

PŘÍRODOVĚDECKÁ FAKULTA UNIVERZITY PALACKÉHO
KATEDRA INFORMATIKY

BAKALÁŘSKÁ PRÁCE

Informační systém se sdílenou znalostní bází



2013

Martin Javorský

Anotace

Cílem této bakalářské práce je vytvoření systému znalostní báze, jež umožňuje podrobnou kontrolu přístupu k jeho obsahu. Předpokládaným uživatelem je menší firma, jež pomocí systému spravuje a sdílí vnitrofiremní postupy a informace a to jak mezi jednotlivými zaměstnanci a odděleními, tak i směrem ke svým klientům.

Rád bych na tomto místě poděkoval vedoucímu práce PhDr. Juraji Mackovi za konzultace a cenné připomínky. Dále bych chtěl poděkovat rodině a přátelům za pomoc při testování systému a podporu při tvorbě práce.

Obsah

Seznam obrázků

Seznam tabulek

1. Úvod

Tato práce si bere za úkol vytvoření sdílené znalostní báze. Inspirací mi byla situace z vlastní zkušenosti, kdy mnohdy dochází k tomu, že firemní postupy a návody uchovávají zaměstnanci na svých pracovních stanicích, místo v centralizované podobě na serveru. Toto chování pak vede k vytváření duplicitních postupů, případně jsou postupy či návody neaktuální a zmatené.

Aplikace umožňuje sdílení mezi jednotlivými uživateli a skupinami i publikaci informací nepřihlášeným uživatelům. Vhodná je tedy pro nasazení jak v rámci vnitrofiremních struktur, tak i jako prostředek pro komunikaci oddělení zákaznické podpory s klienty. Díky aplikaci mají uživatelé snadný přístup k informacím. V případě klientů zákaznické podpory pak odpadá zatížení pracovníků firmy, již nemusí opakovaně řešit otázky na podobné dotazy, které si uživatelé mohou dohledat prostřednictvím webové aplikace.

2. Uživatelská dokumentace

V rámci uživatelské dokumentace jsou popsány jednotlivé části informačního systému. Jejich vzájemná interakce a pracovní scénáře. Pozornost je věnována především systému oprávnění, který hraje v efektivním používání aplikace důležitou roli.

2.1. Články a kategorie

Informační systém obsahuje soubor informací - články, návody, postupy atd., které vytvářejí a spravují příslušní uživatelé. Tyto informace jsou sdruženy do logických celků - kategorií.

Články tvoří jádro celého systému. Každý článek má svého autora a skupinu, kam patří. Při vytvoření nového článku je jako autor článku nastaven uživatel, jenž článek vytvořil a jako skupina článku, skupina do které daný uživatel patřil. Tyto údaje jsou později měnitelné uživatelem s patřičnými oprávněními.

Přístup uživatelů k článkům a kategoriím je definován systémem oprávnění (viz. kapitola ??). Kromě textu, jenž tvoří obsah článku, může článek obsahovat i přiložené soubory. Tyto přílohy mohou být libovolného formátu a ukládají se přímo na serveru. Soubory je možné nahrát k článku dodatečně nebo rovnou při jeho vytvoření. Postup vytvoření článku je blíže popsán v kapitole ?. Po zobrazení daného článku je možné uložit jeho obsah do .pdf souboru, případně vygenerovat ZIP archiv, který obsahuje článek v podobě HTML¹ souboru i všechny jeho přílohy.

Kategorie pak zastávají roli pomyslných adresářů - obsahují jak články, tak i další podkategorie. Jejich primárním účelem je seskupovat články a podkategorie do tématických okruhů a tím zvyšovat přehlednost celého systému.

2.2. Uživatelské role a skupiny

V rámci aplikace existuje několik typů uživatelských rolí (viz. tabulka ??). Samotní uživatelé jsou členěni do skupin.

Každá skupina má svého správce a může vlastnit články a kategorie. Jeden uživatel je současně členem pouze jedné skupiny. Pokud není stanoveno jinak, dědí uživatel oprávnění ze skupiny jejímž je členem.

¹HyperText Markup Language

akce	Správce systému	Správce skupiny	Uživatel	Nepřihlášený uživatel
Vytvořit skupinu	ano	ne	ne	ne
Smazat skupinu	ano	ne	ne	ne
Změnit skupinu	ano	ne	ne	ne
Vytvořit kategorii	ano	ano	ano ¹	ne
Smazat kategorii	ano	ano	ano	ne
Změnit kategorii	ano	ano	ano	ne
Vytvořit uživatele	ano	ano	ne	ne
Smazat skupinu	ano	ano	ne	ne
Změnit uživatele	ano	ano	ano ²	ne
Vytvořit článek	ano	ano	ano	ne
Smazat článek	ano	ano	ano	ne
Změnit článek	ano	ano	ano	ne

Tabulka 1. Základní uživatelské role a oprávnění

2.2.1. Správce aplikace

Tento uživatel má možnost spravovat celý systém. Může měnit a přidávat uživatelské účty, skupiny, články a kategorie. Může také libovolně měnit jejich oprávnění. Je to jediný uživatel, který má možnost editovat nastavení skupin a určovat jejich administrátory.

2.2.2. Správce skupiny

Jedná se o uživatele spravujícího skupinu. Jeden uživatel může být správcem více skupin. Tyto skupiny mu přiděluje správce aplikace. Do skupin, kterých je správcem, může přidávat nové uživatele. Tyto uživatele může vytvářet, případně vybírat z uživatelů aktuálně nepatřících do žádné skupiny. Uživatele ve svých skupinách může také editovat. Má možnost měnit články a kategorie skupiny, které je správcem. Spravuje oprávnění ke všem článkům, kategoriím a uživatelům v rámci jemu přidělených skupin.

2.2.3. Uživatel

Uživatel má možnost se přihlásit do systému a tím zobrazit články, které mu jiní uživatelé zpřístupní. Může také vkládat nové články a podkategorie do kategorií, k nimž má oprávnění. Nad vlastněnými články a kategoriemi má plnou kontrolu.

¹podkategorii kategorie kterou vlastní

²sebe samého

2.2.4. Nepřihlášený Uživatel

Jedná se o případ, kdy k informačnímu systému přistupuje nepřihlášený uživatel. V takovém případě se návštěvníkovi zobrazuje pouze obsah, jenž je určen jako veřejný.

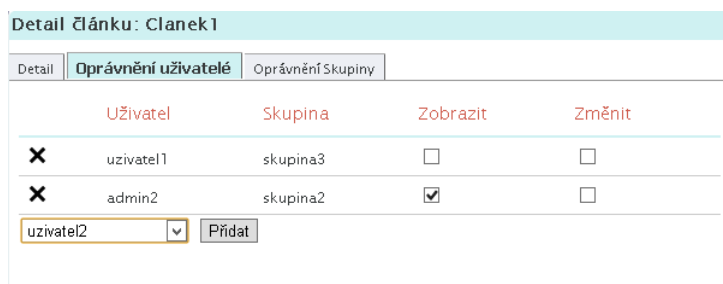
2.3. Systém oprávnění

Pro interakci s články a kategoriemi (dále uváděných souhrnně jako objekty) existují dva typy oprávnění. Oprávnění definující možnost zobrazení daného objektu a oprávnění určující zda je daný objekt měnitelný.

Tato oprávnění se mohou uplatňovat na článek i kategorii a mohou se vztahovat jak na skupinu, tak i na konkrétního uživatele. Tímto způsobem je možné skupině přidělit určité oprávnění k danému objektu a při tom explicitně určit konkrétního uživatele z téže skupiny, jemuž se stejné oprávnění zakáže, případně naopak. Oprávnění týkající se konkrétního uživatele převládají nad oprávněními pro skupinu.

2.3.1. Zobrazit

Držitel tohoto oprávnění má povoleno zobrazit obsah daného objektu. Pokud toto oprávnění není zatrženo, má držitel zakázáno zobrazení (viz. ??). Pokud právo není definováno, přebírá uživatel nastavení ze své skupiny.



Detail článku: Článek 1				
Detail		Oprávnění uživatelé		Oprávnění Skupiny
	Uživatel	Skupina	Zobrazit	Změnit
✘	uzivatel1	skupina3	<input type="checkbox"/>	<input type="checkbox"/>
✘	admin2	skupina2	<input checked="" type="checkbox"/>	<input type="checkbox"/>
	<input type="text" value="uzivatel2"/>			<input type="button" value="Přidat"/>

Obrázek 1. Oprávnění uživatele

2.3.2. Změnit

Pokud se toto oprávnění vztahuje ke kategorii, má jeho držitel právo přidávat do kategorie nové články a podkategorie. V případě, že je oprávnění nastaveno pro článek, umožňuje držiteli měnit obsah článku, jako je text, přílohy atd.

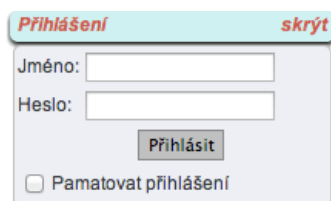
Upravovat oprávnění nastavená na daném objektu mohou jen správce systému, správce skupiny vlastníci objekt a uživatel vlastníci objekt.

2.4. Příklady použití

2.4.1. Přihlášení a odhlášení

Pokud k aplikaci přistupujeme bez předešlého přihlášení, tak se po načtení stránek internetovým prohlížečem zobrazí pouze veřejně dostupný obsah.

Pro zobrazení neveřejného obsahu je nutné provést přihlášení do systému. V pravé liště (viz. obrázek ??) vyplníme přihlašovací údaje a klikneme na tlačítko přihlásit. Je možné označit volbu zapamatování přihlašovacích údajů. Touto volbou se do počítače uloží přístupové údaje. Při opětovném přihlášení pak odpadá nutnost zadávání přístupových údajů, ty jsou již předvyplněné. Po úspěšném přihlášení se pak v přihlašovacím panelu zobrazí informace o přihlášeném uživateli a tlačítko „Nastavení“.



Obrázek 2. Přihlášení uživatele





Odhlášení provedeme na stejném místě jako přihlášení. U odhlášení je možnost "Zapomenout přihlášení". Jejím označením odstraníme případně lokálně uložené údaje, které jsme nechali zapamatovat při předešlém přihlášení.

2.4.2. Nastavení

Tlačítko pro přístup do sekce „Nastavení“ se zobrazí jen přihlášenému uživateli. Zobrazí se ve stejném panelu, v jakém se zadávali přihlašovací údaje. Tato část aplikace slouží pro správu celého systému.

Jednotlivé položky menu nastavení obsahují odkazy pro editaci uživatelů, skupin, článků a kategorií. Položky se zobrazují uživatelům dle úrovně jejich oprávnění. Po rozkliknutí položky v menu se zobrazí výpis prvků dané kategorie. Například, při rozkliknutí položky „Nastavení článků“ se zobrazí výpis všech článků které daný uživatel vlastní (viz. obrázek ??). Uživatel tak může manipulovat jen s objekty, k nimž má přístup. Uskutečněné změny se provádí na serveru asynchronně prostřednictvím technologie AJAX² (viz. kapitola ??). To vede k větší uživatelské přívětivosti a plynulosti práce, není totiž nutné při každé změně provádět opětovné načtení celé stránky.

²Asynchronous JavaScript and XML

	Název	Kategorie	Autor	Vytvořeno	Skupina	Veřejné
 	Clanek20	Kategorie3	uzivatel1	13. 5. 2013 14:22:24	skupina3	<input type="checkbox"/>
 	Clanek21	Kategorie3	uzivatel1	13. 5. 2013 14:22:39	skupina3	<input type="checkbox"/>

Obrázek 3. Výpis článků

2.4.3. Zobrazení detailu

Z celkového výpisu lze pak velmi jednoduše zobrazit detail dané položky. Kliknutím na ikonu detailu, jež se vždy nachází zcela vlevo na řádku náležící danému prvku, se zobrazí vyskakovací okno (viz. obrázek ??) s bližšími informacemi. Charakter informací se liší dle objektu, ke kterému se vztahuje. Zpravidla ale obsahuje detailní informace a výpis oprávnění, která má daný prvek přidělena. Z jednoho vyskakovacího okna lze stejným způsobem otevřít detail dalšího objektu a otevřít tak jeho detailní výpis. Pro zachování přehlednosti a jednoduchosti systému však lze mít současně otevřeno pouze jedno takové vyskakovací okno.

Detail uživatele: uziv_jmeno1 uziv_prijmeni1 (uzivatel1)

Detail	Články uživatele	Kategorie uživatele	Články - oprávnění	Kategorie - oprávnění
Přihlašovací Jméno:	<input type="text" value="uzivatel1"/>			
Heslo:	<input type="text" value="Heslo123456"/>			
Křestní Jméno:	<input type="text" value="uziv_jmeno1"/>			
Příjmení:	<input type="text" value="uziv_prijmeni1"/>			
E-mail:	<input type="text" value="ccc@aa.as"/>			
Přihlášení povoleno:	<input checked="" type="checkbox"/>			
Skupina:	<input type="text" value="skupina3"/> ▼			
<input type="button" value="Změnit"/>				

Obrázek 4. Detail uživatele

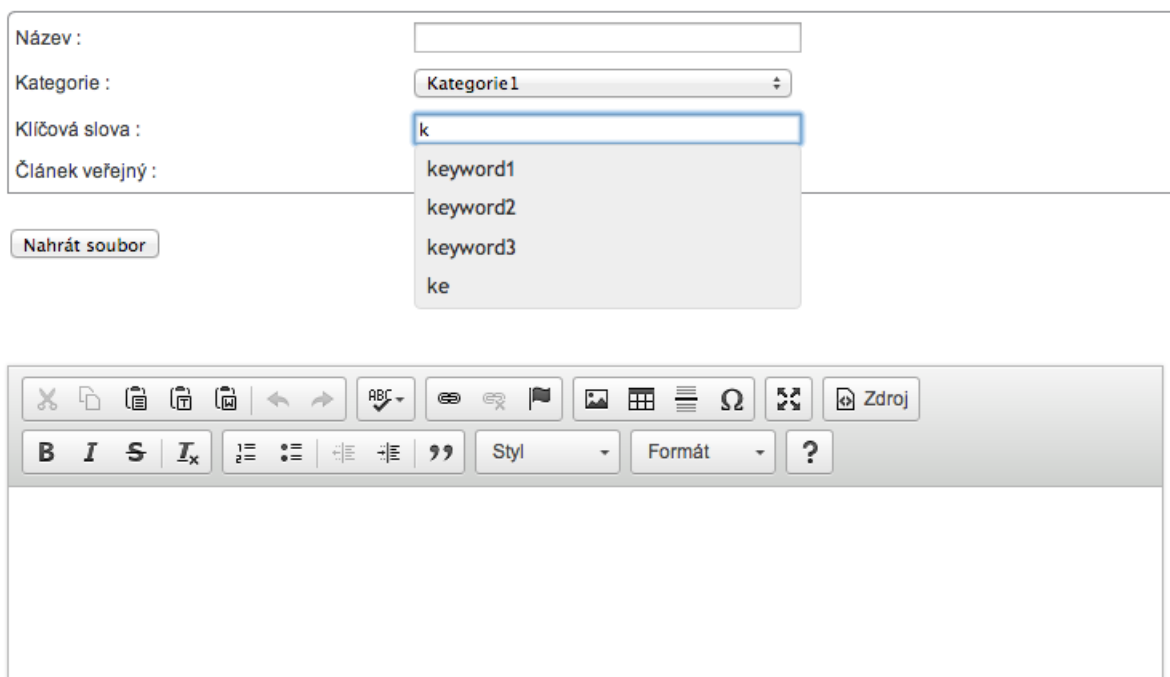
2.4.4. Vytvoření článku

Pro vytvoření nového článku musí být uživatel přihlášen do systému (viz. kapitola ??) . Následně po kliknutí na ikonu znázorňující vytvoření nového článku, jež se nachází nad a pod výpisem článků v sekci „Nastavení“, dojde k přesměrování na stránku určenou k editaci nového článku (viz. obrázek ??).

Zde je nutné zadat název. Ze seznamu dostupných kategorií vybrat jednu do které bude článek přidán.

Dále je vhodné doplnit klíčová slova charakterizující článek. Při jejich zadávání jsou již existující klíčová slova našeptávána z databáze klíčových slov. Tato databáze se postupně rozrůstá, klíčová slova, která jsou u článku použita a v databázi neexistují, jsou do ní automaticky ukládána.

Jak již bylo zmíněno dříve, k článku je možné přiložit dodatečné soubory. Dialogové okno pro přidání souborů zobrazíme kliknutím na tlačítko „Nahrát soubor“. Soubory nahrajeme prostým přetažením do prostoru pro ně vyhrazeného nebo pomocí tlačítka „Vyberte“ zvolíme patřičné soubory na lokálním disku. Jestliže chceme nahraný obrázek použít v rámci textu článku, přidáme jej do textu kliknutím na ikonu znázorňující „plus“. Ta se zobrazuje v seznamu přiložených souborů a vloží odkaz na obrázek do textu textového editoru. Změnit vlastnosti takto přidaného obrázku můžeme následně kliknutím pravým tlačítkem na samotný obrázek v rámci textového editoru. Článek uložíme tlačítkem „uložit“.



The image shows a web form for creating a new article. It has four main input fields: 'Název' (Name), 'Kategorie' (Category), 'Klíčová slova' (Keywords), and 'Článek veřejný' (Public article). A 'Nahrát soubor' (Upload file) button is located below the 'Klíčová slova' field. A dropdown menu is open under 'Klíčová slova', displaying a list of keywords: 'k', 'keyword1', 'keyword2', 'keyword3', and 'ke'. Below the form is a rich text editor toolbar with various icons for text formatting and editing.

Obrázek 5. Nový článek

2.4.5. Vyhledávání článků

V horní liště stránek se nachází dvě textová pole pro zadání hledaného textu.

Obě pole obsahují našeptávač, jenž usnadňuje uživateli zadávání často hledaných výrazů. Pole s klíčovými slovy automaticky při psaní doplňuje výrazy, které jsou obsaženy v databázi klíčových slov. Uživatel má tedy okamžitou vizuální kontrolu nad možnými hledanými parametry. Je možné zadat více klíčových slov najednou.

V druhém textovém poli se vyhledává dle slov, která jsou obsažena v textu v některém z článků. I v tomto případě je zabudovaná nápověda zobrazující nejčastěji hledané výrazy.

3. Programátorská dokumentace

Vývoj aplikace jsem prováděl ve vývojovém prostředí Microsoft Visual Studio 10. Jedná se o velmi efektivní nástroj pro rychlý a kvalitní vývoj aplikací. Při programování projektu jsem velmi ocenil propracovaný debugger i integrovaný webový server „Visual Studio Development Server“. Tento webový server je výchozím serverem pro debugger. Je nutné brát na zřetel, že běží s jiným bezpečnostním nastavením než případný klasický webový server (IIS ³ server) a nemusí odhalit chyby, které se vyskytnou po nasazení aplikace do ostrého provozu na klasický webový server.

Pro tvorbu webové aplikace je použita technologie ASP.NET, jež je součástí .NET Frameworku. Tento Framework vyvinula společnost Microsoft v roce 2002. Obsahuje mimo jiné jednotné prostředí pro běh kódu (CLR ⁴). Ten umožňuje běh knihoven v rámci frameworku bez ohledu na programovací jazyk, v němž byl napsán. Nejpoužívanějšími jazyky v rámci .NET Frameworku jsou C # a Visual Basic, podporované jsou ale i mnohé další. Webové aplikace používající .NET Framework jsou před nasazením na server předkompilované. Kompilace zahrnuje výhody jako lepší rychlost načítání, než stránky využívající běžné skriptovací jazyky a vyšší stabilitu. Kód je během kompilace kontrolován kompilátorem a případné problémy jsou odhaleny před nasazením na server.

Jako programovací jazyk jsem zvolil C #. Jedná se o moderní objektově orientovaný jazyk podobný jazyku C či Java, jež v roce 2000 vytvořila společnost Microsoft. Možnosti použití jsou velmi široké, počínaje webovými aplikacemi, formulářovými aplikacemi pro Windows až po různé databázové programy.

Uživatelská data, se kterými aplikace pracuje, jsou uložena v rámci relační databáze Microsoft SQL Serveru 2008. Tento systém pro správu relační databáze je vytvořený společností Microsoft. Mezi jeho přednosti patří integrovaná podpora XML ⁵ jazyka, snadná škálovatelnost a bezpečnost. Obsahuje také řadu nástrojů pro usnadnění správy serveru a optimalizaci dotazů a struktury dat.

3.1. Architektura aplikace

Pro potřeby aplikace bylo nutné zajistit, aby jednotlivé stránky měly stejný vzhled a některé společné části se stejnou funkcionalitou. Pro splnění těchto požadavků je v rámci ASP.NET, od verze ASP.NET 2.0, možné použít tzv. „Master Page“.

³Internet Information Services

⁴Common Language Runtime

⁵Extensible Markup Language

Jedná se o jednu hlavní stránku, jež obsahuje společné prvky a vzhled pro celou aplikaci, lépe řečeno pro ostatní stránky označeny jako tzv. „Content Page“, volně přeloženo jako „stránky s obsahem“. Tyto stránky pak jen doplňují stránku hlavní a to na místech k tomu určených. Takový prostor v rámci hlavní strany vymezuje ovládací prvek s názvem „ContentPlace“. Dalo by se tedy říci, že hlavní stránka funguje jako šablona. Tento model velice zjednodušuje vývoj a údržbu aplikací. Jejich stránky pak mají jednotné grafické zveřejnění a liší se jen obsahem na předem definovaných místech.

Během návrhu projektu jsem se snažil dodržet principy třívrstvé architektury a oddělovat od sebe prezentční, aplikační a datovou vrstvu.

- **Prezentční vrstva** je obsažena v rámci samotných .aspx souborů a prezentována HTML kódem obsaženým především v „Master Page“. Grafické styly pro HTML kód jsou uloženy v CSS souborech.
- **Aplikační vrstva** prezentuje samostatnou aplikační logiku. V rámci tříd a jejich metod umožňuje bezpečnou výměnu dat mezi zbylými dvěma vrstvami.??
- **Datová vrstva** poskytuje prostor pro ukládání uživatelských dat a základní datovou integritu (viz. kapitola ??).

3.2. Externí knihovny

3.2.1. Ajax Control Toolkit

ASP.NET AJAX Control Toolkit je projekt vytvořený nad Microsoft ASP.NET AJAX frameworkem. Je volně dostupný pod BSD [?] licenci. Tato knihovna obsahuje více než 40 ovládacích prvků, které velkou měrou usnadňují tvorbu interaktivních ASP.NET aplikací a rozšiřují standardní nabídku ovládacích prvků.

ASP.NET AJAX je implementace technologie Ajax od firmy Microsoft, jež je zaměřena na využití v rámci ASP.NET. Nabízí mimo jiné technologie na straně serveru umožňující odpovídat na částečná zpětná volání a obslužný kód umožňující ASP.NET komunikovat prostřednictvím JSON⁶ s kódem na straně klienta. Tím je dosažena vysoká interaktivita stránek, které pro zobrazení nových informací nebo aktualizaci stávajících nepotřebují plné zpětné volání.

⁶Javascript Object Notation

3.2.2. iTextSharp

iTextSharp je open-source odnož iText knihovny psaná v jazyku C#. Slouží mimo jiné pro vytváření nových PDF souborů, konverzi XML souborů do PDF, vytváření interaktivních PDF formulářů. Tuto knihovnu používám pro exportování článků do PDF souborů.

3.2.3. DotNetZip

Knihovna pro manipulaci se .zip soubory. Lze ji použít pod jakýmkoliv .NET programovacím jazykem. V rámci aplikace slouží ke generování .zip archivů. Její použití je velmi snadné, níže je ukázka kódu, kdy je předem určený adresář zkomprimován a přímo vrácen klientovi jako HTTP odpověď.

```
var zip = new ZipFile();
zip.AddDirectory(directoryName);
HttpContext.Current.Response.ContentType = "application/zip";
string headerValue = "filename=" + directoryName + ".zip";
HttpContext.Current.Response.AddHeader("Content-Disposition", headerValue);
zip.Save(HttpContext.Current.Response.OutputStream);
```

3.2.4. CKEditor

Pro vkládání a editaci HTML textu v rámci článků jsem zvolil CKEditor. Jedná se o WYSIWYG editor distribuovaný pod GPL [?]licencí. Editor je vyvíjen přes deset let a obsahuje verzi, jež je integrovatelná do ASP.NET jako ovládací prvek.

Pro potřeby bakalářské práce bylo nutné umožnit vkládání obrázků do textu článku. Tyto jsou nahrány na serveru v adresáři patřící danému článku. CKEditor tuto funkcionalitu přímo nepodporuje, dá se jí dosáhnout rošířením editoru implementací souborového manažera, např. CKFinder, jenž ovšem není dostupný pod volně šiřitelnou licencí a jeho použití je zpoplatněno.

Situaci jsem vyřešil přidáním tlačítka pro vložení odkazu, které se vyskytuje u každého přiloženého souboru. Toto tlačítko následně vkládá HTML značku odkazující na přiložený obrázek, jenž byl před tím na server nahrán při vytváření či editaci článku. Tímto se tedy podařilo dosáhnout velmi podobné funkcionality, jež je navíc pro potřeby aplikace plně dostavující.

3.2.5. jQuery

jQuery je framework, jenž se snaží ulehčit práci s JavaScriptem, čehož se snaží dosáhnout stručností a přehledností. Umožňuje snadno vyhledávat

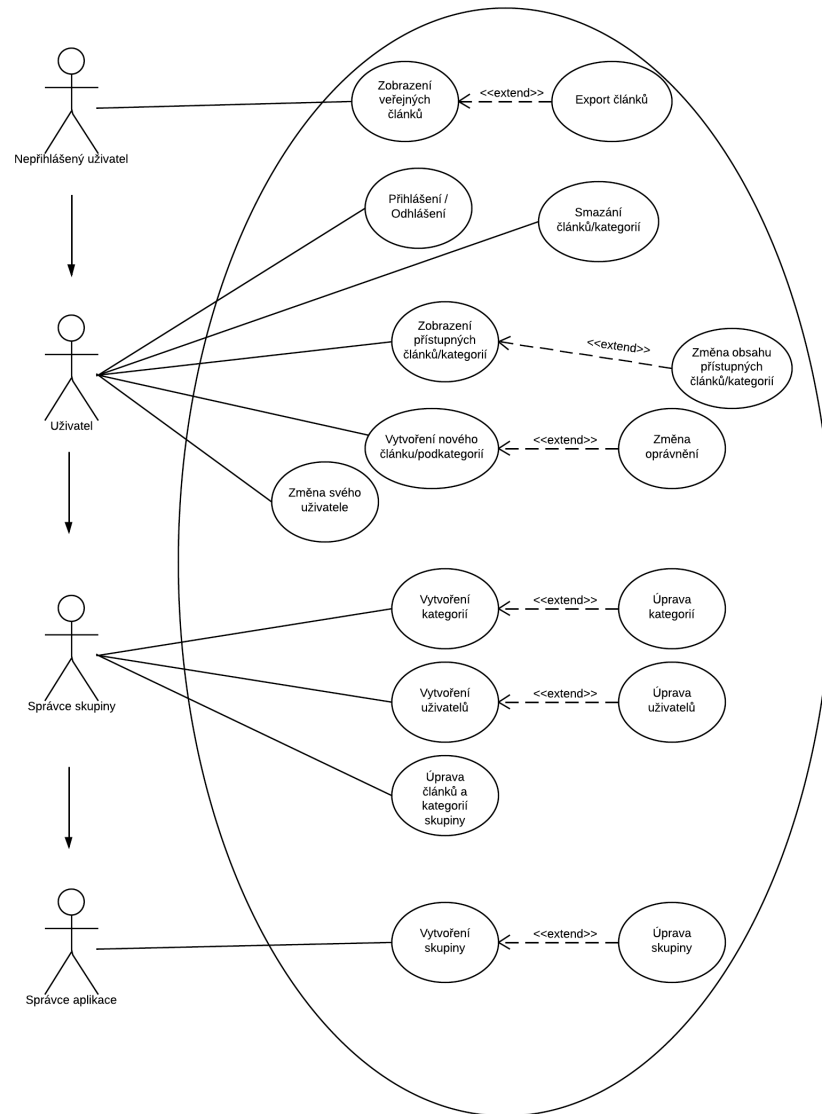
CSS elementy v DOMu⁷, podporuje také řetězení příkazů. Obsahuje mnoho zabudovaných efektů a podporu AJAX, které v rámci projektu používám.

Příklad zřetězení příkazů:

```
\$('.myClassFadeout').delay(3000).fadeOut(3000);
```

⁷Document Objekt Model

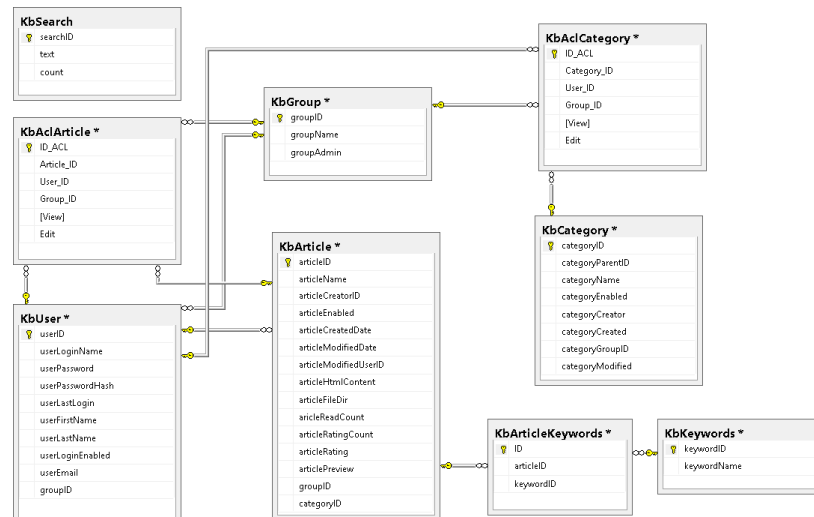
3.3. USE CASE



3.4. Databáze

Relační schéma databáze je zobrazeno na obrázku ??

Obrázek 6. Datový model



3.4.1. Tabulka kbUser

Tabulka (viz. obrázek ??) obsahuje informace o uživateli. Primárním klíčem je atribut „userID“. Atribut „groupID“ je cizí klíč odkazující na uživatelovu skupinu.

	Column Name	Data Type	Allow Nulls
PK	userID	int	<input type="checkbox"/>
	userLoginName	varchar(50)	<input type="checkbox"/>
	userPassword	varchar(50)	<input type="checkbox"/>
	userPasswordHash	varchar(50)	<input type="checkbox"/>
	userLastLogin	smalldatetime	<input checked="" type="checkbox"/>
	userFirstName	varchar(50)	<input checked="" type="checkbox"/>
	userLastName	varchar(50)	<input checked="" type="checkbox"/>
	userLoginEnabled	bit	<input type="checkbox"/>
	userEmail	varchar(50)	<input checked="" type="checkbox"/>
	groupID	int	<input type="checkbox"/>

Obrázek 7. Tabulka kbUser

3.4.2. Tabulka kbGroup

Tabulka (viz. obrázek ??) s informacemi o skupině. „categoryID“ odkazuje na kategorii kam daný článek náleží.

	Column Name	Data Type	Allow Nulls
🔑	groupID	int	<input type="checkbox"/>
	groupName	nvarchar(50)	<input type="checkbox"/>
	groupAdmin	int	<input type="checkbox"/>

Obrázek 8. Tabulka kbGroup

3.4.3. Tabulka kbArticle

Tabulka (viz. obrázek ??) obsahuje informace o článcích. Primárním klíčem tabulky je atribut „articleID“. Cizí klíče „groupID“, „articleCreatorID“ určují skupinu a uživatele vlastníci článek. „categoryID“ odkazuje na kategorii kam daný článek náleží.

	Column Name	Data Type	Allow Nulls
🔑	articleID	int	<input type="checkbox"/>
	articleName	nvarchar(100)	<input checked="" type="checkbox"/>
	articleCreatorID	int	<input type="checkbox"/>
	articleEnabled	bit	<input checked="" type="checkbox"/>
	articleCreatedDate	date	<input checked="" type="checkbox"/>
	articleModifiedDate	datetime	<input checked="" type="checkbox"/>
	articleModifiedUserID	int	<input checked="" type="checkbox"/>
	articleHtmlContent	nvarchar(MAX)	<input checked="" type="checkbox"/>
	articleFileDir	nvarchar(50)	<input checked="" type="checkbox"/>
	articleReadCount	int	<input checked="" type="checkbox"/>
	articleRatingCount	int	<input checked="" type="checkbox"/>
	articleRating	int	<input checked="" type="checkbox"/>
	articlePreview	nvarchar(MAX)	<input checked="" type="checkbox"/>
	groupID	int	<input type="checkbox"/>
	categoryID	int	<input type="checkbox"/>

Obrázek 9. Tabulka kbArticle

3.4.4. Tabulka kbCategory

Tabulka (viz. obrázek ??) obsahující informace o kategorii. Atribut „groupID“ je omezen referenční integritou.

	Column Name	Data Type	Allow Nulls
?	categoryID	int	<input type="checkbox"/>
	categoryParentID	int	<input checked="" type="checkbox"/>
	categoryName	nvarchar(100)	<input type="checkbox"/>
	categoryEnabled	bit	<input checked="" type="checkbox"/>
	categoryCreator	int	<input type="checkbox"/>
	categoryCreated	datetime	<input checked="" type="checkbox"/>
	categoryGroupID	int	<input type="checkbox"/>
	categoryModified	datetime	<input checked="" type="checkbox"/>

Obrázek 10. Tabulka kbCategory

3.4.5. Tabulka kbSearch

Tabulka (viz. obrázek ??) obsahující výrazy, které se v rámci aplikace vyhledávali. Ukládá se zde záznam o každém provedeném hledání.

	Column Name	Data Type	Allow Nulls
?	searchID	int	<input type="checkbox"/>
	text	varchar(50)	<input type="checkbox"/>
	count	int	<input type="checkbox"/>

Obrázek 11. Tabulka kbSearch

3.4.6. Tabulka kbAclArticle

Jedná se o vazební tabulku (viz. obrázek ??) uchovávající informace o přidělených oprávněních k jednotlivým článkům. Cizí klíče této tabulky jsou „Article_ID“, „Group_ID“, „User_ID“.

	Column Name	Data Type	Allow Nulls
	ID_ACL	int	<input type="checkbox"/>
	Article_ID	int	<input type="checkbox"/>
	User_ID	int	<input type="checkbox"/>
	Group_ID	int	<input type="checkbox"/>
	[View]	bit	<input checked="" type="checkbox"/>
	Edit	bit	<input checked="" type="checkbox"/>

Obrázek 12. Tabulka kbAclArticle

3.4.7. Tabulka kbKeywords

V této tabulce (viz. obrázek ??) jsou uložena jednotlivá klíčová slova pro články.

	Column Name	Data Type	Allow Nulls
	keywordID	int	<input type="checkbox"/>
	keywordName	varchar(50)	<input type="checkbox"/>

Obrázek 13. Tabulka kbKeywords

3.4.8. Tabulka kbArticleKeywords

Vazební tabulka (viz. obrázek ??) spojující články a klíčová slova.

	Column Name	Data Type	Allow Nulls
	ID	int	<input type="checkbox"/>
	articleID	int	<input type="checkbox"/>
	keywordID	int	<input type="checkbox"/>

Obrázek 14. Tabulka kbArticleKeywords

3.4.9. Tabulka kbAclCategory

Jedná se o vazební tabulku (viz. obrázek ??) uchovávající informace o přidělených oprávněních k jednotlivým kategoriím. Cizí klíče této tabulky jsou „Article_ID“, „Group_ID“, „User_ID“.

	Column Name	Data Type	Allow Nulls
	ID_ACL	int	<input type="checkbox"/>
	Category_ID	int	<input type="checkbox"/>
	User_ID	int	<input type="checkbox"/>
	Group_ID	int	<input type="checkbox"/>
	[View]	bit	<input checked="" type="checkbox"/>
	Edit	bit	<input checked="" type="checkbox"/>

Obrázek 15. Tabulka kbAclCategory

3.5. Popis tříd

3.5.1. kbAccess

Třída kontrolující přístup k jednotlivým objektům systému.

- `articleAccess(int articleID)` Metoda vracející informaci, zda aktuální uživatel přistupující k článku má možnost jeho zobrazení. Jako parametr slouží identifikátor článku, což je současně i jeho databázový primární klíč.
- `articleEditable(int articleID)` Obdoba předešlé metody, v tomto případě se ale kontroluje, zda-li je článek editovatelný.

3.5.2. kbArticle

Třída reprezentující článek v rámci systému. Mezi její nejzajímavější metody patří:

- `HTMLToPdf(int articleID)` Metoda převádějící HTML kód stránky do PDF dokumentu. K čemuž se využívá externí knihovna (viz. kapitola??).
- `saveAsHtml(int articleID)` Tato metoda generuje HTML dokument z kódu obsaženého v databázi. Následně společně s příloženými soubory vytváří ZIP archiv a ten vrací klientovi (viz kapitola??).
- `ShortenString(string stringToShorten, int newLength)` Metoda zkracující text článku, který se pak používá v rámci náhledu.

3.5.3. kbArticleFiles

Třída pracující se soubory na disku, její metody se starají o vytváření a mazání dat. Vracejí také fyzickou a virtuální cestu k příloženým souborům.

- `deleteFile(string absolutePath)` Smazání konkrétního souboru.
- `articleFilesDataTable(int articleID)` Přetěžovaná metoda, která vrací „DataTable“ datovou strukturu, jež obsahuje informace o příložených souborech. Tato vrácená datová tabulka je následně použita jako zdroj dat pro zobrazení příložených souborů.

3.5.4. kbCategory

Třída reprezentující kategorii. Důležitou metodou je například metoda:

- `dataTableEditablesCategories()` Tato metoda vrací datovou tabulku, kde jsou obsaženy všechny kategorie, které má přihlášený uživatel právo měnit.

3.5.5. kbGroup

Třída reprezentující skupinu.

3.5.6. kbHtmlUtils

Třída obsahující metody pro úpravu HTML kódu.

- `extractTextFromHTML(string body)` Metoda vrací čistý text z HTML kódu bez HTML značek.

3.5.7. kbSearch

Třída obsahující metody pro hledání v článcích dle klíčových slov a dle obsahu.

3.5.8. kbSession

Třída rozšiřující standartní Session třídu o proměnné používající se v rámci aplikace.

3.5.9. kbUser

Třída reprezentující uživatele systému.

3.6. Instalace

Aplikace pro svůj běh vyžaduje IIS server s podporou .NET Frameworku v minimální verzi 4.0 a MS SQL Server. Jelikož se jedná o webovou aplikaci, je její instalace poměrně snadná.

- **Webová aplikace**

1. V rámci administrace IIS vytvoříme nový projekt (webovou stránku), jehož domovský adresář později nastavíme na adresář „webroot“, jenž je obsažen v komprimovaném souboru KB.zip.
2. Z instalačního média rozbalíme soubor KB.zip. Tento soubor již obsahuje kompletní adresářovou strukturu včetně všech potřebných knihoven.

3. Uživateli pod nímž běží proces IIS serveru, standardně uživatel „IIS_IUSRS“, povolíme zapisovat do adresáře „data“, ten se nachází v kořenu rozbalené adresářové struktury.
4. Povolíme IIS serveru zapisovat do „tmp“ adresáře, který se nachází v „wwwroot“ adresáři.

- **Databáze**

1. Do SQL Serveru naimportujeme přiloženou databázi „databaze.mdf“. To provedeme otevřením „Microsoft SQL Server Management Studia“. V rozbalovacím menu znázorňujícím strukturu serveru klikneme pravým tlačítkem myši na položku databází a zvolíme možnost připojit („attach“) novou databázi, následně se zobrazí průvodce pro připojení, kterého dokončíme.
2. Vytvoříme nového databazového uživatele a nastavíme jej jako vlastníka nové databáze.
3. Upravíme soubor Web.config, který je konfiguračním souborem aplikace. Zde změníme hodnotu „ConnectionStringDatabase“ proměnné. Upravíme přihlašovací údaje do databáze tak, aby odpovídaly nově vytvořené databázi.

- **První spuštění** V tomto kroku by měla být aplikace přístupna na URL adrese, která se nastavila v rámci projektu IIS serveru. Základní přihlašovací jméno pro systémového správce je „admin“ a heslo „admin“.

3.7. Testování aplikace

Testování a vývoj aplikace jsem prováděl na virtuálním serveru zřízeném pro tento účel. Na tomto stroji je aktuálně aplikace zprovozněna a je zde i přístupná. Můžeme ji tedy otestovat bez nutnosti spuštění projektu v rámci Microsoft Visual Studia nebo jeho instalace na jiný webový server. Jako adresu pro přístup můžeme použít přímo IP adresu serveru <http://88.208.119.13>. V aplikaci jsou testovací data a testovací uživatelé (viz. tabulka ??).

Uživatelské jméno	Heslo	Skupina	Role
superuser	Heslo123456	X	Správce systému
admin1	Heslo123456	skupina1	Správce skupiny
admin2	Heslo123456	skupina2	Správce skupiny
uzivatel1	Heslo123456	skupina1	Uživatel
uzivatel2	Heslo123456	skupina2	Uživatel

Tabulka 2. Testovací přístupy

V rámci vývoje byla aplikace testována posledními verzemi nejvíce používaných prohlížečů. Poslední verze v době psaní této práce jsou:

- Google Chrome verze 26
- Mozilla Firefox verze 21
- Microsoft Internet Explorer 10

Na žádném z těchto prohlížečů nebylo zaznamenáno nestandardní chování nebo jiné chování vedoucí k pádu aplikace.

3.8. Bezpečnost

V rámci provozu aplikace bylo nutné vyřešit několik bezpečnostních otázek.

- **Přihlášení do systému**, metodiku přihlášení jsem vytvořil sám. Aplikace nepoužívá „Login“ funkcionality ASP.NET. Rozhodl jsem se pro vlastní metodiku především proto, abych získal praktické zkušenosti s programováním přihlašovacího mechanismu, a také abych měl v rámci aplikace maximální kontrolu nad přihlašovaním uživatelů.

Hesla uživatelů jsou šifrována pomocí MD5 algoritmu a ukládána do databáze. V aktuálním provedení jsou hesla ukládána i v prostém textu, aby bylo možné zobrazit v rámci administrace. Ukládání nešifrovaných hesel lze chápat jako bezpečnostní riziko, obecně není dobrou praxí hesla v nešifrovaném stavu ukládat.

- **Přístup k přílohám**, pro zamezení přístupu k přílohám přímo přes URL odkaz směřující na daný soubor, jsem přistoupil k řešení, kdy adresář se soubory umísťují mimo domovský adresář projektu, mimo tzv. „Document Root“. V rámci aplikace je pak vytvořena metoda, která kontroluje přístup k přílohám a určuje, zda je daný soubor zpřístupněn aplikací uživateli. V rámci ASP.NET má totiž aplikace přístup na jakákoliv data na serveru, pokud má od administrátora serveru potřebná oprávnění, ne jen do domovského adresáře stránek.
- **Validace vstupů** je prováděna jak na straně klienta, tak i na straně serveru. Na straně serveru se jedná se především o kontrolu, zda nedochází k duplicitě některých údajů jako například přihlašovacích jmen, názvů skupin apod. Pro validaci je použit „Custom Validator Control“, tedy ovládací prvek umožňující tvorbu vlastních kontrolních metod.

Závěr

Záměrem této bakalářské práce bylo vytvoření informačního systému se sdílenou znalostní bází, která umožňuje export dat do různých formátů a dokáže účinně spravovat přístupy k jednotlivým částem systému.

Velkou výhodou aplikace spatřuji v systému kontroly přístupu k informacím. Ten umožňuje prostřednictvím přehledné administrace efektivním způsobem spravovat jednotlivá uživatelská oprávnění. Využití tak nalezne především v prostředí, kdy k aplikaci přistupuje více uživatelů.

Dle mého názoru se mi podařilo úspěšně vytvořit aplikaci splňující zadání. Při budoucím vývoji aplikace by mělo velký význam nasazení různých analytických modulů, které by přinesly důležité informace o využití aplikace. Také systém článků je navržen tak, aby po menších úpravách umožnil zobrazování multi-mediálních příloh přímo na stránce článku, nikoliv jen jako stahovatelnou přílohu.

Conclusions

The aim of this thesis was to develop an information system with a shared knowledge base that allows you to export data into various formats and is able to effectively manage access to system components.

As a big advantage of application is access control to informations. It allows effectively manage individual user permissions through administration. Usage of application is especially in environments with multiple users accessses.

In my opinion, I was able to successfully create an application which meets specifications. In future developmnet is crucial to buil different analytical modules, which would provide important analytical informations about system usage. Also system of articles id designed in way, which after minor modifications, will be able to view mutlimedia attachments directly from page of article, not only as a downloadable attachment.

Reference

- [1] Nagel, Christian a kol. *C# Programujeme profesionálně*. Vydání první, Computer Press a.s., Brno, 2009, ISBN 978-80-251-2401-7.
- [2] Eller, Frank. *C# - začínáme programovat*. GRADA Publishing a.s., Praha, 2002, ISBN 80-247-0324-6
- [3] Hotek, Mike. *Microsoft SQL Server 2008 krok za krokem*. Computer Press a.s., Brno, 2009, ISBN 978-80-251-2466-6
- [4] Ajax Control Toolkit, dostupné z <http://ajaxcontroltoolkit.codeplex.com/>
- [5] iTextSharp, dostupné z <http://sourceforge.net/projects/itextsharp/>
- [6] DotNetZip, dostupné z <http://dotnetzip.codeplex.com/>
- [7] CKeditor, dostupné z <http://ckeditor.com/>
- [8] BSD licence, dostupné z <http://www.netbsd.org/about/redistribution.html>
- [9] GNU licence, dostupné z <http://www.gnu.org/licenses/gpl.html>

4. Obsah přiloženého CD

`bin/`

Kompletní adresářová struktura webové aplikace ZNALOSTNÍ BÁZE (v ZIP archivu) pro zkopírování na webový server. Adresář obsahuje i všechny potřebné knihovny pro bezproblémový provoz na webovém serveru.

`doc/`

Dokumentace práce ve formátu PDF, vytvořená dle závazného stylu KI PřF pro diplomové práce, včetně všech příloh, a všechny soubory nutné pro bezproblémové vygenerování PDF souboru dokumentace (v ZIP archivu), tj. zdrojový text dokumentace, vložené obrázky, apod.

`src/`

Kompletní zdrojové texty webové aplikace ZNALOSTNÍ BÁZE se všemi potřebnými (převzatými) zdrojovými texty, knihovnami a dalšími soubory pro bezproblémové vytvoření adresářové struktury pro zkopírování na webový server (v ZIP archivu).

`readme.txt`

Instrukce pro nasazení webové aplikace ZNALOSTNÍ BÁZE na webový server, včetně požadavků pro její provoz, a webová adresa, na které je aplikace nasazena pro testovací účely a pro účel obhajoby práce.

Navíc CD/DVD obsahuje:

`data/`

Ukázková a testovací data použitá v práci a pro potřeby obhajoby práce.

`install/`

Instalátory aplikací, knihoven a jiných souborů nutných webové aplikaci, které nejsou standardní součástí operačního systému.

U veškerých odjinud převzatých materiálů obsažených na CD/DVD jejich zahrnutí dovoluují podmínky pro jejich šíření nebo přiložený souhlas držitele copyrightu. Pro materiály, u kterých toto není splněno, je uveden jejich zdroj (webová adresa) v textu dokumentace práce nebo v souboru `readme.txt`.