Slovenská technická univerzita v Bratislave
FAKULTA ELEKTROTECHNIKY A INFORMATIKY
Katedra informatiky a výpočtovej techniky

Softvérová podopora organizovania vedeckej konferencie

Tímový projekt
Projektová dokumentácia

Tím 4:
Peter Kósa (kosa@decef.elf.stuba.sk)
Peter Lopeň (lopenpet@decef.elf.stuba.sk)
Richard Richter (virgo@fornax.sk)
Róbert Trebula (trebular@decef.elf.stuba.sk)
Ján Zázrivec (zazrivec@decef.elf.stuba.sk)

17. december 2001

Obsah

Zadanie

Pre úspešné zorganizovanie vedeckej konferencie treba zvládnuť niekoľko typických aktivít, ako sú:

- prezentácia konferencie;
- prihlasovanie účastníkov na konferenciu;
- zasielanie a oponovanie príspevkov;
- vytlačenie zborníka, apod.

Hoci sa vedecké konferencie líšia svojím obsahom a veľkosťou, podstata činností spojených s organizovaním konferencie je podobná.

Cieľom projektu je vytvoriť softvérový systém podporujúci organizovanie vedeckých konferencií. Organizátorom a programovému výboru má systém pomôcť prezentovať, plánovať a riadiť vedeckú konferenciu. Systém musí zohľadniť rôzne alternatívy pri organizovaní konferencie a má byť konfigurovateľný podľa špecifických požiadaviek a potrieb konferencie. Systém musí diferencovať používateľov v závislosti od role, ktorú hrajú vo vedeckej konferencii.

Systém musí umožniť základné aktivity pri organizovaní konferencie, ako sú:

- prihlasovanie účastníkov na konferenciu;
- elektronické posielanie a prijímanie príspevkov;
- evidovanie údajov o prihlásených príspevkoch a účastníkoch.

Systém musí poskytnúť podporu programovému výboru konferencie pri spoločnej komunikácii a rozhodovaniach týkajúcich sa najmä oponovania a výberu príspevkov. Treba uľahčiť najmä:

- priraďovanie oponentov ku príspevkom;
- posielanie príspevkov na oponovanie;
- monitorovanie aktuálneho stavu v procese oponovania.

Vytvorený produkt by mal poslúžiť pri organizovaní medzinárodnej konferencie ADBIS 2002.

1 Úvod

Tento dokument popisuje naše riešenie zadania *Softvérová podpora organizovania vedeckej konferencie*, ktoré je súčasťou našej práce na predmete Tímový projekt. Obsah prvej etapy (analýza, špecifikácie a hrubý návrh) je náplňou kapitol 2 až 4.1. V kapitolách 4.2 a ďalších je opísaný výsledok našej práce v rámci etapy 2 – ide o korekcie prvej etapy a o používateľskú príručku k prototypu.

Organizovanie vedeckej konferencie ponúka množstvo problémov. Náročnosť organizovania konferencie v dobe globalizácie – v dobe informačnej spoločnosti – neustále rastie. Dnes je možné zvládnuť aj veľké konferencie pomerne pružne – vyžaduje si to ale, samozrejme, iné prostriedky. Dnes organizátori konferencií disponujú možnosťami, ku ktorým ich predchodcovia nemali prístup – a to stačí ísť o jediné desaťročie dozadu. Elektronická komunikácia účastníkov pred samotnou konferenciou – najmä využitie technológie WWW – umožňuje zvládnuť prípravu konferencie za kratší čas, resp. za ten istý čas si môžu účastníci konferencie vymeniť viacej dôležitých informácií.

Ak je možné zvýšiť množstvo informácií, ktoré sa všetky zúčastnené strany dozvedia, vznikne potreba tieto informácie spracovať. Je možné, že sa o všetko postará skupina ľudí – samotní organizátori. Rovnako tak je ale možné, že časť úloh bude automatizovaná, pričom človek zasiahne, len ak to bude nutné. Je jasné, že nenastala doba, kedy stroje budú posudzovať príspevky posielané do konferencie, ale je možné napríklad priradiť príspevku kľúčové slová a na základe nich uľahčiť výber oponenta, alebo takýto výber priamo navrhnúť.

Z naznačeného vyplýva, že vhodný informačný systém by mohol byť neoceniteľným pomocníkom pri organizovaní vedeckej konferencie.

1.1 Štruktúrovanie dokumentov

Odovzdávaná práca obsahuje nasledujúce dokumenty:

- **Projektová dokumentácia**, ktorá sa zaoberá postupom riešenia našeho projektu. Obsahuje analýzu, špecifikáciu a návrh.
- **Používateľská príručka** k prototypu systému Con4U, čo je požadovaný výstup pre druhú etapu.
- **Posudok práce konkurenčného tímu** – tj. posudok, ktorý sme napísali my na prácu tímu Confess.
- **Posudok našej práce**, ktorý vypracoval tím Confess. Tento posudok sa zaoberá naším dokumentom po kapitolu 4.1 včítane.
- **Naše vyjadrenie sa k posudku**.
- **Dokumentácia k riadeniu projektu**, ktorá obsahuje všetky zápisy, ponuku, plány a ostatné dokumenty, ktoré s riadením súvisia.
- Prípadné ďalšie dokumenty k riadeniu, ktoré vznikli ako samostatné šablóny, bude na ne odkazované z dokumentácie k riadeniu projektu.

Dokumenty sú v práci zoradené v uvedenom poradí. Dokumenty budú od seba oddelené farebnými papiermi, pričom každý z dokumentov má buď výrazný titul alebo samostatnú titulnú stranu. Každý samostatný dokument má samozrejme vlastné číslovanie strán.

1.2 Zmeny oproti predchádzajúcej etape

Pri riešení projektu došlo k niekoľkým zmenám, ktoré spôsobili, že niektoré časti dokumentácie k etape 1 (po kapitolu 4.1 včítane) nie sú celkom aktuálne. Medzi najvážnejšie zmeny patrí:

- Dohodla sa terminológia v oblasti **tém** (Topic) a **kľúčových slov** (Keyword). Témy konferencie sú konfigurovateľné; pre príspevok autor definuje, k akým témam patrí; oponenti definujú svoje preferencie tém. Témy sú teda tým, čo sa v etape 1 nazývalo kľúčovými slovami. Kľúčové slová tvoria dnes len informačné pole príspevku, ktoré sa v systéme nijak ďalej nespracováva.

- Statické stránky budú v správe Con4U, ba čo viac, bude možné pri ich správe definovať viditeľnosť pre jednotlivé roly.

Uvedené zmeny sú najväčšími nezrovnalosťami v etape 1, preto sme ich uviedli hneď v úvode.

1.3 Použité notácie

Pri väčšine diagramov, ktoré sa objavia v tejto dokumentácii, sa držíme notácie UML.

V kapitole 3.1 sme použili diagram, ktorého notácia je popísaná priamo na mieste.

Napokon pri popise formulárov sme použili štruktúrovaný text, ktorého komponenty sú tiež bližšie vysvetlené priamo na mieste.

2 Analýza

2.1 Jestvujúce systémy

Pred diskusiou so zákazníkom o požiadavkách na systém boli preštudované požiadavky, ktoré boli dodané ako súčasť zadania. Boli identifikované tieto základné požiadavky:

- plánovanie vedeckej konferencie;
- riadenie vedeckej konferencie;
- prezentovanie vedeckej konferencie;
- konfigurovateľnosť systému.

Vykonali sme analýzu existujúcich systémov, ktoré sa zaoberajú manažmentom vedeckých konferencií. Ako hlavný zdroj referencií poslúžila stránka – ACM Special Interest Group Governing Board: Summary of Conference Management Software[1]. Táto stránka je síce z roku 1999, takže niektoré linky a nej boli nefunkčné, poskytla však pomerne veľký počet odkazov na jednotlivé systémy na manažovanie konferencií. Pri štúdiu existujúcich produktov sme postupovali v poradí, v akom boli na tejto stránke uvedené a pri každom produkte sme si všímali najmä nové rysy systému, ktoré prinášali nové dátové, funkčné či iné prvky.

V nasledujúcich podkapitolách bude stručný popis jednotlivých systémov so zameraním na nové vlastnosti, ktorými sa tento produkt odlišuje a ktoré budú zapracované do celkového zoznamu možných vlastností systému na menežovanie vedeckej konferencie.

2.1.1 Confman

Nórsky systém Confman[2], ktorého posledná verzia je z roku 2000, sa javí ako jeden z najprepracovanejších, čo sa týka používateľnosti. Je veľmi dobre konfigurovateľný a pomerne jednoducho poňatý.

Okrem funkcií bežných pre všetky takéto systémy, má navyše tieto pozoruhodné vlastnosti:

- riadené posielanie mailov účastníkom konferencie;
- on-line oponovanie príspevkov;
- abstrakty článkov na webe;
- pri registrácii účastníkov konferencie je možné vypňovať aj položky týkajúce sa platenia poplatkov za účasť;
- posudok je tu poňatý ako formulár s prednastavenými otázkami a možnými odpoveďami (vo forme select-boxov). Tieto otázky a odpovede sú konfigurovateľné.

V systéme sa dá identifikovať niekoľko typov používateľov s príslušnými prípadmi použitia:

- **autor príspevku** pridáva svoj príspevok do systému a je informovaný o postupe pri jeho oponovaní a prijímaní;

[1] http://www.acm.org/sigs/sgb/summary.html
[2] http://confman.unik.no/~confman/ConfMan/

- **člen programovej komisie** posudzuje jednotlivé príspevky a odporúča/neodporúča ich na prijatie;
- **účastník konferencie** podáva prihlášku s vyplnením predvolených (konfigurovateľných) položiek;
- **administrátor** riadi zasielanie mailov autorom príspevkov, priraďuje články oponentom, prezerá a riadi prihlásených účastníkov na konferenciu. Navyše konfiguruje všetky konfigurovateľné položky v rôznych častiach systému.

Systém ako taký sa môže nachádzať v jednom zo stavov:

- prijímanie príspevkov a posudkov;
- hlasovanie komisie o tom, ktoré príspevky prijať a ktoré nie;
- finálna fáza a konečné rozhodnutie.

Zmeny aktuálnej fázy vykonáva samozrejme administrátor.

2.1.2 Edas

Systém Edas[3] z newyorskej Columbia University nemá na webe funkčnú demonštračnú verziu, takže informácie sme získavali len z dokumentácie.

Oproti už preštudovanému systému Confman sa líši v týchto smeroch:

- namiesto fáz konferencie sú tu **stavy príspevku** – podaný abstrakt, podaný príspevok, priradený na posudzovanie, posúdený, prijatý/neprijatý na konferenciu;
- člen programovej komisie si môže sám vybrať, ktoré príspevky bude oponovať;
- automatické priraďovanie príspevkov oponentom podľa miery zhody kľúčových slov príspevku a preferencií oponenta;
- v systéme je možné podávať opravy (corrections) jednotlivých príspevkov po ich odoslaní, aj ako reakciu na posudky.

2.1.3 Puma

Systém Puma – Publication Management System[4] z inej newyorskej univerzity – Cornell University – bol pôvodne len prípadovou štúdiou použitia ich vlastného databázového systému Predator, no napriek tomu je na dobrej úrovni. Na web-stránke projektu sú odkazy na demonštračnú verziu systému, sú však nefunkčné.

V tomto systéme sú oddelené rozhrania (interfaces):

- **oponent** – posudzuje články;
- **administrátor** – priraďuje články oponentom.

[3] `http://www.cs.columbia.edu/~hgs/edas/edas.html`
[4] `http://www.cs.cornell.edu/database/predator/puma/puma.html`

Pozoruhodnou vlastnosťou systému Puma je, že sa cez tento systém nedajú posielať samotné príspevky. Všetky príspevky do systému vkladá ručne administrátor. Tento nedostatok je vysvetľovaný tým, že poslanie článku cez web-rozhranie by bolo príliš pomalé a nestihlo by sa vykonať počas jedného behu CGI skriptu. Toto je potrebné uvážiť a otestovať.

Posudky sú modifikovateľné aj po odoslaní. Sú tu zavedené tzv. **vlákna posudkov** (review threads). Jedná sa o to, že sa vytvára diskusia oponentov o jednom článku, ktorá sa zaznamenáva do stromovej štruktúry priradenej k tomuto článku.

Oponent – nazývaný rozhodca – do systému zadáva svoje preferencie, články akého typu by chcel (prípadne nechcel) oponovať. Administrátor má možnosť vytvárať si správy vo forme poradí príspevkov podľa hodnotenia oponentov a to podľa rôznych kritérií.

Systém je „mäkký“, čo sa týka dodržiavania termíov skončenia jednotlivých fáz, takže je možné aj dodatočné posielanie posudkov, hlasov a podobne, ktoré však už v závislosti na prebiehajúcej fáze neovplyvnia výsledky skončených fáz.

Systém sa zaoberá **konfliktnými posudkami** – ak sú na jeden príspevok radikálne odlišné posudky od niekoľkých oponentov, tieto konflikty sa musia vyriešiť. Práve na to sú vhodné diskusie k jednotlivým článkom. Z takejto diskusie by malo vyplynúť, ktorý z posudkov je správnejší, prípadne kto z oponentov svoj posudok zreviduje.

Puma má podrobne rozčlenené fázy – stavy systému:

1. posielanie abstraktov;
2. posielanie príspevkov;
3. posielanie preferencií oponentov;
4. priradenie oponentov príspevkom;
5. posielanie posudkov;
6. rozriešenie konfliktných posudkov;
7. príprava na stretnutie programovej komisie – generovanie reportov podľa rôznych kritérií;
8. aplikácia výsledkov stretnutia programovej komisie – posielanie oznamu o prijatí/neprijatí príspevku autorom.

2.1.4 Witan Web

Produkt Witan Web[5] z kanadskej Národnej výskumnej rady je v krajine svojho vzniku hojne používaný. Prináša niekoľko nových myšlienok, ktoré by mohli byť zapracované do návrhu nášho systému.

Pretože je nasadzovaný v mulitjazyčnej krajine, zaoberá sa znakovými sadami a jazykmi dokumentov.

Oponenti môžu prostredníctvom systému Witan Web žiadať o konzultáciu, pomoc od iných oponentov týkajúcu sa niektorého posudzovaného príspevku.

Skoro v každom formulári je pole „dodatočné informácie“, kam sa zapisujú všetky záznamy pre ktoré nie sú predvolené otázky.

Zaujímavosťou je položka „študentský príspevok“ vo formulári pridávania článku.

Položky formulárov v tomto systéme nie sú konfigurovateľné priamo prostredníctvom web-rozhrania, čo je asi najväčší nedostatok tohoto systému.

[5] `http://witanweb.iit.nrc.ca`

2.2 Technologická analýza

Po začiatočnej analýze zadania sa ako nevyhnutná ukázala potreba hlbšie sa oboznámiť s technologickými postupmi a metódami umožňujúcimi realizovať riešenie ako klient/server aplikáciu. Z tohto dôvodu sa v nasledujúcom texte zaoberáme jednotlivými technológiami a skúmame ich z hľadiska použitia v navrhovanom systéme.

2.2.1 Služba WWW

Základnou technológiou, ktorú použijeme v riešení, je služba World Wide Web (WWW). Táto služba vznikla na pode CERNu v rokoch 1989–90 a je založená na troch pojmoch: **HTML**, **HTTP** a **URL**.

HTML – Hypertext Markup Language HTML je značkovací jazyk založený na SGML na popis štruktúry webovských stránok. Je platformovo nezávislý. Prvá formálna špecifikácia je HTML 2.0, ktorá obsahuje prostriedky pre základné formátovanie a štrukturovanie dokumentu, možnosť vkladať obrázky a vytvárať formuláre. Z dôvodu neštandardizovaného rozširovania výrobcami prehliadačov vzniklo v roku 1996 konzorcium WšC, ktoré vybralo spoločnú podmnožinu rozšírení a schválilo HTML 3.2.

V súčasnosti je aktuálna verzia HTML 4.0, ktorá predstavuje veľký skok vpred. Z najdôležitejších vylepšení treba spomenúť podporu kaskádových štýlov, skripty vložené do stránky, mulitmediálne objekty, rozšírené možnosti tabuliek, podporu viacerých jazykov, rozšírenie formulárov a rámy. Takmer kompletnú podporu HTML 4.0 majú prehliadače Internet Explorer 4.0 a 5.0 a tiež Netscape Navigator. V roku 1999 vzniká XHTML 1.0. Ide o štandard HTML 4.0 upravný tak, aby stránky v XHTML predstavovali XML dokumenty.

HTTP – Hypertext Transfer Protocol HTTP predstavuje protokol na prenos stránok medzi webovským serverom a prehiadačom. Ide o jednoduchý aplikačný a bezstavový protokol vystavaný nad protokolom TCP modelom požiadavka/odpoveď, čo prináša niektoré nevýhody pre webovské aplikácie. Existuje niekoľko verzií HTTP 0.9, HTTP 1.0, HTTP 1.1 – najnovší, podporovaný vo väčšine serverov a prehliadačov.

URL – Uniform Resource Locator URL je adresa, ktorá jednoznačne identifikuje zdroj v Internete. Nie sú obmedzené len na službu WWW, ale pokrývajú celé spektrum služieb (WWW, FTP, e-mail, telnet). Počas komunikácie servera a prehliadača sa pomocou protokolu HTTP vymieňajú požiadavky používateľa (URL) a odpovede servera (HTML).

2.2.2 Platformy

Náš projekt bude klient/server aplikácia. Preto musíme uvažovať, na akom operačnom systéme bude bežať klientská a serverová časť systému. Pri ďalšej analýze budeme uvažovať, ktoré techniky sú prístupné pre ktorú platformu. Existujú dve veľké rodiny operačných systémov: Unix a Windows.

Windows Do tejto rodiny patria operačné systémy Windows 9x/ME, Windows NT/2000/XP. Tieto operačné systémy panujú v olbasti stolných počítačov vďaka ich pohodlnému používaniu. Majú zabudovný webovský prehliadač Microsoft Internet Explorer. Windows NT podporuje procesory x86, DEC Alpha, MIPS R4000, PowerPC.

Unix Pojem UNIX predstavuje veľkú rodinu operačných systémov. Veľmi významným predstaviteľom tejto rodiny je Linux. Je voľne šíriteľný s otvoreným zdrojovým kódom, čo podporuje jeho rozšírenie. Dnes predstavuje robustné a spoľahlivé riešenie. Podporuje obrovskú škálu procesorov. Bežne sa v jeho distribúciach dodáva prehliadač Netscape Navigator, server Apache, niektoré distribúcie obsahujú dokonca aj databázové servery MySQL alebo PostreSQL. V oblasti serverov je veľmi populárny pre jeho dostupnosť, spoľahlivosť a rýchlosť.

2.2.3 Webovské prehliadače a servery

V predchádzajúcej kapitole sa niekoľkokrát vyskytli slová ako prehliadač alebo webovský server. Je preto vhodné, aby sme sa teraz zaoberali na trhu dostupnými prehliadačmi a servermi – a to predovšetkým z pohľadu rôznych platforiem.

WWW prehliadače

- **Microsoft Internet Explorer** patrí medzi najrozšírenejšie prehliadače na platforme Windows. V súčasnosti je aktuálna verzia 5.5.
- **Netscape Navigator** existuje vo verziách pre platformu Windows aj Unix. Vývojári ocenia XML, JavaScript, HTML4 a CSS podporu. Aktuálnou je verzia 6.0.
- Jeden z novších prehliadačov predstavuje **Opera 5.0**, ktorá ponúka podporu pre väčšinu webovských štandardov (HTML 4, XML, SSL, CSS, JavaScript a DOM). Existuje pod operačné systémy Windows aj Unix.

V ďalšom texte sa budú uvedené technológie skúmať z pohľadu týchto prehliadačov.

WWW servery

- K najrozšírenejším WWW serverom patrí bezpochyby server **Apache** (60% podiel na trhu), ktorého verzia 1.3 existuje pre platformy Windows aj Unix. Je voľne šíriteľný a jeho licencia umožňuje aj komerčné využitie. Má modulárnu architektúru a je ľahko rozšíriteľný. Je robustný a bezpečný.
- Medzi ďalšie serery, ktoré budeme v nasledujúcom texte uvažovať, patria **Internet Information Server** 5.0 (IIS), určený na komerčné nasadenie a **Personal Web Server** (PWS) vhodný pre jednoduchšie a menej rozsiahle aplikácie (prácu off-line). Oba sú určené pre operačný systém Windows. IIS je primárny server pre prostredie Windows NT verzia Server aj Workstation. Dynamické stránky nie sú ponúkané pomocou CGI, ale používa sa programovacie rozhranie Internet Server Application Programing Interface (ISAPI). Aplikácie ISAPI sú v podstate dynamické knižnice. Príklad ich použitia predstavujú aplikácie ASP.

2.2.4 HTML editory

Z dôvodu uniformného výstupu a zjednodušenia tvorby HTML stránok je vhodné sa ešte pred začatím samotných prác rozhodnúť, aký spôsob tvorby použijeme. HTML kód sa dá samozrejme zapisovať pomocou obyčajných textových editorov ako je Notepad v prostredí Windows, či Vim alebo Emacs pod Unixom. Ak však HTML stránky vytvárame často je vhodné uľahčiť si prácu použitím editoru obsahujúceho podporu pre písanie HTML kódu. Z tohto pohľadu môžeme editory rozdeliť do niekoľkých skupín:

Textové editory s podporou HTML Obľúbenými predstaviteľmi tejto skupiny sú editory **HomeSite** a **HTML-Kit**, ktoré boli vytvorené špeciálne pre potreby jazyka HTML. Zvýraznenie syntaxe ponúkajú aj klasické editory ako **Vim** a **Emacs**.

Štrukturálne editory Oproti predchádzajúcej skupine ponúkajú dôležitú vlastnosť navyše – neumožňujú vytvárať syntakticky nesprávnu stránku. Do miesta ktoré editujeme môžeme vložiť iba tie tagy, ktoré sa tam môžu objaviť. Nestane sa nám teda, že stránka obsahuje skrížené elementy alebo chýbajúce ukončovacie tagy. Veľmi dobrým programom z tejto kategórie je **Wedit**. V praxi je možné použiť použiť aj kombináciu editoru Emacs a nadstavby **PSGML**.

WYSIWYG editory Táto kategória zahrňuje editory, s ktorých podporou môžeme vytvárať HTML stránky bez znalosti HTML. WYSIWYG editor sa ovláda podobne ako textový procesor – píšeme text, môžeme meniť písmo, zarovnanie, vkladať obrázky a podobne. Výsledok sa však ukladá vo formáte HTML a počas editácie sa zobrazuje približne tak, ako bude zobrazený v prehliadači. Problémom týchto editorov je, že zaostávajú za vývojom HTML, a tiež to, že generovaný HTML kód je pre následnú ručnú editáciu neprehľadne sformátovaný. Najznámejším editorom z tejto skupiny je Microsoft FrontPage. Podobnú funkčnosť poskytuje aj Netscape Composer.

2.2.5 Kaskádové štýly (CSS)

Cascading style sheets (CSS) je deklaratívny jazyk odstraňujúci niektoré nevýhody klasického HTML. Štýly umožňujú definovať spôsob zobrazenia každého elementu na stránke. Štýl však nie je priamo súčasťou textu stránky, a tak môže byť zápis stránky prehľadnejší a dobre štruktúrovaný. Naviac štýly umožňujú definovať jednotný vzhľad určitého elementu v celom dokumente jedným spôsobom – nemusíme ho opakovať u každého elementu. Pokiaľ štýl uložíme do externého súboru, môže ho využívať viacero stránok naraz. Definícia designu všetkých stránok je potom na jednom mieste. Pri požiadavke na zmenu vzhľadu stránok stačí upraviť štýl a zmeny sa automaticky uskutočnia vo všetkých stránkach. Dobrú podporu CSS majú prehliadače Internet Explorer 4.0 a 5.0, Netscape Navigator 6.0 a Opera. Z týchto dôvodov by bolo vhodné použiť túto techniku aj v nami navrhovanom riešení.

2.2.6 Design webovských stránok

Na tomto mieste uvádzame niekoľko základných pravidiel, ktoré by sa mali dodržať pri návrhu webovského interfejsu celej aplikácie:

Grafický design

- Vhodné kombinácie farieb;
- farba pozadia a textu by mala byť zvolená tak, aby sa text dobre čítal;
- nepoužívať obrázky na pozadí;
- ako farba pozadia je vhodná biela a svetlé odtiene ďalších farieb;
- u všetkých grafických objektov uvádzať alternatívne textové popisy.

Organizácia stránok

- Hypertext umožňuje vytvoriť ľubovolnú štruktúru, ale človek je zvyknutý na konzervatívnejšie usporiadania – väčšinou na hierarchiu;
- na každej stránke by mala byť navigačná lišta;
- jedna stránka by mala obsahovať logický celok informácií;
- stránka by mala byť dobre štruktúrovaná pomocou nadpisov a zoznamov;
- obsah by sa mal oddeliť od vzhľadu a mali by sa použiť CSS.

Odkazy

- Vyhnúť sa techickým pojmom;
- odkazy by mali byť interaktívne (napr. meniť farbu);
- v rámci jedného servera používať relatívne odkazy.

Dokument

- Každá stránka by mala obsahovať identifikačné údaje;
- malo by byť jasné, na ktorom serveri sa nachádzame;
- používať metadata na lepšiu klasifikáciu a vyhľadávanie.

2.2.7 Podpora pre webovské aplikácie

Pokiaľ chceme prostredie webovského prehliadača využiť ako platformu pre tvorbu aplikácie, máme k dispozícii mnoho možností. V zásade sa však dajú rozdeliť na dve skupiny podľa toho, kde sa vykonáva kód aplikácie. Jednu z nich predstavujú aplikácie, ktoré bežia v samotnom prehliadači (klientovi). V tejto skupine sú k dispozícii nasledujúce riešenia:

1. Klientské skripty – JavaScript, VBScript a ďalšie

   Priamo do kódu HTML stránky môžeme zapisovať krátke programy v jednoduchých jazykoch ako je JavaScript, JScript alebo VBScript. Tieto sa využívajú predovšetkým na kontrolu údajov vyplnených používateľom a na zvýšenie interaktivity stránky. Jazyk JavaScript je dokonale štandardizovaný ako ECMAScript. Podpora uvažovaných prehliadačov je nasledovná:

   - **JavaScript 1.0** – Netscape 2.0 a IE 3.0;
   - **JavaScript 1.1** – Netscape 3.0 a IE 4.0;
   - **JavaScript 1.2** – Netscape 4.0 a čiastočne aj IE 4.0;
   - **JavaScript 1.3** – Netscape 4.5 a IE 5.0.
   - **VBScript** pracuje iba v Internet Exploreri.

Dynamické HTML (DHTML) Dynamické HTML stavia na využití klientských skriptov. Navyše je teraz celá stránka a ostatné komponenty prehliadača prístupná pomocou objektového modelu dokumentu (DOM). Obsah stránky sa dá interaktívne meniť aj po jej načítaní, čo umožňuje vytvárať skutočne prispôsobivé užívateľské rozhranie. Dynamické HTML je podporované prehliadačmi Internet Explorer 4.0 a 5.0 ako aj Netscape Navigatorom od verzie 4.0.

Java applety Java applet je program napísaný v jazyku Java, ktorému sa vyhradí časť HTML stránky. Na tejto časti si applet hospodári úplne vo vlastnej réžii. Preto sa applety hodia pre aplikácie, kde je potrebná vysoká interaktivita pri komunikácii s používateľom a klasické klientské skripty už nestačia. Java applet má prístup k väčšine knižníc jazyka Java, a preto môže na komunikáciu s ostatnými aplikáciami používať takmer ľubovoľný protokol a umožňuje vytvoriť multimediálne aplikácie (grafika, animácia, hudba). Použitie appletov je obmedzené na novšie prehliadače, ktoré ich však vo väčšine podporujú.

ActiveX komponenty Použitie ActiveX komponentov je rovnaké ako u Java appletov. Rozdiel je v tom, že ActiveX komponenty sú spustiteľné iba na platforme Windows. Nižšia prenositeľnosť je však vynahradená väčším výkonom oproti Jave.

2. Skriptovanie na serveri

   Druhú skupinu predstavujú riešenia, pri ktorých je aplikácia vytvorená tak, že nejak program na serveri dynamicky podľa potreby generuje HTML kód. Klientské technológie sú v tomto prípade použité prevažne na zlepšenie funkčnosti používateľského rozhrania a pri zvyšovaní interaktivity stránok. Navyše, klientské technológie nemajú prístup k databázam, ktoré sú uložené na serveri. Generovanie stránky v dobe požiadavky tak umožňuje reagovať na konkrétne podmienky a požiadavky používateľa.

Server Side Includes (SSI) SSI sú jednoduché príkazy, ktoré sa zapisujú priamo do HTML stránky ako komentár. Umožňujú vykonávanie jednoduchých činností, ako je vloženie iného súboru do stránky alebo vypísanie dátumu poslednej modifikácie dokumentu. Dnes podporujú SSI všetky bežne používané servery.

CGI skripty CGI skripty majú svoj názov odvodený od rozhrania CGI – Common Gateway Interface. Toto rozhranie definuje spôsob, ktorým webovský server komunikuje s programami a umožňuje jeho prepojenie s ľubovoľným programovacím jazykom. Pokiaľ si používateľ vyžiada od servera stránku, ktorá má byť vygenerovaná CGI skriptom, predá server skriptu parametre zadané používateľom a spustí ho. Skript potom na svojom štandardnom výstupe vygeneruje odpoveď pre prehliadač v jazyku HTML. Táto odpoveď je zachytená webovským serverom a zaslaná prehliadaču. CGI skripty sú najvšeobecnejším a najuniverzálnejším riešením pre webovské aplikácie. Dajú sa písať v ľubovoľnom jazyku – najpoužívanejšie sú príkazové shelly, Perl, C/C+. Ich nevýhodou je však vysoká prevádzková réžia. Pre každú požiadavku musí byť vytvorený nový proces, v ktorom sa spúšťa CGI skript. Preto sa toto riešenie nedá použiť pre servery, na ktoré pristupuje niekoľko desiatok užívateľov súčasne. Podporu pre rozhranie CGI má dnes každý webovský server. Pokiaľ sa na tvorbu skriptov použije jazyk nezávislý na platforme, môžeme výslednú aplikáciu použiť na webovských serveroch v Unixe i vo Windows. Pri použití CGI skriptov by sa mal klásť zvýšený dôraz na bezpečnosť.

FastCGI skripty FastCGI skripty sú snahou o zrýchlenie vykonávania CGI skriptov. Skripty upravené pre toto rozhranie sú pri prvej požiadavke zavedené do pamäti. Ďalšie požiadavky už obsluhuje program, ktorý už je zavedený v pamäti. FastCGI skripty napísané v jazyku C sú dnes asi najvýkonnejším riešením. Sú používané predovšetkým v prostredí Unixu.

ISAPI a NSAPI moduly ISAPI je špeciálne rozhranie pre komunikáciu medzi webovským serverom a aplikáciou. Umožňuje, aby bol program neustále zavedený v pamäti a nemusel sa znovu načítavať a spúšťať pre každú požiadavku. ISAPI sa používa so serverom IIS aj keď toto rozhranie podporujú aj niektoré ďalšie servery. Servery firmy Netsape používajú vlastné rozhranie NSAPI.

Active Server Pages (ASP) ASP je technológia Microsoftu určená pre tvorbu webovských aplikácií. Má modulárnu štruktúru, takže umožňuje použitie takmer ľubovoľného programovacieho jazyka. Microsoft štandardne ponúka VBScript a JScript. Od ďalších firiem sa dajú získať interpreteri Perlu a Pythonu. V ASP sa dajú bez problémov používať COM objekty. Nevýhodou ASP je ich úzke zviazanie s technológiami Microsoftu. ASP beží len v prostredí IIS a PWS.

ColdFusion ColdFusion je vyvinutý Allaire. Jeho syntax je veľmi podobný syntaxi HTML, čo uľahčuje začínajúcim programátorom webovských stránok jeho používanie. Dnes je dostupný aj pre Windows NT aj pre Linux. Jeho nevýhodou je, že nie je voľne šíriteľný.

Personal Home Pages (PHP) PHP je jednoduchý, voľne šíriteľný a pritom výkonný skriptový jazyk. Zapisuje sa priamo do HTML kódu a umožňuje prirodzeným spôsobom kombinovať HTML s príkazmi. PHP sa dá prevádzať na ľubovoľnom webovskom serveri (platforma Windows alebo Unix), aj keď najlepších výsledkov sa dosiahne v spojení so serverom Apache (PHP je skompilované ako modul a je trvale uložené v pamäti). Navyše PHP obsahuje priamo podporu pre prácu s najrozšírenejšími databázovými servermi. Aktuálna je verzia 4.0.

Servlety v Jave Java applety umožňujú beh javovských aplikácií v prehliadači. Podobne servlety zase umožňujú beh Javy na serveri a tým jej využitie pre webovské aplikácie. Servlety predstavujú robustné riešenie a vďaka CORBE môžu komunikovať s ďalšími časťami informačného systému (databázy, ...).

Java Server Pages (JSP) Pri použití JSP sa podobne ako v ASP alebo v PHP priamo do HTML kódu zapisujú príkazy. V prípade JSP sú zapisované v jazyku Java. Špeciáalny servlet zabezpečuje aby bola JSP stránka vždy po svojej modifikácii automaticky prložená do byte-code. To uľahčuje vývoj a na druhej strane je aplikácia značne výkonná, pretože skripty sú vyvolávané už v preloženom tvare. Ide o pomerne novú technológiu, ktorý zataaľ podporujú iba servery od firmy Sun a Apache.

Všetky vyššie uvedené technológie môžeme rozdeliť do dvoch skupín podľa toho, či je pred použitím potrebné previesť program do binárnej podoby alebo či ide o iterpretované jazyky. Pokiaľ vo vývoji aplikácie použijeme jazyk ako C/C++ alebo Java, je potrebné pred použitím aplikáciu skompilovať. To spomaľuje vývoj, ale prináša zrýchlenie počas behu aplikácie. Pri interpretovaných jazykoch ako PHP, JavaScript, Perl je situácia opačná. Rýchlejší vývoj je kompenzovaný pomalším vykonávaním. Úbytok rýchlosti však väčšinou nie je nijak významný vzhľadom na fakt, že skripty webovských aplikácií bývajú pomerne jednoduché.

3. Porovnanie jazykov pre klientské skripty

   V prípade, ak by sme sa nejakú časť aplikácie rozhodli implementovať pomocou skriptov vykonávaných u klienta v prehliadači, je vhodné porovnanie dostupných skriptovacích jazykov.

   JavaScript JavaScript je kompaktný, platformovo nezávislý, na objektovom prístupe založený jazyk na vývoj aplikácií pre Internet. Pôvodne bol navrhnutý firmou Netscape Communications Corporation a po jeho úspechu prišiel aj Microsoft s podobným jazykom pod názvom JScript. Príkazy JavaScriptu sú priamo vložené v HTML stránke a sú interpretované prehliadačom.

   VBScript VBScript je skriptovací jazyk alebo presnejšie skritovacie prostredie implementované podmnožinou programovacieho jazyka Visual Basic. Výsledkom zjednodušenia je produkt, ktorý je ľahko použiteľný. Je to asi aj najväčšia výhoda oproti JavaScriptu. Veľká sila VBScriptu spočíva v možnosti sprístupniť množstvo ActiveX komponentov. To je súčasne dôvodom jeho najvážnejšieho nedostatku – je podporovaný iba v prehliadačoch IE.

4. Porovnanie jazykov pre serverovské skripty

   V prípde voľby skriptov na strane servera ako technológie ktorú bude naše riešenie využívat je potrebné bližšie preskúmať aj jednotlivé jazyky, ktoré prichádzajú do úvahy. Hľadiská, z ktorých sa treba na tieto jazyky pozerať sú nasledujúce:

   - **Výkonnosť** – je zrejmé, že kompilované jazyky sú rýchlejšie ako interpretované. Časť z toho je spôsobená potrebou zavedenia kópie interpretera nevyhnutného na spustenie programu napísaného v skriptovacom jazyku, ako je napríklad Perl. Ďalšie zrýchlenie kompilovaných programov je aj dôsledkom použitia optimalizácií závislých na platforme, pre ktorú je program kompilovaný. Väčšinou však rozdiel vo výkonnosti nie je rozhodujúcim faktorom pre voľbu jazyka.
   - **Bezpečnosť** – vo všeobecnosti najviac bezpečnostných problémov pochádza z programovacích chýb a nedostatočného pochopenia prostredia, v ktorom skript existuje a beží.
   - **Spoľahlivosť** – mal by sa použiť jazyk, ktorého stabilita sa na použitej platforme dokázala. Je vhodné, ak je jazyk stabilný na viacerých platformách.

- **Udržovateľnosť** – toto je často parameter, na ktorý sa berie najväčší ohľad pri voľbe programovacieho jazyka. Veľa programátorov používa interpretované jazyky z dôvodu, že sú ľahsie čitateľné, ľahšie na pochopenie, jednoducho udržiavateľné a testovateľné. Umožňujú inkrementálny, iteratívný vývoj a testovanie namiesto cyklu vytvorenie/modifikovanie – preloženie – testovanie – krokovanie. Tieto faktory môžu dramaticky urýchliť vývoj.
- **Prenositeľnosť** – prenositeľný kód je základom dobre navrhnutého programu. Aj v tomto prípade vychádzajú interpretované jazyky ako dobrá voľba, pretože väčšina interpreterov existuje pre viacero platforiem.

Perl Perl je interpretovaný jazyk optimalizovaný pre prehľadávanie textových súborov a získavanie informácií z týchto súborov a vytvorenie ďalšieho textového súboru na základe týchto informácií. Je tiež vhodný pre množstvo úloh súvisiacich so správou systému. Je vytvorený tak, aby bol skôr praktický (ľahko používateľný, efektívny, kompletný) ako krásny. Je dostupný a stabilný na platformách Unix aj Windows. Spolu s webovským serverom Apache vytvárajú výkonnú a populárnu kombináciu. Jedným z najznámejších projektov využívajúcich tieto dve technológie je projekt mod_perl, ktorý patrí k najstarším a najúspešnejším. Umožňuje vloženie Perl interpretera do Apachu a tým sprístupňuje interné informácie servera programom v perle.

Python Python je prenositeľný, interpretovaný, objektovo-orientovaný jazyk. Má elegantnú syntax a malé množstvo vstavaných vysoko úrovňových údajových typov. Môže byť systematicky rozširovaný pridávaním nových modulov implementovaných v kompilovanom jazyku ako je C alebo C++. Tieto rozširujúce moduly môžu definovať nové funkcie a premenné alebo tiež nové dátové typy. Obsahuje úplnú možinu operácií s reťazcami (vrátane regulárnych výrazov) a odľahčuje používateľa od správy pamäte. Tieto a ďalšie vlastnosti ho robia ideálnym pre vývoj prototypov a ostatných ad-hoc programovacích úloh. obsahuje tiež generický spôsob, ako konvertovať objekt do prúdu bajtov a opačne, čo môže byť použité pri tvorbe roznych distribuovaných modelov. Je k dispozícii na platforme Unix, Windows, DOS, OS/2 a Macintosh a existuje ako modul do webovského servera Apache.

Tcl Tcl je výkonný skriptovací jazyk použiteľný na väčšine dostupných patforiem. Ponúka všetky typické vysoko úrovňové programovacie prostriedky ako sú: reťazce premennej dĺžky, asociatívne polia, zoznamy, štruktúry, regulárne výrazy a schopnosť definovať procedúry priamo počas vykonávania programu. Všetky dátové štruktúry sú dynamicky alokované. Tcl je programovací jazyk určený na vkladanie do iných aplikácií ako konfiguračný a rozširujúci jazyk. Je navrhnutý tak, aby bol ľahko rozšíriteľný pomocou nových primitív napísaných v jazyku C. Tcl používa rovnakú reprezentáciu pre dáta aj pre program, čo umožňuje uloženie programu do premenných, súborov alebo databýzy. Taktiež podporuje udalosťami riadené programovanie a IPC medzi viacerými Tcl aplikáciami.

PHP Jednou z veľa výhod PHP je jeho rýchlosť. Ak ASP používa VBScript, spustí COM objekt. Ak píše niečo, používa iný COM objekt, ďalší, ak pristúpi k databáze, ďalší pre súborový systém, atď. V PHP všetko beží v jednom adresnom priestore, čo znamená väčšiu rýchlosť, lebo nevzniká potreba komunikovať s iným objektom v inom procese.

Oproti Perlu bol PHP vyvinutý pre generovanie dynamických webovských stránok, nie na všeobecné spracovanie textových súborov, a preto má zabudovaných veľa špeciálnych prostriedkov (napr. prístup k objektom HTML stránky cez premenné).

Pre programátorov skúsených v Java/JavaScript/C/C++/Perl je ľahko zvládnuteľná gramatika jazyka, nakoľko je blízka gramatike týchto jazykov. Ďalšou výhodou je integrácia s MySQL. PHP má veľa prostriedkov na správu a udržiavanie MySQL databázy.

2.2.8 Databázové servery

Ako jedna z častí v nvrhovanom riešení bude pravdepodobne vystupovať databáza. Je preto vhodné vykonať anlýzu aj v tejto oblasti. Medzi najrozšírenejšie produkty v tejto oblasti patria Oracle, MS SQL Server, Sybase, Progress. Vzhľadom na fakt, že sa jedná o drahé softvérové produkty, zameriame sa skôr na analýzu voľne šíriteľných databázových serverov, medzi ktoré patria napríklad MySQL alebo PostgreSQL.

MySQL MySQL je jednoduchý databázový SQL server. Je priamo podporovaný PHP. Umožňuje na jednom počítači pracovať s viacerými databázami. Existuje pre platformu Windows aj Unix. Medzi hlavné výhody patrí:

- schopnosť obsluhovať neobmedzené množstvo používateľov súčasne;
- kapacita na uchovanie 50 000 000 záznamov;
- veľmi rýchle vykonávanie príkazov;
- ľahký a efektívny systém používateľských privilégií.

MySQL má navyše širokú používateľskú základňu, čo zabezpečuje, že kód je dobre testovaný.

PostgreSQL PostgreSQL je databázový server takmer úplne kompatibilný so štandardom SQL92. Podporuje teda väčšiu podmnožinu jazyka SQL ako MySQL. Sever beží na Unixe, klientské programy existujú pod Unix aj Windows. Podporuje vnorené príkazy SELECT, pohľady a transakcie. Obsahuje interfajsy pre C/C++, Perl, PHP, Python, TCL, bash, ODBC. Jeho sila spočíva vo veľkej rozširovateľnosti: používateľom definované údajové typy, operátory, funkcie. Taktiež podporuje dedenie a má bohatú množinu dátových typov: boolean, časové intervaly, viac dimenzionálne polia a ľD geometrické typy. Navyše je dobre zdokumentovaný. Jeho najväčšou nevýhodou oproti MySQL je rýchlosť. Ďalšou nevýhodou je fakt, že nie je hierarchický – databýzy môžu obsahovať iba tabuľky a nie ďalšie databázy. Podobne ako MySQL aj PostgreSQL je priamo podporovaný PHP.

2.2.9 Závery technologickej analýzy

Pre náš projekt sme si vybrali kombináciu produktov uvedenú v tabuľke 2.
Hlavné kritériá nášho výberu v poradí podľa dôležitosti boli:

- Platformová nezávislosť, prenositeľnosť;

webowský server	**Apache**
skriptovanie na strane servera	**PHP**
skriptovanie na strane klienta	**JavaScript**
databáza	**MySQL**

Tabuľka 2: Vybrané produkty pre náš systém

- efektívnosť;
- nízka cena tvorby a používania;
- jednoduchá implementácia.

Webovský server Apache je dostupný na obidvoch platformách (dokonca bezplatne) a má veľmi dobré ukazovatele efektívnosti.

Jazyk PHP je priamo podporovaný serverom Apache ako jeho modul pre obidve platformy a je najrýchlejším jazykom pre interpretované skripty. Programátori nášho tímu už majú skúsenosti s jeho používaním.

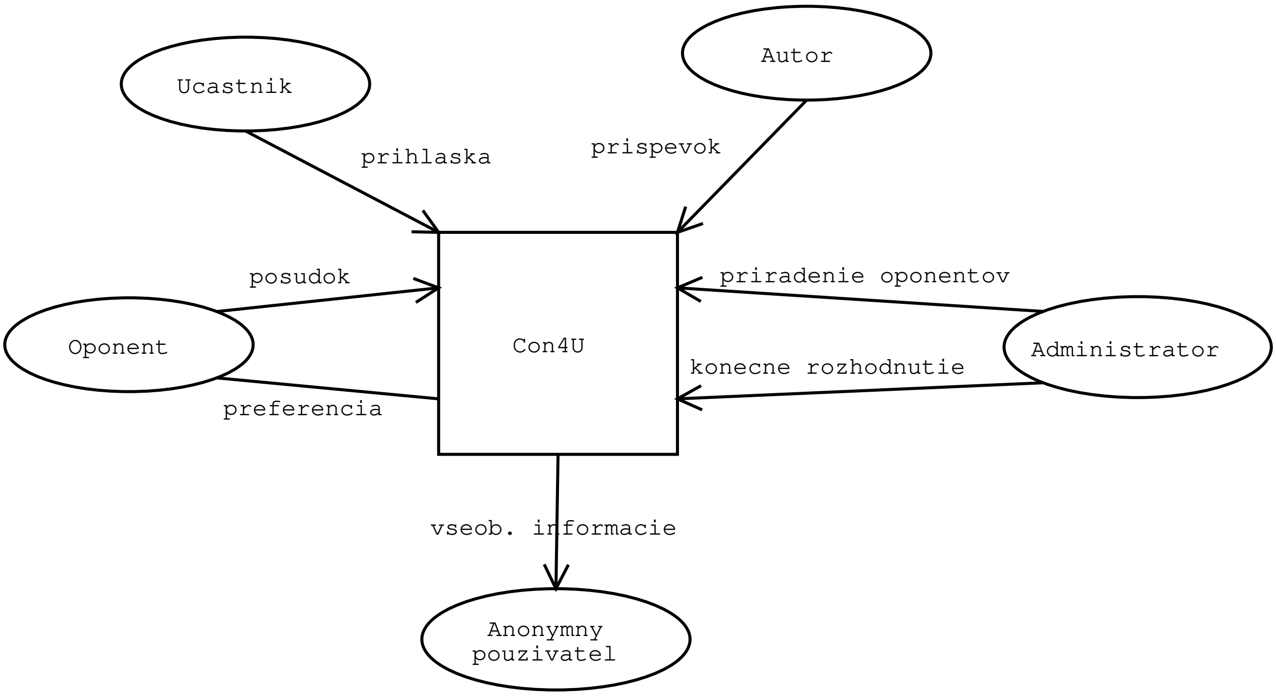
Databázový systém MySQL, tak isto ako PHP a Apache, je platformovo nezávislý. Je veľmi rýchly aj pri väčšom množstve dát. Všetky vyššie uvedené prostriedky možno získať bezplatne.

JavaScript je jediný skript na strane klienta, ktorý je podporovaný všetkými analýzovanými prehliadačmi.

3 Špecifikácia

3.1 Kontext systému

Obrázok 1 zobrazuje kontext systému – jeho okolie a ľudí, ktorý s ním prichádzajú do kontaktu. Systém je znázornený obdĺžnikom uprostred, okolné entity oválmi, vstupy a výstupy šípkami.



Obrázok 1: Kontext systému

Kontext vyplýva v mnohom zo zadania – máme tu typických používateľov systému (**účastník**, **oponent**, **anonýmny používateľ**, **administrátor** a **autor** príspevku), ako aj údaje, ktoré jasne vyplývajú z požadovaných funkcií systému. Ak má vedieť systém prijímať prispevky od jednotlivých autorov, objaví sa tu vstup nazvaný **príspevok**. Podobne od účastníka konferencie sa prijíma **prihláška**. Oponent zadáva svoje **preferencie**, čo sa týka kľúčových slov (tie popisujú jednotlivé témy) a zasiela **posudok** na pridelený príspevok. Anonymný používateľ sa dostane iba k **všeobecným informáciam**. Napokon administrátor systému – čo je vlastne predseda komisie – má rozhodujúce slovo pri **pridelení oponentov** na príspevky a vykonáva **konečné rozhodnutia** vo väčšine riešených otázok.

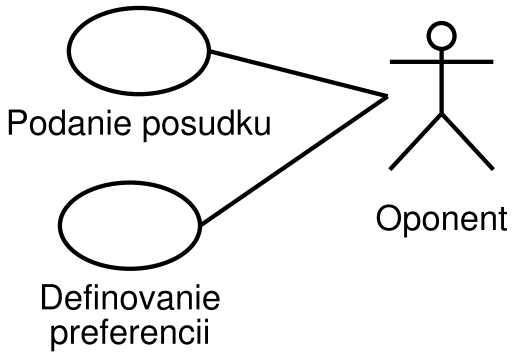
Funkcie, cez ktoré jednotliví používatelia pristupujú k systému, sú popísané v nasledujúcej časti, potom nasleduje popis modelu údajov a napokon sú špecifikovaní používatelia – tým by sa mali vyjasniť pojmy, ktoré sú v kontextovom diagrame použité.

3.2 Funkcie systému

3.2.1 Funkcie systému z pohľadu oponenta

Systém bude realizovať rozhranie, pomocou ktorého sa budú môcť oponenti:

- Prihlásiť a definovať svoje preferencie. Preferencie oponenta vyjadrujú znalosti v danej

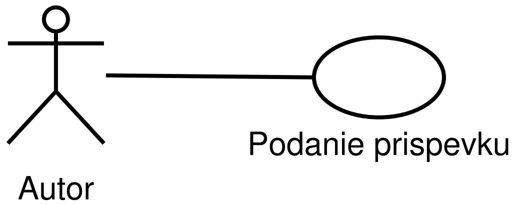


Obrázok 2: Akcie oponenta

oblasti, ktorá je definovaná kľúčovými slovami. Tieto preferencie bude možné vybrať z preddefinovanej množiny tém.

- Podanie vlastného posudku na pridelený článok. Pri tejto akcii oponent zároveň vyplní hodnotiace kritéria, ktoré budú stanovené pri konfigurácii systému.

3.2.2 Funkcie systému z pohľadu autora

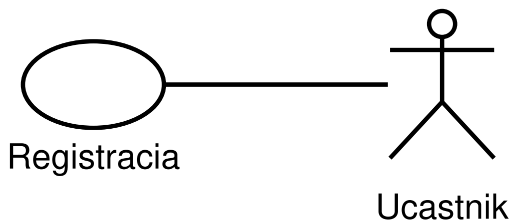


Obrázok 3: Akcie autora

Rozhranie v prípade prístupu autora bude umožňovať:

- Podanie príspevku do systému. Pri podávaní sa zároveň zvolí téma príspevku – bude možné vybrať si z definovaných kľúčových slov, ktoré vhodne popisujú obsah príspevku.

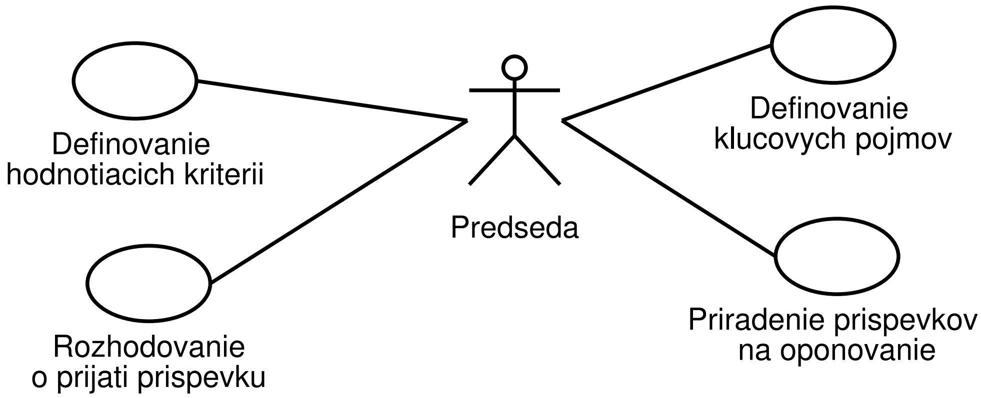
3.2.3 Funkcie systému z pohľadu účastníka



Obrázok 4: Akcie účastníka

Systém bude poskytovať možnosť podávania prihlášky na konferenciu prostredníctvom internetu. Toto bude umožnené vytvorením rozhrania pre vyplnenie registrácie účastníka konferencie.

3.2.4 Funkcie systému z pohľadu predsedu



Obrázok 5: Akcie predsedu

Funkcie systému pre predsedu konferencie budú umožňovať:

- Definovanie kľúčových pojmov konferencie. Pri tejto akcii bude definovať kľúčové slová, ktoré charakterizujú jednotlivé tématické okruhy konferencie.
- Definovanie hodnotiacich kritérií umožňuje definovať kľúčové slová, ktoré budú vhodne pomenovávať jednotlivé hodnotiace kritéria.
- Priradenie príspevkov na oponovanie. Pri tomto úkone predseda bude môcť priradiť prijaté články prihláseným oponentom, pričom mu môže pomôcť systém na základe definovaných tém článku a preferencií oponenta.
- Rozhodonvanie o prijatí príspevku. V tejto akcii bude predseda rozhodovať o sporných článkoch, ktoré neboli automaticky prijaté, prípadne zamietnuté.

3.3 Model údajov

3.3.1 Popis jednotlivých dátových entít:

Clovek abstraktná dátová entita, ktorá bude slúžiť iba ako entita, ktorá definuje základné atribúty pre identifikáciu používateľa systému.

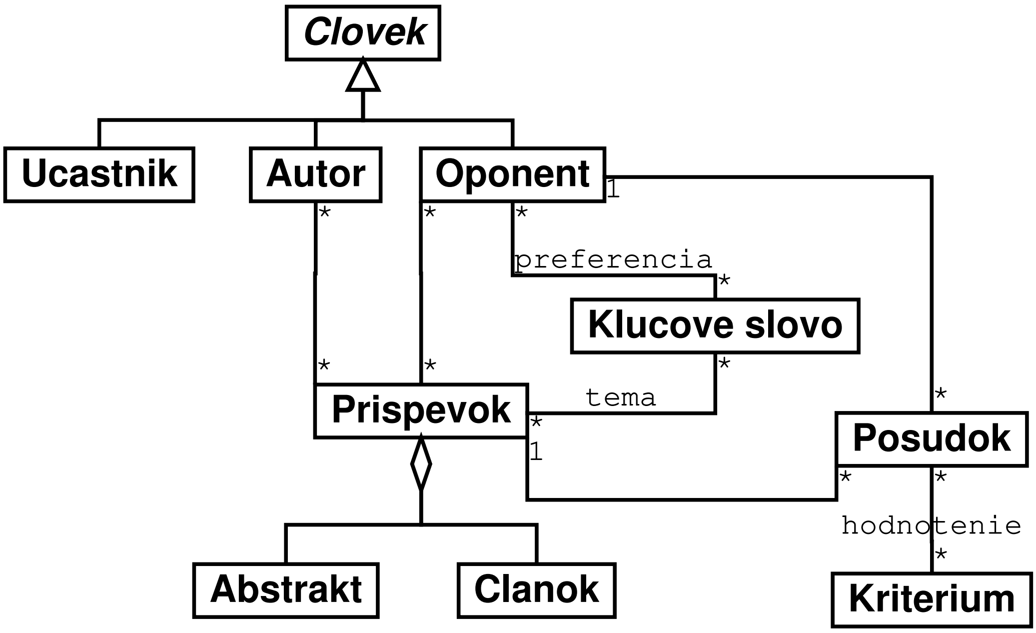
Ucastnik rozširujúca entita entity *Clovek*. Táto entita je iba obohatená o niektoré atribúty nutné pre evidenciu účastníkov konferencie.

Autor dátová entita rozširujúca entitu *Clovek*. Bude obsahovat údaje o autoroch článkov.

Oponent entita rozširujúca entitu *Clovek*, podobne ako *Autor*. Jej atribúty budú presnejšie definovať oponenta.

Klucove slovo ide o dátovú entitu, ktorá pomenováva tématické okruhy konferencie a zároveň preferencie oponentov. Tieto dve navzájom rôzne entity bolo nutné vsunúť do jednej, kvôli automatickému priraďovaniu príspevkov k oponentom.

Prispevok dátová entita, ktorá obsahuje príspevok autora, pre systém bude viditeľný aspoň abstrakt a samotný text článku.



Obrázok 6: Model údajov v systéme vedeckej konferencie

Posudok dátová entita obsahujúca text posudku na daný článok.

Kriterium dátová entita obsahujúca textový popis kritéria, ktoré by mal spĺňať príspevok.

3.3.2 Popis relácií:

Autor – Prispevok Definuje vzťah, že autor napísal niekoľko príspevkov, prípadne ešte ani jeden. V opačnom smere vyjadruje táto relácia možnosť napísania príspevku aspoň jedným autorom, prípadne kolektívom autorov.

Prispevok – klucove slovo: tema Definuje priradenie jednému príspevku niekoľko možných tématických okruhov – kľúčových slov. Pre príspevok nie je nutné definovať tému (priradiť mu kľúčové slová), no je to vhodné. Jedno kľúčové slovo môže byť použité na identifikáciu niekoľkých príspevkov, z čoho vyplýva možnosť mať v systéme niekoľko článkov na tú istú tému. Samozrejme nie je nutné priradiť príspevkom všetky kľúčové slová.

Prispevok – oponent Definuje vzťah medzi oponentom a zaslaným príspevkom. Vyjadruje priradenie príspevku aspoň jednému, prípadne viacerým oponentom. Oponent nemusí mať priradený ani jeden príspevok.

Oponent – klucovne slovo: preferencie Vyjadruje vzťah medzi oponentom a jeho preferenciami. Svoje preferencie si oponent zvolí podľa toho, aké tématické okruhy ovláda a je schopný ich oponovať.

Oponent – posudok Popisuje, že daný oponent napísal posudok. Je možné, aby oponent napísal niekoľko rôznych posudkov, prípadne žiaden. Každý posudok má definovaného iba jedného oponenta, nemôže existovať jeden a ten istý posudok od viacerých oponentov.

Posudok – prispevok Opisuje vzťah medzi posudkom a príspevkom. Jeden príspevok môže mať niekoľko rôznych posudkov, prípadne ani jeden. Posudok môže byť priradený nanajvýš jednému príspevku.

Posudok – kriterium: hodnotenie Táto relácia vyjadruje vzťah medzi posudkom a dátovou entitou kritéria – ide vlastne o hodnotenie – pričom táto relácia definuje aj výšku samotného hodnotenia. Nie je striktne povedané, že je nutné používať hodnotenie – vtedy posudok nebude mať priradené ani jedno kritérium. V opačnom prípade je možné priradiť všetky kritéria k posudku.

3.4 Používatelia systému

Systém Con4U budú cez webovské rozhranie používať používatelia, ktorý budú vystupovať v rôznych rolách. V ďalších častiach budeme opisovať možné roly používateľov tohto systému.

Každý používateľ (okrem anonymného používateľa) bude mať svoje prihlasovacie údaje a pre prácu so systémom sa bude pomocou nich autentifikovať. Jedna osoba môže disponovať viacerými rolami – ak bude nejaký používateľ súčasne administrátorom aj oponentom, bude mať priradené len jedno prihlasovacie meno a heslo.

3.4.1 Anonymný používateľ

Anonymný používateľ má možnosť si prezerať len statické web-stránky o konferencii, ktoré v skutočnosti ani nebudú pod správou Con4U. Tieto môžu zahŕňať informácie o samotnej konferencii (program, niektorí účastníci, sponzori, organizátori a podobne) a aj informácie konkrétnejšieho charakteru, ktoré môžu mať podobný dizajn ako stránky spravované Con4U systémom a môžu s nimi byť aj hypertextovo zviazané, ale nie sú ním generované.

Okrem toho môže byť systém nastavený tak, že si anonymný používatelia môžu prezerať aj niektoré informácie o prijatých článkoch – napríklad mená autorov a názvy článkov, alebo aj abstrakty. Táto možnosť môže byť zapnutá pri nastavovaní konfiguračných parametrov systému administrátorom.

O anonymných používateľoch systém nebude evidovať žiadne informácie. Anonymní používatelia nemôžu meniť žiadne dáta v systéme a nemôu si prezerať žiadne iné dáta v systéme ako tie, ktoré boli vopred povolené administrátorom.

3.4.2 Účastník konferencie

Úcastník konferencie zadáva osobné údaje prípadne iné údaje do formulára. Tieto údaje použíjú organizátori pri plánovaní organizačných detailov konferencie. Niektoré položky (napríklad týkajúce sa platenia poplatkov za účasť) sa vzťahujú síce k účastníkovi konferencie, vypĺňať ich však bude administrátor.

O účastníkoch konferencie bude systém evidovať údaje (ako je popísané v 3.3). Časť týchto údajov bude do systému vkladať sám účastník, časť bude zapisovať administrátor. Účastník si môže prezerať a modifikovať len údaje o svojej prihláške na konferenciu – žiadne iné.

3.4.3 Autor príspevku

Autor príspevku do vedeckej konferencie bude pri podávaní článku zadávať svoje osobné údaje, údaje o článku a článok samotný. Tento článok bude posudzovaný oponentami a prijatý alebo neprijatý na konferenciu. Ako údaje o článku bude autor vyznačovať kategórie – kľúčové slová, ktorých sa jeho článok týka. Zoznam týchto kľúčových slov bude konfigurovateľný a bude ich zadávať administrátor.

Autor si môže prezerať a modifikovať okrem svojich osobných údajov ešte údaje o svojich článkoch. Nebude môcť prezerať žiadne iné údaje okrem tých, ktoré si bude prezerať ako anonymný používateľ.

3.4.4 Oponent – člen programovej komisie

Člen programovej komisie si na začiatku práce so systémom vyberie z preddefinovaných kategórií (kľúčových slov) svoje preferencie. Podľa týchto preferencií mu bude systém prideľovať príspevky na oponovanie. Toto prideľovanie príspevkov na oponovanie bude vykonávať administrátor, Con4U mu v tom však bude radiť vytváraním zoznamov možných oponentov podľa poradia najväčšej zhody oponentových preferencií a kľúčových slov článku.

Na samotné recenzovanie pridelených článkov bude mať možnosť si všetky tieto články zo systému stiahnuť. Po ich preštudovaní oponent vyplní formulár posudku s preddefinovanými otázkami, týkajúcimi sa rôznych kritérií kvality článku, napr. originalita, správnosť, forma, zhoda s predmetom konferencie a podobne. Nakoniec určí mieru svojho odporúčania/neodporúčania prijatia tohoto článku do programu konferencie.

Oponent nebude poznať ostatných oponentov priradených k článku a nebude vidieť ani hodnotenie článku ostatnými oponentami. Bude si môcť prezerať, prípadne modifikovať len svoje vlastné posudky.

3.4.5 Administrátor – predseda programovej komisie

Administrátor na začiatku nakonfiguruje systém pre danú konferenciu – určí možné kľúčové slová (oblasti) článkov, nastaví možnosti prezerania dát pre anonymných používateľov, priradí používateľov do rolí členov programovej komisie.

Počas etapy prijímania článkov a preferencií oponentov bude sledovať pridávané články, zjavne nevhodné príspevky (žarty, omyly, viacnásobne sa opakujúce články a podobne) bude rušiť.

Ďalšou úlohou predsedu programovej komisie je prideľovanie článkov jednotivým oponentom na posudzovanie. Túto činnosť mu bude Con4U uľahčovať generovaním zoznamov vhodných oponentov k článkom (prípadne aj zoznamov vhodných článkov k oponentom) podľa miery zhody kľúčových slov článku a preferencií oponentov.

Počas bežiacej etapy posudzovania článkov bude sledovať progres jednotlivých oponentov pri posudzovaní vo forme tabuliek s počtom ukončených a chýbajúcich posudkov a jeho úlohou bude pripomenúť zaostávajúcim oponentom, že by mali dôslednejšie plniť svoju úlohu.

Po skončení etapy posudzovania bude mať predseda spolu s členmi programovej komisie k dispozícii generované poradia úspešností článkov podľa rôznych kritérií. Ich spoločnou úlohou bude určiť budú prijaté, a ktoré zamietnuté.

Po tom, ako sa programová komisia dohodne na zozname prijatých článkov, predseda tieto články vyznačí v systéme Con4U a systém to oznámi mailami autorom článkov. Generovaním zoznamov prijatých článkov sa úloha administrátora a aj celého systému Con4U končí.

4 Návrh

4.1 Hrubý návrh dátového modelu

Hrubý návhr dátového modelu je oproti modelu z kapitoly 3.3 obohatený o významné atribúty jednotlivých dátových etít. Zároveň boli rozbité niektoré asociačné väzby, nakoľko aj tie obsahujú isté atribúty potrebné pre presnejší opis vzťahu. Zároveň sa tým eliminovali nežiadúce vzťahy M ku N. Vzťahy M ku N, ktoré nenesú v asociácii žiadne atribúty okrem cudzích kľúčov, boli ponechané bez zmeny.

4.1.1 Popis atribútov neväzobných dátových entít

Dátové entity ktoré boli opísané vo fáze analýzy a špecifikácie, sa v tejto časti už opisovať nebudú a dátové entity, ktoré vznikli až vo fáze hrubého návrhu, budú stručne opísané, aj keď budú mať rovnaký význam ako nimi nahradzované asociácie.

Clovek

meno, priezvisko skutočné meno osoby, ktorá chce používať systém ako účastník, autor, prípadne oponent;

e-mail kontaktnú e-mailovú adresu danej osoby;

heslo prístupvoé heslo do systému.

Prispevok

abstrakt abstrakt príspevku – ten je nutný vyplniť extra, aby mohol systém vytvárať program konferencie;

clanok celý článok autora;

dátum prijatia dátum skutočného prijatia príspevku;

stav atribút identifikujúci stav príspevku (prijatý na konferenciu alebo zamietnutý, pripravená alebo nepripravená prezentácia, odprezentovaný príspevok, atď...).

Klucove slovo

meno jednoduchý názov kľúčového slova;

popis presnejší opis významu kľúčového slova.

Posudok

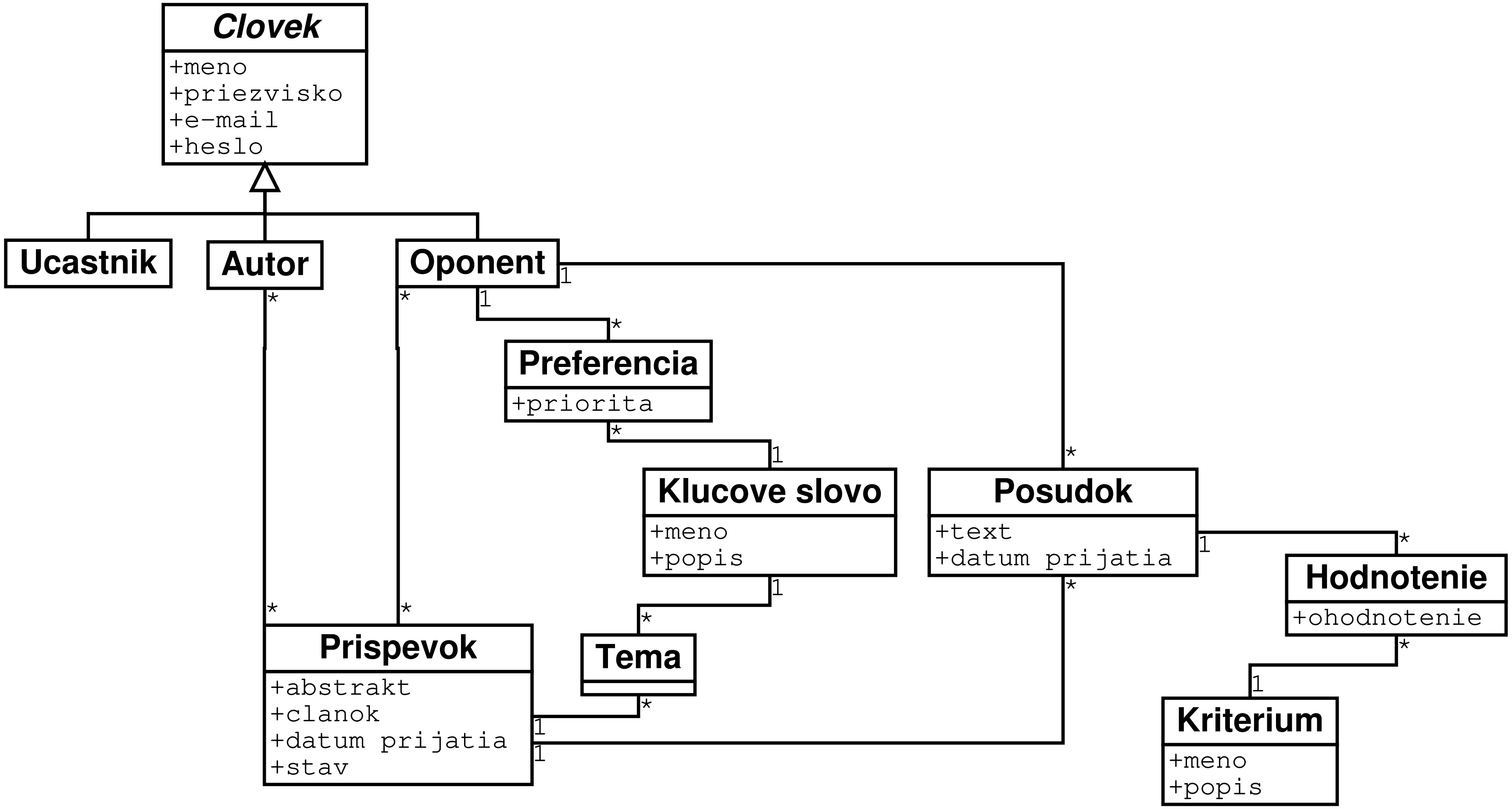
text obsah posudku, ktorý napísal oponent na článok;

dátum prijatia dátum skutočného prijatia posudku.

Kriterium

meno jednoduchý názov kritéria;

popis presnejší opis významu daného kritéria.



Obrázok 7: Hrubý návrh dátového modelu

4.1.2 Popis atribútov väzobných dátových entít

Preferencia dátová entita vyjadrujúca priradenia jednotlivých kľúčových slov – preferencií jednotlivým oponentom.

priorita určuje číselne vyjadrenie znalosti danej preferencie oponenta.[6]

Tema dátová entita určujúca vzťah medzi článkom a kľúčovým slovom – témou. Každý článok spadá pod jednu alebo niekoľko tém konferencie, autor nemusí túto väzobnú entitu nutne vytvárať (cez web-rozhranie aplikácie). Toto bude mať za následok, že programová logika nebude vedieť automaticky priradiť článok konkrétnemu oponentovi podľa preferencií.

Hodnotenie dátová entita určujúca vzťah medzi posudkom a jednotlivými hodnotiacimi kritériami, zadaných predsedom komisie.

ohodnotenie číselne vyjadrenie miery splnenia daného kritéria, na ktoré odkazuje hodnotenie.

4.2 Architektúra systému

Systém Con4U je vystavaný na architektúre klient-server, pričom väčšina funkčnosti je viazaná na jednotlivé HTTP požiadavky. Web-rozhranie bude ťažiskom celého systému – to spolu s použitím jazyka PHP určuje architektúru. Jednotlivé funkcie systému budú spúšťané požiadavkami od klientov, vykoná ich PHP modul bežiaci v adresnom priestore HTTP servera (v našom prípade je to Apache), pričom priamo z tohto procesu sa bude pristupovať na databázu. Je to technika vyskúšaná na množstve projektov obdobného typu – a nakoľko náš systém nepatrí do kategórie rozsiahlych projektov, je to dostatočné riešenie.

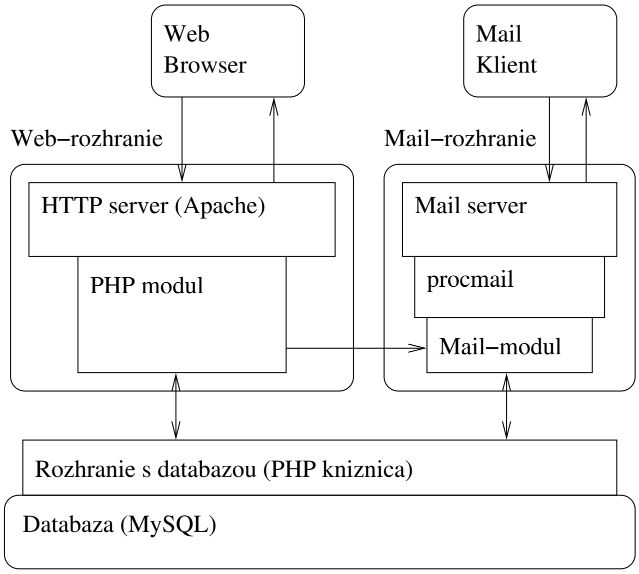
Systém obsahuje aj poštový modul, lebo pre mnohých používateľov môže byť mail prijateľnejšou formou pre odosielanie príspevkov, ale aj pre registráciu. Rovnako tak bude možné posielať e-mailom potvrdenie o zaplatení účastníkovi (ak si to bude želať), alebo informovať autora o prijatí (resp. neprijatí) jeho článku. Aj tu sú všetky činnosti riadené udalosťami.

Architektúra je znázornená na obrázku 8. Ako je vidieť, používateľ môže pristúpiť k systému buď pomocou prehliadača WWW, alebo využije mail klienta. Kým cez web-rozhranie sú prístupné všetky funkcie systému, funkcie mail-rozhrania budú zrejme obmedzenejšie. Hlavným obmedzením je, že funkcie administrácie sa cez toto rozhranie nedajú vykonávať. Toto je ale pochopiteľné obmedzenie – administrátor (predseda komisie, napr.) má obvykle systém prístupný po lokálnej sieti a nemal by mať žiadny technický problém pri využívaní web-rozhrania. Druhým dôvodom je potom aj bezpečnosť. Napokon – odozva mail-rozhrania nie je v intenciách toho, čomu sa hovorí komunikácia v reálnom čase – a tá je zrejme pri nastavovaní parametrov konferencie prakticky nutná. Administrátor potrebuje hneď vidieť, či je všetko nastavené tak, ako to vyžaduje.

Popíšme si teraz v krátkosti jednotlivé znázornené celky a komunikáciu medzi nimi.[7] Web browser (jeho používateľ) zasiela vo všeobecnosti žiadosť o zobrazenie nejakej stránky, na ktorej môže byť aj formulár. Po jeho vyplnení je tento spracovaný modulom web-rozrania a údaje, ktoré majú byť perzistentné, sa uložia do databázy. Veľmi podobná je komunikácia

[6] V systéme sa bude používať automatické priraďovanie číselných hodnôť na základe zotriedeného vstupu preferencií od oponenta.

[7] Podrobnejší opis nasleduje v časti 4.3.



Obrázok 8: Architektúra systému

mailom – používateľ napríklad dostane predpripravený formulár, ktorý vyplní a odošle. Náš systém všetko spracuje a opäť uloží do databázy. Odpoveďami sú buď ďalšie web-stránky, alebo maily. Je možné si povšimnúť, že PHP modul v HTTP serveri komunikuje s Mail-modulom. K tomu dochádza, ak užívateľ cez web vykoná akciu, ktorá je potvrdzovaná mailom[8], alebo napríklad cez web požiada o zaslanie mailového formulára. Opačná komunikácia nemá samozrejme zmysel.

PHP modul vo web-interface sa zaoberá len web-rozhraním, mail-modul zase iba mail-rozhraním. Spoločné funkcie pracujúce s databázou budú v jednej knižnici, ktorá bude využívaná oboma modulmi. V tejto by mali byť ukryté všetky potrebné funkcie pre databázu – knižnica by mala obaľovať všetky SQL príkazy.

4.3 Moduly systému

Ako bolo naznačené v časti o architektúre, systém má tri hlavné celky:

- modul **Web-rozhranie**;
- modul **Mail-rozhranie**;
- modul **Databázové rozhranie**.

4.3.1 Web-rozhranie

Web-rozhranie zabezpečuje komunikáciu používateľa so systémom pomocou prehliadača WWW. Technológie, ktoré použijeme v tomto module, už boli spomenuté: HTTP server **Apache** a modul **PHP**. PHP stránky budú zabezpečovať interaktívnu komunikáciu s používateľom. Implementácia bude realizovaná ako vo forme PHP stránok, tak aj vo forme knižnice komponentov.

[8] Nemusí ísť nutne o toho istého používateľa. Napríklad, keď predseda v role administrátora prijme príspevok, autorovi príspevku sa pošle e-mailom potvrdenie (ak si tak zvolil).

Knižnica komponentov bude vo forme jednoduchých funkcií implementovať väčšinu vizuálnych prvkov stránky – ide najmä o prvky formulára. Identifikovali sme napríklad potrebu častého pohľadu na obsah tabuľky z databázy vo forme tabuľky v HTML. Tento pohľad je vo väčšine prípadou veľmi podobný, takže vytvoríme komponentu, ktorá na požiadanie vráti hotový kód HTML – ten potom už len vložíme do stránky.[9]

Bezpečnosť bude realizovaná technikou cookies, pričom mechanizmy pre prácu s reláciou si zrejme vytvoríme všeobecne tak, aby sme nemuseli na každej PHP stránke explicitne pracovať s cookies (opäť to budú funkcie v knižnici).

Komunikácia s databázou je realizovaná výhradne cez modul Databázové rozhranie.

4.3.2 Mail-rozhranie

Pre tento modul je potrebné zvoliť vhodnú technológiu umožňujúcu automatické spracovanie mailov. Jednoduchým a súčasne dostatočne vhodným nástrojom je **procmail**, ktorý umožňuje jednoduchým spôsobom poslať obsah mailu zvolenému skriptu. Ak sa ukáže, že PHP je vhodný na úlohy, ktoré budú mať skripty vykonávať, zvolíme tento jazyk, aby sme použili jednotný jazyk vo všetkých moduloch. Ako prakticky istou alternatívou je potom Perl. Rozhodnutia o implementačnom prostredí pre mail-modul však vykonáme neskôr.

Spomínané skripty zabezpečujú odosielanie mailov na požiadanie web-rozhrania, spracovanie prišlých mailov, ukladanie všetkých potrebných informácií do databázy, ako aj ich vyberanie pre prípad generovaných a odosielaných mailov. Komunikácia s databázou je samozrejme výhradne cez modul rohrania s databázou.

Tento modul vyžaduje SMTP server a procmail – oboje nainštalované na cieľovom prostredí.

4.3.3 Databázové rozhranie

Myšlienka databázového rozhrania je veľmi jednoduchá – zabaliť všetky príkazy v jazyku SQL do funkcií. Volanie funkcií bude v kóde, ktorý toto rozhranie využíva, vyzerať lepšie, bude kratšie – a potenciálne umožňuje zmeniť databázu bez zmeny vyšších modulov.[10] Všetky prvky jazyka SQL sa týmto izolujú do jedného kompaktného modulu. Môžme vytvoriť elegantné funkcie na prácu s asociačnými tabuľkami, ktoré popisujú vzťahy M ku N. Využívanie modulu by malo výrazne zprehľadniť kód v moduloch rozhraní.

Modul neslúži na ukrývanie tabuliek, atribútov, a podobne – súvisí to s tým, že mapovanie dátového modelu do tabuliek je veľmi priamočiare, takže ho netreba zapúzdrovať do iného modelu.

4.4 Návrh formulárov pre web-rozhranie

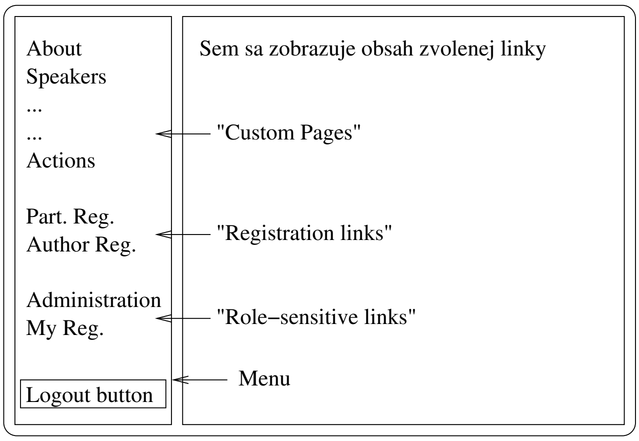
Na web stránke konferencie je možné rozlíšiť niekoľko typov informácií. Ide o statické stránky, ktoré informujú o konferencii, pričom ich paleta sa mení konferencia od konferencie.

[9] Ak ostaneme u tohto príkladu, komponenta *tableView* by ako parametre prijímala *meno tabuľky*, ďalej *zoznam stĺpcov*, ktoré budú (v zvolenom poradí) zobrazené, a napokon informáciu o *štýle* vykreslenia tabuľky. Výsledkom bude spomínaný HTML kód tabuľky.

Štýl vykreslovania môže napríklad hovoriť o tom, či bude tabuľka len na prezeranie, alebo bude možné s položkami vykonávať isté operácie – napríklad v každom riadku sa vykreslí checkbox a dole tlačítko so zvoleným textom. Čo sa vykoná, na tom tejto komponente nemusí záležať, ona len vykreslí tabuľku a doplní ju prvkami formulára, ak to bude požadované.

[10] Je známe, že hoci je SQL do istej miery štandard, rôzne databázy majú rôzne pomenovania typov napríklad pre binárne dáta – týmto sa tieto problémy elegantne obchádzajú.

Ďalej ide o linky k niekoľkým štandardným formulárom – *Registrácia účastníka* (**Participant Registration**), *Registrácia autora* (**Author Registration**). Nasledujú linky, ktoré by mali byť prístupné podľa jednotlivých rolí: pre administrátora linka *Administrácia* (**Administration**), pre oponenta *Moje posudky* (**My Reviews**), pre autora *Moje príspevky* (**My Papers**), a napokon každý bude mať linku *Moje registračné informácie* (**My Registration**). Všetky linky budú v konečnej verzii v anglickej podobe, a naďalej budeme používať len anglický tvar. Samozrejme je možné samotný text linky zmeniť. Všetky dôležité linky budú vložené do ľavého menu, ako je znázornené na obrázku 9.



Obrázok 9: Návrh obrazovky web-rozhrania

Podobne ako iné konferenčné systémy, aj my sme zvolili model, kedy je menu vľavo a obsah sa premieta do pravej hlavnej časti obrazovky. Ako je vidieť na obrázku, menu obsahuje niekoľko liniek, pričom sa začína stránkami, ktorých povaha je informačná, potom sú linky pre registráciu, a nakoniec nasledujú linky závislé na roli prihláseného používateľa. Tlačítkom *Logout* sa používateľ odhlasuje. V prípade, že používateľ nie je prihlásený, miesto poslednej skupiny je na obrazovke ponúknutá možnosť prihlásenia (vstupné riadky *Login name* a *Password* a tlačítko *Login*).

V systéme Con4U je možné konfigurovať všetky linky na informačné stránky. Tieto stránky sú nazvané termínom *Custom pages*. V administrácii je možnosť nastaviť mená liniek, ich poradie, ako aj natiahnúť obsah stránky do systému. Je dokonca možné nastaviť viditeľnosť stránky pre jednotlivé roly, pričom toto nastavenie nie je možné jednoducho obísť, lebo všetky custom-stránky sú prístupné cez jednu PHP stránku, ktorá kontroluje prístup ku custom-stránkam.

V nasledujúcich kapitolách si popíšeme funkcie formulárov prístupných cez jednotlivé uvedené linky. Každý formulár (označený ako typ FORM) bude krátko opísaný a budú vymenované prvky, ktoré sa na ňom nachádzajú. Pre každý prvok bude uvedený typ komponenty, ktorou bude na formulári predstavovať daný prvok. Možné komponenty sú: TEXT, CHECKBOX, SELECT, RADIO, PASSWORD, TEXTAREA – teda typické prvky formulárov jazyka HTML. BUTTON je tlačítko, ktorého akcia bude popísaná priamo na mieste. Komponenta LIST je zoznam hodnôt, z ktorých je možné jednu vybrať.

Navyše k týmto budeme používať komponentu TABLE, ktorá bude veľmi podobná PHP komponente *tableView* spomínanej ako typický príklad PHP komponenty z modulu web-rozhrania. Pre tabuľku budú vymenované jej stĺpce, ktoré budú označené typom ATTRIBUTE. Okrem toho využijeme aj atribút *Select* (typ ATTRIBUTE-CHECKBOX), ktorý slúži na označovanie údajov tabuľky, aby s nimi bolo možné realizovať akcie (napríklad mazať označených používateľov).

LINK je linka na iný udaný formulár, pričom môže byť podľa potreby kombinovaná s iným atribútom (BUTTON-LINK je tlačítko na iný formulár, ATTRIBUTE-LINK je atribút tabuľky, ktorý je súčasne linkou, ...).

4.4.1 Štandardné formuláre

Confirmation (FORM) Formulár určený na potvrdzovanie akcií, ktoré si potvrdenie vyžadujú (napríklad mazanie používateľov). Obe tlačítka na formulári slúžia ako LINK na formulár, z ktorého bola potvrdzovaná akcia vyvolaná – ak nebude uvedené inak.

- **Message** (TEXT) Popis potvrdzovanej udalosti.
- **OK** (BUTTON) Potvrdenie akcie.
- **Cancel** (BUTTON) Odmietnutie akcie.

4.4.2 Registration of Authors

Po zvolení tejto linky na hlavnej stránke sa v pravej časti obrazovky sprístupní formulár Register.

RegisterAuthor Tento formulár slúži na zadanie registračných informácií o používateľovi, ktorý ich potvrdením vstupuje do role *Author*. Z tohto formulára je možné zadať údaje potrebné na vyplnenie atribútov v tabuľke tblPerson – *Login, Password, FirstName, MiddleName, Surname, Titles, Email, Organization.*

- **Login** (TEXT) Textové pole umožňujúce zaregistrovanie prihlasovacieho mena.
- **Password** (TEXT) Textové pole umožňujúce zadať heslo.
- **Retype Password** (TEXT) Textové pole na potvrdenie zadaného hesla.
- **First Name** (TEXT) Textové pole umožňujúce zadanie mena.
- **Middle Name** (TEXT) Textové pole umožňujúce vstup mena.
- **Surname** (TEXT) Textové pole umožňujúce zadanie priezviska.
- **Titles** (TEXT) Textové pole umožňujúce zadať tituly používateľa.
- **Email** (TEXT) Textové pole umožňujúce vstup emailovej adresy.
- **Organization** (TEXT) Textové pole umožňujúce evidovať príslušnosť k nejakej organizácii.
- **Notify** (CHECKBOX) Ide o prepínač, ktorým používateľ požiada od systému správu v podobe mailu v prípade prijatia článku.
- **OK** (BUTTON-LINK: *Login*) Tlačítko na zapísanie zmien do systému.

4.4.3 Registration of Participant

Po zvolení tejto linky na hlavnej stránke sa v pravej časti obrazovky sprístupní formulár Register, ktorý je rozšírením formuláru pre autorou o položky, ktoré sa týkajú platieb za konferenciu, apod.

RegisterParticipant Tento formulár slúži na zadanie registračných informácií o používateľovi, ktorý ich potvrdením vstupuje do role *Participant*. Z tohto formulára je možné zadať najdôležitejšie údaje potrebné na vyplnenie atribútov z tabuľky tblPerson – *Login, Password, FirstName, MiddleName, Surname, Titles, Email, Organization* – ako aj údaje, ktoré sa týkajú len účastníka (tabuľka tblParticipant) – *Address*, *Nationality*, *Fee*, *FeePaid*, *Phone* a *RegState*.

Login (TEXT) Textové pole umožňujúce zaregistrovanie prihlasovacieho mena.

Password (TEXT) Textové pole umožňujúce zadať heslo.

Retype Password (TEXT) Textové pole na potvrdenie zadaného hesla.

First Name (TEXT) Textové pole umožňujúce zadanie mena.

Middle Name (TEXT) Textové pole umožňujúce vstup mena.

Surname (TEXT) Textové pole umožňujúce zadanie priezviska.

Titles (TEXT) Textové pole umožňujúce zadať tituly používateľa.

Email (TEXT) Textové pole umožňujúce vstup emailovej adresy.

Organization (TEXT) Textové pole umožňujúce evidovať príslušnosť k nejakej organizácii.

Notify (CHECKBOX) Ide o prepínač, ktorým používateľ požiada od systému správu v podobe mailu, ktorá potvrdzuje zaplatenie poplatku za konferenciu.

Address (TEXTAREA) Adresa účastníka.

Nationality (TEXT) Národnosť účastníka.

Fee (SELECT) Výber ceny z ponúkaných možností (pre rôzne organizácie môžu byť rôzne cenové ponuky).

Phone (TEXT) Telefónne číslo na účastníka.

State (SELECT) Výber štátu účastníka.

OK (BUTTON-LINK: *Login*) Tlačítko na zapísanie zmien do systému.

4.4.4 My Registration

Po zvolení tejto linky sa v pravej časti obrazovky sprístupní formulár *MyRegistration*. Tento obsahuje všetky potrebné údaje z tblParticipant a ak je používateľ účastníkom, tak potom aj údaje z tabuľky tblParticipant.

MyRegistration Tento formulár slúži na zobrazenie a zmenu registračných informácií o používateľovi (role *Author*, *Reviewer*). Jeho obsah je závislý od uvedených rolí, ktoré používateľ v systéme má. Z tohto formulára je možné meniť najdôležitejšie atribúty z tabuľky tblPerson – *Login, Password, FirstName, MiddleName, Surname, Titles, Email, Organization* (a pre účastníka aj atribúty z tabuľky tblParticipant). Ak sa niektoré pole zobrazí len pre niektoré role, sú tieto uvedené za popisom v zátvorkách.

Login (TEXT) Textové pole umožňujúce zmenu prihlasovacieho mena.

Password (TEXT) Textové pole umožňujúce zadať nové heslo.

Retype Password (TEXT) Textové pole na potvrdenie zadaného hesla.

First Name (TEXT) Textové pole umožňujúce zmenu (opravu) mena.

Middle Name (TEXT) Textové pole umožňujúce zmenu (opravu) mena.

Surname (TEXT) Textové pole umožňujúce zmenu (opravu) priezviska.

Titles (TEXT) Textové pole umožňujúce zmeniť tituly používateľa.

Email (TEXT) Textové pole umožňujúce zmenu (opravu alebo aktualizáciu) emailovej adresy.

Organization (TEXT) Textové pole umožňujúce evidovať a meniť príslušnosť k nejakej organizácii.

Notify (CHECKBOX) Ide o prepínač, ktorým používateľ v roli *Author* požaduje, aby mu systém zaslal správu v podobe mailu v prípade prijatia článku. Pri roli *Participant* sa notify týka zaslania potvrdenia o zaplatení – tj. pozvánka na konferenciu. (*Author*, *Participant*)

All Topics (LIST) Zoznam tém v systéme, z ktorých si *Reviewer* môže vybrať ním preferované. (*Reviewer*)

Selected Topics (LIST) Zoznam tém, ktorých týkajúce sa články Reviewer uprednostňuje. Medzi týmito dvomi zoznamami je možné prehadzovať položky pomocou dvoch tlačítiek. (*Reviewer*)

Address (TEXTAREA) Adresa účastníka. (*Participant*)

Nationality (TEXT) Národnosť účastníka. (*Participant*)

Fee (SELECT) Výber ceny z ponúkaných možností (pre rôzne organizácie môžu byť rôzne cenové ponuky). (*Participant*)

Fee Paid (výpis dátumu) Informácia o dátume zaplatenia poplatku – nie je možné editovať. (*Participant*)

Phone (TEXT) Telefónne číslo na účastníka. (*Participant*)

State (SELECT) Výber štátu účastníka. (*Participant*)

Change (BUTTON-LINK: *MyRegistration*) Tlačítko na zapísanie zmien do systému.

4.4.5 My Papers

Po zvolení linky My Papers sa v pravej časti obrazovky sprístupní formulár MyPapers.

MyPapers Tento formulár umožňuje používateľovi v roli *Author* prezerať zoznam článkov, ktoré zaslal do konferencie, ich stav (prijatý, neprijatý) a jednotlivé pripomienky, ktoré k článkom majú oponenti. Taktiež obsahuje odkaz na formulár pre pridanie članku do konferencie.

Papers (TABLE) Tabuľka obsahuje najdôležitejšie údaje o článku a priebehu jeho akceptácie na konferencii.

Name (ATTRIBUTE-LINK: *EditPaper*) Názov članku.

State (ATTRIBUTE) Informácia o stave, ktorá môže zahŕňať aj počet pridelených oponentov.

Add (BUTTON-LINK: *EditPaper*) Tlačítko na pridanie ďalšieho článku do konferencie.

EditPaper Tento formulár umožňuje používateľovi v roli *Author* pridávať nové články do konferencie, alebo editovať údaje v už zaslaných článkoch. Zadávané údaje zodpovedajú atribútom tabuľky tblPerson. Obsahuje tiež komponent na priradenie tém, ktoré súvisia s článkom.

- **Name** (TEXT) Textové pole umožňujúce zadať názov článku, ktorým sa článok bude identifikovať pre používateľov konferencie.
- **Authors** (TEXT) Textové pole umožňujúce zadať zoznam autorov článku.
- **Body** (TEXT) Textové pole umožňujúce zadať názov súboru, ktorý obsahuje samotný článok a bude prenesený na stranu servera.
- **Abstract** (TEXTAREA) Viacriadkové textové pole umožňujúce autorovi priamo zadať abstrakt článku.
- **KeyWords** (TEXT) Textové pole umožňujúce zadať kľúčové slová, ktorých sa článok dotýka.
- **All Topics** (LIST) Zoznam tém v systéme, z ktorých si *Author* môže vybrať tie, ktoré sa dotýkajú ním zasielaného článku.
- **Selected Topics** (LIST) Zoznam tém, ktoré *Author* vyberie ako súvisiace s článkom.
- **Comments** (odstavec textu pre každý komentár oponenta) Tu sa nachádza zoznam pripomienok od oponentov.
- **OK** (BUTTON-LINK: *MyPapers*) Tlačítko na potvrdenie zmenených údajov a ich zápis do systému.

4.4.6 My Reviews

Po zvolení linky *My Reviews* sa do pravej časti obrazovky zobrazí formulár *MyReviews*.

MyReviews (FORM) V tomto formulári je centrálnym prvkom tabuľka článkov pridelených oponentovi na posudzovanie. Je možné dať si zobraziť len články, pre ktoré autor ešte posudok nezaslal, články, ktoré už posúdil alebo všetky články. Rovnako tak tu bude výrazne vypísaná informácia o termíne skončenia procesu posudzovania.

- **To Review** (CHECKBOX) Budú zobrazené príspevky, ktoré má oponent ešte posúdiť. Na začiatku bude tento zapnutý.
- **Reviewed** (CHECKBOX) Budú zobrazené príspevky, ktoré oponent už posudzoval. Na začiatku bude tento zapnutý.
- **Papers** (TABLE) Tabuľka bude obsahovať v riadkoch záznamy o jednotlivých príspevkoch priradených oponentovi na posudzovanie:
  - **Name** (ATTRIBUTE) Meno príspevku.
  - **Review Date** (ATTRIBUTE) Ak ide o článok, ktorý už oponent posúdil, zobrazí sa dátum posudku.
  - **Recommandation** Hodnota položky Odporúčanie na prijatie.

ReviewPaper (FORM) Základom tohto formulára sú položky hodotenia príspevku podľa kritérií zaradené v tabuľke. Na úvod formulára sú vypísané vybrané informácie o príspevku.

Ranking (TABLE) Tabuľka obsahuje hodnotiace atribúty postupne pre všetky kritériá.

Criteria (ATTRIBUTE) Meno kritéria.

Value (ATTRIBUTE-SELECT) Tu oponent vyberie hodnotu pre dané kritérium.

Comment (TEXTAREA) Pole pre komentár k tomuto hodnoteniu.

Description (ATTRIBUTE) Bližší opis kritéria.

Recommendation (SELECT) Z týchto možností oponent vyberie svoje celkové odporúčanie ohľadom prijatia či neprijatia tohto príspevku na konferenciu.

Comment to author (TEXT) do tohto textového poľa oponent napíše svoje odporúčania či bližšie poznámky pre autora posudzovaného príspevku.

Comment to program comitee (TEXT) toto textové pole slúži na komentáre oponenta pre členov programovej komisie týkajúce sa posudzovaného článku.

Ok (BUTTON) Potvrdenie zmien a uloženie formulára do databázy.

4.4.7 Administration

Po zvolení linky *Administration* sa celé ľavé menu zmení a objavia sa v ňom nasledovné položky:

Users (LINK: *AdminUsers*) Administrácia používateľov.

Review Assignment (LINK: *AdminReviewAssignment*) Priraďovanie oponentov k článkom.

Custom Pages (LINK: *AdminCustomPages*) Administrácia custom-stránok.

Con4U Setup (LINK: *AdminProperties*) Nastavovanie globálnych vlastností systému.

Ranking Criterias (LINK: *AdminRankingCriteria*) Administrácia hodnotiacich kritérií.

Topics (LINK: *AdminTopics*) Zadávanie tém konferencie.

AdminUsers (FORM) Tento formulár slúži na administráciu používateľov. Je možné vybrať si role, ktorých používatelia sa zobrazia. Ďalej je možné priamo z tohto formulára mazať viacerých používateľov. Pre tento účel bola použitá komponenta TABLE s checkboxom pre každú položku, pričom po stlačení tlačítka *Delete* sa označení používatelia odstránia zo systému (vyžaduje sa potvrdenie operácie).

Roles (CHECKBOX skupina) – skupina štyroch checkboxov pre roli *Administrator*, *Author*, *Reviewer* a *Participant*. V tabuľke sa zobrazujú len tí používatelia, ktorý patria aspoň do jednej zo zaškrtnutých rolí.

Users (TABLE) Tabuľka bude obsahovať vybrané najdôležitejšie atribúty z tabuľky tblPerson:

Surname (ATTRIBUTE)
FirstName (ATTRIBUTE)
MiddleName (ATTRIBUTE)
Titles (ATTRIBUTE)
Organization (ATTRIBUTE)

Email (ATTRIBUTE)

Role (ATTRIBUTE) Obsahuje vypísané role, ktorými používateľ disponuje.

Select (ATTRIBUTE-CHECKBOX)

Login (ATTRIBUTE-LINK: *AdminUserEdit*)

Add (BUTTON-LINK: *AdminUserEdit*) Tlačítko pre pridanie nového používateľa.

Delete (BUTTON-LINK: *Confirmation*) Vymazanie vybraných používateľov pomocou komponenty – na túto operáciu je požadované potvrdenie, ktoré nasleduje na samostatnom formulári. Po potvrdení sa obnoví formulár *AdminUsers*.

AdminUserEdit (FORM) Formulár AdminUserEdit slúži na editáciu už existujúceho záznamu, prípadne naplnenie vytvoreného záznamu (pridanie používateľa). Bude obsahovať jednotlivé položky záznamu z tabuľky tblParticipant:

Login (TEXT)

Password (TEXT)

First Name (TEXT)

Middle Name (TEXT)

Surname (TEXT)

Titles (TEXT)

Email (TEXT)

Organization (TEXT)

Role (CHECKBOX) Štvorica checkboxov pre každú rolu: *Administrator*, *Participant*, *Reviewer*, *Author*.

Registration Date (TEXT) Dátum registrácie.

Add/Edit (BUTTON-LINK: *AdminUsers*) Potvrdenie záznamu a znovu načítanie formuláru *AdminUsers*. Popis bude buď *Add* alebo *Edit*, čo závisí na akcii, ktorú formulár momentálne realizuje.

Cancel (BUTTON-LINK: *AdminUsers*) Znovu načítanie formuláru *AdminUsers* bez uloženia záznamu.

AdminReviewAssigmment (FORM) Toto je úvodný formulár pre priraďovanie oponentov na jednotlivé príspevky. Obsahuje prehľad príspevkov a počty oponentov pri každom príspevku. Na tomto formulári neprebieha samotné priraďovanie – na to slúžia nasledovné dva formuláre (*AdminAtricleReviews* a *AdminReviewArticles*).

Papers (TABLE) Tabuľka všetkých článkov evidovaných systémom bude obsahovať tieto atribúty:

Article name (ATTRIBUTE-LINK: *AdminReviewArticle*) Názov článku. Po kliknutí na článok sa dostaneme na formulár, kde priraďujeme oponentov k tomuto (jedinému) príspevku.

Date of receiving (ATTRIBUTE) Dátum prijatia článku do systému.

Number assigment rewievers (ATTRIBUTE) Počet priradených oponentov.

Select *(*ATTRIBUTE-CHECKBOX)

Delete (BUTTON-LINK: *Confirm*) Vymaže označené články.

AdminArticleReviews (FORM) Formulár slúžiaci na priradenie oponentov jednému príspevku.

Reviews (TABLE) Tabuľka bude obsahovať zoznam všetkých oponentov. Tabuľka bude zotriedená podľa zhody preferencií oponenta s témami príspevku.

Name (ATTRIBUTE-LINK: *AdminReviewArticles*) Meno oponenta.

Numbers of articles to be done (ATTRIBUTE) Počet posudkov, ktoré má daný oponent priradené.

Selected (ATTRIBUTE) Obsahuje informáciu, či oponent už oponuje tento článok. Na začiatku práce s formulárom sa stav zhoduje s atribútom *Select*. Táto informácia je vhodná najmä po použití automatického návrhu (*Suggest Selection*), aby mal administrátor kontrolu aj nad pôvodným stavom priradenia.

Select (ATTRIBUTE-CHECKBOX) Na začiatku sú označení oponenti, ktorí oponujú daný príspevok. Tento stav je možné zmeniť (ako ručne, tak aj tlačítkom *Suggest Selection*).

Suggest Selection (BUTTON) Navhrhne priradenie oponentov pre príspevok tak, aby ich počet sedel s nakonfigurovaným počtom oponentov pre článok. Pri návrhu sa postupuje podľa poradia v zozname, ale prihliada sa na množstvo príspevkov, ktoré už oponent má pridelené.

Submit (BUTTON-LINK: *Confirmation*) Potvrdenie zmien priradení.

Reset (BUTTON) Obnovenie pôvodného stavu formulára.

AdminReviewArticles (FORM) Úlohov tohto formulára je uľahčiť priraďovanie jednotlivých príspevkov oponentom.

Articles (TABLE)

Name (ATTRIBUTE-LINK: *AdminArticleReviews*) Názov príspevku.

Autor (ATTRIBUTE) Meno autora.

Numbers of rewievers (ATTRIBUTE) Počet oponentov priradených k tomuto príspevku.

Selected (ATTRIBUTE) Obsahuje informáciu, či je článok oponovaný týmto oponentom. Na začiatku práce s formulárom sa stav zhoduje s atribútom *Select*. Táto informácia je vhodná najmä po použití automatického návrhu (*Suggest Selection*), aby mal administrátor kontrolu aj nad pôvodným stavom priradenia.

Select (ATTRIBUTE-CHECKBOX) Na začiatku sú označené tie príspevky, ktoré daný oponent oponuje. Tento stav je možné zmeniť (ako ručne, tak aj tlačítkom *Suggest Selection*).

Suggest Selection (BUTTON) Navhrhne priradenie oponentov pre príspevok tak, aby ich počet sedel s nakonfigurovaným počtom oponentov pre článok. Pri návrhu sa postupuje podľa poradia v zozname, ale prihliada sa na množstvo príspevkov, ktoré už oponent má pridelené.

Submit (BUTTON-LINK: *Confirmation*) Potvrdenie zmien priradení.

Reset (BUTTON) Obnovenie pôvodného stavu formulára.

AdminCustomPages (FORM) V tomto formulári je možné pridávať a odoberať statické stránky do/zo systému. Ďalej sa tu určuje viditeľnosť týchto stránok pre jednotlivé roly.

Pages (TABLE) Tabuľka bude obsahovať atribúty:

Name (ATTRIBUTE-LINK: *AdminCustomPageEdit*)
MimeType (ATTRIBUTE)
Protection (ATTRIBUTE)
Roles (ATTRIBUTE) Textový výpis rolí, ktoré môžu pristúpiť na túto stránku.
Select (ATTRIBUTE-CHECKBOX)

Add (BUTTON-LINK: *AdminCustomPageEdit*) Pridanie ďalšej statickej stárnky.

Delete (BUTTON-LINK: *Confirmation*) Zmazanie vybraných stránok.

AdminCustomPageEdit (FORM)

Name (TEXT) Názov stránky (text linky, titul).

MimeType (COMBOBOX) – Mime typu objektu.

Body (BUTTON-LINK: upload stránky)

Role Author (ATTRIBUTE-CHECKBOX) Určuje, či stránka je prístupná autorom (prihláseným do systému).

Role Participant (ATTRIBUTE-CHECKBOX)

Role Reviewer (ATTRIBUTE-CHECKBOX)

Submit (BUTTON-LINK: *AdminCustomPages*) Uloženie zmien do databázy.

Cancel (BUTTON-LINK: *AdminCustomPages*) Znovu načitanie formuláru *AdminCustomPages* bez uloženia zmien.

AdminProperties (FORM) Tento formulár je určený na nastavenie všeobecných parametrov konferencie. Na formulári bude umiestnená tabuľka, kde prvý stĺpec bude označovať názov parametra a druhý stĺpec jeho hodnotu.

Parameters (TABLE) Tabuľka bude obsahovať atribúty z tabuľky tblParameters*:*

ParameterName (ATTRIBUTE) - Statický text.
ParameterValue (ATTRIBUTE-TEXT) - Editovateľný text.

Jednotlivé záznamy tabuľky nebude možné mazať a ani pridávať. Pojde o tabuľku kde bude možné editovať iba stĺpec s atribútom *parameterValue* (ATTRIBUTE). Jednotlivé parametre sú do systému pevne zavedené.

Ako príklad záznamov pre túto tabuľku sú:

DeadLineForAutors – dátum posledného možného poslania článku do konferencie.

DeadLineForRewievers – dátum posledného možného poslania posudku na článok do konferencie.

NumberRewieversForPaper – počet oponentov na článok.

MaximumPapersForRewiever – maximálny počet článkov na oponenta.

Submit (BUTTON-LINK:*Confirmation* (FORM)) Potvrdenie všetkých zmien hodnôt atribútov parameterValue v tabuľke *Parameters*.

Reset (BUTTON) Znovunačítanie formuláru.

AdminRankingCriteria (FORM) Tento formulár je určený pre manažovanie jednotlivých hodnotiacich kritérií, ich bližší popis a definíciu.

AllCriterias (TABLE) tabuľka obsahujúca všetky mená hodnotiacich kritérií.

Name (ATTRIBUT-LINK: *AdminRankingEdit*) Meno hodnotiaceho kritéria.

Description (ATTRIBUT) Bližší popis hodnotiaceho kritéria.

Select (ATTRIBUT-CHECKBOX)

Add (BUTTON-LINK: *AdminRankingEdit*)

Delete (BUTTON-LINK) Vymazanie vybraných hodnotiacich kritérií.

AdminRankingEdit (FORM)

Name (TEXT) Meno kritéria.

Description (TEXTAREA) Text, ktorý bližšie popisuje kritérium.

All Values (LIST) Zoznam všetkých hodnôt všetkých kritérií.

Add Value (BUTTON-LINK) Tlačidlo na pridanie hodnoty do množiny hodnôt tohto kritéria (*Criteria Values*).

Remove Value (BUTTON-LINK) Odstránenie hodnoty z množiny pre editované kritérium.

Criteria Values (LIST) Vybrané hodnoty pre editované kritérium.

New Value (TEXT) Ak chceme pridať novú hodnotu pre kritérium, sem napíšeme jej označenie.

Add New Value (BUTTON-LINK) Pridá novú hodnotu do *Criteria Values*.

Ok (BUTTON-LINK: *AdminRankingCriteria*) Potrvdenie zmien vo formulári.

Cancel (BUTTON-LINK:*AdminRankingCriteria*)

AdminTopics (FORM) Administrátor v tomto formulári bude môcť definovať jednotlivé témy konferencie, tj. pridávať, mazať, meniť názov a popis témy.

Topics (TABLE) Bude obsahovať atribúty z tabuľky tblTopics:

Name (ATTRIBUTE-LINK: *AdminTopicEdit*) Meno témy.

Description (ATTRIBUTE) Bližšie špecifikuje zvolenú tému.

Select *(*ATTRIBUTE-CHECKBOX)

Add (BUTTON-LINK: *AdminTopicEdit*) Pridanie novej témy.

Delete (BUTTON-LINK: *Confirmation*) Zmaže vybrané záznamy.

AdminTopicEdit (FORM) Formulár slúži na editáciu, už existujúceho záznamu, prípadne naplnenie novo vytvoreného záznamu. Bude obsahovat jednotlivé položky záznamu z tabuľky tblTopic:

Name (TEXT) Meno témy.

Description (TEXTAREA) Popis témy.

Submit (BUTTON-LINK) Potvrdenie záznamu a načítanie *AdminTopics*.

Cancel (BUTTON-LINK: *AdminTopics*) Načítanie *AdminTopics* bez uplatnenia zmien.

4.5 Návrh dátového modelu – verzia 2

Počas fázy návrhu a prototypovania sme postupne upravili, doplnili a zjemnili dátový model. Hrubý návrh dátového modelu uvedený v kapitole 4.1 je tu doplnený o atribúty, upravený podľa reakcií zákazníka na prototyp a rozvinutý do fyzického modelu dát.

Na rozdiel od modelu uvedeného v hrubom návrhu sme prešli k anglickým názvom entít a atribútov. Je to tak preto, lebo sme dátový model priamo transformovali do tabuliek databázového systému v implementácii a chceli sme zachovať korešpondujúce názvy tabuliek s názvami entít v modeli.

4.5.1 Logický model údajov

V tejto časti je uvedený logický model dát, ktorý vychádza z modelu uvedeného v kapitole 3.3, ale je pozmenený a doplnený, ako bolo uvedené v úvode tejto podkapitoly. Nasleduje popis entít, niektorých atribútov a väzieb medi entitami s dôrazom na zmeny oproti pôvodnej verzii modelu. Samotný model je znázornený na obrázku 10.

Person Entita označujúca používateľa systému, ktorý je v systéme evidovaný a jednoznačne identifikovateľný. Zahŕňa účastníkov konferencie (pozri aj entitu *Participant*), autorov príspevkov, členov programovej komisie a administrátorov systému.

Login, Password Atribúty slúžiace pre zabezpečenie prístupu do systému.

First name, Middle name, Surname, Titles Meno a akademické tituly osoby.

Email, Organization Elektronická adresa osoby a jej pôsobisko.

IsAuthor, IsReviewer, IsAdmin Položky určujúce roly osoby v systéme. O jednotlivých rolách v systéme hovorí kapitola 3.4. Jedna osoba môže patriť do viacerých rolí a o priradení osôb do rolí oponenta a administrátora rozhoduje administrátor systému. Položka *IsAuthor* sa nastaví automaticky, ak osoba podá do systému príspevok do konferencie.

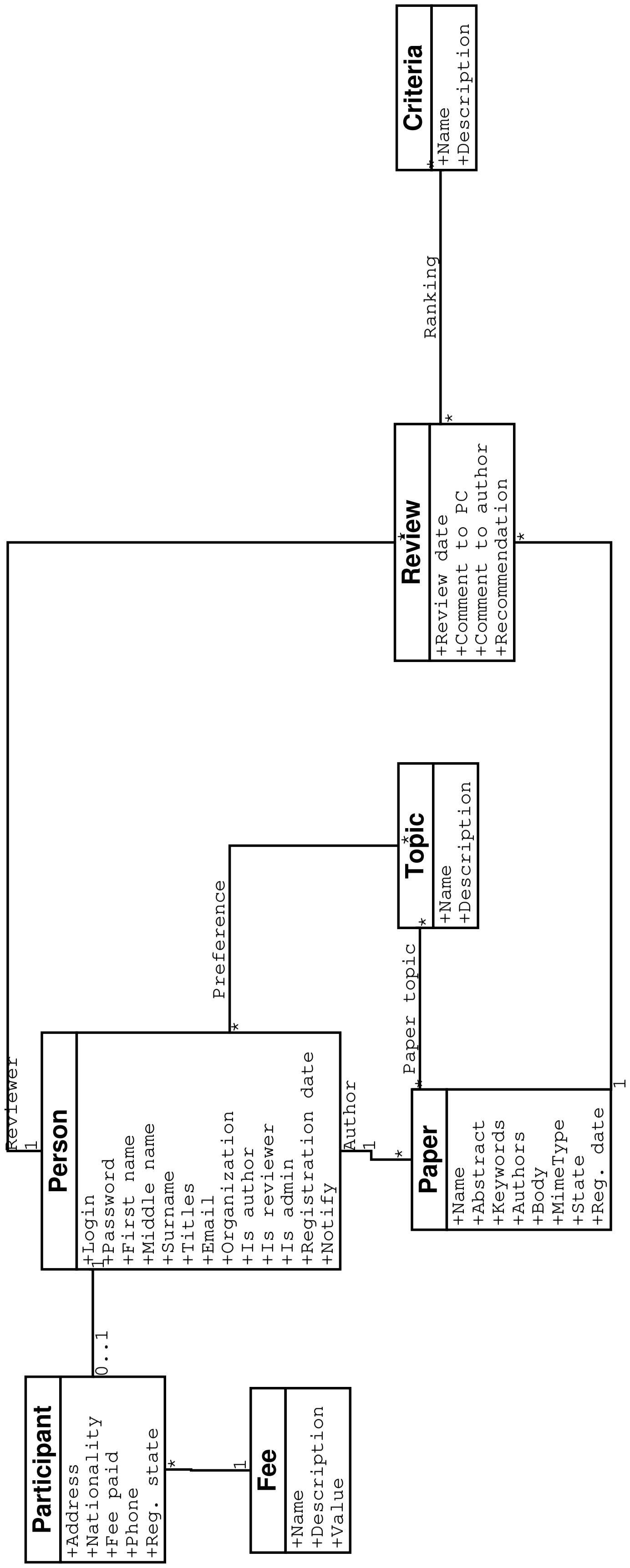
RegDate Dátum registrácie osoby do systému.

Notify Určuje, či si používateľ želá byť notifikovaný mejlom o prípadných zmenách, o prijatí či neprijatí článku a podobne.

Participant Entita zahrňujúca údaje o účastíkovi konferencie. Inštancia tejto entity sa vytvorí pre inštanciu entity *Person*, keď sa daná osoba zaregistruje ako účastník konferencie.

Address, Nationality, Phone Ďalšie doplňujúce atribúty pre účastníka konferencie pridané k tým, ktoré sa o ňom evidujú ako o osobe v inštancii entity *Person*.

FeePaid Dátum zaplatenia registračného poplatku. Ak poplatok ešte nebol uhradený, bude tu uložená nejaká definovaná hodnota (napríklad nula).



Obrázok 10: Logický model údajov

RegState Stav registrácie. Registrácia účastníka sa môže nachádzať v rôznych stavoch, ktoré závisia napríklad od zaplatenia poplatku, predbežnej registrácie a podobne.

Fee Entita poplatok za účasť na konferencii. Je možných viacero rôznych typov účastníckych poplatkov, s rôznou cenou, napríklad v závislosti od členstva účastníka v nejakej organizácii podporujúcej konferenciu, od toho, či sa jedná o študenta alebo napríklad telesne postihnutého.

Name, Description Krátky názov poplatku a jeho bližší textový opis. Tu budú uvedené podmienky, za ktorých je možné použiť tento typ poplatku.

Value Peňažná hodnota poplatku v mene definovanej globálne pre celý systém.

Paper Entita príspevok do konferencie (článok). Toto je jedna z fundamentálnych entít v celom systéme Con4U, pretože k príspevku patrí jeho autor, oponenti, posudky.

Name, Authors Názov článku a jeho autori. Je potrebné rozlíšiť atribút *Authors*, ktorý je textovou položkou, do ktorej sú zapísaní všetci autori článku tak, ako sú uvedení na titulnej strane článku od atribútu *Author*, ktorý je väzbou na inštanciu entity *Person* a označuje autora, ktorý pridal tento článok do systému.

Abstract, Keywords Textové položky, ktoré obsahujú abstrakt a kľúčové slová príspevku. Znovu je potrebné rozlíšiť kľúčové slová (*Keywords*) od tém (*Topic*). Množina tém je globálna pre všetky príspevky a autor príspevku pre každý príspevok vyberie z tohto zoznamu tie, ktorých sa tento príspevok týka. Kľúčové slová sú voľným textom, ktoré zadáva autor a niekedy sa tiež uvádzajú v samotnom príspevku.

Body Samotné telo článku vo forme súboru.

MimeType Typ (formát) tela článku – súboru.

State Stav príspevku môže nadobudnúť jednu z hodnôt – prijatý, zamietnutý, ešte nerozhodnuté atď.

RegDate Dátum zaslania príspevku do systému.

Topic Téma konferencie. Tieto témy sa používajú pri bližšom určení článku (každému článku autor priradí, ktorých tém sa článok týka) a slúžia aj na vyjadrenie preferencií oponentov (každý oponent si vyberie, ktoré témy preferuje a podľa toho mu budú prideľované príspevky na oponovanie).

Name, Description Názov a bližší popis témy.

Review Entita posudok na príspevok. Táto entita je vo viacerých vzťahoch s niekoľkými inými entitami a preto je vhodné bližšie vysvetliť niektoré vzťahy aj na tomto mieste. Jeden oponent je autorom niekoľkých posudkov (na rôzne príspevky). Na jeden príspevok píšu posudky niekoľkí oponenti. V každom posudku sa príspevok hodnotí podľa všetkých kritérií.

ReviewDate Dátum podania posudku do systému.

CommentToAuthor Poznámky, odporúčania, komentáre oponenta pre autora príspevku.

CommentToPC Poznámky, odporúčania, komentáre oponenta pre ostatných členov programovej komisie. Tieto môžu slúžiť ako podpora pri finálnom rozhodovaní o prijatí či neprijatí článku na konferenciu.

Recommendation Celkové odporúčanie oponenta ohľadom prijatia alebo neprijatia článku na konferenciu.

Criteria Entita kritérium hodnotenia príspevku. Príklady takýchto kritérií sú napr. originalita, vhodnosť, prínos a podobne. Príspevok sa posudkom hodnotí podľa všetkých týchto kritérií a pri hodnotení podľa každého kritéria sa príspevku priraďuje hodnota, určujúca mieru naplnenia daného kritéria, napríklad výborná originalita, menšia vhodnosť na konferenciu, veľmi veľký prínos...

Name, Description Krátke meno kritéria a jeho obsiahlejší opis, ktorý môže zahŕňať aj návod na spôsob hodnotenia podľa tohto kritéria.

Preference Väzba určujúca, že daná téma *(topic)* je preferovaná daným oponentom (*reviewer*). Každý oponent môže mať niekoľko preferencií.

PaperTopic Väzba určujúca, že príspevok (*paper*) sa týka danej témy (*topic*). Jeden príspevok sa môže týkať viacerých tém, jednej témy sa môže týkať viac príspevkov.

Ranking Väzba opisujúca hodotenie príspevku v posudku (*review*) podľa kritéria (*criteria*). Každý posudok hodnotí príspevok podľa viacerých (všetkých) kritérií a podľa jedného kritéria sa hodnotí vo viacerých (všetkých) posudkoch.

4.5.2 Fyzický model údajov

Fyzický model dát vznikol transformáciou z logického modelu dát (jeho normalizáciou) a pridaním niektorých ďalších entít, väzieb a atribútov, ktoré riešia fyzickú realizáciu dát potrebných na zabezpečenie niektorých funkcií systému.

Vo fyzickom modeli sú už bližšie špecifikované typy atribútov entít.

V slovnom opise uvádzame len tie atribúty a väzby, ktoré boli pridané oproti logickému modelu.

Person Entita.

Participant Entita.

Fee Entita.

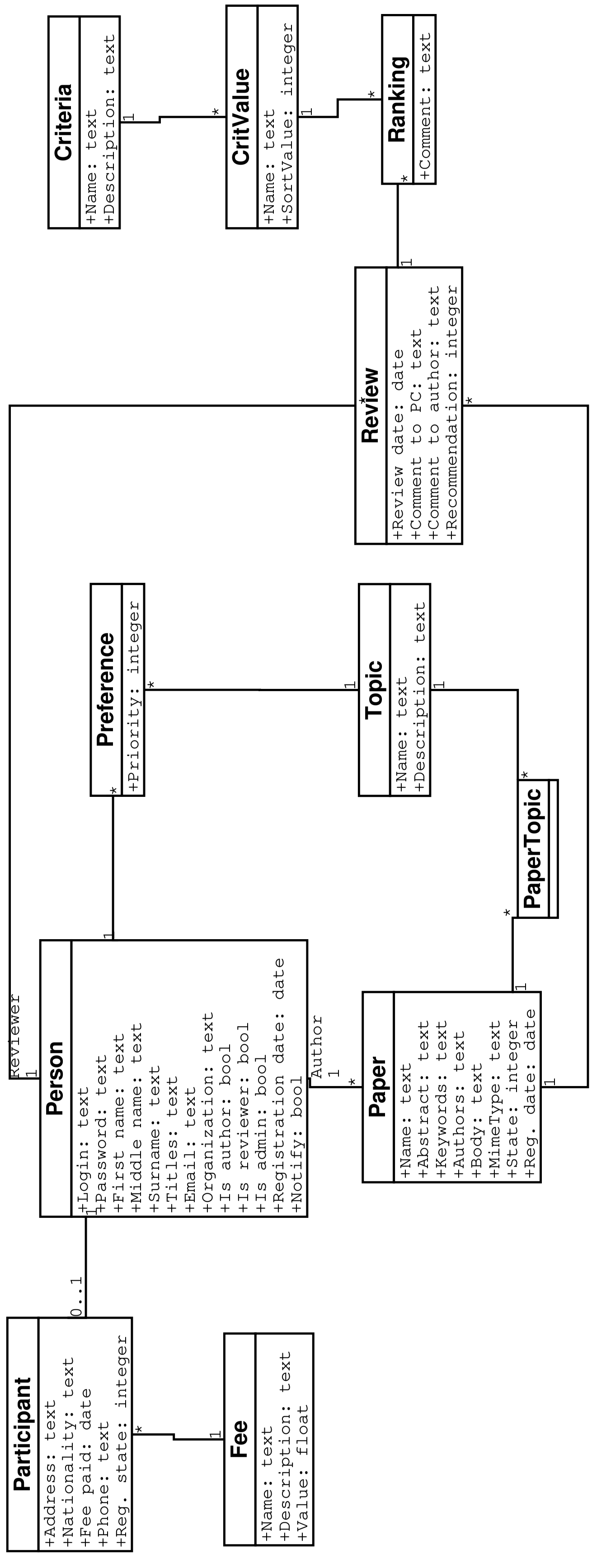
Paper Entita.

Topic Entita.

Review Entita.

PaperTopic Entita vzniknutá transformáciou väzby *PaperTopic* z logického modelu dát. Určuje, že príspevok sa týka témy.

Preference Entita vzniknutá transformáciou väzby *Preference* z logického modelu dát. Určuje, že oponent preferuje tému a mieru preferencie vyjadrenú prioritou.



Obrázok 11: Fyzický model údajov

Priority Priorita preferencie oponenta k téme.

Ranking Entita vzniknutá z logického modelu transformáciou väzby *Ranking*. Určuje hodnotenie posudzovaného príspevku podľa kritéria. Hodnotenie je vyjadrené priradením inštancie entity *CritValue* k inštancii entity *Ranking*.

Criteria Entita. Oproti dátovému modelu je entita *Criteria* s entitou *Ranking* zviazaná cez *CritValue*, čo je akýsi zoznam hodnôt pre jednotlivé kritéria.

CritValue Entita označujúca možnú hodnotu hodnotenia príspevku v posudku. Niekoľko položiek entity má vždy spoločný cudzí kľúč do tabuľky *Criterium*. Jedna inštancia entity teda určuje ako hodnotenie, tak aj kritérium, ktorého sa hodnotenie týka.

Name Textové vyjadrenie hodnotenia.

SortValue Aby bolo možné radiť textové označenia hodnotení, každé hodnotenie má priradenú túto pomocnú hodnotu.

StaticPageObject Entita pre stránky a iné objekty webovej prezentácie konferencie. Tieto objekty slúžia na dynamické, aktuálne informácie, pokyny, novinky a podobne, pridávané do systému dynamicky – za behu konferencie. Systém potom prezentuje linky na tieto objekty a na požiadanie tieto objekty prezentuje používateľom.

Name Názov objektu, ktorý bude použitý aj ako hypertextová linka na tento objekt.

MimeType Typ (formát) dokumentu – objektu.

Body Samotné telo objektu vo forme súboru.

Protection Atribút, ktorý určuje, či objekt bude prezentovateľný anonymným používateľom, autorom príspevkov, registrovaným účastníkom konferencie alebo oponentom. Možné sú aj kombinácie implementované bitovým súčtom hodnôt.

5 Záver

V druhej etape sme rozvinuli predbežný návrh, pričom nám veľmi pomohli priebežné prezentácie prototypu. Všetky navrhované súčasti systému chceme aj implementovať – sústredili sme sa na užšiu časť organizácie konferencie, podrobne sme ju preskúmali a sme pripravený ju celú realizovať.

Podstatným výstupom etapy 2 je aj používateľská príručka, ktorá tvorí logicky samostatný dokument.