

Západočeská univerzita v Plzni  
Fakulta aplikovaných věd  
Katedra informatiky a výpočetní techniky

## **Diplomová práce**

# **Konferenční systém**

Plzeň, 2012

Bc. Tomáš Klásek

## PODĚKOVÁNÍ

Děkuji vedoucím této diplomové práce Prof. Ing. Václavu Skalovi, CSc., který je autorem myšlenky vytvoření tohoto systému a Doc. Dr. Ing. Janě Klečkové, které převzala vedení a svými cennými radami mě dovedla k úspěšnému dokončení celé práce.

**ABSTRAKT**

Konferenční systém můžeme chápat buď jako fyzické zařízení (mikrofony, porty,...), které je použito při pořádané konferenci, nebo jako podpůrný software pro konání konferencí. Právě druhou zmiňovanou možností se tato práce zabývá. Před konáním konference je zapotřebí vybrat příspěvky, které budou prezentovány. Nejprve je tedy potřeba, aby autoři nahráli do systému svůj příspěvek a pro zajištění kvality a odbornosti je mu přiřazen recenzent, který provede hodnocení příspěvku a navrhne rozhodnutí, zda příspěvek přijmout, odmítnout, nechat přepracovat apod. Tím jsou, zjednodušeně řečeno, vybrány příspěvky, které budou prezentovány na konferenci. Konferenční systém však může nabídnout i další podpůrná rozšíření pro její konání, jako je např. tvorba harmonogramu. Cílem této práce je navrhnout a implementovat takovýto systém, který bude spravovat příspěvky, recenze a uživatelské role (autor, recenzent, člen programového výboru, předseda programového výboru, administrátor), a umožní tedy vybrat kvalitní příspěvky.

## **ABSTRACT**

...

Whole of the work is written in Czech language only.

## PROHLÁŠENÍ

Prohlašuji, že jsem diplomovou práci vypracoval samostatně a výhradně s použitím citovaných pramenů.

V Plzni dne .....

---

## Obsah

Úvod.....	2
1 Konferenční systémy .....	4
1.1 Systém pro podporu odborných konferencí .....	4
1.2 OCS – Open Conference System .....	5
1.3 ConfTool .....	5
1.4 ConfMaster.....	5
1.5 CyberChair .....	6
1.6 Confious - The Conference Nous.....	6
1.7 Další konferenční systémy .....	6
2 Použitelné technologie a přenositelnost .....	8
2.1 Programovací prostředky .....	8
2.1.1 PHP.....	8
2.1.2 ASP. NET (Active Server Pages).....	9
2.1.3 Python.....	10
2.1.4 Perl – Practical Extraction and Report Language .....	10
2.1.5 Java server pages - JSP.....	10
2.2 Přenositelnost .....	11
2.3 Webové servery.....	12
2.3.1 Apache HTTP server .....	12
2.3.2 Microsoft IIS (Internet Information Server).....	12
2.3.3 Další webové servery .....	12
2.4 Systémy řízení báze dat.....	12
2.4.1 MySQL.....	13
2.4.2 PostgreSQL .....	13
2.4.3 CouchDB .....	14
2.5 Frontend .....	16
2.6 Aplikace potřebné pro vývoj.....	16
3 Realizační část.....	19
3.1 Analýza úlohy .....	19
3.1.1 Úvod do problematiky a analýza požadavků na konferenční systém	19

---

3.1.2	Organizace konference .....	19
3.1.3	Klasifikace uživatelů .....	20
3.1.4	Příspěvek do konference .....	21
3.1.5	Časová osa a termíny.....	24
3.1.6	Recenze .....	24
3.1.7	Import dat .....	24
3.1.8	Export .....	26
3.2	Ukládání dat .....	28
3.3	Klientské rozhraní .....	28
3.4	Systémové požadavky na straně serveru a klienta .....	28
3.5	Programovací jazyk.....	29
3.6	Systém řízení báze dat a DTO objekty.....	32
3.7	Zabezpečení aplikace .....	35
3.8	Konfigurace aplikace .....	36
3.9	Tvorba dokumentu k odeslání do prohlížeče .....	36
3.10	Vzhled aplikace .....	37
3.11	Vlastní aplikace .....	37
3.11.1	Registrace autora .....	37
3.11.2	Vložení příspěvku.....	38
3.11.3	Rozdělení příspěvků .....	38
3.11.4	Recenzování příspěvku.....	38
3.12	Rozbor aplikace z hlediska aplikační vrstvy .....	39
3.12.1	Základ aplikace.....	39
3.12.2	Přidání modulu .....	44
3.12.3	Systém řízení báze dat.....	45
3.12.4	Šablony a vzhled .....	46
3.12.5	Databáze .....	48
3.12.6	Logování událostí.....	48
3.12.7	Emailová notifikace.....	49
3.13	Rychlost.....	50
3.14	Autorizace uživatele .....	51
4	Závěr.....	53

---

LITERATURA A POUŽITÉ ZDROJE .....	55
SEZNAM POUŽITÝCH ZKRATEK A TERMÍNŮ .....	57
SEZNAM TABULEK.....	59
SEZNAM OBRÁZKŮ .....	60
PŘÍLOHY .....	61
A.    Uživatelský manuál.....	61
B.    Instalační manuál .....	61
C.    Programátorský manuál .....	61
D.    Objektově relační mapování .....	62



## Úvod

V dnešní době se pořádá stále více odborných konferencí. To nahrává i růstu poptávky po systémech podpory konferenčních systémů z důvodu celkem velkého ulehčení veškerých přípravných a organizačních procedur. Jelikož se jedná s drtivou většinou o internetové systémy, mají tak přístup k systému všichni aktéři a to odkudkoliv, kde je internet. Díky tomuto médiu je možné vypustit ostatní komunikační prostředky, které bylo nutné používat dříve, tím je myšleno např. posílání příspěvků poštou nebo je zdlouhavě faxovat. S rozvojem internetu se příspěvky začaly posílat emailovou poštou, již to značně ušetřilo komunikační prodlevy a čas. V dnešní době jsou již internetové aplikace tak rozvinuté, že je možné jejich pomocí provozovat i konferenční systém.

Cílem této diplomové práce je vytvořit právě takovouto internetovou aplikaci, která bude fungovat jako podpůrný systém pro pořádání konferencí. Tato diplomová práce navazuje na dvě bakalářské práce studentů fakulty aplikovaných věd, kteří společně vypracovali první verzi konferenčního systému. Prvním ze studentů je Michaela Beranová s bakalářskou prací nazvanou „Systém pro podporu odborných časopisů“ a druhý studentem je Jan Danišík s prací „Systém pro podporu odborných konferencí“. Tyto práce byly však použité pouze jako inspirace funkcionality systému a byla od nich čerpána základní datová struktura, kterou však bylo potřeba přizpůsobit a rozšířit pro potřeby tohoto nového systému.

Požadavky na nový konferenční systém byly od počátku víceméně jasně dané a v průběhu se již jen upřesňovali detaily. Systém má umožnit pořadateli konference snadno a rychle spravovat příspěvky autorů a jejich recenze. Z příspěvků jsou následně vybrány příspěvky pro konferenci. Funkcionalita systému je rozdělena podle rolí. Tyto tvoří autoři, recenzenti, programový výbor a hlavní administrátor. Autoři přidávají své příspěvky do konference, případně je následně editují. Recenzenti hodnotí příspěvky autorů. Recenzentům jsou tyto příspěvky přiděleny členy programového výboru nebo administrátorem konference. Po vyhodnocení je rozhodnuto, zda příspěvek bude či

nebude přijat, anebo jej bude třeba doplnit – opravit. O tom je informován autor prostřednictvím e-mailu.

Administrátorovi konference je umožněno dávkové přidávání uživatelů, přidělování recenzentů k příspěvkům, rozhodnutí o příspěvcích a mazání uživatelů. Dávkové úlohy jsou realizovány prostřednictvím CSV souborů. Rovněž může příspěvky a hodnocení exportovat z databáze do HTML souborů.

# 1 Konferenční systémy

Tato práce není jediná svého druhu, na internetu se nabízí za poplatek nebo volně k použití již hotové aplikace. Velká část funkcionality je podobná, tato práce by ji však měla ještě rozšířit o další možnosti a měla by být plně konkurenceschopná. To je i hlavní náplní práce. Nyní si krátce popíšeme několik ostatních konferenčních systémů.

## 1.1 Systém pro podporu odborných konferencí

Systém pro podporu odborných konferencí [1] je jedním z českých konferenčních systémů. Tento systém nabízí kompletní funkcionalitu od přihlašování autorů až po konečné publikování ve sborníku. V systému fungují uživatelé ve třech rolích a to: autor, recenzent a administrátor. Komunikace probíhá přes zabezpečené připojení a emailové zprávy, které rozesílá aplikace uživatelům s informacemi a upozorněními. Aplikace běží podle základní časové osy:

1. autor přihlásí příspěvek na konferenci (anotace a název)
2. recenzenti učiní prvotní posouzení a rozhodnutí o přijetí či nepřijetí
3. systém informuje autory o výsledku posouzení
4. autoři zašlou plné znění příspěvku
5. recenzenti provedou zhodnocení příspěvku
6. autoři zašlou konečné verze příspěvků
7. systém informuje autory o přijetí konečné verze příspěvku.

Nevýhodou může být malé administrativní členění a běh aplikace je hostován u jejího autora.

---

## 1.2 OCS – Open Conference System

Volně dostupná aplikace OCS [2] pod licencí GNU, lze ji stáhnout a provozovat na vlastním serveru. Umožňuje správu více konferencí a jejich sekcí. Uživatelé mohou vystupovat v 5 rolích: správce, správce sekce, recenzent, autor, čtenář. Aplikace umožňuje i použití rozšíření (pluginů), jako jsou exporty a importy, platby, reporty, témata a další. Je ji možno rovněž lokalizovat a v základu je anglicky. Podporuje také plánování konference včetně budov, místností, událostí či mimořádných událostí. Je možné ji provozovat na mnohých platformách (Windows, Linux, BSD, Solaris, Mac, OS X), napsána je ve skriptovacím jazyce PHP a jako SŘBD je možné použít MySQL či PostgreSQL.

## 1.3 ConfTool

Aplikace ConfTool [3] je poskytována ve dvou provedeních. První z nich je standardní, která je určena pro menší události a pro maximálně 150 účastníků se základní funkcionalitou. Nabízena pod VSIS ConfTool licencí [4]. Druhou verzí je ConfTool Pro, která již podporuje všechny možné role a plnou funkcionalitu. Podporuje osm standardních rolí a pět administrátorských. Standardními jsou autor, spoluautor, moderátor, účastník, člen programového výboru a meta recenzent a administrátorskými rolemi jsou administrátor konference (admin), asistent konference (asistent), předseda programového výboru (předseda), předseda konference, registrátor/recepce/front desk. Nabízí platební moduly, například i pro některé bankovní instituce celého světa a platbu kartami. Systém je možné instalovat na platformách Unix i Windows. Aplikace je psána ve skriptovacím jazyce PHP. Jako SŘBD užívá MySQL. Poskytovatel systému rovněž nabízí možnost hostování na vlastním serveru, za „rozumnou cenu“.

## 1.4 ConfMaster

ConfMaster [5] je poskytován pouze jako služba, čili jej není umožněno stažení a nainstalování na vlastní webový server. Systém má dlouholetou historii sahající až do let 1997, kdy byla vyvinuta aplikace ConfMan, která byla použita v roce 2003 jako

---

základ systému ConfMaster. Aplikace je napsaná v jazyce PHP, lze ji provozovat na webovém serveru Apache se SŘBD MySQL a platformě Linux. Poskytovatel bohužel mnoho podrobnějších informací na stránkách aplikace neuvádí.

### **1.5 CyberChair**

System CyberChair [6] je užíván již od roku 1996, do března 2000 byl aktualizován, pak již jen do října 2002 bylo provedeno několik oprav. Je šířen pod GNU licenci. Autor poskytuje i rozšířenou verzi CyberChairPRO jako službu. V systému vystupuje uživatel ve třech možných rolích (autor, recenzent, člen programového výboru). Poskytuje pouze základní funkcionalitu, kterou je nahrání příspěvku, jeho recenze a hodnocení. Aplikace je vyvinutá v programovacím jazyce Python a je provozována jako webová.

### **1.6 Confious - The Conference Nous**

Confious patří k dalším webovým aplikacím. Jedná se o hostovanou službu na stránkách autora. Rozlišuje 4 typy rolí: člen programového výboru – nastavuje a řídí celý proces konference; senior člen programového výboru – osoby dohlížející na recenzní řízení; člen programového výboru – má za úkol napsat recenze k příspěvkům a poslední „kontaktní osoby“ (CP), které předkládají příspěvky. Jednou z funkcionalit je systém i rozpoznávání konfliktů mezi autory a jejich recenzenty na základě podobnosti emailových adres či spoluautorství. Také umožňuje automatické přidělování příspěvků recenzentům.

### **1.7 Další konferenční systémy**

Konferenčních systémů je na internetu k dispozici velké množství. Liší se především licencí, pod kterou je poskytována; také je můžeme rozdělit na služby (aplikace jsou hostované na webovém serveru nebo je k dispozici stažitelný soubor). Jako velice vhodný by se jevil OCS, co se funkcionality týká. Jelikož zde jde o konkrétní funkcionalitu, tak její základní podoba nabízí mnoho aplikací nebo mnoho

hostovaných služeb. Zástupci ostatních jsou: CORG, OpenConf, ConfMan, HotCRP, EasiChair, EDAS, PCA, Puma, ReviewIt, SIGACT EPC, SIGDA S/W Conf, SIGGRAPH, Online, SIGPLAN S/W, START, WINPE, WitanWeb...

## 2 Použitelné technologie a přenositelnost

Jedním z požadavků na systém je jeho přenositelnost mezi různými operačními systémy na serveru. Systém musí fungovat jak na serverech s operačním systémem od firmy Microsoft, tak i na linuxových.

Internetový systém umožňuje přístup z různých částí světa, a proto musí být zajištěno kódování umožňující správné zobrazení a hlavně vkládání textů i v různých exotických fontech.

Systém lze koncipovat různými technologiemi, které budou mít stejnou základní vlastnost, a to, že budou typu klient-server. Klientem se zde předpokládá úzký klient, kterým bude webový prohlížeč. Na serveru má aplikace sloužit primárně pro dolování a ukládání dat do SRŘBD. V klientovi mohou být data reprezentována třemi základními technologiemi a to: java applet, flash nebo čistě HTML kód. První dvě zmiňované by přenesly část aplikační úrovně až ke klientu, který by rovněž tvořil vrstvu zobrazovací. V této práci se jeví čistý HTML kód jako ideální volba, jak pro snadný vývoj aplikace, tak i pro rychlé přenášení dat a snadné provádění změn.

### 2.1 Programovací prostředky

Webovou aplikaci je možné psát pomocí různých programových prostředků nebo i jejich kombinacemi. Architektura webové aplikace je většinou tvořena třemi vrstvami. Ty tvoří databáze, vlastní logika aplikace a uživatelské rozhraní. Dle zvoleného uživatelského rozhraní a databáze je dobré volit i programovací prostředek celé aplikace.

#### 2.1.1 PHP

Hypertextový preprocesor, který je nejrozšířenější mezi skriptovacími jazyky, používaný na generování a obsluhu webových aplikací. Je vyvinut speciálně pro nasazení na internetu včetně intranetu. Jeho jednoduchá syntaxe vychází z programovacích jazyků C, Perlu a Javy. Je oblíben především pro svou jednoduchost,

---

nezávislost na platformě, možnosti psát jednoduché příkazy přímo do HTML a pro velmi rozsáhlé knihovny funkcí.

### **2.1.1.1 PHP Framework**

Poslední dobou se jedná o čím dál rozšířenější způsob vytváření webových aplikací. Jde o to, že se programátor může starat o potřebnou logiku stránek a nemusí se starat např. o komunikaci mezi programovými vrstvami. Prezentační vrstva je dostatečně oddělena od logické. Potřebného vzhledu je dosahováno pomocí šablon. Komunikaci s databázovou vrstvou zajišťuje speciální rozhraní; zde není problém přechodu mezi různými systémy řízení báze dat (SŘBD). Mezi nejrozšířenější a nejoblíbenější u nás patří Nette, Zend a Ror.

#### ***2.1.1.1.1 ZEND***

Jeden z velmi populárních a celosvětově používaný je Framework. Nabízí velmi mnoho zajímavých rozšíření a možností. Standardně podporuje databáze MySQL, MSSQL, SQLite, Oracle, PostgreSQL, IBM DB2. Mezi zajímavé moduly, které obsahuje, patří např. Zend\_Pdf nebo Zend\_Mail, a tím i podporu pro snadné, přímé generování souborů typu PDF nebo rozesílání emailových zpráv.

#### ***2.1.1.1.2 NETTE***

Tento český a relativně nový Framework získává na oblibě hlavně díky rychlosti a nízké paměťové náročnosti. Rovněž podporuje čistý návrh aplikací, moderní trendy, jakými jsou např. AJAX/AJAJ, SEO a další. Samozřejmostí jsou i ladící nástroje spořicí mnoho drahého času při vývoji a hledání chyb.

### **2.1.2 ASP. NET (Active Server Pages)**

Tato technologie již není vázaná na použití operačního systému Windows, jak tomu bylo dříve. Na rozdíl od předchozí verze ASP se tato nová technologie výrazně



---

liší. Je založena na CLR (Common Language Runtime). Je tedy možné využívat všechny programovací jazyky podporující CLR. Z těchto jazyků jmenujme především nejoblíbenější C#, Visual Basic .NET, ale je možné programování i v Pythonu a Perlu. Výhodou aplikací založené na ASP.NET, je rychlejší zpracování, oproti čistě skriptovacím jazykům díky předkompilování do knihoven a nemusí být parsovány při každém zpracování. Rovněž je zachyceno více chyb již v průběhu vývoje.

### **2.1.3 Python**

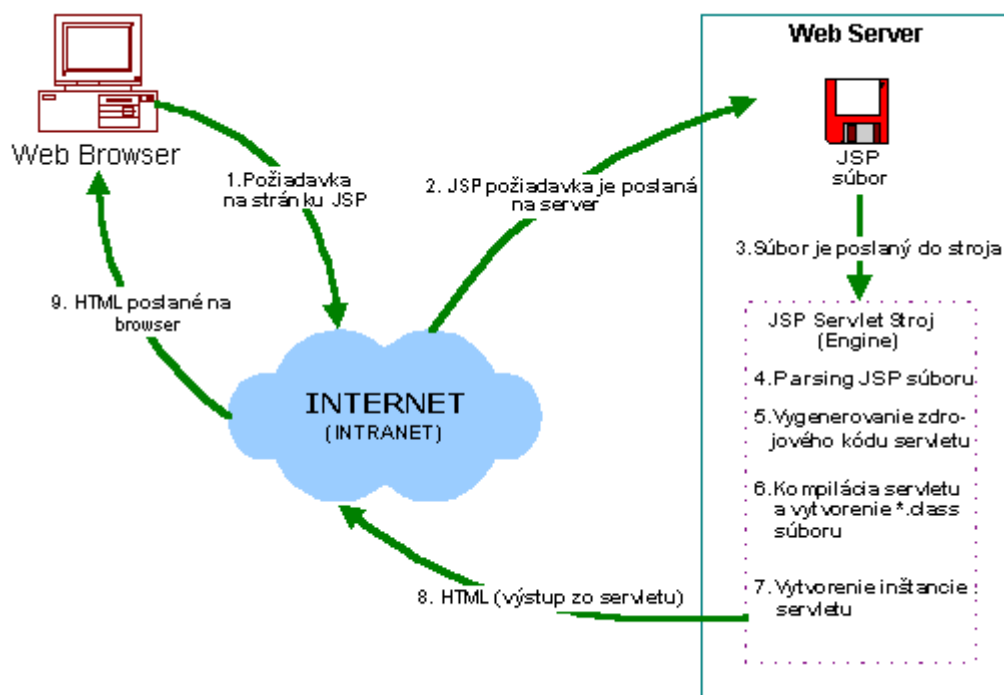
Interpret Pythonu je dostupný pro téměř všechny operační systémy. Možné problémy, na které je třeba si dát pozor, jsou odlišnosti jednotlivých systémů. Tyto odlišnosti se týkají oddělovačů cest souborového systému, konce řádků a kódování češtiny. Vnitřní kódování je UTF-8, takže zde není problém používání v mezinárodním nasazení. Je však potřeba dát si pozor na nastavení vstupního a výstupního kódování. Zvláštností je například OS Windows, kde na konzoli je kódování CP852, ale vstup z klávesnice je CP1250. Tuto nevýhodu lze však odstranit užitím knihovny `os.pat`; pak již rozdíly programátor řešit nemusí.

### **2.1.4 Perl – Practical Extraction and Report Language**

Jedná se o celkem oblíbený skriptovací jazyk. Svou strukturou je podobný PHP nebo C. Snad nejlepší a nejrozšířenější server-side jazyk pro CGI (Common Gateway Interface). Nabízí, kromě spousty jiného, i velmi dobrou podporu pro práci s regulárními výrazy.

### **2.1.5 Java server pages - JSP**

Nástroj pro psaní webových aplikací založený na programovacím jazyce Java, syntaxí podobný ASP nebo PHP. Výhodou je silná typová kontrola. Kód je přeložen do bytecodu (servletu) a servlet je následně vykonán serverem.



Obrázek 2.1 obsluha servletem [7]

Výhody proti ASP jsou především v přenositelnosti, nezáleží tedy na operačním systému serveru.

## 2.2 Přenositelnost

Jedním ze základních požadavků je přenositelnost. Přenositelností je myšlena možnost použití na různých operačních systémech a hardwarových architekturách. Většina výše uvedených technologií přenositelnost umožňuje až na výjimky, kterými je použití technologie .NET. Ta je vázána na užití webového serveru od společnosti Microsoft a tím je velmi i omezena přenositelnost. Z těchto důvodů je nutno vybírat i technologie z hlediska přenositelnosti. Jako nejlepší se zdá volba technologie běžící na serverech apache, které jsou schopné fungovat na serverech s různým operačním systémem včetně operačního systému od firmy Microsoft.

---

## **2.3 Webové servery**

### **2.3.1 Apache HTTP server**

Apache HTTP server [8] projekt je open-source webový server. Vznikl jako daemon napsaný Robem McCoolem v Národním Center for Supercomputing Application na univerzitě v Illinois. Rob McCool následně však vývoj httpd zastavil. Mnoho webmasterů opravovalo chyby a doplňovalo svá rozšíření. Skupina z těchto webmasterů se spojilo za účelem koordinace a vydání záplaty. Osm z nich je možné provozovat na všech operačních systémech pro servery a osobní počítače. Nynější podíl na trhu je 60,10% dle [9].

### **2.3.2 Microsoft IIS (Internet Information Server)**

Tento webový server je vázaný na platformu MS Windows. Má téměř stejně dlouhou historii, jako server Apache. Lze jej rozšiřovat o doplňující moduly, jako je podpora PHP, MOD Rewrite, FastCGI a další. Nyní má podíl na trhu 20,04%.

### **2.3.3 Další webové servery**

Kromě dvou zmiňovaných http serverů s největším podílem na trhu jsou zde i menší: nginx (3. nejrozšířenější 7,57%), Google Web Server, lighttpd a Sun, které jsou však minimálně rozšířené.

## **2.4 Systémy řízení báze dat**

K uchování dat je zapotřebí systému, který nám zaručí rychlý přístup k datům, snadnou správu konzistentnosti dat a samozřejmě kvalitní zabezpečení dat. Na trhu je možno volit jak z placených, tak i bezplatných verzí. Popíšeme zde několik nejrozšířenějších SRBD:

## 2.4.1 MySQL

MySQL [10] [11] je jeden z nejrozšířenějších a asi i nejpopulárnější databázový systém. Je to hlavně díky tomu, že je poskytován ve dvou verzích, kde je jedna z nich volně ke stažení a druhá je komerční. MySQL koupila firma Oracle, avšak nechala vývoj jako open-source. Komerční verze jsou nabízeny za desítky až stovky tisíc. Hlavními přednostmi je rychlá instalace, nízké nebo nulové náklady na vlastnictví, výkonnosti rozsáhlých databází a velmi dobrou podporou a dokumentací. Ve srovnání třeba s dalším volným SŘBD<sup>i</sup> PostgreSQL je rychlejší. Je k němu plná podpora, standardní sémantika a syntaxe dle normy SQL:2008, avšak některé funkce se liší a jsou vylepšeny. Přehled podporovaných standardů nalezneme na [12]. Ale jsou zde však některá vylepšení; jedním příkladem za všechny je dotaz *SELECT SQL\_CALC\_FOUND\_ROWS ...* s použitím klauzule *LIMIT* pro omezení a následně *SELECT @rows = FOUND\_ROWS()*. Je to efektivní způsob, jak získat celkový počet položek, např. pro stránkování, a vrací řádky jako bez klauzule *LIMIT*.

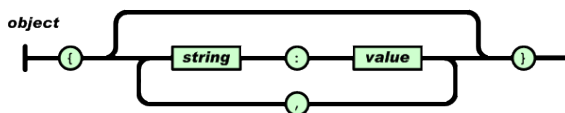
## 2.4.2 PostgreSQL

PostgreSQL [13] [14] je další z řady relačních databázových systému s open-source kódem. Je ho možné provozovat na všech rozšířených operačních systémech a je šířen pod BSD licencí [15]. Jedná se o velice svobodnou licenci a umožňuje jak bezplatné, tak komerční využití včetně modifikací databázového systému. Nejnovější verze (v době psaní práce 9.0.4) podporuje většinu hlavních rysů dle normy SQL:2008 (rovněž označována jako ISO/IEC 9075:2008). Ze 179 povinných vlastností pro plnou shodu, PostgreSQL splňuje alespoň 160. Dále podporuje běh procedur napsaných v několika programovacích jazycích a to v Perlu, Pythonu, jazyce C nebo PL/pqSQL, který vychází z PL/SQL. Existují však varianty nativních rozhraní i pro ostatní programovací jazyky. Hlavními přednostmi je rozšiřitelnost, možnost rozšiřování o nové datové typy, funkce, operátory, procedurální jazyky a agregační funkce. Další výhodou je možnost používat doplňky např. i rozšíření o funkcionalitu MySQL, Oraclu.

### 2.4.3 CouchDB

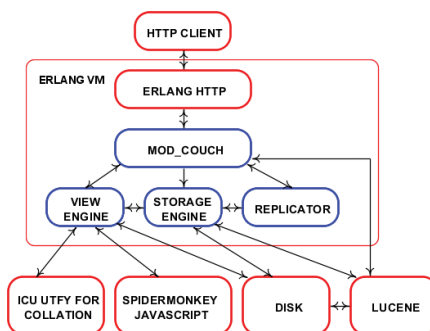
CouchDB [16] je relativně nový systém, který začal být vyvíjen v roce 2005. Je psán programovacím jazykem Erlang [17]. Aplikační rozhraní je zpřístupněno přes protokol HTTP a pohledy jsou stylu MapReduce. Nejedná se, jak by si někdo mohl myslet, o relační ani objektový databázový systém, nýbrž o dokumentově orientovaný databázový systém. Dokumentem je zde jakási flexibilní jednotka, která může být jakkoli složitá a je jen na nás a naší aplikaci, co vše může být v dokumentu. Struktura dokumentu je totožná s JSON objektem viz obrázek „

Obrázek 2.2 JSON objekt“.



Obrázek 2.2 JSON objekt

Na Obrázek 2.3 Struktura CouchDB je zobrazena základní struktura práce s úložištěm a komunikace s okolím. Tento systém také nepoužívá zámky. Při změně dat je vytvořena nová revize dokumentu, a dokud nejsou všechna data zapsána, je ostatním klientům k dispozici starší revize dokumentu. Tato metoda zápisu je přínosem z hlediska času, ale je potřeba o něco více diskového prostoru pro nové revize dokumentů. Dolování dat je velmi jednoduché a poslouží nám např. telnet, kde můžeme přímo psát požadavky na data. CouchDB však nabízí sofistikované AJAX administrační rozhraní pod názvem Futon.



Obrázek 2.3 Struktura CouchDB

Nyní si ukážeme jak komunikovat s tímto databázovým systémem. Dotaz tvoří klasická hlavička protokolu http. Odešleme např. GET požadavek o výpis všech databází. Tyto požadavky můžeme odesílat např. pomocí telnetu po připojení k serveru.

```
GET /_all_dbs http/1.1
```

Server vrátí odpověď:

```
HTTP/1.1 200 OK
Server: CouchDB/0.10.0a799862 (Erlang OTP/R13B)
Date: Mon, 10 Aug 2009 07:02:56 GMT
Content-Type: text/plain;charset=utf-8
Content-Length: 3
Cache-Control: must-revalidate
[test, autori]
```

Výsledkem neboli navrácenými daty je zde JSON pole [test, autori].

Příkaz pro vytvoření databáze není opět nijak složitý:

```
PUT http://localhost:4231/nova_db HTTP/1.1
```

Server vrátí odpověď:

```
HTTP/1.1 200 OK
Server: CouchDB/0.10.0a799862 (Erlang OTP/R13B)
Date: Mon, 10 Aug 2009 07:02:56 GMT
Content-Type: text/plain;charset=utf-8
Content-Length: 3
Cache-Control: must-revalidate
[„ok“:true]
```

Pokud bychom se pokusili vytvořit již vytvořenou databázi, pro příklad opět stejnou jako v předchozím příkladu.

```
PUT http://localhost:4231/nova_db HTTP/1.1
```

Server vrátí odpověď (dále budou příklady již bez úvodních hlaviček):

```
{"error": "file_exists", "reason": "The database could not be
created, the file already exists."}
```

Jelikož tato práce se nezabývá samotnými SŘBD jako takovými, může si čtenář na adrese zdroje [16] prostudovat veškeré možnosti práce s tímto systémem.

## **2.5 Frontend**

Poslední vrstvou třívrstvých aplikací je prezentační část. Jelikož se jedná o webovou aplikaci, budou skripty generovat validní HTML kód s daty, formuláři a navigačními prvky. K oddělení od aplikační vrstvy je zde předpoklad použití šablon a finální vzhled bude určen kaskádovými styly. Toto řešení umožňuje snadnou změnu vzhledu a ovládání finální aplikace. Pro zajištění výsledného zobrazení v různých verzích klientských webových prohlížečů je potřeba striktně dodržovat nové standardy HTML kódu a kaskádových stylů. To bylo ještě před pár lety těžko zajistitelné, jelikož různé verze a výrobci nedodržovali striktně standardy nebo si je implementovali „po svém“ a výsledný vzhled byl odlišný.

Jedním z možných řešení je použití flashových šablon, což by vyžadovalo již větší úpravu aplikační vrstvy a její alespoň částečné přesunutí do flashe. Flash na rozdíl od předchozí možnosti nám dává větší záruku shodnosti zobrazení na různých klientských prohlížečích a operačních systémech.

## **2.6 Aplikace potřebné pro vývoj**

Při vývoji třívrstvé aplikace je možné použití min. tří různých specializovaných aplikací na každou vrstvu.

Na samém počátku vývoje softwaru je dobré použít CASE nástroj pro návrh aplikace, databázové struktury, diagramů tříd, případů užití, komunikace, aktivit, nasazení, balíčků, objektový diagram, diagram časování, přehledu interakcí a spousty dalších dílčích návrhů. Leckteré CASE nástroje generují strukturu databáze přímo v SQL jazyce, základní třídy v konkrétních programovacích jazycích, ORM kód mapování apod.

Aplikace psaná ve skriptovacích jazycích, jako je např. PHP, nepotřebuje pro vývoj žádnou speciální aplikaci a vystačíme si s obyčejným textovým editorem, jakým může být např. poznámkový blok ve MS Windows nebo Vi v Linuxu, či jiné nenáročné textové editory. Jediným omezujícím faktorem je znalost programovacího prostředí,

---

protože textové editory, na rozdíl od IDE, neposkytují našeptávání kódu, zvýraznění syntaxe a podobné užitečné pomůcky. Použití textového editoru však již není možné u čistě skriptovacích jazyků, kde je nutné zdrojový kód nejprve zkompileovat a následně jej vystavit na webový server.

Zobrazovací vrstvu tvoří značkovací HTML jazyk, a proto platí víceméně totéž, jako pro skriptovací jazyky. Opět musíme znát HTML značky, a co který kaskádový styl umí. Zde však méně znalí HTML kódu a kaskádových stylů mohou využít nepřeberné množství aplikací pro tvorbu HTML webů. Téměř všechny podporují jak grafické zobrazení výsledného vzhledu, tak i zobrazení „zdrojového“ (čistého) HTML kódu, či kaskádových stylů. V aplikacích tohoto typu si většinou vytvoříme představovanou webovou stránku a stránka je uložena do třech základních typů souborů. Prvním z nich jsou použité obrázky tvořící pozadí, loga, či jiné složitější grafické prvky (tlačítka a ovládací prvky obecně, oddělovače, apod.). Druhým typem souboru obsahující informace o vzhledu a rozložení jednotlivých částí je CSS, obsahující kaskádové styly. Je však možné uložení kaskádových stylů přímo do HTML kódu webové stránky mezi značky `<style>...</style>`. Posledním, třetím typem souboru je HTML soubor tvořený značkami. Jelikož se předpokládá použití HTML souboru jako šablony, tak neobsahuje klasické texty, které se mají koncovým uživateli zobrazit, ale speciální tvar jakýchsi proměnných, za které je následně dosazován obsah aplikační vrstvou starající se o šablony. Většinou se proměnné dávají do složených závorek, avšak v závislosti na použití knihovny pro plnění šablon mohou obsahovat i řídicí struktury programovacího jazyka nebo jemu podobnou formu, která je zpracována jako kód např. cykly, podmínky.... Taková proměnná může vypadat takto: `{menu}`.

Další vrstvu tvoří úložiště dat. Zde se předpokládá použití webového rozhraní jednotlivých SŘBD nainstalovaná většinou na cílovém serveru. Je to kvůli zabezpečení, databázové systémy umožňují většině uživatelů připojovat se pouze z localhost. Není to však pravidlem a můžeme vytvořit i uživatele, který bude mít práva přístupu i z ostatních počítačů v síti. To je vhodné například při použití speciálních programů umožňujících správu databáze s grafickým rozhraním. Jako minimum poslouží uživateli databáze i příkazový řádek, ze kterého může, v rámci oprávnění, provádět totéž jako



v grafických programech. Zde je opět omezujícím faktorem lepší znalost dotazovacího jazyka SQL.

---

## 3 Realizační část

### 3.1 Analýza úlohy

#### 3.1.1 Úvod do problematiky a analýza požadavků na konferenční systém

Konferencí nazýváme většinou setkání odborníků, kteří diskutují nad určitým tématem, vyměňují si zkušenosti a názory, případně výsledky výzkumu.

Vlastní konference ve skutečnosti začíná již nějakou dobu před jejím vlastním konáním. Musí se připravit program, a proto je nutné vybrat témata, která se budou prezentovat. Takže tím se dostáváme k tomu, že je potřebujeme vybrat, ale zatím není z čeho, proto do konference na dané téma přihlašují své články autoři. Pro zajištění kvality konference je potřeba vybrat zejména ty články, které se věnují danému tématu a mají potřebnou kvalitu. Od toho jsou zde recenzenti, kteří vytvoří posudek na přihlášený článek. Na základě tohoto posudku a hodnocení je článek schválen k projednávání na vlastní konferenci.

#### 3.1.2 Organizace konference

Vědecká konference je většinou organizována buď vědeckým sdružením nebo skupinou výzkumníků se společným zájmem. Velká setkání bývají organizována profesionálními organizátory konferencí.

Konference je ohlášena prostřednictvím rozeslání emailu a nazvána „call for paper“ nebo „call for abstract“ těm, kteří jsou zařazeni v seznamu k danému tématu. Je oznámeno předpokládaným přednášejícím, jak mají předložit svůj abstrakt nebo článek, jaké formalities má splňovat, jakého tématu se má týkat a také mezní termín pro jeho předložení. Předkládání článků poslední dobou bývá čím dál častěji pomocí řízených online systémů.

### **3.1.3 Klasifikace uživatelů**

Uživatele neboli osoby, kterých se týkají konference, můžeme rozdělit do několika málo skupin. Tyto skupiny pak mají v celém životním cyklu jisté role. Jedna osoba však může zastávat i více rolí (např. autor může být i recenzentem jiného článku).

#### **3.1.3.1 Autor**

Autorův životní cyklus začíná registrací do systému. Během registrace zadá kontaktní a další nutné informace o sobě. Úkolem autora je do systému podat k danému tématu a konferenci svůj příspěvek (článek). Povinnými údaji jsou titulek, abstrakt, vybrané téma a příloha v PDF formátu s článkem. Také zadá případné spoluautory a oblast, které se článek týká. Po té čeká na výsledek recenzí a posudků. V poslední fázi nahraje finální (opravenou, či doplněnou) verzi.

#### **3.1.3.2 Recenzent**

Úloha recenzenta začíná po skončení registrace článků a po nahrání článků do systému. Obdrží emailovou notifikaci, že byl vybrán do programové komise, spolu s přihlašovacími údaji. Recenze a hodnocení může upravovat, dokud neskončí termín pro recenze. Články k recenzi a hodnocení jsou mu přiděleny. Kontroluje mimo jiné i správnost zařazení v sekci, které se dané téma týká. Hodnotí článek na základě kritérií, která zadala programová komise a napíše komentář pro autora a pro komisi.

#### **3.1.3.3 Co-chair - člen programového výboru**

Jsou vybráni z recenzentů dříve konaných konferencí.

#### **3.1.3.4 IPC – programový výbor**

Ze skupiny recenzentů se vyberou při dalším konání konference senior recenzenti a z nich členové programového výboru. Má za úkol projednávat hodnocení příspěvků a rozhodovat o jeho přijetí, či nepřijetí.

### **3.1.3.5 Super user – administrátor konference**

Zadává detaily konference: název, data, email, URL hlavní stránky konference, místo a termín konání konference apod. Spravuje workshopy a data jejich fází. Jmenuje členy IPC. Přiděluje článkům recenzenty. Vkládá kritéria a otázky pro formulář recenze. Spouští automatickou emailovou notifikaci recenzentů, jaké články jim byly přiděleny. V jakékoliv fázi může cokoli měnit a přidávat články. Spouští notifikaci autorů o přijetí či zamítnutí článků, tato zpráva autorům obsahuje komentář recenzenta. Importuje seznamy uživatelů, přiděluje články recenzentům, seznam s uživateli ke smazání. Obdobně exportuje uživatele, články dle různých parametrů.

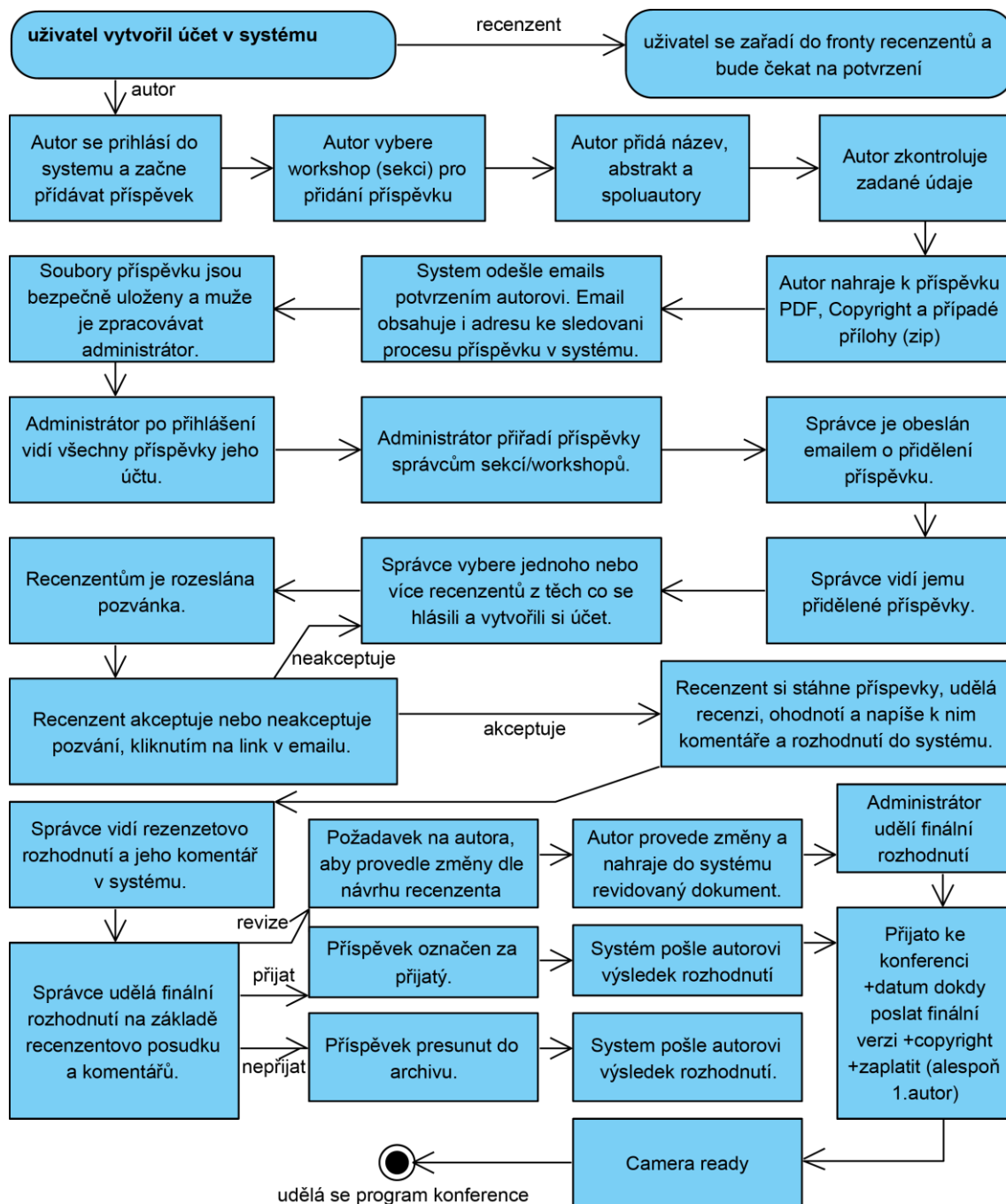
### **3.1.3.6 Administrátor**

Instaluje konferenční systém, nastavuje základní parametry systému a má přístup ke všem datům a funkcím.

## **3.1.4 Příspěvek do konference**

Jak jistě víme, konference se skládá z prezentací odborných článků. Před samotnou konferencí však musí být vybrány příspěvky, dle jejich tématu a úrovně a pak mohou být odprezentovány.

Nyní si představíme životní cyklus takového článku před konáním konference.



Obrázek 3.1 životní cyklus příspěvku

V první fázi autor článku zveřejní v systému titulky, abstrakt, spoluautory a článek. Pokud je vše v pořádku a všechny články jsou úspěšně nahrány, může nastat další fáze. V této jsou články rozděleny mezi recenzenty, kteří jej prostudují, napíší recenzi a článek ohodnotí. Na jeho základě je článek zařazen do několika kategorií a označen jako přijatý či odmítnutý. Tím se dostáváme do další fáze, kdy je potřeba, aby autor svůj článek opravil, nebo doplnil, a spolu s rebuttal<sup>1</sup> a finálním článkem nahrál zpět do systému. Tabulka 3.1 uvádí, kdo, jak a kdy může se článkem nakládat. Hlavní akce jsou poddání, recenze, verdikt a revize.

role \ fáze	1.	2.	3.	4.	5.	6.
autor	registrace + nahrání názvu a abstrakt	nahrání plného znění příspěvku	N/A	N/A	znovu-nahrání	
co-chair workshopu	číst	číst	číst + nahrát recenzi za někoho	číst příspěvky & recenze	čtení & zápis	
co-chair konference	číst	číst	- dtto -	číst příspěvky & recenze	číst & zápis	
IPC	N/A	N/A	N/A	číst příspěvky & recenze	N/A ? – pročíst rebutal OK?	
recenzent	N/A	N/A	napsání posudku	N/A	N/A ? -	
superuser						+ změnit rozhodnutí IPC

Tabulka 3.1 - činnosti aktérů v jednotlivých fázích

System bude příspěvkům přidělovat generovaný identifikátor tvořený velkým písmenem následovaný dvouciferným číslem, které je navíc prvočíslo.

Příklad: A11, A13, ..., A97, B11, B13, ..., Z97

Takto lze vygenerovat maximálně  $25 * 21$  tj. 525 jedinečných identifikátorů příspěvků.

<sup>1</sup>rebuttal - vyvracení tvrzení

### 3.1.5 Časová osa a termíny

Aktéři jsou pro jednotlivé fáze omezeni termíny kdy a co smějí provádět. Je zde celkem 5-6 termínů pro:

- |                     |                       |
|---------------------|-----------------------|
| 1. podání abstraktu | 5. revize             |
| 2. podání příspěvku | 6. případná změna IPC |
| 3. recenze          | rozhodnutí            |
| 4. IPC - rozhodnutí |                       |

Po prošlém termínu mohou příslušnou akci nad příspěvkem provádět pouze superuživatelé či administrátor.

### 3.1.6 Recenze

Recenzenti jsou přiřazeni a nyní mají za úkol projít přidělené příspěvky. Jejich úkolem je nejen sepsat komentáře pro autora a komisi, ale i ohodnotit příspěvek dle předem zadaných otázek a musí se pouze držet předem definované stupnice hodnocení.

### 3.1.7 Import dat

Importována data by měla být ve formátu CSV, což je textový soubor, který obsahuje pouze textová data. Jednotlivé sloupce jsou odděleny většinou středníkem (znak „;“). Každý řádek představuje jeden záznam.

#### 3.1.7.1 Formát pro přidání uživatelů

Importovaný soubor má být textový soubor formátu CSV. Aplikace používá UTF-8 kódování, proto při importu je potřeba si dát pozor na kódování souboru! Řádkovým vstupem bude právě 5 sloupců představující následující:

- |             |           |
|-------------|-----------|
| 1. jméno    | 4. email  |
| 2. příjmení | 5. funkce |
| 3. heslo    |           |

příklad:

```
Josef;Novák;heslo123;josef.novak@email.com;a
```

rozklíčování:

Řádek obsahuje uživatele jménem Josef Novák, emailem „josef.novak@email.com“, přihlašovat se bude heslem „heslo123“ a „a“ označuje autora

zkratky:

- a – autor
- r – recenzent
- d – IPC
- p – chair
- c - admin

### 3.1.7.2 Formát pro smazání uživatelů

Importovaný soubor má být textový soubor formátu CSV. Řádkovým vstupem bude právě 1 sloupec představující následující:

1. email

příklad:

```
josef.novak@email.com
```

rozklíčování:

řádek obsahuje uživatele s identifikátorem a emailem „josef.novak@email.com“, ten je po importu vymazán z aplikace.

### 3.1.7.3 Formát pro přiřazení recenzentů

Importovaný soubor má být textový soubor formátu CSV. Řádkovým vstupem budou 3 a více sloupců představující následující:

1. email – email uživatele/recenzenta pro párování
2. počet přidělovaných článků
3. značka/y jednotlivých článků



příklad:

```
user@emaildomain.com;4;A11;A13;A17;A19
```

rozklíčování:

řádek obsahuje uživatele s emailem „user@emaildomain.com“, počet článků, a následuje/í identifikátory článku/ů

#### 3.1.7.4 Formát s rozhodnutím o člancích

Importovaný soubor má být textový soubor formátu CSV. Řádkovým vstupem budou právě 2 sloupce představující následující:

- |                         |                      |
|-------------------------|----------------------|
| 1. identifikátor článku | 2. verdikt – zkratka |
|-------------------------|----------------------|

příklad:

```
A11;x
```

```
A13;a
```

rozklíčování:

Řádek obsahuje identifikátor článku a zkratku verdiktu. Konkrétně tedy článek s označením A11 není přijat (rejected), zato článek A13 je přijat (accepted)

zkratky:

- |                 |                     |
|-----------------|---------------------|
| • x – rejected  | • s – short         |
| • z – cancelled | • p – poster        |
| • j – journal   | • t – tutorial      |
| • f – full      | • c – communication |

#### 3.1.8 Export

Systém by měl umožňovat exportovat různé části a data ze systému do příslušných adresářů na serveru. *Případné budoucí rozšíření práce je možné udělat export do komprimovaného archivu, který je nabídnut rovnou ke stažení.*

### 3.1.8.1 Export článků

Nejprve se vytvoří adresáře dle verdiktu (např. full, short, poster,...). V těchto adresářích se další adresáře vytvoří podle identifikátoru článků. Do vytvořeného adresáře se nakopírují všechny příložené soubory. Také se zde vytvoří HTML soubor obsahující nejdůležitější data o článku jako je: autor, abstrakt, příložené soubory, hodnocení apod.

Struktura by mohla vypadat následovně:

```
+ root_exportu
    + full          //adresář s fullpapery
      + A11 //identifikátor příspěvku
        - A11.html          // HTML verze
        - A11_xx.pdf       // 1. verze příspěvku
        - A11_full.pdf    // finální verze
        - A11_copy.txt    // copyright
        - A11_rebuttal.pdf
        - ...              // další přílohy
      +A13
        - A13.html
        - A13_xx.pdf
        - ...
    + short
      +A17
        - A17.html
        - A17_xx.pdf
        - ...
```

### 3.1.8.2 Export recenzí

Exportované recenze budou uloženy do souboru v HTML formátu na serveru v adresáři určeném, tj. pro export. Jméno souboru je tvořeno identifikátorem příspěvku a jménem recenzenta.

Příklad: A11\_Novak.html

### 3.1.8.3 Export abstraktů

Exportované příspěvky budou uloženy do souboru v HTML formátu na serveru v adresáři určeném, tj. pro export. Obsahuje výpis všech abstraktů příspěvků.

Příklad: `abstracts.html`

## 3.2 Ukládání dat

K ukládání dat může být použit jakýkoliv databázový systém s velkou podporou přenositelnosti, která je jedním z hlavních požadavků na aplikaci (například PostgreSQL, MySQL). Rozbor použitelných systémů řízení báze dat je rozebrán v kapitole 2.4.

## 3.3 Klientské rozhraní

Uživatelské rozhraní bude generováno dynamicky php-skriptem do tzv. HTML tagů. Vzhled bude nastavován a měněn pomocí kaskádových stylů (CSS). Jako rozhraní na straně klienta by mohl posloužit i „klasický“ program s okny, ale kvůli nezávislosti na systému, systémových požadavcích a také kvůli typu aplikace, se očekává použití webového rozhraní.

Rozhraní generuje zobrazovací část kódu, která načítá z jednotlivých modulů data, která musí být předpřipravena daným modulem k přímému zobrazení. Kvůli modularitě nelze předávat jen „čistá“ data, ale musí být přidány informace o vzhledu. Jednotlivé moduly mají vlastní šablony, které jsou plněny daty z modulů. Modul vrací tedy kompletní HTML kód s doplněnými daty.

## 3.4 Systémové požadavky na straně serveru a klienta

Tato aplikace pro svou činnost vyžaduje internetový webový server s nainstalovanou podporou PHP skriptů a databázový server MySQL a na straně uživatele vyžaduje webový prohlížeč, který implementoval kvalitně základní funkce

---

DOM LEVEL1. Ty jsou zprostředkovány interním skriptovacím jazykem prohlížeče, kterým je JavaScript (nebo jeho Microsoft verze JScript). Nutnost podpory tohoto jazyka v prohlížeči je proto nezbytná. Dále je nutné, pro správné zobrazení, aby prohlížeč podporoval technologii kaskádových stylů (CSS). Aplikace byla vyvíjena pro PHP od verze 5.0.0 a MySQL verze 5.0.16. a jako testovací webové prohlížeče byly použity: Chrome 10, ©Microsoft Internet Explorer 8.0 a FireFox 4.

Pro ukládání veškerých dat byl zvolen databázový systém MySQL, každá datová třída má svou tabulku v databázi a data aplikačních tříd, které využívají data různých datových tříd, jsou uchovány pomocí indexů dané datové třídy potažmo jejich databázových tabulek. Pro aplikaci větších rozměrů, což si myslím, že tato aplikace je, to ani jinak nejde.

### **3.5 Programovací jazyk**

Programovacím jazykem byl zvolen PHP a jeho verze 5, konkrétně byla aplikace vyvíjena na PHP verze „5.2.4-2ubuntu5.24“. Verze od 5.0 již dostatečně podporují objektové programování a vychází z programovacích jazyků Java a C. Tento jazyk byl zvolen hlavně proto, že se jedná o internetovou aplikaci. Je k ní potřeba přistupovat prostřednictvím internetu a lze ji užívat z různých počítačů z celého světa. Rovněž je přenositelný na všechny operační systémy serverů. Na webu poskytovatele je také velká bezplatná dokumentace a mnoho problémů je již prodiskutováno na vývojářských fórech.

Přes vyspělost tohoto programovacího jazyku jsem narazil na jednu celkem významnou chybu tohoto programovacího jazyka. Jedná se o Late Static Bindings (LSB) a je dobře popsán na webové adrese [18]. Problém „pozdní statické vazby“, jak by se dal přeložit, by měl být opraven v PHP až ve verzi 6. Je však možné, že bude zpětně přidán i do verzí 5. Tento BUG<sup>2</sup> je veden pod číslem #52741 – OOP late static bindings bug [19], veden pro verzi PHP 5.3.3. Zkráceně řečeno lze říci, že nestatické

---

<sup>2</sup> BUG – programátorská chyba, či chyba v programu, produkující nesprávný nebo neočekávaný výsledek

odkazování pomocí `$this->` se řídí zákony dědičnosti, kdežto statické volání `trida::` se zákony dědičnosti neřídí.

Stručný příklad:

```
class Rodic {
    protected static function kdo(){
        echo __CLASS__."\n";
    }
    public static function test(){
        return static::kdo();
    }
}
class Ditel extends Rodic {
    protected static function kdo() {
        echo __CLASS__."\n";
    }
}
class Dite2 extends Rodic {}
Rodice::test(); // vypise Rodic
Ditel::test(); // vypise Ditel
Dite2::test(); // vypise Rodic místo Dite2
```

Z těchto důvodů je potřeba užití nestatického obsahu a bylo nutné přepracování tříd, jejich dědičnosti, typů a předávání statických proměnných.

V této práci byla pro základní nakládání s daty vytvořena abstraktní třída. Jejím hlavním úkolem je dolování a ukládání dat do SŘBD. Byla zde zároveň přizpůsobena standardní metoda `__get()`, která je volána při každém volání proměnné třídy. PHP standardně podporuje pouze pár základních datových typů (boolean, integer, float, array, string, object, resource, null). Hlavní přepracování spočívá v možnosti využití více objektového programování bez nutnosti „ručního“ vytváření instancí objektů a proměnné vrací rovnou instanci objektu. Využívá se zde vlastního mapování, kde každá proměnná je tvořena polem. To tedy umožňuje si nadefinovat vlastní datové typy, jako je například typ objekt nebo pole objektů, které nejsou primárně tímto programovacím jazykem podporovány.

Příklad definování datového typu objektového pole:

```
public $vars = array(
    attachments => array(
        type           => 'array',
        className      => 'cyPaperAtt',
        getter         => "getPaperAttachmentsByPaperId",
        args           => array("attachement", "id")
    )
)
```

Všechny proměnné třídy jsou uloženy do jedné proměnné `$vars` typu pole. Kromě této proměnné je ve třídě definována ještě druhá proměnná `$data`, ve které jsou uloženy základní informace o třídě.

```
protected $data = array(databaseName=>"cyPaper", className=>"cyPaper");
```

Ke všem proměnným se však nemusí přistupovat přes pole, ale jsou volány jako klasické proměnné, tedy např. `$this->databaseName`.

Pro mapování dat 1:N byl přidán datový typ pole, do kterého jsou načteny objekty cizích tříd. Nejedná se o přímo mapovaný atribut databázových entit na proměnnou, ale je namapována do pole. Názornější bude ukázka:

```
public $vars = array(
    attachments => array(
        type           => 'array',
        className      => 'cyPaperAtt',
        getter         => "getPaperAttachmentsByPaperId",
        args           => array("attachement", "id")
    ),
    ...
);
```

Jelikož snahou bylo co nejlepší využití objektového programování, byly třídy doplněny o metodu `__toString()` a o tuto metodu byla doplněna i základní abstraktní třída. K tomuto kroku bylo přistoupeno pro lepší a snazší výpis objektových datových typů. V každé třídě si můžeme předvolit, jak se bude prezentovat a jaká základní data o sobě vrátí. Pro ilustraci si uvedeme objekt příspěvku a jeho proměnnou `author`.

```
//vytvoření instance existujícího příspěvku ID rovno 1
$paper = new cyPaper(1);

//výpis autora
echo $paper->author;
```

// výstupem je rovnou text se jménem autora: např. „Josef Novák“

Pro upřesnění: v aplikaci je umožněno mapování i objektových datových typů – viz dále v textu.

### 3.6 Systém řízení báze dat a DTO objekty

Pro tuto práci byl vybrán relační databázový systém MySQL a konkrétně vyvíjen na verzi 5.0.51. Volba byla především na základě přenositelnosti, rychlosti zpracování požadavků a samozřejmě velkou roli zde hraje i to, že je zde možnost využití nekomerční verze. Rovněž je pro tento systém poskytována velmi dobrá dokumentace.

Pro správu MySQL jsem používal webové rozhraní PhpMyAdmin [20].

V rámci této aplikace je rovněž použito objektově relační mapování (ORM). DTO (data transfer object), jsou zde použity pro objektovou reprezentaci dat. DTO jsou mapovány na třídy se stejnými názvy a mapovány takto:

entita	třída	Poznámka
csCmsMenu	csCmsMenu	
cyCoauthor	cyCoauthor	Spoluautoři
cyConference	cyConference	základní data konference
cyConferenceFee	cyConferenceFee	poplatky konference
cyConferenceRegistration	cyConferenceRegistration	registrace na konferenci
cyCountry	cyCountry	číselník států
cyCurrency	cyCurrency	číselník měn
cyDiet	cyDiet	číselník diet – pro registraci na konferenci
cyFile	cyFile	
cyHotel	cyHotel	seznam hotelů – reg. na konf.
cyHotelRoomType	cyHotelRoomType	číselník typů pokoj
cyLanguages	cyLanguages	číselník jazyků
cyLog	cyLog	tabulka pro log systému
cyPaper	cyPaper	příspěvky do konference
cyPaperAtt	cyPaperAtt	přílohy k příspěvkům
cyPaperType	cyPaperType	číselník typů příspěvků
cyPermission	cyPermission	práva uživatele
cyPermissionGroup	cyPermissionGroup	uživatelské skupiny (role)
cyPermissionGroupSites	cyPermissionGroupSites	práva uživ. skupin (rolí)
cyPermissionGroupUsers	cyPermissionGroupUsers	uživatelé ve skupině

<b>cyReview</b>	cyReview	recenze příspěvků
<b>cyReviewAssignment</b>	cyReviewAssignment	přidělení recenzenta příspěvku
<b>cyReviewMain</b>	cyReviewMain	hlavní recenze
<b>cyReviewQuestion</b>	cyReviewQuestion	otázky do recenzí
<b>cyReviewRating</b>	cyReviewRating	číselník hodnocení recenze
<b>cyReviewResult</b>	cyReviewResult	
<b>cySection</b>	cySection	sekce (workshop)
<b>cyText + cyTextDescription</b>	cyText	texty + jazykové mutace
<b>cyUserAccommodation</b>	cyUserAccommodation	info o ubytování uživatele
<b>cyUserPayment</b>	cyUserPayment	platby uživatele
<b>cyUserSkill</b>	cyUserSkill	zkušenosti uživatele v dané sekci (workshopu)
<b>cyUserSkillGrade</b>	cyUserSkillGrade	číselník zkušeností

**Tabulka 3.2 - mapování databázových entit na aplikační třídy**

Výše jsme uvedli specifickou správu proměnných, toho je rovněž aktivně využito i při práci s SŘBD. Jelikož u každé proměnné víme přesně, o jaký datový typ se jedná, můžeme toho využít při ukládání dat a jejich zpětném dolování. Atributy jsou přesně mapovány na proměnné tříd.

Dolování a ukládání dat není potřeba řešit v každé třídě, je řešeno v rodičovské abstraktní třídě pomocí dvou metod. První z nich je metoda `get()`, která se stará o dolování dat z SŘBD. Názvy atributů jsou stejné jako názvy proměnných, tak je lze snadno přiřadit. Této vlastnosti je rovněž využito pro ukládání dat do SŘBD metodou `store()`. Připojení k databázi není inicializováno automaticky s vytvořením instance třídy, ale až v okamžiku, kdy potřebujeme opravdu provést akci uložení či dolování dat.

Získaná data z aplikace nejsou ukládána přímo do proměnných. Je zde využito rovněž upravené standardní metody `__set($valueName, $value)`. Standardní funkce metody je, že přiřadí proměnné její hodnotu a takto vypadá tělo standardní metody:

```
$this->valueName = $value;
```

Ve třídách jsou nadefinovány proměnné jako pole, tak bylo potřeba upravit i metodu `__set(...)`. A hodnota proměnné je ukládá do tohoto pole s indexem `value`. Tělo přepracované metody `__set(...)` má podobu:

```
$this->vars[$valueName][value] = $value;
```



V aplikaci samozřejmě tato metoda obsahuje i kontrolní mechanismy, kontrolující existence proměnných, které zde jen pro znázornění nejsou uváděny.

Nyní si ukážeme, jak je možné při dolování dat z databáze, potažmo při následném volání jednotlivých proměnných mapovat datové typy. Pro příklad asi nejzajímavější datový typ – objekt.

Ve třídě je definována proměnná typu objekt:

```
class cyPaper{
    protected $vars = array(
        author => array(
            type           => 'object',
            className      => 'cyUser'
        )
    );
}
```

Z takto nadefinované proměnné je zcela jasné, že se jedná o datový typ objekt třídy `cyUser`. Po vytvoření nové instance třídy (`new cyPaper()`) se s proměnnými neděje vůbec nic, není tedy vytvářena nová instance třídy `cyUser` a nespotebovává tak systémové prostředky. Pokud však vytváříme novou instanci již existujícího záznamu (`new cyPaper(1)`), nedolují se ihned data z SŘBD, ale dolují se, až když jsou opravdu potřeba. V této chvíli je při ukládání dat z databáze do objektu zavolána i metoda `__set()`, která uloží přečtenou hodnotu na správné místo k proměnné – viz výše. Nyní by byla hodnota standardním postupem uložena do proměnné, a pokud bychom chtěli získat objekt z našeho příkladu, tedy třídy `cyUser`, museli bychom vykonat následující kroky:

```
#vytvoření nové instance existujícího objektu cyPaper
$paper = new cyPaper(1);

#v této proměnné by byla hodnota vydolovaná z databáze a uložená v ní
$paper->id_author;

#za předpokladu nepoužívání gettrů pro přístup přímo k proměnným,
#bychom tímto kódem získali instanci objektu cyUser
$objekt_uzivatel = new cyUser($paper->id_author);
```

Tuto část kódu bychom tedy museli psát pokaždé, pokud bychom chtěli zjistit, jakého autora vlastně daný příspěvek má.

Po úpravách metod `__set()` a `__get()` a využití nově mapovaných proměnných lze získat instanci autora (třída `cyUser`) mnohem elegantnějším způsobem:

```
#vytvoření nové instance existujícího objektu cyPaper - stejné
$paper = new cyPaper(1);

$objekt_uzivatel = $paper->author;

#nebo pokud je použit getter:
$objekt_uzivatel = $paper->getValue("author");

#můžeme rovněž využít toho, že upravená metoda __get() pro datový
objekt vrací rovnou objekt a vypsát třeba rovnou email autora
echo $paper->author->email; // výstup: josef.novak@email.com

#další možností je nechat pracovat přepsanou metodu třídy cyUser
__toString

echo $paper->author; //výstup např.: Josef Novák
```

Výstup přenechaný na metodě `__toString` je možné upravit v každé třídě, nebo lze využít základní přepsání rodičovské abstraktní třídy, která místo standardního `ID` objektu, vypíše naše `ID` v systému a jméno třídy.

### 3.7 Zabezpečení aplikace

Při prvním přístupu do aplikace je vytvořeno jedinečné `Session ID`, ke kterému je po odeslání a ověření uživatelských dat a práv, je danému `ID` přidělen uživatel a nastavena jistá životnost. Dokud uživatel s aplikací pracuje, životnost `ID` se posouvá. Pokud je uživatel nečinný, tak po určitém čase platnost tohoto `ID` vyprší a uživatel je dotázán na zadání uživatelského jména a hesla. `Session ID` je na straně serveru uchovávána v databázi a na straně klienta je většinou (v závislosti na nastavení webového prohlížeče) uložena v cookie nebo předávána v parametru `URL` adresy.

Spolu se `Session ID` se nese i informace o uživateli a jeho právech, která jsou ověřována při vstupu na konkrétní stránku, případně lze přístup zamítnout.

### 3.8 Konfigurace aplikace

Pro celou aplikaci jsou 2 druhy parametrů a konstant:

- prvními jsou pevné a neměnné, jako je nastavení cest (path) pro různé části, jako jsou šablony, třídy, ..., parametry databáze – server, port, uživatelské údaje pro připojení, tj. jméno a heslo a mnoho dalších parametrů. Tyto parametry jsou uloženy v konfiguračním souboru, který je umístěn mimo adresářovou strukturu, do které se dá vstoupit přes webové rozhraní. Tím je částečně zajištěna bezpečnost toho souboru, která je pro tento soubor nutná kvůli obsahu citlivých údajů – jmen a hesel
- dalšími parametry jsou ty, které je možné měnit v administrační části a jsou uloženy v databázi

### 3.9 Tvorba dokumentu k odeslání do prohlížeče

Celý dokument, který se následně odesílá do prohlížeče, je sestaven z několika částí, které se mění dle načteného modulu. Jako základní navigační prvek je menu, které je generováno automaticky dle položek v menu a dle aktuálního zvoleného bodu v menu se vytvoří nabídka menu jen k danému modulu či části aplikace.

Z důvodu zamezení zbytečného načítání všech knihoven je vytvořena funkce `__autoload($jméno_třídy)`, která je podporována od verze PHP5. Umožňuje, jak už sám její název napovídá, automatické načtení potřebných tříd. Konkrétně: pokud je volán `$objekt = new Trida();` je zavolána funkce `__autoload(Trida)` a dojde k načtení třídy, tělo této funkce je samozřejmě upraveno tak, aby byl soubor načítán ze správného adresáře.

Volba modulu, který má být načten, je dán parametry v URL adrese, ze které je extrahován a dojde k automatickému zpracování. Zde se nachází dvě možnosti jak načíst, potažmo odeslat, data z modulu. První z nich je ta, že se vytvoří celá stránka s HTML kódem a je následně odeslána. Tato možnost nastává, pokud je pod číslem menu uveden jako typ stránky „body“. Další možností je, že hodnota tohoto atributu je

„code“ a zpracování probíhá obdobně jako v předchozím případě s tím rozdílem, že není vytvořena celá HTML stránka, ale je odeslána pouze část, která je načtena dle parametrů z daného modulu. Tato alternativa je využívána především v JavaScript funkcích, kdy je pomocí tohoto skriptu načítána pouze část stránky nebo jen pro získání „čistých“ dat bez formátování a následné zpracování JavaScript.

### 3.10 Vzhled aplikace

Aplikace je vyvíjena s co největší snahou o zachování třívrstvé architektury. Proto je v aplikaci využívána tvorba dokumentu pomocí šablon, které obsahují jen HTML kód a jsou do něj dosazována pouze data. Kód dokumentu je validní dle norem W3C, což by mělo zajišťovat zobrazení ve všech prohlížečích s minimálními odchylkami.

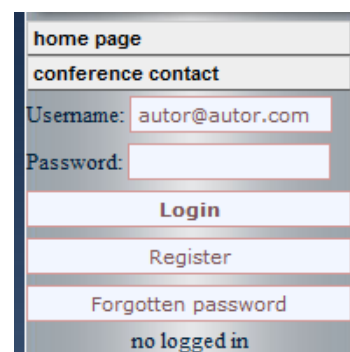
Dále je vzhled upravován pomocí *kaskádových stylů* (CSS), kterými je možné určovat kompletní vzhled výsledného zobrazení a rozložení jednotlivých částí. Kaskádovými styly lze nastavit i různý vzhled pro různá zobrazovací media, jako je tiskárna, monitor či projektor, aniž bychom museli měnit HTML kód.

Celý vzhled je tedy možno změnit pouhou záměnou souboru CSS, obsahující informace o pozici a vzhledu všech prvků stránky. Díky tomuto systému je možné vytvořit snadno a poměrně rychle několik verzí vzhledu. Bez velké námahy je tedy možné, aby si mohli uživatelé tento vzhled sami změnit nebo tuto možnost přidělit administrátorovi, který by pak mohl měnit šablony pro jednotlivé konference.

### 3.11 Vlastní aplikace

#### 3.11.1 Registrace autora

Pro pořádání konference je alfou omegou zveřejnění kvalitních článků. Přihlásit článek může jeho autor až po zaregistrování do vlastní aplikace.



Obrázek 5 - základní menu

---

Slouží k tomu jednoduchý formulář, ke kterému se dostane přes základní menu. Viz Obrázek 5 - základní menu.

Po vyplnění a odeslání registračního formuláře (viz Uživatelský manuál – Obrázek 3) přijde uživateli na zadanou emailovou adresu odkaz, kterým je potřeba potvrdit email a aktivovat uživatelský účet v aplikaci.

### **3.11.2 Vložení příspěvku**

Vkládat nové příspěvky může přidávat uživatel, který je členem skupiny „Autoři“, pomocí tlačítka „přidat příspěvek“ (viz Uživatelský manuál – Obrázek 5). Zobrazí se formulář (viz Uživatelský manuál – Obrázek 5) kde je možné vybrat „workshop“, zadat titulek a napsat abstrakt příspěvku. Pokud je autorů více, je možné je přidat k příspěvku. Dále nahraje kompletní příspěvek konference ve formě PDF souboru, copyright a případné přílohy. Aplikace odešle autorem potvrzující email s odkazem k příspěvku.

### **3.11.3 Rozdělení příspěvků**

Po ukončení etapy vkládání příspěvků administrátor konference je rozdělí mezi správce jednotlivých sekcí / workshopu. Správci obdrží email o jim přiděleném příspěvku a jejich úkolem je nyní přidělit je jednomu nebo více recenzentům. Recenzentům je rovněž poslána pozvánka a mají na výběr ji přijmout či odmítnout pomocí odkazů v emailu. V případě odmítnutí musí správce workshopu přidělit nového recenzenta.

### **3.11.4 Recenzování příspěvku**

Tuto činnost má na starosti přidělený recenzent. Příspěvky si musí stáhnout z webové aplikace. Následně jim udělí hodnocení, napíše recenzi a případný komentář. Do systému uloží rozhodnutí, co s daným příspěvkem dál, zda je přijat či nepřijat.

## 3.12 Rozbor aplikace z hlediska aplikační vrstvy

Pro účely této aplikace byl vytvořen vlastní framework, který umožňuje snazší a rychlejší programování a tím soustředění se na vlastní aplikační problémy. Největší výhodou je implementování objektově-relačního mapování (ORM) proměnných. Díky tomu lze snadno přidávat a spravovat nové proměnné. Implementované ORM rovněž umožňuje mapování cizích klíčů pro vazby 1:1 na proměnné třídy a rovnou jim přiřadí instanci patřičného objektu. Tato funkcionality velmi usnadní práci s proměnnými, ve kterých by bylo standardně uloženo „pouze“ ID dotyčného objektu. Je samozřejmě také implementována mapování i vazeb 1:N a to do proměnné jako pole objektů.

### 3.12.1 Základ aplikace

Jádro aplikace tvoří adresář `core/phplib`, kde jsou nejnütnější knihovny pro aplikaci. Dále je nutný konfigurační soubor (`config/config.inc.php`). Práci se SŘBD umožňuje a spravuje třída `cyDb`, nacházející se v adresáři `core/classes`.

#### 3.12.1.1 Abstraktní třída `abClass`

Od této třídy (`abClass`) se dědí (`extends`) všechny třídy modulu. Implementuje správu proměnných, dolování a ukládání dat do SŘBD. Rovněž implementuje základní metody, které je možno v potomcích přepsat pro základní nakládání s daty. Význačné metody jsou například:

- `get()` – získá dat z SŘBD
- `store()` – uložení dat do SŘBD
- `getAll()` – pro získá všech dat třídy,
- `getSelect()` – navrácí HTML komponentu `SELECT` naplněnou daty
- `__construct()` - defaultní konstruktor,
- `__toString()` – defaultní výpis o třídě,
- `getValue()` – vrací hodnotu proměnné
- `setValue()` – vložení hodnoty do proměnné
- ...

vybrané privátní metody:

`dbInit()` – inicializace databáze  
`__get()` – upravený getter proměnných  
`__set()` – upravený setter proměnných  
 ...

### 3.12.1.2 Vzorová třída

Pro názornost nejprve ukázka např. třídy pro uložení informace o uživateli ve skupině:

```

class cyGroupUser extends abClass{
  public static $class_name = "cyGroupUser" ;
  public static $databaseName = "cyPermissionGroupUsers";
  protected $data=array(
    databaseName=>"cyPermissionGroupUsers",
    className => "cyGroupUser");
  public $vars = array(
    group_id => array(
      type=>'object',
      className => 'cyGroup',
      getter=> "getSelect",
      args=>array("group_id", "object")),
    user_id => array(
      type=>'object',
      className => 'cyUser',
      getter=> "getSelect",
      args=>array("user_id", "object"))
  );
  /*přetížená metoda __toString() */
  function __toString(){
    $this->checkAndGet();//dolování dat, pokud ještě nejsou
    return $this->getValue('group_id')
      ." - ".$this->getValue('user_id');
  }
}

```

Ve třídách je vlastní správa proměnných. Základem jsou v ní dvě význačné proměnné `$data` a `$vars`. V první zmiňované (`$data`) jsou uloženy do pole základní informace o třídě. Druhou tvoří proměnné, do kterých se ukládají hodnoty konkrétní třídy. Jedná se opět o pole. Zvláštností je, že zde můžeme používat typ

object. Následuje `className` udávající typ objektu. Pro názornost si ukážeme příklad: Vytvoříme-li instanci třídy `cyGroupUser`, a chceme získat objekt skupiny, ve které se uživatel nachází, museli bychom, v případě standardní práce s třídou, vykonat tento kód:

```
class cyUserGroup(){public group_id; ...}
$objekt = new cyUserGroup(1); # - ID je 1
$instanceSkupiny = new cyGroup($objekt->group_id);
```

V této aplikaci, díky vlastnímu zpracování proměnných, lze celý kód pro získání instace třídy skupiny zjednodušit takto (viz předchozí ukázka třídy):

```
$objekt = new cyUserGroup(1);
$instanceSkupiny = $objekt->group_id;

#možná by se hodilo lepší pojmenování proměnné na pouhé group → ...
$instanceSkupiny = $objekt->group;
```

Typy proměnných, které jsou použitelné a rozlišované, jsou následující:

object (className, popř getter a args), date, datetime, text, varchar, int, enum (set=>array(položky enum,...))...

### 3.12.1.3 Objektově relační mapování databáze

Pro uchovávání dat je aplikace úzce spojena s databází a její strukturou. Pro snazší provádění změn a práci s proměnnými byl vyvinut mechanismus mapující proměnné databáze na proměnné tříd. Aplikace rovněž mapuje nejen proměnné, ale i vztahy mezi nimi, a to pomocí cizích klíčů databáze. Proměnná (atribut databázové entity) odkazující pomocí cizího klíče na jiný objekt (entitu) je označena jako typ objekt a je s ním zacházeno jako s objektem.

Tento mechanismus mapování má na starost metoda `mapSQLvars` ve třídě `abClass`:

```
#nacteni sloupce = promennych
$this->Database->array_query($sloupc,
"SHOW FULL COLUMNS FROM ".$this->databaseName."");
```



```

#nacteni cizich klicu -> pro dany sloupec(promennou) to znamena, ze se
s ni bude pracovat jako s objektem
$this->Database->array_query($klice,
    "SELECT table_name,column_name, `referenced_table_name`,
    `referenced_column_name`
    FROM information_schema.KEY_COLUMN_USAGE
    WHERE TABLE_SCHEMA = '".DATABASE_NAME.'"
    AND TABLE_NAME      = '". $this->databaseName.'"
    AND REFERENCED_TABLE_NAME IS NOT NULL;"
);
#projdem vsechny nactene sloupce
foreach ($sloupec as $k => $sloupec){
    $isObject = false;
    #pokud je cizim klicem - odkaz na novy objekt
    if (is_array($klice)) foreach($klice as $kk => $vv){
        if ($vv["column_name"]===$sloupec["Field"]){
            $isObject =true;
            break;
        }else $isObject =false;
    }
}
#namapovani atributu na promenne
#typ promenne
(! $isObject)?
    preg_match("/([a-z0-9+])\([0-9+\)\$/",
    $sloupec["Type"],$matches):null;
    (! $isObject&&empty($matches[1]))?
    preg_match("/([a-z0-9+)]$/",
    $sloupec["Type"],$matches):null;
    $this->vars[$sloupec["Field"]]["type" ]=
    ($isObject)?"object":$matches[1];

    #muze nabývat nullových hodnot
    $this->vars[$sloupec["Field"]]["null" ] = $sloupec["Null"];
    #zda se jedná o klíčovy atribut
    $this->vars[$sloupec["Field"]]["key" ] = $sloupec["Key"];
    #pokud je klic = MUL - jedná se o objekt
    $this->vars[$sloupec["Field"]]["object" ] = $sloupec["Key"]=="MUL";
    #pokud je klic = PRI - jedná se o primární objekt
    $this->vars[$sloupec["Field"]]["primary" ] = $sloupec["Key"]=="PRI";
    #defaultní hodnota
    $this->vars[$sloupec["Field"]]["default" ] = $sloupec["Default"];
    $this->vars[$sloupec["Field"]]["extra" ]= $sloupec["Extra"];
    #textový popis atributu - komentář atributu
    $this->vars[$sloupec["Field"]]["title" ]= $sloupec["Comment"];
    #pokud má atribut maximalní délku
    preg_match("/\([0-9+\)\$/",
    $sloupec["Type"],$matches);
    (! $isObject)?$this->vars[
    $sloupec["Field"]]["maxLength" ]= $matches[1]:null;
    if ($isObject){
        #pokud je objektem / nastavení tridy objektu (entity)
        if (is_array($klice)) foreach($klice as $kk => $vv){
            if ($vv["column_name"]===$sloupec["Field"]){
                $this->vars[$sloupec["Field"]]["className" ] =
                    $vv["referenced_table_name"];
                break;
            }
        }
    }
}

```

```

#typ enum + namapujeme hodnoty výčtu
if (strpos($sloupec["Type"], "(")&&!$isObject){
$this->vars[$sloupec["Field"]]["type"] = substr($sloupec["Type"],0,
strpos($sloupec["Type"], "("));
if ($this->vars[$sloupec["Field"]]["type"]=="enum" ){
$this->vars[$sloupec["Field"]]["set"] = explode(',',
substr($sloupec["Type"],
strpos($sloupec["Type"], "(")+1,
-1));
if (is_array($this->vars[$sloupec["Field"]]["set"]))
foreach ($this->vars[$sloupec["Field"]]["set"] as $k=>$v)
$this->vars[$sloupec["Field"]]["set"][$k] = trim($v, "");
}
}

```

### 3.12.1.4 Třída – číselník

Třída nemusí fyzicky existovat, tj. nemusí být vytvořen žádný soubor. Stačí vytvořit v databázi tabulku s příslušným názvem, který se jmenuje stejně jako požadovaná třída. Systém, pokud takovouto třídu nenalezne fyzicky v souborovém systému, vytvoří její definici „za běhu“. K této proceduře slouží metoda `__autoload($className):`

```

function __autoload($class_name){
$directories = $GLOBALS['cfg']['class']['classdir'];
foreach($directories as $directory){#pro každý adresář
#zjistíme, zda existuje soubor, pokud ano, načteme a konec cyklu
if(file_exists($directory.$class_name . '.php')){
require_once($directory.$class_name . '.php');
return;
}
}
#pokud třída neexistuje, vytvoříme její definici
if (!class_exists($class_name)) {
eval ("
class $class_name extends abClass{
public static \$class_name = \"$class_name\";
public static \$databaseName = \"$class_name\";
protected \$data=array(
databaseName=>\"$class_name\",
className=>\"$class_name\"
);
protected \$vars=array(); #promenna pro promenne

function __toString(){
return \$this->title.\"\\\";
}
}
");
}
}

```

Jelikož dědí veškeré vlastnosti abstraktní třídy `abClass`, můžeme s ní ihned pracovat jako s objektem. Aplikace mapuje proměnné a cizí klíče z databázové tabulky. U proměnných mající v definici cizí klíč je nastaven příznak, že se jedná o proměnnou typu objekt (odkazuje na jiný objekt). Předchozí umožňuje snadno vytvářet netriviální objekty.

Omezujícím prvkem je zde také třída `abClass`, která sice obsahuje základní metody pro práci s daty, ale potřebujeme-li vlastní použití metody třídy, nezbývá nic jiného, než třídu vytvořit fyzicky v souborovém systému a metodu do ní implementovat.

Práci s daty takovéto třídy nám umožňuje podpora menu, kde stačí vytvořit příslušnou položku a jako typ zpracování se uvede `class` a název třídy (`entity`). Aplikace pak s daty pracuje pomocí základních universálních souborů, kterými jsou:

- `uni_call.php` – vstupní soubor – umožňující výpis, editaci, mazání a náhled
- `uni_index.php` – výpis všech záznamů
- `uni_edit.php` – editace konkrétního záznamu
- `uni_delete.php` – smazání jednoho záznamu
- `uni_view.php` – náhled jednoho záznamu

### 3.12.2 Přidání modulu

Modulem rozumějme část aplikace, mající na starost vybranou oblast. Pro přidání je nutné provést několik základních kroků:

1. uložení skriptů na server
2. přidání příslušných položek menu pomocí webového rozhraní
3. použité třídy nahrát do adresáře `cylibs`
4. případně vytvořit databázovou strukturu a data potřebná pro modul

**Příklad:** modul pro překládání (language)

1. potřebné skripty: `index.php`, `edit.php`, `delete.php` nahrát na web
2. na webu: `setting>menu` . přidat pro každý zpracovatelný skript položku:

`index.php` – název: překladač, jako menu: ano, adresář: language, soubor:  
`index.php`

`edit.php` – název: editace překladu, jako menu: ne, adresář: language,  
soubor: `edit.php`

`delete.php` – název: smazání překladu, jako menu: ne, adresář:  
language, soubor: `delete.php`

3. třídu `cyLanguage` nahrát do adresáře `core/classes`
4. vytvořit příslušnou DB strukturu: vytvoření DB tabulek `cyLanguage` – seznam jazyků a `language_words` – seznam překladů
5. tvoření odkazů -> `text.(ID-položky-menu).jazyk`, příklad: `seznam-prekladu.154.2` (`text:seznam-prekladu, id:154,jazyk:2`) – id položky je závislé na položce

### 3.12.3 Systém řízení báze dat

Aplikace je vyvinutá na MySQL. Pokud bychom chtěli aplikaci provozovat na jiném systému, stačí zaměnit či příslušně upravit třídu `cyDb`, jako jediná komunikuje s databázovým systémem.

Jelikož je v systému využíváno přímé objektově relační mapování, při změně databázového systému se rovněž musí přizpůsobit i toto mapování v metodě `mapSQLvars` ve třídě `abClass`.

Pro práci s databází je na počátku zpracování aplikace vytvořeno perzistentní připojení k databázi, které je uchováváno v globální proměnné `$db`. V abstraktní třídě je definována metoda inicializace databáze, která každé zpracovávané třídě předá handle k této globální proměnné. V každé třídě tedy existuje proměnná `$this->Database`, pomocí které komunikuje s databázovým systémem.

Inicializace databáze:

```
$this->dbInit();
```

Příklad vykonání SQL dotazu:

```
$this->Database->query(
    "DELETE FROM {$this->databaseName} WHERE id='{$this->ID}'");
```

Případně dotaz, vracející pole položek:

```
$this->Database->array_query($items,
    "SELECT id FROM ".$this->databaseName.";")
);
```

## 3.12.4 Šablony a vzhled

### 3.12.4.1 Šablony

V této aplikaci je maximálně použito šablon pro oddělení vzhledu od aplikační úrovně. Šablony se nacházejí, až na výjimky, v adresáři `tpl` u každého modulu a základní šablony jsou uloženy v kořenovém adresáři aplikace v adresáři s názvem rovněž `tpl`.

Struktura šablony je totožná s klasickou webovou stránkou s jediným rozdílem, že hlavičku a tělo tvoří pouze základní šablona stránky `index.tpl`. Ostatní šablony tvoří částečné fragmenty HTML stránky. Šablony však mohou obsahovat i PHP skripty, a jejich umístění zde je spíše výjimečné, kdyby nešlo použít jiné volby, nebo by výrazně znamenalo usnadnění problému.

O plnění šablon daty a jejich zpracování se stará třída `Template`. Pomocí parametrů konstrukturu je předán adresář se souborem šablony a druhým parametrem je způsob zpracování proměnné, které nebyla přiřazena žádná data. Možnosti jsou tyto: nechat zanechat a vypsat název proměnné (`keep`), odebrat (`remove`), anebo ji zakomentovat (`comment`).

Šablony obsahují tzv. proměnné, do kterých jsou aplikační vrstvou vkládána data. Vložení probíhá vyhledáním patřičné proměnné v šabloně a jejím nahrazením daty aplikace. Struktura proměnné šablony je tvořena složenými závorkami a mezi nimi název – `{variable}`.

Vkládání více dat za jednu proměnnou, je umožněno tzv. bloky. Viz:

```
<!-- BEGIN blok_promennych -->
{promenna}
<!-- END blok_promennych -->
```

Příklad použití třídy `Template`:

```

$tplakce = new Template($_path["admin_tpl"], "nbsp");
$tplakce->set_file("page", "uni_index_action.tpl");
$tplakce->set_block("page", "akce", "akces");
foreach($akce as $k => $v ){
    $tplakce->set_var(array(
        "URL"=> $sess->url("$k.${v[id]}.${ln?id={$ca[$i]->Id()}" ),
        "Action"      => $lan->Translate($v[title]),
        "Action_value" => $actionTitle,
        "URL_img"     => $v[imgUrl],
    ));
    $tplakce->parse("akces", "akce", true);
}
$tplakce->parse("out", "page");
echo $tplakce->getphp("out");

```

Příklad šablony pro předchozí příklad `uni_index_action.tpl`:

```

<!-- BEGIN akce -->
<a href="{URL}"
  onClick="
    return confirmLink(this, '{Action}:
    {Action_value}')">
  
</a>
<!-- END akce -->

```

### 3.12.4.2 Vzhled

Základ celkového vzhledu lze měnit pomocí kaskádových stylů. Kaskádové styly jsou uloženy v adresáři `css` nacházející se v kořenovém adresáři aplikace. Základem jsou tyto tři základní soubory:

`layout.css` - rozložení elementů hlavní stránky, nastavení jejich velikosti a okrajů

`style.css` - informace o vzhledu, okrajích a dalších nastavení elementů HTML

`menu.css` - informace o vzhledu menu

Hlavním stavebním prvkem stránky je šablona `index.tpl`, který tvoří základ celé webové aplikace. V těle jsou významné elementy:

`<div id="contentHolder">` - obsahuje celou webovou stránku

`<div id="menu">` - obsahuje menu tvořené seznamem `<ul>` a prvky `<li>`

`<div id="content">` - obsahuje veškerý zbylý obsah webové stránky

```
<h1 id="nadpis"> - nadpis obsahu
```

Tělo základu webové aplikace vypadá následovně:

```
<body>
  <div id="contentHolder">
    <div id="top">
      <div id="logo" />
      <h1 id='main_title'>{main_title}</h1>
      <div id="jazyky">{lngs}</div>
    </div>
    <div id="menu">{menu}</div>
    <div id='content'>
      <h1 id='nadpis'>{textTitle}</h1>
      {text}
    </div>
  </div>
</body>
```

### 3.12.5 Databáze

Zde se budeme věnovat především významným tabulkám, nutným pro minimální funkčnost samotného systému. Tuto skupinu tvoří tedy následující:

`active_session` – uloženy aktivní session a čas

`auth_user` – uživatelská tabulka

tvůřící základní rozšíření (+ modul language):

`csCmsMenu` – položky menu, lze

`cyLanguage` – seznam jazykových mutací

`language_words` – termíny k překladu a jejich překlady

### 3.12.6 Logování událostí

Jisté významné události jsou logovány do příslušné databázové tabulky. Logování má na starost třída `cyLog`, ve které jsou implementovány statické metody a to buď pro konkrétní, nebo obecné události. Konkrétní událostí je například přiřazení autora či recenzenta ke článku. Tato událost je zalogována tímto způsobem:

---

```
cyLog::Assigned($status,$user_id,$paper_id);
```

- proměnná `$status` předává informaci o tom, zda se jedná o autora či recenzenta
- proměnná `$user_id` předává informaci o uživateli
- proměnná `$paper_id` předává informaci o článku

Proměnné `$user_id` a `$paper_id` mohou být předávány jak v objektové formě tak pouze jejich ID.

### 3.12.7 Emailová notifikace

V aplikaci je rovněž potřeba zajistit informovanost uživatelů o různých událostech, které se jich týkají. K tomu slouží třída `cySendMail` a její statické metody pro konkrétní události. Je úzce spjata se třídou `cyLog`, kde jsou téměř všechny stejné události logovány a proto se většina notifikací rozesílá současně s jejich zalogováním. Na příkladu si ukážeme odeslání informace, že byl článku přiřazen recenzent. Tato událost je tedy logována třídou `cyLog` a její statickou metodou `Assigned` a je zde zavolána statická metoda `sendReviewAssignment` třídy `cyLog`. Konkrétní příklad:

```
cySendMail::sendReviewAssignment($user, $paper);
```

- proměnná `$user` předává informaci o uživateli
- proměnná `$paper` předává informaci o článku

Obě proměnné jsou předávány jako objekty uživatele a článku.



### 3.13 Rychlost

V počátku jsme si dali za úkol také otestovat rychlost. Co se týká přístupů k webu a webové aplikaci, je závislá rychlost zpracování především na konfiguraci serveru, ať již po hardwarové nebo softwarové stránce, a rychlostí připojení do internetové sítě jak uživatelského počítače, tak i serveru. Primárním předpokladem je fungování konferenčního systému pro odborníky v grafické sféře, proto lze téměř s jistotou očekávat, že přílohy u příspěvků budou dosahovat až několika desítek megabajtů. Zde je však problém omezení nastavení na serverové stránce maximální velikosti souboru. Je tedy dost dobře možné, že uživatel (autor) bude muset svou přílohu rozdělit do několika parciálních souborů. Podle prvotních testů roste s počtem uživatelů i doba, po kterou se větší soubory nahrávají na server.

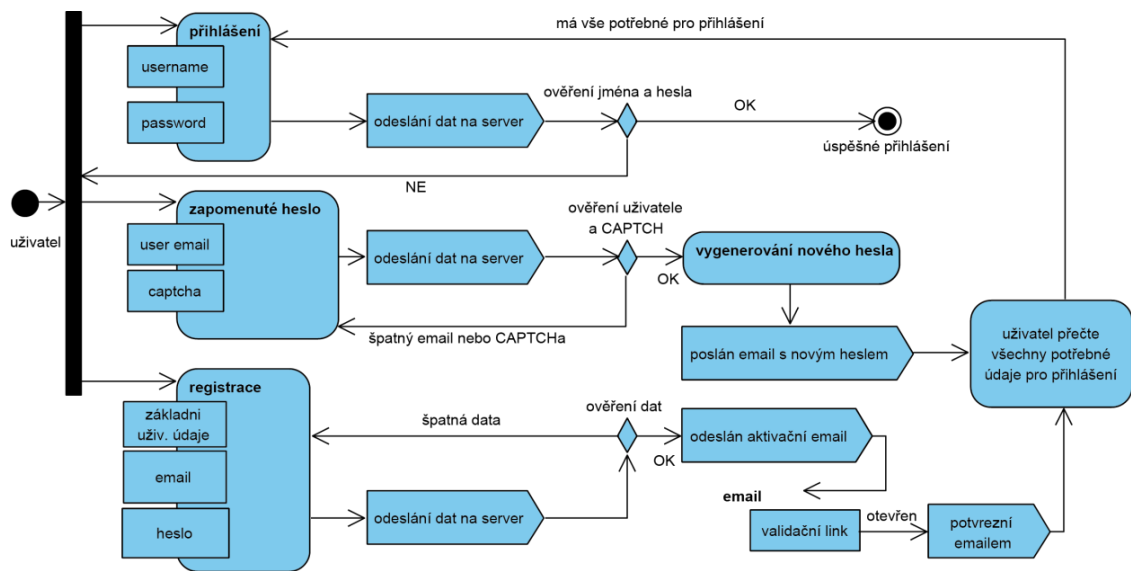
K provedení testu byla vytvořena aplikace – internetová stránka, která simulovala počet uživatelů přistupujících k aplikaci prostřednictvím počtu vláken, nebo spíše odeslání formulářových dat pomocí technologie AJAX. Nejedná se tedy o zcela objektivní test, jelikož tyto „vláknoví uživatelé“ se dělili o hw kapacitu testovacího počítače a kapacitu přenosové linky do sítě internet. (použitý počítač – notebook HP nx9420, dvoujadrový procesor, 4 GB RAM, 750GB 7200 RPM HDD a internetové připojení WiFi 15-20Mbit/s download / 5Mbit upload)

uživatel \ MB	1	2	5	10	20
0,25	0,251 s	0,538 s	0,823 s	1,513 s	2,165 s
0,5	0,435 s	0,720 s	1,365 s	2,252 s	4,729 s
1	0,590 s	0,896 s	2,151 s	4,851 s	9,538 s
2	0,957 s	1,987 s	4,238 s	7,952 s	17,159 s
5	1,997 s	4,124 s	10,834 s	22,483 s	52,382 s
10	5,968 s	9,835 s	21,657 s	58,452 s	103,425 s
25	15,357 s	24,235 s	63,357 s	120,254 s	242,622 s
50	24,175 s	54,846 s	132,756 s	249,168 s	498,251 s

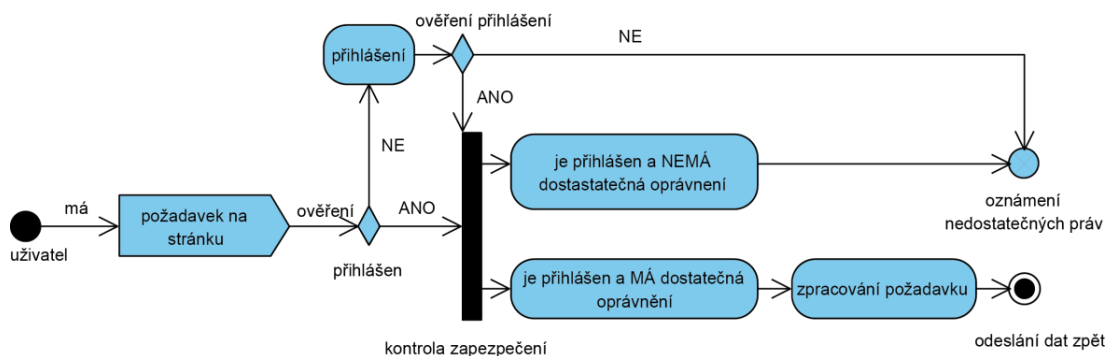
Z tabulky je vidět, že doba se s počtem uživatelů zvyšuje exponenciálně. To může být ale způsobeno výše uvedenými problémy s kapacitami hw a je tedy možné, že pokud by uživatelé přistupovali z různých částí internetu, čas potřebný pro nahrávání příloh by se nezvyšoval takto rychle.

### 3.14 Autorizace uživatele

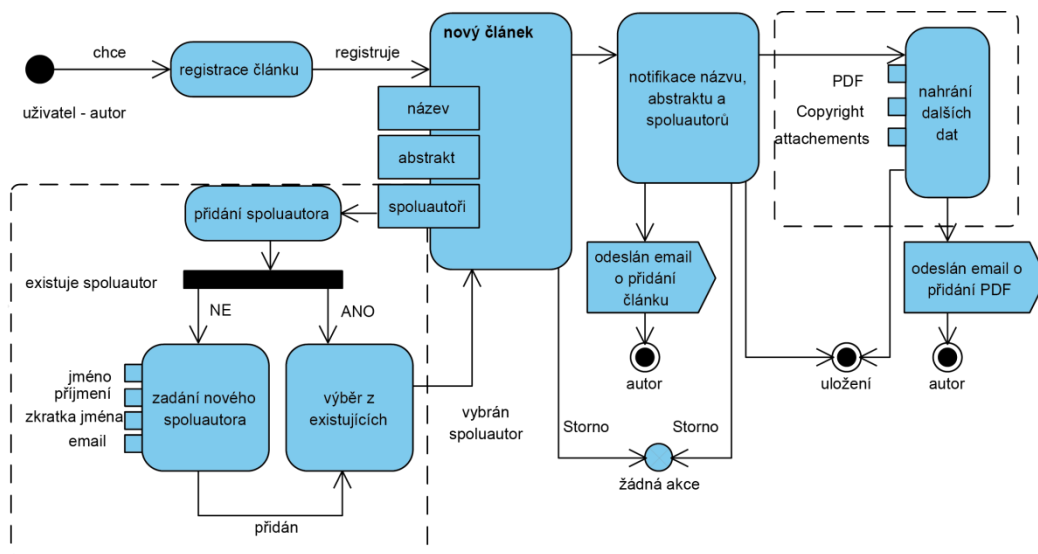
Na následujících obrázcích si názorně ukážeme proces přihlašování, registrování a ověřování práv uživatele.



Obrázek 3.3 schéma přihlášení a registrace



Obrázek 3.4 ověření práv



Obrázek 3.5 registrace nového příspěvku

## 4 Závěr

Při práci na této aplikaci jsem vytvořil základ vlastního php Framework, takže další rozšíření se již budou přidávat snáze a rychleji. Dále zde je použito objektivě relační mapování databází, což minimalizuje možnost chybného zápisu proměnných při dolování či ukládání dat a práce s nimi, vzhledem k benevolentnosti programovacího jazyka PHP. Byla rovněž vyvinuta vlastní správa proměnných, díky které bylo také možné pracovat s proměnnými jako s objekty a automatické vytváření jejich instancí.

Funkcionalita aplikace splňuje požadavky zadání, které byly v průběhu vývoje upřesňovány vedoucími práce. Je tedy možné zadat do systému příspěvek (název a abstrakt), je zaslán email o novém příspěvku autorovi a administrátorovi konference, v druhé fázi je umožněno nahrání plného znění příspěvku. Následuje přidělení recenzentů k příspěvkům, které je možné provést buď hromadným importem recenzentů, anebo ručním přiřazením recenzenta k příspěvku. Recenzentovi je po přihlášení umožněno napsat posudek příspěvku a určit verdikt. Systém průběžně zasílá notifikace zúčastněným stranám – uživatelům a administrátorovi. Následuje fáze nahrání opraveného příspěvku a určení finálního verdiktu nad ním. Systém umožňuje import uživatelů (přidání, smazání), přiřazení recenzentů k příspěvkům a rozhodnutí o příspěvku. Exportovat je možné finální příspěvky, abstrakty a recenze.

Hlavní devizou této aplikace na rozdíl od ostatních konferenčních systémů je právě výše zmiňovaná funkcionalita importů a exportů. Dalším rozdílem od ostatních konferenčních systémů je unikátní označování příspěvků, kdy je za označení použito velké písmeno a následuje dvojčíferné prvočíslo (např. A13, X31, ...).

Tento systém byl vyvíjen jako jednokonferenční, ale rovnou připravován na podporu více konferencí. Databázová struktura je víceméně připravena, akorát je zapotřebí upravit části php kódu. Dalším možným rozšířením systému by byla také možnost archivace konferencí.

Při vývoji této aplikace byly problémy většinou s benevolentností PHP jazyka a také bug. Jelikož se jedná o celkem rozsáhlou problematiku se složitými vztahy, bylo

z počátku problematické pochopení návazností a vztahů mezi jednotlivými událostmi a uživatelskými skupinami.

---

## LITERATURA A POUŽITÉ ZDROJE

- [1] Topinfo. [Online]. <http://www.topinfo.cz/cz/system-pro-podporu-odbornych-konferenci>
- [2] Public Knowledge Project. [Online]. <http://pkp.sfu.ca/?q=ocs>
- [3] Harald Weinreich. ConfTool. [Online]. <http://www.conftool.net>
- [4] ConfTool. [Online]. [http://www.conftool.net/en/vsis\\_conftool\\_license.html](http://www.conftool.net/en/vsis_conftool_license.html)
- [5] P. D. T. Preuss. ConfMaster - The Conference Management System: ConfMaster Home. [Online]. [www.confmaster.net](http://www.confmaster.net)
- [6] R. v. d. Stadt. Borbala Online Conference Services - CyberChair. [Online]. <http://www.borbala.com/cyberchair/>
- [7] M. Branický. Interval.cz. [Online]. <http://interval.cz/clanky/javaserver-pages-pro-vsechny/>
- [8] The Apache HTTP Server Project. [Online]. [http://httpd.apache.org/ABOUT\\_APACHE.html](http://httpd.apache.org/ABOUT_APACHE.html)
- [9] Netcraft. [Online]. <http://news.netcraft.com/archives/2011/02/15/february-2011-web-server-survey.html>
- [10] Oracle Corporation. MySQL :: The world's most popular open source database. [Online]. <http://www.mysql.com/>
- [11] Oracle. Oracle a MySQL | Databáze | Oracle. [Online]. <http://www.oracle.com/cz/products/database/mysql/index.html>
- [12] Oracle. Oracle. [Online]. [http://download.oracle.com/docs/cd/E11882\\_01/server.112/e17118/ap\\_standard\\_sql003.htm#g14](http://download.oracle.com/docs/cd/E11882_01/server.112/e17118/ap_standard_sql003.htm#g14)
- [13] About. [Online]. <http://www.postgresql.org/about/>
- [14] PostgreSQL. [Online]. <http://www.pgsql.cz/index.php/PostgreSQL>
- [15] Open Source Initiative. [Online]. <http://www.opensource.org/licenses/bsd-license.php>
- [16] Apache CouchDB. [Online]. <http://couchdb.apache.org/>
- [17] erlang.org. Erlang Programming Language. [Online]. <http://www.erlang.org/>
- [18] E. Kneuss. colder.ch. [Online]. <http://www.colder.ch/news/08-24-2007/28/late-static-bindings-expl.html>

- [19] flyguy@gmail.com. (2010, ) PHP :: Bug #52741 :: OOP late static bindings bug. [Online]. <http://bugs.php.net/bug.php?id=52741>
- [20] phpMyAdmin devel team. (2011, ) phpMyAdmin. [Online]. [www.phpmyadmin.net/home\\_page/index.php](http://www.phpmyadmin.net/home_page/index.php)
- [21] P. Stěhule. (2005, ) ROOT.CZ. [Online]. <http://www.root.cz/clanky/mysql-vs-postgresql-vs-firebird/>
- [22] The Firebird Project. Firebird - The RDBMS that's going where you're going. [Online]. <http://www.firebirdsql.org/>
- [23] SQLite About. [Online]. <http://www.sqlite.org/>
- [24] I. ., Sybase. Advantage Database Server - Relational Data Management System and Software - Sybase Inc. [Online]. <http://www.sybase.com/products/databasemanagement/advantagedatabaseserver>
- [25] IBM- DB2 for Linux, UNIX and Windows Features and Benefits. [Online]. <http://www-01.ibm.com/software/data/db2/linux-unix-windows/features.html>
- [26] Database 11g | Oracle Database 11g | Oracle. [Online]. <http://www.oracle.com/cz/products/database/index.html>

---

## SEZNAM POUŽITÝCH ZKRATEK A TERMÍNŮ

### **API** Application Programming Interface

- rozhraní, které programátor určuje volání procedur, knihoven apod. ve svém zdrojovém kódu

### **CSS** Cascading Style Sheets

- kaskádové styly

### **HTML** HyperText Markup Language

- značkovací jazyk pro zápis WWW dokumentů

### **HTTP** HyperText Transfer Protocol

- protokol pro přenos hypertextu – základní přenosový protokol WWW

### **path** (cesta)

- zde součást URI označující umístění dokumentu na serveru

### **query** (dotaz)

- zde součást URI (za otazníkem) obsahující dodatečné informace

### **Rebuttal**

- vyvracení tvrzení

### **session-id** (identifikátor relace)

- termín užívaný pro jedinečnou identifikační informaci odlišující uživatele

### **SŘBD**

- systém řízení báze dat (angl. DBMS – database management systém)

### **tag**

- obecný název pro značku v jazyce HTML; každá značka má konkrétní význam

### **URI** (Uniform Resource Identifier)



- jednotný identifikátor zdroje – univerzální schéma užívané pro adresování zdrojů

**URL** (Uniform Resource Locator)

- jednotný ukazatel na zdroj – podmnožina URI identifikuje dokument místem uložení

**web**

- zde ve smyslu obsah konkrétního HTTP serveru

**WWW** (World Wide Web)

- celosvětová pavučina – služba internetu pro přenos hypertextu (HTML) přes HTTP

## SEZNAM TABULEK

Tabulka 1.1 - činnosti aktérů v jednotlivých fázích.....	23
Tabulka 4.1 - mapování databázových entit na aplikační třídy .....	33

---

## SEZNAM OBRÁZKŮ

Obrázek 2.1 obsluha servletem [7] .....	11
Obrázek 2.2 JSON objekt.....	14
Obrázek 2.3 Struktura CouchDB .....	14
Obrázek 3.1 životní cyklus příspěvku.....	22
Obrázek 5 - základní menu .....	37
Obrázek 3.3 schéma přihlášení a registrace .....	51
Obrázek 3.4 ověření práv .....	51
Obrázek 3.5 registrace nového příspěvku .....	52

## **PŘÍLOHY**

**A. Uživatelský manuál**

**B. Instalační manuál**

**C. Programátorský manuál**

## D. Objektově relační mapování

Třída	Název atributu	Jméno tabulky	Název sloupce
cyUser	user_id	auth_user	user_id
Active_sessions	sid name val changed user	active_sessions	sid name val changed user
Auth_user_log	id from time browser	auth_user_log	id from time browser
CsCmsMenu	id file dir active asMenuItem title method article_id docTitle docTitle1 docTitle2 image morder la	csCmsMenu	id file dir active asMenuItem title method article_id docTitle docTitle1 docTitle2 image morder la
CsMenu	id to_id up_id file article_id active asMenuItem title method docTitle docTitle1 docTitle2 image morder la	csMenu	id to_id up_id file article_id active asMenuItem title method docTitle docTitle1 docTitle2 image morder la
CyCoauthor	id	cyCoauthor	id

CyConference	id name date place url acronym mail chair_mail is_submission_ opened is_discussion_ opened nb_reviewers_ per_paper  size_off_ballot mail_on_abstract_ _submission mail_on_paper_ _upload mail_on_review_ _submission	cyConference	id name date place url acronym mail chair_mail is_submission_ -opened is_discussion_ -opened nb_reviewers_ per_paper size_off_ballot t mail_on_abstra ct_ _submission mail_on_paper_ _upload mail_on_review_ _submission
CyCountry	ID region short de en code nl fr	cyCountry	ID region short de en code nl fr
CyDiet	id name	cyDiet	id name
CyFile	id type code created_time name	cyFile	id type code created_time name
CyHotel	id name	cyHotel	id name
CyHotelRoomType	id type	cyHotelRoomType	id type
CyImage	id code name description	cyImage	id code name description

	created_time modified_time		created_time modified_time
CyLanguages	languages_id languages_name languages_code  languages_image languages_directory languages_sort_order  languages_active	cyLanguages	languages_id languages_name languages_code languages_image languages_directory languages_sort_order languages_active
CyLog	id type time message	cyLog	id type time message
CyPaper	id sign title abstrakt keywords version status section2 submit_lock review_lock used_in phase	cyPaper	id sign title abstrakt keywords version status section2 submit_lock review_lock used_in phase
CyPaperAtt	id sign file type	cyPaperAtt	id sign file type
CyPaperType	id title short	cyPaperType	id title short
CyPermission	id read write delete	cyPermission	id read write delete
CyPermission-Group	id name	cyPermissionGroup	id name
CyPermission-GroupSites		cyPermissionGroupSites	

	id read write delete		id read write delete
CyPermission-GroupUsers	id	cyPermissionGroupUsers	id
CyReview	id cycle comment conf_comment rating rating1 rating3 rating4 rating5 rating6 rating7 rating8 rating9 rating10 status id_main_review	cyReview	id cycle comment conf_comment rating rating1 rating3 rating4 rating5 rating6 rating7 rating8 rating9 rating10 status id_main_review
CyReview-Assignment	id	cyReviewAssignment	id
CyReviewMain	id comment ipc_comment status cycle	cyReviewMain	id comment ipc_comment status cycle
CyReviewQuestion	id number_question question strength	cyReviewQuestion	id number_question question strength
CyReviewRating	id num_rating text_rating category describe_rating	cyReviewRating	id num_rating text_rating category describe_rating
CyReviewResult	id result	cyReviewResult	id result
CySection	id	cySection	id



	id_sequence name is_conference sub_deadline  sub_deadline_msg rev_deadline  rev_deadline_msg ipc_deadline ipc_over_deadline  ipc_deadline_msg fin_sub fin_deadline  fin_deadline_msg		id_sequence name is_conference sub_deadline sub_deadline_msg sg rev_deadline rev_deadline_msg sg ipc_deadline ipc_over_deadline  ipc_deadline_msg sg fin_sub fin_deadline fin_deadline_msg sg
CySections	id_section id_sequence id_conference name is_conference sub_deadline  sub_deadline_msg rev_deadline  rev_deadline_msg ipc_deadline ipc_over_deadline  ipc_deadline_msg fin_sub fin_deadline  fin_deadline_msg	cySections	id_section id_sequence id_conference name is_conference sub_deadline sub_deadline_msg sg rev_deadline rev_deadline_msg sg ipc_deadline ipc_over_deadline  ipc_deadline_msg sg fin_sub fin_deadline fin_deadline_msg sg
CyText	id authorText author modified modified_time created_time isPublished published_time	cyText	id AuthorText Author Modified Modified_time Created_time IsPublished Published_time
CyText_Description	headline termin intro	cyText_Description	headline termin intro

	contents meta_description keywords link_text		contents meta_description keywords link_text
CyUserAccommodation	id arrival departure stored	cyUserAccommodation	id arrival departure stored
CyUserPayment	id arrived_currency arrived note received	cyUserPayment	id arrived_currency arrived note received
CyUserSkill	id rev_accept	cyUserSkill	id rev_accept
CyUserSkillGrade	id name Language_words id original translation	cyUserSkillGrade	id name language_words id original translation
CyConference-Fee	id title amount id_conf_reg	cyConferenceFee	id title amount id_conf_reg
CyConference-Registration	id fa_address swift iban vat	cyConferenceRegistration	id fa_address swift iban vat
CyUserPay	id amount	cyUserPay	id amount
CyCurrency	id title symbol rate	cyCurrency	id title symbol rate



*DOKUMENTACE A UŽIVATELSKÝ, INSTALAČNÍ, A PROGRAMÁTORSKÝ MANUÁL*

# Konferenční systém

---

## Přílohy

### Obsah příloh

1	Uživatelský manuál.....	5
1.1	Prvotní menu .....	6
1.2	Registrace .....	6
1.3	Zapomenuté heslo (Forgotten password) .....	8
1.4	Přihlášení.....	8
1.4.1	Úvodní stránka.....	8
1.4.2	Odebrání uživatele .....	9
1.4.3	Přidání uživatelovy odborné znalosti.....	10
1.4.4	Autor - menu .....	10
1.5	Příspěvky.....	11
1.5.1	Autor – přidání nového příspěvku .....	11
1.5.2	Autor – editace příspěvku .....	13
1.5.3	Administrátor – přehled příspěvků .....	14
1.5.4	Administrátor - přidání nového příspěvku.....	15
1.5.5	Administrátor – editace příspěvku .....	16
1.5.6	Náhled.....	17
1.6	Konference .....	18
1.6.1	Přehled .....	18
1.6.2	Editace .....	19
1.7	Workshopy – sekce .....	20
1.7.1	Přehled .....	20
1.7.2	Editace .....	21
1.8	Uživatelé .....	22
1.8.1	Přehled .....	22
1.8.2	Náhled.....	23
1.8.3	Editace .....	24
1.8.4	Výpis logu.....	26
1.8.5	Nastavení práv uživatele.....	27
1.9	Uživatelské skupiny .....	27
1.9.1	Přehled uživatelských skupin.....	28
1.9.2	Editace .....	28
1.9.3	Nastavení práv skupiny.....	28
1.10	Uživatelé a skupiny .....	29
1.10.1	Přehled .....	29
1.10.2	Editace .....	29
1.11	Přiřazení příspěvků pro hlavní recenze .....	30
1.12	Přiřazení spoluautorů k příspěvkům.....	30
1.12.1	Přehled .....	30
1.12.2	Editace .....	30
1.13	Členové IPC.....	31
1.13.1	Přehled .....	31
1.13.2	Úvodní stránka po přihlášení -dole.....	31
1.14	Diety .....	32
1.14.1	Přehled diet .....	32
1.14.2	Editace diet .....	32
1.15	Recenze.....	33

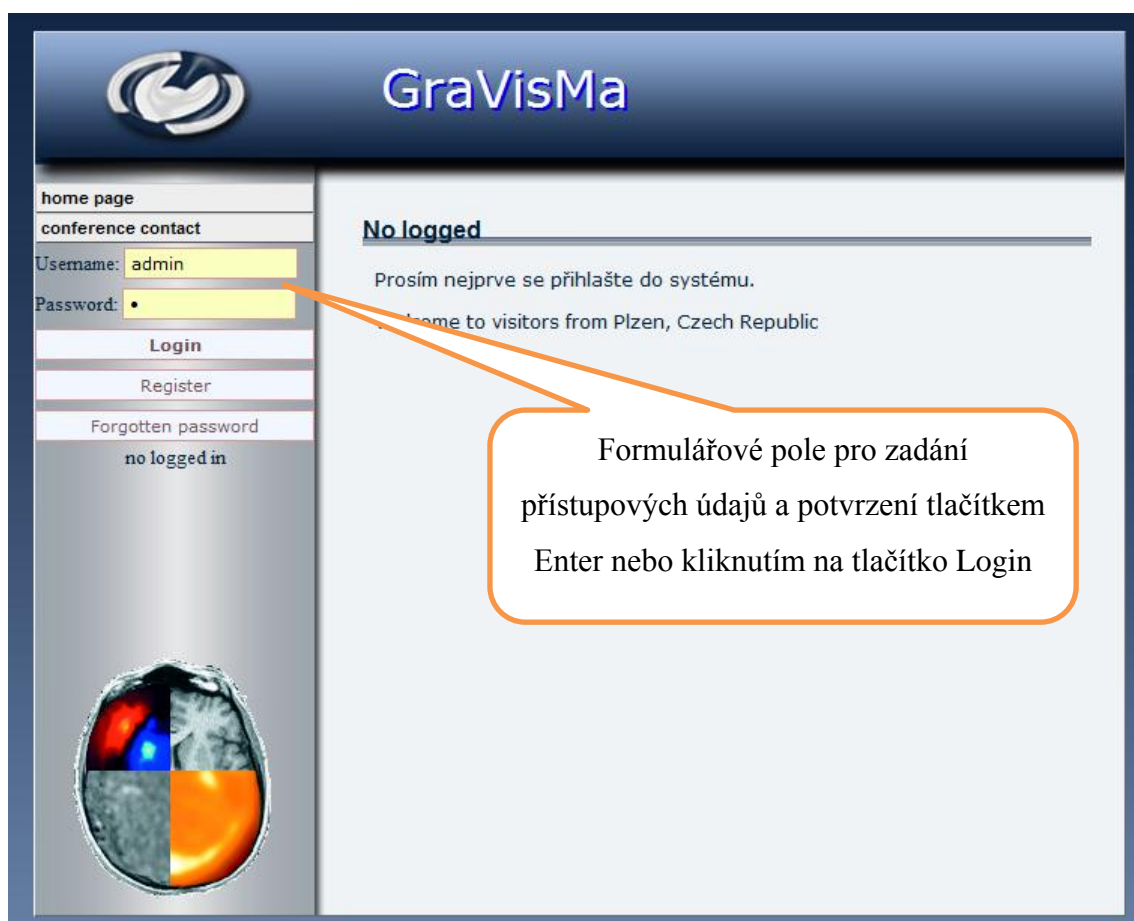
1.15.1	Přehled přiřazení recenzí k příspěvkům.....	33
1.15.2	Přehled recenzí.....	33
1.15.3	Editace .....	34
1.16	Otázky k recenzím.....	34
1.16.1	Přehled .....	34
1.16.2	Editace .....	35
1.17	Hodnocení recenzí .....	35
1.17.1	Přehled .....	35
1.17.2	editace .....	35
1.18	Přijetí žádosti o možnost recenzovat .....	36
1.19	Stupně uživatelských odborných znalostí .....	36
1.19.1	Přehled .....	36
1.19.2	Editace .....	37
1.20	Odborná kvalifikace uživatelů.....	37
1.20.1	Přehled .....	37
1.20.2	Editace .....	37
1.21	Konferenční poplatky .....	38
1.21.1	Přehled .....	38
1.21.2	Editace .....	38
1.22	Měny.....	38
1.22.1	Přehled .....	39
1.22.2	Editace .....	39
1.23	Platby uživatele .....	39
1.23.1	Přehled .....	39
1.23.2	Editace .....	40
1.24	Registrace na konferenci .....	40
1.25	Registrace na konferenci – admin .....	41
1.25.1	Přehled .....	42
1.25.2	Editace .....	42
1.26	Hotely a ubytovací zařízení .....	42
1.26.1	Přehled .....	43
1.26.2	Editace .....	43
1.27	Platby uživatelů .....	43
1.27.1	Přehled .....	43
1.27.2	Editace .....	44
1.28	Ubytování uživatelů.....	44
1.28.1	Přehled .....	44
1.28.2	Editace .....	45
1.29	Import .....	46
1.29.1	Import nových uživatelů .....	46
1.29.2	Import seznamu uživatelů ke smazání .....	47
1.29.3	Import přiřazení recenzentů k příspěvkům .....	47
1.29.4	Import rozhodnutí o příspěvcích.....	47
1.30	Export .....	48
1.30.1	Export abstraktů.....	49
1.30.2	Export finálních příspěvků.....	49
1.30.3	Export recenzí .....	49
1.31	Systémový administrátor .....	49
1.31.1	Menu .....	50
1.31.2	Menu .....	50
1.31.3	Přehled položek menu.....	50

1.31.4	Editace položky menu.....	51
1.32	Texty.....	52
1.32.1	Přehled.....	52
1.32.2	Editace.....	52
1.33	Slovník.....	53
1.33.1	Přehled textových termínů a jejich překladu.....	54
1.33.2	Editace překladu.....	54
2	Instalační manuál.....	55
2.1	Systémové požadavky.....	55
2.2	Instalace aplikace - souborový systém.....	55
2.3	Instalace struktury a dat do databáze.....	56
2.4	Nastavení konference.....	56
3	Programátorský manuál.....	57
3.1	Základ aplikace.....	57
3.1.1	Abstraktní třída abClass.....	57
3.1.2	Vzorová třída.....	57
3.1.3	Objektově relační mapování databáze.....	59
3.1.4	Třída – číselník.....	61
3.2	Přidání modulu.....	62
3.3	Systém řízení báze dat.....	63
3.4	Šablony a vzhled.....	63
3.4.1	Šablony.....	63
3.4.2	Vzhled.....	64
3.5	Databázové úložiště.....	65
4	ER model.....	66

## 1 Uživatelský manuál

Aplikace správy konference, je vytvořena jako většina standardních webových aplikací. Proto by s ní nemělo být obtížné pracovat a snahou autora bylo i intuitivní ovládání. K snadnému řešení případných nejasností by měl sloužit tento dokument.

Při příchodu na webovou adresu konferenčního systému, se nám zobrazí úvodní stránka. Abychom mohli s touto aplikací pracovat, je bezpodmínečně nutné mít vytvořené a aktivní uživatelské přístupové údaje. Pokud je nemáme, můžeme poslat dotaz na správce, který je vytvoří, anebo pravděpodobnější bude, vytvořit si základní konto svépomocí pomocí registračního formuláře, který se nám zobrazí po kliknutí na tlačítko „Register“. Pokud již přístupová data známe a uživatelské konto máme aktivní, nic nám nebrání přihlášení do systému. Jak se přihlásit vidíme na tomto obrázku:




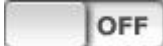





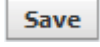





Obrázek 1 - úvodní zobrazení

Jak je již vepsáno v obrázku, je nutné zadat uživatelské jméno, heslo a následně stisknout klávesu ENTER nebo kliknout na tlačítko „Login“.



Základní ovládací prvky tvoří většinou intuitivní grafické ikony, jejichž význam je následující:

	- přidání nové položky		- položka aktivní
	- smazání položky		- položka neaktivní
	- editace položky		- zobrazení náhledu položky
	- práva uživatele nebo skupiny		- zobrazení náhledu položky
	- logovací výpis přístupů		- tlačítko uložit
	- zástupná ikona, pro stažení souboru		- tlačítko zpět
	- informační ikona pro zobrazení dalšího textu (text se zobrazí po najetí myši nad ikonu)		

## 1.1 Prvotní menu

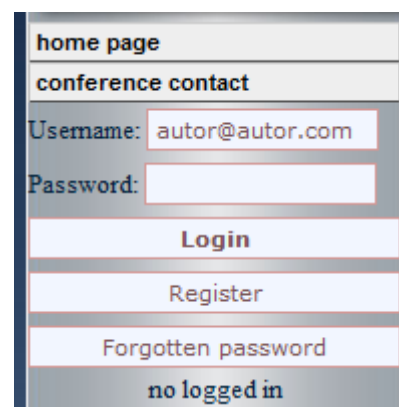
Toto menu je zobrazeno, než se uživatel přihlásí.

Jsou zobrazeny dvě základní položky menu i bez přihlášení:

- homepage – hlavní stránka pořádané konference
- kontakt konference (konference contact) – emailový odkaz na kontaktní osobu konference

Kromě základních položek menu, se zde nacházejí tři tlačítka:

- přihlášení (Login) – tlačítko pro přihlášení po vyplnění uživ. jména a hesla
- zapomenuté heslo (Forgotten password) – po vyplnění uživatelského jména a kliknutí na toto, je vygenerováno nové heslo a odesláno na kontaktní email.
- registrace (Register) – kliknutím na toto je zobrazen registrační formulář



The screenshot shows a vertical menu with the following items from top to bottom: 'home page', 'conference contact', a login form with 'Username: autor@autor.com' and a 'Password:' field, a 'Login' button, a 'Register' button, a 'Forgotten password' link, and 'no logged in' at the bottom.

Obrázek 2 - základní menu a přihlašovací formulář

## 1.2 Registrace

Nemáme-li, tedy přihlašovací údaje, přistoupíme k registraci. Po odeslání registračního formuláře systém zašle na zadanou emailovou adresu potvrzovací odkaz. Tím se aktivuje uživatelské konto.

- uživatelské jméno (username) – tato položka je pouze informační a nelze ji měnit, u nových „samo registrací“ je uživatelským jménem kompletní emailová adresa

Registration form fields:

- Username:
- Titul1:
- First name:
- Last name:
- Titul2:
- Sex:  male  female
- Institution:
- Department:
- Email:
- Confirm email:
- Password:
- Confirm password:
- Phone:
- Fax:
- Dietary:
- Address:
- ZIP code:
- City:
- Country:
- Homepage:
- Comment:

CAPTCHA: Z E H W C  
Tom's ConfSys

Obrázek 3 - registrační formulářObrázek 4

- skupiny (group) – skupiny, ve kterých je uživatel členem

- titul před jménem (title1)
- křestní jméno (first name) – křestní jméno uživatele
- příjmení (last name) – příjmení uživatele
- titul za jménem (title2)
- instituce (institution) – instituce, za kterou uživatel vystupuje v systému
- oddělení (department) – oddělení instituce
- email a potvrzení (email, confirm email) – email je potřeba zadat 2x stejně pro kontrolu proti překlepům, na tento email budou uživateli odesílány notifikace systémem a v případě zapomenutí hesla i nově vygenerované
- telefon (phone)
- fax (fax)
- strava (vegetarian) – zde je na výběr, co uživatel jí a nejí, zda vše, ryby, je vegan nebo vegetarián, toto je použito pro zajištění stravy, pokud se uživatel zúčastní konference
- adresa (address) – ulice
- PSČ (ZIP)
- město (city)
- země (country)
- webová stránka (homepage)
- komentář (comment) – zde o sobě může uživatel zanechat nějakou poznámku, či popis
- Captcha – potvrzovací kód, který je potřeba opsat z obrázku. Slouží k zablokování spamovacích robotů.

Po registraci je uživatel automaticky zařazen do skupiny „autorů“.

### 1.3 Zapomenuté heslo (Forgotten password)

Uživatel si může nechat vygenerovat nové heslo, pokud staré zapomene, zadáním uživatelského jména a kliknutím na „zapomenuté heslo“ (forgotten password). Systém vytvoří nové heslo, které odešle na email zadaný při registraci.

### 1.4 Přihlášení

#### 1.4.1 Úvodní stránka

Po přihlášení se zobrazí úvodní stránka uživatele se základním přehledem jeho zadaných dat a funkcemi, které může v systému provádět.

**Logged in**

Dear user **t t3** you are logged-in.  
Please, [view](#) or [edit](#) or [change password](#) your personal data.

**Paper list:**

id	sign	author	title	version	status	Workshop	pdf	supplement	copyright	Action
11	A02	t t3 [Bahamas]	muzu?	0	poster	GNU/Tesla/Fermi Programming (EPOCH_DECISION)				
13		t t3 [Bahamas]	Title author	0	submitted		 [-1 Bytes]	 [-1 Bytes]	 [-1 Bytes]	
14		t t3 [Bahamas]	Title your Paper 153	0		GNU/Tesla/Fermi Programming (EPOCH_DECISION)	 [-1 Bytes]	 [-1 Bytes]	 [-1 Bytes]	

[Add paper](#)

**Payment list:**

Person	Currency	Arrived	Note	Conference	Received
No item...					

**Fields of your expertise :**  
(if you want be reviewer)

Person	Section	Grade
t t3 [Bahamas]	Rendering Techniques (EPOCH_DECISION)	very confident

[Add expertise](#)

[Remove me](#)

Obrázek 5 - úvodní stránka po přihlášení

Všem uživatelům je v hlavičce umožněno zobrazit, editovat a měnit heslo pomocí třech odpovídajících odkazů.

Je-li přihlášen autor, následuje výpis vlastních přidaných příspěvků do konference. K přidání nového příspěvku slouží tlačítko „přidat příspěvek“ (add paper).

Dále je zde seznam plateb, které uživatel provedl. Jedná se o platby za přihlášky na konferenci apod.

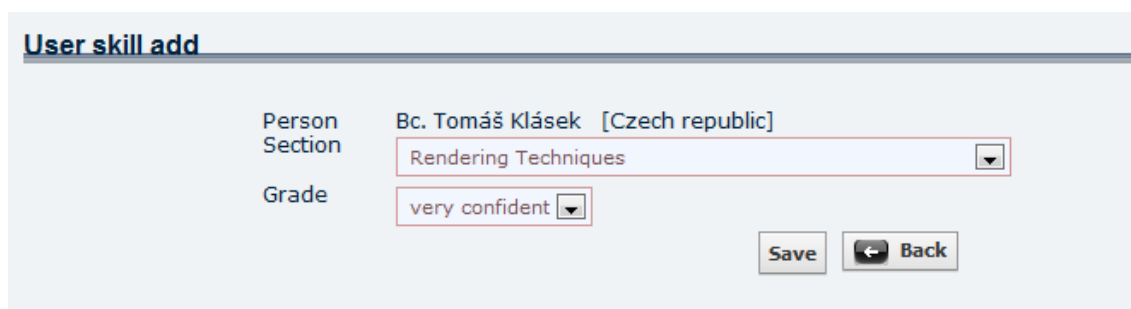
Následuje seznam uživatelových odborných znalostí pro daný workshop. Může si sám přidávat, či odebírat. Pokud zadá alespoň jednu položku, je zařazen do seznamu recenzentů. Tento však musí administrátor potvrdit, aby bylo možné pro ostatní články vybrat tohoto uživatele za recenzenta.

#### 1.4.2 Odebrání uživatele

Pokud se uživatel rozhodne dále v systému nefigurovat, provede to tlačítkem „odebrat mě“ (Remove me).

### 1.4.3 Přidání uživatelské odborné znalosti

Kliknutím na tlačítko „přidat zkušenost“ (add expertise), které se nachází na úvodní stránce uživatele po přihlášení.



**User skill add**

Person Section: Bc. Tomáš Klásek [Czech republic]

Section: Rendering Techniques

Grade: very confident

Save Back

Obrázek 6 - přidání znalosti uživatele

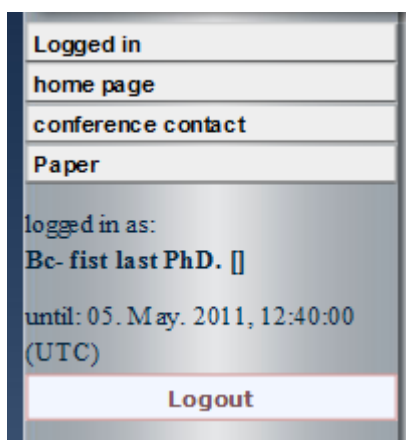
Ze seznamu vybereme workshop a příslušnou odpornou kvalifikaci pro tento workshop. Následně stačí jen kliknout na tlačítko uložit.

V seznamu úrovně odborné znalosti je možné vybrat tyto položky:

- vynikající (excellent)
- velmi jistý (very confident)
- jistý (confident)
- malé zkušenosti (little experience)
- žádné zkušenosti (no experience)

Přidáním odborné znalosti se uživatel zařadí do seznamu eventuálních recenzentů, ze kterých správce vybírá recenzenty příspěvků.

### 1.4.4 Autor - menu



Obrázek 7 - menu autora

## 1.5 Příspěvky

### 1.5.1 Autor – přidání nového příspěvku

Přidávat nové příspěvky mohou členové skupin administrátor nebo autor. Přidávání článku autorem je dvoufázové. V první nejprve zadá základní údaje o příspěvku, jako je název workshopu, název příspěvku, abstrakt a spoluautory. Po uložení a následné editaci již může přidávat přílohy.

The screenshot shows a web form titled "Paper edit". It contains the following fields and controls:

- Workshop:** A dropdown menu with "Rendering Techniques" selected.
- Title:** A text input field containing "Novy ukazkovy paper".
- Abstract:** A large text area containing "a k tomu ukazkovy abstrakt".
- Author:** A text field containing "t t3 [Bahamas]".
- Co-Authors:** A text field containing "author test [Czech republic]".
- Buttons:** An "add co-author" button (with a dotted border), a "Save" button, and a "Back" button (with a left arrow icon).

Obrázek 8 - přidání nového příspěvku autorem

- workshop (workshop) – zařazení příspěvku
- název (title)
- abstrakt (abstract)
- autor (author) – pouze informační – autor sám nemůže měnit
- spoluautoři – seznam spoluautorů a kliknutím na tlačítko přidat spoluautora (add co-author), lze vybrat ze seznamu autori, či jej přidat do systému (k tomu je nutné znát jméno a emailovou adresu)

ve druhé fázi – po uložení a znovueditování, lze přidávat další položky

**Paper edit**

Workshop

Title

Abstract

pdf

supplement

copyright

Attachments

Author

Co-Authors

phase

Obrázek 9 - editace příspěvku

- pdf – přiložený hlavní příspěvek ve formátu pdf
- supplement – příloha
- copyright – formát \*.copy,
- přílohy (attachments) – seznam příloh, po stisku tlačítka „přidat soubor“ (add-file) se zobrazí formulář pro výběr souboru, počet těchto příloh není omezen počtem

## 1.5.2 Autor – editace příspěvku

The screenshot shows the 'Paper edit' interface with the following fields and values:

- Workshop:** Geometric Computing
- Title:** Novy ukazkovy paper
- Abstract:** a k tomu ukazkovy abstrakt
- pdf:** DIP\_ titulni.pdf (with a 'Procházet...' button)
- supplement:** AndroidManifest.xml (with a 'Procházet...' button)
- copyright:** ZCUrootCA.der (with a 'Procházet...' button)
- Attachments:** ConfSys\_workflow.pdf
- Author:** t t3 [Bahamas]
- Co-Authors:** Tom Omega



































Buttons at the bottom include 'add file', 'add co-author', 'Save', and 'Back'.


Obrázek 10 - editace příspěvku

- workshop (workshop) – zařazení příspěvku
- název (title)
- abstrakt (abstract)
- pdf – přiložený hlavní příspěvek ve formátu pdf
- supplement – příloha
- copyright – formát \* .copy,
- přílohy (attachments) – seznam příloh, po stisku tlačítka „přidat soubor“ (add-file) se zobrazí formulář pro výběr souboru, počet těchto příloh není omezen počtem
- autor (author) – pouze informační – autor sám nemůže měnit
- spoluautoři – seznam spoluautorů a kliknutím na tlačítko přidat spoluautora (add co-author), lze vybrat ze seznamu autorů, či jej přidat do systému (k tomu je nutné znát jméno a emailovou adresu)



### 1.5.3 Administrátor - přehled příspěvků

id	sign	author	title	ver.	status	Works hop	pdf	supplement	copyright	Action
2	A02	Bc. Tomáš Klásek [Czech republic]	Title your Paper	3	submitted	GNU/Tesla/Fermi Programming (EPOCH_DECISION)	 [418 Bytes]	 [108.836 KB]	 [108.836 KB]	 
7	A02	Prof. Ing. Vaclav Skala CSc. [Czech republic]	Title your Paper	1	poster	GNU/Tesla/Fermi Programming (EPOCH_DECISION)				 
8		t2 f1 t2 []	Paper t2	0	poster	GNU/Tesla/Fermi Programming (EPOCH_DECISION)				 
9	A02	t t3 [Bahamas]	title	1	short paper	GNU/Tesla/Fermi Programming (EPOCH_DECISION)				 
10		t t3 [Bahamas]	nazev paperu	0	submitted	Rendering Techniques (EPOCH_DECISION)	 [5.525 kB]			 
11	A02	t t3 [Bahamas]	m uzu?	0	poster	GNU/Tesla/Fermi Programming (EPOCH_DECISION)				 
13		t t3 [Bahamas]	Title author	0	submitted		 [-1 Bytes]	 [-1 Bytes]	 [-1 Bytes]	 
14		t t3 [Bahamas]	Title your Paper 153	0		GNU/Tesla/Fermi Programming (EPOCH_DECISION)	 [-1 Bytes]	 [-1 Bytes]	 [-1 Bytes]	 
15		t t3 [Bahamas]	Title your Paper 155	0	submitted	Rendering Techniques (EPOCH_DECISION)	 [-1 Bytes]	 [-1 Bytes]	 [-1 Bytes]	 
16		t t3 [Bahamas]	test 201	0		GNU/Tesla/Fermi Programming (EPOCH_DECISION)	 [1.188 kB]			 

1 2 3 

Obrázek 11 - seznam příspěvků

### 1.5.4 Administrátor - přidání nového příspěvku

The screenshot shows the 'Paper edit' form with the following fields and controls:

- Title:** Text input field.
- Abstract:** Large text area for the abstract.
- Author:** Bc. Tomáš Klásek [Czech republic]
- Co-Authors:** add co-author button
- Version Status:** full paper (dropdown menu)
- Workshop:** Rendering Techniques (dropdown menu)
- Submit lock:**
- Review lock:**
- Sign:** Text input field
- pdf:** Text input field with Procházet... button
- supplement:** Text input field with Procházet... button
- copyright:** Text input field with Procházet... button
- Attachments:** add file button
- Reviewers:** add reviewer button
- Main reviewers:** add main reviewer button
- Buttons:** Save, Back

Obrázek 12 - přidání příspěvku administrátorem

- název (title)
- abstrakt (abstract)
- autor (author) – pouze informační – autor sám nemůže měnit
- spoluautoři – seznam spoluautorů a kliknutím na tlačítko přidat spoluautora (add co-author), lze vybrat ze seznamu autori, či jej přidat do systému (k tomu je nutné znát jméno a emailovou adresu)
- statut (status) - lze měnit rozhodnutí o příspěvku
- workshop (workshop) – zařazení příspěvku
- zamknoutí nahrávání (submit lock)
- zamknutí recenzování (review lock)
- zkratka (sign) - tvořena velkým písmenem abecedy a prvočíslem, standardně generováno systémem

- pdf – přiložený hlavní příspěvek ve formátu pdf
- supplement – příloha
- copyright – formát \*.copy,
- přílohy (attachments) – seznam příloh, po stisku tlačítka „přidat soubor“ (add-file) se zobrazí formulář pro výběr souboru, počet těchto příloh není omezen počtem
- recenzenti (reviewers) – zde je možno přiřadit a odebrat recenzenty k/od příspěvku
- hlavní recenzent (main reviewer)

### 1.5.5 Administrátor - editace příspěvku

The screenshot shows the 'Paper edit' interface with the following fields and values:

- Title:** Title your Paper
- Abstract:** neo japonsky マカマチ/オトイト/マカマチ
- Author:** Bc. Tomáš Klásek [Czech republic]
- Co-Authors:** add co-author
- Version:** 3
- Status:** submitted
- Workshop:** GNU/Tesla/Fermi Programming
- Submit lock:**
- Review lock:**
- Sign:** A02
- pdf:** 2008\_03\_29\_01\_fci\_CDMA\_WL500g.Premium\_FS.tar (Procházet...)
- supplement:** busajb7.jpg (Procházet...)
- copyright:** busajb7.jpg (Procházet...)
- Attachments:** add file
- Reviewers:** Bc- fist last PhD. [X] (add reviewer)
- Main reviewers:** Bc- fist last PhD. [X]

Obrázek 13 - editace příspěvku administrátorem

- název (title)
- abstrakt (abstract)
- autor (author) – pouze informační – autor sám nemůže měnit

- spoluautoři – seznam spoluautorů a kliknutím na tlačítko přidat spoluautora (add co-author), lze vybrat ze seznamu autori, či jej přidat do systému (k tomu je nutné znát jméno a emailovou adresu)
- statut (status) - zde měnit rozhodnutí o příspěvku
- workshop (workshop) – zařazení příspěvku
- zamknutí nahrávání (submit lock)
- zamknutí recenzování (review lock)
- zkratka (sign) - tvořena velkým písmenem abecedy a prvočíslem, standardně generováno systémem
- pdf – přiložený hlavní příspěvek ve formátu pdf
- supplement – příloha
- copyright – formát \*.copy,
- přílohy (attachments) – seznam příloh, po stisku tlačítka „přidat soubor“ (add-file) se zobrazí formulář pro výběr souboru, počet těchto příloh není omezen počtem
- recenzenti (reviewers) – zde je možno přiřadit a odebrat recenzenty k/od příspěvku

### 1.5.6 Náhled

Náhled se zobrazí, pokud chce autor editovat cizí, či již needitovatelní příspěvek. Rovněž je zobrazen v přehledech po kliknutí na jeho název.

**Paper view**

Title  
Abstract  
Author  
Co-Authors Prof. Ing. Vadav Skala CSc. [Czech republic]  
Version  
Status submitted  
Workshop  
Submit lock unlocked  
Review lock  
Sign  
pdf  
supplement  
copyright  
Attachments bg.jpg  
Reviewers Bc- fist last PhD. []  
Bc. Tomáš Klásek PhD : []  
Main reviewers t2 f1 t2 []

[← Back](#)

---

**Assigned reviews for cycle**

Title	conference comment	rating	status	cycle	Reviewer
komentar review paper 3	confi comment	2.00	reviewed	0	Bc- fist last PhD. []

---

**Cycles**


0

Obrázek 14 - náhled příspěvku



## 1.6 Konference

Administrátor má v menu i položku nastavení → konference, kde může spravovat a měnit různá nastavení. Nynější verze pracuje pouze s jednou konferencí. Do budoucna se počítá s podporou více konferencí.

### 1.6.1 Přehled

Nyní je v seznamu uvedena pouze jediná konference, kterou můžeme, pomocí editační ikony (  ) v pravé části, editovat.

**Conference Management System**

Name	Date	Place	URL	Action
GraVisMa	2011-08-31	Ostrava, Czech Republic	http://web19.srv2.loisachtal.net/dip/	 

1

Obrázek 15 - seznam konferencí

## 1.6.2 Editace

Zobrazí se tento formulář:

**conference edit**

Name	<input type="text" value="GraVisMa"/>
Date	<input type="text" value="2011-08-31"/>
Place	<input type="text" value="Ostrava, Czech Republic"/>
URL	<input type="text" value="http://web19.srv2.loisachtal.net/di"/>
Email	<input type="text" value="tomas@klasek.cz"/>
Chair email	<input type="text"/>
Acronym	<input type="text"/>
Is submission opened	<input checked="" type="radio"/> true <input type="radio"/> false
Is discussion opened	<input checked="" type="radio"/> true <input type="radio"/> false
Number reviewers per page	<input type="text" value="3"/>
Size off ballot	<input type="text" value="999"/>
Mail on abstract submission	<input checked="" type="radio"/> true <input type="radio"/> false
Mail on paper upload	<input checked="" type="radio"/> true <input type="radio"/> false
Mail on review submission	<input checked="" type="radio"/> true <input type="radio"/> false

Obrázek 16 - editace konference

Kde můžeme zadat tyto položky:

- název (name) – název konference
- datum (date) – datum konání
- místo (place) – místo konání
- webová adresa (URL) – adresa s detaily a dalšími informacemi o konferenci
- email – emailová adresa, jejímž jménem se odesílají systémové oznámení a zprávy
- email předsedy – emailová adresa na předsedu konference
- zkratka (acronym) – zkrácený název konference
- povolení přispívání (is submission opened) – povolení/zakázání nahrávání příspěvků
- povolení diskuze (is discussion opened) – povolení/zakázání diskuze nad příspěvkem
- počet recenzentů na stránku (number reviewers per page)
- zaslání email po uložení abstraktu (mail on abstract submission) – odeslání potvrzovacího emailu s webovou adresou k uloženému abstraktu – příspěvku

- zaslání emailu po nahrání pdf příspěvku (mail on paper upload) – odeslání potvrzovacího emailu s webovou adresou uloženého příspěvku
- zaslání emailu zrecenzování příspěvku (mail on review submission) – odešle potvrzující email po uložení recenze

## 1.7 Workshopy – sekce

V konferenčním systému je každý příspěvek přiřazen k určitému, předem zadanému, workshopu dle tématu. Tyto jsou administrátorem předem zadané a autor si je již vybírá ze seznamu.

Přehled, přidávání (🌐), editování (📄) a mazání (🗑️) může defaultně provádět administrátor konference. V základním přehledu je zobrazen kompletní seznam workshopů s daty jednotlivých částí (epoch), neboli deadline, do kdy je povoleno provádět příslušné činnosti. Krom dat je v přehledu zobrazen i popis epoch.

Pokud by se stalo, že autor nenalezne téma svého příspěvku, může být administrátorem vytvořeno téma např. „ostatní“. Podle příspěvků v tomto workshopu, lze pak přidat patřičné téma a příspěvek do daného workshopu přesunout, nebo je zde ponechat.

### 1.7.1 Přehled

Workshops											
Name	submission deadline	submission deadline message	review deadline	review deadline message	ipc deadline	ipc over deadline	ipc deadline msg	final submission	final deadline	final deadline message	Action
Rendering Techniques	2010-12-22 22:22:00	konec submission	2010-12-24 22:22:00	konec review	2011-01-21 00:00:00	2011-01-13 00:00:00		2011-03-25 00:00:00	2011-03-25 00:00:00	deadline	
GNU/Tesla/Fermi Programming	2010-12-22 22:22:00		2010-12-24 22:22:00		0000-00-00 00:00:00	0000-00-00 00:00:00		2010-12-31 22:22:00	0000-00-00 00:00:00		
Geometric Computing	2011-03-22 22:22:00	submission deadline	2011-04-22 22:22:00	review deadline	2011-05-22 22:22:00	0000-00-00 00:00:00	ipc deadline	2011-05-22 22:22:00	2011-06-22 22:22:00	final deadline	
Surface Meshing, Shape Modeling	0000-00-00 00:00:00		0000-00-00 00:00:00		0000-00-00 00:00:00	0000-00-00 00:00:00		0000-00-00 00:00:00	0000-00-00 00:00:00		
Physically Based Modeling	0000-00-00 00:00:00		0000-00-00 00:00:00		0000-00-00 00:00:00	0000-00-00 00:00:00		0000-00-00 00:00:00	0000-00-00 00:00:00		
Image Based Modeling	0000-00-00 00:00:00		0000-00-00 00:00:00		0000-00-00 00:00:00	0000-00-00 00:00:00		0000-00-00 00:00:00	0000-00-00 00:00:00		
Medical Visualization	0000-00-00 00:00:00		0000-00-00 00:00:00		0000-00-00 00:00:00	0000-00-00 00:00:00		0000-00-00 00:00:00	0000-00-00 00:00:00		
Scientific Visualization	0000-00-00 00:00:00		0000-00-00 00:00:00		0000-00-00 00:00:00	0000-00-00 00:00:00		0000-00-00 00:00:00	0000-00-00 00:00:00		
Computational Photography	0000-00-00 00:00:00		0000-00-00 00:00:00		0000-00-00 00:00:00	0000-00-00 00:00:00		0000-00-00 00:00:00	0000-00-00 00:00:00		
High Definition Range Images	0000-00-00 00:00:00		0000-00-00 00:00:00		0000-00-00 00:00:00	0000-00-00 00:00:00		0000-00-00 00:00:00	0000-00-00 00:00:00		

1 2 3 

Obrázek 17 - přehled workshopů

## 1.7.2 Editace

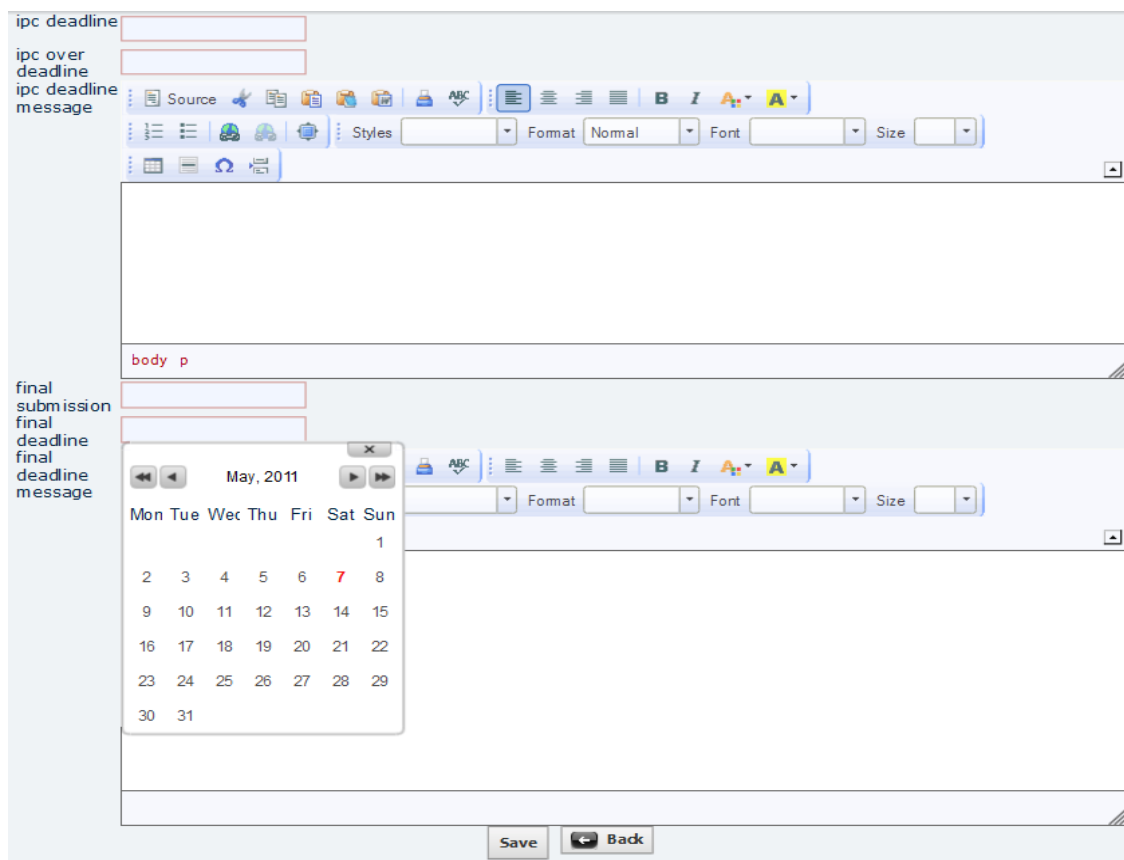
The screenshot shows the 'Workshop edit' interface. On the left, there are labels for the form fields: 'name', 'is conference', 'submission deadline', 'submission deadline message', 'review deadline', and 'review deadline message'. The 'name' field contains 'Testovací workshop'. The 'is conference' field has radio buttons for 'true' (selected) and 'false'. The 'submission deadline' field is empty, and a calendar widget is open over it, showing the month of May 2011 with the 5th selected. The 'submission deadline message' and 'review deadline message' fields are empty text areas with rich text editor toolbars. The 'review deadline' field is also empty.

Obrázek 18 - editace workshopu 1. část

V editaci workshopu je potřeba vyplnit, kromě názvu, i několik termínů uzávěrek částí, které jsou limitující pro práci s příspěvkem.

- název (name) – název workshopu / sekce
- uzávěrka přidávání příspěvků (submission deadline) – mezní termín, do kterého je možné přidávat nové příspěvky, nahrávat články a přílohy k nim. Tento termín je zavazující především pro autory. Datum a čas uzávěrky je možné zadat buď zadáním textové vepsání ve formátu RRRR-MM-DD hh:mm:ss, nebo kliknutím na textové pole myši. Zobrazí se kalendář a datum je možné zvolit pouhým kliknutím a vybráním příslušného data, čas se nastaví automaticky na 00:00:00. Čas je potom možné přepsat, dle potřeby.
- zpráva k této uzávěrce (submission deadline message) – zpráva, kterou je možné přidat k termínu uzávěrky
- termín uzávěrky revizí (review deadline) – mezní termín pro vytvoření a uložení recenze příspěvku
- zpráva k uzávěrce revizí (review deadline message) – zpráva k termínu revize





Obrázek 19 - editace workshopu 2. část






















- termín IPC (IPC deadline) – termín pro zkontrolování revizí a jejich rozhodnutí
- (IPC over deadline)
- (IPC deadline message) – zpráva k termínu vyjádření IPC
- termín uzávěrky nahrávání finální revize příspěvku (final submission) – termín, do kterého je potřeba nahrát zrevidovaný příspěvek
- zpráva k termínu posledních revizí (final deadline message)

## 1.8 Uživatelé

Uživatelské účty defaultně spravuje administrátor.

### 1.8.1 Přehled

Seznamu uživatelů nalezne v menu „nastavení→uživatelé“. S uživateli lze provádět následující úkony: přidávat (+), mazat (x), editovat (📄), prohlížet a měnit práva (👤), prohlížet výpis přihlášení (📄). V základním přehledu je zobrazeno: uživatelské jméno, celé jméno, stát uživatele, uživatelské dovednosti a v jakých skupinách je zařazen. Uživatelé můžeme rovněž zamezit přístup do systému deaktivováním (ON OFF) konta, nebo je aktivovat (OFF).

Users					
Username	Name [institution-department]	Country	Expert in	Groups	Action
admin	Bc. Tomáš Klásek [Czech republic]	Czech republic	<b>Geometric Computing (EPOCH_IPC)</b> (very confident), <b>Rendering Techniques (EPOCH_DECISION)</b> (excellent)	admin, authors, IPC	   <input checked="" type="checkbox"/> ON
br	tit frist last tit2 [Bahamas]	Bahamas			   <input checked="" type="checkbox"/> ON
brn	[]				   <input checked="" type="checkbox"/> ON
a	[]				   <input checked="" type="checkbox"/> ON
Tt	[]				   <input checked="" type="checkbox"/> ON
skala	Prof. Ing. Vaclav Skala CSc. [Czech republic]	Czech republic	<b>Rendering Techniques (EPOCH_DECISION)</b> (excellent), <b>Geometric Computing (EPOCH_IPC)</b> (confident)	admin, authors	   <input checked="" type="checkbox"/> ON
tomas@klasek.cz	Bc. Tomáš Klásek PhD : []			reviewers	   <input checked="" type="checkbox"/> ON

Obrázek 20 - přehled uživatelů

## 1.8.2 Náhled

User view	
Username	admin
Permission	admin
Shortname	
Longname	Bc. Tomáš Klásek
Email	tomas@klasek.cz
Titul 1	Bc.
Firstname	Tomáš
Surname	Klásek
Titul 2	
Institution	ZCU
Department	FAV
Web	http://klasek.cz
Phone	+420777667779f
Fax	+420777667779
Address	adresa
Zip	37004
City	Český Krumlov
Country	Czech republic
Sex	male
Dietary	non-restriction
Main session	GNU/Tesla/Fermi Programming (EPOCH_DECISION)
Sec session	Geometric Computing (EPOCH_IPC)
	

Obrázek 21 - náhled uživatele

### 1.8.3 Editace

Username: admin

Group:  (dropdown menu with options: none, admin, authors)

Titul1:

First name:

Last name:

Titul2:

Institution:

Department:

Email:

Confirm email:

Password:

Confirm password:

Phone:

Fax:

Vegetarian:  (dropdown menu)

Address:

ZIP code:

City:

Country:  (dropdown menu)

Homepage:

Main session:  (dropdown menu)

Secondary session:  (dropdown menu)

Comment:

Status:  (Unchecking disable this user.)

Obrázek 22 - editace uživatele

Administrátor má právo vidět a měnit veškeré uživatelské položky. Jedinou položkou, kterou nelze nikde a nikomu zobrazit je uživatelské heslo. Heslo lze měnit, uživatelem, či administrátorem. Položky formuláře se změnou údajů tvoří vesměs základní údaje o uživateli:

- uživatelské jméno (username) – tato položka je pouze informační a nelze ji měnit
- skupiny (group) – skupiny, ve kterých je uživatel členem

- titul před jménem (title1)
- křestní jméno (first name) – křestní jméno uživatele
- příjmení (last name) – příjmení uživatele
- titul za jménem (title2)
- instituce (institution) – instituce, za kterou uživatel vystupuje v systému
- oddělení (department) – oddělení instituce
- email a potvrzení (email, confirm email) – email je potřeba zadat 2x stejně pro kontrolu proti překlepům, na tento email budou uživateli odesílány notifikace systémem a v případě zapomenutí hesla i nově vygenerované
- telefon (phone)
- fax (fax)
- strava (vegetarian) – zde je na výběr, co uživatel jí a nejí, zda vše, ryby, je vegan nebo vegetarián, toto je použito pro zajištění stravy, pokud se uživatel zúčastní konference
- adresa (address) – ulice a č.p.
- PSČ (ZIP)
- město (city)
- země (country)
- webová stránka (homepage)
- hlavní sekce (main section) – hlavní předmět zájmu uživatele
- vedlejší sekce (secondary section) – vedlejší předmět zájmu uživatele
- komentář (comment) – zde o sobě může uživatel zanechat nějakou poznámku, či popis
- stav (status) – pokud je políčko zaškrtnuté, je uživatel aktivní. Pokud jej však sám uživatel odškrtně a uloží změny, nebude se moci do systému přihlásit.

## 1.8.4 Výpis logu

Zobrazení kdy a odkud se uživatel přihlásil a jaký při tom použil webový prohlížeč.

Log for Bc. Tomáš Klásek

Time	From	Browser
2011-05-05 12:52:52	147.228.181.195	Mozilla/5.0 (Windows NT 6.1; rv:2.0) Gecko/20100101 Firefox/4.0
2011-05-05 12:02:07	147.228.181.195	Mozilla/5.0 (Windows NT 6.1) AppleWebKit/534.30 (KHTML, like Gecko) Chrome/12.0.742.16 Safari/534.30
2011-05-04 19:07:40	147.228.181.195	Mozilla/5.0 (Windows NT 6.1) AppleWebKit/534.30 (KHTML, like Gecko) Chrome/12.0.742.16 Safari/534.30
2011-04-03 20:14:53	147.228.233.70	Mozilla/5.0 (Windows NT 6.1) AppleWebKit/534.27 (KHTML, like Gecko) Chrome/12.0.712.0 Safari/534.27
2011-03-31 16:32:44	81.201.60.169	Mozilla/5.0 (Windows NT 6.1; rv:2.0) Gecko/20100101 Firefox/4.0
2011-03-31 14:41:30	147.228.180.15	Mozilla/5.0 (Windows NT 6.1; rv:2.0) Gecko/20100101 Firefox/4.0
2011-03-30 19:52:21	147.228.233.240	Mozilla/5.0 (Windows NT 6.1; rv:2.0) Gecko/20100101 Firefox/4.0
2011-03-23 08:35:34	81.201.60.169	Mozilla/5.0 (Windows NT 6.1) AppleWebKit/534.24 (KHTML, like Gecko) Chrome/11.0.696.16 Safari/534.24
2011-03-21 22:13:44	81.201.60.169	Mozilla/4.0 (compatible; MSIE 6.0; Windows CE; IEMobile 7.6)
2011-03-21 10:44:42	147.228.233.100	Mozilla/5.0 (Windows NT 6.1) AppleWebKit/534.24 (KHTML, like Gecko) Chrome/11.0.696.14 Safari/534.24
2011-03-18 19:10:34	81.201.60.169	Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 6.1; Trident/4.0; GTB6.6; SLCC2; .NET CLR 2.0.50727; .NET CLR 3.5.30729; .NET CLR 3.0.30729; Media Center PC 6.0; Tablet PC 2.0; .NET4.0C)

Obrázek 23 - výpis přihlášení uživatelů

Další systémový záznam s významnými událostmi.

type	time	message	id_user
login	2011-05-05 17:32:07	User 1 logout	1
login	2011-05-05 17:32:16	User Bc- fist last PhD. [] sucessfullyloged	57
login	2011-05-05 17:40:14	User 57 logout	57
reg	2011-05-05 17:41:03	Email for new password sented to user Bc. Tomáš Klásek PhD : []	56
login	2011-05-05 17:41:23	User Bc. Tomáš Klásek PhD : [] sucessfully loged	56
login	2011-05-05 17:43:41	User Bc. Tomáš Klásek [Czech republic] sucessfully logged	1
login	2011-05-05 17:43:58	User 56 logout	56
login	2011-05-05 17:44:13	User tt3 [Bahamas] sucessfully logged	60
login	2011-05-05 20:22:26	User nobody logout	0
login	2011-05-05 20:22:38	User nobody logout	0
login	2011-05-07 11:00:09	User Bc. Tomáš Klásek [Czech republic] sucessfully logged	1
login	2011-05-07 15:27:52	User Bc. Tomáš Klásek [Czech republic] sucessfully logged	1
login	2011-05-08 09:34:48	User tt3 [Bahamas] sucessfully logged	60
login	2011-05-08 11:09:46	User tt3 [Bahamas] sucessfully logged	60
login	2011-05-08 11:12:51	User Bc. Tomáš Klásek [Czech republic] sucessfully logged	1
sub	2011-05-08 11:18:30	User tt3 [Bahamas] uploaded file DIP_titulni.pdf to paper tt3 [Bahamas]	60
sub	2011-05-08 11:18:30	User tt3 [Bahamas] uploaded file AndroidManifest.xml to paper tt3 [Bahamas]	60
sub	2011-05-08 11:18:30	User tt3 [Bahamas] uploaded file ZCUrootCAder to papertt3 [Bahamas]	60
sub	2011-05-08 11:18:30	User 60 uploaded file ConfSys_workflow.pdf to paper 60	60
sub	2011-05-08 11:21:55	User tt3 [Bahamas] uploaded file DIP_titulni.pdf to paper tt3 [Bahamas]	60
sub	2011-05-08 11:21:55	User tt3 [Bahamas] uploaded file AndroidManifest.xml to paper tt3 [Bahamas]	60
sub	2011-05-08 11:21:55	User tt3 [Bahamas] uploaded file ZCUrootCAder to papertt3 [Bahamas]	60
sub	2011-05-08 11:21:55	User 60 uploaded file ConfSys_workflow.pdf to paper 60	60

Obrázek 24 - záznam významných událostí systému

### 1.8.5 Nastavení práv uživatele

Perms for Bc. Tomáš Klásek Tomáš Klásek

id	Title	Read	Write	Delete
1	Conference Management System	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
2	Papers	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
3	Texty	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
9	dictionary	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
12	Menu	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
15	Review	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
18	IPC board	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
21	Hotel	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
24	No logged	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
25	Room type	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
28	Main review	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
31	Coauthors	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
34	User accomodation	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
37	User payment	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
40	User skills	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
43	User skill grades	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
47	User groups	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
50	Diet	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
117	Logged in	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
61	Review questions	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
62	Review questions edit	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
64	Assigned papers for main review new	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

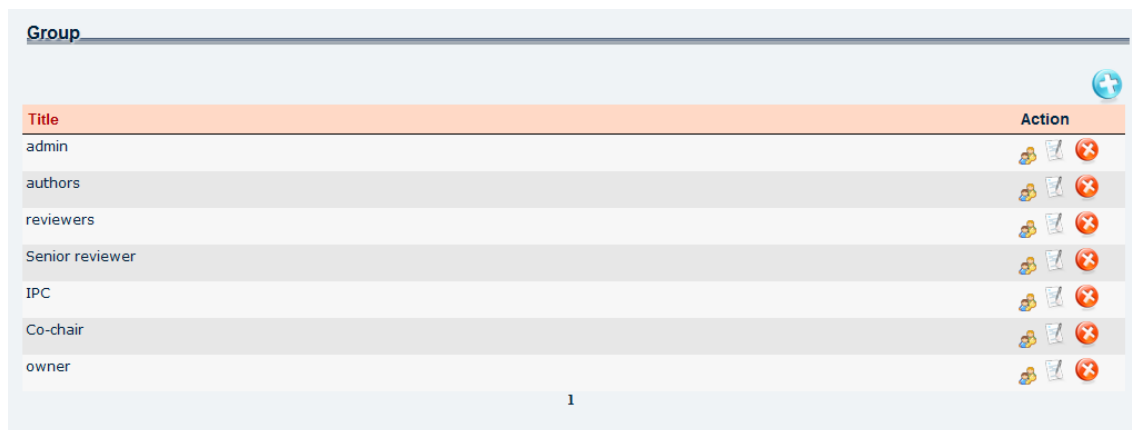
Obrázek 25 - nastavení práv uživatele

Práva uživatele jsou nastavována konkrétnímu uživateli, na rozdíl od skupinových práv. Jsou rozdělena pro každou položku na 3 části, které lze provádět. První je možnost čtení (zobrazení) dané položky (read), druhou je možnost zápisu (write) a uložení a třetí je možnost mazání (delete). Zaškrtnutí či odškrtnutí položky ve sloupci read má také vliv na zobrazení dané položky v menu.

### 1.9 Uživatelské skupiny

Uživatelé jsou zařazováni do skupin především kvůli správě práv pro celou množinu uživatelů najednou.

### 1.9.1 Přehled uživatelských skupin

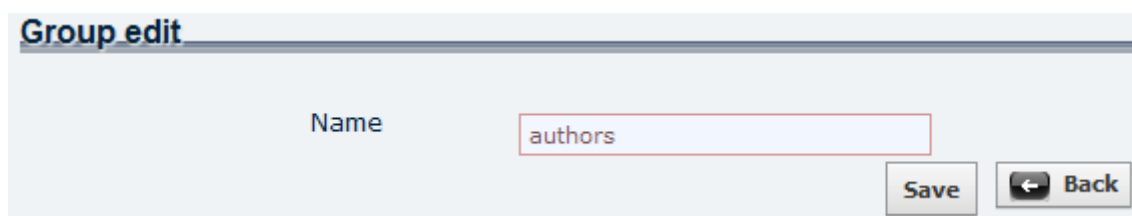


Title	Action
admin	
authors	
reviewers	
Senior reviewer	
IPC	
Co-chair	
owner	

1

Obrázek 26 - přehled skupin uživatelů

### 1.9.2 Editace



**Group edit**

Name

Obrázek 27 - editace suživatelské skupiny

- název (name) – textový popisek názvu skupiny

### 1.9.3 Nastavení práv skupiny



Group permission					
Perms for group admin					
id	Title	Read	Write	Delete	
1	Conference Management System	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
2	Papers	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
3	Texty	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
9	dictionary	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
12	Menu	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
15	Review	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
18	IPC board	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
21	Hotel	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	

Obrázek 28 - nastavení práv skupiny

Skupiny slouží především k rozdělení uživatelů a přidělení práv dle skupin, aby se nemusely přidělovat jednotlivým uživatelům zvlášť. Jsou rozdělena pro každou položku

na 3 části, které lze provádět. První je možnost čtení (zobrazení) dané položky (read), druhou je možnost zápisu (write) a uložení a třetí je možnost mazání (delete). Zaškrtnutí či odškrtnutí položky ve sloupci read má také vliv na zobrazení dané položky v menu.

## 1.10 Uživatelé a skupiny

Jednotliví uživatelé jsou rozděleni do skupin. Administrátor může uživatele libovolně přidávat a odebírat ze skupin.



### 1.10.1 Přehled

User groups			
Person	Group		Action
Bc. Tomáš Klásek [Czech republic]	admin		 
Prof. Ing. Vaclav Skala CSc. [Czech republic]	admin		 
t t3 [Bahamas]	authors		 
Bc. Tomáš Klásek [Czech republic]	authors		 
Bc- fist last PhD. []	Co-chair		 
Prof. Ing. Vaclav Skala CSc. [Czech republic]	authors		 
[]	reviewers		 
Bc- fist last PhD. []	reviewers		 
Bc. Tomáš Klásek [Czech republic]	IPC		 
t1 t7 šěš Phd [Czech republic]	authors		 

1 2 3 

Obrázek 29 - přehled uživatelů

### 1.10.2 Editace

User group edit	
Person	Tomáš Klásek [Czech republic] 
Group	authors 
<input type="button" value="Save"/> <input type="button" value="Back"/>	

Obrázek 30 - editace uživatele a skupiny

- uživatel (person)
- skupina (group)



Při editaci lze měnit jak uživatele, tak jeho skupinu. Při přidávání nového přiřazení do skupiny je tedy nutné vybrat jak uživatele (person), tak i cílovou skupinu (group).

## 1.11 Přiřazení příspěvků pro hlavní recenze

Assigned papers for main review															
id	sign	author	title	abstract	version	status	Major section	Minor section	Submit lock	Review lock	used in	pdf	supplement	copyright	Action
2	A02	Bc. Tomáš Klásek [Czech republic]	Title your Paper	neco japonsky マンガマガイト シトルバシノ マビ	3	submitted	GNU/Tesla/Fermi Programming (EPOCH_DECISION)	Rendering Techniques (EPOCH_DECISION)	unlocked		NULL	[418 Bytes]	[106.938 KB]	[106.938 KB]	
30		Bc. Tomáš Klásek [Czech republic]	japonsky paper	ヒゲイスイ コリチ ビンラ スズチ チラス チカ カカラシ ス体ニシナコ ンモラカラコ ナカヒ...	0	full paper	Geometric Computing (EPOCH_IPC)	Rendering Techniques (EPOCH_DECISION)		unlocked					

## 1.12 Přiřazení spoluautorů k příspěvkům

Primárně se spoluautoři dají spravovat i při editaci daného příspěvku. Administrátor však má k dispozici i celkový přehled s možností editace.

### 1.12.1 Přehled

Coauthors		
Author	Paper	Action
Bc. Tomáš Klásek [Czech republic]	17 -	
Prof. Ing. Vaclav Skala CSc. [Czech republic]	17 -	
Bc. Tomáš Klásek [Czech republic]	8 -	
Prof. Ing. Vaclav Skala CSc. [Czech republic]	8 -	
Prof. Ing. Vaclav Skala CSc. [Czech republic]	3 -	
Bc. Tomáš Klásek [Czech republic]	19 -	
Bc. Tomáš Klásek [Czech republic]	20 -	
Prof. Ing. Vaclav Skala CSc. [Czech republic]	21 -	
tit frist last tit2 [Bahamas]	21 -	
Prof. Ing. Vaclav Skala CSc. [Czech republic]	22 -	

1 2

### 1.12.2 Editace






User

Paper

- uživatel (user)
- příspěvek (paper)

## 1.13 Členové IPC

### 1.13.1 Přehled


IPC board							
Name [institution-department]	Email	Country	Groups	Expert in	Last login	Dietary	Action
Bc. Tomáš Klásek [Czech republic]	tomas@klasek.cz	Czech republic	admin, authors, IPC	<b>Geometric Computing (EPOCH_IPC)</b> (very confident), <b>Rendering Techniques (EPOCH_DECISION)</b> (excellent)	2011-05-05 12:52:52	non-restriction	    

### 1.13.2 Úvodní stránka po přihlášení -dole

**Payment list:**

Person	Currency	Arrived	Note	Conference	Received
No item...					

**Fields of your expertise :**  
(if you want be reviewer)

Person	Section	Grade
t3 [Bahamas]	Rendering Techniques (EPOCH_DECISION)	very confident 


[Add expertise](#)













[Remove me](#)

## 1.14 Diety

### 1.14.1 Přehled diet

**Diet**



Index	Name	Action
1	non-restriction	 
2	no-beef	 
3	no-pork	 
4	fish-only	 
5	vegetarian	 
6	vegan	 

1

### 1.14.2 Editace diet





















**Diet edit**

Name

- název (name) – název typu stravovacího návyku











## 1.15 Recenze

### 1.15.1 Přehled přiřazení recenzí k příspěvkům

Review assignment		
Person	Paper	Action
Prof. Ing. Vaclav Skala CSc. [Czech republic]	5 -	 
Bc- fist last PhD. []	4 -	 
Bc- fist last PhD. []	2 - A02	 
Bc- fist last PhD. []	7 - A02	 
Bc- fist last PhD. []	3 -	 
Bc- fist last PhD. []	5 -	 
t2 f l t2 []	4 -	 
Bc- fist last PhD. []	24 -	 
Bc. Tomáš Klásek PhD : []	3 -	 
Bc- fist last PhD. []	28 -	 

1

### 1.15.2 Přehled recenzí

Review								
Code paper	Cycle	Comment for author	Confidential comment	rating	status	Reviewer	Main review	Action
3 -	0	komentar review paper 3	confi comment	2.00	reviewed	Bc- fist last PhD. []	3 - com	
7 - A02	1			1.67	reviewed	Bc- fist last PhD. []		
4 -	2			0.00	assigned	Bc- fist last PhD. []		
5 -	1			0.00	assigned	Bc- fist last PhD. []		
4 -	2	a		3.67	reviewed	t2 f l t2 []		
24 -	0			0.00	assigned	Bc- fist last PhD. []		
3 -	1			0.00	assigned	Bc- fist last PhD. []		
2 - A02	3			0.00	assigned	Bc- fist last PhD. []		
3 -	1			0.00	assigned	Bc. Tomáš Klásek PhD : []		
28 -	0			0.00	assigned	Bc- fist last PhD. []		

1

### 1.15.3 Editace

Cycle: 0

Code paper: 2 - A02

Comment for author: komentar review paper 3

Confidential comment: confi comment

Rating: 2.00

Rating questions: Je to skutečne tak? 3 | Acceptable | short paper

Dalsi otazka k recenzii .. libi? :) 0 | Totally Unacceptable | reject



Reviewer Main review: Bc- fist last PhD. [] 3 - com

- komentář pro autora (comment for author)
- důvěrný komentář (confidential comment)
- otázky k recenzím a jejich hodnocení – viz následující podkapitola

### 1.16 Otázky k recenzím

Zobrazují se v recenzii příspěvku

#### 1.16.1 Přehled

Review questions		
Question	Strength	Action
Je to skutečne tak?	1	 
Dalsi otazka k recenzii .. libi? :)	2	 

1

## 1.16.2 Editace

**Review questions edit**

Question

Strength

- název (question) – název otázky
- váha (strength) – váha otázky

## 1.17 Hodnocení recenzí

Možné hodnoty hodnocení odpovědí na otázky v recenzi.

### 1.17.1 Přehled

**Review rating**

Id	Rating number	Text rating	Category	Describe	Action
1	0	Totally Unacceptable	reject	Strong Reject	 
2	1	Poor	reject	Must Be Rejected	 
3	2	Marginal	short paper	Might Be Accepted as Short Paper	 
4	3	Acceptable	short paper	Recommended as Short Paper	 
5	4	Good	full paper	Might Be Accepted as Full Paper	 
6	5	Very Good	full paper	Recommend as Full Paper	 
7	6	Excellent	full paper	Strongly Recommended as Full Paper	 

1

### 1.17.2 editace

**Review rating edit**

Rating number

Text rating

Category  ▼

Describe

- pořadí (rating number)
- textové hodnocení (text rating)
- kategorie (category)

- textový popisek hodnocení (describe)

## 1.18 Přijetí žádosti o možnost recenzovat

Administrátorovi se zobrazí přehled žádostí uživatelů (recenzentů) a je na něm, aby je potvrdil. Po potvrzení se budou zobrazovat v seznamu pro výběr recenzentů

User reviewer request accept				
Person	Section	Grade	Reviewer	Action
Prof. Ing. Vaclav Skala CSc. [Czech republic]	Rendering Techniques (EPOCH_DECISION)	excelent	false	
Bc. Tomáš Klásek [Czech republic]	Geometric Computing (EPOCH_IPC)	very confident	false	
Prof. Ing. Vaclav Skala CSc. [Czech republic]	Geometric Computing (EPOCH_IPC)	confident	false	
Bc. Tomáš Klásek [Czech republic]	Rendering Techniques (EPOCH_DECISION)	excelent	false	
t t3 [Bahamas]	Rendering Techniques (EPOCH_DECISION)	very confident	false	
[]	Rendering Techniques (EPOCH_DECISION)	excelent	false	
[]	Geometric Computing (EPOCH_IPC)	very confident	false	
[]	GNU/Tesla/Fermi Programming (EPOCH_DECISION)	little experien	false	
t1 t4 nic CsC. [Czech republic]	Rendering Techniques (EPOCH_DECISION)	excelent	false	

## 1.19 Stupně uživatelských odborných znalostí

Stupně odborných znalostí jsou využity uživateli pro kvalifikaci ve workshopech a pro recenze.

### 1.19.1 Přehled

User skill grades	
Name	Action
excelent	
very confident	
confident	
little experien	
no experience	

## 1.19.2 Editace

**User skill grade edit**

Name

- název (name) - Textový popisek výše odborné znalosti

## 1.20 Odborná kvalifikace uživatelů

### 1.20.1 Přehled

**User skills**

Person	Section	Grade	Action
Prof. Ing. Vaclav Skala CSc. [Czech republic]	Rendering Techniques (EPOCH_DECISION)	excelent	
t2 f1 t2 []	Rendering Techniques (EPOCH_DECISION)	little experien	
Bc. Tomáš Klásek [Czech republic]	Geometric Computing (EPOCH_IPC)	very confident	
Prof. Ing. Vaclav Skala CSc. [Czech republic]	Geometric Computing (EPOCH_IPC)	confident	
Bc. Tomáš Klásek [Czech republic]	Rendering Techniques (EPOCH_DECISION)	excelent	
t t3 [Bahamas]	Rendering Techniques (EPOCH_DECISION)	very confident	
[]	Rendering Techniques (EPOCH_DECISION)	excelent	
[]	Geometric Computing (EPOCH_IPC)	very confident	
[]	GNU/Tesla/Fermi Programming (EPOCH_DECISION)	little experien	
Bc- fist last PhD. []	GNU/Tesla/Fermi Programming (EPOCH_DECISION)	very confident	

1 2

### 1.20.2 Editace

**User skills edit**

User

Section

Grade

- uživatel (user)



- workshop (sekce)
- stupeň znalosti

## 1.21 Konferenční poplatky

Evidence poplatků spojených s konferencí. Tyto poplatky se zobrazují při registraci na konferenci.

### 1.21.1 Přehled

Conference Fee						
					10	+ Add new
Id	Note	Amount	Currency	Conference	Action	
1	Registration	120	Euro [€]	GraVisMa ( 2011-08-31 / Ostrava, Czech Republic )	 	
2	Per paper	10	Euro [€]	GraVisMa ( 2011-08-31 / Ostrava, Czech Republic )	 	

1

### 1.21.2 Editace

Conference Fee	
Note	<input type="text" value="Registration"/>
Amount	<input type="text" value="120"/>
Currency	<input type="text" value="Euro [€][1]"/> ▼
Conference	<input type="text" value="GraVisMa ( 2011-08-31 / Ostrava, Czech Republic ) [1]"/> ▼
<input type="button" value="Save"/> <input type="button" value="Back"/>	

- popis (note)
- částka (amount)
- měna (currency)
- konference (conference) – konference, které se poplatek týká

## 1.22 Měny

V systému existuje evidence měn, včetně jejich kurzů. Měny jsou používány pro konferenční poplatky a platby.

### 1.22.1 Přehled

Currency			
Des cription	Symbol	Rate	Action
1	Euro	€	1
2	Dolar US	\$	1.23

### 1.22.2 Editace

Currency	
Description	Euro
Symbol	€
Rate	1
<input type="button" value="Save"/> <input type="button" value="Back"/>	

- popis (description)
- symbol
- kurz (rate)

## 1.23 Platby uživatele

Přehled plateb uživatele, které se vážou na konferenční poplatky. Administrátor má přehled o všech poplatcích, které může rovněž editovat či přidávat. Platby jsou přímo vázány na konkrétní uživatele, registraci a poplatek.

### 1.23.1 Přehled

User pay							
Id	Amount	Note	Currency	Conference registration	Conference fee	User	Action
1	120		Euro [€]	Tomáš Klásek [Czech republic] - GraVisMa ( 2011-08-31 / Ostrava, Czech Republic )	Registration - 120	Bc. Tomáš Klásek [Czech republic]	
2	10	Paper A13	Euro [€]	Tomáš Klásek [Czech republic] - GraVisMa ( 2011-08-31 / Ostrava, Czech Republic )	Per paper - 10	Bc. Tomáš Klásek [Czech republic]	

### 1.23.2 Editace

**User pay**

Amount	<input type="text" value="10"/>
Note	<input type="text" value="Paper A13"/>
Currency	<input type="text" value="Euro [€][1]"/>
Conference registration	<input type="text" value="Tomáš Klásek [Czech republic] - GraVisMa ( 2011-08-31 / Ostrava, Czech Republic )[1]"/>
Conference fee	<input type="text" value="Per paper - 10 [2]"/>
User	<input type="text" value="Bc. Tomáš Klásek [Czech republic][1]"/>

- částka (amount)
- poznámka (note)
- měna (currency)
- registrace na konferenci (conference registration)
- dílčí konferenční poplatek (conference fee)
- uživatel (user)

### 1.24 Registrace na konferenci

Pokud se uživatel / osoba chce zúčastnit konference, je potřeba, aby se na ni přihlásil, případně zaplatil poplatky s tím související. K tomuto účelu má k dispozici každý uživatel možnost zaregistrovat se pomocí následujícího formuláře.

### Conference registration

**Conference**  
 Conference

**Personal data**  
*This data you can change in your profile.*  
 Person **Bc. Tomáš Klásek [Czech republic]**  
 E-mail [tomas@klasek.cz](mailto:tomas@klasek.cz)  
 Dietary non-restriction

**Organization address**  
 Institution **ZCU**  
 Department **FAV**  
 Address   
 Country   
 VAT:   
 SWIFT:   
 IBAN:

**Workshop**  
 Participate in this workshop:

**Fees**  
 Registration  ×  =   
 Per paper  ×  =   
 Sum

**Papers for conference**  
*Data about papers can be changed after send this registration.*  
 A11 - Title your Paper

Autorům je umožněno připojit prezentované příspěvky.



### 1.25 Registrace na konferenci – admin

Aministrátor má k dispozici přehled všech registrací, které může editovat, či ručně přidávat další.

### 1.25.1 Přehled

Conference registrations - all

10

	User	Invoice address	Swift	IBAN	VAT	Country	Conference	Action
1	Tomáš Klásek [Czech republic]	fakturacni adresa, Plzen zcu	444	114	CZ-49777513	Belgium	GraVisMa ( 2011-08-31 / Ostrava, Czech Republic )	 

1

### 1.25.2 Editace

Conference registrations - all

User: Tomáš Klásek [Czech republic][145]

Invoice address: fakturacni adresa, Plzen zcu

Swift: 123

IBAN: 456

VAT: 789

Country: Gambia[78]



Conference: GraVisMa ( 2011-08-31 / Ostrava, Czech Republic )[1]

- uživatel (User)
- fakturační adresa (Invoice address)
- swift
- iban
- DIČ (VAT)
- stát (Country)
- konference (conference)

### 1.26 Hotely a ubytovací zařízení

V systému je i možnost zadávání hotelů pro evidenci o ubytování účastníků.

### 1.26.1 Přehled

Hotel		
Index	Name	Action
1	hotel slavie	 

### 1.26.2 Editace



Hotel edit	
Name	<input type="text"/>
	<input type="button" value="Save"/> <input type="button" value="Back"/>

- název (name) – název ubytovacího zařízení

## 1.27 Platby uživatelů

Přehled došlých plateb od uživatelů.

### 1.27.1 Přehled

User payment					
Person	Currency	Arrived	Note	Conference	Action
Bc. Tomáš Klásek [Czech republic]	CZK	1500	Poplatek	GraVisMa ( 2011-08-31 / Ostrava, Czech Republic )	 
Prof. Ing. Vaclav Skala CSc. [Czech republic]	CZK	1500	testovací poplatek	GraVisMa ( 2011-08-31 / Ostrava, Czech Republic )	 

1

## 1.27.2 Editace

**User payment edit**

Person

Currency

Arrived

Note

Source **B** *I*

Styles  Format  Font  Size

Poplatek

Conference

- uživatel (person)
- měna (currency)
- obnos (arrived)
- poznámka (note)
- konference (conference)

## 1.28 Ubytování uživatelů

Uživatelé do systému mohou zadat odkdy, dokdy a kde budou ubytováni.

### 1.28.1 Přehled

**User accomodation**

Person	Arrival	Departure	Conference	Hotel	Room type	Stored	Action
tit frist last tit2 [Bahamas]	2010-09-08 00:00:00	2010-06-17 00:00:00	GraVisMa ( 2011-08-31 / Ostrava, Czech Republic )	hotel slavie	single	requested	
Bc. Tomáš Klásek [Czech republic]	2010-09-08 00:00:00	2010-06-17 00:00:00	GraVisMa ( 2011-08-31 / Ostrava, Czech Republic )	hotel slavie	single	requested	

1

## 1.28.2 Editace

**User accomodation edit**

Person	Bc. Tomáš Klásek [Czech republic]
Arrival	2010-09-08 00:00:00
Departure	2010-06-17 00:00:00
Conference	GraVisMa ( 2011-08-31 / Ostrava, Czech Republic )
Hotel	hotel slavie
Room type	single
Stored	<input checked="" type="radio"/> requested <input type="radio"/> booking <input type="radio"/> booked

Save Back

- uživatel (osoba)
- příjezd (arrival)
- odjezd (departure)
- konference (conference)
- hotel (hotel)
- typ pokoje (room type)
- fáze (stored) – požadováno/rezervace/rezervováno

**User accomodation edit**

Person	Bc. Tomáš Klásek [Czech republic]
Arrival	2010-09-08 00:00:00
Departure	2010-06-17 00:00:00
Conference	GraVisMa ( 2011-08-31 / Ostrava, Czech Republic )
Hotel	hotel slavie
Room type	single
Stored	<input checked="" type="radio"/> requested <input type="radio"/> booking <input type="radio"/> booked

Calendar: June, 2010

Mon	Tue	Wed	Thu	Fri	Sat	Sun
	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30				

Back

U datumových položek lze samozřejmě využít zobrazeného kalendáře.



## 1.29 Import

Aplikace umožňuje importovat dávku uživatelů a provádět s nimi základní operace, jako je přidání, smazání uživatelů, přiřazení recenzentů, nebo importovat rozhodnutí o příspěvcích.

The screenshot shows a web interface titled "Import CSV file". It contains four distinct sections, each with a header bar and a form below it:

- Add new users:** A header bar with the title, followed by a "Select file:" input field, a "SELECT" button with a folder icon, and an "Upload" button.
- Delete users:** A header bar with the title, followed by a "Select file:" input field, a "SELECT" button with a folder icon, and an "Upload" button.
- Assign reviews:** A header bar with the title, followed by a "Select file:" input field, a "SELECT" button with a folder icon, and an "Upload" button.
- Papers verdicts:** A header bar with the title, followed by a "Select file:" input field, a "SELECT" button with a folder icon, and an "Upload" button.

### 1.29.1 Import nových uživatelů

Zvláště při importu si musíme dát pozor na kódování importovaného souboru, jelikož aplikace a veškerá data jsou zpracovávána v UTF-8. Při importu uživatelů je v seznamu uvedeno jméno a příjmení, kde kódování hraje klíčovou roli pro správné interpretování národních znaků.

#### Formát pro přidání uživatelů

Importovaný soubor má být textový soubor formátu CSV. Aplikace používá UTF-8 kódování, proto při importu je potřeba si dát pozor na kódování souboru! Řádkovým vstupem bude právě 5 sloupců představující následující:

- |             |           |
|-------------|-----------|
| 1. jméno    | 4. email  |
| 2. příjmení | 5. funkce |
| 3. heslo    |           |

příklad:

```
Josef;Novák;heslo123;josef.novak@email.com;a
```

rozklíčování:

Řádek obsahuje uživatele jménem Josef Novák, emailem „josef.novak@email.com“, přihlašovat se bude heslem „heslo123“ a „a“ označuje autora

zkratky:

- a – autor
- r – recenzent
- d – IPC

### 1.29.2 Import seznamu uživatelů ke smazání

#### Formát pro smazání uživatelů

Importovaný soubor má být textový soubor formátu CSV. Řádkovým vstupem bude právě 1 sloupec představující následující:

1. email

příklad:

```
josef.novak@email.com
```

rozklíčování:

řádek obsahuje uživatele s identifikátorem a emailem „josef.novak@email.com“, ten je po importu vymazán z aplikace.

### 1.29.3 Import přiřazení recenzentů k příspěvkům

#### Formát pro přiřazení recenzentů

Importovaný soubor má být textový soubor formátu CSV. Řádkovým vstupem budou 3 a více sloupců představující následující:

1. email – email uživatele/recenzenta pro párování
2. počet přidělovaných článků
3. značka/y jednotlivých článků

příklad:

```
user@emaildomain.com;4;A11;A13;A17;A19
```

rozklíčování:

řádek obsahuje uživatele s emailem „user@emaildomain.com“, počet článků, a následuje/í identifikátory článku/ů.

### 1.29.4 Import rozhodnutí o příspěvcích

### Formát s rozhodnutím o člancích

Importovaný soubor má být textový soubor formátu CSV. Řádkovým vstupem budou právě 2 sloupce představující následující:

1. identifikátor článku

2. verdikt – zkratka

příklad:

A11 ; x

A13 ; a

rozklíčování:

Řádek obsahuje identifikátor článku a zkratku verdiktu. Konkrétně tedy článek s označením A11 není přijat (rejected), zato článek A13 je přijat (accepted)

zkratky:

- x – rejected
- z – cancelled
- j – journal
- f – full
- s – short
- p – poster
- t – tutorial
- c – communication

### 1.30 Export

Pro snazší stahování dat, aplikace umožňuje export a stažení dat, dle základních parametrů.



### 1.30.1 Export abstraktů

Vygenerován je HTML soubor, obsahující abstrakty uložených příspěvků dle jejich typu. Možné typy jsou *full paper*, *short paper* a *poster*.

### 1.30.2 Export finálních příspěvků

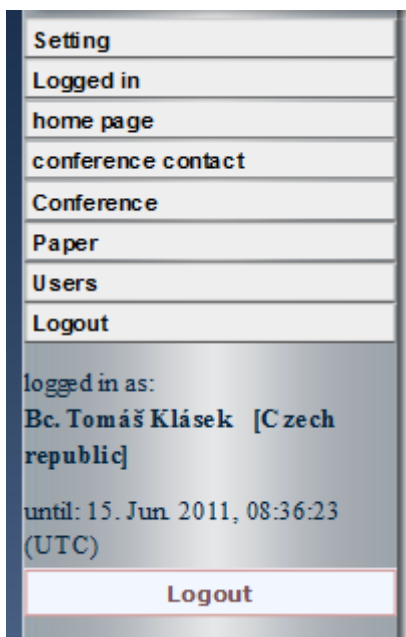
Finální příspěvky jsou uloženy do archivu (ZIP) včetně všech příložených souborů. Rovněž je vygenerována HTML stránka, obsahující základní přehled o příspěvku. Tento soubor lze generovat pro typy příspěvku *full paper*, *short paper* a *poster*.

### 1.30.3 Export recenzí

## 1.31 Systémový administrátor

Má přístup ke všem údajům a jejich podstatnou část může i kdykoliv měnit, nezávisle na stavu systému, etapách jednotlivých workshopů apod.

### 1.31.1 Menu



### 1.31.2 Menu

Slouží pro přidání a zpracování přístupu ke skriptovému souboru, který zároveň může být také položkou v menu.

### 1.31.3 Přehled položek menu

Menu

Index	Title	Order	File	Action
117	Logged in	0	file:logged_in.inc	
112	Product image delete	0	file:delete.php	
118	home page	0	href:http://wscg.zcu.cz	
119	conference contact	0	href:mailto:skala@kiv.zcu.cz	
120	Workshop edit	0	file:edit.php	
121	Logged in	0	file:logged_in.inc	
1	Setting	1	file:index.php	
15	Review	1	file:index.php	
122	Review delete	1	file:delete.php	
2	Paper	2	file:index.php	

1 2 3 4 5 6 7 8 9 10 >>

### 1.31.4 Editace položky menu

**Menu edit**

Menu

Site:

Subtitles:  
1. line 1. variable

Up menu:

1. line 2. variable

2. line

Activ:

Show as menu item

Method:  file  fileOnly  article  code  href

Article id:

Diractory:











File:

- název (site) – název položky
- nadpis (subtitle) – nadpis pro danou položku
- nadmenu (up menu) – zanoření položky v menu pod jinou, či na kořen
- aktivní (aktive)
- jako položka menu (show as menu item) – zda se má zobrazovat v menu
- metoda zpracování (method)
  - file – standardní zpracování, výstup vypsán do standardní šablony
  - fileOnly – standardní zpracování, avšak výstup pouze z tohoto souboru
  - article – zobrazí níže vybraný článek jako hlavní obsah stránky
  - code – zpracování např. pouze jedné metody třídy (využití pro JavaScript)
  - href – položka menu jako libovolný odkaz
- text id (article id) – číslo textu, který se zobrazí
- adresář (direktory) – adresář, kde se nachází soubor
- soubor (file) – název souboru, či skriptu

## 1.32 Texty

Potřebujeme-li přidat delší text do webové stránky, lze to udělat pomocí těchto textů. Výhody: lze spravovat přes webové rozhraní, podpora více jazyků. Nevýhody: potřeba přidat do šablony proměnou, kam se text zobrazí a do aplikačního kódu příslušné naplnění této proměnné. Viz programátorský manuál kapitola šablony 3.4.1.

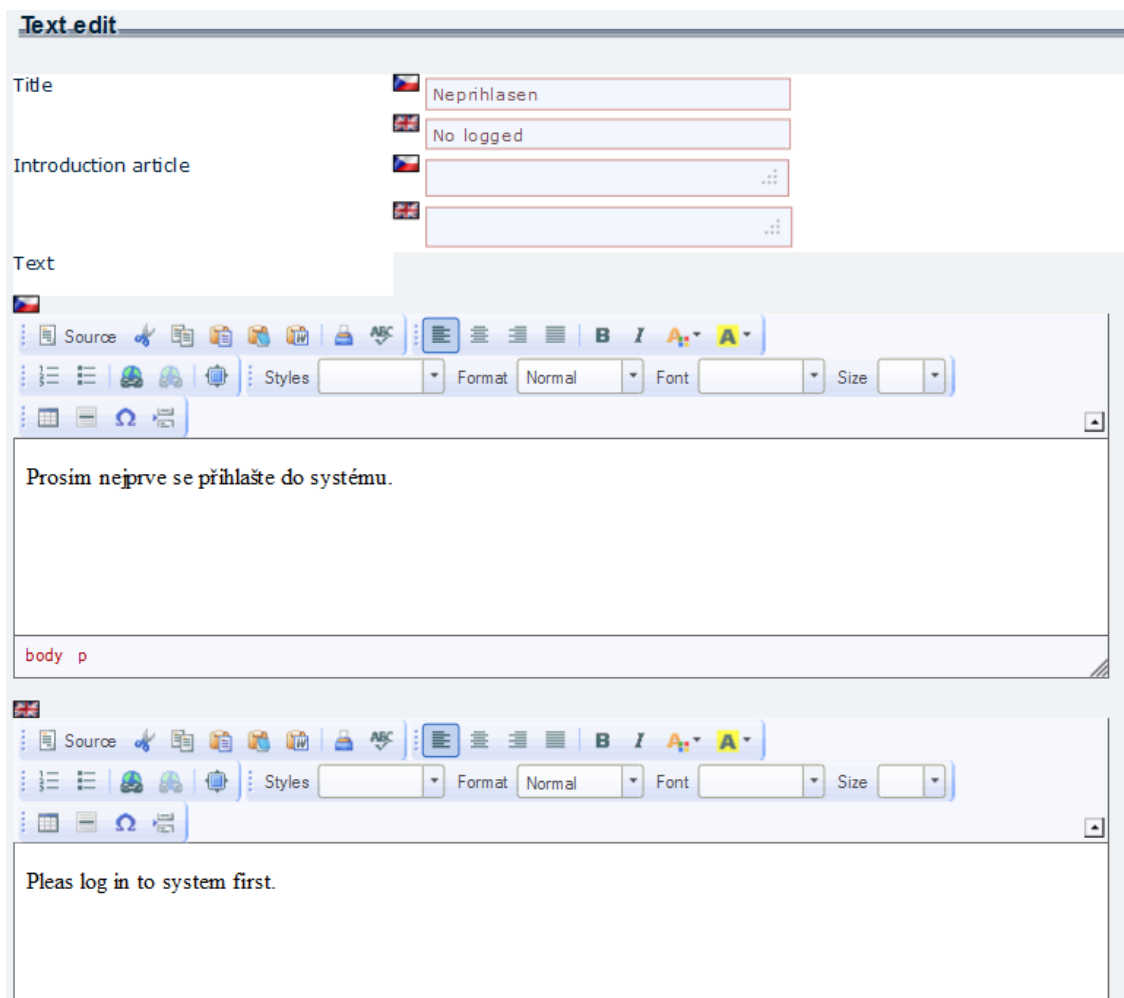
### 1.32.1 Přehled

Texty				
Title	Author	Edited	Published	Action
Napište nám	tomas	admin	07.03.2009 00:16	 
Nepřihlasen	tomas	admin	15.02.2011 08:36	 
O nás	tomas	admin	16.01.2011 20:03	 
Papers	tomas	admin	08.02.2011 07:30	 
Přihlášen	tomas	admin	14.02.2011 22:18	 

1

### 1.32.2 Editace

Texty se zadávají pro každý povolený jazyk pro překlady.



- titulek (title)
- úvodní odstavec (introduction article)
- článek (text)

### 1.33 Slovník

Aplikaci je možné provozovat v několika jazykových mutacích. Jazyky však musí být povoleny či přidány v systému. Pro všechny povolené jazyky je v databázi vytvořena kopie základních termínů pro překlad. Pokud je přidán (použit) do aplikace nový termín je automaticky přidán do databáze a do doby, než je přeložen je možné jej označit.



### 1.33.1 Přehled textových termínů a jejich překladu

dictionary

Original Translated Language

All  dean filter

Index	Original	Translated	Language	Action
34	Please allow JavaScript	Prosím povolte zpracování JavaScriptu nebo použijte prohlížeč, který jej umožňuje zpracovávat.	Czech	
31	Print	Print	Czech	
35	Please allow JavaScript	Please allow JavaScript!	English	
1	O nás	O nás	Czech	
13	Napište nám	Napište nám	Czech	
29	Top	Nahoru	English	
16	Kontaktní osoby	Kontaktní osoby	Czech	
10	Individuální program	Individuální program	Czech	
11	Individuální program	Individual Offer	English	
26	impressum	impressum	English	

← 1 2 3 4 5 6 7 8 9 10 → >>

### 1.33.2 Editace překladu

Dictionary

Original  
Please allow JavaScript

Translation

- překlad (translation)

## 2 Instalační manuál

### 2.1 Systémové požadavky

- http server
- podpora zpracování PHP verze 5.0 a vyšší
- relační databázový systém MySQL 5.0 a vyšší

### 2.2 Instalace aplikace - souborový systém

1. instalace aplikace probíhá rozbalením a nakopírováním skriptů a adresářové struktury na webový server.
2. změnit uživatelská práva pro zápis na adresáře, do kterých budou nahrávány soubory pomocí této aplikace

adresáře:

```
+ uploaded
    + paper
      +files
        +export
```

3. upravit konfigurační soubory v adresáři config

`config.database.php` – zadat data potřebná pro připojení k databázi

`config.path.php` – pokud jsme zvolili jiné než standardní složky, je třeba nastavit správné cesty

`config.constants.php` – zde se nacházejí volitelné parametry, kterými můžeme ovlivňovat některé základní funkce, např. konstanty:

`SITE_DEFAULT_LOGGED_IN_ID` - první stránka, která se zobrazí po přihlášení (id odpovídá id menu)

`SITE_DEFAULT_ANONYMOUS_ID` - první stránka, která se zobrazí bez přihlášení (id odpovídá id menu)

Dalších konfiguračních souborů `config.XXX.php` obsahují další základní nastavení, které je zřejmé z jejich názvu.

4. pokud se aplikace nachází na serveru v uživatelském adresáři a přístup je tedy přes tildu (např. ~novak/web), je potřeba změnit v adresáři ./web/ soubor .htaccess a na druhém řádku odkomentovat # RewriteBase / a změnit na RewriteBase /~novak/web a
5. vrátit práva kořenového adresáře na původní práva

### 2.3 Instalace struktury a dat do databáze

Před instalací musíme mít přístupová práva k SŘBD a znát název hostitele (většinou localhost). Mezi zdrojovými soubory se nachází i soubor install.sql, obsahující kompletní databázovou strukturu s úvodními daty. Tento soubor je potřeba naimportovat pomocí libovolného nástroje pro správu SŘBD. Zde se jedná o databázový systém MySQL u kterého se nejčastěji používá webová aplikace PhpMyAdmin [20].

### 2.4 Nastavení konference

1. defaultní jméno je admin a heslo admin
2. po přihlášení je pro správné fungování aplikace potřeba nastavit minimálně email a url adresu konference. Toto nastavení se nachází v menu Conference a podmenu Conference edit

Tímto pokud jste provedli všechny kroky instalace správně, měla by aplikace nyní bezchybně fungovat.

Pokud potřebujete podporu, nebo se vyskytly potíže, neváhejte kontaktovat autora [tomas@klasek.cz](mailto:tomas@klasek.cz).

## 3 Programátorský manuál

### 3.1 Základ aplikace

Jádro aplikace tvoří adresář `phplib`, kde jsou nejnütnější knihovny pro aplikaci. Dále je nutný konfigurační soubor (`config/config.php`). Práci se SŘBD umožňuje třída `cyDb`, nacházející se v adresáři `cylibs`.

#### 3.1.1 Abstraktní třída `abClass`

Od této třídy (`abClass`) se dědí (`extends`) všechny třídy modulu. Implementuje správu proměnných, dolování a ukládání dat do SŘBD. Rovněž implementuje základní metody, které je možno v potomcích přepsat, pro základní nakládání s daty. Význačné metody jsou například:

```
get () – získá dat z SŘBD
store () – uložení dat do SŘBD
getAll () – pro získá všech dat třídy,
getSelect () – navrácí HTML komponentu SELECT naplněnou daty
__construct () - defaultní konstruktor,
__toString () – defaultní výpis o třídě,
getValue () – vrací hodnotu proměnné
setValue () – vložení hodnoty do proměnné
...
vybrané privátní metody:
dbInit () – inicializace databáze
__get () – upravený getter proměnných
__set () – upravený setter proměnných
...
```

#### 3.1.2 Vzorová třída

Pro názornost nejprve ukázka např. třídy pro uložení informace o uživateli ve skupině:

```
class cyGroupUser extends abClass{
    public static $class_name = "cyGroupUser" ;
    public static $databaseName = "cyPermissionGroupUsers";
    protected $data=array(
        databaseName=>"cyPermissionGroupUsers",
```

```

        className => "cyGroupUser");
public $vars = array(
    group_id => array(
        type=>'object',
        className => 'cyGroup',
        getter=> "getSelect",
        args=>array("group_id","object")),
    user_id => array(
        type=>'object',
        className => 'cyUser',
        getter=> "getSelect",
        args=>array("user_id","object"))
);
/*přetížená metoda __toString() */
function __toString() {
    $this->checkAndGet();//dolování dat, pokud ještě nejsou
    return $this->getValue('group_id')
        ." - ".$this->getValue('user_id');
}
}

```

Ve třídách je vlastní správa proměnných. Základem jsou v ní dvě význačné proměnné \$data a \$vars. V první zmiňované (\$data) jsou uloženy do pole základní informacemi o třídě. Druhou tvoří proměnné, do kterých se ukládají hodnoty konkrétní třídy. Jedná se opět o pole. Zvláštností zde je, že zde můžeme používat typ object. Následuje className udávající typ objektu. Vytvoříme-li následně instanci třídy cyGroupUser, a chceme získat objekt skupiny, ve které se uživatel nachází, museli bychom v případě standardní práce ze třídu vykonat tento kód:

```

class cyUserGroup() {public group_id; ...}
$objekt = new cyUserGroup(1); // - ID je 1
$instanceSkupiny = new cyGroup($objekt->group_id);

```

V této aplikaci, díky vlastnímu zpracování proměnných, lze celý kód pro získání instance třídy skupiny zjednodušit takto (viz předchozí ukázka třídy):

```

$objekt = new cyUserGroup(1);
$instanceSkupiny = $objekt->group_id; //možná by se hodilo
lepší pojmenování proměnné na pouhé group → ...
$objekt->group

```

Typy proměnných, které jsou použitelné a rozlišované, jsou následující:

object (className, popř getter a args), date, datetime, text, varchar, int, enum (set=>array(položky enum,...))...

### 3.1.3 Objektově relační mapování databáze

Pro uchovávání dat je aplikace úzce spojena s databází a její strukturou. Pro snazší provádění změn a práci s proměnnými, byl vyvinut mechanismus mapující proměnné databáze na proměnné tříd. Aplikace rovněž mapuje nejen proměnné, ale i vztahy mezi nimi, a to pomocí cizích klíčů databáze. Proměnná (atribut databázové entity) odkazující pomocí cizího klíče na jiný objekt (entitu) je označena jako typ objekt a je s ním zacházeno jako s objektem.

Tento mechanismus mapování má na starost metoda `mapSQLVars` ve třídě `abClass`:

```
#nacteni sloupce = promennych
$this->Database->array_query($sloupce,
    "SHOW FULL COLUMNS FROM ".$this->databaseName.";");

#nacteni cizich klicu -> pro dany sloupec(promennou) to
znamena, ze se s ni bude pracovat jako s objektem
$this->Database->array_query($klice,
    "SELECT table_name,column_name,
    `referenced_table_name`,
    `referenced_column_name`
    FROM information_schema.KEY_COLUMN_USAGE
    WHERE TABLE_SCHEMA = '".DATABASE_NAME.'"
    AND TABLE_NAME = '". $this->databaseName.'"
    AND REFERENCED_TABLE_NAME IS NOT NULL;"
);
#projdeme vsechny nactene sloupce
foreach ($sloupce as $k => $sloupec){
    $isObject = false;
    #pokud je cizim klicem - odkaz na novy objekt
    if (is_array($klice)) foreach($klice as $kk => $vv){
        if ($vv["column_name"]==$sloupec["Field"]){
            $isObject =true;
            break;
        }else $isObject =false;
    }
}
#namapovani atributu na promenne
#typ promenne
(!$isObject)?
    preg_match("/([a-z0-9]+)\ ([0-9]+\ )$/",
        $sloupec["Type"],$matches):null;
(!$isObject&&empty($matches[1]))?
    preg_match("/([a-z0-9]+)$/",$sloupec["Type"],$matches):null;
$this->vars[$sloupec["Field"]]["type" ]=
    ($isObject)?"object":$matches[1];
```



### 3.1.4 Třída – číselník

Třída nemusí fyzicky existovat, tj. nemusí být vytvořen žádný soubor. Stačí vytvořit v databázi tabulku s příslušným názvem, který se jmenuje stejně jako požadovaná třída. Systém, pokud takovouto třídu nenalezne fyzicky v souborovém systému, vytvoří její definici „za běhu“. K této proceduře slouží metoda `__autoload($className)`:

```
function __autoload($class_name){
    $directorys = $GLOBALS['cfg']['class']['classdir'];
    foreach($directorys as $directory){//for each directory
        //see if the file exists
        if(file_exists($directory.$class_name . '.php')){
            require_once($directory.$class_name . '.php');
            return;
        }
    }
    //pokud trida neexistuje vytvořime její definici
    if (!class_exists($class_name)) {
        eval ("
        class $class_name extends abClass{
            public static \$class_name = \"$class_name\";
            public static \$databaseName = \"$class_name\";
            protected \$data=array(
                databaseName=>\"$class_name\",
                className=>\"$class_name\"
            );
            protected \$vars=array(); #promenna pro promenne

            function __toString(){
                return \$this->title.\" \";
            }
        }
        ");
    }
}
```

Jelikož dědí veškeré vlastnosti abstraktní třídy `abClass`, můžeme s ní ihned pracovat jako s objektem. Aplikace mapuje proměnné a cizí klíče z databázové tabulky. U proměnných mají v definici cizí klíč, které je nastaven příznak, že se jedná o proměnnou typu objekt (odkazuje na jiný objekt). Předchozí umožňuje snadno vytvářet netriviální objekty.

Omezujícím prvkem je zde také třída `abClass`, která sice obsahuje základní metody pro práci s daty, ale potřebujeme-li vlastní použití metody třídy, nezbyvá nic jiného, než třídu vytvořit fyzicky v souborovém systému a metodu do ní implementovat.

Práci s daty takovéto třídy nám umožňuje podpora menu, kde stačí vytvořit příslušnou položku a jako typ zpracování se uvede `class` a název třídy (`entity`). Aplikace pak s daty pracuje pomocí základních universálních souborů, kterými jsou:



- `uni_call.php` – vstupní soubor – umožňující výpis, editaci, mazání a náhled
- `uni_index.php` – výpis všech záznamů
- `uni_edit.php` – editace konkrétního záznamu
- `uni_delete.php` – smazání jednoho záznamu
- `uni_view.php` – náhled jednoho záznamu

### 3.2 Přidání modulu

Modulem rozumějme část aplikace, mající na starost vybranou oblast. Pro přidání je nutné provést několik základních kroků:

1. uložení skriptů na server
2. přidání příslušných položek menu pomocí webového rozhraní
3. použité třídy nahrát do adresáře `cylibs`
4. případně vytvořit databázovou strukturu a data potřebná pro modul

**Příklad:** modul pro překládání (language)

1. potřebné skripty: `index.php`, `edit.php`, `delete.php` nahrát na web
2. na webu: `setting>menu` . přidat pro každý zpracovatelný skript položku:  
`index.php` – název: překladač, jako menu: ano, adresář: `language`, soubor:  
`index.php`  
`edit.php` – název: editace překladu, jako menu: ne, adresář: `language`,  
soubor: `edit.php`  
`delete.php` – název: smazání překladu, jako menu: ne, adresář:  
`language`, soubor: `delete.php`
3. třídu `cyLanguage` nahrát do adresáře `phplib/cylibs`
4. vytvořit příslušnou DB strukturu: vytvoření DB tabulek `cyLanguage` – seznam jazyků a `language_words` – seznam překladů
5. tvoření odkazů -> `text.(ID-položky-menu).jazyk`, příklad: `seznam-prekladu.154.2` (`text:seznam-prekladu, id:154,jazyk:2`) – id položky je závislé na položce

### 3.3 Systém řízení báze dat

Aplikace je vyvinutá na MySQL, pokud bychom chtěli aplikaci provozovat na jiném systému, stačí zaměnit, či příslušně upravit třídu `cyDb`. Ta jako jediná komunikuje s databázovým systémem.

Ve třídách děděných od třídy `abClass` se používá pro přístup ke třídě `cyDb` proměnná `$this->Database` a následně se již volají funkce této třídy.

Například takto pro získání výsledku rovnou do pole `$vysledek`:

```
$this->Database->array_query($vysledek, "SELECT * FROM db_table");
```

### 3.4 Šablony a vzhled

#### 3.4.1 Šablony

V této aplikaci je maximálně použito šablon pro oddělení vzhledu od aplikační úrovně. Šablony se nacházejí, až na výjimky, v adresáři `tpl` u každého modulu. Základní šablony jsou uloženy v kořenovém adresáři aplikace v adresáři s názvem rovněž `tpl`.

Struktura šablony je totožná s klasickou webovou stránkou s jediným rozdílem, že hlavičku a tělo tvoří pouze základní šablona stránky `index.tpl`. Ostatní šablony tvoří částečné fragmenty HTML stránky. Šablony však mohou obsahovat i PHP skripty a jejich umístění zde je spíše výjimečné, kdyby nešlo použít jiné volby, nebo by znamenalo výrazné usnadnění problému.

O plnění šablon daty a jejich zpracování se stará třída `Template`. Parametry konstruktoru je jí předán adresář se souborem šablony a druhým parametrem je typ zpracování proměnné, které nebyla přiřazena žádná data. Možnostmi je: nechat zanechat a vypsat název proměnné (`keep`), odebrat (`remove`), anebo ji zakomentovat (`comment`).

Šablony obsahují tzv. proměnné, do kterých jsou aplikační vrstvou vkládána data. Vložení probíhá vyhledáním patřičné proměnné v šabloně a jejím nahrazením daty aplikace. Struktura proměnné šablony je tvořena složenými závorkami a mezi nimi název – `{variable}`.

Vkládání více dat za jednu proměnnou, je umožněno tzv. bloky. Viz:

```
<!-- BEGIN blok_promennych -->
{promenna}
<!-- END blok_promennych -->
```

Příklad použití třídy Template:

```
$tmplakce = new Template($_path["admin_tpl"], "nbsp");
$tmplakce->set_file("page", "uni_index_action.tpl");
$tmplakce->set_block("page", "akce", "akces");
foreach($akce as $k => $v ){
    $tmplakce->set_var(array(
        "URL"=> $sess->url ("{$k.{$v[id]}.{$ln?id={$sca[$i]}-
            >Id()}),
        "Action"    => $lan->Translate($v[title]),
        "Action_value" => $actionTitle,
        "URL_img"    => $v[imgUrl],
        $tmplakce->parse("akces", "akce", true);
    }
}
$tmplakce->parse("out", "page");
echo $tmplakce->getphp("out");
```

Příklad šablony pro předchozí příklad uni\_index\_action.tpl:

```
<!-- BEGIN akce -->
<a href="{URL}"
    onClick="
        return confirmLink(this, '{Action}:
        {Action_value}')">
    
</a>
<!-- END akce -->
```

### 3.4.2 Vzhled

Základ celkového vzhledu lze měnit pomocí kaskádových stylů. Kaskádové styly jsou uloženy v adresáři `css` nacházející se v kořenovém adresáři aplikace. Základem jsou tyto tři základní soubory:

`layout.css` - rozložení elementů hlavní stránky, nastavení jejich velikosti a okrajů

`style.css` - informace o vzhledu, okrajích a dalších nastavení elementů HTML

`menu.css` - informace o vzhledu menu

Hlavním stavebním prvkem stránky je šablona `index.tpl`, který tvoří základ celé webové aplikace. V těle jsou významné elementy:

`<div id="contentHolder">` - obsahuje celou webovou stránku

`<div id="menu">` - obsahuje menu tvořené seznamem `<ul>` a prvky `<li>`

`<div id="content">` - obsahuje veškerý zbylý obsah webové stránky

```
<h1 id="nadpis"> - nadpis obsahu
```

Tělo základu webové aplikace vypadá následovně:

```
<body>
  <div id="contentHolder">
    <div id="top">
      <div id="logo" />
      <h1 id='main_title'>{main_title}</h1>
      <div id="jazyky">{lngs}</div>
    </div>
    <div id="menu">{menu}</div>
    <div id='content'>
      <h1 id='nadpis'>{textTitle}</h1>
      {text}
    </div>
  </div>
</body>
```

### 3.5 Databázové úložiště

Zde se budeme věnovat především významným tabulkám, nutným pro minimální funkčnost samotného systému. Tuto skupinu tvoří tedy následující:

`active_session` – uloženy aktivní session a čas

`cyUser` – uživatelská tabulka

tvořící základní rozšíření (+ modul `language`):

`csCmsMenu` – položky menu, lze

`cyLanguage` – seznam jazykových mutací

`language_words` – termíny k překladu a jejich překlady

## 4 ER model

