intTextBox-1;
60   }
61 }

```

(10.3) Výpis ze souboru java.js

```

1 <?php
2
3 $url="http://rozvrh.vsb.cz/Reporting/Spreadsheet?idtype=id&objectclass=staff&template=staf
4 fout&identifier=";
5 $err1="Zadali jste neplatné ID pedagoga nebo server http://rozvrh.vsb.cz/ je dočasně
6 nedostupný.";
7 $err2="Zadejte ID pedagoga";
8 ?>

```

(10.4) Výpis ze souboru global_vars.php

```

001 <?php
002 function ics($id) {
003 // Vars
004     $hollydays=array(1=>"01.05.", "08.05.", "05.07.", "06.07.", "28.09", "28.10.",
    "17.11.", date("d.m.", strtotime("+1 day", easter_date(2013))));
005     $h_format="d.m.";
006     $ics format="Ymd\THis";
007     $r format="Y-m-d";
008     $period=array(1=>"+7 hour +15 minute", "+8 hour", "+9 hour", "+9 hour +45 minute",
    "+10 hour +45 minute", "+11 hour +30 minute", "+12 hour +30 minute", "+13 hour +15
    minute", "+14 hour +15 minute", "+15 hour", "+16 hour", "+16 hour +45 minute", "+17
    hour +45 minute", "+18 hour +30 minute");
009     $week int="+1 week";
010     $period int="+45 minute";
011     require("global vars.php");
012 //Program
013     if (!servertest($id)) die($err1);
014     $html = new DomDocument;
015     $html->loadHTMLFile($url.$id);
016     $load=$html->getElementsByTagName("td");
017     $j=1; $k=1;
018     for ($i=8; $i<$load->length-4; $i++) {
019         if ($j==1) {$data[$k]["key"]=$load->item($i)->nodeValue;}
020         elseif ($j==2) {$data[$k]["description"]=$load->item($i)->nodeValue;}
021         elseif ($j==3) {$data[$k]["period"]=$load->item($i)->nodeValue;}
022         elseif ($j==4) {$data[$k]["duration"]=$load->item($i)->nodeValue;}
023         elseif ($j==5) {$data[$k]["weeks"]=str_split($load->item($i)->nodeValue);}
024         elseif ($j==7) {$data[$k]["location"]=$load->item($i)->nodeValue;}
025         elseif ($j==8) {$data[$k]["students"]=$load->item($i)->nodeValue; $j=0;
    $k=$k+1;}
026         $j=$j+1;
027     }
028     $load=$html->getElementsByTagName("span");
029     foreach ($load as $load) {
030         >nodeValue; if ($load->getAttribute("class")=="footer-0-0-2") $start array[d]=$load-
031         >nodeValue; if ($load->getAttribute("class")=="footer-0-0-4") $start array[m]=$load-
032         >nodeValue; if ($load->getAttribute("class")=="footer-0-0-6") $start_array[y]=$load-
033     }
034     $start string=$start array[y]."-".$start array[m]."-".$start array[d];
035     $ics.="BEGIN:VCALENDAR\r\n"
036     ."PRODID:Rorvrh_.$id
037     ."\r\nVERSION:2.0\r\n"
038     ."BEGIN:VTIMEZONE\r\n"
039     ."TZID:Central Europe Standard Time\r\n"
040     ."BEGIN:STANDARD\r\n"
041     ."DTSTART:16011028T030000\r\n"
042     ."RRULE:FREQ=YEARLY;BYDAY=-1SU;BYMONTH=10\r\n"
043     ."TZOFFSETFROM:+0200\r\n"
044     ."TZOFFSETTO:+0100\r\n"
045     ."END:STANDARD\r\n"
046     ."BEGIN:DAYLIGHT\r\n"
047     ."DTSTART:16010325T020000\r\n"
048     ."RRULE:FREQ=YEARLY;BYDAY=-1SU;BYMONTH=3\r\n"
049     ."TZOFFSETFROM:+0100\r\n"
050     ."TZOFFSETTO:+0200\r\n"
051     ."END:DAYLIGHT\r\n"
052     ."END:VTIMEZONE\r\n";
053     foreach ($data as $data) {
054         $date = new DateTime($start string);
055         $first=true;
056         foreach ($data["weeks"] as $weeks) {
057             if ($weeks=="1") {
058                 $break=false;
059                 if ($first) {
060                     settype($data["period"], integer);
061                     settype($data["duration"], integer);
062                     $days=floor($data["period"]/14);
063                     $hours=$data["period"]%14;
064                     $hours_end=$hours+$data["duration"]-1;
065                     $start_int="+.$days." day ".$period[$hours];
066                     $end int = "+.$days." day ".$period[$hours end];
067                     $first=false;
068                 }
069                 $date_reset=$date->format($r_format);

```



```

070         $date->modify($start_int);
071         foreach ($hollydays as $hol) {if ($date->format($h format)==$hol)
    {$break=true; break;}}
072         if (!$break) {
073             $dtstart=$date->format($ics_format);
074             $date = new DateTime($date_reset);
075             $date->modify($end_int);
076             $date->modify($period_int);
077             $dtend=$date->format($ics_format);
078             $ics.="BEGIN:VEVENT
079 CLASS:PUBLIC
080 DTSTAMP:".date($ics_format,time())."\r\n"
081 ."DTSTART;TZID=\"Central Europe Standard Time\":".$dtstart."\r\n"
082 ."DTEND;TZID=\"Central Europe Standard Time\":".$dtend."\r\n"
083 ."SUMMARY;LANGUAGE=cs:".$data["description"]."\r\n"
084 ."LOCATION:".$data["location"]."\r\n"
085 ."DESCRIPTION:Studijní skupina: ".$data["students"]." \\n Kód předmětu:
    ".$data["key"]."\r\n";
086             if (preg_match("/^[0-9]+$/", $GET["alarm"]) &&
    ($GET["units"]=="minut" || $GET["units"]=="hodin" || $GET["units"]=="dnů")) {
087                 if ($GET["units"]=="minut") $alarm="-PT".$GET["alarm"]."M";
088                 elseif ($GET["units"]=="hodin") $alarm="-
    PT".$GET["alarm"]."H";
089                 elseif ($GET["units"]=="dnů") $alarm="-P".$GET["alarm"]."D";
090                 else $alarm="";
091                 $ics.="BEGIN:VALARM".$alarm."\r\n"
092 ."TRIGGER:".$alarm."\r\n"
093 ."ACTION:DISPLAY".$alarm."\r\n"
094 ."DESCRIPTION:Reminder".$alarm."\r\n"
095 ."END:VALARM".$alarm."\r\n";
096             }
097 $ics.="END:VEVENT\r\n";
098         }
099         $date = new DateTime($date_reset);
100     }
101     $date->modify($week_int);
102 }
103 }
104 $ics.="END:VCALENDAR";
105 return $ics;
106 }
107
108 function servertest($id) {
109     require("global_vars.php");
110     $html = new DomDocument;
111     $html->loadHTMLFile($url.$id);
112     $load=$html->getElementsByTagName("td");
113     if ($load->item(0)->nodeValue=="Host Key") return true;
114     else return false;
115 }
116 ?>

```

10.5) Výpis ze souboru function.php

```

1 <?php
2 require("function.php");
3 header('content-type: text/html; charset=UTF-8');
4 echo ics($_GET["teacherID"]);
5 ?>

```

(10.6) Výpis ze souboru rozvrh.php

```

01 <?php
02 require("function.php");
03 require("global_vars.php");
04 header('content-type: text/html; charset=UTF-8');
05 if (servertest($_GET["teacherID"])):
06 header('Content-Disposition: attachment;
    filename="rozvrh '.$_GET["teacherID"].'.ics");
07 echo ics($_GET["teacherID"]);
08
09 else :?>
10 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
11 <html xmlns="http://www.w3.org/1999/xhtml">
12 <head>
13 <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
14 <title>VŠB-TUO schedule exporter</title>
15 </head>
16
17 <body>
18 <?php echo $err; ?>
19 </body>
20 </html>
21 <?php endif ?>

```

(10.7) Výpis ze souboru ics.php

```

1 Options +FollowSymlinks
2 <IfModule mod_rewrite.c>
3 RewriteEngine on
4 RewriteRule rozvrh_(.*)? /rozvrh.php?teacherID=$1 [QSA]
5 </IfModule>

```

10.8.) Výpis ze souboru .htaccess

```

001 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
002 <html xmlns='http://www.w3.org/1999/xhtml'>
003 <head>
004 <meta http-equiv="Content-Type" content="text/html; charset=windows-1250" />
005 <title>staffout</title>
006 <style type="text/css">
007 body{font: 10pt "Times New Roman", serif; }
008 span.scientia-footer { font: bold 150% "Times New Roman", serif; color: rgb(255,0,0);
    }
009 div.scientia-footer-message { text-align: center }
010 table.header-border-args{
011 table.footer-0-args{
012 table.footer-border-args{
013 span.footer-0-0-0{text-align: center;vertical-align: bottom;}
014 span.footer-0-0-1{text-align: center;vertical-align: bottom;}
015 span.footer-0-0-2{text-align: center;vertical-align: bottom;}
016 span.footer-0-0-3{text-align: center;vertical-align: bottom;}
017 span.footer-0-0-4{text-align: center;vertical-align: bottom;}
018 span.footer-0-0-5{text-align: center;vertical-align: bottom;}
019 span.footer-0-0-6{text-align: center;vertical-align: bottom;}
020 table.spreadsheet {}
021 tr.columnTitles {}
022 col.column0 {}
023 col.column1 {text-align: left;vertical-align: middle;}
024 col.column2 {}
025 col.column3 {}
026 col.column4 {}
027 col.column5 {}
028 col.column6 {text-align: center;vertical-align: middle;}
029 col.column7 {text-align: center;vertical-align: middle;}
030
031 table.label-0-args{
032 table.label-border-args{
033 span.label-0-0-0{background-color: rgb(192,192,192); }
034
035 </style>
036 </head>

```

```
037
038 <body>
039 <table class='spreadsheet' cellspacing='0' cellpadding='2%' border='1'>
040   <col class='column0' /><col class='column1' /><col class='column2' /><col
      class='column3' /><col class='column4' /><col class='column5' /><col class='column6'
      /><col class='column7' />
041 <tr class='columnTitles'>
042 <td>Host Key</td>
043 <td>Description</td>
044 <td>Period in Week</td>
045 <td>Duration of Scheduled Activities</td>
046 <td>Teaching week pattern</td>
047 <td>Named Availability</td>
048 <td>Locations</td>
049 <td>Student Sets</td>
050 </tr>
051 <tr>
052 <td>K352052702P/01</td>
053 <td>Management počítačových sítí</td>
054 <td>65</td>
055 <td>6</td>
056 <td>00100000000000000000000000000000000000000000000000000000000000000000</td>
057 <td>&nbsp;</td>
058 <td>PORF233</td>
059 <td>SN2AUT01K</td>
060 </tr>
061 <tr>
062 <td>K352050801P/01</td>
063 <td>Operační systémy a programování</td>
064 <td>65</td>
065 <td>6</td>
066 <td>00001000000000000000000000000000000000000000000000000000000000000000</td>
067 <td>&nbsp;</td>
068 <td>PORF233</td>
069 <td>SN1AUT01K</td>
070 </tr>
071 <tr>
072 <td>352052702P/01</td>
073 <td>Management počítačových sítí - předn.</td>
074 <td>33</td>
075 <td>2</td>
076 <td>11111111111111000000000000000000000000000000000000000000000000000000</td>
077 <td>.</td>
078 <td>PORF233</td>
079 <td>SN2AUT01</td>
080 </tr>
081 <tr>
082 <td>352052702C/01</td>
083 <td>Management počítačových sítí - cvič.</td>
084 <td>35</td>
085 <td>3</td>
086 <td>11111111111111000000000000000000000000000000000000000000000000000000</td>
087 <td>.</td>
088 <td>PORF233</td>
089 <td>SN2AUT01</td>
090 </tr>
091 <tr>
092 <td>352050801P/01</td>
093 <td>Operační systémy a programování - předn.</td>
094 <td>2</td>
095 <td>2</td>
096 <td>11111111111111000000000000000000000000000000000000000000000000000000</td>
097 <td>.</td>
098 <td>PORF233</td>
099 <td>SN1AUT01</td>
100 </tr>
101 <tr>
102 <td>352050801C/01</td>
103 <td>Operační systémy a programování - cvič.</td>
104 <td>4</td>
105 <td>3</td>
106 <td>11111111111111000000000000000000000000000000000000000000000000000000</td>
107 <td>.</td>
108 <td>PORF233</td>
109 <td>SN1AUT01</td>
110 </tr>
111 </table>
```

```
112 <table class='footer-border-args' border='0' cellspacing='0' width='100%'><tr>
113 <td>
114 <table cellspacing='0' border='0' width='100%' class='footer-0-args'>
115 <col align='left' /><col align='center' /><col align='right' />
116 <tr>
117 <td><span class='footer-0-0-0'>VŠB-TU Ostrava, Letní semestr 2012/13</span><span
class='footer-0-0-1'> - Začátek výuky </span><span class='footer-0-0-2'>11</span><span
class='footer-0-0-3'>. </span><span class='footer-0-0-4'>02</span><span class='footer-
0-0-5'>. </span><span class='footer-0-0-6'>2013</span></td><td></td><td></td>
118 </tr>
119 </table>
120 </td>
121 </tr>
122 </table>
123
124 </body>
125 </html>
```

(10.9) Ukázka zdroje dat ve formátu HTML pedagoga foj74

Seznam příloh

Soubory na CD

.htaccess

function.php

global_vars.php

ics.php

import.php

java.js

rozvrh.php

style.css