MASARYKOVA UNIVERZITA
FAKULTA INFORMATIKY



Systém pro správu rezervací a zdrojů kolaborativních systémů

BAKALÁŘSKÁ PRÁCE

Markéta Trachtová

Brno, jaro 2011
login: xtracht
UČO: 255498

Prohlášení

Prohlašuji, že tato bakalářská práce je mým původním autorským dílem, které jsem vypracovala samostatně. Všechny zdroje, prameny a literaturu, které jsem při vypracování používala nebo z nich čerpala, v práci řádně cituji s uvedením úplného odkazu na příslušný zdroj.

Markéta Trachtová

Vedoucí práce: RNDr. Michal Procházka

Poděkování

Ráda bych poděkovala panu RNDr. Michalu Procházkovi za vedení bakalářské práce a jeho cenné rady poskytované v průběhu práce.

Shrnutí

Cílem práce je zajistit systém spravující rezervace služeb a zdrojů kolaborativních systémů. Musí umožňovat přístup již autentizovaných uživatelů, konkrétně přes autentizační službu České akademické federace identit eduID.cz, a být využitelný v systémech pracujících s paketovými zrcadly. Na základě požadavků a informací o současných systémech s podobným zaměřením jsem navrhla a implementovala rezervační webovou aplikaci. Předmětem rezervace jsou služby a zdroje náležící kolaborativnímu systému, dále je možné rezervovat externí zdroje, sdílet dokumenty a zvát účastníky.

Klíčová slova

Kolaborativní systémy, vzdálená spolupráce, webová aplikace, videokonference, eduID.cz, PHP, Model-View-Controler

Obsah

1 **Úvod** 2
2 **Existující kolaborativní systémy** 4
2.1 *EVO* 6
2.2 *FlashMeeting* 8
2.3 *Sumarizace* 9
3 **Systém pro správu rezervací a zdrojů kolaborativních systémů** 11
3.1 *Česká akademická federace identit eduID.cz* 11
3.1.1 O federaci 11
3.1.2 Paketové zrcadlo 12
3.2 *Návrh systému* 13
3.2.1 Struktura systému 13
3.2.2 Správa rezervačních záznamů 15
3.2.3 Správa systému 17
3.3 *Návrh implementace* 19
3.3.1 Návrhové vzory 20
3.3.1.1 Model-View-Controler 20
3.3.1.2 Singleton 21
3.3.2 Funkce modelu v rámci rezervačního systému 21
3.3.2.1 Datové úložiště 21
3.3.2.2 Manipulace s datovým úložištěm 23
3.3.3 Funkce pohledu v rámci rezervačního systému 24
3.3.4 Funkce řadiče v rámci rezervačního systému 25
4 **Implementace** 26
5 **Rozšíření do budoucna** 28
6 **Závěr** 29
Literatura 30

1 Úvod

Potřeba spolupráce mezi lidmi je stále aktuální. V dnešním světě propojeném Internetem a telefonní sítí není problém komunikovat na velké vzdálenosti. Kolaborace se tedy přenesla do virtuálního prostředí, aby poskytla podporu pro překonání geografických rozdílů. Zvláště výhodným se tento způsob spolupráce ukazuje v geograficky distribuovaných organizacích, kde umožňuje podstatně redukovat počet služebních cest a zefektivňuje interní komunikaci.[1]

Aplikace umožňující vzdálenou spolupráci uživatelů jsou označovány jako kolaborativní systémy. Na Internetu jich najdeme nepřeberné množství. Počítáme mezi ně sdílená úložiště dokumentů, email, wiki stránky[1], videokonference a řadu dalších. Převážně se jedná o aplikace s jedním primárně podporovaným způsobem kolaborace a případné služby sloužící k jeho podpoře. Pořádat on-line seminář, například o digitálních fotografiích, je možné pomocí programu Skype[2]. Sdílet ukázkové fotografie, o kterých se na semináři hovoří, umožní webová alba aplikace Picasa[3]. Dohodu na společném datu konání usnadní Google kalendář. Výčet komplementárních systémů by mohl pokračovat přes speciální wiki na sdílení informací o fotoaparátech až po systémy umožňující účastníkům zarezervovat multimediální místnosti pro realizaci semináře.

Programy se mohou funkčně doplňovat, ale jedná se o separátní entity, které nejsou schopny vzájemné komunikace. Způsobují tak nemalé problémy uživatelům, kteří musí chybějící spojení obstarat sami. Jedná se nejčastěji o koordinaci uživatelů před začátkem schůzky a spojení externích zdrojů s cílovým systémem. Výhody vzdálené spolupráce tímto způsobem zanikají, protože čas a námaha ušetřená díky vzdálené spolupráci se vrací v podobě komplikované organizace účastníků, zdrojů a služeb.

Náplní této práce je hledání a tvorba samostatné aplikace řešící plánování různých aspektů vzdálené spolupráce uživatelů a poskytující získané informace existujícímu kolaborativnímu systému. Výsledná webová aplikace by měla být zaměřena na plánování kolaborativních setkání a rezervaci externích zdrojů s ním spojených. Jejím účelem nebude realizovat vlastní schůzky, ale získat data od uživatele a distribuovat je dál. Příjemcem informací bude kolaborativní systém, který podle získaných parametrů uskuteční samotné setkání. Jelikož se očekává nasazení systému v rámci České akademické federace identit eduID.cz[4], která sdružuje akademické ústavy z území ČR a umožňuje jim vzájemné využití identit uživatelů při přístupu k síťovým službám, je potřeba, aby systém pra-

1. http://cs.wikipedia.org/wiki/Wiki/
2. http://www.skype.com/
3. http://picasa.google.com/
4. http://www.eduid.cz/wiki/eduid/index/

coval s již autentizovanými uživateli. Federace deleguje autentizaci uživatele na jeho domovskou organizaci, která poskytuje veškerá potřebná data k autorizaci uživatele. Aplikace bude potenciálně využitelná pro různé typy systémů podporujících vzdálenou spolupráci.

V první části práce se nachází souhrn problematiky současných kolaborativních systémů s konkrétními příklady aplikací. Je v ní zahrnuta první kapitola, čili úvod práce, usilující o vysvětlení jejího obsahu a kontextu ze kterého vychází.

Druhá kapitola poskytuje náhled na současné kolaborativní systémy nezbytný k určení nedostatků, o jejichž řešení práce usiluje. Obsahuje možnosti rozdělení aplikací, nejvýhodnějších kolaborativních technologií a představení nejčastějších problémů společných pro existující systémy vzdálené spolupráce. Její součástí je i podkapitola popisující dva existující systémy, EVO a Flashmeeting, implementující řeąšení jednoho z často se objevujících problémů při rezervaci kolaborativní služby.

Třetí kapitola umožňuje čtenáři bližší seznámení s Českou akademickou federací identit eduID.cz.

Druhá část práce je věnována tvorbě webové aplikace. Zahrnuje čtvrtou kapitolu soustředěnou na návrh systému, jeho popis a charakteristiku poskytovaných funkcí.

Pátá kapitola si klade za cíl seznámit čtenáře s návrhem implementace. Popisuje využité technologie návrhových vzorů a způsob jejich realizace ve vytvářeném systému. Zde se nachází popis struktury databáze, grafického zpracování a chování aplikace.

Šestá kapitola je věnována čistě jen implementačním detailům systému. Popisuje použité technologie a důvody pro jejich využití.

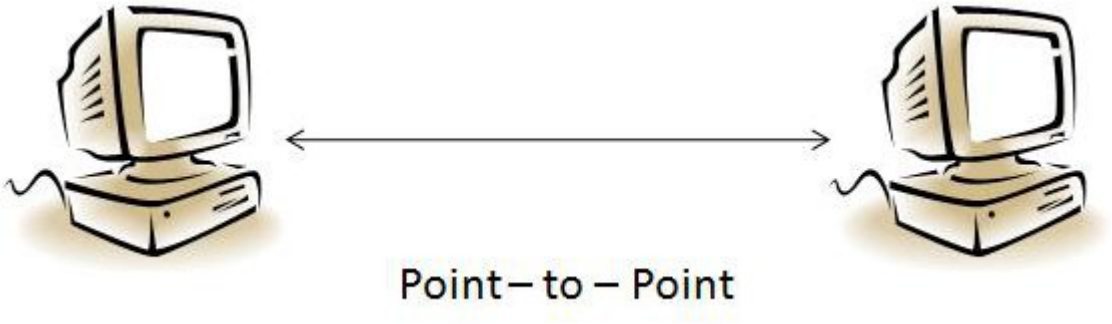
Sedmá kapitola poskytuje náhled do možností budoucího využití systému, možnosti jeho rozšíření.

Poslední osmá kapitola zahrnuje celkové zhodnocení práce. Rekapituluje problémy, které si kladl za cíl vyřeąit, které si systém kladl za cíl vyřešit, způsob řešení a přínosy vzniklé vytvořeným systémem.

2 Existující kolaborativní systémy

Systémů podporujících vzdálenou spolupráci uživatelů je velké množství. Lze je dělit například na základě poskytovaných služeb. Existují systémy jako GoogleDocs[1] nebo Zoho[2] zaměřené na sdílení dat různých formátů přes datové úložiště umístěné na Internetu. Jiné často využívané aplikace sloužící ke kolaboraci jsou textově orientované. Do této kategorie patří i populární Internetová služba instant messaging (IM)[3] umožňující komunikaci pomocí textových zpráv předávaných v reálném čase. IM je veřejnosti dostupný pomocí klientů jako ICQ, Pidgin nebo QIP[4]. Ke kolaborativním systémům se řadí i sociální sítě typu Facebook a Twitter, které se v poslední době těší velké oblibě.

Jiné rozdělení kolaborativních systémů je možné dle počtu koncových zařízení. Nejjednodušší je varianta přímé spojení dvou uživatelů typu point-to-point. Využita jsou pouze dvě koncová zařízení, jedno pro každého uživatele. Typicky jsou koncovými zařízeními běžné počítače. Podle poskytované kolaborativní služby mohou, ale nemusí být speciálně vybaveny. Například pro chat dvou uživatelů postačí klasický počítač s přístupem k Internetu. Oproti tomu IP telefonie vyžaduje minimálně přítomnost mikrofonu a reproduktorů.

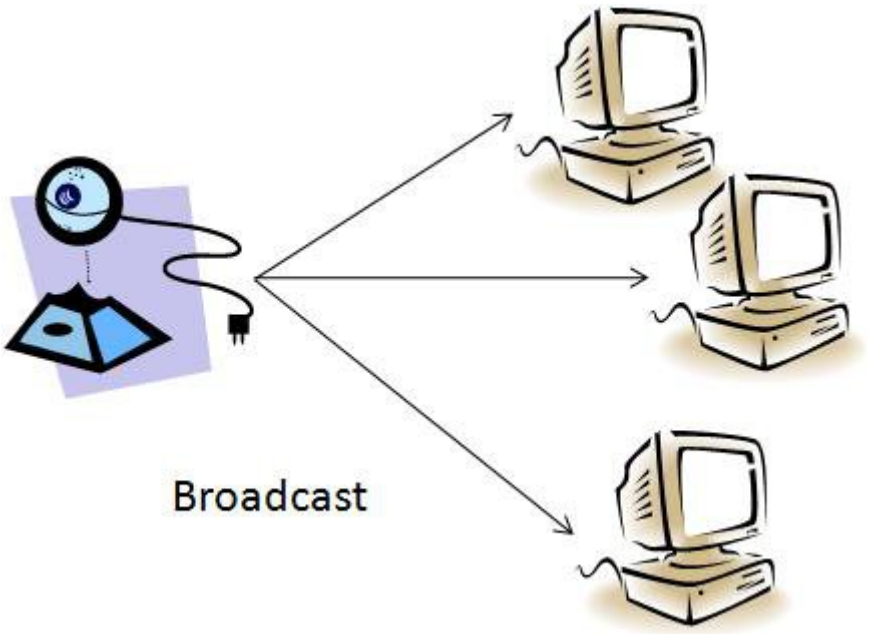


Obrázek 2.1: Spojení typu point to point

Dalším typem spojení je point-to-multipoint. V tomto případě existuje jedna centrální stanice, jejíž vysílání je určeno všem ostatním. Pokud chce vysílat některá z periferních stanic musí být komunikace delegována nejprve přes centrální stanici. Speciálním typem point-to-multipoint spojení je broadcast,

1. docs.google.com/
2. http://www.zoho.com/
3. http://cs.wikipedia.org/wiki/Instant_messaging/
4. http://www.icq.com/cz, http://www.pidgin.im/, http://cs.wikipedia.org/wiki/QIP/

čili skupinové vysílání. V tomto případě může vysílat pouze centrální stanice, ostatním je dovoleno jen naslouchat. Příkladem může být audiovizuální přenos lékařského zákroku různým pozorovatelům. Vysílat v takovémto případě může pouze jeden z účastníků, ostatní naslouchají. Koncová zařízení musí být pro tento druh komunikace většinou specializovaná.



Obrázek 2.2: Vysílání typu broadcast

Poslední možností je typ multipoint-to-multipoint, kdy je účastníků větší množství a všichni mohou vysílat i přijímat data. Koncová zařízení mohou být běžné počítače, speciální multimediální stanice, monitory a další. Příkladem jsou porady distribuovaných vývojářských týmů, pomocí videokonferenčních hovorů.

Právě videokonference jsou ukázkou toho nejlepšího, co je pro spolupráci uživatelů k dispozici. Díky nim je umožněn nejpřirozenější druh lidské komunikace, tedy rozhovor "tváří v tvář". Docíleno je toho souběžný přenosem obrazu i zvuku v reálném čase.

Realizace jednoduchého videokonferenčního přenosu mezi dvěma účastníky je navíc nenáročnou záležitostí. Potřeba je jen počítač s webovou kamerou, mikrofonem a připojením k Internetu. Systémy zprostředkovávající tento druh hovorů jsou volně dostupné. Mezi ty nejznámější patří Skype nebo CamFrog.[2]

Síla videokonferencí však tkví především v možnosti spolupráce velkého počtu uživatelů. Hromadná virtuální setkání již vyžadují jak softwarovou, tak

hardwarovou základnu. Přístroje umožňující videokonference, při nichž se připojuje více lidí ze stejného místa, jsou nákladné na pořízení i údržbu. Zvláště má-li být docíleno co nejlepší kvality přenášeného obrazu a zvuku. Většinou jsou organizacemi pořizovány do jedné nebo dvou místností, případně nárazově zapůjčovány specializovanými firmami. Systémy obsluhující takové konferenční místnosti a realizující videokonference bývají vytvářeny pro konkrétního zákazníka, dle jím požadovaných parametrů a možností, taktéž za patřičnou cenu. Nejsou tedy běžně využitelné.

S videokonferencemi jsou spojeny i všechny nejběžnější problémy kolaborativních systémů. K realizaci jediného videokonferenčního hovoru s více účastníky je potřeba podstoupit nepříjemné množství organizačních procedur. Protože neexistuje žádný centrální systém, který by umožňoval z jednoho místa koordinovat různé kolaborativní systémy, musí uživatel zajistit vše potřebné individuálně. Počínaje rozhodováním o způsobu a času kolaborativní schůzky, přes rezervaci vybavení a zdrojů potřebných pro realizaci setkání až po poskytnutí výsledků schůzky těm, co na ní nebyli přítomni. Všechno je zařizováno separátními aplikacemi.

Některé kolaborativní systémy úspěšně spojují alespoň část z výše zmíněných problémů. Příkladem jsou dvě volně dostupné aplikace, EVO a Flashmeeting, specializované na realizaci videokonferenčních hovorů s možností rezervací zdrojů systému.

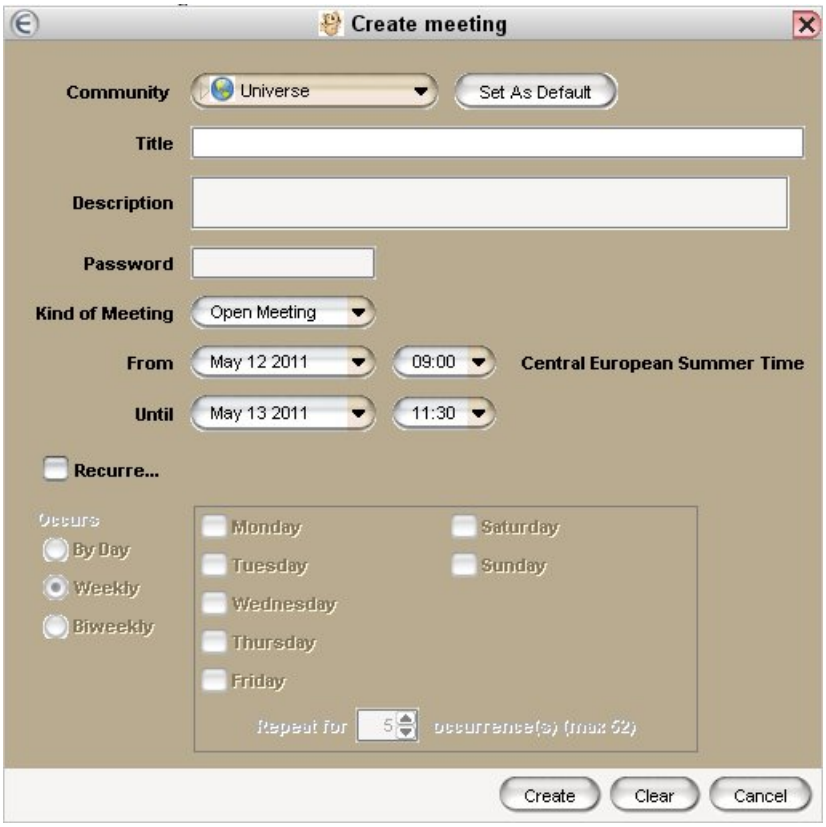
2.1 EVO

Enabling Virtual Organization System (EVO) vznikl pro podporu spolupráce lidí zapojených do programu fyziky vysokých energií, především pro LHC (urychlovač částic Large Hadron Collider). Jedná se o webovou aplikaci zaměřenou na podporu a realizaci videokonferenčních hovorů. Všechny uskutečněné videokonference mají k dispozici doplňkové funkce pro nahrávání celé schůzky, sdílení dokumentů, sdílenou tabuli, soukromý i veřejný instant messaging včetně informace o stavu (přítomen, zaneprázdněn a podobně) a další.[3]

V rámci systému lze pořádat takzvané "ad-hoc" schůzky nebo setkání plánovaná (rezervovaná) předem. Rezervovaná setkání poskytují svým účastníkům jistotu dostatečné výpočetní kapacity systému.

Rezervace může vytvářet kterýkoli přihlášený uživatel. V rámci rezervace jsou zadávány parametry budoucí konference. Prvním je výběr komunity, ke které bude konference v rámci systému patřit. Komunity sdružují uživatele do skupin dle zájmů. Základní komunita obsahující všechny uživatele je nazvaná Universe. Následujícími parametry jsou název konference, stručný popis charakterizující téma, o kterém se bude hovořit, a datum a čas konání. Název

konference a popis mají pouze informativní hodnotu pro uživatele. Čas začátku a konce konference je informativní údaj určující dobu, po kterou mají účastníci vyhrazenu videokonferenční kapacitu. Na schůzku se lze přihlásit ještě před počátečním časem a po uplynutí koncového času mohou účastníci pokračovat v konverzaci. Nejsou systémem odpojeni. Pro omezení přístupu ke konferenci slouží parametr heslo. Pokud je nějaké vloženo, mohou ke schůzce přistupovat jen vlastníci hesla. Dalším parametrem je určení typu konference. Ta může být plenární, tedy povinná pro všechny účastníky nebo s volným přístupem, tedy typ multipoint-to-multipoint nebo broadcast. Velice užitečný je poslední parametr umožňující nastavit periodické opakování schůzky a to v intervalech denně, týdně, po čtrnácti dnech. Celkem je povoleno maximálně 52 opakování.



Obrázek 2.3: Rezervace v EVO

Pro využití systému se uživatel musí registrovat a jelikož je aplikace založena na Javě je potřeba mít nainstalováno Java runtime enviroment (JRE). Průběh konferencí zajišťuje kombinace klienta zvaného Koala a serveru Panda. Klienta je nutné stáhnout do uživatelského počítače, kde poběží jako součást sofwarové

infrastruktury a měl by být schopen pracovat pod třemi hlavními operačními systémy Windows, Linux, MacOS.

Hlavními prvky zobrazenými Koalou jsou seznam všech plánovaných konferencí a instant messaging v rámci celé komunity, zde nazvané Universe, případně jiných komunit, kterých je uživatel členem. Ke každé konferenci je zobrazeno několik základních informací jako je začátek, konec a počet připojených uživatelů.

V případě probíhajícího videokonferenčního hovoru je místo seznamu konferencí v Koale zobrazen jmenný seznam účastníků schůze a volby pro zapnutí a vypnutí doplňkových funkcí. Přenášené video je zobrazováno v separátním okně.

2.2 FlashMeeting

Flashmeeting vznikl jako akademický projekt zaměřený na studium on-line spolupráce. Jde o jednoduchý webový videokonferenční nástroj založený na pluginu od firmy Adobe. Tento systém umožňuje realizovat videokonferenční schůzku pouze pokud je předem rezervovaná, protože server poskytující videokonferenční službu má omezenou kapacitu zdrojů.

Rezervovat konference v systému může jen uživatel s patřičnými právy. Systém podporuje tři rozdílně privilegované uživatele Guest Account, Sign In Account a Booker Account. Guest account slouží pro uživatele připojující se jednorázově na základě pozvánky od jiného uživatele. Pro Sign In Account je potřeba se rezervovat. Tento účet je určen častým účastníkům konferencí, kteří ale nepotřebují zakládat konference vlastní. Posledním účtem je Booker Account, který poskytuje přístup ke všem možnostem systému. Jeho základní vlastností je povolení k rezervaci kapacity serveru a realizaci konference.[4]

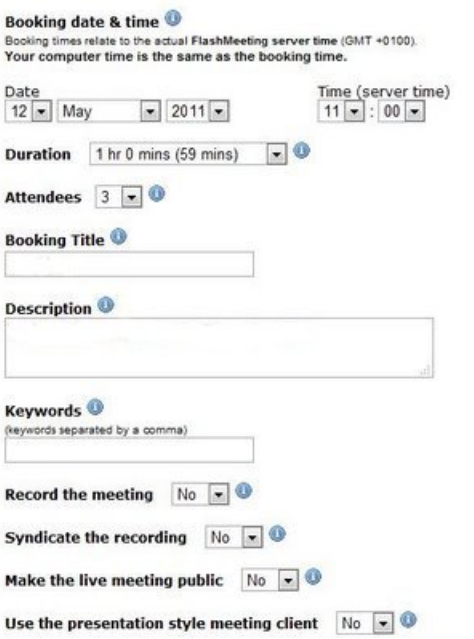
Pro vytvoření rezervace je potřeba zadat datum a čas začátku konference. Před tímto časem není možné schůzku začít. Doba trvání konference musí být přesná. Účastníci jsou pět minut před jejím koncem varováni zvukovým signálem. Po uplynutí nastavené hodnoty je spojení mezi uživateli přerušeno. Z důvodů omezených zdrojů videokonferenčního serveru je nutné zadat přesný počet předpokládaných uživatelů. Těm, jež přesahují tento počet nebude ke konferenci umoľněn přístup. Dalšími zadávanými parametry jsou název, stručná charakterizace účelu a klíčová slova týkající se konference.

Kromě základních parametrů je v rezervaci možnost povolit/zakázat nahrávání schůzky a případnou manipulaci s nahraným materiálem a volby pro různý typ zobrazení videokonferenčního okna.

Systém poskytuje i doplňkové služby pracující souběžně s videokonferencemi, tedy nahrávání schůzek, sdílení souborů, sdílenou tabuli, veřejný i soukromý instant messaging a další.

Celá videokonference se uskutečňuje přímo ve webovém prohlížeči. Komunikace probíhá pouze pomocí hromadného vysílání ostatním účastníkům, kteří se, pokud chtějí vysílat, zařadí do fronty čekajících na slovo. Je zde i možnost přerušit hovořícího a následně mu slovo vrátit.

Hlavními zobrazovanými prvky jsou video vysílané hovořícím, seznam ostatních účastníků schůzky, tlačítka pro zastavení/start vysílání a aktuální záložka s chatem, hodnocením hovořícího, sdílenými webovými adresami, extra službami nebo informacemi o systému.



Obrázek 2.4: Rezervace ve FlashMeeting

2.3 Sumarizace

Kolaborativní systémy EVO a Flashmeeting jsou webově orientované videokonferenční systémy umožňující vytvářet rezervace konferencí.

V rámci správy rezervací oproti systému EVO chybí Flashmeetingu možnost nastavit konferenci pro periodické opakování a je zde nevýhodný faktor omezení počtu účastníků. Pokud na někoho při rezervaci zapomenete nebude se moci připojit k probíhajícímu hovoru kvůli nedostatečné kapacitě systému. U tohoto systému je nutné mít ve skupině účastníků alespoň jednoho účastníka s pravomocemi pro tvorbu rezervací, jinak se spolupráce nebude moci uskutečnit.

Dalším problémem EVO je, že volby nastavené během rezervace není možné změnit v průběhu konference, což může být nepříjemné především u nahrávání schůzky.

Z hlediska poskytovaných funkcí je Evo také ve výhodě. Kromě základních funkcí, které mají s Flashmeetingem společné, nabízí další podpůrné služby, jako integrovanou bránu pro telefonické připojení ke konferenci, sdílení pracovní plochy nebo automatickou podporu časových zón.

Při hodnocení běžné využitelnosti mají oba systémy své problémové partie. U systému Evo mezi ně patří stažení klienta do uživatelského počítače, což se nemusí setkat s kladnou odezvou zabezpečení operačního systému. Také zobrazení všech účastníků právě probíhající konference v jednom okně a je při jejich vyšším počtu značně nepřehledná.

U Flashmeetingu působí největší problém neustále ?padající? plugin[5], což přerušuje probíhající videokonferenci. Nepříjemná je i nutnost počkat si na právo k hovoru, pokud nechcete násilně přerušovat právě vysílajícího uživatele.

Přestože jsou oba vybrané systémy vhodné pro kolaboraci uživatelů, nedají se využít k zajištění služeb v jiné infrastruktuře, než pro kterou byly vybudovány. V zásadě žádný ze systémů, na které jsem narazila, EVO a Flashmeeting nevyjímaje, není přenositelný do prostředí disponujícího vlastní databází uživatelů, což vylučuje nasazení v rámci České akademické federace identit. Dále ačkoli jsou oba systémy zaměřené na videokonference, dokonce umožňují systémovou kapacitu rezervovat, neřeší žádným způsobem koordinaci uživatelů při plánování schůzek.

Závěrem tedy je že mezi současnými kolaborativními systémy není žádný splňující cíle definované v úvodu práce. Z toho důvodu jsem se rozhodla vypracovat novou vyhovující aplikaci. Postup vývoje nového systému je popsán v následující části této práce.

5. `http://cs.wikipedia.org/wiki/Plugin/`

3 Systém pro správu rezervací a zdrojů kolaborativních systémů

Primárním cílem této práce je zajistit systém pro správu rezervací a zdrojů kolaborativních systémů. Aplikace musí podporovat práci v rámci České akademické federace identit eduID.cz. Příkladem kolaborativního systému, pro nějž budou rezervace určeny, jsou videokonference zprostředkované paketovými zrcadly. Před vlastním návrhem systému je ještě potřeba seznámit se s prostředím, pro které je určen, tedy federací eduID.cz a pojmem paketových zrcadel.

3.1 Česká akademická federace identit eduID.cz

3.1.1 O federaci

Cílem České akademické federace identit eduID.cz (dále jen eduID.cz) je poskytnout svým členům rámec pro vzájemné využívání identit uživatelů při řízení přístupu k síťovým službám za současné ochrany osobních údajů[1]. Členy jsou vysoké školy a jiné vzdělávací organizace sídlící v ČR. Pro umožnění přístupu k těmto službám federace využívá open source projektu Shibboleth od asociace Internet2. Shibboleth poskytuje nástroje pro single sign-on(SSO)[2]. SSO v podstatě znamená, že uživatel prokáže svou identitu (typicky zadáním jména a hesla) jen jednou, načež může volně využít síťové služby poskytované federací.

Každý uživatel má v rámci federace svou domovskou organizaci. Ta provozuje autentizační službu. Identita uživatele bude vždy ověřena autentizační službou jeho domovské organizace nehledě na to, kde leží služba, kterou chce využít. Domovská organizace zároveň poskytuje službě informace o uživateli, a je tedy poskytovatelem identity (Identity provider).

Druhou stranou celého procesu je chráněná služba a server, který ji poskytuje, označovaný jako poskytovatel služeb (service provider). V případě, že se kdokoli pokusí přistoupit k poskytované službě, je automaticky přesměrován na svého poskytovatele identity. V ideálním případě se zde prokáže jeho totožnost a vrátí se zpět na server, na kterém se nachází služba, kterou chtěl využít. Mohou ale nastat problémy pokud poskytovatel služby neví odkud uživatel pochází. Nebude tedy vědět, na kterého poskytovatele identit se obrátit. Tato situace je řešená službou WAYF[3] (Where Are You From). Ta se vloží do procesu hned potom, co se uživatel pokusí o přístup ke službě. Poskytující server zjistí, že neví,

1. K této problematice více: `http://www.eduid.cz/wiki/eduid/index`
2. `http://en.wikipedia.org/wiki/Single_sign-on/`
3. `http://www.eduid.cz/wiki/cztestfed/howto/wayf/index/`

odkud uživatel pochází, a přesměruje ho na službu WAYF. Klasickým chováním WAYF je, že předloží uživateli stránku s výběrem různých poskytovatelů identit. Poté, co si uživatel zvolí, ke kterému patří je na ni službou přesměrován. Díky tomu není potřeba, aby každý poskytovatel služeb znal všechny poskytovatele identit.[5]

Vzhledem k povaze federace jako kolaborativního prostředí je často potřeba, aby se mezi poskytovatelem identity a poskytovatelem služby předávaly informace o uživateli. Ty musí mít jednotnou strukturu, která je v případě federace eduID.cz definována třídou eduPerson. Tato třída je rozšířením standardní třídy inetOrgPerson (definována v RFC 2798)[4] o atributy potřebné pro vzdělávací instituce a vzdálené ověřování.

Operátorem federace je organizace CESNET[5], jež zajišťuje práci federace jako celku a provádí registraci členů, poskytuje uživatelskou podporu, řeší bezpečnostní incidenty či koordinuje dění ve federaci.

3.1.2 Paketové zrcadlo

UDP zrcadlo, neboli reflektor, umožňuje emulaci multicastu pomocí samostatných unicastových vysílání. Jednotlivá zrcadla jsou instancemi programu emulujícího některé vlastnosti multicastu za použití unicastového vysílání. Základní funkcí každého reflektoru je replikace a přeposílání dat, která se k němu dostanou od jednoho připojeného uživatele (odesilatele) ke všem ostatním, a samozřejmě i předávání dat od ostatních uživatelů směrem k odesilateli. Jednotlivé reflektory udržují mimo jiné i seznam ostatních zrcadel, která jsou navzájem propojena a tvoří tak komunikační síť.[6]

V současnosti používaná zrcadla jsou odvozena od reflektoru, který napsal roku 1998 Julian Highfield na universitě Loughborough. Původní RUM byl napsán v programovacím jazyce C, měl omezený a pevně daný počet účastníků a pracoval s protokolem RTP. Tento nástroj se však více nerozšířil kvůli neschopnosti splnit narůstající požadavky účastníků.

V rámci sdružení CESNET byla následně vytvořena nová implementace zrcadla, která zahrnovala několik funkcí nevyskytujících se v původním programu Juliana Highfielda. Reflektor RUM2 je napsán v jazyce C++ a využívá modulární přístup, což umožňuje přeposílaná data kromě replikace i zpracovávat. Oproti původní implementaci RUM2 dovoluje řídit reflektor během jeho provozu a měnit funkce, které poskytuje účastníkům. Pro komunikaci se zrcadlem, a tím i pro řízení přístupu a administraci byl použit RAP(Reflector Administration Protocol), díky kterému komunikace probíhá obousměrně. Mezi další funkce,

4. `http://www.rfc-editor.org/rfc/rfc2798.txt/`
5. `http://www.cesnet.cz/`

které jsou zlepšením vzhledem ke starší variantě, náleží například škálovatelnost komunikace nebo udržování provozních záznamů ("logů").

3.2 Návrh systému

Primárním účelem navrhovaného systému bude správa rezervací a zdrojů obecného kolaborativního systému. Nebude se zabývat konkrétním typem kolaborativního systému, bude k němu pouze delegovat informace zadané uživatelem. Rezervace služeb a zdrojů kolaborativního systému budou zajišťovat jeho schopnost uskutečnit plánovanou schůzku. Předejde se tak situacím, kdy v čase konference není na serveru dostatečné místo pro uložení dokumentů, nebo výpočetní kapacita pro realizaci audiovizuálního přenosu. V současných systémech se ošetření tohoto problému vyskytuje jen zřídka.

Dále bude aplikace poskytovat podporu koordinace uživatelů. Obecně platí, že existující systémy poskytují možnosti interaktivní spolupráce uživatelů v reálném čase, ale chybí jim integrovaný způsob jak takové schůzky naplánovat, tedy zajistit, aby všichni potenciální účastníci věděli, kdy a jakým způsobem bude spolupráce realizována. Prozatím se ke koordinaci účastníků využívají speciální aplikace, telefony, kalendářní systémy nebo jiné alternativní způsoby, což je zdlouhavé a chaotické. Podpora koordinace obsažená v systému má umožnit organizátorovi předat informace účastníkům bez potřeby opustit aplikaci, ve které se nachází.

Dalším řešeným problémem bude rezervace externích zdrojů. Externími zdroji budou potenciálně jakákoli zařízení (kamery, projektory, tablety a podobně), místnosti a služby externích systémů potřebné pro hladký průběh kolaborativního setkání. Jelikož tyto položky mohou mít omezené parametry, například počet, bude potřeba mít možnost jejich zamluvení předem.

V aplikaci bude implementováno i sdílení dokumentů před, během i po skončení kolaborativní schůzky. Bude tak vytvořeno úložiště souborů týkajících se probraného tématu.

3.2.1 Struktura systému

Hlavním nositelem informací bude v aplikaci rezervační záznam. Ponese všechny informace (dobu trvání, rezervované externí zdroje a podobně) využívané kolaborativním systémem. Toto pojetí vychází z charakteru současných aplikací pro zprostředkování vzdálené spolupráce. Pro ně platí, že každá probíhající schůzka využívá několik podobných parametrů nehledě na to, jestli je realizovaná audiovizuálním přenosem, chatem nebo třeba sdílenou tabulí. Tyto shodné parametry jsou základními daty v rezervačním záznamu. Typicky se jedná o název, čas

konání, popis a poznámky ke konferenci. Na rezervační záznam jsou vázány i volitelné služby podporující specifické typy kolaborace. Systém například bude umožňovat sdílení dokumentů, což by při použití aplikace třeba v kombinaci s kolaborativním systémem určeným ke sdílení dokumentů bylo duplicitní. Mezi volitelné parametry bude patřít zmíněné sdílení dokumentů, rezervace externích zdrojů a pozvání účastníků. Poslední položkou, kterou bude rezervační záznam obsahovat, budou informace potřebné pro správu aplikace. Všechny výše zmíněné údaje jsou blíže popsány v podkapitole Správa rezervačních záznamů.[6]

Součástí rezervační aplikace budou i funkce sloužící ke správě volitelných služeb vázaných na rezervační záznam. Tyto funkce za prvé umožní správu účastníků konferencí, respektive informací o uživatelích, kteří již v minulosti přistupovali k systému. Na základě čehož se zjednoduší manipulace se systémem. Pokud vznikne potřeba zjistit obecné informace o některém z uživatelů, nebude nutné hledat v jeho domovské organizaci, ale najdeme si daného uživatele ve vlastní databázi. Za druhé tyto funkce umožní správu externích zdrojů. V tomto případě bude platit, že aby bylo možno jakýkoli zdroj rezervovat, musí být nejprve uložen v systémové databázi zdrojů. Budou tak omezeny případy rezervace jednoho zdroje vícekrát na stejný čas. Podrobný popis správy volitelných služeb se nachází v podkapitole Správa systému.[7]

Navrhovaný rezervační systém bude podporovat dva typy uživatelů a dle jejich pravomocí budou funkce systému rozděleny do dvou skupin. Toto rozdělení je zvoleno pro dosažní přehlednosti systému za současného udržení jeho funkcí. Zároveň tím bude umožněna lepší ochrana citlivých údajů uživatelů.

Prvním typem uživatelů budou takzvaní běžní uživatelé, kteří přistupují k systému s cílem vytvořit rezervaci pro kolaborativní službu a nezajímají se o nic jiného. Po přihlášení do systému jim bude k dispozici zobrazení informačních zpráv (aktualit) o stavu systému a přístup ke správě vlastních rezervací. Nebude jim umožněno zadávat do systému nové externí zdroje a upravovat informace o jiných uživatelích a především budou moci přistupovat pouze k rezervačním záznamům u kterých figurují v seznamu účastníků nebo které vlastní.

Druhým typem uživatelů budou administrátoři. Po přihlášení budou moci k aplikaci přistupovat jako běžní uživatelé, ale i jako její správci. V roli správců jim bude umožněno upravovat cizí rezervační záznamy a spravovat externí zdroje včetně zadávání nových do systému. V jejich možnostech bude také spravovat systém i rezervační záznamy.

6. Str.15 této práce
7. Str.17 této práce

3.2.2 Správa rezervačních záznamů

Tvorba rezervačního záznamu. Hlavním účelem systému bude vytvářet rezervace kolaborativních schůzek. Při rezervaci budou nejprve zadány základní údaje o budoucí konferenci. Povinnými údaji bude název, datum a čas začátku i konce konference, vlastník a status. Volitelné bude vložení základní charakteristiky obsahu konference a poznámek.

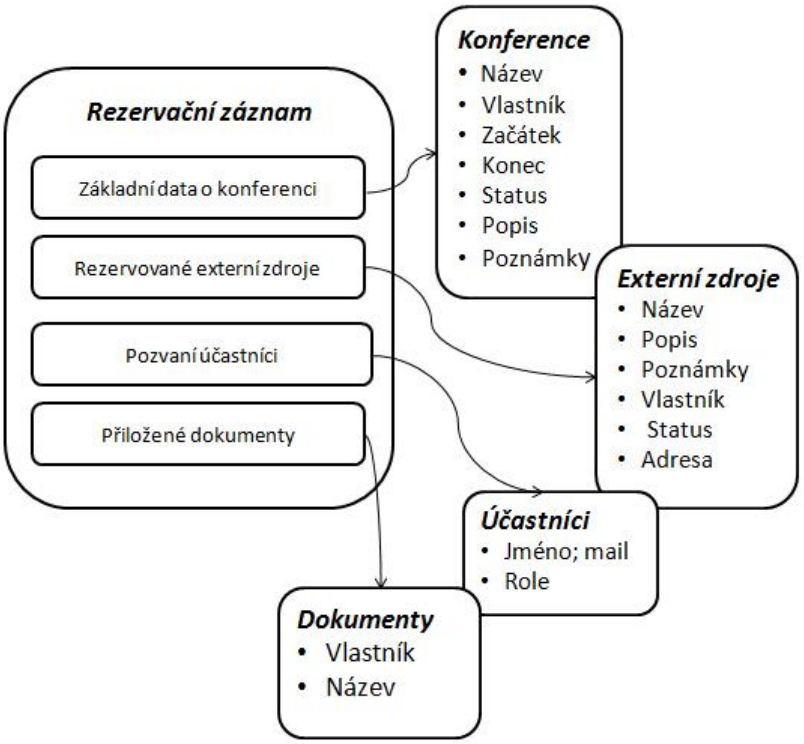
Povinné položky budou esenciální pro realizaci plánované schůzky kolaborativním systém a pro rezervační systém ke správě rezervačního záznamu. Název bude mít jen informativní hodnotu pro uživatele. Datum a čas začátku a konce konference budou většinou podstatný jak pro systém realizující schůzku, tak pro její účastníky. Některé systémy poskytující možnosti ke vzdálené spolupráci tyto hodnoty považují za orientační údaj. Jiné naopak striktně dodržují předem danou dobu trvání schůzky, a ukončují spojení účastníků po jejím uplynutí. Jak bude kolaborativní systém přistupovat k času obdrženému od rezervační aplikace bude jen na něm. Vlastník bude pro rezervaci velmi důležitým atributem, protože s výjimkou administrátora bude jediný oprávněný s ní manipulovat. Za vlastníka bude automaticky volen zakladatel rezervace, což bude možné změnit na na kohokoli z databáze uživatelů. Status bude informovat uživatele o aktuálním stavu, v jakém se konference nachází. Možnosti budou in preparation, pro konferenc,i která ještě neproběhla, running charakterizující právě probíhající konferenci, done označující proběhlou konferenci a canceled značící konferenci zrušenou ještě před jejím začátkem.

Po zadání základních údajů bude mít uživatel možnost rezervovat externí zdroje, které budou pro čas konference k dispozici. Uvidí i zdroje, které budou pro daný čas již zabrané a v případě potřeby bude moci přenastavit čas konference na dobu, kdy bude jím požadovaný zdroj dostupný.

Následující část tvorby rezervačního záznamu umožní pozvat ke konferenci účastníky. Pozvánky budou realizovány formou hromadných emailů oznamujících adresátům základní informace o budoucí schůzce a odkaz na detail rezervačního záznamu. Účastníci budou moci být z řad uživatelů, kteří již dříve se systémem pracovali a jsou v jeho databázi, nebo se bude jednat o uživatele, kteří ještě s aplikací neměli nic společného. V prvním případě bude stačit uživatele vybrat ze seznamu uživatelů systému. Ve druhém případě bude muset zakladatel záznamu zadat emailovou adresu zvaného účastníka ručně. V této fázi bude také možné přiřadit účastníkům role, které budou zastávat v rámci konference. Ty budou určovat omezení jejich komunikace. Role budou následující: "participant" , tedy účastník jenž může vysílat i přijímat data; "speaker" jenž může pouze vysílat; "moderator" který může může přijímat i vysílat a může měnit práva ostatních za běhu kolaborativní služby, pokud to její poskytovatel umožní. Tyto tři role jsou zobecněním všech nejčastěji se vyskytujících pravomocí účastníků

v existujících kolaborativních systémech.

Poslední fází rezervace bude eventuální nahrání dokumentů pro sdílení s ostatními účastníky konference.



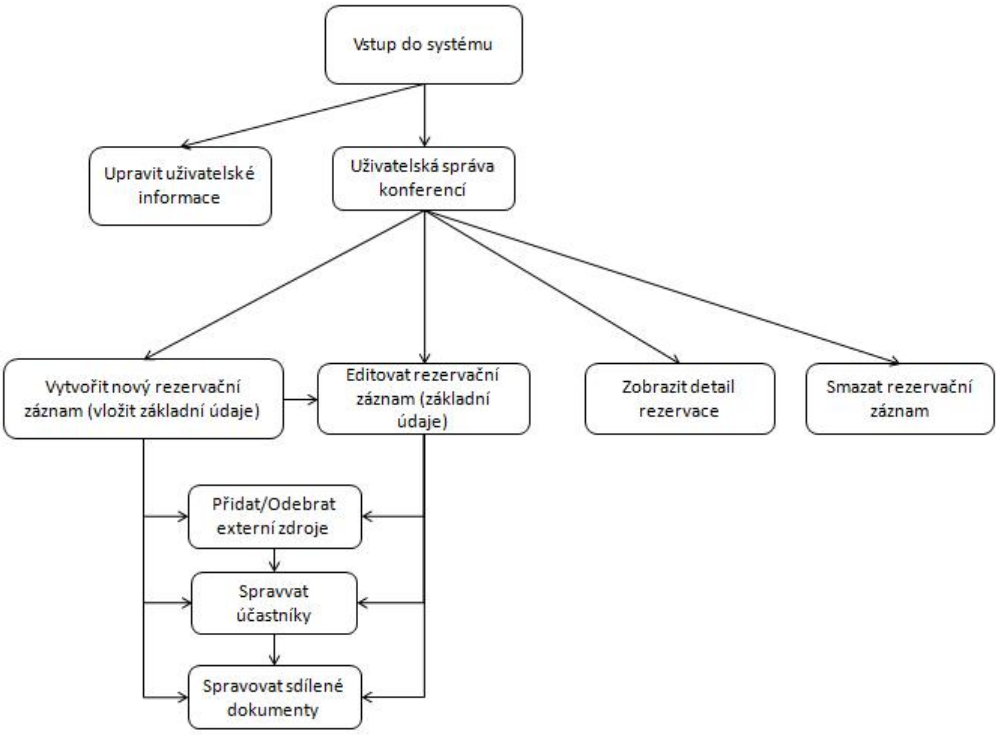
Obrázek 3.1: Správa záznamů

Úprava rezervačního záznamu. Upravovat rezervaci bude moci jen její vlastník nebo administrátor. Ze základních údajů bude možné změnit vše kromě názvu. Na odebrání či přidání externího zdroje nebudou jiné limitace než časové. Požadovaný zdroj v jiném čase než původním už nemusí být k dispozici. Protože bude ke každé rezervaci udržován seznam pozvaných účastníků včetně jejich emailů bude možné kdykoli poslat kterémukoli z nich informační zprávu o změně údajů. Na přidání a odebrání sdílených dokumentů nebudou kladeny žádné podmínky.

Zobrazení rezervačních podrobností. Tato volba umožní zobrazení celého rezervačního záznamu jen pro čtení. Vypíší se základní údaje, seznam účastníků

s jejich rolemi a stavem pozvánky, rezervovaná zařízení a názvy sdílených dokumentů.

Smazání rezervačního záznamu. Rezervační záznam bude moci smazat pouze jeho vlastník nebo administrátor.



Obrázek 3.2: Provázání prvků správy konferencí pro běžného uživatele

3.2.3 Správa systému

V sekci správy systému bude možné zpravovat rezervační záznamy, libovolné kon, že administrátor nebude jejím vlastníkem. Dalšími položkami spadajícími pod správu rezervačního systému budou služby pro správu uživatelů, správu zdrojů.

Správa uživatelů Do správy uživatelů patří zobrazení seznamu všech uživatelů, jejich detailních informací s možností úpravy a smazání uživatele. Jaké podrobnosti o uživateli jsou zobrazeny závisí na tom, jaké informace poskytla

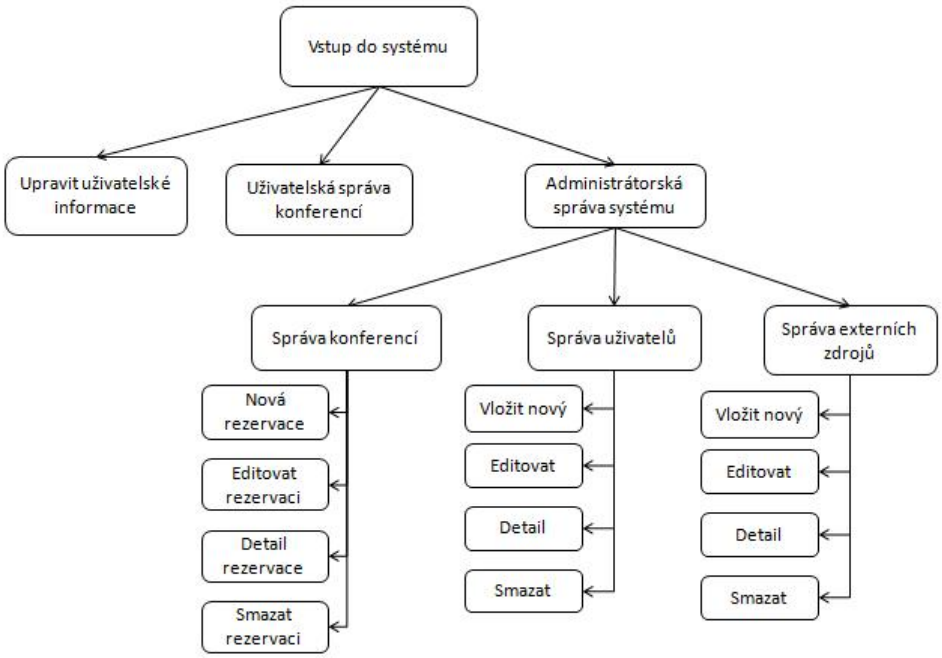
jeho domovská organizace při jeho přihlášení do systému, a na případných změnách za strany administrátora nebo uživatele samotného. Informace o uživatelích přistupující k systému budou uchovány v systémové databázi. Tímto se značně zjednoduší správa systému a manipulace s uživateli při správě rezervačních záznamů.

Prozatím bude aplikace zaměřena na nasazení v rámci federace eduID.cz, detaily o uživateli tudíž zprostředkuje poskytovatel identity uživatele. Očekávané budou atributy $eduPersonPrincipalName$, sn, cn, o, $eduPersonAffiliation$ a $mail$. Atribut $eduPersonPrincipalName$ slouží pro jednoznačnou identifikaci uživatele mezi členskými organizacemi. Atribut sn (surname) typicky obsahuje příjmení uživatele, cn (common name) většinou obsahuje celé jméno uživatele a o (organization name) název domovské organizace uživatele. Atribut $eduPersonAffiliation$ charakterizuje vztah uživatele k jeho domovské organizaci (student, zaměstananec, a tak dále.) a poslední očekávaný atribut, $mail$, je preferovanou emailovou adresou uživatele v rámci federace.[7]

Kromě atributů získaných od poskytovatele identit bude moci sám uživatel nebo administrátor přidat atribut $publicmail$, reprezentující sekundární emailovou adresu pro případ problémů s adresou poskytnutou jeho domovskou organizací. Dalšími položkami bude volba statutu a administrátorských pravomocí. Status bude popisovat aktivitu uživatele vzhledem k systému. Možnosti budou "active" , pokud bude častým uživatelem systému, tato hodnota bude implicitní pro všechny uživatele. "Passive" pokud uživatel delší dobu nepracoval se systémem nebo již není členem federace. Poslední položkou bude identifikace uživatelského účtu, tedy určení zda je uživatel administrátorem. Přítomny budou dvě možností. Implicitní hodnota "no" charakterizující běžného uživatele a hodnota "yes" určující administrátorská práva.

Správa zdrojů V rámci správy zdrojů je možné vkládat nové záznamy o zdrojích. Ke každému zdroji bude potřeba dodat název, stručný popis a případně poznámky, přičemž poslední dvě položky nejsou povinné. Dále je nutné zadat vlastníka zařízení a status určující stav v jakém se zařízení právě nachází. Vlastník bude mít pouze informativní hodnotu a implicitně tak bude označen zadavatel zdroje. Vlastníka bude možné změnit na kohokoli, kdo bude uložen v databázi systému. Status poslouží k přehledu stavu zařízení. Může nabývat hodnot "active" , "in preparation" nebo "canceled". Automaticky je u každého nastaven status "active", oznamující uživatelům bezproblémovou funkčnost zařízení. Pro případy dočasných problémů se zdrojem bude určena hodnota "in preparation". Pokud nebude zdroj z jakéhokoli důvodu dlouhodobě k dispozici bude být jeho stav změněn jeho vlastníkem nebo administrátorem systému na "cancled". Poslední položkou bude adresa na které bude zdroj k nalezení.

Dále bude možné zdroj vymazat z databáze nebo zobrazit jeho detailní výpis, který bude obsahovat všechny informace zadané při jeho vkládání do systému.



Obrázek 3.3: Provázání prvků správy systému

3.3 Návrh implementace

Po specifikaci požadovaných funkcí a promyšlení jejich kombinace a prezentace uživateli bylo přistoupeno k návrhu jejich realizace. Než bude možné se popisu do vlastního popisu návrhu implementace, bude potřeba popsat návrhové vzory, které byly pro návrh implementace využity.

3.3.1 Návrhové vzory

Protože bylo již na počátku této práce rozhodnuto, že implementace bude provedena objektově orientovaným přístupem, nabídlo se využití příslušných návrhových vzorů. Ty popisují obecné řešení častých problémů vyskytujících se při navrhování systémů. Můžeme si je představit jako šablonu použitelnou v různých situacích[8].

Nejedná se o knihovny nebo jiný typ souborů, které by se daly vložit do programu, ale spíše o vodítko k tomu jak program sestavit. V rámci OOP[9]. se jedná většinou o popis vztahů a chování mezi jednotlivými třídami a objekty. Při této práci byly využity návrhové vzory Model-View-Controler a Jedináček.[8]

3.3.1.1 Model-View-Controler

Tento vzor do značné míry zasahuje do architektury aplikace, často jej tedy najdeme pod označením architektonický vzor spíš než návrhový. MVC je využíván pro tvorbu webových aplikací a frameworků[10]. Ačkoli existuje více variant tohoto vzoru, které se rozcházejí v detailech, typicky jde jen o rozdílně silné provázání mezi jednotlivými segmenty aplikace. Základní myšlenkou MVC je rozložení aplikace na tři rozdílné části do češtiny přeložené model, pohled a řadič.

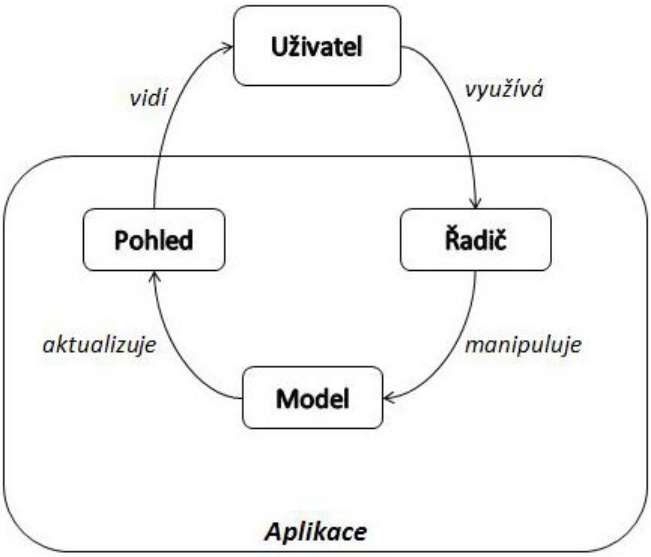
- Mode l(model) - je zodpovědný za datovou vrstvu aplikace. Rozhodně neplatí, že by model byl srovnatelný s databází. Tato část MVC zahrnuje i funkce pro práci s daty. Zpracovává jakékoli dotazy týkající se dat jako jsou aktualizace nebo čtení a zapisování údajů.
- Pohled (view) - spravuje presenční vrstvu aplikace. Dostává údaje které jsou určeny na vypsání a může mít vlastní presenční logiku, která výpis zajistí. Pohled si můžeme představit jako jednotlivé stránky zobrazované uživateli. Ve striktním pojetí MVC architektury by zobrazované údaje měly pocházet pouze od řadiče, v některých případech je ale možné se setkat i s daty předanými modelem. Ve skutečnosti by mělo být pohledu naprosto jedno, odkud data pocházejí, důležité je poznat jejich typ a formát.
- Řadič (controler) - jak už název napovídá řadič je zodpovědný za řídící vrstvu. Získává informace o přáních uživatele z pohledů a předává dotazy modelu. Následně od modelu získá potřebné informace a předá je

8. http://www.tonymarston.net/php-mysql/model-view-controller.html/
9. Objektove Orientované Programování http://en.wikipedia.org/wiki/Object-oriented_programming/
10. http://cs.wikipedia.org/wiki/Framework/

pohledu k zobrazení. Řadič se často skládá z více komponent, které sami o sobě mohou implementovat úplně jiné návrhové vzory.

3.3.1.2 Singleton

Pomocí návrhového vzoru Singleton (Jedináček) zajišťujeme, že v určitý čas existuje pouze jedna instance určité třídy. To je potřeba například při komunikaci s databází, kdy není vhodné současné otevření více než jednoho komunikačního kanálu.



Obrázek 3.4: Princip modelu MVC

3.3.2 Funkce modelu v rámci rezervačního systému

Model, podle návrhového vzoru MVC, bude v systému reprezentován datovým úložištěm a objektovými třídami s ním manipulujícími.

3.3.2.1 Datové úložiště

Data využívaná systémem budou uchována v databázových tabulkách. Budou dělitelné na dva typy – statické číselníky a relační tabulky. Statické číselníky budou udržovat data, u kterých se nepočítá s častými změnami. Relační tabulky budou uchovávat informace o spojitostech mezi číselníky.

Statické tabulky První statická tabulka bude udržovat základní informace o rezervačních záznamech. Bude vedena pod názvem $tblConference$ a její položky naplní uživatel během rezervace kolaborativní schůzky. Tabulka bude obsahovat jednoznačný identifikátor rezervace v rámci systému a všechny povinné základní údaje o budoucí konferenci. Atributy uložených záznamů budou jednoznačný identifikátor rezervace v systému ($IDConference$), název budoucí konference ($cnfName$), čas začátku ($cnfStar$) a konce schůzky ($cnfFinish$), identifikační číslo vlastníka ($cnfOwnerID$), popis charakteru konference ($cnfDescription$), poznámky ($cnfNote$) a status ($cnfStatusENUM$). Jejich význam je popsán v kapitole Správa rezervačních záznamů[11].

Druhá tabulka, řadící se mezi statické, bude udržovat informace o všech uživatelích, kteří kdy přistoupili k systému. Bude tak zjednodušovat manipulaci s uživateli a správu systému. Ponese název $tblUser$ a krom identifikátoru uživatele v rámci systému bude obsahovat především údaje poskytnuté domovskou organizací uživatele. Celkem bude udržovat dvanáct atributů. Prvním bude jednotná identifikace v rámci systému (IDUser) ta bude vycházet z druhého atributu $PersonPrincipalName$. Následující atributy $CommonName$, $SurName$, $OrganizationName$, $PersonalAffiliation$ a $Mail$ budou odpovídat atributům získaným od poskytovatele identity uživatele. Status, administrace, veřejný mail a případné poznámky budou uchovány v atributech $usrStatusENUM$, $usrAdminENUM$, $usrMailPublic$ a $usrNote$. Popis významu všech atributů se nachází v kapitole Správa systému.[12]

Třetím číselníkem bude $tblPhysicalSource$ udržující informace o externích zdrojích, které budou k dispozici systému. Údaje bude do této tabulky dodávat administrátor při vyplnění formuláře o novém zdroji. V tabulce budou atributy $phsName$ obsahující název zdroje a $phsOwnerID$ určující vlastníka. $PhsDescription$ a $phsNote$ poslouží k obecné charakterizaci zdroje. $PhsStatusENUM$ bude definovat status zdroje. Všechny obsažené atributy jsou opět popsány v kapitole Správa systému.[13]

Relační tabulky Relační tabulky budou spojovat rezervace s externími zdroji (tabulka $tblRelConfPSrc$), uživateli ($tblRelConfUser$) a cestami k dokumentům ($tblRelConfDoc$). V prvních dvou tabulkách bude vždy uchován identifikátor rezervačního záznamu a podle toho, jestli se jedná o propojení rezervačního záznamu s uživatelem nebo se zdrojem bude dalším atributem identifikátor příslušného uživatele či zdroje. Dalším společným údajem je začátek a konec plánované schůzky. Obě tabulky také budou uchovávat atribut status. Pro externí

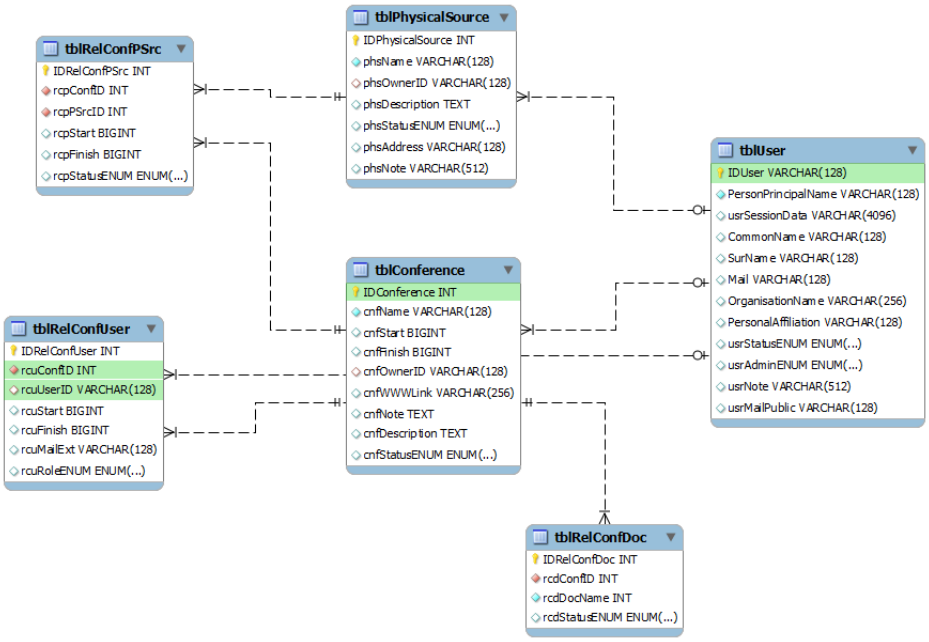
11. Str.15 této práce
12. Str.17 této práce
13. Str.17 této práce

zdroje to bude "active" a "canceled" . Status v relační tabulce mezi uživateli a rezervačním záznamem určí roli uživatele při účasti na konferenci. Možnosti budou "praticipant" pro účastníka s možností vysílat i přijímat data, "speaker" pro účastníka jehož jedinou možností bude vysílání. "Moderator" pro účastníka moderujícího konverzaci. Poslední relační tabulka bude udržovat identifikátor rezervace a cestu k uloženému dokumentu.

3.3.2.2 Manipulace s datovým úložištěm

Jednotlivé tabulky databáze budou reprezentovat objekty ve smyslu OOP. Hlavní třídou poskytující metody pro zajištění inicializace a uzavření spojení s databází bude cDB. Metody této třídy přistupují přímo k datovému úložišti a v případě potřeby ošetřují vzniklé chyby. Tato třída bude dědičná ostatním třídám obsahujícím metody pro zpracování dat. Mezi tyto metody budou patřit $cDBConf$, $cDBUser$ a $cDBPSrc$. Všechny budou mít velice podobné vlastnosti a účel. Budou obsahovat metody sestavující jednotlivé dotazy pro databáze, které budou následně metodami ze třídy cDB provedeny.

Takovéto rozdělení značně podpoří bezpečnost a přehlednost implementace. Pokud by práci s databází zajišťovala jedna jediná třída s množstvím metod, byla by implementace netriviální a chaotická.



Obrázek 3.5: Databázový model

3.3.3 Funkce pohledu v rámci rezervačního systému

Vše, na co se může uživatel přímo podívat, a metody, které to budou zobrazovat, patří do pohledu. Návrh grafického rozložení stránek vycházel z funkcí požadovaných po systému. V předcházejících kapitolách jsou uvedeny struktury a podrobnosti rozdělení funkcí systému, z těchto údajů byl vytvořen rámec pro to co má která zobrazovaná stránka vykonat a jaké prvky k tomu poslouží. Například pro zobrazení seznamů uložených dat bude využito tabulky, pro zadání dat určených k uložení do databáze poslouží formuláře a tak dále.

K návrhu konkrétního rozložení prvků na zobrazovaných stránkách byly vytvořeny prototypy jednotlivých stránek. Tyto prototypy prošly velkým množstvím úprav a jejich výslednou podobu můžete shlédnout v elektronické příloze této práce. Způsob prototypování obrazovek byl výhodný díky lehké realizaci a okamžité viditelnosti změn. K jejich tvorbě byl využit OpenOffice.org Impress verze 3.2[14]. Jelikož jsou si některé stránky svou stavbou podobné nebylo potřeba vytvářet prototypy každé z nich, ale jen reprezentativního vzorku.

Třídy, jejichž metody budou vytvářet a kompletovat vzhled stránek, budou rozděleny do dvou skupin určených charakterem zobrazovaných údajů. Faktu, že si budou některé stránky v mnohém podobné tak bude využito i pro jejich zobrazení. První skupinou bude prezentace seznamů dat získaných z modelu, tedy seznamy číselníků. Jejich zobrazení budou zajišťovat dědici třídy $cPageList$. Metody samotné třídy $cPageList$ se budou soustředit na prvky shodné pro všechny stránky tohoto typu. Jejími dědici budou třídy zajišťující zobrazení konkrétních seznamů a volající metody z mateřských tříd, čímž zkompletují vzhled stránky. Bude platit, že se seznamy pracujícími s různými typy dat budou pracovat různé třídy.

Druhá skupina metod bude pracovat se zobrazením detailních formulářů a výpisů. Příkladem může být stránka sloužící pro zadání základních dat rezervačního záznamu nebo zobrazující detailní informace o nějakém uživateli. Protože i tyto typy budou mít společné vlastnosti bude existovat třída $cPageDetail$ která se na ně zaměří. O vypsání rozdílných obsahů stránek se budou starat třídy $cPageDetailConf$ v případě správy detailních informací rezervačních záznamů. $CpageDetailPsrc$ bude ošetřovat zobrazení pro správu externích zdrojů a $cpageDetailUser$ výpisy správy uživatelů.

Všechny třídy obsahující metody zaměřené na prezentaci dat uživateli budou dědit třídu $cPage$. Ta bude sloužit pro zajištění všeho potřebného pro zobrazení neustále se opakujících položek stránek jako nadpisy, hlavičky a patičky stránek a další. Její součástí budou i metody pro testování vstupních dat zadávaných uživatelem a zobrazení oznámení o chybách, pokud budou zadaná data nekorektní.

---

14. `http://www.openoffice.cz/`

3.3.4 Funkce řadiče v rámci rezervačního systému

Obecně je řadič prostředníkem mezí uživatelem a modelem. V navrhovaném systému nebude jako jediný řešený objektovým způsobem, protože by to nepřineslo v rámci implementace žádné výhody. Jeho úkolem bude zachytit informace o akcích uživatele a podle typu akce rozhodnout, co dál. Například po stisknutí odkazu na libovolné stránce příslušný řídící skript zachytí tuto akci a přesměruje uživatele k jeho cíli.

4 Implementace

Pro implementaci aplikace byl využit skriptovací programovací jazyk PHP verze 5. Využito je především jeho objektově orientované podpory. Hlavním důvodem pro využití PHP je jeho specializace na tvorbu webových stránek. Protože vyvíjená aplikace musí být dostupná přes Internet, bylo potřeba využívat jazyka, který to umožní. Další motivací pro jeho využití byla jeho nativní podpora různých databázových systémů a rozsáhlá uživatelská základna. Všechny ostatní technologie využité k implementaci systému byly vybírány tak, aby s jazykem PHP co nejlépe pracovaly.[9]

Jako server provádějící PHP skripty byl použit Apache[1]. Kombinace PHP a Apache je prověřená stovkami uživatelů. Pro realizaci databáze je využito systému MySQL[2]. Tento typ databáze má v prostředí PHP propracovanou podporu. Informace uchovávané v systému jsou ukládané v úložišti dat typu InnoDB[3]. Tento typ databáze je vhodný pro prostředí, které se často mění, čímž webová aplikace rozhodně je. Pro založení databáze rezervačního systému slouží soubor rsCreateDatabase.sql. Po jeho spuštění v systému MySQL, bude vytvořena databáze $dbRS$ a tabulky $tblConference$, $tblPhysicalSource$, $tblUser$, $tblRelConfPSrc$, $tblRelConfUser$, $tblRelConfDoc$. Význam těchto tabulek je popsán v kapitole Návrh implementace[4]. Metody bezprostředně pracující s datovým úložištěm využívají k přístupu rozšířenou podporu PHP MySQLi (MySQL Improved)[5], která kromě procedurálního přístupu ošetřuje i přístup objektový.

Stránky jsou uživateli zobrazovány značkovacím jazykem XHTML. Tento striktní značkovací jazyk se vyznačuje vynucováním totální párovanosti znaků a to i u běžně nepárových tagů jako je <br>, <hr> nebo <img>. Znamená to, že všechny tagy musí mít počáteční i koncový tag. Třídy realizující zobrazení jsou uloženy ve stejnojmenných souborech s příponou php. XHTML kod je v jednotlivých php skriptech ukládán v podobě řetězce do jednoduchého pole, které je vráceno jako výstup metody. Základní třída pro zobrazování stránek je cPage, protože vlastní metodu $showHTML$($f_array), která slouží pro výpis xhtml kódu při vyhodnocení skriptu na straně serveru. Metoda $showHTML$($f_array) během vyhodnocování odešle vygenerovanou stránku do prohlížeče sérií příkazů echo(). Prohlížeč tedy dostane už jen čistou XHTML syntaxi, kterou zobrazí. Metoda $showHTML$($f_array) má jeden parametr, kterým je pole obsahující v každé položce jeden řádek HTML tagů. Dalšími důležitými funkcemi obsa-

---

1. `http://www.apache.org/`
2. `http://www.mysql.com/`
3. `http://dev.mysql.com/doc/refman/5.0/en/innodb-storage-engine.html/`
4. `stránka`
5. `http://php.net/manual/en/book.mysqli.php/`

ženými v této třídě jsou metody pro testování vstupních parametů zadaných uživatelem. Jedná se o běžné testy na požadovanou délku zadávaného řetězce, kontrolu korektně zadaných emailových adres a podobně.

Řídící funkce v systému jsou uloženy v souborech $ctrlConf.php$, $ctrlPSrc.php$ a $ctrlUser.php$. Jako jediné v systému není řízení implementováno objektově orientovaným přístupem, ale imperativně.

Každý skript obsahuje dva hlavní prvky, funkci $getActionXXXX()$ a přepínač (switch). Za XXXX lze doplnit libovolnou ze zkratek $Conf$, $Psrc$, $User$ podle toho. jaký aspekt systému má ošetřovat. Funkce $getActionXXXX()$ detekuje akci na stránkách (akce tlačítek, odkazů) a jejím výstupem je identifikátor akce. Ten se stane jediným parametrem přepínače. Switch zadaný parametr vyhodnotí a provede příslušnou reakci. Pokud bychom se pohybovali ve správě uživatelů aplikace, bude funkce $getActionUsers()$ uložená v souboru $ctrlUsers.php$. Při kliknutí na odkaz sloužící pro zobrazení detailů o uživateli by funkce $getActionUsers()$ detekovala naše počínání a identifikátor akce by předala přepínači. Switch by nás na základě dané akce přesměroval na stránku obsahující výpis požadovaných informací.

V systému je implementován i mechanismus session (sezení) pro předávání parametrů mezi stránkami. Předávání informací mezi jednotlivými stránkami je obecně možné pomocí cookies nebo v parametrech URL. Nevýhodou cookies je jejich značná nespolehlivost protože jsou uživateli v současné době hojně zakazována. Parametry URL adresy nejsou vůbec zabezpečeny. Pro eliminaci nevýhod předchozích dvou způsobů budou informace mezi stránkami aplikace přenášeny ve speciálních proměnných pole $_SESSION. Toto pole je speciální pro jazyk PHP.[9]

Rezervační systém udržuje v tabulce uživatelů pro každého uživatele položku $usrSessionData$. V této položce je udržován řetězec vniklý po uplatnění metody $session_encode()$ na pole $_SESSION. Při dalším přístupu uživatele do systému je při startu sezení využita metoda $session_decode()$, která naopak údaj v $usrSessionData$ dekóduje. Výsledek dekódování je zapsán do pole $_SESSION.

5 Rozšíření do budoucna

Rozvržení a implementace rezervačního systému usiluje o udržení co největší přehlednosti. Mohlo by se zdát, že velké množství vytvořených tříd a souborů aplikaci jako celek znepřehlední. Avšak ve chvíli, kdy bude zájem o rozšíření její funkcionality, se ukáže, že právě tato vlastnost umožňuje přidání funkce či služby s minimálními zásahy do již existující struktury.

Aplikace by výhledově mohla pracovat i s více kolaborativními systémy současně. Potom by bylo nutné začlenit informace týkající se těchto systémů do struktury aplikace a vytvořit funkce starající se o jejich správu. Rezervační záznamy by tak dle vybraného kolaborativního systému mohly nabízet rezervaci služeb a zdrojů specifických jen pro ten konkrétní systém.

Potenciální rozšiřující službou by mohlo být vyhledávání uživatelů, zdrojů nebo rezervačních záznamů dle jejich atributů. Napomohlo by to administrátorům při správě aplikace, ale i běžným uživatelům pro zorientování se v systémových položkách.

Další možností pro zvýšení komfortu uživatelů by bylo tvoření periodických rezervací. Uživatelé konající pravidelné konference by tak nemuseli vytvářet stále nové rezervace. K tomuto rozšíření by se dalo navázat i tvorbou takzvaných mailing listů. Ty by sloužily při tvorbě rezervačního záznamu k primárnímu zobrazování seznamu uživatelů, se kterými vlastník rezervace dříve komunikoval.

Jiným příkladem by mohlo být přidání služby informující o aktuálních změnách v systému. Informativní služba by tak ošetřila například výpadky systému, kolaborativních služeb nebo problémů s externími zdroji.

6 Závěr

Výsledkem této práce je webová aplikace stručně charakterizovatelná jako mezičlánek pro potřeby uživatelů plánujících vzdálenou spolupráci a kolaborativním systém, umožňujícím realizaci vzdálené spolupráce. Soustřeďuje se na rezervaci služeb a zdrojů systémů pro vzdálenou spolupráci. Mimoto umožňuje rezervovat pro specifikovaný čas externí zdroje (zařízení, místnosti a podobně) a implementuje služby pro koordinaci účastníků budoucích kolaborativních jednání. Dále aplikace umožňuje sdílení dokumentů, správu uživatelů a správu externích zdrojů.

Sdružení všech výše zmíněných funkcí do jednoho systému představuje řešení největšího problému současné vzdálené spolupráce. Tímto problémem je potřeba využití několika separátních, vzájemně nekomunikujících systémů k přípravě a realizaci jediného kolaborativního setkání.

Důležitým rysem aplikace je nepřítomnost služby pro autentizaci uživatelů. Je tak umožněna její využitelnost jako součásti již existujících systémů disponujících vlastní autentizační službou.

Výstupem systému bude kompletní rezervační záznam, dle kterého bude možné nastavit parametry systému realizujícího kolaborativní setkání. Forma a způsob předání získaných údajů kolaborativnímu systému prozatím nejsou určeny. Budou záviset na vlastnostech konkrétního kolaborativního systému.

Literatura

[1] HOŠKOVÁ, K. *Kolaborativní systémy*. [online] 2008, [cit. 2011-05-22] Dostupná z WWW: http://kisk.phil.muni.cz/wiki/Kolaborativn\%C3\%AD_syst\%C3\%A9my.

[2] DEAN, T. *Network+ Guide to Networks*. USA, 2009. ISBN 1-423-90245-9.

[3] *EVO User guide*. [online] 2008, [cit. 2011-05-22] Dostupná z WWW: http://evo.vrvs.org/evoGate/help.jsp?EvO_Manual.

[4] *Flashmeeting About*. [online], [cit. 2011-05-22] Dostupná z WWW: http://flashmeeting.open.ac.uk/about.html.

[5] SATRAPA, P. *Shibboleth - identifikujte se jen jednou*. [online] 2005, [cit. 2011-05-22] Dostupná z WWW: http://www.lupa.cz/clanky/shibboleth/.

[6] *Videokonference::Zrcadlo*. [online], [cit. 2011-04-12] Dostupná z WWW: http://miro.cesnet.cz/about_the_project/reflector.cz.html.

[7] *EduPerson Specification(200312)*. [online] 2003, [cit. 2011-05-22] Dostupná z WWW: http://middleware.internet2.edu/eduperson/docs/internet2-mace-dir-eduperson-200312.html.

[8] BÖHMER, M. *Zend Framework Programujeme webové aplikace v PHP*. Computer Press, Brno, 2009. ISBN 978-80-251-2965-4

[9] GILMORE, W. J. *Velká kniha PHP a MySQL 5 - kompendium znalostí pro začátečníky i profesionály*. Zoonerpres, leden 2007. ISBN 80-86815-53-6

[10] *Česká akademická federace identit eduID.cz*. [online], [cit. 2011-05-20] Dostupná z WWW: http://www.eduid.cz/wiki/eduid/index.

[11] DENEMARK JIŘÍ, HOLUB PETR, HLADKÁ EVA. *RAP - Reflector Administration Protocol (RAP - Reflector Administration Protocol). : CESNET z.s.p.o., 2003. 32 pp. 9/2003*. [online] [cit. 2011-05-22] Dostupná z WWW: http://www.cesnet.cz/doc/techzpravy/2003/rap/rap.pdf.

[12] HLADKÁ EVA, HOLUB PETR, PROCHÁZKA MICHAL. *Videokonference za zdí. Zpravodaj ÚVT MU. 2007, roč. XVII, č. 5, s. 8-12*. ISSN 1212-0901

[13] NOVAKOV, I. *Shibboleth Identity Provider And User Privacy*. [online] 2010, [cit. 2011-05-22] Dostupná z WWW: http://www.cesnet.cz/doc/techzpravy/2010/shibboleth-privacy/.

[14] *Point-to-point (telecommunications)*. [online], [cit. 2011-05-22] Dostupná z WWW: http://en.wikipedia.org/wiki/Point-to-point_(telecommunications).

[15] ŘÍHA, D. *Trendy ve výzkumu počítačově podporované kolaborace v kontextu informační vědy, ÚISK FF UK*. [online] září 2001, [cit. 2011-05-22] Dostupná z WWW: http://www1.cuni.cz/~rihad/med/trendy.htm.

[16] *The Model-View-Controller (MVC) Design Pattern for PHP*. [online], [cit. 2011-04-20] Dostupná z WWW: http://www.tonymarston.net/php-mysql/model-view-controller.html.

[17] *UDP Packet Reflector Hacks*. [online], [cit. 2011-05-10] Dostupná z WWW: http://spirit.lboro.ac.uk/mug/mug.html.

Příloha A

Obsah přiloženého CD

Elektornická verze bakalářské práce:

- bp.pdf

Zdrojové kódy systému:

- rs/
  - php/
    - class/
      - cCache.php
      - cCalendar.php
      - cCtrl.php
      - cDB.php
      - cDBConf.php
      - cDBPSrc.php
      - cDBUser.php
      - cDBRelConfUser.php
      - cPage.php
      - cPageDet.php
      - cPageDetConf.php
      - cPageDetPSrc.php
      - cPageDetUser.php
      - cPageList.php
      - cPageListConf.php
      - cPageListPSrc.php
      - cPageListUser.php
      - cSession.ba.php
      - cSession.php
      - cTag.php
      - c_tblHeader.php
    - ctrlConf.php
    - ctrlPSrc.php

        - ctrlUser.php
    - sql/
        - rsCreateDatabase.sql
        - rsInsertUsers.sql
        - rsInsertDataUsers.sql
        - rsInsertDataPSrc.sql
    - inc/
        - rsConst.inc
        - rsDBSupport.inc
        - rsError.inc
        - support.inc
    - css/
        - rs_def.css
    - index.html

Doplňující dokumenty:

- doc/
    - prototypy_obrazovek_users.pdf
    - prototypy_obrazovek_admin.pdf
    - diagram_databaze.jpg
    - apache_enviroment.jpg

Příloha B

Zavedení systému

Aplikace byla vyvíjena v operačním systému Windows Vista Home Premium. Bylo použito PHP verze 5, Apache verze 2.2 a MySQL verze 5. Složky php, sql, inc a css byly uloženy v adresáři odpovídající konstantě DOCUMENT_ROOT serveru Apache. Podrobné nastavení serveru Appache je ke shlédnutí na přiloženém CD ve složce doc jako soubor apache_enviroment.jpg.

Při přenosu souborů je důležité dodržet strukturu adrsářů. Pro tvorbu a naplnění databáze ukázkovými daty slouží souvbory v adresáři rs/sql. rsCreateDatabase.sql tvoří strukturu aplikace, rsInsertUsers.sql nastavuje parametry pro spojení s databází. Ostatní soubory umožňují naplnit databázi daty. Při plnění databáze ukázkovými daty musí být prvně spuštěn skript rsInsertDataUsers.sql.

Systém je spouštěn přes soubor index.html. Pomocí nějž jsou zadána data o uživateli pro simulaci autentizační službu.