

Sem vložte zadání Vaší práce.

ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE
FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
KATEDRA SOFTWAREVÉHO INŽENÝRSTVÍ



Bakalářská práce

Nová verze databáze konferencí

Vít Zdrubecký

Vedoucí práce: Ing. Petr Fišer, Ph. D.

21. prosince 2012

Poděkování

Doplňte, máte-li komu a za co děkovat. V opačném případě úplně odstraňte tento příkaz.

Prohlášení

Prohlašuji, že jsem předloženou práci vypracoval samostatně a že jsem uvedl veškeré použité informační zdroje v souladu s Metodickým pokynem o etické přípravě vysokoškolských závěrečných prací.

Beru na vědomí, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorského zákona, ve znění pozdějších předpisů, zejména skutečnost, že České vysoké učení technické v Praze má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle § 60 odst. 1 autorského zákona.

V Praze dne 21. prosince 2012

.....

České vysoké učení technické v Praze
Fakulta informačních technologií

© 2012 Vít Zdrubecký. Všechna práva vyhrazena.

Tato práce vznikla jako školní dílo na Českém vysokém učení technickém v Praze, Fakultě informačních technologií. Práce je chráněna právními předpisy a mezinárodními úmluvami o právu autorském a právech souvisejících s právem autorským. K jejímu užití, s výjimkou bezúplatných zákonných licencí, je nezbytný souhlas autora.

Odkaz na tuto práci

Zdrubecký, Vít. *Nová verze databáze konferencí*. Bakalářská práce. Praha: České vysoké učení technické v Praze, Fakulta informačních technologií, 2012.

Abstract

Sem doplňte ekvivalent abstraktu Vaší práce v angličtině.

Keywords Nahradte seznamem klíčových slov v angličtině oddělených čárkou.

Abstrakt

V několika větách shrňte obsah a přínos této práce v češtině. Po přečtení abstraktu by se čtenář měl mít čtenář dost informací pro rozhodnutí, zda chce Vaši práci číst.

Klíčová slova Nahradte seznamem klíčových slov v češtině oddělených čárkou.

Obsah

Úvod	1
1 Požadavky a cíle	3
1.1 Barevná indikace dat	3
1.2 Změny v archivu	3
1.3 Obnovení konference	4
1.4 Skupiny	4
1.5 Rozšířená kategorizace	5
1.6 Příspěvky ke konferencím	5
1.7 Oprava chyb z předešlé verze aplikace	6
2 Analýza a návrh	7
2.1 Použité technologie	7
2.2 Skupiny	10
3 Realizace	11
3.1 Barevná indikace dat	11
3.2 Změny v archivu	11
3.3 Obnovení konference	13
3.4 Uživatelské kategorie	13
3.5 Skupiny	13
3.6 Filtr skupin a kategorií	15
3.7 Příspěvky ke konferencím	17
3.8 Oprava chyb z předešlé verze aplikace	17
3.9 Další úpravy a vylepšení	18
3.10 Databáze	21
3.11 Bezpečnost	21

4 Testování	25
4.1 Validita XHTML a CSS	25
4.2 Internetové prohlížeče	25
4.3 Servery	26
Závěr	27
A Seznam použitých zkratk	29
B Obsah příloženého CD	31

Seznam obrázků

2.1	Use case diagram skupin	10
3.1	ER diagram struktury databázen	22

Seznam tabulek

3.1	Nové sloupce v databázi	23
-----	-----------------------------------	----

Úvod

Databáze konferencí je dlouhodobě vyvíjený dynamický webový informační systém, sloužící primárně k evidenci konferencí a jejich správě. Během několikaletého používání a vývoje se v něm postupně objevily další důležité funkce a vlastnosti, jako je komunikace mezi uživateli pomocí fóra a emailu, archivování konferencí a jejich kategorizace.

První verze systému byla vytvořena Bc. Janem Kemrem v roce 2006 a umístěna na serveru Katedry počítačů FEL na ČVUT. Bylo implementováno ukládání konferencí a jejich sdílení mezi uživateli aplikace, vyhledávání, diskusní fórum a další prvky. O dva roky později v započaté práci pokračoval Ing. Petr Galík. Došlo k rozšíření aplikace o ukládání konferencí do databáze a jejich následné filtrování podle požadavků jednotlivých uživatelů na základě témat, ukládání do archivu a jiné.

V roce 2011 se systém dočkal třetí verze díky Bc. Jakubu Římalovi. Ten rozšířil možnosti kategorizace, zlepšil uživatelské rozhraní, opravil řadu chyb a přidal nové vlastnosti.

Tato bakalářská práce má za úkol navázat tam, kde kolegové skončili. To znamená zavést další funkce, podporující a usnadňující pohyb uživatelů v aplikaci, rozšířit a případně vylepšit dosavadní možnosti, a v neposlední řadě opravit chyby a nedostatky předchozí verze.

Požadavky a cíle

1.1 Barevná indikace dat

Už dříve bylo uživatelům umožněno nastavit si u libovolné konference, zda-li si chtějí nechat posílat emaily v závislosti na blížících se datech spojených s danou konferencí. To se mimo jiné vztahuje na začátek konání konference, na konečný termín pro odeslání příspěvků, rozesílání rozhodnutí o schválení příspěvků a datum, do kterého se musí vyhotovit finální verze příspěvků. Uživatel si pro tyto události stanoví vlastní časový předstih (nebo si ponechá ten původní) a na jeho základě je poté vyrozuměn.

Tuto činnost má na starost skript v souboru `naplanovat.php`. Ten je spouštěn periodicky, prohlíží databázi a porovnává termíny konferencí s uživatelskými předstihy a současným datem. Nakonec se postará o rozeslání emailů.

Vzhledem k možné četnosti a periodicitě, se kterou uživatel takové emaily dostává, bylo navrženo, aby se některé důležité termíny daly zvýrazňovat už v samotné aplikaci a byly tak okamžitě a snadno viditelné. Daný předstih, se kterým se mají data porovnávat, je jen jeden a platí tedy hromadně pro všechny konference.

1.2 Změny v archivu

Ukončené konference se přesouvají do archivu. Dříve se tato akce prováděla ručně, ale nyní se o to automaticky stará skript v souboru `naplanovat.php`. Vzhledem k dlouhodobému fungování této aplikace se počet archivních konferencí přirozeně rozrostl a vyhledávání v nich, potažmo jejich prohlížení, se může snadno stát časově náročným úkonem.

1. POŽADAVKY A CÍLE

V archivu funguje stejně jako v seznamu aktivních konferencí filtr témat, vyhledávání podle názvu i stránkování. Tyto funkce jsou ale mnohdy nedostačující, tak bylo navrženo specifičtější filtrování výsledku. Konference by měly jít filtrovat podle roku a uživatel by měl mít možnost nechat si zobrazit jen poslední ročníky archivních konferencí. Druhá jmenovaná funkce je podstatná z toho důvodu, aby se usnadnilo vytváření nových ročníků již existujících konferencí. Také bude zavedena nová vlastnost, a to zvýraznění těch konferencí, které mají alespoň jednoho nearchivního potomka (ne nutně přímého).

Dále je potřeba uživateli umožnit archivní konferenci editovat nebo úplně odstranit. Tyto funkce jsou zatím přístupné a koncipované pouze pro aktivní konference.

1.3 Obnovení konference

Jedná se o vytvoření nového ročníku stávající konference. Výraz „obnovení“ byl zvolen proto, že se tato funkce vztahovala pouze na archivní konference. Znamenalo to tedy, že nelze obnovit konferenci dokud se nepřesune do archivu, což nelze vynutit jinak než změnou termínu konání a automatickým přesunem. Uživatel by ale měl mít tuto možnost k dispozici, takže se do aplikace přidá. Také bylo rozhodnuto o změně názvu tohoto procesu z „obnovení“ na „vytvoření nového ročníku“.

1.4 Skupiny

Důležitou novinkou, spojenou s větším zásahem jak do struktury databáze tak aplikace, vyžaduje zavedení skupin. Ty mají sloužit jako další prvek pro zjednodušení zobrazování konferencí. Kromě toho, že konference patří do různých témat, by měla patřit také do skupin. Ty by si vytvářeli a spravovali sami uživatelé, přičemž by pak jejich vlastní výběr sloužil jako podklad pro další filtr, který by ve spojení s kategoriemi zaručil větší přehlednost výsledné množiny konferencí.

Cílem této novinky je taková situace, kdy má uživatel k dispozici přehled konferencí vyskytujících se jen v jeho skupinách a patřících do kategorií, které jsou navázané zase jen na jeho skupiny.

1.5 Rozšířená kategorizace

Podstatnou součástí aplikace je možnost zařazovat jednotlivé konference do témat. Při výpisu konferencí má potom uživatel možnost tento výstup specifikovat pomocí filtru témat, tzn. vybere si, jakou podmnožinu všech konferencí chce vidět. Tento filtr funguje na hlavní stránce, v seznamu osobních konferencí a v archivu.

Původní verze obsahovala správcem vytvořené kategorie. To bylo později zrušeno ve prospěch výrazně rozsáhlejší a podrobnější kategorizace, založené na ACM klasifikaci. Jedním z podstatných úkolů této práce je opětovně zavést předchozí typ tematických okruhů pro konference, který bude poté fungovat jako alternativa k ACM filtru. Uživateli vytvořené kategorie se budou moci vázat k výše zmíněným skupinám a pomocí spojení těchto dvou prvků bude vytvořen nový typ filtru na všech stránkách, zobrazujících seznam konferencí.

1.6 Příspěvky ke konferencím

Příspěvkem může být..... Dosud si uživatelé zaznamenávali své poslané i přijaté příspěvky sami a to u každé konference zvlášť. V databázi k tomu slouží tabulka osobkonf, ve které se udržuje osobní nastavení uživatele vzhledem k daným konferencím. Vytvoření nového záznamu vyžaduje, aby si uživatel přidal konferenci mezi své osobní a až potom má možnost nastavení měnit. Počet příspěvků je využíván ve všech seznamech konferencí. Nejdříve počet těch, které se vztahují přímo ke konkrétnímu ročníku, a poté součtu ze všech předešlých ročníků (čili rodičovských konferencí). Problém vyvstane ve chvíli, když je smazán uživatel, který má svoje příspěvky v databázi zaznamenány. Spolu s ním se smažou i všechny data s ním související, tedy i jeho osobní nastavení. Informace o počtu příspěvků je tak ztracena a data zobrazovaná v aplikaci jsou nepřesná. Kromě této nekonzistence také nikde nejsou uvedeny názvy příspěvků.

Bylo proto rozhodnuto o vytvoření nové tabulky, představující samotný příspěvek. Data budou vázána k příslušné konferenci a zruší se tak závislost na uživateli. U každé konference půjde zobrazit seznam jejích příspěvků, které se budou dát odstranit, editovat a vkládat nové.

1.7 Oprava chyb z předešlé verze aplikace

1.7.1 Nefunkční řazení a počet zobrazených položek v seznamu

Seznam konferencí lze řadit (vzestupně i sestupně) pomocí jakéhokoliv sloupce (tedy termínů, zkratky, místa i příspěvků). Toto ale ve většině situací nefunguje.

1.7.2 Odsunutí tabulky konferencí

V případě, že si seznam konferencí zobrazí nepřihlášený uživatel, je tabulka na stránce výrazně odsunuta dolů.

Analýza a návrh

2.1 Použité technologie

Tato práce rozšiřuje aplikaci vyvíjenou již delší dobu a to s pomocí osvědčených nástrojů, čili se při výběru potřebných technologií nekonaly žádné změny.

2.1.1 Serverové nástroje

2.1.1.1 Apache

Softwarový webový server, vyvinutý pro Linux a obvykle běžící na unixových systémech. Díky rozsáhlé vývojářské komunitě lze ale spustit na většině dnešních operačních systémů. Brzo po svém vzniku se z něj stal nejpoužívanější HTTP server a tuto pozici si stále drží.

Apache je spuštěný i na serveru Katedry číslicového návrhu FIT ČVUT, kde je Databáze konferencí umístěna.

2.1.1.2 MySQL

Nejpoužívanější multiplatformní relační databázový systém, vlastněný společností Oracle. Samotné MySQL s sebou nenese žádné grafické uživatelské rozhraní a disponuje pouze nástroji pro práci v příkazové řádce, přičemž může uživatel kdykoliv sáhnout po jedné z mnoha „front-end“ desktopových nebo webových aplikací, které umožňují veškeré nutné podmínky ke správě databáze. Dotazy se v MySQL provádí, jak z názvu vyplývá, pomocí jazyka SQL.

2.1.1.3 Cron

Plánovač úloh, fungující na UNIXových systémech, umožňuje nastavovat automatické spouštění nějakých úkolů (např. skriptů) v určitých periodických intervalech. Jeho hlavní součástí je crontab – tabulka s příkazy, které se mají spouštět, spolu s informacemi kdy se mají spouštět.

Cron je využíván pro pravidelnou kontrolu dat v databázi, na jejímž základě se poté data upravují a o změnách jsou případně vyrozuměni uživatelé pomocí rozesílaných emailů.

2.1.2 Programovací jazyky

2.1.2.1 PHP

PHP (původně Personal Home Page, dnes znám pod svým rekurzivním akronymem PHP: Hypertextový preprocesor) je skriptovací programovací jazyk, používaný zejména k tvorbě dynamických internetových stránek a aplikací. Díky své jednoduchosti, rozsahu využití, multiplatformosti a množství dostupných funkcí se jedná o jednoznačně nejrozšířenější skriptovací jazyk na webu. Honosí se podporou mnoha databázových systémů, hostingových služeb a podrobnou dokumentací.

V aplikaci má na starosti většinu logiky. Stará se o komunikaci mezi serverem a klientem, spolupracuje s databází a šablonovacím systémem Smarty.

2.1.2.2 SQL

Zkratka ze Structured Query Language (strukturovaný dotazovací jazyk). Využívá se pro práci s daty v relačních databázích. Jeho syntax je jednoduchý a intuitivní, vychází s anglického mluveného slova.

Aplikace komunikuje s databází výhradně pomocí tohoto jazyka.

2.1.2.3 XHTML

XHTML (zkratka anglického extensible hypertext markup language – rozšiřitelný hypertextový značkovací jazyk), patří do rodiny XML značkových jazyků, se používá k vytváření dokumentů zobrazovaných v Internetu. Jedná se o HTML rozšířené o další vlastnosti z XML, čímž jsou na jeho strukturu kladena přísnější pravidla.

Všechny výstup, vracející se ke klientovi, je v aplikaci řešený právě pomocí XHTML.

2.1.2.4 JavaScript

Jedná se o objektově orientovaný skriptovací jazyk, běžně implementován jakožto součást webových prohlížečů. Slouží převážně k vylepšování uživatelského prostředí aplikace a zprostředkovává větší dynamičnost webových stránek. Jeho syntaxe značně čerpá z jazyka C, potažmo z Java, se kterou má ale kromě toho dost málo společného. Kód se načte uživateli v prohlížeči spolu s požadovanou stránkou a v případě, že má daný klient JS povolen, s ním poté sám pracuje.

Databáze konferencí tohoto jazyku využívá především ke zpříjemnění a zjednodušení mnohých funkcí a operací, se kterými se uživatel setkává, jako je například rychlá a hromadná změna dat ve formulářích nebo skrývání a zobrazování obsahu stránek. Dále často slouží ke kontrole vstupu, zadaného uživatelem, čímž odpadá nutnost častého odesílání dat na server a dramaticky se tak zvyšuje rychlost a dynamika pohybu v aplikaci.

2.1.2.5 CSS

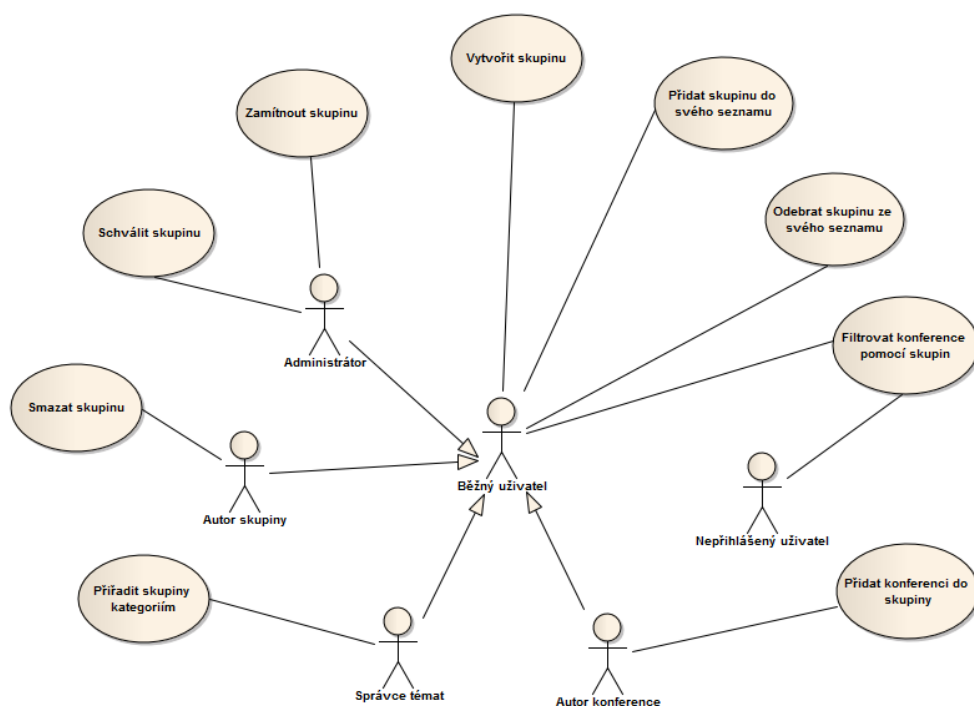
Anglicky Cascading Style Sheets – kaskádové styly, slouží k popisu prezentační sémantiky dokumentu, napsaného pomocí značkovacího jazyka, jako je XHTML nebo XML. Má tedy na starosti vzhled a formátování samotného obsahu. Jeho zásadní výhodou je právě oddělení prezentace od obsahu a tedy snadná a přehledná tvorba těchto dvou částí nezávisle na sobě.

Každá stránka v aplikaci používá stejné dva soubory s CSS (jeden hlavní a jeden pro tisk), na základě kterých je zobrazována v prohlížeči. Téměř vše formátování je obsaženo právě v nich.

2.1.2.6 Smarty

Webový šablonovací systém, určený pro oddělení prezentace od aplikační logiky a rozšiřující možnosti jazyka PHP. Jeho syntax je intuitivní a velmi podobný právě PHP. Mezi jeho výhody patří rychlost, snadná správa a fakt, že se jedná o otevřený software. Smarty kompiluje požadované šablony jako PHP skripty a o ty se poté sám stará, čili uživatel se může zaměřovat pouze na samotnou tvorbu prezentace. Volání Smarty se provádí přímo z PHP skriptu – vytvoří se nová třída Smarty, té se předají potřebné proměnné a šablona, která se má vykreslit.

Tento nástroj byl zaveden v minulé verzi aplikace Bc. Jakubem Římalem a představoval zásadní změnu v její struktuře. Většina prezentační vrstvy byla oddělena od logiky a radikálně se díky tomu zvýšila přehlednost kódu a naopak se snížilo jeho opakování. V této bakalářské práci nebylo třeba využívat novější verze, protože veškeré potřebné prvky už byly k dispozici.



Obrázek 2.1: Use case diagram skupin

2.2 Skupiny

Možnosti, funkce a správa týkající se skupin budou notně využívat jak uživatelských rolí a jimi stanovených práv v aplikaci, tak různých rolích závislých na případech užití, které vychází z konkrétní situace.

Role běžného uživatele bude obnášet základní operace, jako je vytváření skupin, spravování svého vlastního seznamu skupin a možností využívat filtr v seznamu konferencí, v osobních i v archivu. Pokud je uživatel autorem konference, připadá mu právo spravovat množinu skupin, do kterých bude daná konference patřit. Dále autor skupiny může svůj výtvar smazat. Uživatel s administrátorskými právy bude mít standardně nejširší pole působnosti a tedy přístup ke všem zmiňovaným funkcím aniž by mu v nich bránilo autorství. Filtr skupin by měl být (stejně jako filtr témat) přístupný i nepřihlášeným uživatelům a to v seznamu a archivu.

Realizace

3.1 Barevná indikace dat

Nastavení této indikace provádí uživatel ve správě svého účtu. Kromě samotného předstihu, který se počítá ve dnech, si lze nastavit, zda se mají blížící se termíny zvýrazňovat v seznamu všech konferencí (tedy na hlavní stránce aplikace) nebo v seznamu osobních konferencí, případně v obou. Dále je zde možnost nechat si zvýraznit termíny, které už jsou uzavřené (tedy jejich deadline již vypršel). Pro ukládání tohoto nastavení byly vytvořeny čtyři nové sloupce v tabulce uživ.

Samotná kontrola termínů poté probíhá vždy u začátku konání konference a deadlineu odeslání příspěvku. Data pro vytvoření jejich finální verze a rozesílání rozhodnutí o (ne)schválení se kontrolují pouze v případě, že si to uživatel přeje (třeba nějaký příspěvek sám odeslal). To se provádí v osobním nastavení u dané konference (tedy je nutné mít konferenci nejdříve zařazenou mezi osobními). O indikaci blížícího se nebo uzavřeného termínu rozhoduje funkce `zkontrolujDatum()`. Ta vrací číselnou hodnotu, která se poté použije při výstupu u patřičného data v tabulce a na jejím základě je pomocí CSS termín zvýrazněn.

3.2 Změny v archivu

3.2.1 Nearchivní potomci konference

Konference si s sebou nese také identifikátor své rodičovské konference (v případě, že žádnou takovou nemá, id se nastaví na nulu). Díky tomu lze snadno u každé konference získat množinu jejích nearchivních potomků –

daná funkce vyhledá přímé potomky, na které poté rekurzivně zavolá sama sebe. V případě, že rekurze dojde až ke konferenci, která je aktivní, vloží se do předávaného pole a dílčí množiny potomků se spolu slučují. Vráť se tedy pole všech potomků, kteří nejsou v archivu. Tento seznam potom uživatel vidí na stránce každé konference, přičemž může k jejím potomkům okamžitě přejít. Hlavně je této funkce ale využíváno přímo v archivu, kde se dané konference s nearchivními potomky v seznamu vyznačují.

3.2.2 Poslední ročník konference

Důležitou součástí, usnadňující uživateli vytvoření nového ročníku konference, je rychlé nalezení ročníku předchozího. Na jeho základě se konference „obnoví“. To znamená, že je uživateli nabídnut formulář s předvyplněnými hodnotami z předchozího ročníku, kde stačí jen změnit potřebné informace a po uložení se vytvoří ročník nový, který se na základě termínu konání zařadí buď do archivu, nebo mezi aktivní konference.

Kvůli tomu je velmi podstatná možnost nechat si zobrazit jen poslední ročníky konferencí v archivu. To jsou ty konference, které mohou i nemusí mít rodiče, ale sami nemají žádného potomka. Tedy jinými slovy se v databázi nenachází žádná taková konference, která by měla daný ročník jako svého rodiče.

3.2.3 Filtr podle roku

Konference má u sebe sloupec rok, který vychází z jejího termínu konání. Na základě toho lze získat množinu všech dostupných let, z ní dát uživateli vybrat a poté vrátit odpovídající výsledek. Tato množina není pevně dána a čerpá přímo z aktuální databáze, čili se v budoucnu nemusí ručně aktualizovat.

Všechny tyto filtry lze kombinovat z ostatními, s vyhledáváním i stránkováním, čili se uživateli dostane výrazně specifičtějšího výsledku.

3.2.4 Odstranění a editace

Dosud šlo odstranit pouze aktivní konferenci, od kterých zatím nebylo možno vytvářet nové ročníky. Čili skript nepočítal s tím, že může mít konference nějakého potomka. Takže stačilo přidat do něj kontrolu, zda na odstraňovanou konferenci navazují nějaké další a pokud ano, informovat o tom uživatele a smazání zabránit. Tento problém by odpadl, pokud by se v databázi zavedl ročník a odtrhl se od konference, což zmiňuji v závěru práce. K editaci archivní konference byl použit stejný skript jako pro aktivní.

3.3 Obnovení konference

Obnovení dosud fungovalo na základě stejného skriptu jako editace. Po volbě tohoto úkonu byl uživateli nabídnut formulář s předvyplněnými hodnotami, založenými na obnovované konferenci. Zda se bude jednat o vytvoření nového ročníku a ne o editaci rozhodovalo porovnáním termínu konání s aktuálním datem a fakt, že se odesílaná konference nachází v archivu. Kvůli faktu, že se vlastně ve stejném skriptu budou odehrávat tři různé úkony (editace, obnovení archivní a obnovení nearchivní), byla přidána nová proměnná, která se serveru předá pouze v případě, že se má vytvořit nový ročník. Na základě ní se rozhodne, že se má rodičem nové konference stát ta stará, inkrementuje se rok i číselníky s daty a je rovnou rozhodnuto, že se bude vytvářet a ne editovat.

3.4 Uživatelské kategorie

Přístup k seznamu uživatelských kategorií se nachází v menu na záložce Správa. Odsud je možno libovolnou kategorii vytvořit, smazat, editovat nebo přejít na seznam skupin, do kterých je přidělena. Samotná kategorie je definována svým názvem a zkratkou. Název se používá při spravování kategorií ve skupině nebo u konference (může být delší a podrobnější), zatímco zkratka se objevuje ve filtru, kde je potřeba místem šetřit. V případě, že uživatel při vytvoření kategorie zkratku nevyplní, je automaticky nastavena na stejný řetězec, který tvoří název. Pokud je třeba změnit množinu skupin, do kterých daná kategorie spadá, jsou zobrazeny všechny aktivní skupiny.

Dále je možno do kategorií zařadit jednotlivé konference. Odkaz na tuto funkci se nachází na stránce s detaily konference a vidí ho pouze uživatel s právem spravovat témata nebo autor konference. Ti jediní také mohou množinu kategorií měnit. Po vytvoření konference je ihned zobrazen odkaz na tento seznam. V případě obnovení konference se nový ročník zařadí do všech kategorií, ve kterých je i původní ročník.

3.5 Skupiny

3.5.1 Vytvoření skupiny

Vytvořit skupinu může každý přihlášený uživatel. Odkaz k tomuto úkonu je mu nabídnut na všech důležitých místech, tedy hned při příchodu do seznamu vlastních skupin, ve správě skupin a při přidávání nových skupin

do svého seznamu. Při vytvoření je nutné vyplnit název skupiny (který je omezen na sto znaků) a nepovinně také popis. Tyto požadavky jsou kontrolovány jak pomocí JS, tak po odeslání na straně serveru. Po vytvoření už se skupina objeví v databázi, ale pouze jako neaktivní – je potřeba ji schválit, což spadá do kompetence administrátorů. Ti po přihlášení a vstupu do svého seznamu skupin vidí, kolik jich čeká na schválení. Když poté přejdou do správy skupin, jsou zde ty čekající vyznačené. Pokud je skupina schválena, označí se v databázi jakožto aktivní a je od té chvíle viditelná všem uživatelům, kteří si ji pak mohou přidat do svého seznamu a používat ve filtru. Pokud je ale skupina zamítnuta, rovnou se z databáze smaže (administrátor je před vykonáním tohoto úkonu varován).

3.5.2 Správa skupin

Jak již bylo výše zmíněno, mají uživatelé s administrátorskými právy rozšířené možnosti správy skupin. Vstup do této sekce se nalézá v osobním seznamu skupin. Zatímco běžný uživatel vidí v seznamu jen základní informace (tedy název, popis, datum vytvoření a autora), admin vidí datum schválení skupiny i jméno uživatele, který skupinu schválil. Kromě zmiňovaného schvalování a zamítání má dále možnost jakoukoliv skupinu smazat bez ohledu na to, zda-li je jejím autorem.

3.5.3 Operace se skupinami

Výchozí bod je pro uživatele stránka se seznamem jeho skupin (těch, které si vybral, ne které sám vytvořil). Tam se mu zobrazí skupiny, které si do seznamu přidal, včetně informací zmíněných výše. S těmito skupinami může provádět tři akce – odebírat je ze seznamu, mazat nebo spravovat kategorie, které do dané skupiny patří. Operace mazání je mu přístupná pouze pokud je autorem skupiny a lze ho provést i v seznamu volných skupin, tedy těch, které si může do seznamu přidat. Datum a čas vložení nebo schválení skupiny je zobrazován v českém formátu. Zobrazovat a měnit množinu kategorií, příslušných skupině, může pouze správce témat.

3.5.4 Skupiny konference

Stejně jako v případě témat může skupiny konferenci přiřazovat pouze uživatel k tomu oprávněný. To je buď autor konference, administrátor nebo uživatel se zvláštními právy. Skupiny, do kterých konference patří, jsou zobrazeny na stránce s jejími detaily. Odtud také vede odkaz na správu této množiny skupin. Ty jsou ve formuláři reprezentovány pouze názvem,

přičemž skupiny které se nalézají v uživatelském seznamu se tučně zvýrazní a pomocí JS je pak lze snadno označit. Dále je možno zvolit všechny skupiny nebo výběr invertovat.

Při vytvoření i obnovení se konference automaticky zařadí do všech skupin uživatele, který je jejím autorem.

3.5.5 Email o neschválených skupinách

V souboru `naplanovat-kontrolu-skupin.php` se nachází funkce, která projde všechny skupiny v databázi a zjistí, jestli se tam nalézají nějaké čekající na schválení. Pokud alespoň jednu takovou najde, rozešle všem administrátorům aplikace email s touto informací. Pravidelné spuštění tohoto skriptu zajišťuje Cron a kontrola je naplánována na každý druhý den.

3.6 Filtr skupin a kategorií

Tento filtr slouží jako alternativa k dosavadnímu filtru témat, založeném na ACM klasifikaci. Využívá se tedy na všech stránkách, kde se zobrazují konference – tzn. hlavní stránka, osobní a archiv. Všechny patří do kompetence jediného skriptu – `index.php`, který si informaci o tom, jakou stránku má zpracovávat a zobrazit, vezme z GET proměnné v URL. Vzhledem k tomu, že se informace o stavu uživatelského nastavení filtru musí někde udržovat i když se zmiňovaný nachází v jiné části aplikace, ukládají se potřebné proměnné v `session`.

Uživatel tedy patří do nějakých skupin, na které jsou vázány určité kategorie. Podobně i každá konference náleží do různých kategorií a skupin. Cílem filtru je, aby uživatel viděl pouze konference, patřící do jeho skupin a na nich závislých kategorií, přičemž si pomocí výběru z těchto dvou množin může měnit výsledný výstup.

Množiny kategorií, které přísluší skupinám, se mohou protínat. Tedy jedna kategorie může patřit do více skupin. Proto jsou položky pro filtr získány jako sjednocení kategorií ze všech uživatelských skupin. Nejprve je tedy nutno získat skupiny a na jejich základě potom poslat do databáze dotaz na odpovídající kategorie. Tato operace se provádí vždy po vstupu uživatele do aplikace. Rozdíl je tom, zda je dotyčný přihlášen – pokud ne, není nikde informace o jeho vybraných skupinách a vyberou se mu tedy všechny dostupné. Pokud se přihlásí je tato množina změněna na jeho skupiny.

Samotný blok, ve kterém je seznam skupin zobrazován, lze díky JS skrývat nebo zviditelňovat v závislosti na tom, zda si uživatel přeje podle skupin

3. REALIZACE

filtrovat. Pokud ne, poměrně dlouhý seznam by mohl překážet. Viditelnost se na základě hodnoty formulářového tlačítka typu checkbox udržuje na všech stránkách, opět s využitím JS. Na stejném principu funguje i volba „zobrazit jen mé skupiny“, která skryje ty položky, které uživatel videt nepotřebuje. Tyto dvě položky jsou zaškrtnuty při každém přechodu na filtr kategorií, protože bez skupin je výběr podle kategorií zbytečný a žádné konference by se tedy nezobrazily. Oba bloky (skupiny i kategorie) mají u sebe možnost invertovat výběr položek, označit všechny a v případě skupin i označit jen ty uživateli.

Při odeslání požadavku k zobrazení konferencí na základě výběru ve filtru (tedy po stisku tlačítka „Jen vybrané“) se nejdříve zkontrolují odeslané kategorie a skupiny, jestli jsou správně označeny v session. Každá položka, která je vybrána a má se podle ní filtrovat, má u sebe patřičný příznak. Po této kontrole jsou vytvořeny dva řetězce (jeden sestaven z kategorií, druhý ze skupin), které se připojí k hlavnímu dotazu do databáze a vyberou se podle nich jen ty konference, které podmínky splňují.

Vzhledem k tomu, že se nastavení uživatelova filtru udržuje v session, musí informace v ní souhlasit s těmi v databázi. Ty se ale můžou kdykoliv změnit – někdo smaže skupinu, edituje název kategorie, nebo přidá kategorii do skupiny. Ostatní uživatelé se tuto změnu nedozví a pracují tak s daty, které nesouhlasí s databází. Proto je nutné vždy při vznesení požadavku obsahujícím filtr porovnat stávající hodnoty v session s těmi v databázi a synchronizovat je. Tedy smazat položky, které již neexistují, přidat ty, které vznikly a změnit ty editované. To je prováděno v případě kategorií pouze když je nastaven jejich filtr jako aktivní, v případě skupin i tehdy, když se filtruje podle témat. Tato synchronizace by se mohla podmínit změnou v daných tabulkách, ale vzhledem k tomu, že engine tabulek skupina a kategorie je typu innoDB (kvůli cizím klíčům, odkazujícím na ně), nelze u nich zjistit přesný čas poslední změny. Kvůli nenáročnosti a rychlosti sladění dat to ale není větší problém.

Filtr obsahuje i tlačítko „Vše“. Pokud ho uživatel vybere a má přitom zvolen filtr kategorií, označí se všechny jeho skupiny a kategorie, i když označeny nebyly a zobrazuje se podle nich. V případě filtru témat jsou zobrazeny všechny konference, spadající do nějakého tématu, bez ohledu na skupiny.

Přepínat mezi oběma typy filtru lze díky tlačítku typu radiobutton, které má na sobě závislou funkci, spuštěnou při změně stavu tohoto tlačítka. Tato funkce pomocí JS odešle formulář. Pokud uživatel nemá v prohlížeči povolen JS, patřičný filtr se mu zobrazí až po stisknutí tlačítka pro zobrazení výstupu (ať už „Jen vybrané“ nebo „Vše“). V případě vyhledávání konference na základě řetězce je uživateli zdůrazněno, že se vyhledává pomocí

nastaveného filtru a je tedy možné, že právě proto není konference vidět. Pokud je tedy zvoleno „Hledat všude“, rozšíří se vyhledávání na maximální rozsah dat daných filtrem.

3.7 Příspěvky ke konferencím

Bylo proto rozhodnuto o vytvoření nové tabulky, představující samotný příspěvek. Data budou vázána k příslušné konferenci a zruší se tak závislost na uživateli. U každé konference půjde zobrazit seznam jejích příspěvků, které se budou dát odstranit, editovat a vkládat nové.

Nově vytvořený příspěvek obsahuje identifikátor konference, ke které patří, svůj název a informaci o tom, jestli je přijatý. V detailu konference se nachází odkaz na její příspěvky. Ty si může prohlédnout pouze přihlášený uživatel. Ten má také potom možnost příspěvky vytvářet, mazat nebo editovat. Zobrazená množina příspěvků se váže buď na danou konferenci, nebo na všechny její rodiče. Toho lze v seznamu dosáhnout jedním kliknutím. U každého příspěvku je také odkaz na detail konference, ke které patří. V přehledu konferencí je potom vidět informace o počtu příspěvků, která slouží i jako odkaz na jejich seznam.

Díky nezávislosti na uživateli se teď tedy potřebná data neztrácí a jsou konzistentní. Informace o příspěvcích, které se nalézají v osobním nastavení uživatelů, už neslouží v aplikaci žádnému jinému účelu, čili se mohou smazat. Byly nahrazeny příznakem, na základě kterého si uživatel zvolí, zda chce mít v aplikaci zvýrazňovány termíny týkající se příspěvků k dané konferenci (jeho finální verze a vyrozumění o přijetí). Volba tohoto příznaku se nachází v osobním nastavení konference.

3.8 Oprava chyb z předešlé verze aplikace

3.8.1 Nefunkční řazení a počet zobrazených položek v seznamu

Ukázalo se, že se jedná o vlastnost PHP, implementovanou ve verzi 4.0.3, sloužící jakožto ochrana proti tzv. DoS (Denial of Service – odmítnutí služby) útoku. V nastavení PHP se nachází proměnná, která omezuje velikost dat odesílaných klientem na server. Přitom rozšířená kategorizace byla v aplikaci řešena tak, že se uživateli nabídla množina témat, ze kterých mohl vybírat, ve formě checkboxů. Zvolená témata jsou s formulářem odeslána a zpracována na serveru. Těchto témat je téměř čtrnáct set a přednasta-

vená maximální velikost POST proměnné je tisíc. Při odeslání formuláře se posílají pouze zaškrtnuté checkboxy, což při výběru více jak dvou třetin témat vyústí v situaci, kdy jsou veškerá data překračující danou hranici ignorována. Vzhledem k tomu, že se jak informace o řazení, tak o počtu zobrazených položek nachází právě až za tématy, vůbec nejsou zohledňovány.

V nastavení PHP je proto třeba zvýšit maximální počet dat odesílaných metodou POST z jednoho tisíce na dva, což je pro požadavky aplikace dostačující. Toho lze dosáhnout dvěma způsoby: buď přímou změnou v konfiguračním souboru PHP – `php.ini`, nebo nepřímo pomocí souboru `.htaccess`, umístěném v adresáři s aplikací. Druhý způsob je praktičtější, protože nemění globální nastavení a změna je pouze místní, ale pro jeho fungování je třeba mít na serveru povolení k takovým úkonům.

3.8.2 Odsunutí tabulky konferencí

Problém se vyskytoval jak na hlavní stránce, tak v archivu. To je dáno tím, že se v obou případech využívá stejná šablona. Právě v této šabloně se nacházela chyba v kódu, způsobující generování odřádkování s každou další konferencí. Týkala se vytváření pomocných odkazů, které se přihlášenému uživateli zobrazí v tabulce konferencí u každé položky. Jedná se o info, editaci, odstranění a fórum dané konference.

Tyto prvky ale nebyly na daném místě potřebné, protože všechny jmenované odkazy se nacházejí na stránce s informacemi o konferenci, čili jen o jedno kliknutí dále. Především ale spolu s nimi výrazně narůstala výška každého řádku tabulky, což při zobrazování většího počtu konferencí ztěžovalo pohyb na stránce. Celá část byla tedy z aplikace odstraněna.

3.8.3 Nefunkční JS u filtru

Vybrané položky ve filtru témat lze invertovat – převrátit hodnotu všech checkboxů. Tato možnost ale nefungovala v archivu a seznamu osobních konferencí. Funkce byla nahrazena novou a funkčnost obnovena.

3.9 Další úpravy a vylepšení

3.9.1 Odkaz na homepage konference

V seznamu je nyní u každé konference odkaz na její domovskou stránku. Ten se nachází také na stránce konference, ale kvůli rychlosti byla zvolena

tato alternativa, která navíc využívá vlastnosti otevření odkazu na nové záložce.

3.9.2 JS u filtru témat

Pokud si uživatel přál v rozšířeném filtru témat označit jednu určitou podskupinu nebo samotné podtéma a měl při tom označeny jiné položky (tedy ne vše najednou), musel si nejdříve nechat zobrazit všechna témata, poté výběr invertovat, aby nebyla označena žádná a teprve pak vybrat například jedno konkrétní téma.

K tomu byla napsána JS funkce, která při výběru hlavního tématu společně s ním buď označí nebo odoznačí všechna jeho podtémata na základě třídy, do které patří. Uživatel tak může rychle vybrat libovolné téma bez nutnosti odesílání požadavku na server.

3.9.3 Odesílání formulářů pomocí JS

U některých prvků, sloužících ke změně množiny zobrazovaných dat, byla přidána JS funkce odesílající příslušný formulář bez nutnosti dělat to ručně pomocí tlačítka. Tato funkce se váže na kliknutí nebo změnu spojenou s daným prvkem. Využívá se toho například při přepínání mezi typy filtru témat, volby roku konání konferencí nebo počtu položek na stránce. V případě, že má klient vypnutý JS, stále tu zůstává možnost odeslat formulář tlačítkem po výběru všech parametrů.

Vzhledem k tomu, že Internet Explorer reaguje na změnu stavu u některých elementů jinak než ostatní prohlížeče (změnu detekuje až když prvek není označen), bylo nutno to ošetřit a prvek označit už při kliknutí na něj.

3.9.4 Redukce a úpravy v kódu

Ve skriptech se nacházelo dost proměnných, které byly zbytečné a pouze ztěžovaly orientaci v kódu. Nebyl naplno využit potenciál systému Smarty, který obsahuje širokou škálu matematických funkcí, intuitivní prostředí pro práci s cykly a podmínkami, výrazně jednodušší a přehlednější syntax a jako parametry předávané ze skriptu přijímá i asociativní pole a objekty.

Například počet zobrazených konferencí na stránce se do Smarty předával v podobě deseti různých proměnných (vybírá se z roletového menu, minimální počet je deset položek a maximální sto, možnosti jsou po deseti), každá z nich náležela jedné možnosti v menu a obsahovala informaci, zda je daná možnost označena či ne. Tento proces lze nahradit pomocí cyklu, zvaném ve Smarty section, prováděném po deseti krocích do maximální

hodnoty sto a jediné proměnné (počet konferencí na stránce), na základě které se ve Smarty daná možnost označí.

Dalším příkladem je řazení vybraných konferencí podle sloupců, které je řešeno pomocí jim náležících tlačítek v hlavičce tabulky. Každý sloupec měl svůj vlastní příznak, zda se řadí zrovna podle něj, jakým směrem (jestli se jedná o vzestupné nebo sestupné řazení) a jestli se u něj má zobrazovat odpovídající šipka. Přitom důležité je pouze jméno sloupce a typ šipky, o zbytek se pomocí podmínek a funkce pro absolutní hodnotu postará Smarty. Počet proměnných se tím snížil o deset a hlavně délka kódu je o více než sto řádků kratší.

Dále se často předávala do šablony skupina proměnných, náležících jednomu objektu (jednotlivé atributy konference nebo uživatele), což se dá nahradit asociativním polem. Také namísto proměnných s textovými řetězci (např. odkazy) lze využít boolean hodnoty a na jejich základě vypisovat text až v šabloně, kam patří.

Tyto a další úpravy v kódu byly průběžně upravovány během vývoje a výsledkem jsou tedy kratší a přehlednější skripty.

3.9.5 Zachování řazení konferencí

Na všech stránkách, kde se zobrazují konference, lze daný výstup seřadit podle některého ze sloupců. Informace o tom se ale přenášela pouze metodou POST, takže se po opuštění stránky nastavila opět na původní hodnotu.

Zachování řazení by šlo vyřešit pomocí metody GET – tedy že by se proměnná předávala mezi stránkami v URL. To by ale znamenalo, že se tato informace musí přenášet i na stránky, kde není potřeba a jenom překáží. Jednodušší a účelnější tedy je uložit odpovídající proměnnou do session a odsud ji pokaždé zavolat. Přepíše se pouze v případě, že se uživatel rozhodne řazení změnit.

3.9.6 Zpětné přiřazení rodiče konferenci

Nově vytvořená konference má identifikátor rodiče nastaven vždy na nulu. Pokud je ale obnovena, jejím rodičem se stává původní konference. Tuto vlastnost nebylo možno v aplikaci ovlivnit a tedy v případě, že byl přiřazen špatný rodič nebo vůbec žádný, šlo to opravit pouze přímou změnou v databázi. Proto bude přidána možnost určovat rodiče již existující konference.

Při obnově konference je ve formuláři přidáno pole pro ID rodiče, předvyplněné na hodnotu původní konference. Vzhledem k tomu, že pro editaci i přidávání je použita stejná šablona, bylo nutné zajistit, aby se toto pole nezobrazovalo a nezpracovávalo v případě vytvoření nové konference, tam

je hodnota rodiče vždy nulová. Při editaci je kontrolováno, zda konference udávaná jako rodičovská v databázi existuje a pokud ano, je přenastavena.

3.10 Databáze

Zavedení nových prvků si vyžádalo podstatný zásah do struktury databáze, ať už v podobě jejího rozšíření nebo změn. Jak skupiny tak uživatelské kategorie potřebují být vázány na více entit a to s kardinalitou typu M:N. Pro každý takový vztah je tedy nutno vytvořit pomocnou vazební tabulku, která je složena z kombinací obou použitých klíčů. Další nová tabulka představuje příspěvek a závisí na konferenci.

Potom byly přidány sloupce do tabulky uživatele `uziv`, které slouží k ukládání jeho preferencí ohledně zvýrazňování termínů v aplikaci. Počty odeslaných a přijatých příspěvků, které si uživatel vedl v osobním nastavení konference, byly zrušeny a nahrazeny jednou boolean hodnotou. A konečně další sloupec u uživatele, ve kterém se uchovává MD5 otisk jeho unikátního řetězce, vygenerovaného po přihlášení a sloužící jako další bezpečnostní prvek.

Dále byla změněna jména indexů v tabulce `forum` a `temauziv` tak, aby odpovídala konvenci nastolené u všech ostatních indexů. To bylo potřeba také proto, že dva indexy spojující témata s uživateli zde zůstaly z první verze aplikace, která místo „téma“ používala pojmu „vědecká skupina“, což by při zavedení nových skupin bylo matoucí.

Kvůli nemožnosti mít v tabulce MySQL dva sloupce typu `timestamp`, které by v případě skupin ukládaly přesný čas při vytvoření a schválení, bylo toto nutno obejít pomocí triggerů.

3.11 Bezpečnost

Uživatelská hesla jsou v databázi uložena jako MD5 otisk, tedy řetězec vytvořený hashovací funkcí, v tomto případě `md5()`. Cílem je, aby se hesla neukládaly pouze jako prostý text a nebylo je tak možné získat přímo z databáze.

Identita uživatele je po úspěšné autentizaci během přihlášení udržována na serveru v `session`, a to v podobě jeho uživatelského jména a `uid` (jednoznačného identifikátoru, pod kterým je uložen v databázi). Na základě těchto informací je potom prováděna veškerá autorizace – tedy kontrola, zda má povolení k přístupu do určitých částí aplikace. To se především vztahuje na všechny skripty, vyžadující přihlášení, ale také prvky, ke kterým mají pří-

3. REALIZACE



Obrázek 3.1: ER diagram struktury databáze

Tabulka 3.1: Nové sloupce v databázi

Název	Typ	Popis
predstih_varovani	int(10)	
uzavrene_terminy	tinyint(1)	
zvyraznit_seznam	int(10)	
zvyraznit_osobni	tinyint(1)	
unikatni_otisk	varchar(32)	
id_skupina	int(10)	
id_autor	int(10)	
id_admin	int(10)	
nazev	varchar(50)	
popis	text	
aktivni	tinyint(1)	
dat_vlozeni	timestamp	
dat_potvrzeni	timestamp	
id_kategorie	int(10)	
nazev	varchar(80)	
zkratka	varchar(50)	
id_prispevek	int(10)	
id_konf	int(10)	
nazev	int(10)	
prijat	tinyint(1)	
prispeveknotify	tinyint(1)	

stup pouze uživatelé se specifickými právy. Kvůli posílení bezpečnosti byla navíc zavedena nová proměnná, do které se po úspěšném přihlášení nastaví unikátní klíč, vygenerovaný pomocí funkce `uniqid()` na základě náhodného čísla. Ten je přidán k uživateli do `session` a poté zašifrovaný algoritmem MD5 a uložen do databáze. Během autorizace jsou potom tyto dva řetězce porovnávány. Potenciální útočník by tak pro podvržení identity skutečného a přihlášeného uživatele musel znát kromě kódu jeho `session` také tento náhodný klíč.

Veškerá vstupní data předávaná do aplikace jsou ještě před tím, než se s nimi začne pracovat, ošetřena PHP funkcí `htmlspecialchars()`. Ta zaručí, že se nikam nemůže dostat škodlivý kód, který by útočnickovi jinak umožnil zasahovat do struktury aplikace. Funguje na základě toho, že speciální znaky převádí na HTML entity, které se tedy potom nechovají jako součást kódu. Typ útoku, který těchto nošetřených vstupů zneužívá, se nazývá Cross-site

3. REALIZACE

scripting (XSS), a jeho dopad se pohybuje od relativně neškodné změny vzhledu stránky až po závažné znefunkčnění aplikace, případně získání citlivých dat uživatelů.

Na stejném principu funguje i tzv. SQL injection, kdy útočník podstrčí svůj škodlivý kód dotazu do databáze a může tak získávat nebo mazat důležité informace. Ošetření všech vstupů, které mohou něčeho takového dosáhnout, je tedy řešeno funkcí `mysqlrealescapestring()`, která stejně jako v předchozím případě znefunkční speciální znaky.

Testování

Veškeré změny a testování probíhaly za pomoci reálných dat, získaných díky vedoucímu práce z databáze na serveru `ddd.fit.cvut.cz`, kde je aplikace spuštěna. Během celého vývoje byly jednotlivé funkce průběžně testovány s pomocí těchto i dalších vhodných dat.

4.1 Validita XHTML a CSS

Všechny stránky byly úspěšně bez chyb testovány v oficiálním validátoru konsorcia W3C jakožto XHTML 1.0 Strict. Stejně tak oba soubory s kaskádovými styly splňují kritéria potřebná pro standard CSS 3.

4.2 Internetové prohlížeče

Aplikace byla testována ve všech nejrozšířenějších webových prohlížečích, přičemž u každého z nich funkčnost odpovídala požadavkům. Vzhled zobrazovaných prvků je také konzistentní, pouze s drobnými odlišnostmi jako je například rozdíl mezi velikostmi písma a mezer u některých blokových elementů. To je dáno rozdílným vykreslovacím jádrem u prohlížečů a na vzhledu a přehlednosti výsledné stránky to není nijak výrazně znát. Testované prohlížeče:

- Google Chrome 23.0
- Mozilla Firefox 17.0
- Opera 12.11
- Internet Explorer 8.0

- Safari 5.1.7

4.3 Servery

Aplikace byla vyvíjena pomocí balíku WAMP verze 2.2, což je soubor nástrojů určených pro tvorbu webových aplikací, sestávající ze serveru Apache, databázového systému MySQL a PHP. Dále byla nasazena na zmiňovaném serveru ddd.fit.cvut.cz Katedry počítačových systémů FIT ČVUT, kde fungují stejné nástroje, pouze se rozcházejí jejich verze. Na obou serverech aplikace funguje výborně, s drobnou výjimkou stránek, ve kterých se využívá stromová struktura rozšířeného filtru témat. U nich je potřeba většího množství operací a objemnějšího výstupu, tedy i delší časové náročnosti, jak již vysvětloval ve své práci jejich autor, Bc. Jakub Římal. Pomalejší načítání daných stránek tedy není vinou serveru.

Zavedení kategorizace konferencí na základě témat, stanovených ACM klasifikací, znamenalo výrazně zvýšený objem přenášených dat – místo desítek definovaných témat, na základě kterých se kategorizovalo dříve, se počet rozrostl na téměř čtrnáct set. Velikost stránky s touto stromovou strukturou, kterou klient obdrží, je zhruba šestkrát větší (každé podtéma je ve struktuře vzhledem k rodičům reprezentováno pomocí obrázku).

Závěr

Všechny požadavky, které si tato práce dala za cíl, byly úspěšně splněny. Největší rozšíření v podobě kategorizace konferencí a jejich následné přidělování do uživatelských skupin poskytuje důležitou alternativu k dosavadnímu hledání a prohlížení konferencí. To hlavně díky rozšířenému filtru, který rychle a přehledně zprostředkuje požadovaný výstup. Dále bylo implementováno množství podpůrných funkcí, usnadňujících pohyb a práci v aplikaci. Chyby, nacházející se v minulé verzi byly odstraněny a zdrojový kód upraven tak, aby byl pro další případné úpravy nebo rozšíření přehlednější.

Další možnosti rozšíření se týkají především technické stránky. Témata zobrazovaná uživateli ve stromové struktuře by bylo efektivnější předávat v podobě asociativního pole. Zjednodušily by se tím potřebné operace v kódu a odpadla by nutnost zvyšovat počet proměnných, předávaných serveru. Objem dat by byl stejný, ale místo každého tématu zvlášť by pracovalo se s jejich polem, což je jen jedna proměnná.

Zásadnějším krokem, zasahujícím dost podstatně do struktury aplikace, by bylo oddělení ročníku od konference. Snížil by se tím počet opakujících se dat v databázi a ročník by se mohl spravovat nezávisle na konferenci a naopak. Tedy změny v konferenci by se promítly do všech jejích ročníků, ale samotný ročník by zůstal samostatný.

Potom je možné aplikaci kompletně přepracovat pomocí některého frameworku. To by vedlo k výrazné redukci kódu, jeho zvýšené přehlednosti a bezpečnosti. Takový krok by ale neměl výraznější vliv na současnou funkcionálníitu.

Seznam použitých zkratk

GUI Graphical user interface

XML Extensible markup language

Obsah přiloženého CD

	readme.txt	stručný popis obsahu CD
	exe	adresář se spustitelnou formou implementace
	src	
	impl	zdrojové kódy implementace
	thesis	zdrojová forma práce ve formátu L ^A T _E X
	text	text práce
	thesis.pdf	text práce ve formátu PDF
	thesis.ps	text práce ve formátu PS