

UNIVERZITA PARDUBICE

FAKULTA ELEKTROTECHNIKY A INFORMATIKY

IMPLEMENTACE REDAKČNÍHO SYSTÉMU V JAZYCE JAVA

DIPLOMOVÁ PRÁCE

2009

Bc. Petr Šejn

UNIVERZITA PARDUBICE

FAKULTA ELEKTROTECHNIKY A INFORMATIKY

IMPLEMENTACE REDAKČNÍHO SYSTÉMU V JAZYCE JAVA

DIPLOMOVÁ PRÁCE

AUTOR PRÁCE: Bc. Petr Šejn

VEDOUČÍ PRÁCE: Ing. Jana Holá, Ph.D

2009

UNIVERSITY OF PARDUBICE
INSTITUTE OF ELECTRICAL ENGINEERING
AND INFORMATICS

IMPLEMENTATION OF REDACTION SYSTEM IN JAVA
THESIS WORK

AUTHOR: Bc. Petr Šejn

SUPERVISOR: Ing. Jana Holá, Ph.D

2009

ZADÁNÍ DIPLOMOVÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Bc. Petr ŠEJN**

Studijní program: **N2646 Informační technologie**

Studijní obor: **Informační technologie**

Název tématu: **Implementace redakčního systému v jazyce JAVA**

Z á s a d y p r o v y p r a c o v á n í :

- Hlavním cílem diplomové práce je tvorba redakčního systému, určeného pro řízení firemního obsahu v rámci intranetu a extraktu firmy.
- Redakční systém bude implementován jako multiplatformní aplikace v jazyce JAVA s využitím na všech platformách operačních systémů.
- Základní požadavky na systém: modularita a otevřenost pro integraci s dalšími systémy, konkurenceschopnost na dle požadovaných standardů v tržním prostředí.
- Návrh systému bude proveden v modelovacím jazyce UML, po korektním návrhu bude aplikace naprogramována v jazyce JAVA za pomoci vývojového softwaru NetBeans.

rozsaň granckycn prací:

Rozsaň pracovní zprávy:

Forma zpracování diplomové práce: **tištěná/elektronická**

Seznam odborné literatury:

1. GÁLA, Libor, POUR, Jan, TOMAN, Prokop. Podniková informatika. 1. vyd. Praha : Grada, 2006. 482 s. ISBN 80-247-1278-4.
2. KERŔKOVSKÝ, Miloslav, DRDLA, Miloš. Strategické řízení firemních informací. 1. vyd. Praha : C.H. Beck, 2003. ISBN 80-7179-730-8.
3. JSP: JavaServer Pages - Podrobný původce, Bary Burd ISBN 80-7226-804-X
4. UML 2 a unifikovaný proces vývoje aplikací - Objektově orientovaná analýza a návrh prakticky, Jim Arlow, Ila Neustadt, ISBN: 978-80-251-1503-9
5. Java - Kuchařka programátora, Ian F. Darwin, ISBN: 80-251-0944-5
6. Head First Servlets and JSP, Bert Bates, Kathy Sierra, Bryan Basham, ISBN: 0596005407
7. Head First Design Patterns, Elisabeth Freeman, Eric Freeman, Bert Bates, Kathy Sierra

Vedoucí diplomové práce:

Ing. Jana Holá, Ph.D.
Katedra elektrotechniky

Datum zadání diplomové práce:

31. října 2008

Termín odevzdání diplomové práce:

22. května 2009



doc. Ing. Simeon Karamazov, Dr.

děkan



doc. Ing. Antonín Kavička, Ph.D.

vedoucí katedry

V Pardubicích dne 4. listopadu 2008

Prohlašuji:

Tuto práci jsem vypracoval samostatně. Veškeré literární prameny a informace, které jsem v práci využil, jsou uvedeny v seznamu použité literatury.

Byl jsem seznámen s tím, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorský zákon, zejména se skutečností, že Univerzita Pardubice má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle § 60 odst. 1 autorského zákona, a s tím, že pokud dojde k užití této práce mnou nebo bude poskytnuta licence o užití jinému subjektu, je Univerzita Pardubice oprávněna ode mne požadovat přiměřený příspěvek na úhradu nákladů, které na vytvoření díla vynaložila, a to podle okolností až do jejich skutečné výše.

Souhlasím s prezenčním zpřístupněním své práce v Univerzitní knihovně Univerzity Pardubice.

V Pardubicích dne 22. 05. 2009

Bc. Petr Šejn

ABSTRAKT

Tato práce je zaměřena na implementaci redakčního systému, jako systému pro správu webového obsahu, v jazyce JAVA. V teoretické části jsou definovány hlavní pojmy z oblasti redakčních systémů a detailně popsány hlavní stavební kameny a funkcionality těchto systémů. Dále je poukázáno na přední programovací jazyky využívané ke tvorbě těchto systémů a vývojové prostředí k tomu určených. Praktická část přehledně prezentuje jednotlivé kroky vývoje, ovládání systému a aplikace využívané k návrhu a tvorbě konkrétního redakčního systému.

ABSTRACT

This work is focused on implementation of redaction system, as a system for web content management, in JAVA language. Key concepts from the redaction systems are defined in the theoretical part and the key building blocks and functionality of these systems are described in detail. It is also highlighted on the front programming languages used for creating such systems and development environment designed for the purpose. The practical part clearly presents different steps of development, administration system and applications used for making design and creation of a specific redaction system.

Obsah

1. Úvod	11
2. Informační systémy ke správě webového obsahu	12
2.1. Správa obsahu – content management	12
2.2. Content Management System	13
2.2.1. Hlavní funkce CMS	14
2.2.2. Provoz CMS	14
2.2.3. Typy CMS podle funkčnosti.....	15
2.3. Web Content Management System.....	16
3. Redakční systém ke správě webového obsahu.....	16
3.1. Charakteristika redakčních systémů	17
3.2. Modulárnost redakčních systémů.....	18
3.3. Využití redakčních systémů.....	19
3.4. Výhody nasazení redakčních systémů	20
4. Technologie a nástroje k tvorbě redakčního systému	21
4.1. Modelovací jazyk UML a nástroj Enterprise Architect.....	22
4.2. Software Case studio	24
4.3. Programovací a jiné jazyky	25
4.3.1. JAVA	25
4.2.3. Technologie .NET	27
4.2.4. Scriptovací jazyk PHP	28
4.3. Databáze	28
4.3.1. Databázový systém Oracle.....	28
4.3.2. Databázový systém Postgresql.....	29
4.3.3. Databázový systém Mysql.....	29
5. Tvorba a implementace redakčního systému v jazyce JAVA.....	31
5.1. Charakteristika implementovaného redakčního systému	31
5.2. Zabezpečení systému	32
6. Postup tvorby aplikace	33
6.1. Analýza a návrh systému.....	33
6.2. Výběr a návrh databáze	45
7. Administrace systému	46
7.1. Spuštění systému	46

7.2.	Autentizace přihlašovacích údajů uživatele	49
7.3.	Modul Uživatelé	50
7.4.	Modul Menu	54
7.5.	Modul Text	56
7.6.	Modul Fotogalerie.....	58
7.7.	Modul Aktuality.....	63
8.	Použité technologie.....	65
9.	Závěr	66
10.	Seznam literatury	67
11.	Seznam obrázků	69

1. Úvod

Tato práce je zaměřena na analýzu, návrh a následnou implementaci redakčního systému v jazyce JAVA. Analýza a návrh je prováděna za pomoci jazyka UML, který prostřednictvím produktu Enterprise Architect představuje velice vhodný nástroj pro správný návrh aplikace. Aplikace je následně implementována v jazyce JAVA v prostředí NetBeans firmy SunMicrosystems.

Teoretická část je rozdělena na dva hlavní celky. V části první jsou vysvětleny pojmy problematiky správy obsahu a redakčních systémů, hlavní typy a charakteristiky. Druhá část je potom zaměřena na technologie a nástroje používané k návrhu a tvorbě redakčních systémů. Zde jsou uvedeny softwary vhodné k analýze a návrhu nových aplikací a zdůrazněny výhody plynoucí z používání těchto softwarů. Dále jsou zde zmíněny přední programovací jazyky dnešní doby, používané k tvorbě redakčních systémů a ukázány některé typy databází, se kterými je možno systém provozovat.

V části praktické je prezentována hlavní architektura systému, použité návrhové vzory, návrh systémové databáze a popsán postup při tvorbě systému od analýzy, přes návrh až po samotnou implementaci aplikace.

2. Informační systémy ke správě webového obsahu

2.1. Správa obsahu – content management

Content management neboli CM je sada procesů a technologií, podporujících životní cyklus digitálních informací. Právě digitální informace jsou často označovány jako „obsah“ nebo správněji „digitální obsah“. Digitální obsah může mít podobu textu, jako textového dokumentu, multimediálního souboru, jako audio nebo video souboru nebo jiného různorodého datového typu, spadajícího do životního cyklu obsahu, který potřebuje být spravován.

Životní cyklus dokumentu digitálního obsahu se sestává z šesti základních fází či etap a to **vytvoření**, **aktualizování**, **publikování**, **převádění**, **archivace** a **destrukce**. Pro příklad - dokument může být vytvořen jedním nebo více autory, kteří ho mohou publikovat. Publikování však může mít více forem. Publikování pak lze řídit i přes tzv. přístupová práva, která určují, kdo s čím bude pracovat a tak i mít možnost data publikovat.

Správa obsahu tvoří propojené, spolupracující postupy. Nejčastěji se skládají z následných rolí a jejich povinností:

- **autor** – stará se o editování a výrobu obsahu,
- **editor** – upravuje obsah a styly dokumentů, včetně jejich lokalizace a překladu,
- **vydavatel** – uvolňuje publikace na určitém uživatelském výstupu systému,
- **administrátor** – Edituje uživatelské účty a přiděluje nebo edituje jejich práva podle jejich funkce. Dále dohlíží a modifikuje přístupová práva nad určitými složkami a soubory,
- **zákazník, host** - osoba, která čte nebo používá publikace po jejich zveřejnění na Internetu.

Hlavní funkcí CM je schopnost správy více verzí daného obsahu dat, daných dokumentů v určitých fázích jejich vývoje. Autoři a editoři se často potřebují vracet ke starším verzím dokumentů.

Další funkcí CM je zahrnutí tvorby, údržby a použití revizních standardů. Každý člen revizního týmu požaduje jasné a stručné revizní standardy, které musí být udržovány k zabezpečení dlouhodobé konzistence a životnosti vědomostní databáze. (1)

2.2. Content Management System

Content Management System (systém pro správu obsahu) je systém užívaný uživateli ke správě obsahu. CMS usnadňuje organizaci, kontrolu a publikaci většího obsahu dokumentů a ostatních souborů typu obrázků nebo multimediální soubor.

Historie

Pojem Content Management System byl dříve užíván k označení systému pro správu webů a správu systémů. Brzy se však systémy pro správu obsahu vyvinuly do systémů, určených především pro tvorbu webových publikací jako například on-line magazíny, noviny, atd. V roce 1995 CNET spojil web dokument management a publikační systém do separované společnosti nazvané Vignette, která otevřela trh s komerčními content management systémy.

S růstem poptávky se rozsah produktů, propagovaných jako systémy pro správu obsahu, značně rozšířil a systémy se dnes mnohem více specifikují a rozdělují do jednotlivých skupin, dle jejich účelů. Lze tedy CMS rozdělit na různé formy. (2)

Formy

CMS systém může mít tyto podoby:

- Web content management system (WCMS) – systém pro správu webového obsahu,
- Dokument management system (DMS) – systém pro správu dokumentů,
- Systém v novinářských společnostech,
- Workflow (životní cyklus) publikovaných článků,
- Single Source CMS – obsah je uložen bez použití relačních databází.

2.2.1. Hlavní funkce CMS

CMS většinou nabízí tyto funkce:

- **import a vytváření** dokumentů a multimediálních souborů,
- **identifikaci** všech klíčových uživatelů systému a jejich rolí,
- vytváření **rolí a povinností** různých obsahových kategorií nebo typů,
- **definici úlohy workflow**, často spojované s hlášením událostí při změně obsahu,
- **záznam a správu** více verzí dokumentů,
- **publikaci obsahu** a úložiště obsahu. Stále více se tyto úložiště obsahu objevují v systémech,
- Úpravu **formátu textu**. Některé systémy podporují automatické nastavení barev a ostatních atributů textů k formátování textu.

2.2.2. Provoz CMS

Systémy pro správu obsahu většinou běží na webových serverech (obrázek 1). Většina systémů nabízí ovládané přístupy pro různé pozice uživatelů formou přístupových práv. Přístupy jsou většinou přes webový prohlížeč nebo FTP, k nahrávání dokumentů. Každá pozice uživatele má na starosti odlišnou práci. Zde je výpis rolí uživatelů.

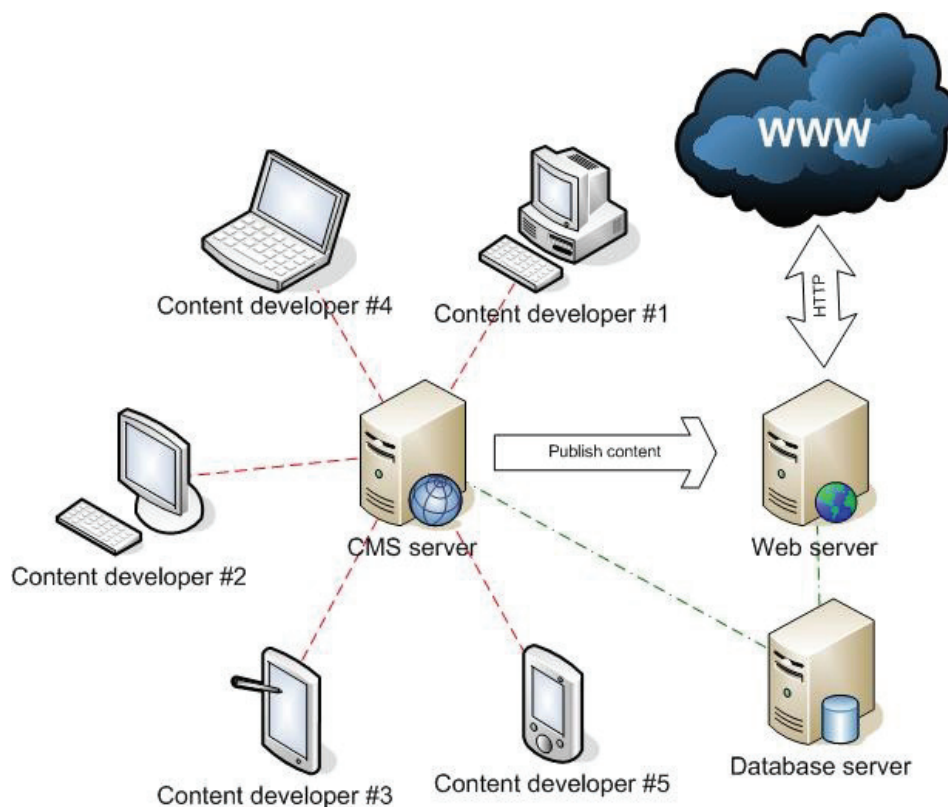
- **Content creator** – přidává dokumenty do systému
- **Copy editor** – komentuje příjem a vyřazení dokumentů
- **Layout editor** – navrhuje vzhled stránek
- **Editor chief** – odpovídá za publikování vytvořených publikací na stránkách

Content management systém kontroluje a pomáhá řídit jednotlivé kroky vývoje dokumentů, dle práce jednotlivých uživatelů systému. Spravuje celý systém a stará se o publikaci dokumentů internetových prezentací.

Obsah a všechny ostatní informace související s danou stránkou jsou většinou uloženy v server-base relačním databázovém systému. CMS pak tedy v databázi udržuje všechny verze dokumentů. Ve větších společnostech tyto server-

base dokumenty potřebují komunikovat s desktopovými aplikacemi a Open Document Management APIs.

Nedávno byly CMS systémy asociovány se systémy CRM (Customer Relation Management nebo Constituent Relationship Management). (2)



Obrázek 1. Integrace CMS serveru do internetu (13)

2.2.3. Typy CMS podle funkčnosti

Module-based CMS – většinu prací v životním cyklu dokumentů obsluhují CMS moduly. Mezi běžné moduly patří vytváření/editace, změna a publikace.

Document transformation language-based CMS – Nabízí jiný přístup při používání CMS a to využitím volných standardů. XSLT-based CMS převádí připravené dokumenty z XML dat a XSLT-šablon. XML Sapiens-based CMS převádí dokumenty ze streamu čistých dat, navrhuje a zajišťuje funkčnost šablon.

Web-based CMS – k práci s daty používá databáze, jako MySQL, PostgreSQL nebo MS SQL, a skriptovací jazyky nebo nástroje jako Coldfusion, PHP, jsp nebo ASP. Data jsou z databáze přes dotazy SQL vybírána a převáděna přímo v Html stránkách nebo ostatních dokumentech a upravována za pomoci Cascading style sheets,

sloužících k úpravě vlastností textu a dokumentu. Tyto WCMS mohou obsahovat mnoho funkcí/modulů jako diskusní fórum, blog nebo třeba emailové rozesílání novinek. (2)

2.3. Web Content Management System

Web Content Management System neboli systém pro správu webového obsahu, je užíváný ke správě a řízení velkého, dynamického webového obsahu dat (HTML dokumentů a jejich obrázků). Měl by obsahovat všechny funkce, kterými disponuje CMS, které se však zaměřují na správu informačního obsahu webové prezentace.

Funkce WCMS

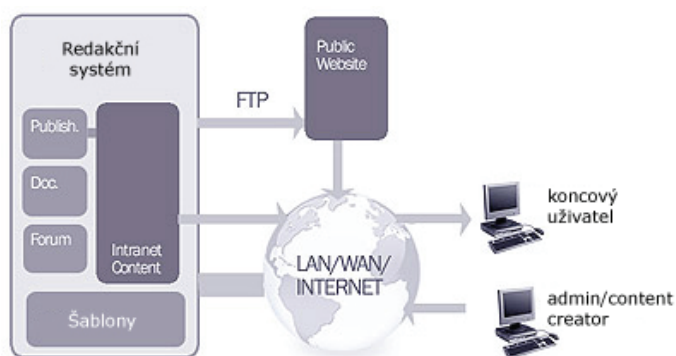
Na rozdíl od CMS, který umožňuje editování, řízení a workflow dokumentů se WCMS zaměřuje na tyto funkce:

- **automatické šablony** – umožňuje vytváření a následné aplikování vizuálních šablon na náš obsah na webu,
- **jednoduchá editace obsahu** – jednoduchá editace textů, obrázků, atd. Většinou je zde použit nějaký z WYSIWYG editorů,
- **modulárnost** – většina CMS systémů obsahuje různé moduly, které lze jednoduše přidat či aktivovat a tím tedy přidat či odebrat určitou funkci tohoto systému,
- **upgrady webových standardů** – zajišťuje upgrady nových funkčních nastavení a nastavení systému k aktuálním webovým standardům,
- **workflow management** – je proces následných a paralelních úloh při vytváření a publikování jednotlivých dokumentů. Od jeho vytvoření, publikace, archivování po jeho destrukci. (3)

3. Redakční systém ke správě webového obsahu

Redakční systémy spadají do množiny systémů pro správu webového obsahu. Hlavním rozdílem je jejich konkrétní zaměření a menší rozsah funkcí, oproti robustním WCMS systémům.

Redakční systém, dále „RS“, vychází z WCMS a slouží k jednoduchému publikování a správě stránek na Internetu pohodlnou editací již existujících webových prezentací a jejich webového obsahu (obrázek 2). Některé RS systémy jsou však tak obsáhlé a disponují takovými funkcemi, že je lze využít i při samotné tvorbě www stránek. Hlavním rysem tohoto systému je jednoduchá ovladatelnost i pro uživatele, počítačové začátečníky, kteří mohou stránky jednoduše a efektivně spravovat bez sebemenších znalostí jazyků na tvorbu webu (html, php, atd.).



Obrázek 2. Redakční systémů a přístup uživatelů (14)

3.1. Charakteristika redakčních systémů

Redakční systémy obsahují nepřehledné množství různých funkcí, jejichž počet a kvalita jednotlivé redakční systémy odlišuje.

Základní funkce, které redakční systém umožňuje:

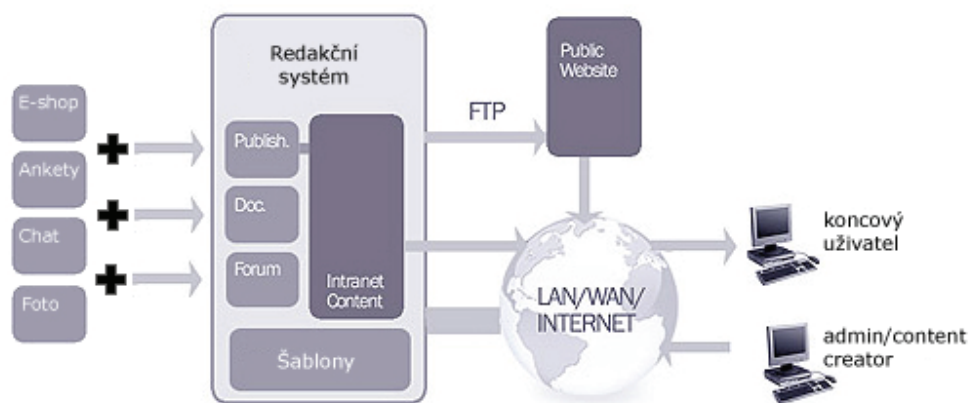
- **decentralizovaná správa obsahu** - znamená, že web může spravovat více uživatelů,
- **úprava článků ve WYSIWYG editorech** – uživatel nemusí znát jazyk html pro úpravu dokumentů,
- **náhled stránek** – umožňuje náhled na stránky a přímo upravovaný blok informací,
- **systém přístupových práv** - všichni uživatelé rs mají přidělená různá práva, které zajišťují, že daný uživatel má možnost úpravy jen daného bloků dat v RS (např.: úprava dokumentů, přidávání fotek, atd.),

- **přístup přes webový prohlížeč** – údržbu obsahu internetové prezentace lze provádět z jakéhokoliv místa s připojením na internet,
- **redakční cyklus** – proces následných úloh při vytváření a publikování webového obsahu. (autor napíše článek, redaktor jej upraví, šéfredaktor rozhodne o publikaci),
- **snadná provázanost příbuzných stránek** – jednoduše provázat příbuzné stránky,
- **publikace různého typu materiálů** – možnost publikace materiálů více formátů: PDF, Doc, Excel, atd.,
- **udržování časové aktuálnosti stránek** - nastavení začátku a konce doby, kdy bude stránka publikována. Důležitá vlastnost pro on-line noviny a časopisy,
- **důsledné oddělení obsahu, logické struktury a vzhledu** – vzhled bude založen na šablonách, pomocí šablon se zobrazí obsah, web bude mít jednotný vzhled,
- **personalizaci** - zobrazuje různý obsah webu podle nastavených preferencí, např. volný a placený obsah,
- **uchovávání verzí obsahu** - umožňuje návrat k předchozí verzi nebo některé části obsahu,
- **stoprocentní dodržování standardů W3C** – dodržování standardů pro psaní webových stránek neboli validace www. Dále stejnorodé promítnutí www stránek v různých prohlížečích.

(4)

3.2. Modulárnost redakčních systémů

Většina redakčních systémů je modulárních (obrázek 3). To znamená, že k základním modulům je možné přidávat další moduly, nabízející rozšiřující funkce daného redakčního systému. Moduly většinou poskytuje sám výrobce RS. Přídavné moduly jsou buďto placené nebo jsou nabízeny zcela zdarma, někdy i včetně zdrojových kódů. Příkladem přídavného modulu může být internetový obchod, diskuse, chat, album fotografií, atd.



Obrázek 3. Moduly redakčního systému

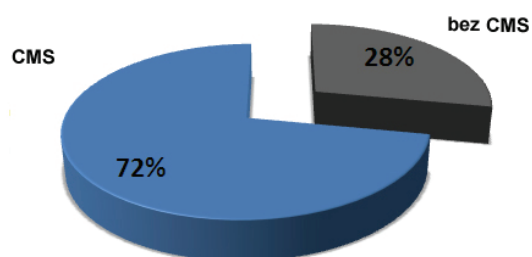
3.3. Využití redakčních systémů

Redakční systémy (dále jen RS) lze využít pro velmi jednoduchou správu www stránek. Uplatní se u firem, které mají zájem o pravidelnou a jednoduchou údržbu své webové prezentace s cílem stálé aktualizace.

Nejčastější nasazení RS:

- komerční webové stránky,
- portály, galerie, katalogy,
- osobní webové prezentace,
- weblogy a blogy,
- intranet / extranet. (5)

Na obrázku 4, lze vidět procentuální využití redakčních systémů na správě webových prezentacích.



Obrázek 4. Využití redakčních systémů. (15)

Typy RS

V dnešní době zvyšující se poptávky po internetových prezentacích, existuje nespočetné množství redakčních systémů, které lze rozdělit na 3 základní skupiny.

- 1) Freeware redakční systémy založené na platformě OpenSource
- 2) Komerční redakční systémy
- 3) Redakční systémy proprietární – vytvořené pro konkrétní nasazení

3.4. Výhody nasazení redakčních systémů

Firma musí na internetového zákazníka zapůsobit pozitivně. V dnešní době uživatel neoceňuje pouze grafické zpracování, ale především aktuálnost, přehlednost a rychlou dostupnost potřebných informací. K tomu firmy využívají možnosti nasazení redakčního systému.

Výhody, které firmě přináší nasazení redakčního systému jsou:

- **jednoduchá správa webového obsahu** – editace, vytváření, přidávání a mazání dokumentů přímo v systému znamená pro firmu jednoduchou správu své internetové prezentace. Spolu s workflow managementem, zajišťujícím životní cykly dokumentů, je správa dokumentů opravdu kvalitní,
- **jednoduchá obsluha** – velká výhoda redakčních systémů je vlastnost, že lze jejich správu provádět z webového prohlížeče, což přináší možnost editace z jakéhokoli místa, s připojením na internet. Uživatel se pouze přihlásí a prezentaci spravuje, odkudkoliv potřebuje.
- **efektivní údržba** – správa již výše zmíněného webového obsahu, tedy dokumentů, je velice efektivní z jednoho prostého důvodu. Uživatel nemusí zdlouhavě přepisovat html dokumenty, jak je tomu ve statických webových stránkách, ale stačí nahrát dokumenty přes redakční systém, který je už následně vypíše či upraví dle potřeb uživatele. Odpadá tak i nutnost dočasného odstavení prezentace při její editaci,
- **finanční efektivita** – nasazením redakčního systému firma většinou ušetří nemalé množství nákladů jinak vzniklých s úpravami prezentace

odborníkem. Redakčním systémem nabízí firmě jednoduchou zprávu prezentace a to tak jednoduchou, že to většinou zvládne jakýkoliv zaměstnanec dané firmy,

- **aktuálnost** – uživatel navštívuje webové stránky, z důvodu zjištění nových informací, nákupu nového zboží, atd. V případě, že by stránky nebyly aktualizovány, přichází firma o většinu zákazníků. Díky redakčnímu systému však není aktualizace takový problém. Díky správě dokumentů je to hračka a tak je aktuálnost webové prezentace pro firmu samozřejmostí,
- **analýza uživatelů** – sledování návštěvnosti, jednotlivých odkazů, či částí firemní prezentace, nebo umístěním anket na stránky, pomáhá při rozšiřování, zdokonalování a zlepšení webové stránky. Tím, že jsou stránky uživatelsky příjemné, docílíme toho, že se na onu webovou prezentaci uživatelé rádi vrací. Analýza nám tedy slouží ke zjištění informací o uživateli,
- **statistiky prodeje** – využívají se při nabízení produktů prostřednictvím elektronického obchodu. Slouží firmě - uživateli k různým statistikám o prodeji z různých hledisek pohledu, jako například: objem prodeje, zisk, obrat atd.

4. Technologie a nástroje k tvorbě redakčního systému

Správný návrh a analýza budoucího systému je prvním krokem ke kvalitní a plně funkční aplikaci. Správným návrhem dokážeme lehce předejít vzniku budoucích problémů, odhalit hlavní nedostatky aplikace a tak značně zefektivnit tvorbu dané aplikace a zvolit správné technologie k její tvorbě. Mezi nejpoužívanější jazyk určený pro tvorbu analýz a návrhů softwaru patří jazyk UML, ve kterém se nejčastěji modeluje v programu Enterprise Architect firmy Spark Systems. Na návrh databází lze využít softwaru CaseStudio firmy Quest software. Nejmodernějšími dnešními jazyky pro tvorbu redakčních systémů jsou: JAVA a .NET.

4.1. Modelovací jazyk UML a nástroj Enterprise Architect

Jazyk UML (Unified Modeling Language) je univerzální jazyk pro vizuální modelování systému. Přestože je nejčastěji spojován s modelováním objektově orientovaných softwarových systémů, má mnohem širší využití, což vyplývá z jeho zabudovaných rozšiřovacích mechanismů.

Jazyk UML byl navržen proto, aby spojil nejlepší existující postupy modelovacích technik a softwarového inženýrství. Diagramy vytvořené v jazyku UML jsou srozumitelné pro lidi, ale navíc je mohou interpretovat i programy CASE. Je nesmírně důležité si uvědomit, že jazyk UML nenabízí žádný druh metodiky modelování.

Unified Proces UP již ovšem metodikou je. Sděluje nám, jaké pracovníky musí využít, jaké činnosti vykonat a jaké produkty vyrobit, aby se nám podařilo sestavit model funkčního softwarového systému.

K modelování systému s využitím jazyku UML nám poslouží software **Enterprise Architect** (EA). EA je profesionální nástroj pro snadnou tvorbu vývojových diagramů a dalších schémat potřebných pro vývoji aplikací. Nástroj při modelování aplikací využívá jazyka UML. Enterprise Architect podporuje mimo jiné i generování zdrojových kódů programovacích jazyků C++, C#, Java, Delphi, VB.Net, Visual Basic, ActionScript a PHP.

Metodika Unified Proces

Metodika UP je založena na iterativním a přírůstkovém procesu. Základní myšlenkou je rozložení velkého softwarového projektu na řadu menších „miniprojektů“. Každý z miniprojektů je považován za iteraci. Metodika UP specifikuje pět základních aktivit:

- požadavky,
- analýza,
- návrh,
- implementace,
- testování.

Iterace se opakují tak dlouho, dokud není dosažena konečná podoba vytvářeného systému.

Požadavky

Požadavky by měly být jediným vyjádřením, co by měl systém dělat, nikoli toho, jak by to měl dělat.

Požadavek lze dále definovat jako specifikaci toho, co by mělo být v systému implementováno. Ke grafickému zobrazení specifikace požadavků používáme tzv. diagramy případů užití (UseCase diagramy).

Use case diagram představuje zobrazení dynamické (funkční) struktury systému z pohledu uživatele. Je primárně určen k definici chování systému, aniž by odhaloval jeho vnitřní strukturu. Obsahuje soubor scénářů pro používání systému. Každý scénář obsahuje sekvenci (posloupnost) událostí, které v jeho rámci probíhají (včetně případných variant). Požadavky na informační systém lze rozdělit:

- a) ne-funkční (např. bezpečnost, cena, počet uživatelů),
- b) funkční (funkcionalita užitečná pro nějakou externí entitu) - ty jsou zobrazitelné prostřednictvím Use Case.

Analýza

Záměrem analýzy je tvorba analytického modelu. Tento model se zaměřuje na to, co systém musí udělat, avšak nezabývá se detaily týkajícími se způsobu, jakým to udělá. Výstupem analýzy jsou dva klíčové artefakty:

- Analytické třídy,
- realizace případů užití.

Návrh

Návrhový model lze považovat za rozpracovanou formu analytického modelu, do něhož jsou přidány detaily a specifická technická řešení. Návrhový model obsahuje stejné typy předmětů jako analytický model, ale veškeré artefakty jsou již plně utvořeny a musí nyní obsahovat rovněž implementační detaily.

Návrhové třídy jsou tedy takové třídy, jejichž specifikace je na takové úrovni, že je lze implementovat.

Implementace a testování

V těchto fázích dochází již k implementaci systému, vycházející z vygenerovaných návrhových tříd. Po kompletní implementaci daného systému přecházíme k testování a odlaďování případných vzniklých chyb. (6)

4.2. Software Case studio

Case studio je software pro vizuální navrhování databázových struktur. Program umožňuje navrhovat entitně relační diagramy pro rozličné databáze (Oracle, MS SQL, DB2, Firebird, Advantage DB server, Interbase, MaxDB, MS Access, MySQL, PostgreSQL a další) a jejich klíčové vlastnosti.

Klíčové vlastnosti:

- entitně relační diagramy (ERD),
- data flow Diagramy (DFD),
- reverse engineering - umožňuje vytvořit model struktury již existující databáze,
- správce verzí - umožňuje porovnávat jednotlivé verze modelů,
- velice detailní logické i fyzické HTML reporty,
- galerie -pro uložení nejčastěji používaných částí modelů,
- podpora uživatelů, uživatelských skupin a uživatelských práv,
- uživatelsky definované šablony, možnost zapisování akcí do "To do" listu,
- datový slovník,
- referenční integrita (deklarativně i za pomoci triggerů),
- možnost vytvářet submodely a další.

4.3. Programovací a jiné jazyky

4.3.1. JAVA

Java je objektově orientovaný programovací jazyk, který vyvinula firma Sun Microsystems a představila 23. května 1995.

Java je jedním z nejpoužívanějších programovacích jazyků na světě. Díky své přenositelnosti je používán pro programy, které mají pracovat na různých systémech počínaje čipovými kartami (platforma JavaCard), přes mobilní telefony a různá zabudovaná zařízení (platforma Java ME), aplikace pro desktopové počítače (platforma Java SE) až po rozsáhlé distribuované systémy pracující na řadě spolupracujících počítačů rozprostřené po celém světě (platforma Java EE). Tyto technologie se jako celek nazývají platforma Java. Dne 8. května 2007 Sun uvolnil zdrojové kódy Javy (cca 2,5 miliónů řádků kódu) a Java je dále vyvíjena jako open source.

Základní vlastnosti:

- **jednoduchý** – jeho syntaxe je zjednodušenou (a drobně upravenou) verzí syntaxe jazyka C a C++,
- **objektově orientovaný** – s výjimkou osmi primitivních datových typů jsou všechny ostatní datové typy objektové,
- **distribuovaný** – je navržen pro podporu aplikací v síti ,
- **interpretovaný** – místo skutečného strojového kódu se vytváří pouze tzv. mezikód (bajtkód). Tento formát je nezávislý na architektuře počítače nebo zařízení. Program pak může pracovat na libovolném počítači nebo zařízení, který má k dispozici interpret Javy, tzv. virtuální stroj Javy - Java Virtual Machine (JVM). V pozdějších verzích Javy nebyl mezikód přímo interpretován, ale před prvním svým provedením dynamicky zkompileován do strojového kódu daného počítače (tzv. just in time compilation - JIT). Tato vlastnost zásadním způsobem zrychlila provádění programů v Javě, ale výrazně zpomalila start programů. V současnosti se převážně používají technologie zvané HotSpot compiler, které mezikód zpočátku interpretují a na základě statistik získaných z této interpretace později provedou překlad často používaných částí do strojového

kódu včetně dalších dynamických optimalizací (jako je např. inlining krátkých metod atp.),

- **robustní** – je určen pro psaní vysoce spolehlivého softwaru. Správa paměti je realizována pomocí automatického Garbage collectoru, který automaticky vyhledává již nepoužívané části paměti a uvolňuje je pro další použití,
- **bezpečný** – má vlastnosti, které chrání počítač v síťovém prostředí, na kterém je program zpracováván, před nebezpečnými operacemi nebo napadením vlastního operačního systému nepřátelským kódem,
- **nezávislý na architektuře** – vytvořená aplikace běží na libovolném operačním systému nebo libovolné architektuře. Ke spuštění programu je potřeba pouze to, aby byl na dané platformě instalován správný virtuální stroj. Podle konkrétní platformy se může přizpůsobit vzhled a chování aplikace,
- **přenositelný** – vedle zmíněné nezávislosti na architektuře je jazyk nezávislý, i co se týká vlastností základních datových typů (je například explicitně určena vlastnost a velikost každého z primitivních datových typů). Přenositelností se však myslí pouze přenášení v rámci jedné platformy Javy (např. J2SE). Při přenášení mezi platformami Javy je třeba dát pozor na to, že platforma určená pro jednodušší zařízení nemusí podporovat všechny funkce dostupné na platformě pro složitější zařízení a kromě toho může definovat některé vlastní třídy doplňující nějakou speciální funkčnost nebo nahrazující třídy vyšší platformy, které jsou pro nižší platformu příliš komplikované,
- **výkonný** – přestože se jedná o jazyk interpretovaný, není ztráta výkonu významná, neboť překladače pracují v režimu „*právě včas*“ a do strojového kódu se překládá jen ten kód, který je opravdu zapotřebí,
- **víceúlohový** – podporuje zpracování vícevláknových aplikací,
- **dynamický** – Java byla navržena pro nasazení ve vyvíjejícím se prostředí. Knihovna může být dynamicky za chodu rozšiřována o nové třídy a funkce, a to jak z externích zdrojů, tak vlastním programem,
- **elegantní** – velice pěkně se v něm pracuje, je snadno čitelný (např. i pro publikaci algoritmů), přímo vyžaduje ošetření výjimek a typovou kontrolu.

Nevýhody:

Proti programovacím jazykům, které provádějí tzv. statickou kompilaci (např. C++), je start programů psaných v Javě pomalejší, protože prostředí musí program nejprve přeložit a potom teprve spustit. Je však možnost využít mechanismů JIT a HotSpot, kdy se často prováděné nebo neefektivní části kódu přeloží do strojového kódu a program se zrychlí.

Další nevýhodou projevující se hlavně u jednodušších programů je větší paměťová náročnost při běhu způsobená nutností mít v paměti celé běhové prostředí. (7)

4.2.3. Technologie .NET

.NET je zastřešující název pro soubor technologií v softwarových produktech, které tvoří celou platformu, která je dostupná nejen pro Web, Windows i Pocket PC. Common Language Infrastructure je standardizovaná specifikace jádra .NET. Základní komponentou je Microsoft .NET Framework, prostředí potřebné pro běh aplikací a nabízející jak spouštěcí rozhraní, tak potřebné knihovny. Pro vývoj .NET aplikací vydal Microsoft Visual Studio .NET.

Platforma .NET nepředepisuje použití žádného programovacího jazyka. Bez ohledu na to, v čem byla aplikace původně napsána, se vždy přeloží do mezijazyka Common Intermediate Language.

Nejpoužívanější programovací jazyky pro vývoj .NET aplikací jsou C#, Visual Basic .NET a Delphi. C# je programovací jazyk podobný jazykům C nebo Java. VB.NET je pokračovatelem jazyka Visual Basic. Pro vývoj webových aplikací .NET nabízí jazyk ASP. (8)

K dispozici je nicméně řada dalších programovacích jazyků, například:

- Managed C++
- F# (funkcionální programovací jazyk)
- J# (jazyk velmi podobný Javě)
- IronPython

4.2.4. Scriptovací jazyk PHP

Původně Personal Home Page, dnes Hypertext Preprocessor, spadá do skupiny serverových skriptovacích jazyků. Vnikl v roce 1996 a od té doby tento jazyk prošel velkými změnami. Je to objektivě orientovaný programovací jazyk, strukturou velmi podobný jazyku C++. Znalost objektivě orientovaného programování (OOP), nebo přímo C++ (nebo nějakého jiného podobného jazyka, například JavaScript) tedy může být při práci v PHP výhodou, není však nutnou podmínkou.

PHP lze charakterizovat ve 3 bodech.

- 1) PHP je zcela zdarma. Lze ho zdarma stáhnout z jeho domovské adresy www.php.net,
- 2) PHP funguje na platformě Windows, Linux tak Mac OS,
- 3) Kvalita PHP pomohla dostat se na pozici, kterou v dnešní době zastává a to jednoho z nejpoužívanějších programovacích jazyků pro tvorbu webu.

(9)

4.3. Databáze

4.3.1. Databázový systém Oracle

Oracle je systém řízení báze dat (Oracle database management system – DBMS), moderní multiplatformní databázový systém s velice pokročilými možnostmi zpracování dat, vysokým výkonem a snadnou škálovatelností.

Přesněji Oracle Corporation je název firmy, oficiální název databázové platformy je Oracle Database.

Aktuální verzí je Oracle Database 11g. Tento systém podporuje nejen standardní relační dotazovací jazyk SQL podle normy SQL92, ale také proprietární firemní rozšíření Oracle (např. pro hierarchické dotazy), imperativní programovací jazyk PL/SQL rozšiřující možnosti vlastního SQL (v tomto jazyce je možné tvořit uložené procedury, uživatelské funkce, programové balíky a trigger), dále podporuje objektivě databáze a databáze uložené v hierarchickém modelu

dat (XML databáze, jazyk XSQL). Dále též obsahuje širokou paletu nástrojů pro podporu snadného nasazení na gridových sítích (písmeno g v označení verze je zkratkou "Growing to Grid"). Grid Computing podporovala i verze 10g (zde písmeno g značí pouze slovo Grid). (10)

4.3.2. Databázový systém PostgreSQL

PostgreSQL je relační databázový systém s otevřeným zdrojovým kódem. Má za sebou více než patnáct let vývoje a zakládá si na spolehlivosti a bezpečnosti. Je šířen pod licenci BSD, které umožňuje volné spojování otevřeného kódu s uzavřeným. Často je srovnáván s další rozšířenou otevřenou databází MySQL.

Funguje pod rozšířenými operačními systémy včetně Windows a Linuxu. Splňuje podmínky ACID, plně podporuje cizí klíče, operace JOIN, pohledy, spouště a uložené procedury. Obsahuje většinu SQL92 a SQL99 datových typů.

PostgreSQL umožňuje běh uložených procedur napsaných v několika programovacích jazycích, v Perlu, v Pythonu, v jazyku C nebo v speciálním PL/pgSQL - jazyku vycházejícím z PL/SQL firmy Oracle.

Předností systému PostgreSQL je rozšiřitelnost. Systém lze obohatit o nové datové typy, funkce, operátory, agregační funkce, procedurální jazyky.

Nevýhodou, zvláště pro jednotlivce a malé firmy, je (hlavně v porovnání s MySQL) malá rozšířenost PostgreSQL na hostingových serverech a menší komunita, která by pomáhala s případnými problémy. (11)

4.3.3. Databázový systém Mysql

MySQL je databázový systém, vytvořený švédskou firmou MySQL AB. Jeho hlavními autory jsou Michael „Monty“ Widenius a David Axmark. Je považován za úspěšného průkopníka dvojího licencování – je k dispozici jak pod bezplatnou licenci GPL, tak pod komerční placenou licenci.

MySQL je multiplatformní databáze. Komunikace s ní probíhá – jak už název napovídá – pomocí jazyka SQL. Podobně jako u ostatních SQL databází se jedná o dialekt tohoto jazyka s některými rozšířeními.

Pro svou snadnou implementovatelnost (lze jej instalovat na Linux, MS Windows, ale i další operační systémy), výkon a především díky tomu, že se jedná o volně šiřitelný software, má vysoký podíl na v současné době používaných databázích. Velmi oblíbená a často nasazovaná je kombinace MySQL, PHP a Apache jako základní software webového serveru.

MySQL bylo od počátku optimalizováno především na rychlost, a to i za cenu některých zjednodušení: má jen jednoduché způsoby zálohování, a až donedávna nepodporoval pohledy, trigger, a uložené procedury. Tyto vlastnosti jsou doplňovány teprve v posledních letech, kdy začaly nejčastějším uživatelům produktu (programátorům webových stránek) již poněkud scházet. (12)

5. Tvorba a implementace redakčního systému v jazyce JAVA

5.1. Charakteristika implementovaného redakčního systému

Aplikace představuje redakční systém určený ke správě internetové prezentace. Aplikace je naimplementovaná v jazyce Java, který umožňuje spuštění na různých platformách. Multiplatformnost je jedním z hlavních cílů - požadavků této aplikace. Mimo aplikace redakčního systému určeného k administraci, tato práce obsahuje ještě ukázkový výstup aplikace, který je naimplementován v jazyce JSP, vycházejícím z Javy a speciálně vyvinutým k tvorbě webových prezentací. Ukázkovou webovou prezentaci si lze prohlédnout na adrese <http://demo.redakcak.cz>.

Redakční systém je spouštěn jako aplikace v systému, nepředstavuje tedy klasickou formu redakčního systému, který je spouštěn prostřednictvím internetového prohlížeče. Systém spuštěný jako aplikace pouze na uživatelských stanicích je mnohem bezpečnější než běh systému dostupného odkudkoli s přístupem na internet. Nabízí větší spolehlivost, dynamičnost a 100% dostupnost.

Redakční systém je dále modulární a obsahuje pět základních modulů na správu webového obsahu, samozřejmě s možností přidání dalších modulů.

Aktuální moduly systému:

- modul Menu,
- modul Text,
- modul Fotogalerie,
- modul Aktuality,
- modul Uživatelé.

5.2. Zabezpečení systému

Přihlašovací údaje jsou ověřovány s databází, přičemž jsou jednotlivé údaje kódovány prostřednictvím MD5 algoritmem, který vychází z algoritmu Message-Digest, což je rozšířená rodina hashovacích funkcí, která vytváří ze vstupních dat výstup (otisk) fixní délky. Otisk je též označován jako miniatura, kontrolní součet (v zásadě nesprávné označení), fingerprint, hash (česky někdy psán i jako heš). Jeho hlavní vlastností je, že malá změna na vstupu vede k velké změně na výstupu, tj. k vytvoření zásadně odlišného otisku. MD5 se prosadil do mnoha aplikací (např. pro kontrolu integrity souborů nebo ukládání hesel). MD5 je popsán v internetovém standardu RFC 1321 a vytváří otisk o velikosti 128 bitů. Byl vytvořen v roce 1991 (Ronaldem Rivestem), aby nahradil dřívější hašovací funkci MD4.

Bohužel, tento algoritmus byl v roce 2004 prolomen a tak je tedy nevhodný pro kódování hesel či jiných citlivých informací. Z důvodu běhu databáze aplikace na hostingu, na kterém běží databáze postgresql verze 8.2.4, obsahující pouze kódování MD5, nebylo možné využít jiného novějšího a bezpečného kódování, zobrazeného níže.

Typy kódovacích algoritmů v novější verzi Postgresql databáze

Aktuální verze postgresql databáze 8.2.3. nabízí kódovací algoritmy dle tabulky 1.

Algorithm	Hashes/sec	For[a-z]	For[A-Za-z0-9]
crypt-bf/8	28	246 years	251322 years
crypt-bf/7	57	121 years	123457 years
crypt-bf/6	112	62 years	62831 years
crypt-bf/5	211	33 years	33351 years
crypt-md5	2681	2.6 years	2625 years
crypt-des	362837	7 days	19 years
sha1	590223	4 days	12 years
md5	2345086	1 day	3 years

Tabulka 1. Kódovací algoritmy (16)

Z tabulky lze snadno vyzorovat nevhodnost použití MD5 kódování oproti jiným, novějším kódovacím jazykům. V případě možnosti běhu databáze na verzi 8.2.3 bych využil kódování za pomoci algoritmu crypt-bf/8.

6. Postup tvorby aplikace

Postup tvorby probíhal v souladu s kroky vytváření projektu za pomoci jazyku UML a metodiky UP.

Prvním krokem byla analýza budoucího systému, což patří k nejdůležitějším částem tvorby aplikace. Správnou analýzou jednoduše se odhalí většina vstupních a výstupních požadavků uživatele a tak lze předejít nepříjemnostem spojených jak s předláváním kódu po jeho implementaci a tak změnám celkové funkčnosti daných částí systému.

V rámci analýzy se vytváří UseCase diagramy a analytické třídy. Po kompletní a korektní analýze budoucího systému následuje detailní návrh systému, při kterém se vytvoří návrhové třídy modelovaného systému.

Po těchto krocích dochází na implementaci systému a dále k jeho testování a nasazení.

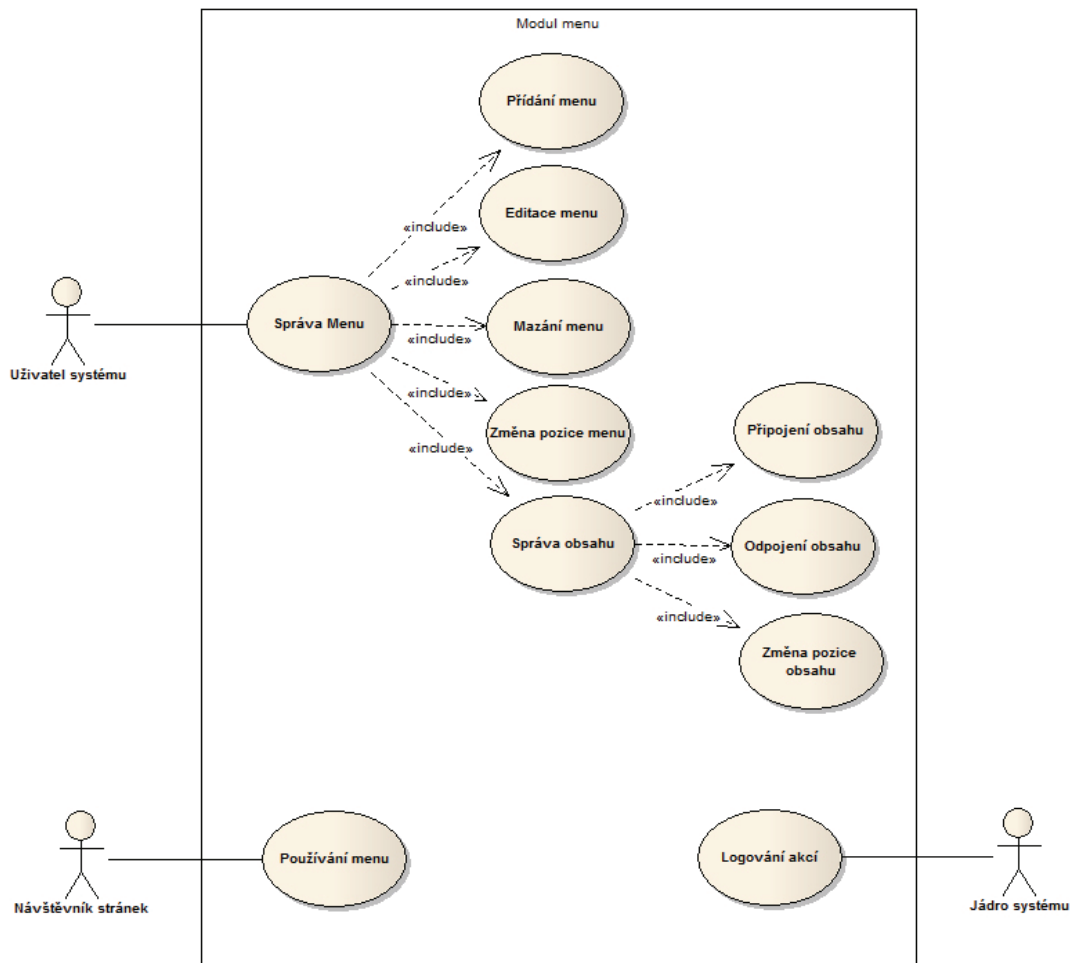
Paralelně s analýzou a návrhem aplikace je nutné navrhnout databázový model redakčního systému.

6.1. Analýza a návrh systému

Vytvoření UseCase diagramů

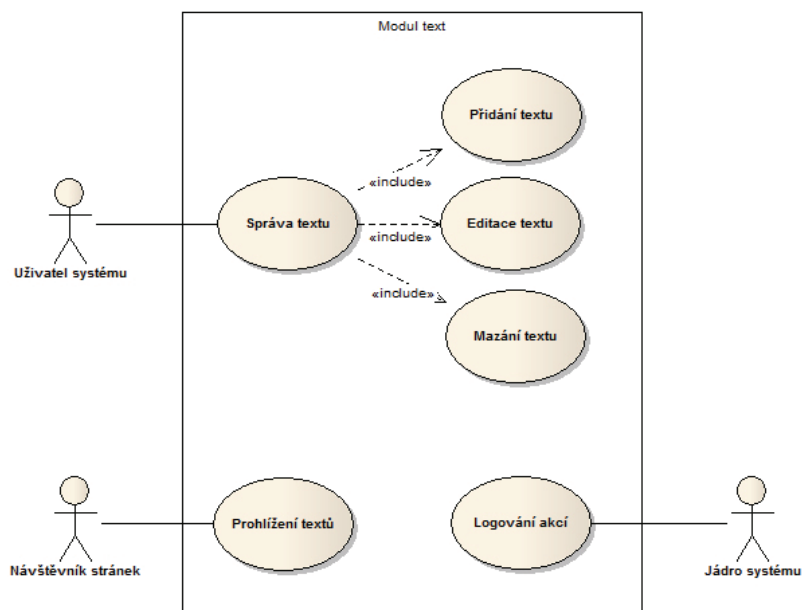
Use case diagram představuje zobrazení dynamické (funkční) struktury systému z pohledu uživatele a je primárně určen k definici chování systému, aniž by odhaloval jeho vnitřní strukturu.

Na stránkách 31 – 34 jsou prezentovány UseCase diagramy jednotlivých modulů systému. Na každém diagramu jsou tři uživatelé systému a namodelovány činnosti, které bude daný uživatel moci v systému provozovat.



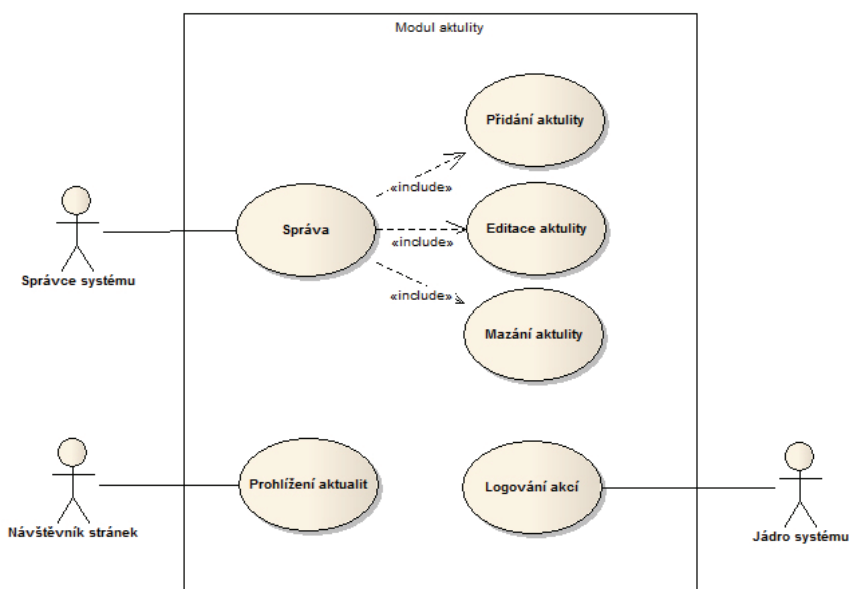
Obr. 5. UseCase modulu Menu

UseCase diagram menu (obr. 5) prezentuje, že tento modul je obsluhován dvěma typy uživatelů a jádrem systému. Prvním typem je uživatel systému, který obsluhuje správu celého menu, od přidávání položek menu, po napojování obsahu na položky menu. Druhým typem je prostý návštěvník stránek, který procházením menu prezentace prohlíží obsah stránek. Posledním činitelem je jádro systému, které automaticky loguje práce s modulem.



Obr. 6. UseCase modulu Text

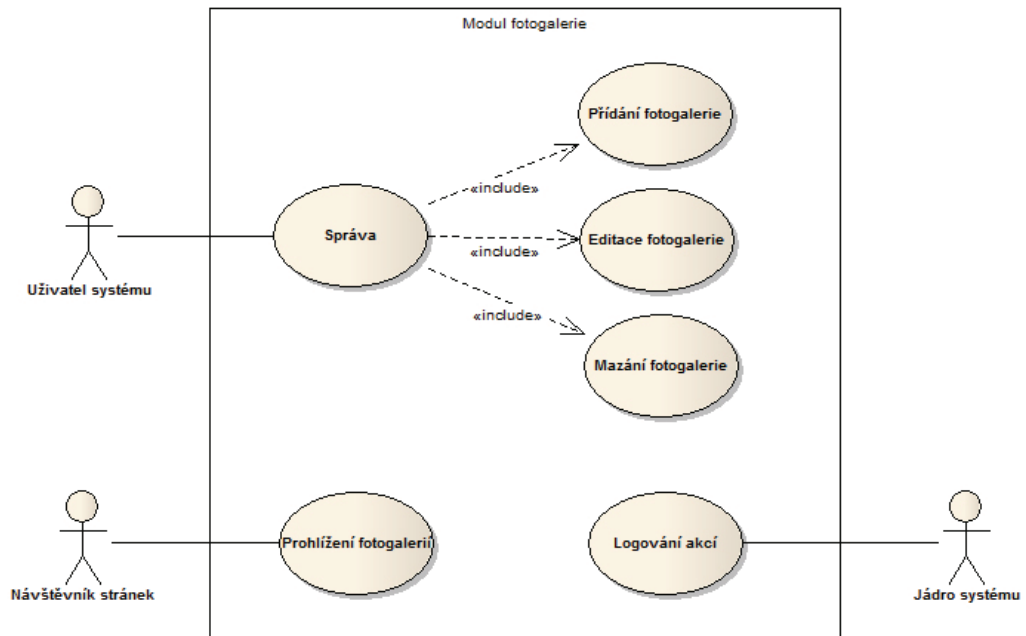
UseCase diagram text (obr. 6) prezentuje, že tento modul je obsluhován dvěma typy uživatelů a jádrem systému. Prvním typem je uživatel systému, který obsluhuje správu celého modulu text. Druhým typem je prostý návštěvník stránek, který procházením prezentace prohlídne texty připojené na dané položky menu. Posledním činitelem je jádro systému, které automaticky loguje práce s modulem.



Obr. 7. UseCase modulu Aktuality

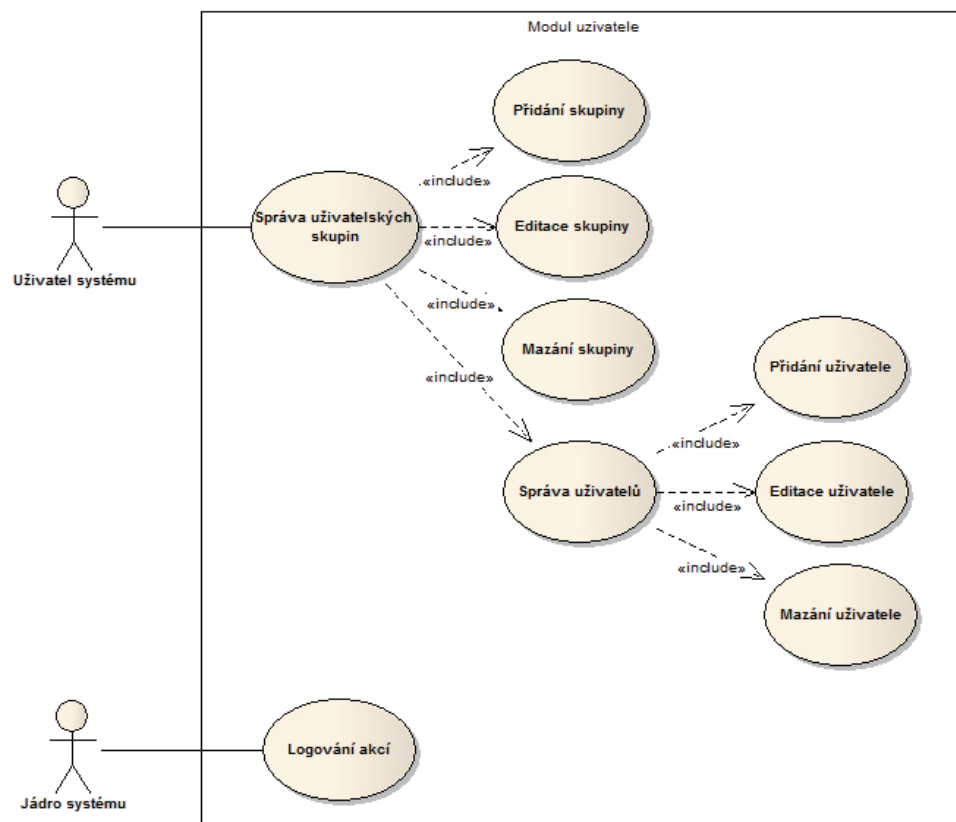
Z UseCase diagramu aktualit (obr. 7) je patrné, že tento modul je obsluhován dvěma typy uživatelů a jádrem systému. Prvním typem je uživatel systému, který

obsluhuje správu celého modulu aktualit. Druhým typem je prostý návštěvník stránek, který procházením prezentace prohlíží aktuality připojené na dané položky menu. Posledním činitelem je jádro systému, které automaticky loguje práce s modulem.



Obr. 8. UseCase modulu Fotogalerie

UseCase diagram fotogalerie (obr. 8) prezentuje, že tento modul je obsluhován dvěma typy uživatelů a jádrem systému. Prvním typem je uživatel systému, který obsluhuje správu celého modulu fotogalerie. Druhým typem je prostý návštěvník stránek, který procházením prezentace prohlíží fotogalerie připojené na dané položky menu. Posledním činitelem je jádro systému, které automaticky loguje práce s modulem.



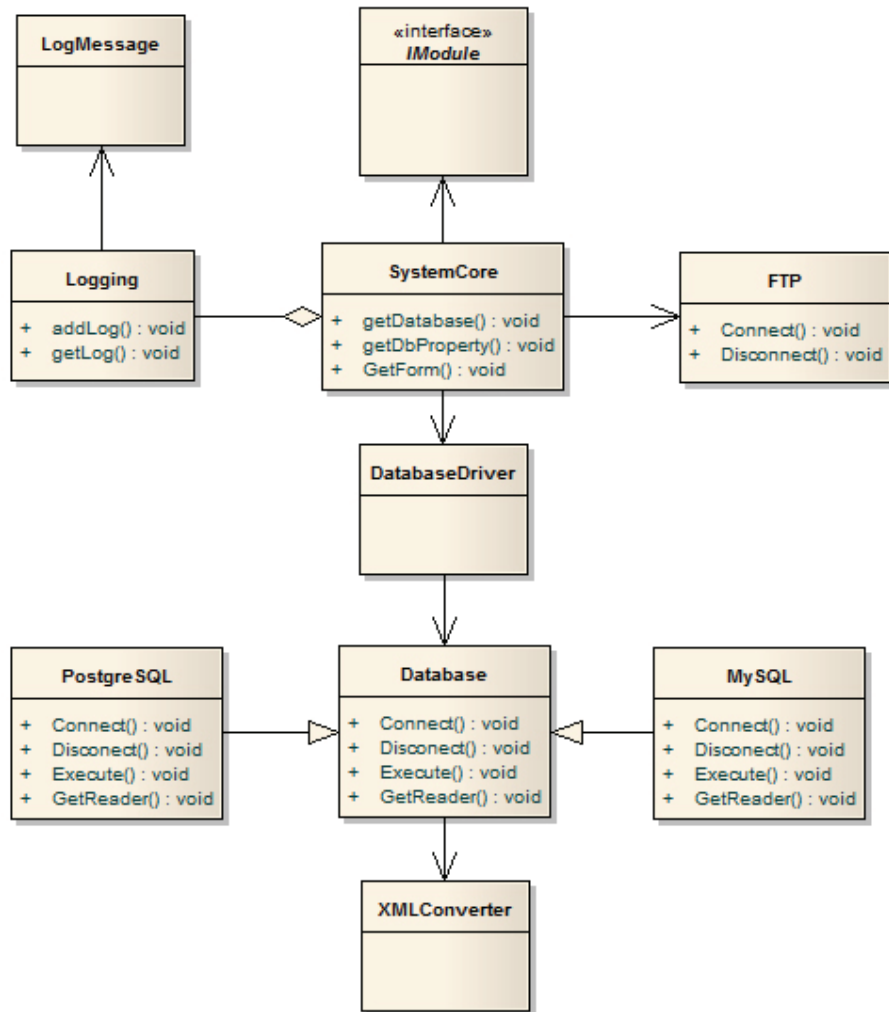
Obr. 9. UseCase modulu Uživatelé

Z UseCase diagramu uživatelé (obr. 9) je patrné, že tento modul je obsluhován jedním typem uživatele a jádrem systému. Uživatel systému obsluhuje správu celého modulu uživatelé. Jádro systému automaticky loguje práce s modulem.

Analytické a návrhové třídy

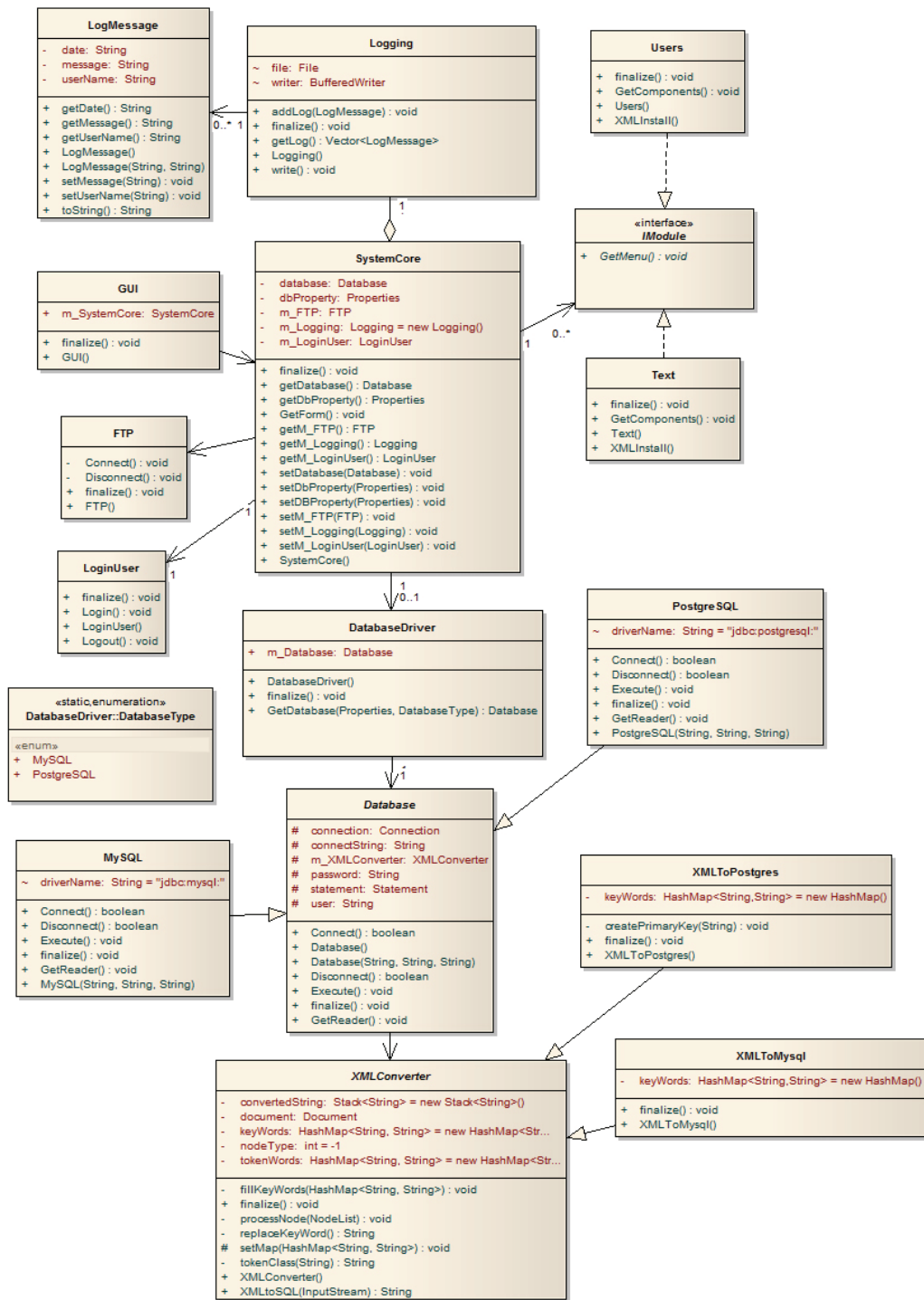
Záměrem analýzy je tvorba analytického modelu. Tento model se zaměřuje na to, co systém musí udělat, avšak nezabývá se detaily týkajícími se způsobu, jakým to udělá. Jedním z výstupů analýzy jsou právě analytické třídy. Po vytvoření analytických přecházíme ke konkrétnější definici těchto tříd a vytváříme návrhové třídy, ze kterých již implementujeme systém.

Na stranách 34 až 40 jsou prezentovány analytické a návrhové třídy jednotlivých modulů prezentující návrh tříd po analýze (analytické třídy) a návrh tříd po kompletní definici (návrhové třídy).

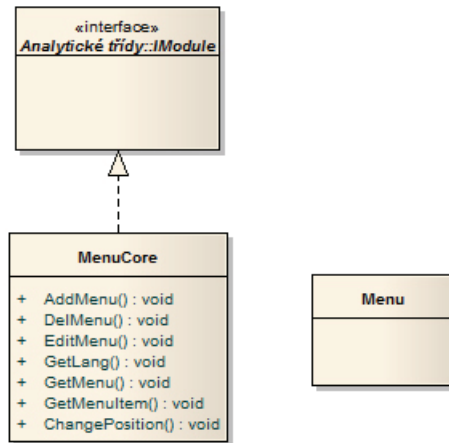


Obr. 10. Analytický třída jádra

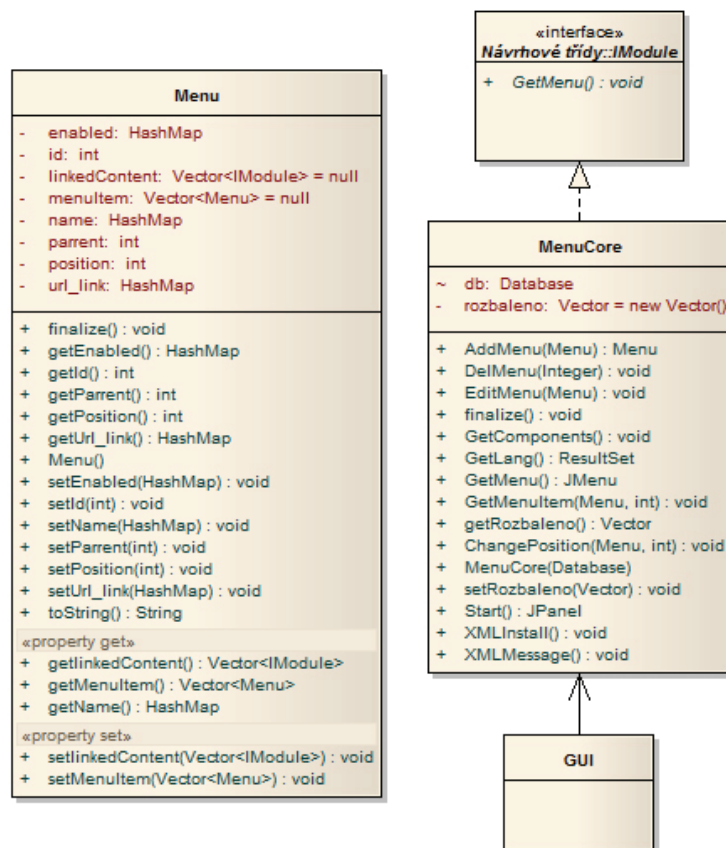
Obrázek 10 znázorňuje analytickou třídu jádra. Obrázek 11 znázorňuje návrhovou třídu systému.



Obr. 11. Návrhová třída jádra systému

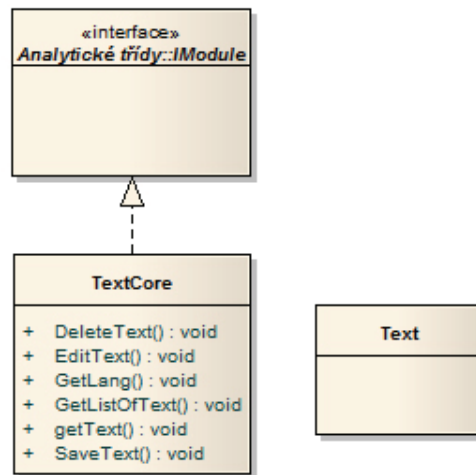


Obr.12. Analytická třída modulu Menu

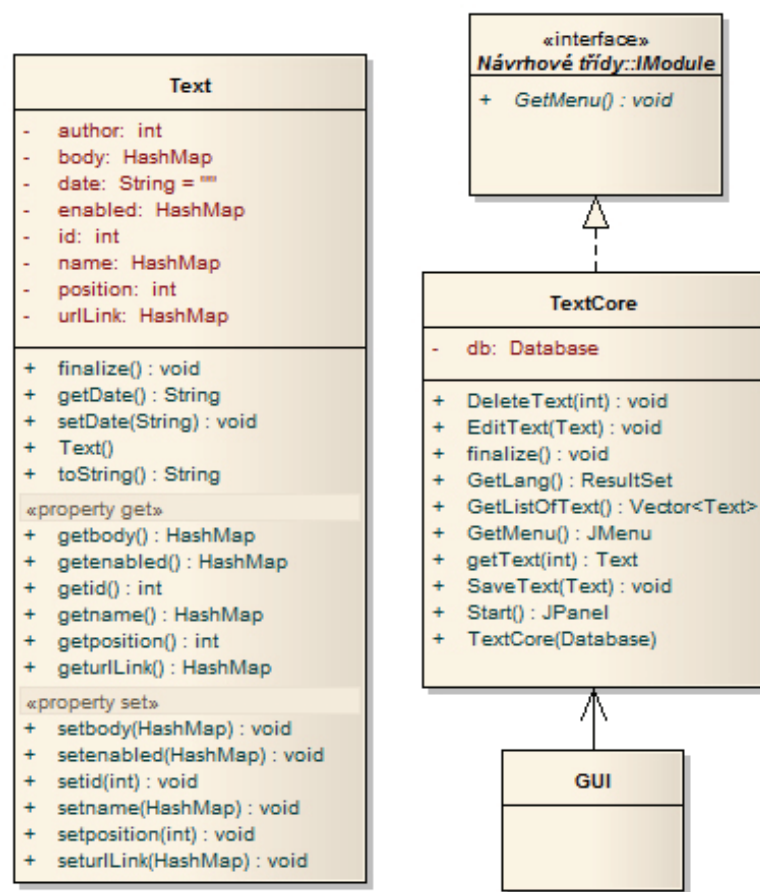


Obr. 13. Návrhová třída modulu Menu

Na obrázku 12 lze pozorovat analytický návrh třídy modulu menu. Na obrázku 13 potom návrhovou třídu modulu menu.

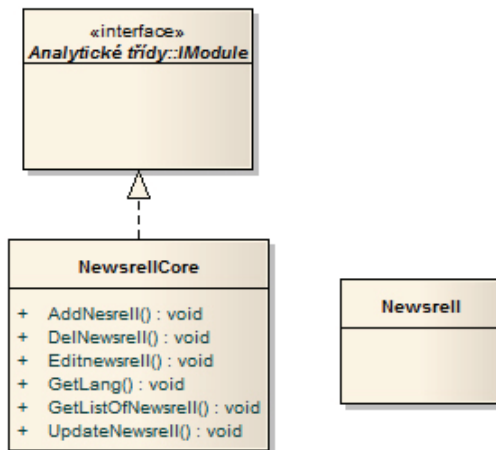


Obr. 14. Analytická třída modulu Text

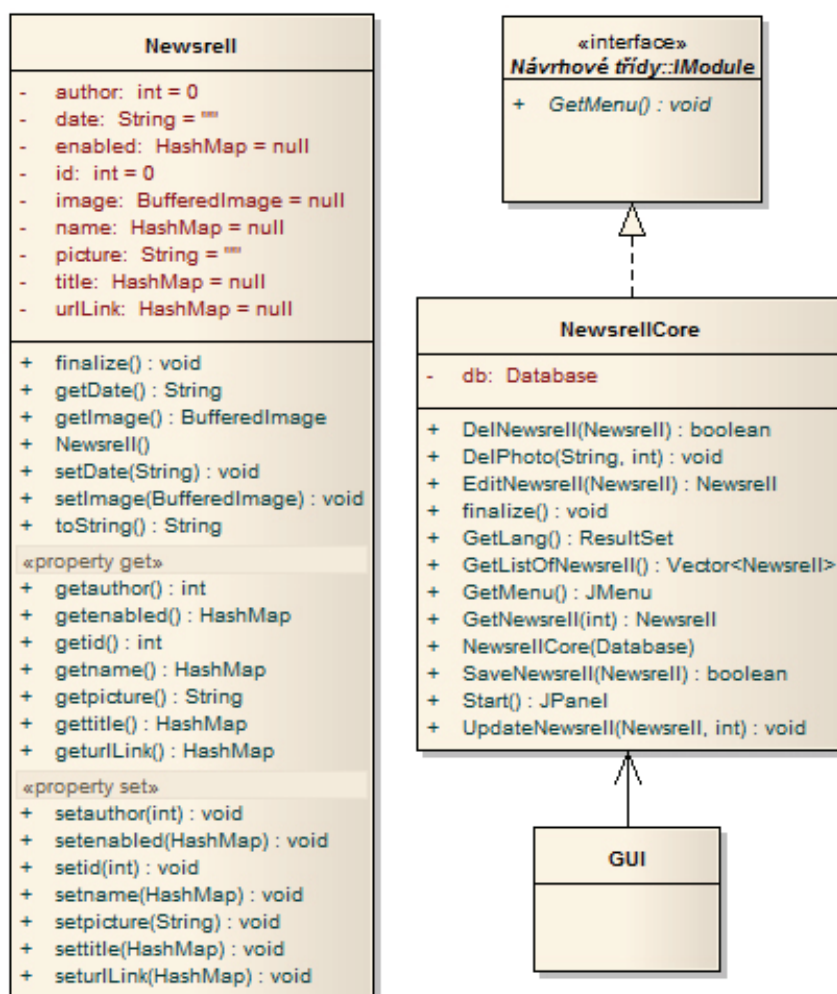


Obr. 15. Návrhová třída modulu Text

Na obrázku 14 lze pozorovat analytický návrh třídy modulu text. Na obrázku 15 potom návrhovou třídu modulu text.

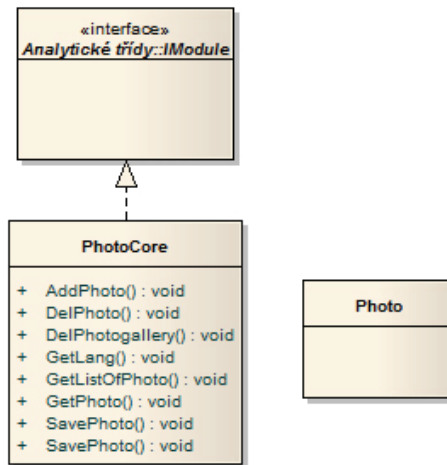


Obr. 16. Analytická třída modulu Aktuality

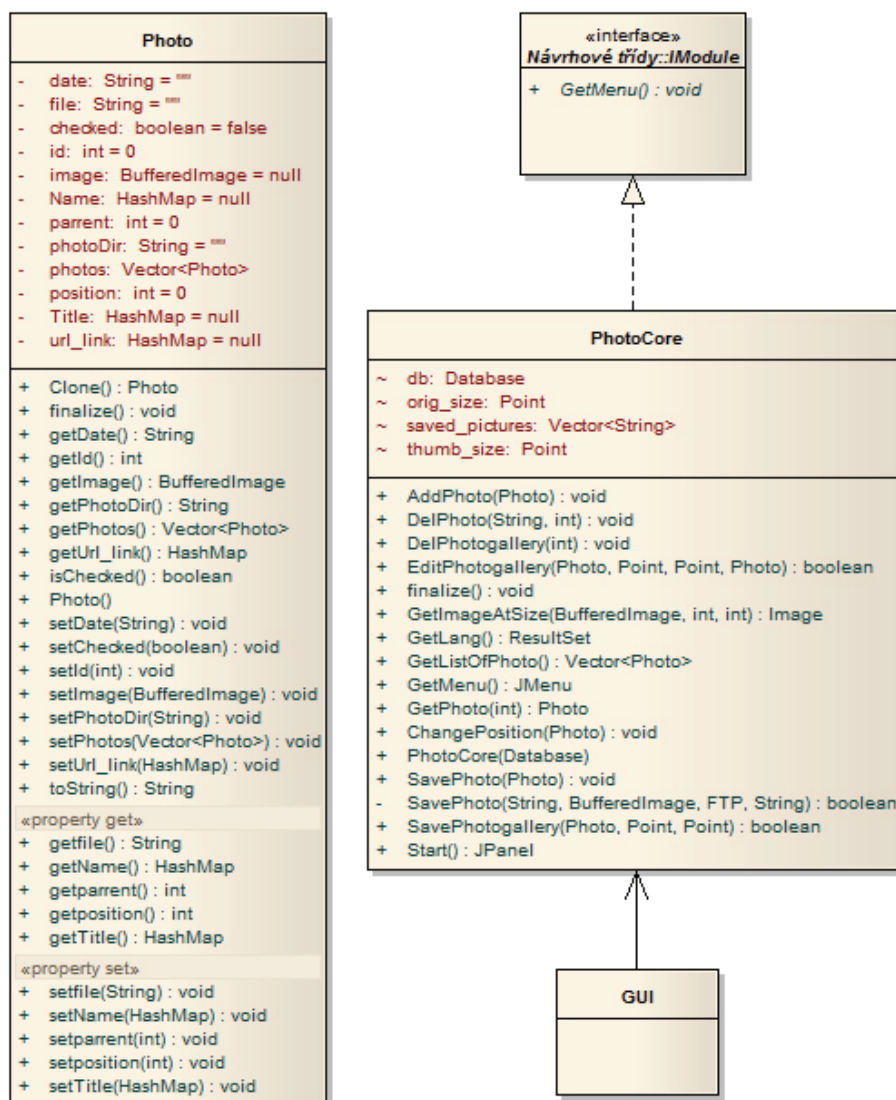


Obr. 17. Návrhová třída modulu Aktuality

Na obrázku 16 lze pozorovat analytický návrh třídy modulu aktuality. Na obrázku 17 potom návrhovou třídu modulu aktuality.

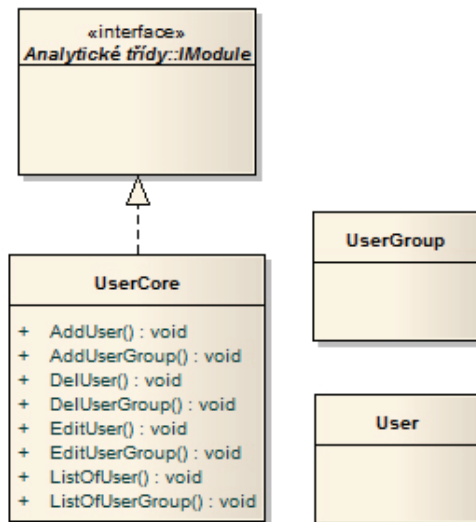


Obr. 18. Analytická třída modulu Fotogalerie

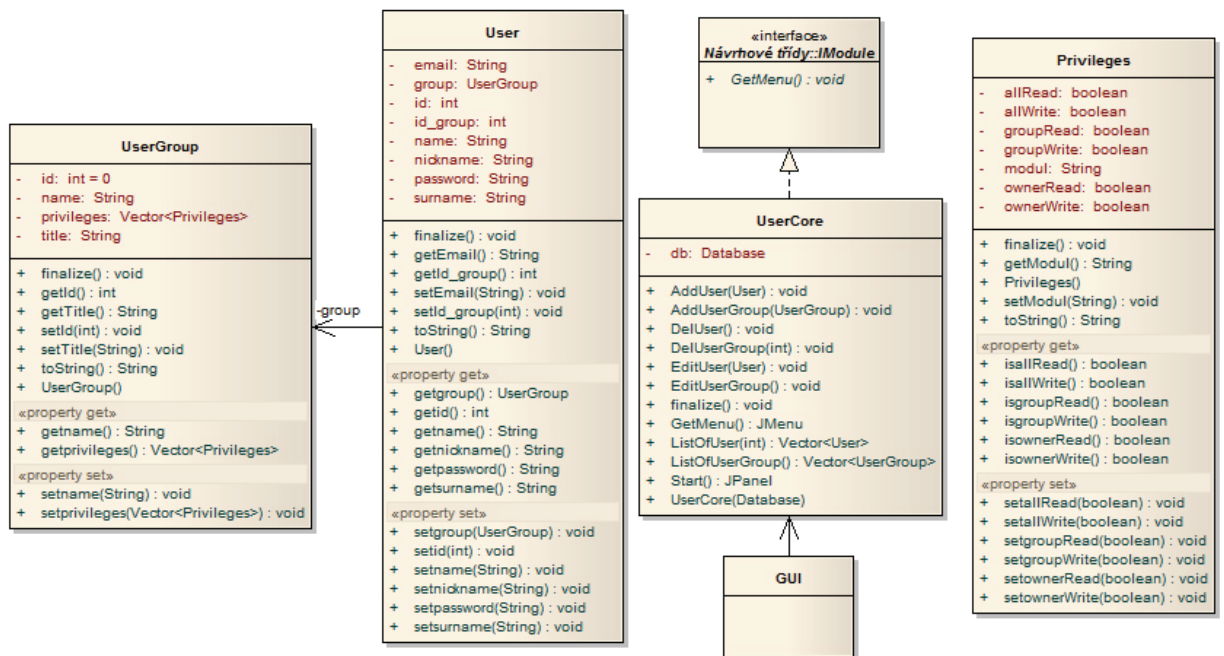


Obr. 19. Návrhová třída modulu Fotogalerie

Na obrázku 18 lze pozorovat analytický návrh třídy modulu fotogalerie. Na obrázku 19 potom návrhovou třídu modulu fotogalerie.



Obr. 20. Analytická třída modulu Uživatelé



Obr. 21. Návrhová třída modulu Uživatelé

Na obrázku 20 lze pozorovat analytický návrh třídy modulu uživatelé. Na obrázku 21 potom návrhovou třídu modulu uživatelé.

6.2. Výběr a návrh databáze

Výběr databáze

Výběr mezi typy databázemi byl vcelku jednoduchý. Při volbě typu systémové databáze bylo možné volit ze tří dnes nejpoužívanějších typů databází.

- MySQL
- PostgreSQL
- Oracle

Databázi MySQL jsem z výběru vyřadil hned v prvním kroku a to při výběru databáze dle jejích funkcí s daty. Databáze MySQL nabízí velice úzkou nabídku pro práci s daty, oproti konkurenčním databázím.

Po prvním selekci tedy zůstaly dva typy databází, Oracle a PostgreSQL. Obě databáze nabízejí velice solidní základnu funkcí pro práci s daty. Oracle je však mezi těmito dvěma typy tím silnějším, ale výběr jsem ještě podrobil dalším kritériím.

- Podpora hostingů těchto typů databází,
- Celková cena provozu databáze.

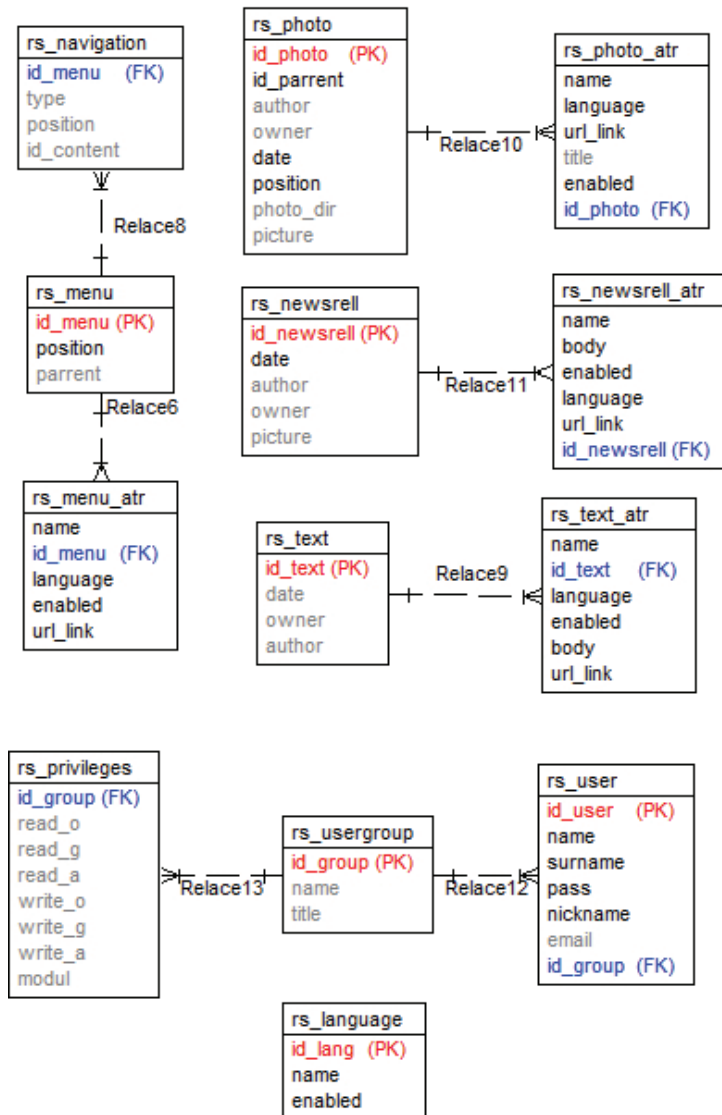
Po uvážení těchto kritérií jsem byl nucen zvolit typ databáze PostgreSQL, hosting který naše firma vlastní a na kterém běží databáze tohoto systému, disponuje PostgreSQL databází a tak by bylo zbytečné a nákladné shánět nový hosting pro provoz databáze Oracle.

V případě, že by firmajevila vyložený zájem o běh databáze na databázovém stroji Oracle, byla zde samozřejmě možnost databázi provozovat na tomto databázovém stroji.

Databázový návrh aplikace

Pro návrh databáze jsem využil softwaru „CaseStudio“ firmy QuestSoftware.

Návrh databáze lze pozorovat na obrázku 22.



Obr. 22. Návrh databáze systému

7. Administrace systému

7.1. Spuštění systému

Při prvním spuštění aplikace čeká uživatele kompletní nastavení systému. Nastavení je rozděleno do čtyř částí. První část obsahuje nastavení databázového přístupu aplikace (obr. 23). Část druhá obsahuje nastavení FTP údajů a adresy výsledné webové prezentace spravované tímto systémem (obr. 24). V části třetí

uživatel nadefinuje jazykové mutace webové prezentace (obr. 25). Část čtvrtá je pro nastavení emailu (obr. 26). Po dokončení nastavení je uživatel informován o dokončení nastavení systému a je vyzván k zalogování (autentizaci) do systému (obr. 27). V systému je automatický spuštěn prohlížeč s webovou prezentací spravovanou tímto systémem. Aplikace obsahuje tři základní položky menu.

- Soubor – sloužící k zavření aplikace,
- Moduly – sloužící ke spouštění jednotlivých modulů systému,
- Nastavení – sloužící k nastavení systému.

Nastavení databáze

1/4 Databáze

Typ databáze PostgreSQL

Hostitel

Uživatel

Heslo

Test databáze

Pokračovat Zrušit

Obr. 23. Okno prvního kroku nastavení systému

Nastavení FTP

2/4 FTP

Hostitel

Uživatel

Heslo

Test FTP

Adresa webové prezentace

http://

Test adresy

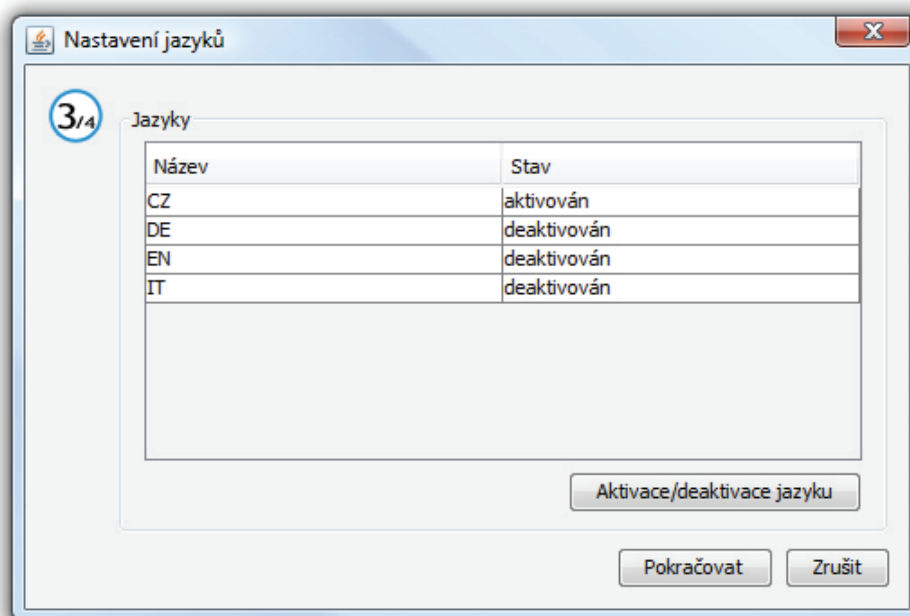
FTP adresář

www

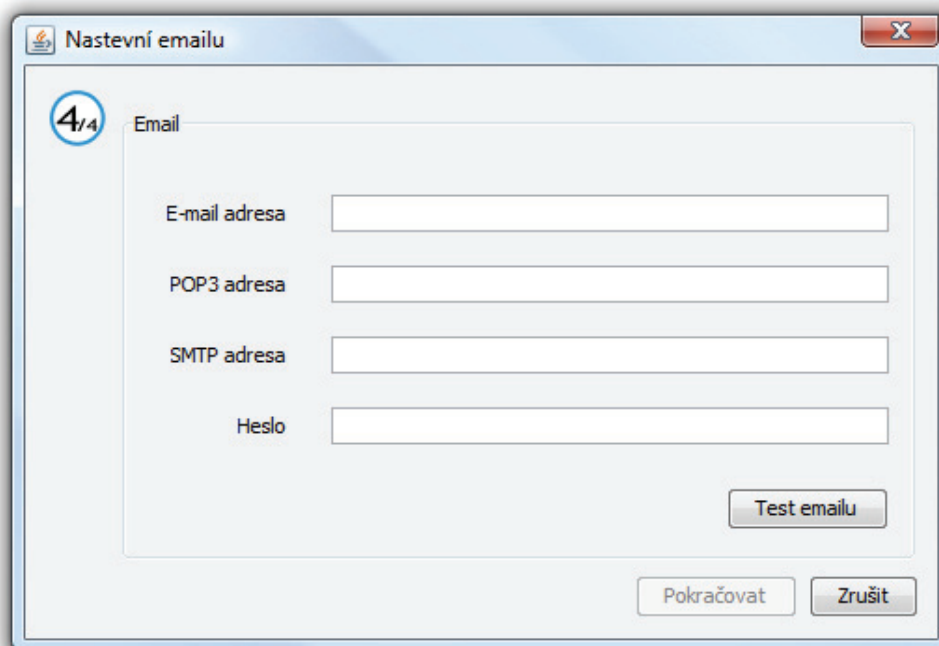
Test adresáře

Pokračovat Zrušit

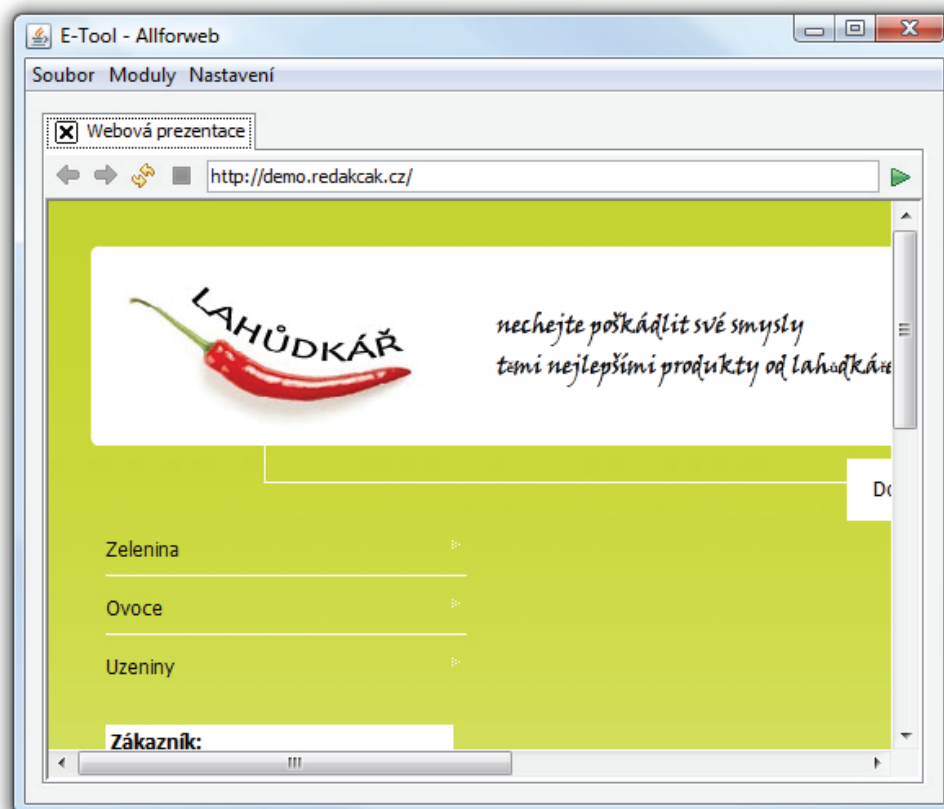
Obr. 24. Okno druhého kroku nastavení systému



Obr. 25. Okno třetího kroku nastavení systému



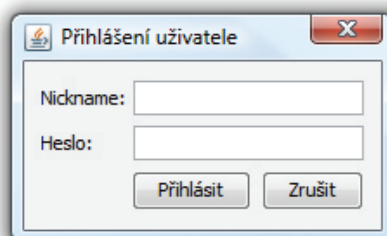
Obr. 26. Okno čtvrtého kroku nastavení systému



Obr. 27. Okno redakčního systému

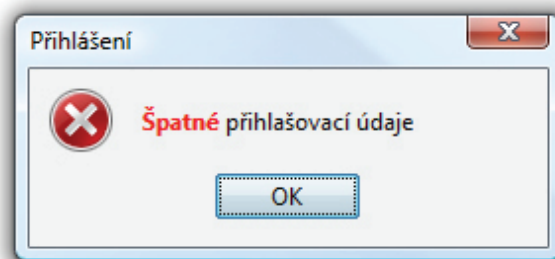
7.2. Autentizace přihlašovacích údajů uživatele

Aplikace je autentizována prostřednictvím formuláře na přihlášení (obr. 28). Uživatel má možnost buď zadat přihlašovací informace a přihlásit se do systému stisknutím tlačítka „Přihlásit“ nebo stisknutím tlačítka „Zrušit“ aplikaci opustit. Po zadání přihlašovacích údajů je vše ověřeno v databázi s údaji v tabulce uživatelů. Jsou-li údaje korektně zadány, je uživatel zalogován do systému.



Obr. 28. Dialog přihlášení uživatele

V opačném případě, tedy zadání špatných přihlašovacích údajů je uživatel upozorněn na špatné zadání (obr. 29) a po potvrzení hlášení znovu požádán o zadání přihlašovacích údajů.

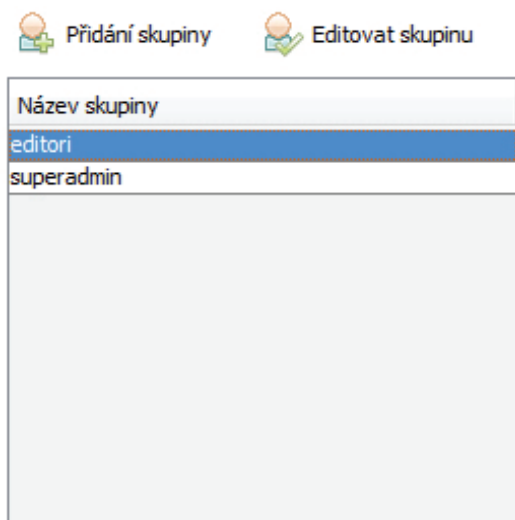


Obr. 29. Dialog upozornění špatného přihlášení

Uživatelé jsou v tomto redakčním systému spravováni užitím modulu uživatelé.

7.3. Modul Uživatelé

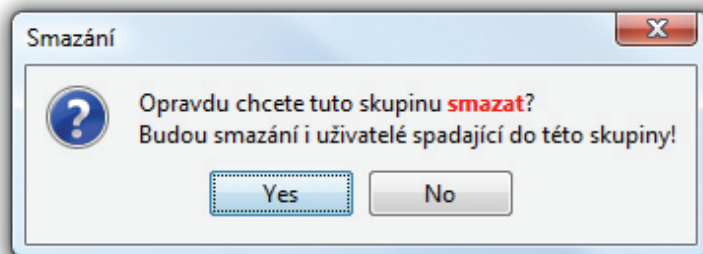
Modul uživatelů slouží ke správě uživatelů systému. Modul spravuje dvě základní skupiny a to uživatelské skupiny, které definují práva skupiny a samotné uživatele, kteří disponují právě jedním typem dané uživatelské skupiny. Uživatelé skupiny jsou vypsané v tabulce skupin (obr. 30).



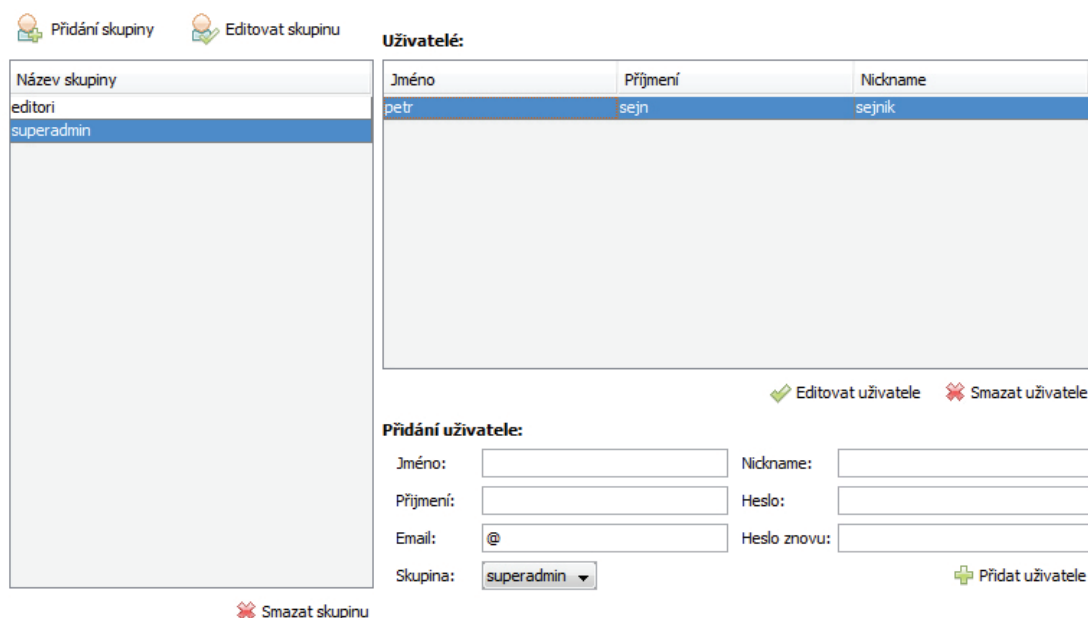
Obr. 30. Tabulka výpisu uživatelských skupin

Po kliknutí na danou skupinu se pod výpisem uživatelských skupin zobrazí ikona kříže sloužící k mazání označené uživatelské skupiny, která po kliknutí zobrazí potvrzovací dialog (obr. 31), a na pravé straně vypíše panel na správu uživatelů spadajících pod danou uživatelskou skupinu (obr. 32).

Kliknutím na tlačítko s názvem „Přidání skupiny“ se vyvolá okno pro přidání uživatelské skupiny (obr. 33). Kliknutím na tlačítko s názvem „Editace skupiny“ se vyvolá okno pro editaci uživatelské skupiny (obr. 34).

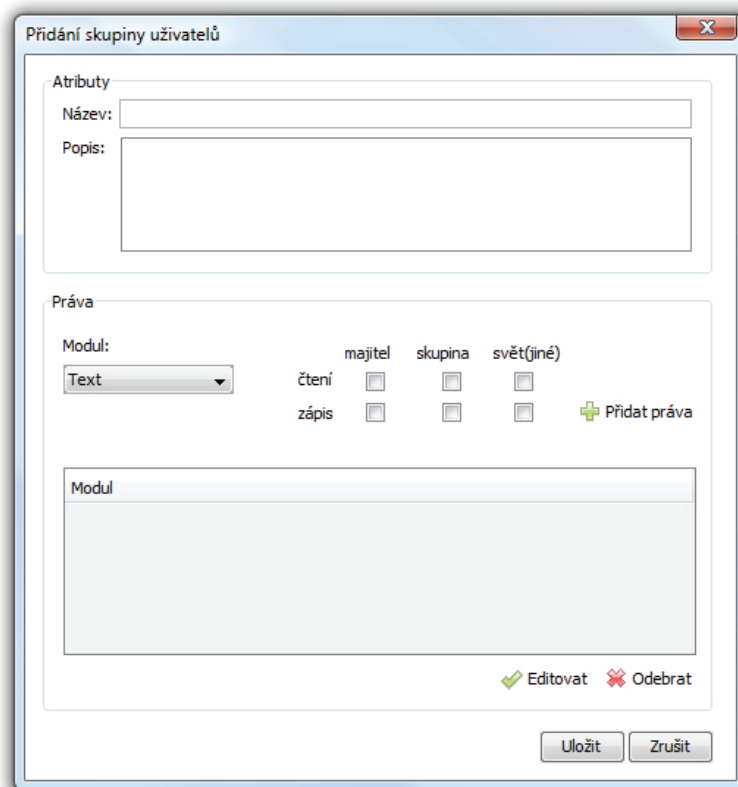


Obr. 31. Dialog na potvrzení smazání uživatelské skupiny



Obr. 32. Panel uživatelů

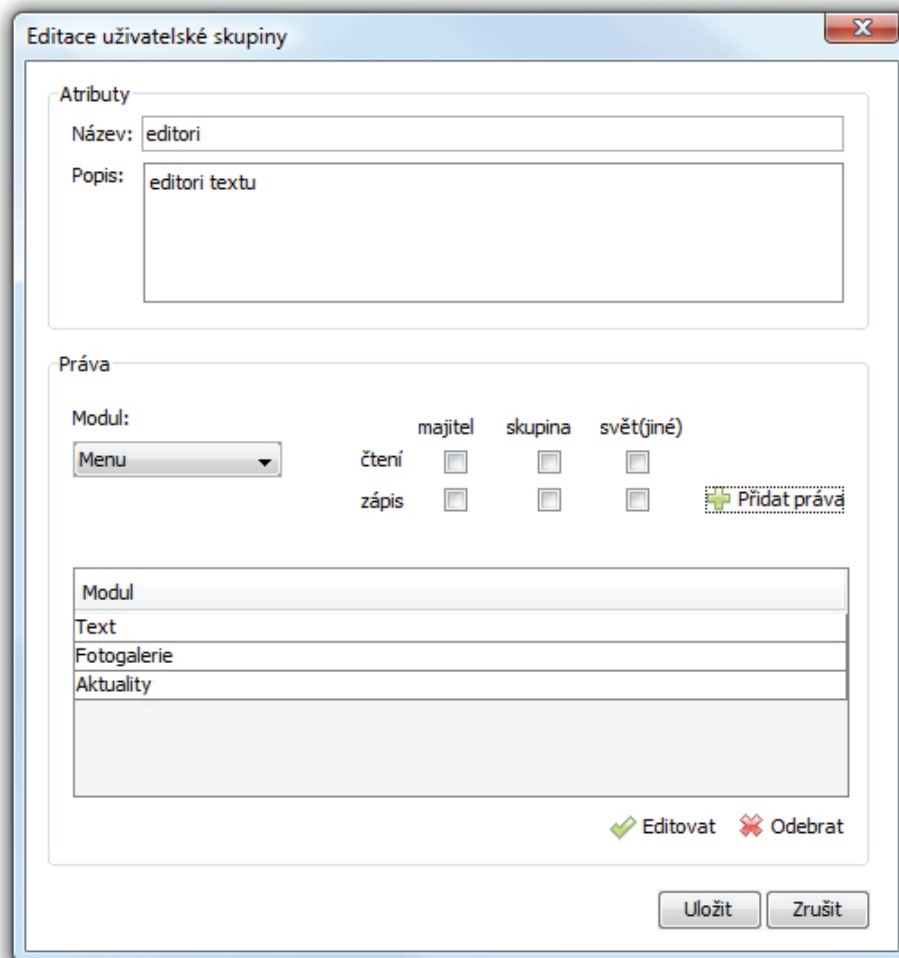
Panel uživatelů obsahuje tabulku pro výpis daných uživatelů, pod kterou jsou zobrazeny tlačítka k editaci a mazání uživatelů. Po kliknutí na tlačítko s názvem „Editovat uživatele“ se zobrazí okno k editaci uživatele (obr. 35). Kliknutím na tlačítko s názvem „Smazat uživatele“ se zobrazí potvrzovací dialog ke smazání daného uživatele (obr. 36).



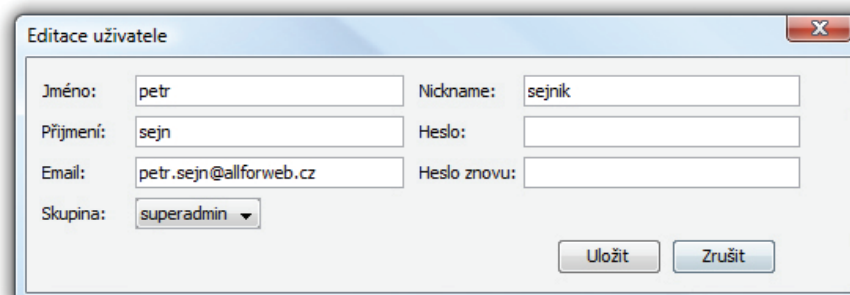
Obr. 33. Okno pro přidání uživatelské skupiny

Okno k přidání nové uživatelské skupiny obsahuje dvě základní sekce. První obsahuje atributy uživatelské skupiny, což představuje název a popis uživatelské skupiny. V části druhé jsou editována práva uživatelské skupiny.

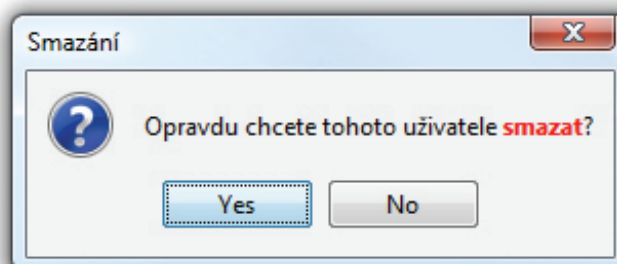
Práva přidáme výběrem určitého modulu z listu modulů vypsáno na levé straně, zaškrtnutím určitých práv k modulu a stisknutím tlačítka s názvem „Přidat práva“. Práva se tímto přidají do tabulky a přidáný modul zmizí z výběru modulů k přidání. Pod tabulkou jsou dále dvě tlačítka sloužící k editaci a mazání již přidáných práv v tabulce. Kliknutím na tlačítko s názvem „Editovat“ se vyvolá nové okno sloužící k editaci daných práv (obr. 37). Kliknutím na tlačítko s názvem „Smazat“ se smažou práva z tabulky.



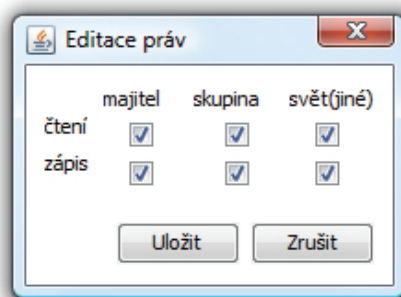
Obr. 34. Okno editace uživatelské skupiny



Obr. 35. Okno editace uživatele



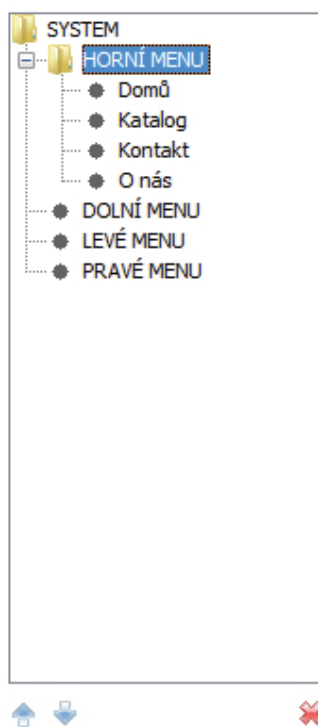
Obr. 36. Dialog na potvrzení smazání uživatele



Obr. 37. Okno editace práv

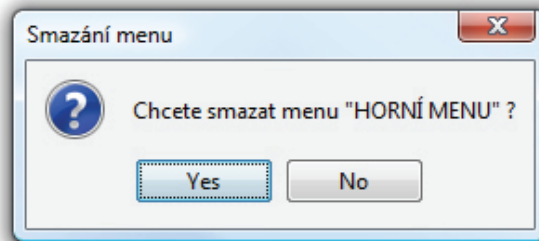
7.4. Modul Menu

Modul menu slouží k vytváření menu webové prezentace a k napojování vytvořeného obsahu na jednotlivé položky menu prezentace. Menu nabízí čtyři typy menu a to: horní, dolní, levé a pravé (obr. 38).



Obr. 38. Tabulka výpisu menu

Pod výpisem menu jsou umístěny šipky sloužící ke změně pozice položek menu a ikona kříže k jejich mazání. Při mazání menu je uživatel požádán o potvrzení smazání položky menu (obr. 39).

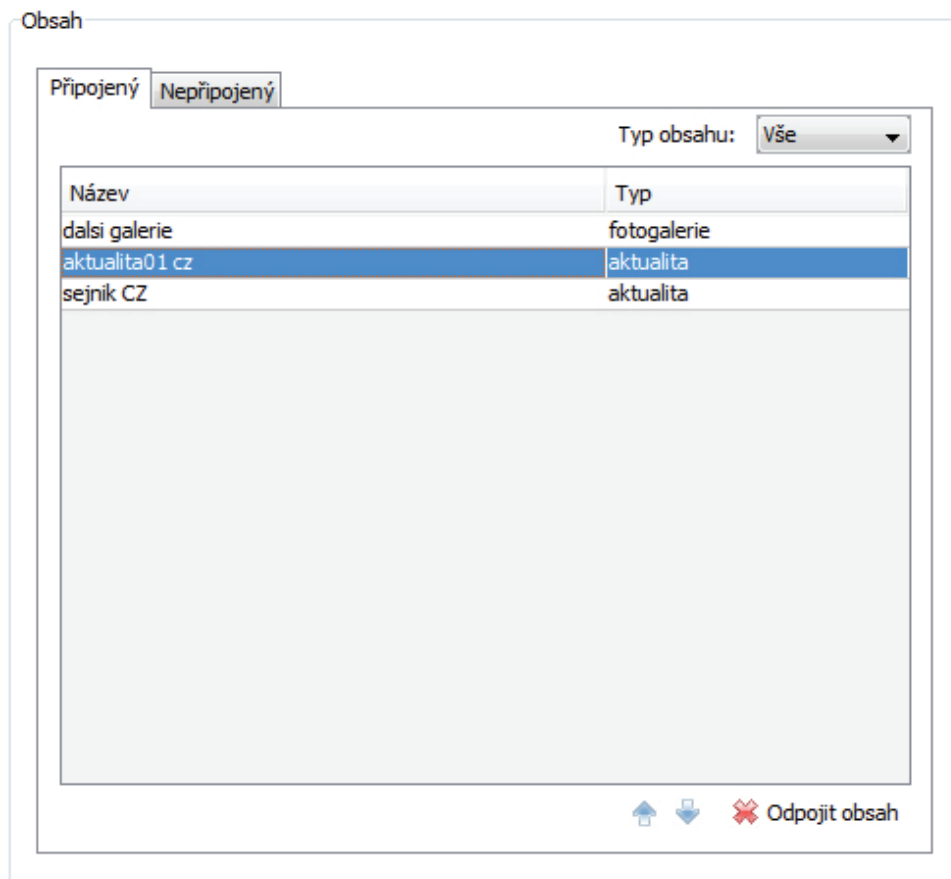


Obr. 39. Dialog mazání menu

Kliknutím na jednu z položek menu se vedle výpisu menu vypíše editační panel k editaci atributů položky menu a přidání nové podpoložky menu aktivní položky menu (obr. 40) a panel na připojování či odpojování obsahu na položku menu (obr. 41).

Obr. 40. Panel atributů menu

Obsah lze zadávat ve více jazycích, které jsou nadefinovány v nastavení aplikace.



Obr. 41. Panel obsahu

Panel obsahu obsahuje dvě záložky. Dle jejich pojmenování je zřejmé že pod záložkou pojmenovanou „Připojený“ uživatel nalezne připojený obsah na danou položku menu. Podobně pod záložkou pojmenovanou „Nepřipojený“ uživatel nalezne doposud nepřipojený obsah na danou položku menu. Uživatel má v obou záložkách možnost výpisu obsahu dle jeho typu nebo všeho.

Dále lze kliknutím na danou položku obsahu obsah jednoduše pozicovat kliknutím na jednu ze šipek v dolní části panelu. Křížem potom lze daný obsah odpojit (v případě připojeného obsahu) nebo zaškrtnutím (v případě nepřipojeného obsahu) obsah připojit.

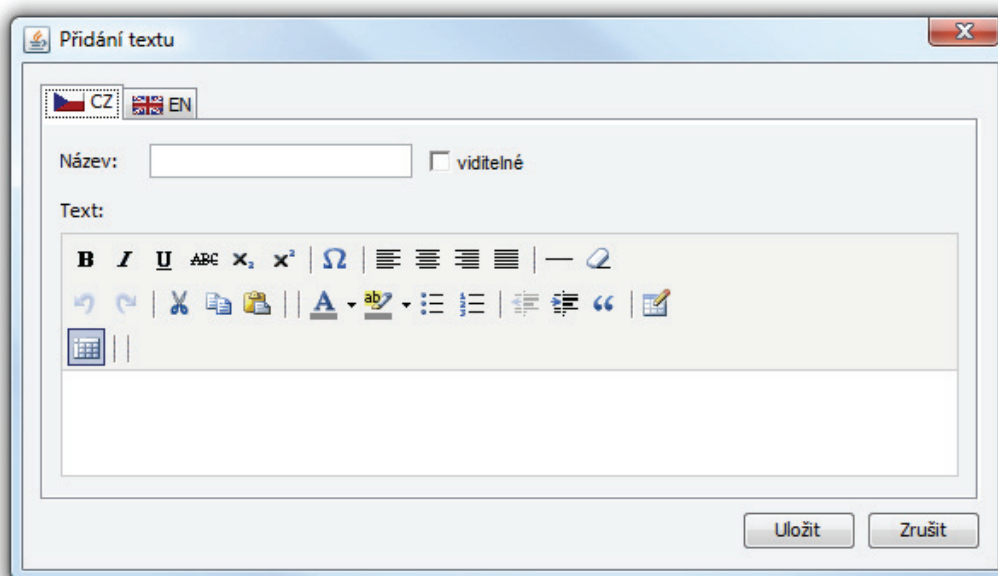
7.5. Modul Text

Modul text slouží ke správě textů zobrazovaných na webové prezentaci. Všechny texty jsou vypsány v tabulce (obr. 42).

Název	Autor	Čas vytvoření
kontakt		24.4.2009
O společnosti		24.4.2009

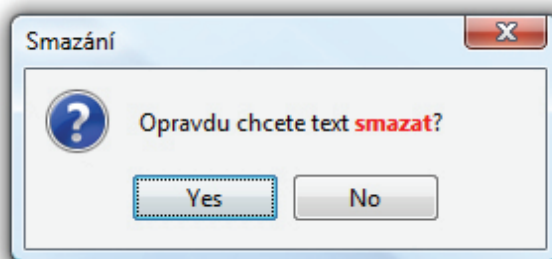
Obr. 42. Okno výpisu textů

Kliknutím na název sloupce lze texty v tabulce jednoduše řadit dle daného sloupce. Tlačítko „Přidání textu“ vyvolá nové okno pro přidání textu (obr. 43). Tlačítko „Smazání textu“ vyvolá dialogové okno na potvrzení smazání daného (obr. 44). Tlačítko „Editace textu“ vyvolá nové okno pro editaci vybraného textu (obr. 45).

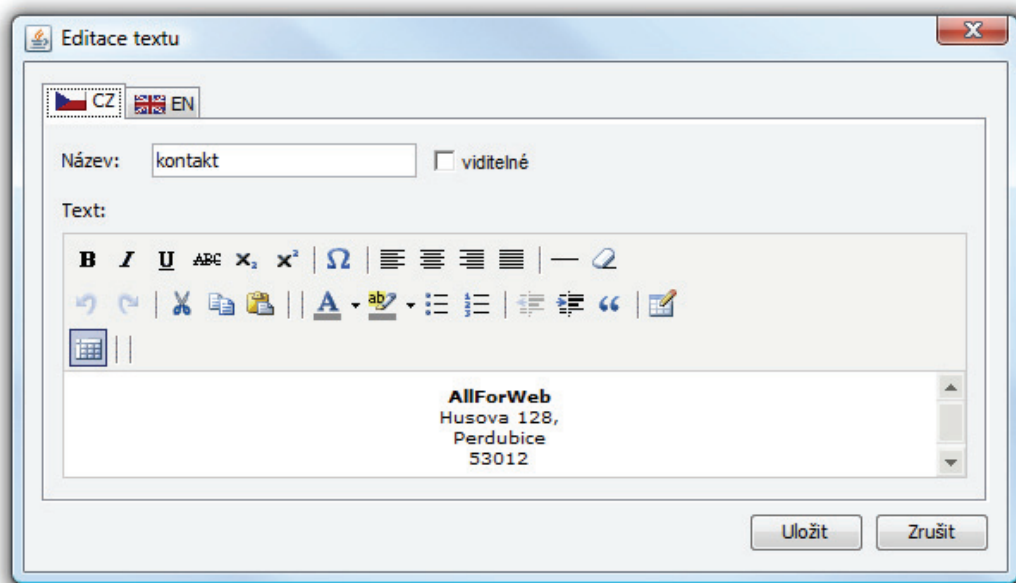


Obr. 43. Okno přidání textu

Okno obsahuje opět jazykové záložky pro zadání textu ve více jazycích. V první části okna je název textu a v části druhé jednoduchý wysiwyg editor pro snadnou editaci textu.



Obr. 44. Dialog mazání textu



Obr. 45. Okno editace textu

7.6. Modul Fotogalerie

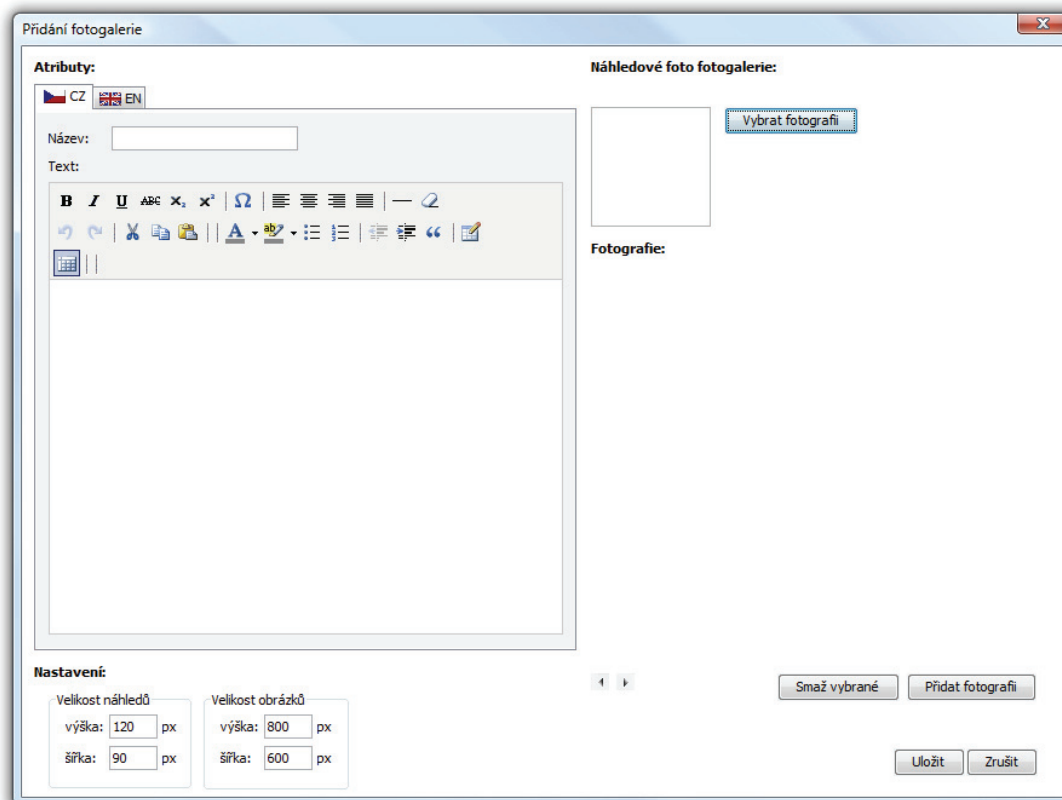
Modul Fotogalerie slouží ke správě fotogalerií. Fotogalerie jsou zobrazeny v panelu (obr. 46).



Obr. 46. Okno přehledu fotogalerie

Tlačítko „Přidání fotogalerie“ zobrazí nové okno pro přidání nové fotogalerie (obr. 47). Jednotlivé fotogalerie lze editovat kliknutím na ikonku klíče pod obrázkem

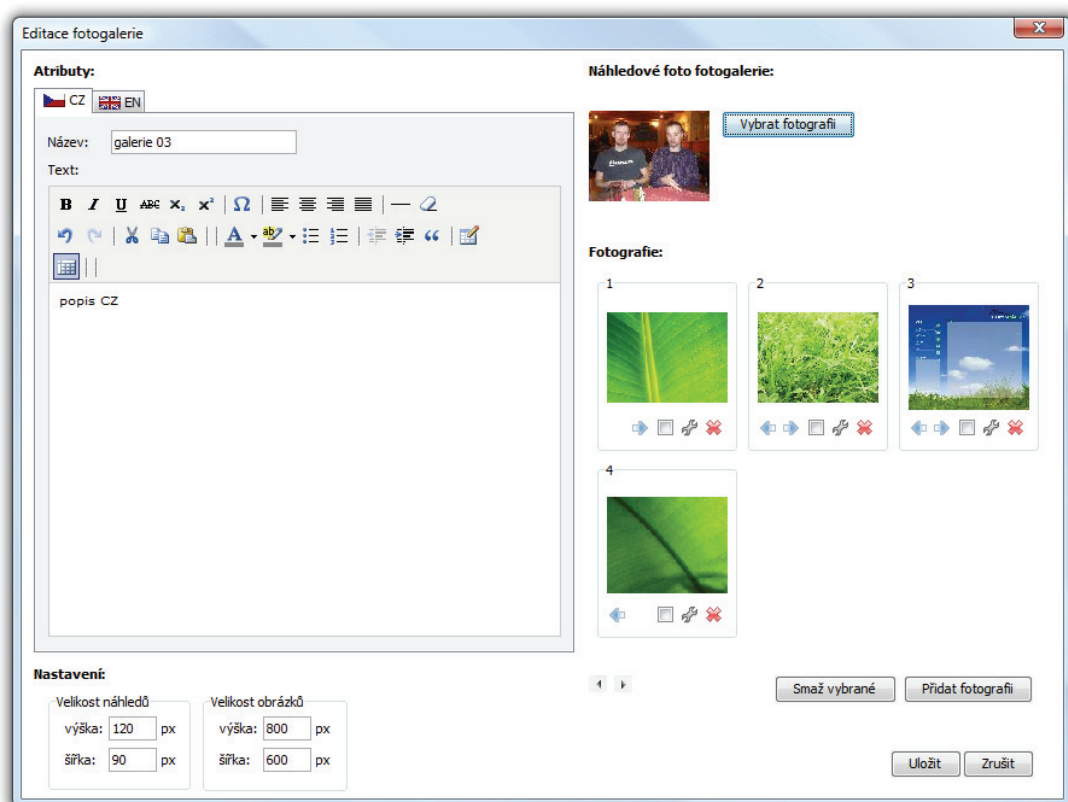
každé fotogalerie, která zobrazí nové okno na editaci fotogalerie (obr. 48). Fotogalerii lze smazat kliknutím na ikonu kříže a potvrzením vyvolaného dialogu (obr. 49).



Obr. 47. Okno pro přidání fotogalerie

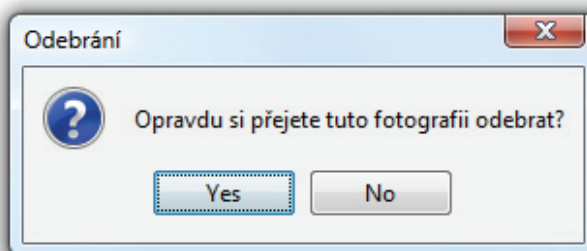
Levá strana okna na přidání fotogalerie slouží k zadání atributů fotogalerie. Horní část na zadání názvu a popisu galerie, spodní část na nastavení velikosti obrázků fotogalerie.

Část pravá slouží k nastavení náhledové fotografie, pod kterou bude fotogalerie prezentována a fotografií, které bude fotogalerie obsahovat. Kliknutím na tlačítko „Vybrat fotografii“ se otevře okno pro výběr fotogalerie (obr. 50). Kliknutím na tlačítko „Přidat fotografii“ se vyvolá okno na přidání fotografie (obr. 51).



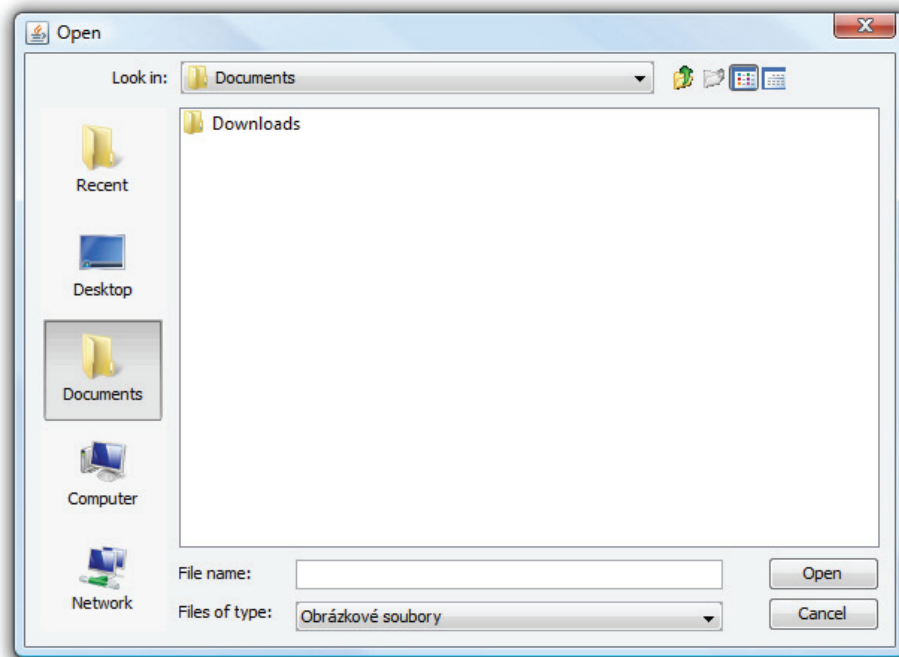
Obr. 48. Okno editace fotogalerie

Každá fotografie v části nazvané „Fotografie“ obsahuje až 5 ovládacích prvků (obr. 52). Šipky umožňující umístění fotografií ve fotogalerii, zaškrťovací pole umožňující mazání více položek najednou nebo výběr náhledové fotografie, v případě, že není náhledová fotografie zvolena, ikonu klíče umožňující editaci fotografie a ikonu kříže sloužící k mazání dané fotografie.



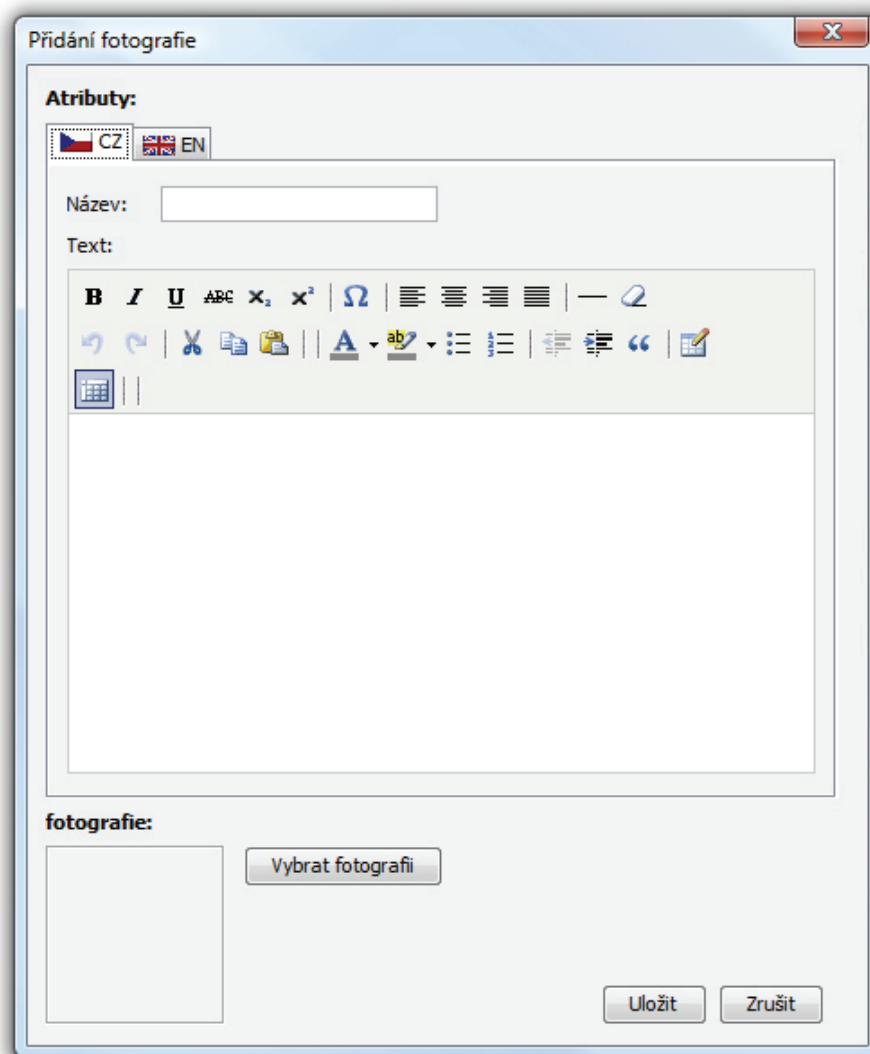
Obr. 49. Okno potvrzovací dialogu

Kliknutím na tlačítko „Yes“ uživatel smaže danou fotogalerii. Kliknutím na tlačítko „No“ uživatel zruší mazání dané fotogalerie a vrátí se na výpis fotogalerií.



Obr. 50. Okno pro dialog na výběr fotografie

Okno na výběr fotografií je klasický „Open Dialog“, pouze s tou změnou, že přes tento dialog lze vybírat pouze obrázkové soubory.



Obr. 51. Okno na přidání fotografie

Okno na přidání fotografie obsahuje opět dvě části. První část slouží k nastavení atributů dané fotografie, které jsou zadávány ve více jazycích dle nastavení systému.

Část druhá slouží k výběru fotografie, stejně jak je tomu u výběru náhledové fotografie.



Obr. 52. Panel fotografie

7.7. Modul Aktuality

Modul Aktualit slouží ke správě aktualit webové prezentace. Aktuality jsou vypisovány v tabulce (obr. 53), ve které je lze třídit kliknutím na název sloupce.

+ Přidat aktualitu ✓ Editovat aktualitu ✖ Smazat aktualitu

Název	Datum
aktualita 01	5.4.2009
aktualita 02	5.4.2009
aktualita 03	6.4.2009

Obr. 53. Okno výpisu aktualit

Kliknutím na tlačítko s názvem „Přidat aktualitu“ se zobrazí okno pro přidání aktuality (obr. 54). Kliknutím na tlačítko s názvem „Editovat aktualitu“ vyvolá okno k editaci označené aktuality v tabulce (obr. 55). Kliknutím na tlačítko s názvem „Smazat aktualitu“ se zobrazí potvrzovací dialog ke smazání vybrané tabulky (obr. 56).

Přidání aktuality

CZ EN

Název: viditelné

Text:

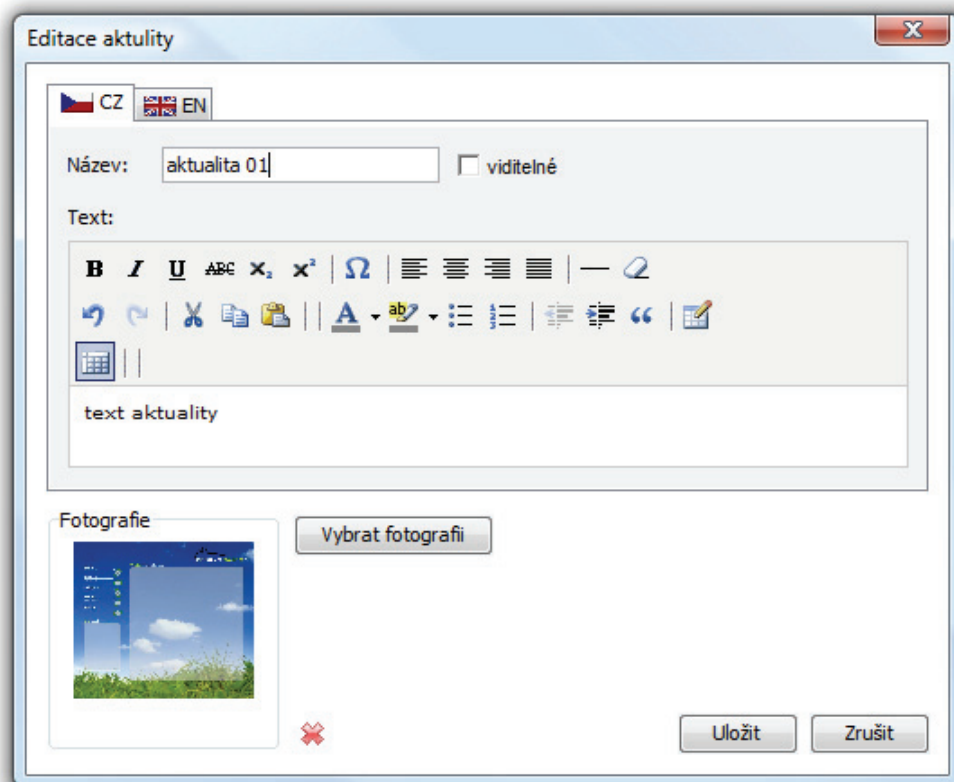
B *I* U ABC x₂ x² Ω | [List Icons] | [Undo] [Redo]

[Copy] [Paste] [Link] [Image] [Table] [Table of Contents] [Page Number] [Page Count] [Page Size] [Page Orientation] [Page Color] [Page Background Image] [Page Background Color] [Page Background Image Size] [Page Background Color Size] [Page Background Image Position] [Page Background Color Position] [Page Background Image Position Size] [Page Background Color Position Size]

Fotografie

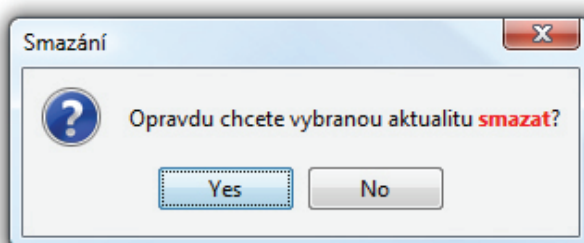
Obr. 54. Okno na přidání aktuality

Okno na přidání aktuality obsahuje dvě části. První část obsahuje zadávací pole k zadání názvu a textu aktuality, které lze zadávat ve více jazycích dle nastavení systému. Druhá část obsahuje pole na zadání obrázku aktuality. Kliknutím na tlačítko s názvem „Vybrat fotografii“ se vyvolá dialog na výběr fotografie.



Obr. 55. Okno k editaci aktuality

V případě že je fotografie u aktuality vybrána je možnost jejího odstranění kliknutím ikonu kříže na pravé straně vedle fotografie nebo její změny výběrem fotografie nové kliknutím na tlačítko s názvem „Vybrat fotografii“.



Obr. 56. Okno - potvrzovací dialog ke smazání aktuality

8. Použité technologie

Jak je již v názvu práce uvedeno, systém byl implementován v jazyce JAVA. K vývoji aplikace v tomto jazyce jsem využil softwaru NetBeans firmy SunMiscrosystems. Systém jsem navrhl v jazyce UML a to v softwaru Enterprise Architect firmy Spark Systems. Databáze byla navrhnutá v softwaru Case Studio firmy Quest Software a v průběhu ladění aplikace testována v softwaru PgAdmin. Jednotlivé technologie jsou popsány v teoretické části této práce.

9. Závěr

Díky návrhu systému modelovanému v jazyce UML se vývoj systému značně zjednodušil a ve velké míře zefektivnil tvorbu eliminací zbytečných změn struktury aplikace a s tím souvisejících chyb.

Implementace systému v jazyce JAVA umožnila provoz aplikace na všech platformách podporovaných systémem JAVA, což bylo hlavním cílem této práce.

Z důvodu časové náročnosti vývoje systému nebyl systém 100% odladěn, a tak dalším krokem bude maximální odladění (otestování všech vstupů a ošetření výjimek) a spuštění testovacího provozu aplikace.

Redakční systém, který byl vytvořen, zahrnuje všechny dnešní požadavky a trendy v oblasti těchto systémů. Redakční systém byl vytvořen s důrazem na možnosti využití a jeho hlavními výhodami tedy jsou:

- multiplatformost,
- modulárnost,
- efektivní správa webové prezentace prostřednictvím jednoduché administrace,
- bezpečnost díky naimplementování aplikačních oken.

Hlavní nevýhodou vytvořeného systému zůstává aktuálně nekompletní otestování z časových důvodů. Proto bude další vývoj systému zaměřen především na testování.

Výhledem do budoucnosti bude tedy především odladění aplikace a dalším krokem bude vývoj dalších modulů a případně nabídka tohoto systému na komerční bázi.

10. Seznam literatury

- [1] WIKIPEDIA. Content management [online]. Aktualizováno 22.5.2009
[cit.: 2.4. 2009]. Dostupné z:
<http://en.wikipedia.org/wiki/Content_management>
- [2] WIKIPEDIA. CMS [online] Dostupné z WWW:
<http://en.wikipedia.org/wiki/Content_management_system>
- [3] WIKIPEDIA. WCMS [online] Dostupné z WWW:
<http://en.wikipedia.org/wiki/Web_content_management>
- [4] WEB4COMPANY [online] Dostupné z WWW:
<<http://www.web4company.cz>>
- [5] *redakční systémy* [online] Dostupné z WWW: <<http://www.artic-studio.net/webove-stranky/redakcni-system/>>
- [6] ARLOW, Jim; NEUSTADT, Ila. *UML a unifikovaný proces vývoje aplikace*. 1. vyd., v Computer Press Brno, 2003. 387 s. ISBN 80-7226-947-X.
- [7] WIKIPEDIA. JAVA [online] Dostupné z WWW:
<<http://cs.wikipedia.org/wiki/Java>>
- [8] WIKIPEDIA. .NET [online] Dostupné z WWW:
<<http://cs.wikipedia.org/wiki/.NET>>
- [9] WIKIPEDIA. PHP [online] Dostupné z WWW:
<<http://cs.wikipedia.org/wiki/PHP>>
- [10] WIKIPEDIA. ORACLE [online] Dostupné z WWW:
<<http://cs.wikipedia.org/wiki/Oracle>>
- [11] WIKIPEDIA. PostgreSQL [online] Dostupné z WWW:
<<http://cs.wikipedia.org/wiki/PostgreSQL>>

- [12] WIKIPEDIA. MySQL [online] Dostupné z WWW:
<<http://cs.wikipedia.org/wiki/MySQL>>
- [13] *obrázek RS* [online] Dostupné z WWW:
<http://www.unixreview.com/documents/s=9900/ur0510h/ur0510h_f1.jpg>
- [14] *obrázek RS* [online] Dostupné z WWW:
<<http://www.claromentis.com/images/diagrams/publish.jpg>>
- [15] *obrázek CMS* [online] Dostupné z WWW:
<<http://buytaert.net/images/drupal/sitepoint-cms-usage.jpg>>
- [16] *tabulka kódování* [online] Dostupné z WWW:
<<http://www.postgresql.org/docs/8.3/static/pgcrypto.html>>

11. Seznam obrázků

- obrázek 1 - Integrace CMS - zdroj: <http://www.unixreview.com>
- obrázek 2 - Redakční systém - zdroj: <http://en.wikipedia.org>
- obrázek 3 - Redakční systém s moduly - zdroj: <http://en.wikipedia.org>
- obrázek 4 - Využití redakčních systémů - zdroj: <http://buytaert.net>
- obrázek 5 - UseCase modulu menu - zdroj: vlastní
- obrázek 6 - UseCase modulu text - zdroj: vlastní
- obrázek 7 - UseCase modulu aktuality - zdroj: vlastní
- obrázek 8 - UseCase modulu fotogalerie - zdroj: vlastní
- obrázek 9 - UseCase modulu uživatelé - zdroj: vlastní
- obrázek 10 - Analytická třída jádra systému - zdroj: vlastní
- obrázek 11 - Návrhová třída jádra systému - zdroj: vlastní
- obrázek 12 - Analytická třída modulu menu - zdroj: vlastní
- obrázek 13 - Návrhová třída modulu menu - zdroj: vlastní
- obrázek 14 - Analytická třída modulu text - zdroj: vlastní
- obrázek 15 - Návrhová třída modulu text - zdroj: vlastní
- obrázek 16 - Analytická třída modulu aktuality - zdroj: vlastní
- obrázek 17 - Návrhová třída modulu aktuality - zdroj: vlastní
- obrázek 18 - Analytická třída modulu fotogalerie - zdroj: vlastní
- obrázek 19 - Návrhová třída modulu fotogalerie - zdroj: vlastní
- obrázek 20 - Analytická třída modulu uživatelé - zdroj: vlastní
- obrázek 21 - Návrhová třída modulu uživatelé - zdroj: vlastní
- obrázek 22 - Návrh databáze systému- zdroj: vlastní
- obrázek 23 - Nastavení systému 1. krok - zdroj: vlastní
- obrázek 24 - Nastavení systému 2. krok - zdroj: vlastní
- obrázek 25 - Nastavení systému 3. krok - zdroj: vlastní
- obrázek 26 - Nastavení systému 4. krok - zdroj: vlastní
- obrázek 27 - Redakční systém - zdroj: vlastní
- obrázek 28 - Přihlášení uživatele - zdroj: vlastní
- obrázek 29 - Špatné přihlášení - zdroj: vlastní
- obrázek 30 - Výpis uživatelských skupin - zdroj: vlastní

- obrázek 31 - Potvrzovací dialog - zdroj: vlastní
- obrázek 32 - Panel uživatelů - zdroj: vlastní
- obrázek 33 - Přidání uživatelské skupiny - zdroj: vlastní
- obrázek 34 - Editace uživatelské skupiny - zdroj: vlastní
- obrázek 35 - Editace uživatele - zdroj: vlastní
- obrázek 36 - Potvrzení smazání - zdroj: vlastní
- obrázek 37 - Editace práv - zdroj: vlastní
- obrázek 38 - Výpis menu - zdroj: vlastní
- obrázek 39 - Mazání menu - zdroj: vlastní
- obrázek 40 - Panel atributů menu - zdroj: vlastní
- obrázek 41 - Panel obsahu - zdroj: vlastní
- obrázek 42 - Výpis textů - zdroj: vlastní
- obrázek 43 - Okno přidání textu - zdroj: vlastní
- obrázek 44 - Potvrzovací dialog - zdroj: vlastní
- obrázek 45 - Editace textu - zdroj: vlastní
- obrázek 46 - Výpis fotogalerií - zdroj: vlastní
- obrázek 47 - Přidání fotogalerie - zdroj: vlastní
- obrázek 48 - Editace fotogalerie - zdroj: vlastní
- obrázek 49 - Potvrzovací dialog - zdroj: vlastní
- obrázek 50 - Výběr fotografie - zdroj: vlastní
- obrázek 51 - Okno přidání fotogalerie - zdroj: vlastní
- obrázek 52 - Fotogalerie - zdroj: vlastní
- obrázek 53 - Výpis aktualit - zdroj: vlastní
- obrázek 54 - Okno přidání aktuality - zdroj: vlastní
- obrázek 55 - Editace aktuality - zdroj: vlastní
- obrázek 56 - Potvrzovací dialog - zdroj: vlastní

- tabulka 1 - Kódovací algoritmy - zdroj: <http://www.postgresql.org>

ÚDAJE PRO KNIHOVNICKOU DATABÁZI

Název práce	Implementace redakčního systému v jazyce JAVA
Autor práce	Bc. Petr Šejn
Obor	Informační technologie
Rok obhajoby	2009
Vedoucí práce	Ing. Jana Holá, Ph.D
Anotace	<p>Tato práce je zaměřena na implementaci redakčního systému, jako systému pro správu webového obsahu, v jazyce JAVA. V teoretické části jsou definovány hlavní pojmy z oblasti redakčních systémů a detailně popsány hlavní stavební kameny a funkcionality těchto systémů. Dále je poukázáno na přední programovací jazyky využívané ke tvorbě těchto systémů a vývojové prostředí k tomu určených. Praktická část přehledně prezentuje jednotlivé kroky vývoje, ovládání systému a v neposlední řadě aplikace využívané k návrhu a tvorbě tohoto konkrétního redakčního systému.</p>
Klíčová slova	SQL, ASP, PHP, JAVA, redakční systém, informační systém, správa obsahu, CMS, WCMS, RS, Oracle, PostgreSQL, MySQL