Implementace a provoz zabezpečeného internetového portálu

Implementation and running secure internet portal

Bc. Radka Braunerová

Diplomová práce
2008

Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky

*** nascannované zadání str. 1 ***

*** nascannované zadání str. 2 ***

ABSTRAKT

Tato diplomová práce se zabývá tvorbou moderního webového portálu s ohledem na informační bezpečnost. Jsou zde uvedeny základní informace o webových aplikacích a současných technologiích používaných pro jejich tvorbu.

Cílem bylo zmapovat situaci na trhu s redakčními systémy, provést jejich srovnání na základě nadefinovaných požadavků a následně vytvořit webový portál, který by splňoval základní požadavky kladené na současné webové aplikace.

Klíčová slova: redakční systém, systém pro správu obsahu, webová aplikace, bezpečnost, validní kód, optimalizace.

ABSTRACT

This diploma work deals with formation of modern web portal with respect to security of information. Reader can find basic facts about web based applications and current technologies, used for building of such web based applications.

Target of this work was, as well, to map situation on the market of editorial systems, make comparison based on defined criteria and set up web based portal system, which would match basic requirements imposed on common web applications.

Keywords: editorial system, content management system, web-based application, security, valid code, optimalization.

Ráda bych touto cestou poděkovala doc. Mgr. Romanu Jaškovi, Ph.D. za pomoc při vypracovávání diplomové práce, za jeho náměty a poznatky.

Poděkování patří i mé rodině a příteli za jejich psychickou podporu a trpělivost.

Prohlašuji, že jsem na diplomové práci pracovala samostatně a použitou literaturu jsem citovala. V případě publikace výsledků, je-li to uvolněno na základě licenční smlouvy, budu uvedena jako spoluautor.

Ve Zlíně …….

Podpis diplomanta

OBSAH

ÚVOD

V posledních letech zažívá celosvětová síť Internet velký rozmach, a to především díky své službě World Wide Web. Na počátku byly „internetové stránky“ většinou realizovány výhradně statickými soubory umístěnými na jednotlivých serverech. Tyto soubory obsahovaly neměnný textový či grafický obsah, nezávisle na čase či požadavcích uživatele. Správu takového webu prováděla jedna či více specializovaných osob, které zajišťovaly přesnost a aktuálnost poskytovaných informací. Tento způsob byl časově náročný, a vyžadoval i specifické znalosti osob, které ruční změny v prezentovaných webových stránkách prováděly.

V současné době, kdy dochází k růstu objemu informací a kdy jsou na webové prezentace kladeny vyšší nároky (např. dynamičnost, rychlá reakce na změny apod.), není již dostačující spravovat web tímto starým způsobem. Do popředí se dostávají tzv. interaktivní webové aplikace, které umožňují měnit obsah na základě požadavků uživatele, vzájemnou komunikaci mezi návštěvníky, změnu obsahu bez nutnosti jeho znovunačtení (obdoba desktopových aplikací) apod. Existuje specifická kategorie softwarových produktů, které usnadňují a zefektivňují správu obsahu webových prezentací. Jedná se o tzv. systémy pro správu obsahu. Od konce 90. let bylo vytvořeno značné množství aplikací, které spadají do této kategorie, a jejich počet neustále vzrůstá.

Záměrem mé diplomové práce je vytvoření webového portálu EAS, který bude implementován ve zvoleném redakčním systému. Před započetím samotné realizace bude nutné provést určité dílčí kroky. Nejprve bude zapotřebí zmapovat trh redakčních systémů, provést analýzu potřeb portálu EAS a na jejím základě porovnat jednotlivé redakční systémy. Při volbě redakčního systému bude brán zřetel na informační bezpečnost a dostupnost.

I. TEORETICKÁ ČÁST

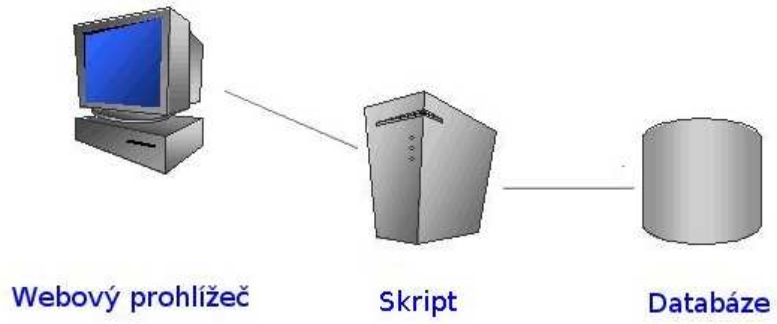
1 WEBOVÁ APLIKACE

Cílem této kapitoly je vysvětlit základní pojmy a principy týkající se webových aplikací. Je zde popsáno, jakými mechanismy lze převést statické stránky na stránky dynamické.

V závěru kapitoly jsou stručně uvedeny základní informace o bezpečnosti webových aplikací a o typech útoků, kterým webové aplikace čelí.

1.1 Vysvětlení pojmu

Webová aplikace je software typu klient-server, který komunikuje s uživatelem nebo jiným systémem prostřednictvím protokolu HTTP. Na straně serveru běží kód zajišťující funkci programu (skript), který často bývá propojen s některou databází (ta uchovává data webové aplikace). Klientem je webový prohlížeč[1], který na základě interakce s uživatelem zasílá serveru jednotlivé požadavky. Server jako reakci na klientův požadavek zasílá odpověď. Formát požadavku i odpovědi je definován ve specifikaci protokolu HTTP.



Obr. 1. Schéma webové aplikace

V případě tvorby webových stránek máme v zásadě k dispozici dvě možnosti: statické či dynamické stránky.

[1] Např. Internet Explorer, Mozilla Firefox či Opera

1.2 Statické webové stránky

Charakteristickou vlastností statických stránek je, že se obsah stránek či vzhled po jejich načtení ve webovém prohlížeči nemění. Příkladem statických stránek mohou být například informace o firmě nebo kontaktní informace.

Statické stránky zobrazují opravdu jen to, co je v nich uvedeno. Pokud např. chcete napsat na konec každé stránky aktuální datum, je potřeba tuto informaci na každou z těchto stránek ručně zadat a v případě aktuálního datumu, tuto hodnotu také každý den aktualizovat. Tuto nepříjemnou vlastnost můžeme odstranit použitím stránek dynamických.

1.2.1 HTML/XHTML

Každá korektní webová stránka je webovému prohlížeči dodána ve standardním formátu HTML/XHTML.

HTML představuje značkovací jazyk sloužící zejména ke standardnímu popisu obsahu a struktury webových stránek. Jazyk HTML je charakterizován množinou značek, tzv. tagů a jejich atributů (doplňující informace). Dokument má předepsanou strukturu, která se odvíjí od jednotlivých verzí HTML.

XHTML definuje zcela nový jazyk pro tvorbu webových stránek na bázi XML. Narozdíl od HTML musí stránka splňovat spoustu pravidel. Striktnost XHTML spočívá především v tom, že pravidla, která byla dříve v HTML doporučována, jsou v XHTML vyžadována.

Základní rozdíly mezi HTML a XHTML vychází z faktu, že XHTML dokument musí být v první řadě validním XML dokumentem:

- Dokument musí začínat XML deklarací, např.:

  <?xml version='1.0' encoding='windows-1250'?>.

- Všechny tagy musí být ukončeny a to včetně nepárových .
- Všechny tagy a jejich atributy musí být psány malými písmeny.
- Všechny hodnoty atributů musí být uzavřeny do uvozovek.
- Je zakázáno křížení elementů (překrytí začátku nebo konce tagu), v následující tabulce (Tab. 1) je uveden příklad správného a nesprávného zápisu:

Tab. 1. Ukázka křížení elementů – správný i nesprávný zápis

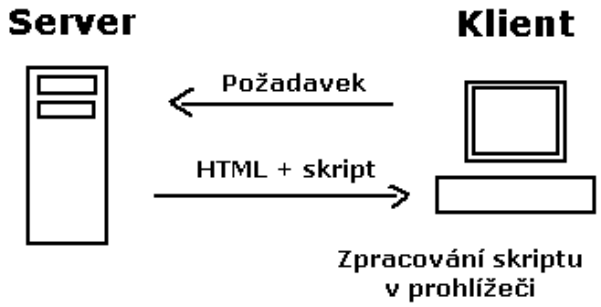
Nesprávně	Správně
text	text

1.3 Dynamické webové stránky

Narozdíl od statických stránek, dynamické webové stránky mohou měnit svůj obsah či vzhled po jejich načtení v prohlížeči. Tato dynamičnost může být zajištěna buď na straně klienta nebo na straně serveru.

1.3.1 Na straně klienta

Za dynamické stránky na straně klienta lze považovat stránky, které vyhovují specifikaci DHTML. Nejedná se o žádný nový jazyk, jde spíše o spolupráci klasického HTML se skritpy spouštěnými na straně klienta (většinou Javascript) a kaskádovými styly CSS. [1]



Obr. 2. Klientský skript

Javascript

Javascript je skriptovací jazyk, který se zapisuje přímo do HTML stránky a jeho vykonávání probíhá na straně klienta v prohlížeči. Syntaxe jazyka vychází z jazyků C a Java.

CSS

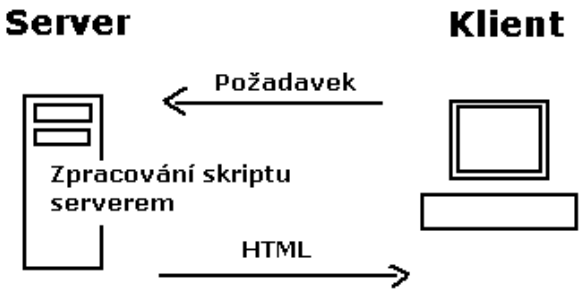
CSS je jazyk umožňující efektivně popisovat podobu webových stránek a styl jednotlivých prvků.

Původní koncepce HTML předpokládala, že veškeré formátování se provede přímo v příslušném tagu. Časem se tato koncepce ukázala jako nesprávná, a proto část týkající se popisu vzhledu byla oddělena od samotného obsahu HTML stránky.

Při správném používání CSS je tedy oddělen obsah od vzhledu, čímž získáme přehledný a jednoduchý HTML kód, výrazně se tím zmenší datová velikost stránky a pokud používáme stejný CSS soubor pro více webových aplikací, můžeme velice snadno změnit kompletní vzhled všech stránek změnou jednoho souboru.

1.3.2 Na straně serveru

Pod pojmem dynamičnost stránky lze kromě DHTML také rozumět schopnost přizpůsobit vzhled na základě předaných parametrů od klienta, a to na serveru, který stránku dynamicky podle těchto parametrů vygeneruje. Tyto parametry jsou často součástí adresy webové stránky, např. http://info.cz/stranka.php?id_stranky=123, nebo je možné je umístit do hlavičky požadavku HTTP, a tím je běžnému uživateli skrýt. [1]



Obr. 3. Serverový skript

Technologie, které se používají pro vygenerování webových stránek na serveru, můžeme rozdělit dle [2] do dvou základních kategorií:

CGI skripty

CGI skriptem se rozumí skript, napsaný v libovolném programovacím jazyce a přeložený do spustitelné podoby (např. exe soubor). Pokud klient takovýto skript volá, server program spustí a výsledek předá klientovi jako odpověď (ve formátu HTML).

Pro psaní CGI skritpů nejčastěji používá programovací jazyk PERL, nicméně bez problémů lze použít i Javu, C/C++ či Pascal.

Skriptové vsuvky vkládané serverem

Příkazy skriptu jsou přímo napsané v (X)HTML souboru. Webový server tyto skripty vyhodnotí, zpracuje a výsledek doplní do výstupního proudu – hotové webové stránky, kterou zašle klientovi.

Základním rozdílem oproti CGI skriptům je způsob provádění programu. Zatím, co CGI skript je soubor stojící mimo vlastní web ve vlastním adresáři, vsuvka je součástí webové stránky a je prováděna při čtení stránky a jejím odesíláním klientovi.

Mezi nejznámnější technologie využívající tento způsob generování stránek, patří zejména:

- SSI

  Jedná se o nejstarší a nejrozšířenější druh vsuvek vkládaných serverem. SSI obsahují jen šest příkazů (#config, #echo, #exec, #flastmod, #fsize a #include), které se vkládají do HTML stránky přímo do komentáře (<!-- .... -->) ve tvaru:

  <!--#příkaz parametr="hodnota"-->

  Server tuto značku nahradí za příslušný textový řetězec, podporuje-li SSI. V opačném případě se řetězec nevyhodnotí a server jej odešle klientovi jako komentář v HTML stránce, který se nezobrazí.

  Bohužel SSI nejsou standardizovaná a podporovaná všemi webovými servery.

- ASP, ASP.NET

  Pro oddělení příkazů ASP od běžného HTML kódu, se používají speciální značky, tzv. tagy <% .... %>, tzn. že k ohraničení příkazů nejsou použity jako u SSI komentáře.

  Skriptovým jazykem pro ASP může být VBScript (skriptový jazyk odvozený od Visual Basic) nebo JavaScript. Je možné na jedné stránce použít oba skriptové jazyky. Z hlediska výkonu to ale není dobré řešení, protože webový server musí použít oba serverové stroje na zpracování jedné stránky. Ani v případě, že nám nezáleží na rychlosti, nemůžeme se spolehnout na tom, že skriptové stroje skončí svoji práci a uloží výsledky do HTML v takovém pořadí, v jakém byly spuštěny.

  V roce 2002 po zavedení platformy .NET firmou Microsoft, byla představena nová verze ASP, s označení ASP.NET. Tyto stránky mohou být napsány v různých programovacích jazycích, např. C#, J#, Visual Basic, C++ a jiné.

  Tradiční ASP jsou uloženy na serveru v souborech s příponou .asp, ASP.NET používá příponu .aspx.

  Ačkoliv je ASP.NET odvozen od technologie ASP, jsou obě technologie velmi odlišné. Klasické ASP vykonávalo při každém zavolání stránky bloky od shora dolů a kód <% .... %> v blocích byl posílán na výstup okamžitě po jeho nalezení překladačem ASP a jeho vykonání. ASP.NET stránky jsou nejprve zkompilovány do tzv. Intermediate Language (IL) a poté je IL zkompilován do nativního kódu, který je

následně s spuštěn. Výsledkem je pak čistý HTML kód. Při všech následujících dotazech na danou stránku, se přímo přistupuje ke zkompilovanému kódu, čímž se oproti klasickému ASP zvýší rychlost reakce serveru na klientův požadavek.

- PHP

  PHP je skriptovací jazyk, který se jako předchozí jazyky, přímo začleňuje do HTML kódu. Pro oddělení PHP kódu od HTML kódu se používá <? .... ?>. Jedná se o interpretovaný jazyk (ne kompilovaný), jehož syntaxe je kombinací několika programovacích jazyků (C, Perl, Pascal a Java).

  Svou popularitu získal především svou nezávislostí na platformě, jednoduché syntaxi a tím, že je šířen bezplatně.

- JSP

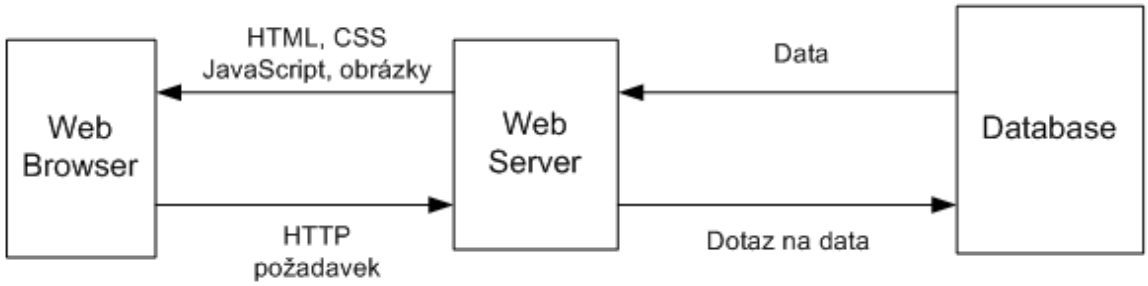
  JSP je technologie umožňující vkládání kódu programovacího jazyka Java přímo do HTML stránky, a to stejně jako u ASP pomocí speciálních značek <% .... %>.

  Při prvním volání JSP stránky, je na straně serveru vygenerován speciální servlet, jehož zdrojový kód je zkompilován a uložen do bytového souboru (.class). Poté je tento soubor pomocí rozhraní JVM spuštěn (je vytvořena instance servletu) a výsledek je poslán klientovi jako HTML stránka.

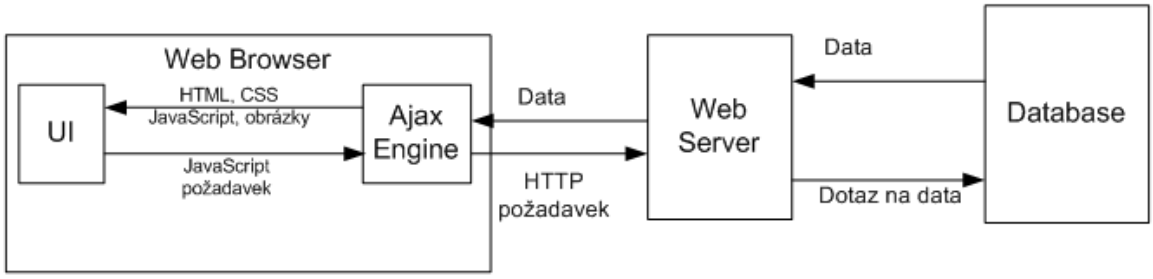
1.3.3 Technologie AJAX

Jedná se o technologii, která prostřednictvím skriptů umožňuje webové stránce komunikaci s webovým serverem. Aplikace jsou vyvíjeny s využitím technologií (X)HTML, Javascriptu, CSS a rozhraní XMLHttpRequest, které umožňuje asynchronní výměnu dat.

Smyslem tohoto přístupu je zamezit starému (původnímu) přístupu spočívajícím v přenášení vždy celé webové stránky a umožnit vytvářet webové aplikace, které by při komunikaci se serverem vyměnili vždy jen nezbytně nutný objem dat. [3]
Prakticky se využívá této technologie u doplňků webových stránek, jako jsou ankety, pomocník při vyhledávání - našeptávač, kterého má např. ww.seznam.cz.



Obr. 4. Tradiční model webové aplikace



Obr. 5. AJAX model webové aplikace

Technologie AJAX se nejčastěji používá za účelem vylepšení uživatelské interakce, zrychlení doby odezvy a snížení zátěže na webové servery a síť obecně. V současnosti se jedná o používanou a rozšířenou techniku pro tvorbu efektivních interaktivních webových aplikací.

1.3.4 Ostatní technologie

Na internetových stránkách se dále můžeme setkat s tzv. applety. Ty narozdíl od již zmíněných servletů, běží pouze na klientských počítačích. Mezi nejznámnější patří applety napsané v jazyku Java. Jejich použití v praxi s sebou ale přináší řadu nevýhod:

- Je nutné mít ve webovém prohlížeči nainstalován příslušný plugin.
- Vyšší nároky na výkon počítače.
- Startování appletu je pomalé.
- Bezpečnostní rizika.

V roce 1996 přišla na trh firma Microsoft s technologií zvanou ActiveX. Jedná se o samostatné programy, které je možno začlenit do HTML stránek. Pracují v prostředí prohlížeče a narozdíl od appletů mohou být naprogramovány v řadě programovacích

jazyků. Ovládací prvky ActiveX jsou však podporovány pouze v prohlížečích Internet Explorer.

Dále se můžeme setkat s vektorově orientovaným grafickým nástrojem Macromedia Flash, jehož pomocí můžeme vytvářet animovanou grafiku, kterou velice snadno zapojit do HTML stránek.

1.4 Hlavní zásady tvorby webových stránek

V poslední době se bohužel velice často setkáváme s tím, že se do tvorby webových stránek pouští lidé, kteří se naučí jen pár základních HTML tagů. Vůbec se nezajímají o validnost kódu, o správný způsob zápisu apod. Bohužel se s touto situací setkáváme i u „profesionálů“, kteří se tvorbou webových portálů zabývají. Přitom dodržení správného způsobu zápisu není vůbec složité a plyne z toho navíc řada výhod, např.:

- Správné zobrazení stránky na všech různých zobrazovacích zařízeních (obrazovka, tiskárna, PDA, mobilní telefony,...) ,...) a ve všech prohlížečích, které podporují (X)HTML od dané verze (verze, kterou určí autor aplikace).
- Zpřístupnění stránek i pro handicapované uživatele.
- Jednodušší tvorba stránek a následně případné úpravy.
- Vyhledaváče si stránky dodržující standardy lépe zaindexují a lépe je umístí ve výsledcích vyhledávání.
- Zaručení kompatibility i s budoucími verzemi prohlížečů.

Samostatnými prostředky pro dodržení zásad tvorby správného webu jsou uvedeny v následujících kapitolách. [4]

1.4.1 Validita kódu

Pod pojmen validita kódu se rozumí soulad (X)HTML kódu s technickými pravidly pro psaní zvoleného značkovacího jazyka[2], v němž je aplikace vytvořena. Tato pravidla pro tvorbu webových stránek vydává a upravuje organizace W3C, kterou založil v roce 1994 jeden z duchovních otců webu, Tim Berners-Lee, v zájmu standardizace HTML a XHTML jazyka. Hlavním cílem webových standardů je zajištění všeobecně přístupných, přehledných a logicky organizovaných internetových dokumentů.

Na začátku dokumentu (webové stránky) je potřeba uvést DOCTYPE, který určuje podle jaké specifikace je daná stránka napsána a říká prohlížeči, jakou množinu tagů má použít.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">
```

Doctype (zkratka pro "deklaraci typu dokumentu" – Document type) informuje validátor, která verze (X)HTML je použita. Musí se užívat vždy na začátku kódu každé webové stránky. Doctype je klíčová komponenta sloužící webovým stránkám: (X)HTML a CSS bez nich nemohou být validovány. [5]

Jakmile je v dokumentu deklarována určitá specifikace, měly by být použity tagy z dané specifikace a správné atributy jednotlivých tagů. Mělo by být také dodrženo správné vnořování tagů, není povoleno křížení elementů.

Validitu zdrojového kódu webových stránek lze zkontrolovat pomocí tzv. validátorů. Nejznámější validátor vytvořilo konsorcium W3C a lze jej nalézt na stránkách http://validator.w3.org. Pokud stránky neobsahují žádnou chybu, validátor potvrdí jejich bezchybnost.

Doporučení W3C definují mimo jiné[3] celou škálu variant jazyka HTML. Pomineme-li starší verze, které není doporučeno používat, je v současnosti aktuální verze HTML 4.01. Z tohoto jazyka vychází jazyk XHTML 1.0, který je pouze restrikcí HTML 4.01 tak, aby

---

[2] Tímto jazykem bývá nejčastější HTML.

[3] Seznam doporučení organizace W3C je možné nalézt na: http://interval.cz/clanky/prehled-standardu-w3c/

splňoval omezení jazyka XML. Jak specifikace HTML 4.01, tak XHTML 1.0 definují jazyk ve třech mutacích:

- přechodové (transitional)
- striktní (strict)
- rámové (frameset)

Validní kód se snáze edituje, je přehlednější, čistší a je vizitkou dobře odvedené práce každého tvůrce webových stránek. Dodržováním standardů může být zajištěno bezchybné fungování a zobrazení v současných webových prohlížečích. Toto nemusí být pravidlem, protože všechny současné prohlížeče mají menší či větší mezery v dodržování standardů, takže validita dokumentu je zatím jen předpokladem, nikoliv zárukou shodného zpracování dokumentu.

1.4.2 Sémantická správnost

Sémantika obecně je disciplína zabývající se významem slov a znaků. Při tvorbě webových stránek představuje sémantická správnost správné používání tagů. Každý HTML tag má přiřazen určitý význam a je tedy důležité použít tento tag pro správný účel.

Mějme např. tag <h1>, kterému je přiřazen význam hlavního nadpisu. Vyhledávač frázi ohraničenou tímto tagem považuje z hlediska obsahu za významnější než zbytek textu (sématicky je jasné, že nadpis obsahuje klíčovou informaci, která určuje obsah textu). Sémantika mu tedy umožňuje poznat, o čem daná stránka vlastně je a při vyhledávání pak snáze tuto stránku najde.

Sémantika se ale týká i odkazů na jednotlivé stránky. Pokud odkaz obsahuje např. text „zahradní nábytek“, je zřejmé, že stránka bude obsahovat informace o zahradním nábytku. Proto je z hlediska sémantiky chybou dávat do odkazu text např. „zde“, který kromě sémantiky má i špatný vliv na použitelnost webu.

1.4.3 Optimalizace pro vyhledávání

Při tvorbě webu existují určitá pravidla, po jejichž aplikaci dochází zpravidla k vyšší návštěvnosti webu (umístěním webu na předních místech ve výsledcích vyhledávání). Globálně se tato činnost nazývá optimalizace webových stránek pro vyhledávače (zkratka SEO).

Základní pravidla SEO optimalizace:

- Zvolení vhodného jména domény.
- Použití klíčových slov (důležitá je správná hustota klíčových slov v textu, jejich rozmístění, ale především jejich výběr, kterému by měla předcházet detailní analýza).
- Zajímavý a hodnotný obsah webu (obsah určitého rozsahu s pestrým významem).
- Množství a kvalita odkazů (použití tzv. přátelských adres[4], důležitá je také struktura odkazů uvnitř daného webu, odkazy by měly být funkční apod.).
- Použití titulků u obrázků a odkazů (tagy alt a title).
- Správné rozmístění zvýrazňovacích tagů (nejlépe na klíčové slovo), využívání elementů pro nadpisy (<h1>, <h2>, (<h3>).
- Použití relevantních titulků stránek a META tagů.

Cílem SEO je především zvýšit návštěvnost webu, zlepšit jeho použitelnost a přístupnost, zvýšit kvalitu webu a tím i jeho konkurenceschopnost a získat nové návštěvníky.

1.4.4 Oddělení obsahu od formátování

Je nutné, aby se jazyk HTML používal pouze pro zápis obsahu, formátování by se mělo provádět pomocí kaskádových stylů CSS. Definice CSS stylů mohou být vloženy přímo do HTML stránky nebo je lze dokonce aplikovat přímo na určitý tag použitím atributu style. Ani jedno z těchto řešení není nejsprávnější, definice stylů by měla být zapsána nejlépe v externím (samostatném) souboru a přímé stylování by se mělo používat minimálně (už proto, abychom zabránili zbytečnému prodlužování souborů přenášených přes pomalá a limitovaná internetová připojení).

Navíc zápis obsahu společně s formátováním v jednom souboru má spoustu nevýhod. Tento soubor je nepřehledný, mohou nastat problémy se zobrazením výstupu na různá zobrazovací zařízení. Pokud chceme změnit formátování např. všech nadpisů a podnadpisů, musíme projít všechny stránky daného webu, což je velice neefektivní.

[4] URL adresy přátelské k vyhledávacím robotům, tzv. Search Engine Friendly (SEF).

1.4.5 Přístupnost stránek (bezbariérový web)

Webové stránky by měly být přístupné všem bez ohledu na operační systém, webový prohlížeč, zobrazovací zařízení či zdravotní handicap uživatele. Přístupný web tedy znamená, že jej mohou používat uživatelé s odlišnými požadavky. Těmi mohou být:

- uživatelé používající zastaralé počítače či webové prohlížeče
- uživatelé používající jiná zobrazovací zařízení
- uživatelé, kteří mají vypnutý javascript, vypnuté zobrazování obrázků
- uživatelé se zhoršeným zrakem, barvoslepí nebo nevidomí uživatelé používající čtečku obrazovky
- uživatelé s poruchou soustředění nebo dyslexií
- uživatelé s poruchou pohybového aparátu
- sluchově postižení uživatelé.

V České republice byly vytvořeny dvě sady pravidel:

Blind Friendly Web

Tato pravidla jsou zaměřena primárně na zrakově postižené. Ještě před nedávnem byl přístupný web zabývající se touto problematikou na adrese http://www.blindfriendly.cz, v současnosti je pravděpodobně z technických důvodů nedostupný. Obdobu můžeme nalézt na stránkách http://www.blindfriendly.sk.

Pravidla tvorby přístupného webu

Pravidla tvorby přístupného webu[5] byla vydána v souladu s novelou Zákona č. 365/2000 Sb. o informačních systémech veřejné správy, která požaduje aby „informace související s výkonem veřejné správy byly uveřejňovány ve formě, která umožňuje, aby se s těmito informacemi v nezbytném rozsahu mohly seznámit i osoby se zdravotním postižením“. Tato pravidla jsou závazná od 1.1.2008. Tedy pokud jsou webové stránky určeny pro veřejnou správu, je dodržení těchto pravidel povinností.

[5] Podrobnější informace naleznete na: http://pristupnost.nawebu.cz/texty/pravidla-standardy.php?full

1.4.6 Použitelnost stránek (ergonomie webu)

Použitelnost stránek určuje, zda jsou webové stránky intuitivně a snadno ovladatelné, přehledné a srozumitelné a zda se uživatelům na daných stránkách dobře orientuje.

Na základě testování a výzkumů vznikla určitá pravidla, která popisují na co jsou uživatelé zvyklí, co jim při používání webových stránek pomůže. Zde jsou uvedeny některé z nich:

- Úvodní stránka by měla obsahovat základní informace o webu, co na něm uživatelé naleznou.
- Každá stránka by měla odkazovat na úvodní stránku. Uživateli by mělo být jasné, kde se v rámci webu nachází.
- Stránky by měli být rozděleny na jasně definované oblasti. Měly by být rozlišeny důležité a méně důležité prvky webu.
- Stránky by neměly obsahovat zbytečné informace.
- Odkazy by měly být viditelné, mělo by být jasné, na co lze na stránce kliknout.

Bohužel nelze jednoznačně určit, co je a co není použitelné, ale je možné použitelnost webu ověřovat. To lze provádět několika způsoby:

Heuristická analýza

Někdy se jí také říká analýza použitelnosti. Jejím cílem je odhalení slabých míst webového portálu. Provádí ji odborník na použitelnost, který využívá pravidla použitelnosti a své znalosti a zkušenosti.

Analýza využívající statistiku návštěvnosti

Statistiky návštěvnosti obsahují zajímavé informace o počtu a struktuře návštěvnosti a následně pomůže zvýšit efektivitu webu. V rámci analýzy návštěvnosti lze zjistit informace jako např.: počet návštěvníků, z jakých stránek návštěvníci na web přišli, na základě jakých klíčových slov, jaké stránky si na webu nejčastěji prohlížejí, z jakých zemí jsou návštěvníci a mnohé další.

Testování na uživatelích

Jedná se o testování za pomocí běžných uživatelů, kteří plní předepsané úlohy zaměřené na použitelnost webu. Prostřednictvím uživatelského testování lze získat neocenitelnou

zpětnou vazbu přímo od reálných uživatelů, kteří během testování odhalují problémy a slabá místa webového portálu.

Podle výzkumů světoznámého odborníka na použitelnost Jacoba Nielsena, jsou tato základní pravidla až dvěma třetinami webů porušována. To v konečném důsledku znamená, že velké množství uživatelů se na dané stránky již nevrátí, nedokončí své objednávky, apod.

2 BEZPEČNOST APLIKACÍ

Bezpečnost webové aplikace lze posuzovat a zajišťovat na mnoha různých úrovních a vrstvách. Počínaje fyzickou ochranou síťové infrastruktury a počítače, na kterém běží aplikační či databázový server, a konče zabezpečením operačního systému, souborového systému, databázového stroje, síťové komunikace apod.

Bezpečnostní problematika je rozsáhlá, a proto jsem se ve své práci zaměřila pouze na aplikační bezpečnost, která je často vývojáři zcela ignorována. V této kapitole jsou popsány nejznámnější a nejčastější metody útoku společně s obecným popisem obrany proti nim.

2.1 Základní typy útoků

2.1.1 Cross Site Scripting (XSS)

Jedná se o jeden z nejběžnějších útoků na webové aplikace. Útočník vloží na určité místo (vstup aplikace[6]) svůj vlastní kód (HTML kód, JavaScript,...), který se při zobrazení stránky, do kterého se ho podařilo umístit, vykoná. Tento kód může manipulovat s obsahem stránky a tím pozměnit zobrazované informace. Útočník může použitím XSS získat soukromé informace, spouštět na systému uživatele škodlivý kód. Navíc může provádět přesměrování na útočníkův web a tím například získat hodnotu cookie pomocí javascriptu (př.: http://www.utocnik.cz?+document.cookie).

2.1.2 Session hijacking (krádež sezení)

Tento způsob útoku spočívá v odchycení identifikační informace klienta a využití této informace k přístupu do aplikace v kontextu. Zcizení a použití této identifikační informace dává útočníkovi pro přístup k aplikace veškerá práva, která má původní uživatel.

Pro pochopení tohoto útoku je důležité porozumět způsobu, jakým se přihlášení a následné ověřování uživatele u webových aplikací provádí. Při prvním vstupu na webovou stránku je uživateli vygenerována jednoznačná a pouze jemu známá informace, tzv. sessionId, kterou se uživatel při každém dalším požadavku na server prokazuje. Pokud zaslaná informace od

[6] Vstupem aplikace může být položka formuláře, parametr v URL adrese či hodnota cookie.

klienta souhlasí s informací uloženou na serveru, je uživatel identifikován a webová aplikace si v session může zjistit o daném uživateli potřebná data (např. proměnná identifikující, zda se uživatel přihlásil platnými přihlašovacími údaji).

Existují 3 způsoby, jak udržet session aktivní: [6]

Pomocí HTTP autentizace

Využívá logovací dialog, který je integrován přímo v prohlížeči. Pokud uživatel požádá o neveřejnou stránku, otevře se dialog pro zadání uživatelského jména a hesla. Zadané údaje jsou odeslány na server a pokud je totožnost uživatele ověřena, jsou tyto údaje zaznamenány v prohlížeči. Při každém dalším požadavku o neveřejnou stránku, jsou tyto údaje zaslány serveru v HTTP hlavičce.

Výhodou je, že tyto údaje nelze odposlouchat pomocí JavaScriptu.

Předávání sessionId v cookie

Jedná se o nejpoužívanější způsob. Po úspěšném přihlášení odešle server sessionId pomocí cookie, které je uložena na klientském počítači. Při každém dalším požadavku o novou stránku, je tato cookie obsahující sessionId zaslány serveru v HTTP hlavičce.

Předávání sessionId v odkazech

SessionId se není uložena jako v předchozím případě v cookie, ale je předávána přímo pomocí odkazů na jednotlivé stránky. Tyto odkazy vypadají následovně:

<a href=“dalsi_stranka.asp?sessionId=123456789“>

a ve všech formulářích je vloženo skryté pole:

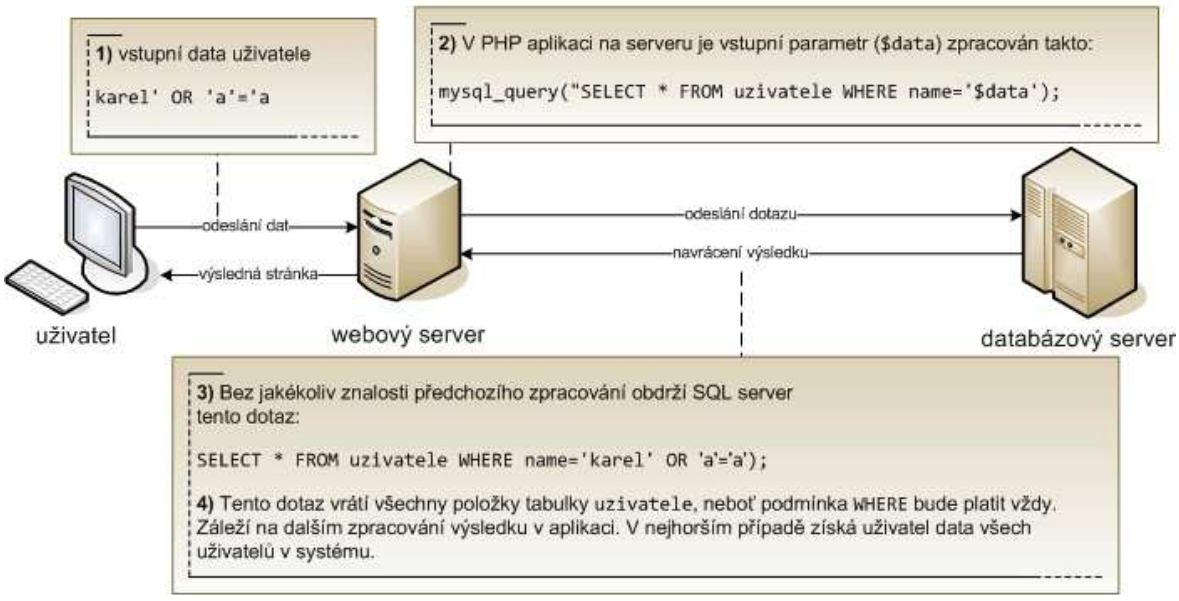
<input type="hidden" name="xessionID" value="123456789">

2.1.3 SQL Injection

SQL Injection (volně lze přeložit např. jako vkládání SQL příkazů) je technika, která umožňuje do existujících příkazů pro práci s databází vložit vlastní SQL příkaz nebo modifikovat stávající, pro dosažení jiného výsledku než pro jaký byl navržen. Typicky se může jednat o modifikaci omezující podmínky při výběru dat z databáze (tj. změna klauzule „where“). Vkládat lze i příkazy které mohou modifikovat data v databázi (např. mazat tabulky, rušit celé databáze).

Princip SQL injection je založen na vložení kódu do databázového příkazu bez ošetření uživatelem zadané hodnoty.

Typický příklad zneužití je znázorněn na obrázku (Obr. 6): [7]



Obr. 6. Typický příklad útoku SQL Injection

Místo vstupních dat „karel‘ OR ‘a’ = ‘a” můžeme aplikace předat také: „karel’; drop table uzivatele;”. Tímto způsobem zrušíme tabulku uzivatele (pokud na to daný uživatel má právo).

Pokud je navíc aplikace napsaná tak, že se do databáze přistupuje pod uživatelem, který je v dané databázi administrátorem nebo vlastníkem, je možné s databází provádět prakticky veškeré manipulace (takový uživatel má neomezená práva ke všem databázovým objektům).

2.2 Prevence a obrana

Webové aplikace musí především ošetřovat veškeré uživatelské vstupy, zda očekávaná data mají předem definovaný formát, délku. Je dobré předem definovat množinu povolených znaků pro konrétní a zakázat všechny ostatní znaky. Číselné vstupy by tak měly obsahovat pouze numerické znaky (např. PSČ bude obsahovat pouze čísla, maximální délka řetězce může být 5), uživatelská jména alfanumerické znaky (případně i tečku, apod.).

Znaky, která mohou znamenat bezpečnostní rizika:

<	>	"	'	;	%	(	)	&	+	-	*	/	\

Kontrola vstupu by měla být prováděna co nejdříve. Kontrola vstupu javascriptem je možná, ale neměla by být jediná, protože se jedná o kontrolu ještě na straně uživatele, kterému nemůžeme důvěřovat (útočník může JavaScript vypnout, popř. data upravit).

Pokud vstupní data nevyhovují podmínkám (obsahují nepovolené znaky), je důležité zobrazit stučné chybové hlášení včetně např. kontaktu na zodpovědnou osobu. Chybová hlášení často obsahují informace o příčinnách a jiné detailní informace (např. názvy tabulek). Je důležité tyto údaje nezobrazovat, protože by je případný útočník mohl využít.

Posledním, ale velmi důležitým doporučením je provoz databáze pod uživatelem s nejmenšími faktickými právy.

3 REDAKČNÍ SYSTÉM

3.1 Vysvětlení pojmu

Pro redakční systém se používají i oborově podobné termíny, jako např. systém pro správu obsahu, publikační systém nebo Content Management System (ve zkratce CMS).

Jedná se o systém sloužící ke tvorbě, distribuci, správě a administraci informací, které chce majitel systému zpřístupnit určité skupině uživatelů pomocí webového prohlížeče.

Hlavním cílem každého takového systému je přehledně a kvalitně zobrazit požadované informace a umožnit uživateli jednoduché ovládání. Častým důvodem pro nasazování redakčních systémů v praxi je fakt, že zjednodušují práci při publikaci textů. Není potom zpravidla zapotřebí mít tým kvalifikovaných pracovníků, starajících se o aktualizaci dat.

Pomocí uživatelského rozhraní, které je v mnoha případech podobné textovému editoru, pak může uživatel jednoduše vytvářet nové stránky. Celá správa webového portálu je prováděna pomocí webového prohlížeče, takže obsah lze upravovat kdekoliv odkudkoliv bez nutnosti cokoliv instalovat.

3.2 Požadavky pro provoz

K provozu redakčního systému obecně nestačí mít nainstalovaný pouze tento systém, ale je zapotřebí mít pro jeho provoz připravené prostředí.

3.2.1 Webový server

Webovým serverem se rozumí program (software) i počítač (hardware) připojený k počítačové síti, který přijímá požadavky od klientů (webový prohlížeč), tyto požadavky zpracuje a vrací zpět odpověď, obvykle jako HTML dokument, obrázek, či dokument v jiném formátu.

Funkce webového serveru spočívá v komunikaci s prohlížečem uživatele pomocí HTTP protokolu, který funguje způsobem dotaz-odpověď. Důležitou vlastností HTTP je, že neuchovává stav relace, jedná o tzv. bezstavový protokol. Pokud tedy uživatel pošle několik požadavků na stejný zdroj, webový server považuje každý z nich za jedinečný. Tato vlastnost není v některých případech žádoucí, proto byl protokol HTTP rozšířen o již

zmíněné cookies, které umožňují uchovávat na serveru informace o stavu spojení na straně uživatele (např. obsah nákupního košíku u elektronického obchodu, idenitu přihlášeného uživatele, uživatelské předvolby, apod.).

V praxi se můžeme nejčastěji setkat s webovými servery:

- Apache HTTP Server (od nadace Apache Software Foundation)
- Internet Information Server (od firmy Microsoft)
- Sun Java System Web Server (od firmy Sun Microsystems)
- IBM WebSphere Application Server (od firmy IBM)

3.2.2 Interpret skriptovacího jazyka

Interpretem jazyka rozumíme program, který umožňuje provádění určitého zdrojového kódu, přičemž tento kód musí podléhat dohodnutým pravidlům a konvencím. Interpret na rozdíl od překladače, nevytváří žádný spustitelný soubor, vykonává pouze předepsanou činnost.

Na webovém serveru musí být nainstalován příšlušný interpret (v závislosti na použitém skriptovacím jazyku) a pokud si prohlížeč vyžádá zobrazení nějaké stránky, interpret kód dané stránky zkompiluje a jako výsledek vrátí HTML stránku.

3.2.3 Databázový server

K tomu aby redakční systém mohl uložit data týkající se obsahu webových stránek (články, obrázky,...), je zapotřebí mít nainstalován databázový server.

Mezi nejznámnější databázové servery patří:

- MySQL
- Microsoft SQL Server
- Oracle

3.3 Základní stavební prvky redakčních systémů

U většiny redakčních systémů lze identifikovat určitě společné stavební prvky. Těmi jsou:

3.3.1 Jádro systému

Tato část má na starosti celkový běh systému, komunikaci mezi moduly, logování akcí, apod. Jejím úkolem je spojení všech částí redakčního systému do jednoho celku.

3.3.2 Funkce pro práci se šablonami

Šablona je speciální dokument, která určuje způsob prezentace webových stránek (jejich vzhled). Díky šablonovému systému jsou při požadavku na zobrazení stránky doplněny do konkrétní šablony příslušná data z databáze. Poté dojde ke kompilaci do výstupních souborů, které se následně zobrazují uživateli ve webovém prohlížeči.

Šablony mohou být uloženy v databázi nebo na specifickém místě v souborovém systému.

3.3.3 Moduly

Nadstavbové moduly nejsou primárně určeny pro běh základního systému. Je možné díky nim systém doplnit o prvky, které uživateli zpříjemní návštěvu, např. uživatelské fórum, statistiky, vyhledávání, sekce pro stahování, apod

3.4 Shrnutí a využitelnost v praxi

Redakční systémy jsou v současnosti používány při implementaci rozlišných webových řešení. Na trhu existuje velké množství těchto systémů[7], více či méně kvalitních, komerční nebo tzv. Open Source[8].

[7] Podrobnější informace o existujících systémech pro správu obsahu naleznete na: http://www.cmsmatrix.org/matrix/cms-matrix

[8] Jedná se o možnost bezplatného hotového řešení včetně zdrojového kódu.

Použití redakčního systému v praxi přináší spoustu výhod:

- Snížení nákladů a doby vývoje webových aplikací.
- Tvorba webových aplikací je zpřístupněna i uživatelům se základními znalostmi klasických „office“ aplikací.
- Zlepšení kvality poskytovaných informací (přesnost, přístupnost, aktuálnost).
- Zjednodušení procesu publikace informací.

Samozřejmě, že redakční systémy mají i své zápory. Přestože lze většinu systémů konfigurovat, nelze vždy pomocí konfigurace splnit všechny požadavky kladené na systém. Další z nevýhod může být bezpečnost a to hlavně proto, že jsou redakční systémy (Open Source řešení) distribuovány i se zdrojovými kódy. V případě, že obsahují bezpečnostní chyby, může je útočník velice rychle zneužít a webové aplikace implementované tímto systém napadnout. Na druhou stranu umožňuje přítomnost zdrojového kódu provádění auditu bezpečnosti a použitelnosti.

II. PRAKTICKÁ ČÁST

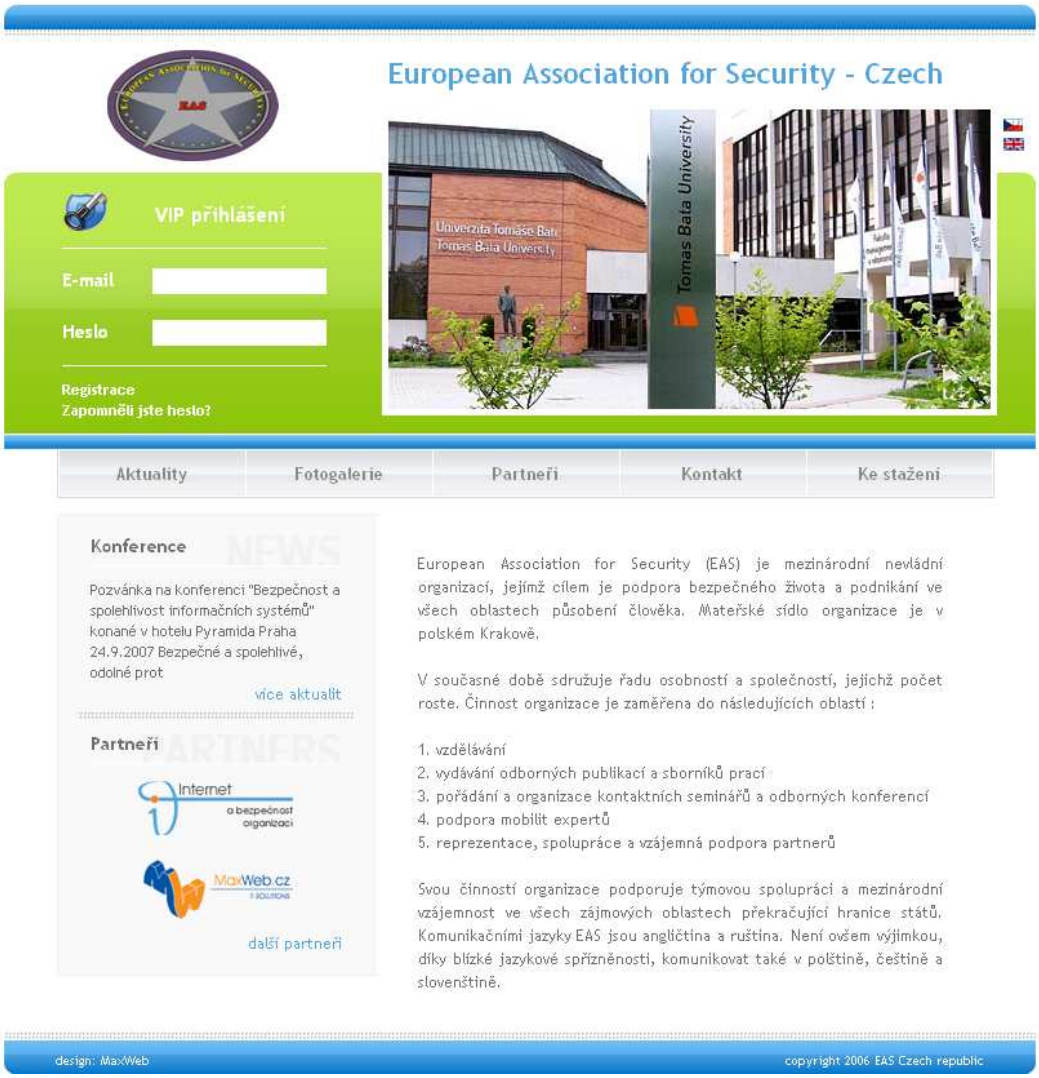
4 ANALÝZA A NÁVRH PORTÁLU EAS

Univerzita Tomáše Bati ve Zlíně je členem nevládní organizace European Association for Security (EAS) a garantem informačního portálu, který je přístupný na webové adrese http://www.eas.utb.cz/.

V této části diplomové práce je uveden popis struktury webového portálu EAS, je provedeno zhodnocení současného stavu, návrh nového portálu a stanovení požadavků.

4.1 Současný stav

Portál EAS představuje jednoduchou webovou prezentaci mezinárodní nevládní organizace, která se zabývá podporou bezpečného života a podnikání, vzděláváním, pořádáním odborných konferencí a seminářů a vydáváním odborných publikací.



Obr. 7. Vzhled stávajícího portálu EAS

4.1.1 Struktura webového portálu

Současná struktura portálu EAS je uvedena v následující tabulce (Tab. 2):

Tab. 2. Stávající struktura portálu EAS

Úvodní stránka	Statická stránka, která popisuje význam zkratky EAS, uvádí jakými oblastmi se organizace zabývá apod.
Aktuality	Zde jsou uvedeny informace o konání konferencí týkajících se bezpečnosti, jejich popis, odkazy na bližší informace.
Fotogalerie	Obsahuje fotografie z různých konferencí, seminářů, workshopů.
Partneři	Jsou zde uvedeny jména jednotlivých partnerů s odkazy na jejich domovské stránky.
Kontakt	Jedná se o kontaktní informace.
Ke stažení	Zde jsou k dispozici soubory, které si návštěvník může z portálu stáhnout. Jedná se především o textové soubory (doc, pdf), či obrázky (jpg), které jsou zatříděné do kategorií: ➢ Dokumenty EAS ➢ Ostatní ➢ Zpravodaj ➢ Všechny soubory
VIP přihlášení	Uživateli se nabízí možnost registrace, přihlášení do systému a v případě zapomenutí hesla jeho opětovné získání. Po přihlášení do systému by měl mít uživatel přístup k diskuzím a k databázi členů EAS. Bohužel se mi do systému nepodařilo zaregistrovat s tím, že registrace bude zprovozněna v nejbližším možném termínu.

4.1.2 Zhodnocení

Portál je implementovaný ve dvou jazykových verzích (v češtině a angličtině). Data se ale v jednotlivých verzích významně liší, především v sekci Aktuality, kde jsou v české verzi uvedeny informace o konferencích, které v anglické verzi chybí. Navíc jsou informace staršího data, nejsou aktuální.

Nemožnost zaregistrovat se mi nedovolila zhodnotit části systému, které jsou dostupné po přihlášení, jako je již zmíněný diskuzní modul nebo možnost přístupu k informacím týkajících se jednotlivých členů EAS.

Zaujal mě velice příjemný design portálu s jednoduchou navigací. Z ostatních hledisek hodnotím ale portál spíše jako nevyhovující.

4.2 Nový návrh portálu

Nový návrh má za cíl odstranit výše uvedené nedostatky a využít možností, které současné webové technologie nabízejí.

Portál bude implementován ve zvoleném redakčním systému. Při návrhu je zapotřebí definovat požadavky, které jsou na daný portál a zvolený redakční systém kladeny.

4.2.1 Technické požadavky

- Software použitý k provozu webového portálu musí být volně šiřitelný. Jedná se především o webový server (např. Apache), databázi (např. MySQL, Postgres).
- Portál musí být nezávislý na operačním systému.
- Programovací jazyk lze zvolit libovolně.

4.2.2 Funkční požadavky

- Základním požadavkem na novou verzi portálu je zachování stávajících funkcí. Ty budou součástí redakčního systému nebo mohou být dodatečně do systému doplněny formou např. nadstavbových modulů. Jedná se především o moduly zajišťující vícejazyčnost, diskuzi/fórum, fotogalerii, přihlášení/registrace do systému a kontaktní formulář.
- Redakční systém bude umožňovat vyhledávání v rámci portálu, popř. statistiky.
- Portál musí umožnit správu uživatelů (registraci, přihlášení) a správu jejich oprávnění.
- Vzhled webových stránek bude dán šablonou obsahující jak zobrazovaný HTML kód, tak i skriptovací jazyk. Výstupní kód bude validní.
- Pro formátování vzhledu budou použity kaskádové styly ve formě externích css souborů.

- Důležité je také zpracování administračního rozhraní, které by mělo být přehledné, jasné. Administrátor by neměl mít větší potíže se v systému zorientovat.
- Podpora instalace redakčního systému přes webové rozhraní, instalace by neměla být složitá.
- Redakční systém by měl mít možnost nabízet přátelské URL adresy.
- Možnost využití WYSIWYG editoru, který administrátorovi webového portálu usnadní práci při vytváření nových stránek, či editaci stávajících, zápisem textu a výběrem jeho formátování.
- Systém by dále mohl podporovat RSS, UTF-8 a FTP.

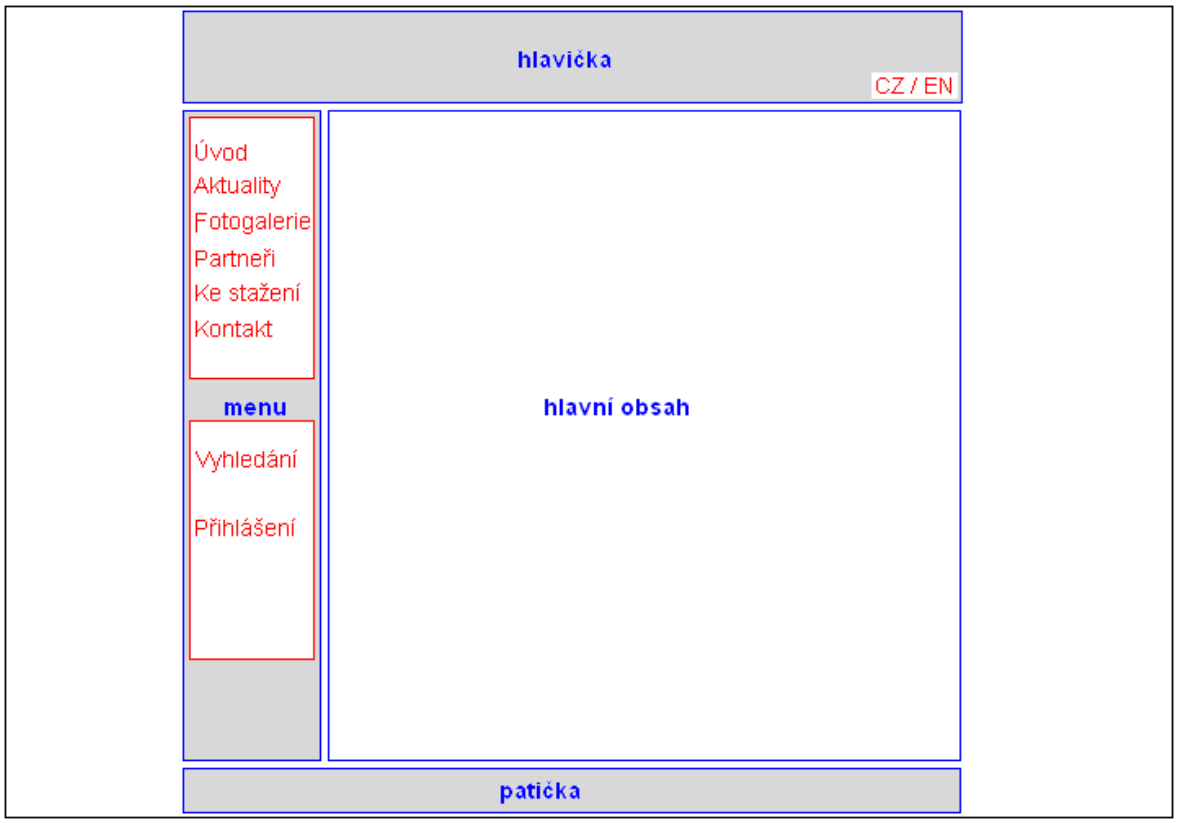
4.2.3 Bezpečnost

- Redakční systém musí být zabezpečen proti základním typům útoků, jako jsou XSS, SQL Injection.
- Redakční systém by měl mít možnost uchovávat historii přihlášených uživatelů.
- V případě potřeby by měl systém ověřovat validitu zadané emailové adresy zaslání aktivačního klíče.
- Administrátor by měl mít přístup k takovým informacím, jakou jsou: kdo je do systému přihlášený, co v systému dělá, a v případě nutnosti ho ze systému odhlásit.

4.2.4 Rozmístění grafických a textových prvků

Grafický návrh je velice důležitou součástí každého webového portálu. Při návrhu byly respektovány určitá pravidla informačního designu, jako např. pravidla týkající se přístupnosti webu (velikost a typ písma, kontrast mezi barvou textu a pozadím apod.).

Grafické rozmístění jednotlivých prvků portálu je znázorněno na následujícím obrázku (Obr. 8).



Obr. 8. Grafické rozmístění objektů na stránce

Návrh předpokládá existenci 4 hlavních částí, jimiž jsou:

- Hlavička - bude obsahovat logo EAS, který bude případně doplněn textem „European Association for Security – Czech“ stejně jako je na stávajícím portálu. Dále by zde mohla být možnost přepínání mezi českým a anglickým jazykem.
- Menu - obsahuje položky shodné se stávajícím portálem, v případě potřeby bude menu doplněno o nové.
- Hlavní obsah - pro zobrazení obsahu jednotlivých stránek.
- Patička - zde budou uvedeny např. informace o tvůrci použité šablony, informace o validitě HTML či CSS.

5 SROVNÁNÍ REDAKČNÍCH SYSTÉMŮ

V této kapitole je na základě specifikace požadavků provedeno porovnání vybraných redakčních systémů, zhodnocení výsledků a výběr jednoho, ve kterém bude následně implementace provedena.

5.1 Jednotlivé redakční systémy

Tato kapitola obsahuje popis základních vlastností vybraných redakčních systémů. Výběr jsem provedla na základě výsledků soutěže *2007 Open Source CMS Award* [8], kde se na prvních 3 místech v soutěži o nejlepší Open Source CMS umístily systémy: Drupal, Joomla! a CMS Made Simple. Dále jsem k této trojici připojila dle mého názoru zajímavé systémy phpRS a Plone.

5.1.1 Drupal

Oficiální stránky: http://www.drupal.cz/

Aktuální verze: 6.2

Základní vlastnosti:

- Podpora přátelských URL adres (navíc možnost definování více aliasů k jedné stránce a po instalaci lokalizačního modulu je možné k překladům dané stránky používat lokalizované cesty, například /en/welcome-to-drupal/ a /cs/vitejte-v-drupalu/).
- Propracovaná správa uživatelů.
- Vyhledávání, lokalizace, podpora vícejazyčnosti.
- Monitorování a logování aktivit (tyto údaje lze použít pro jednoduché statistiky).
- Velké množství dostupných rozšiřujících modulů.
- Workflow pro práci na složitějších projektech, podpora RSS.
- Možnost volby nástroje pro editaci obsahu (WYSIWYG editor, Texy!, Latex, HTML a další).
- Složitější uživatelské rozhraní.

Hodnocení:

Drupal je vyspělým systémem, který je navržen pro tvorbu složitějších stránek. Je navržen tak, aby jádro systému bylo co nejmenší a tím se docílilo stability a vysokého výkonu.

Vše je v systému označeno jako "node" neboli jednotka obsahu - může to být příspěvek, anketní otázka, běžná stránka nebo i vlastní definovaný typ. Každý node může obsahovat komentáře, může se zobrazovat různými způsoby, může být verzován a podobně. Výhodou tohoto návrhu je, že moduly mohou s obsahem pracovat jednotně.

5.1.2 Joomla!

Oficiální stránky: http://www.joomla.cz/

http://www.joomla.org/

Aktuální verze: 1.5.3

Základní vlastnosti:

- Registrace a administrace uživatelů, různá přístupová práva uživatelů.
- Zobrazování novinek, blogy, diskuzní fóra.
- Ankety, komentáře, hodnocení článků.
- Reklamní systém bannerů, kalendář, tisknutelné verze stránek, RSS export.
- Vyhledávání, lokalizace, podpora vícejazyčnosti.
- Velké množství dostupných rozšiřujících modulů.
- Jednoduchá, přehledná a propracovaná administrační část.

Hodnocení:

Joomla! je propracovaný systém používaný na celém světě, od tvorby jednodušších osobních stránek, přes obsáhlejší firemní prezentace, až k velkým firemním webovým stránkám.

Systém je u internetové veřejnosti velice oblíben, protože nabízí možnost stáhnutí velkého množství komponent nebo šablon vzhledu.

5.1.3 CMS Made Simple

Oficiální stránky: http://www.cmsMadeSimple.org

Aktuální verze: 1.2.4

Základní vlastnosti:

- Podpora přátelských URL adres, generuje validní kód.
- Nápověda integrovaná přímo v systému.
- Jednoduchá správa uživatelů, skupin.
- Minimální serverové požadavky.
- Jednoduchá instalace a aktualizace.
- Integrovaný správce souborů.

Hodnocení:

U nás tento systém není příliš známý, a to i přesto, že má již administrační rozhraní kompletně v češtině. Jedná se o velice zajímavý systém s jednoduchým ovládáním, který se hodí na malé až střední projekty. Umožňuje nastavit práva jednotlivým uživatelům i skupinám, umožňuje upravit vzhled šablon a podporuje doplňkové moduly. Nevýhodou může být obtížná realizace vícejazyčného webového portálu, což by mělo být odstraněno až v příští verzi 2.0.

5.1.4 PhpRS

Oficiální stránky: http://www.phprs.cz

Aktuální verze: 2.8.1

Základní vlastnosti:

- Správa tzv. informačních bloků, které tvoří základní stavební jednotku prezentační části systému.
- Registrace a správa uživatelů.
- Díky tzv. Download managementu lze zpřístupnit jakékoliv soubory webového portálu a sledovat statistiky jejich "stahování".

- Anketní subsystém (obsahuje interní ochranu, které se snaží zabránit vícenásobnému hlasování jedné osoby).
- Komentářový subsystém (umožňuje uživatelům komentovat vydané články, systém rozlišuje mezi registrovanými a anonymními uživateli).
- Statistický modul pro sledování celkové návštěvnosti webu.
- Vlastní interní zálohovací subsystém.

Hodnocení:

Systém phpRS patří mezi nejlepší české redakční systémy. Jeho síla je v přehlednosti a jednoduchosti ovládání administrace webu. Bohužel je k dispozici jen omezené množství komponent, což od použití spoustu zájemců odradí.

5.1.5 Plone

Oficiální stránky: http://plone.org/

Aktuální verze: 3.3.1

Plone je modulární redakční systém postavený na aplikačním serveru Zope, který je napsán v jazyce Python. Jeho ovládání je přeloženo do mnoha jazyků včetně češtiny, pro Windows a Macy jsou k dispozici instalátory.

Je snadno rozširitelný pomocí vlastních workflows, objektových portálových typu s vlastními daty, metadaty, metodami a vzhledem.

Základní vlastnosti:

- Podpora přátelských URL adres.
- Vizuální HTML editor, grafický editor pro úpravu stránek.
- Workflow pro práci na složitějších projektech, podpora RSS.
- Verzování, historie, uzamykání a odemýkání.
- Podpora vícejazyčnosti, podpora FTP.
- Automatická změna měřítka obrázků, generování miniatur (thumbnail).

Hodnocení:

Jedná se o uživatelsky přívětivý systém, který klade velký důraz na standardy W3C konsorcia, jednoduchou a čistou sémantiku. U nás tento systém zatím není příliš rozšířen. Neexistuje v současné době ani větší množství šablon, vzhled je však možno změnit úpravou v CSS kódu. Hodí se především pro rozsáhlé webové portály.

Je velice vhodný pro projekty, u nichž je zapotřebí propracovaný workflow systém.

5.2 Porovnání systémů

Porovnání jednotlivých systémů jsem provedla v tabulkovém procesoru MS Excel, ve kterém jsem sepsala jednotlivé požadavky kladené na redakční systém. Vzhledem k tomu, že jednotlivé požadavky mohou mít různou důležitost, přiřadila jsem každému z nich určitou váhu (v rozmezí 1 - 5, čím větší číslo, tím větší důležitost). Na základě informací z oficiálních stránek redakčních systémů, recenzí, diskuzí a také z vlastních zkušeností (díky demo verzím na [9]) jsem zjišťovala, zda systém dané požadavky splňuje.

5.2.1 Srovnávací tabulky

Vzhledem k tomu, že nebylo technicky možné zobrazit srovnávací tabulku se všemi redakčními systémy dohromady, bylo potřeba ji rozdělit:

Tab. 2. Vyhodnocení systémů Drupal a Joomla!

		Drupal v. 6.2			**Joomla! v. 1.0.15**		
Kritérium	**váha**	**Komentář**	**hodnoc.**	**body**	**Komentář**	**hodnoc.**	**body**
Technické požadavky							
Webový server	2	Apache, IIS	2	4	Apache	2	4
Databáze	2	MySQL, Postgres	2	4	MySQL	2	4
Operační systém	2	jakýkoliv	2	4	jakýkoliv	2	4
Programovací jazyk	1	PHP	2	2	PHP	2	2
Funkční požadavky							
Vícejazyčnost	5	ANO	2	10	ANO - modul	2	10
Diskuze	5	ANO	2	10	ANO - modul	2	10
Fotogalerie	5	ANO - modul	2	10	ANO - modul	2	10
Přihlášení	5	ANO	2	10	ANO - modul	2	10
Kontaktní formulář	3	ANO - modul	2	6	ANO	2	6
Vyhledávání	4	ANO	2	8	ANO	2	8
Statistiky	3	ANO	2	6	ANO	2	6
Správa uživatelů a oprávnění	5	ANO	2	10	ANO	2	10
Správa šablon	5	ANO	2	10	ANO	2	10
Administrační rozhraní	5	online, složitejší	1	5	online, jednoduché	2	10
Instalace	3	web. rozhraní, jednod.	2	6	web. rozhraní, jednod.	2	6
Přátelské URL adresy	4	ANO	2	8	ANO	2	8
WYSIWYG Editor	5	ANO - modul	2	10	ANO	2	10
Podpora RSS	2	ANO	2	4	ANO	2	4
Podpora UTF-8	4	ANO	2	8	ANO	2	8
Bezpečnost							
Ochrana proti XSS	5	nejednoznačné	1	5	nejednoznačné	1	5
Ochrana proti SQL injekcím	5	nejednoznačné	1	5	nejednoznačné	1	5
Historie přihlášení	4	ANO	2	8	ANO	2	8
Ověření emailové adresy	3	ANO	2	6	ANO	2	6
Session management	4	ANO	2	8	ANO	2	8
Vyhodnocení				**167**			**172**

Tab. 3. Vyhodnocení systémů CMS Made Simple a phpRS

		CMS Made Simple v. 1.2.4			**phpRS v. 2.8.1**		
Kritérium	**váha**	**Komentář**	**hodnoc.**	**body**	**Komentář**	**hodnoc.**	**body**
Technické požadavky							
Webový server	2	Apache, IIS	2	4	Server s podporou PHP	2	4
Databáze	2	MySQL, Postgres	2	4	MySQL	2	4
Operační systém	2	jakýkoliv	2	4	jakýkoliv	2	4
Programovací jazyk	1	PHP	2	2	PHP	2	2
Funkční požadavky							
Vícejazyčnost	5	ANO - limitované	1	5	ANO - limitované	1	5
Diskuze	5	ANO - modul	2	10	ANO - modul	2	10
Fotogalerie	5	ANO - modul	2	10	ANO - modul	2	10
Přihlášení	5	ANO - modul	2	10	ANO - modul	2	10
Kontaktní formulář	3	ANO - modul	2	6	NE	0	0
Vyhledávání	4	ANO - modul	2	8	ANO	2	8
Statistiky	3	ANO	2	6	ANO	2	6
Správa uživatelů a oprávnění	5	ANO	2	10	ANO	2	10
Správa šablon	5	ANO	2	10	ANO	2	10
Administrační rozhraní	5	online, jednoduché	2	10	online	2	10
Instalace	3	web. rozhraní, jednod.	2	6	web. rozhraní, složitější	1	3
Přátelské URL adresy	4	ANO	2	8	ANO	2	8
WYSIWYG Editor	5	ANO	2	10	ANO - modul	2	10
Podpora RSS	2	ANO - modul	2	4	ANO - modul	2	4
Podpora UTF-8	4	ANO	2	8	ANO	2	8
Bezpečnost							
Ochrana proti XSS	5	nejednoznačné	1	5	nejednoznačné	1	5
Ochrana proti SQL injekcím	5	nejednoznačné	1	5	nejednoznačné	1	5
Historie přihlášení	4	NE	0	0	ANO - modul	2	8
Ověření emailové adresy	3	nezjištěno	1	3	nezjištěno	1	3
Session management	4	nezjištěno	1	4	ANO	2	8
Vyhodnocení				**152**			**155**

Tab. 4. Vyhodnocení systému Plone

		Plone v. 3.1.1		
Kritérium	**váha**	**Komentář**	**hodnoc.**	**body**
Technické požadavky				
Webový server	2	Apache, Nginx, IIS	2	4
Databáze	2	Zope	1	2
Operační systém	2	jakýkoliv	2	4
Programovací jazyk	1	Python	2	2
Funkční požadavky				
Vícejazyčnost	5	ANO	2	10
Diskuze	5	ANO - modul	2	10
Fotogalerie	5	ANO	2	10
Přihlášení	5	ANO - modul	2	10
Kontaktní formulář	3	ANO - modul	2	6
Vyhledávání	4	ANO	2	8
Statistiky	3	ANO - modul	2	6
Správa uživatelů a oprávnění	5	ANO	2	10
Správa šablon	5	ANO	2	10
Administrační rozhraní	5	online	2	10
Instalace	3	run installer, složitější	1	3
Přátelské URL adresy	4	ANO	2	8
WYSIWYG Editor	5	ANO	2	10
Podpora RSS	2	ANO	2	4
Podpora UTF-8	4	ANO	2	8
Bezpečnost				
Ochrana proti XSS	5	nejednoznačné	1	5
Ochrana proti SQL injekcím	5	nejednoznačné	1	5
Historie přihlášení	4	ANO - modul	2	8
Ověření emailové adresy	3	ANO - modul	2	6
Session management	4	ANO - modul	2	8
Vyhodnocení				**167**

5.2.2 Vyhodnocení a závěr

Zjišťování zda daný systém splňuje určité kritérium či nikoliv, bylo časově velice náročné a bohužel u některých systémů se mi nepodařilo zjistit výsledek.

Celkově byly požadavky kladené na funkcionalitu všemi porovnávanými systémy splněny, buď je daná funkčnost přímo implementovaná v systému, nebo ji je možné doinstalovat v podobě nadstavbového modulu. Kvůli obtížné implementaci vícejazyčných webů byly jako první vyřazeny systém CMS Made Simple a phpRS.

Co se týče bezpečnosti systémů, především ochrany proti XSS a SQL injekcím, je vyhodnocení nejednoznačné. Všechny srovnávané systémy se s touto problematikou

potýkají, ale vždy když je taková bezpečnostní chyba objevena, je odstraněna aktualizací stávající verze nebo v nové verzi. Všeobecně lze říci, že bezpečnějším systémem je ten, jehož verze se již delší dobu používá. Je zde totiž předpoklad, že případné problémy v systému či modulu byli již odhaleny a odstraněny.

Z celkového hodnocení na základě požadovaných parametrů je vidět, že zbývající systémy (Drupal, Joomla!, Plon) jsou si velice podobné a vybrat nejlepší z nich, je prakticky nemožné. Všechny systémy se totiž v základních schopnostech shodují.

Plon se dle mého názoru hodí spíše pro rozsáhlé projekty a tím webový portál EAS není.

Zbývají tedy dva systémy, Drupal a Joomla!. Volba mezi nimi je složitá a není jednoznačná. U obou systémů nebyl problém s instalací a nastavením. Menší komplikace přišly až v části samotné tvorby portálu. Systém Drupal pro mě osobně byl zbytečně komplikovaný a tvorba portálu byla časově náročnější než v konkurenčním systému Joomla!. Lze to jistě přičíst faktu, že její předek (Mambo) byl kdysi komerčním projektem a jeho prostředí je vypracovanější, přehlednější. Joomla! se tedy stává vítězem.

6 IMPLEMENTACE PORTÁLU EAS

Implementace portálu EAS byla provedena v redakčním systému Joomla!. Původně měla být nasazena již nová verze 1.5 Stable, která má kompletně přepsané jádro systému, což s sebou přináší spoustu výhod, na druhou stranu bohužel ale i nekompatibilitu se staršími doplňky. Pro starší verzi je k dispozici více než 2700 doplňků, pro novou verzí jen několik desítek. Například komponenta Joom!fish zajišťující vícejazyčnost je stále ve verzi beta a není doporučena k použití na ostrém webu. Z toho důvodu je zvolena starší, ale stabilní a osvědčená verze 1.0.15. Pozdější migrace na novou verzi bude možná.

Jak již bylo zmíněno v kapitole 3.2, je zapotřebí mít pro běh redakčního systému připravené určité prostředí (Apache, PHP, MySQL). Využila jsem lokální systém Uniform Server verze 3.5-Apollo, ve kterém jsem portál kompletně připravila.

Uniform Server 3.5 je balík softwarových produktů obsahující:

- Apache Server 2.0.59
- Skriptovací jazyk PHP 5.2.3
- Databázový server MySQL 5.0.41
- Databázové pracovní prostředí phpMyAdmin 2.10.2

Pro zajištění korektního zobrazování českých znaků ve webovém prostředí, jejich ukládání a čtení z databáze, bylo zvoleno kódování UTF-8. V současné době se jedná o nejpoužívanější kódování, které podporuje češtinu, je vhodné při použití více jazyků a navíc je úsporné a tedy vhodné pro prostředí internetu.

6.1 Instalace redakčního systému Joomla

Na stránkách http://www.joomla.org jsou k dispozici všechny dostupné verze systému Joomla! zabalené v zip souborech. Po stáhnutí souboru Joomla_1.0.15-Stable-Full_Package.zip obsahujícího verzi 1.0.15 Stable, jej bylo zapotřebí rozbalit a celou adresářovou strukturu nahrát na lokální server. Následovalo vytvoření databáze v prostředí phpMyAdmin, kde se nastavilo odpovídající kódování na „utf8_czech_ci“. V posledním kroce bylo zapotřebí zadat ve webovém prohlížeči adresu http://localhost/eas, čímž se spustila samotná instalace systému Joomla!, která se skládá z celkem šesti kroků. Postupně se vyplnily informace jako např. host name, uživatelské jméno, název databáze.

Po úspěšné instalaci bylo nutné smazat adresář „installation" ze serveru, čímž se zpřístupnila na adrese http://localhost/eas úvodní obrazovka uživatelské části.

Podrobnější informace o instalaci prostředí a systému Joomla! naleznete na [10].

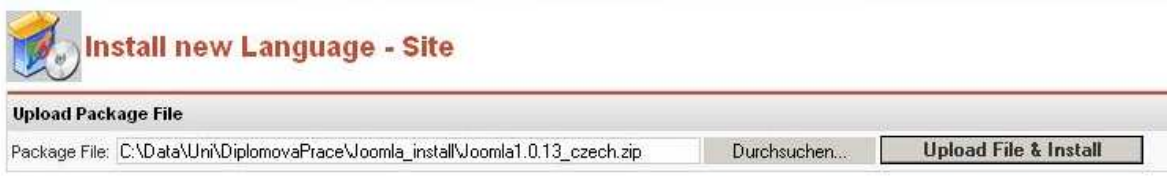
6.2 Tvorba webového portálu

Samotná tvorba webových stránek se provádí v administrátorské části, která je dostupná přes webové rozhraní na adrese http://localhost/administrator.

6.2.1 Prvotní nastavení

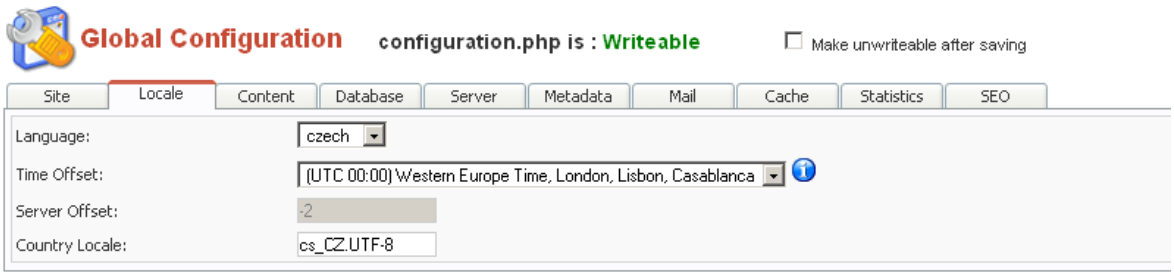
Čeština

Nejprve bylo nutné nainstalovat češtinu[9] a to v menu **Installers->Languages** (Obr. 9).



Obr. 9. Instalace češtiny

Nainstalovanou češtinu se následně nastavila jako výchozí, což se provedlo v menu **Site ->Global Configuration->**záložka **Locale**. Zde bylo zapotřebí vybrat jazyk „czech“ a do položky **Country local** zadat „cs_CZ.UTF-8“ (Obr. 10).



Obr. 10. Nastavení češtiny jako výchozího jazyka

[9] Soubor Joomla1.0.13_czech.zip

Tím se nastavilo, že se proměnné, které vrací server přes funkci Locale, budou zobrazovat v češtině (např. názvy dní, měsíců, text tlačítka zpět), zajistí se správné třídění (a, b, c, č, d) a zobrazení správného formátu pro datumy (3.3.2006 místo 2006/3/3).

Optimalizace SEO

Joomla! má zabudovanou funkčnost SEO, kterou lze aktivovat v případě použití webového serveru Apache[10], který má povolený mod_rewrite (Apache modul). Nejprve bylo nutné přejmenovat soubor „.htaccess.txt“ umístěný v hlavní složce systému Joomla! na „.htaccess“. Ve většině případů není zapotřebí obsah tohot souboru ručně měnit. V menu **Site->Global Configuration->** na záložce **SEO** se aktivuje „Search Engine Friendly URLs“ pro přepisování dynamických adres na statické a „Dynamic Page Titles“ pro dynamické tvoření titulku (nadpisu) každé stránky.

Velmi důležité je nastavení názvu webového portálu, které se provede v menu **Site ->Global Configuration->**záložka **Site**, kde se do položky **Site name** zapíše jednoznačně popisující název webu. Jedná se o metainformaci, která se objevuje jako titul (Title) v každé stránce webového portálu a je velice důležitým prvkem SEO.

Posledním krokem je vyplnění metainformací celého webového portálu, které jsou také pro vyhledávače důležité. Jedná se o vyplnění položek description (popis stránky) a keywords (klíčová slova), v menu **Site->Global Configuration->**záložka **Metadata**.

6.2.2 Instalace potřebných komponent a uživatelské šablony

Pro splnění funkčních požadavků uvedených v kapitole 4.2 bylo zapotřebí rozšířit základní verzi instalace systému Joomla! o nové prvky. To se provede pomocí instalace volně dostupných rozšiřujících komponent (v menu **Installers->Components**).

V základní instalaci systému Joomla! jsou předinstalované 2 základní šablony frontendu (uživatelského prostředí), které nebyly vyhovující především z důvodu jejich nevalidního kódu. Z volně dostupných zdrojů byla zvolena šablona clubtvk_beautiful_day, která se nainstalovala v menu **Installers->Templates-Site**. Jedná se o šablonu plně validní,

[10] Nelze použít při použití IIS (Internet Information Services)

jednoduchou, avšak pro webový portál EAS z grafického hlediska nevyhovující. Z tohoto důvody byly provedeny především grafické úpravy umístění jednotlivých prvků na stránce a stávající obrázky byly nahrazeny novými.

V následující tabulce (Tab. 5) jsou uvedeny všechny doinstalované komponenty a šablony, potřebné pro implementaci webového portálu.

Tab. 5. Seznam nainstalovaných komponent, šablon

Použití	Název	Instalační soubor
Vícejazyčnost	Joom!Fish v. 1.8.2	JoomFish_1.8.2.zip
Fotogalerie	RSGallery2 v. 1.14.3	com_rsgallery2_legacy_1.14.3.zip
Diskuzní fórum	FireBoard v. 1.0.4	FireBoard_1.0.4_Stable_CompletePackage.zip
Seznam uživatelů	Userlist v. 2.5	Userlist_2_5.zip
SEF adresy	JoomSEF v. 2.2.6	com_joomsef-2.2.6.zip
Šablona	Clubtvk_beautiful_day	clubtvk_beautiful_day.zip

V kapitole 6.2.1 bylo mimo jiné nastaveno používání přátelských adres místo dynamických. Toto nastavení využívalo zabudovanou komponentu SEF, která podporuje vytváření adres ve tvaru:

- http://localhost/eas/component/option,com_frontpage/Itemid,1/lang,cs/

Komponenta JoomSEF umožňuje vytvářet čitelnější a přijatelnější adresy pro vyhledávací roboty a to ve tvaru:

- http://localhost/eas/uvodni-stranka-eas

Pro zajištění správné funkčnosti komponenty JoomSEF bylo nutné změnit nastavení v souboru .htaccess.

Pro nastavení používání zabudované komponenty SEF vypadá část kódu souboru .htaccess následovně:

```
########## Begin - Joomla! core SEF Section
############## Use this section if using ONLY Joomla! core SEF
## ALL (RewriteCond) lines in this section are only required if you
actually
## have directories named 'content' or 'component' on your server
## If you do not have directories with these names, comment them #
RewriteCond %{REQUEST_FILENAME} !-f
RewriteCond %{REQUEST_FILENAME} !-d
#RewriteCond %{REQUEST_URI} ^(/component/option,com) [NC,OR]
RewriteCond %{REQUEST_URI} (/|\.htm|\.php|\.html|/[^.]*)$  [NC]
RewriteRule ^(content/|component/) index.php
#
########## End - Joomla! core SEF Section
########## Begin - 3rd Party SEF Section
```

Pro nastavení používání komponenty JoomSEF vypadá část kódu souboru .htaccess takto:

```
########## Begin - 3rd Party SEF Section
############## Use this section if you are using a 3rd party (Non Joomla!
core) SEF extension - e.g. OpenSEF, 404_SEF, 404SEFx, SEF Advance, etc
#
#RewriteCond %{REQUEST_URI} ^(/component/option,com) [NC,OR]
RewriteCond %{REQUEST_URI} (/|\.htm|\.php|\.html|/[^.]*)$  [NC]
RewriteCond %{REQUEST_FILENAME} !-f
RewriteCond %{REQUEST_FILENAME} !-d
RewriteRule (.*) index.php
#
########## End - 3rd Party SEF Section
```

V jednom okamžiku může být aktivní jen jedna část kódu, takže v případě použití komponenty JoomSEF je nutné řádky pro nastavení komponenty SEF zakomentovat pomocí znaku #.

6.2.3 Vytvoření jednotlivých stránek

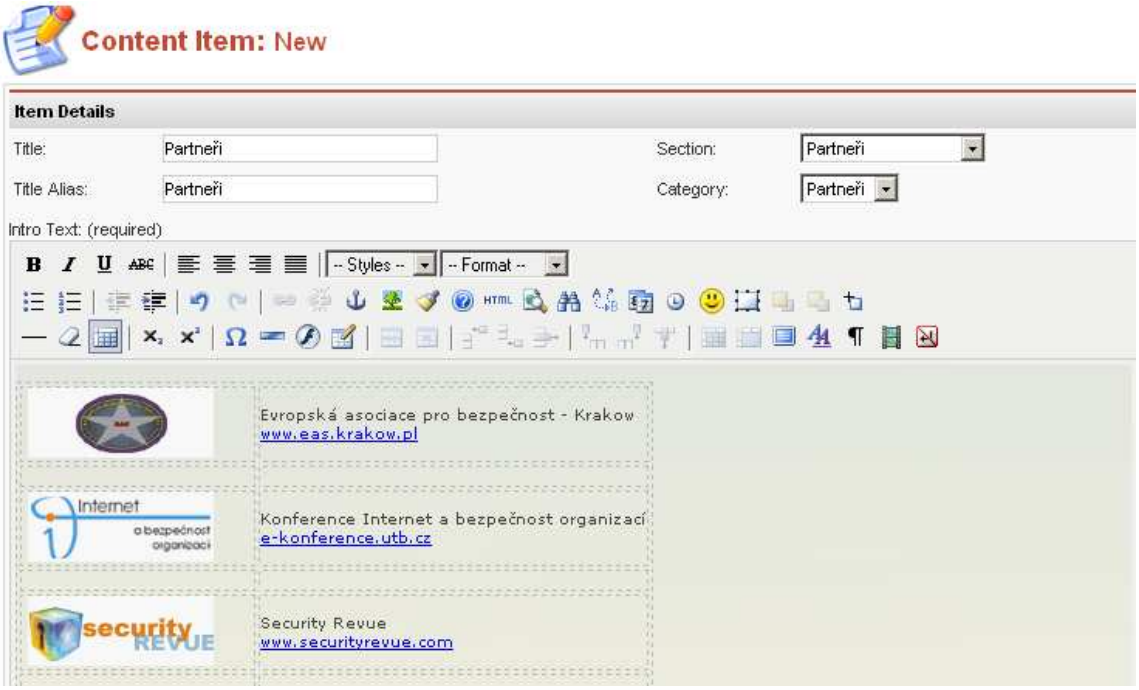
Celá struktura jednotlivých stránek je v systému Joomla! definována pomocí sekcí a kategorií, tzn. že každá vytvořená stránka musí být začleněna do určité sekce a kategorie. Před vytvořením stránek bylo potřeba tyto sekce a kategorie nadefinovat.

Webový portál EAS se skládá ze sedmi hlavních částí a ty byly nadefinovány jako sekce (v menu **Content->Section Manager**).Vzhledem k tomu, že většina z hlavních částí portálu (sekce) není dále dělena na menší části, byly vytvořeny kategorie stejného názvu jako sekce (v menu **Content->Category Manager**). Jedinou výjimkou je sekce „Ke stažení“, která se skládá ze tří kategorií. Přehled sekcí a kategorií je udává následující tabulka (Tab. 6):

Tab. 6. Přehled vytvořených sekcí a kategorií

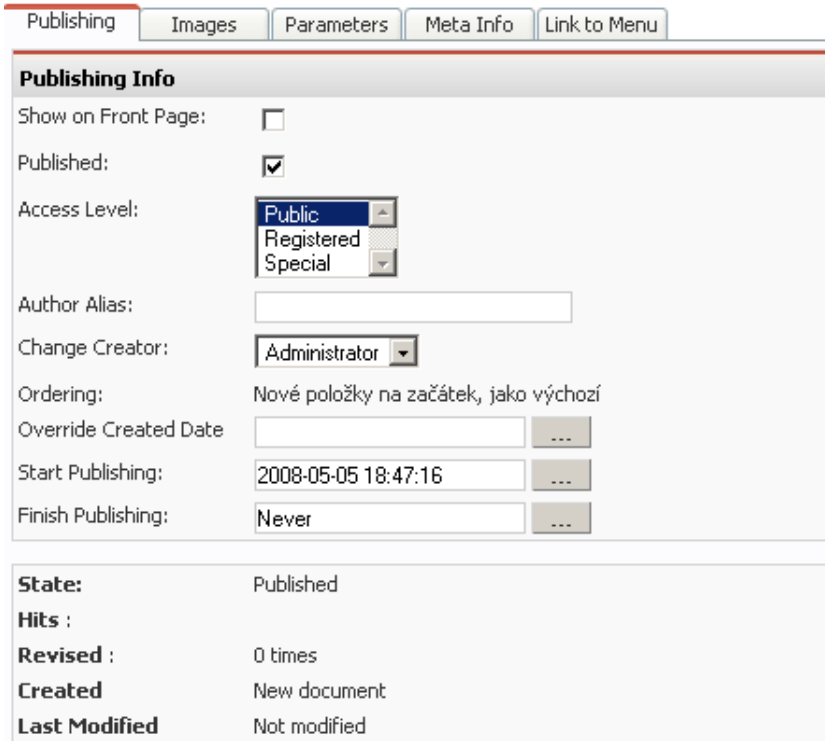
Sekce	Kategorie
Úvodní stránka	- Úvodní stránka
Aktuality	- Aktuality
Fotogalerie	- Fotogalerie
Partneři	- Partneři
Ke stažení	- Dokumenty EAS - Zpravodaj - Ostatní soubory
Kontakt	- Kontakt
Diskuze	- Diskuze

Po vytvoření sekcí a kategorií bylo možné začít s vytvářením jednotlivých stránek portálu (v menu **Content->All Content Items->New**). V první části se zadává název stránky (např. Partneři) a poté se provede výběr sekce a kategorie, ke kterým je vytvářená stránka přirazena (Obr. 11). Následuje vyplnění textu stránky a to pomocí vestavěného WYSIWYG editoru TinyMCE. K dispozici je použití náhledu zdrojového kódu, ve kterém lze provést případné úpravy přímo v HTML kódu.



Obr. 11. Vytvoření nové stránky – základní informace

V levé části (Obr. 12) je pak možné nastavit doplňující informace k právě vytvářené stránce, např. autor stránky, datum vytvoření, datum změny, možnost nastavení přístupu na danou stránku (přístupná všem nebo jen registrovaným uživatelům), možnost připojení obrázků, doplnění metainformací (velmi důležité pro vyhledávače) apod.



Obr. 12. Vytvoření nové stránky – doplňující informace

Tímto způsobem se vytvořily všechny stránky webového portálu. Jak je vidět na obrázku (Obr. 13), byla nastavena „Úvodní stránka“ jako implicitní domovská stránka (tzv. Front Page), čímž se zajistí, že při zadání URL adresy portálu do webového prohlížeče se zobrazí právě tato stránka.

Obr. 13. Seznam vytvořených stránek se zvýrazněnou domovskou stránkou

6.2.4 Doplnění překladů pomocí Joom!Fish

Komponenta Joom!Fish umožňuje díky přehlednému manageru (Obr. 14) jednoduše spravovat překlady jednotlivých částí vytvářeného webového portálu.

Obr. 14. Manager komponenty Joom!Fish

V části „Translation“ po vybrání příslušného jazyka a elementu, se postupně doplňovaly příslušné překlady. V našem případě bylo zapotřebí doplnit anglické překlady pro elementy:

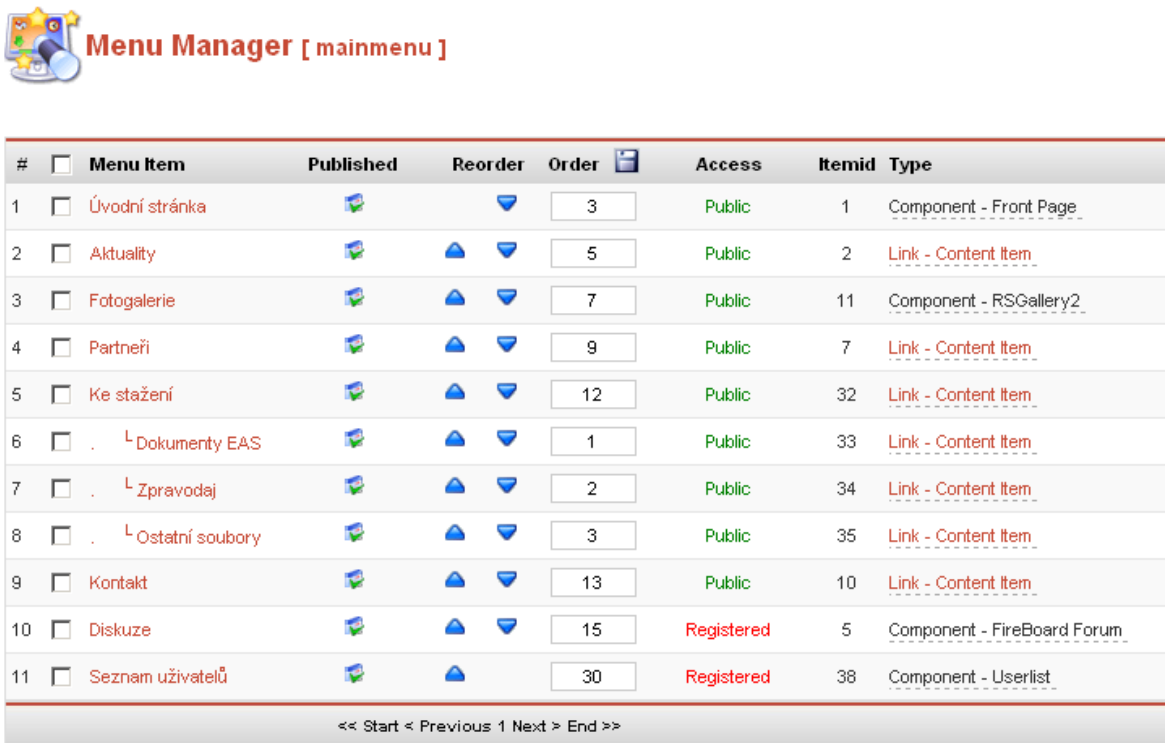
- Categories – přeložení názvů jednotlivých kategorií.
- Contents – obsahy jednotlivých stránek.
- Menus – jednotlivé položky menu.
- Modules – název přihlašovacího modulu (Login form) a hlavního menu (Main menu).

  Pozn. Bylo nutné doplnit překlad názvů uvedených modulů i v českém jazyce
- Categories – názvy jednotlivých kategorií, především kategorie sekce Download.

Tím se zajistilo, že se po výběru anglického jazyka na všech stránkách zobrazovaly jen texty v anglickém jazyce.

6.2.5 Tvorba menu

V systému jsou implicitně předdefinovaná jednotlivá menu, která lze využít. Vzhledem k jednoduchosti celé struktury portálu byla všechna navigační menu kromě jednoho (dostupné přes **Menu->main menu**) odstraněna. Pomocí tlačítka „New“ jsem postupně vytvářela jednotlivé položky menu, které mohou odkazovat na odlišné typy objektů. Může se jednat o odkazy na nainstalované komponenty, vytvořené stránky, URL adresy na jiné webové portály, apod. Každé položce menu je možné nastavit, zda bude přístupná všem bez omezení nebo pouze registrovaným uživatelům. Jak je vidět na obrázku (Obr. 15), položka menu odkazující na diskuzní fórum a seznam uživatelů je přístupná pouze registrovaným uživatelům.



Obr. 15. Jednotlivé položky menu

6.2.6 Potřebné úpravy pro získání validního kódu

Jedním z nešvarů systému Joomla! je velké množství šablon, komponent a modulů, které nejsou validní[11]. Vzhledem k tomu, že validita kódu byla z jedním z požadovaných parametrů kladených na nový portál, bylo zapotřebí upravit zdrojové kódy u používaných komponent a modulů.

Jednalo se o velké množství úprav v HTML kódu tak, aby výsledný kód splňoval podmínky validity. Úpravy byly provedeny v komponentách Contact, JoomFish a RSGalleryě, které používaly nepodporované atributy některých tagů, chybné definice kaskádových stylů (Obr. 16), apod.

[11] Mělo by být odstraněno v nové verzi 1.5 Stable

```
369 if( $inc_jf_css && file_exists( $mosConfig_absolute_path. '/modules/mod_jflanguageselection.css' ) ) {
370     ?>
371  <link href="<?php echo $mosConfig_live_site;?>/modules/mod_jflanguageselection.css" rel="stylesheet" type="text/css" />
372     <?php
373 }
```

nesprávný zápis

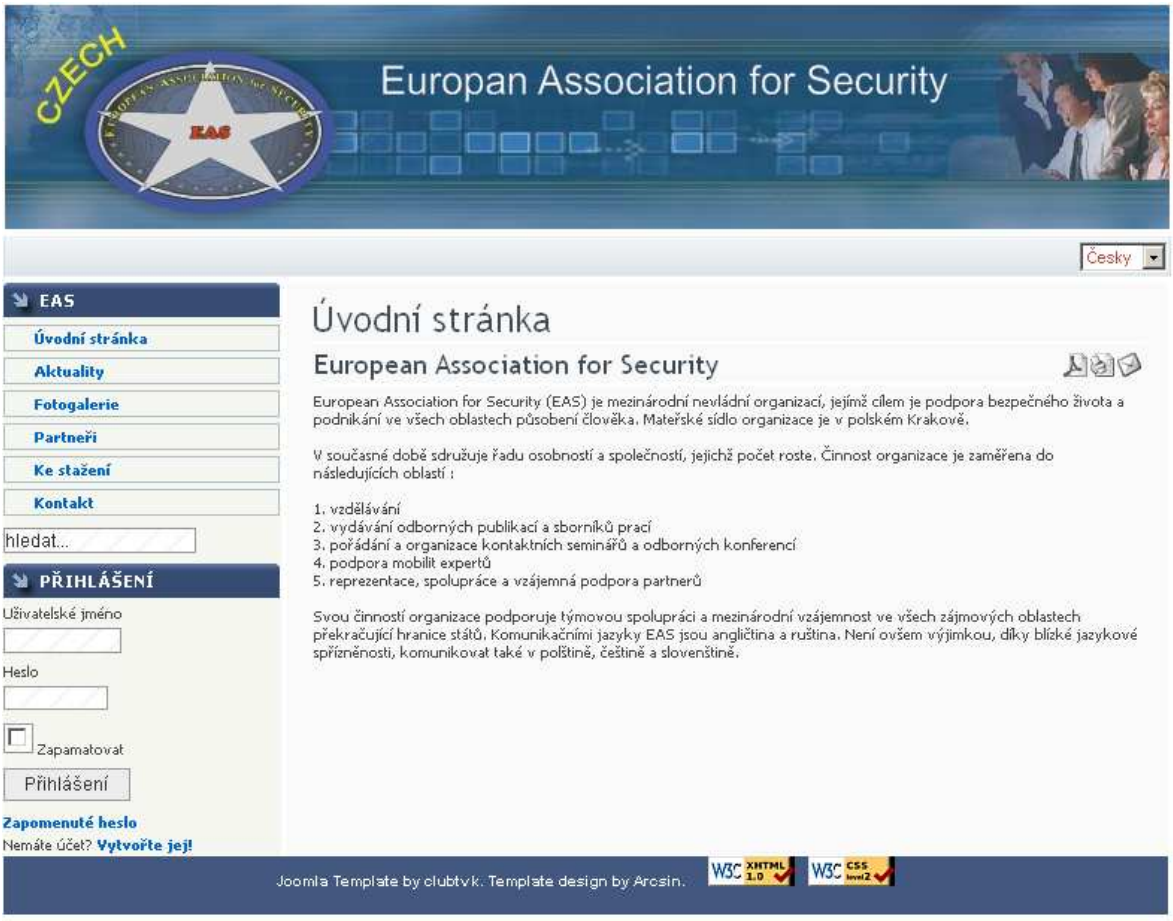
```
368
369 if( $inc_jf_css && file_exists( $mosConfig_absolute_path. '/modules/mod_jflanguageselection.css' ) ) {
370     ?>
371  <style type="text/css">@import url("<?php echo $mosConfig_live_site;?>/modules/mod_jflanguageselection.css");</style>
372     <?php
373 }
```

správný zápis

Obr. 16. Ukázka úpravy HTML kódu

6.3 Uživatelská část nového portálu

Jak je vidět na následujícím obrázku (Obr. 17) zobrazujícím domovskou stránku portálu, je v horní části obrazovky umístěné logo organizace EAS a rozbalovací menu pro volbu požadovaného jazyka. Jsou k dispozici dvě možnosti: česky a anglicky. V levé části je umístěno hlavní menu s odkazy na jednotlivé části portálu a přihlašovací formulář s možností registrace nového uživatele a přihlášení již zaregistrovaného uživatele.



Obr. 17. Domovská stránka EAS

6.3.1 Aktuality

V této části jsou zobrazeny jednotlivé články obsahující aktuální informace o konání seminářů, konferencí nebo kongresů. První se vždy zobrazí nejaktuálnější článek, na starší lze libovolně přecházet pomocí tlačítek „Další“ a „Předcházející“.

6.3.2 Fotogalerie

V současnosti je k dispozici pouze galerie obsahující fotografie z workshopu EAS konaného u příležitosti konference Internet a bezpečnost organizací. Komponenta RSGallery2 nabízí 2 možnosti prohlížení fotografií. První možností je tzv. Slideshow po jejímž výběru jsou jednotlivé snímky automaticky zobrazovány v zadaném pořadí. Druhou možností je standardní prohlížení fotografií s možností přepínání mezi jednotlivým snímky pomocí tlačítek „Předcházející“ a „Následující“.

6.3.3 Kontaktní formulář

Na stránce jsou uvedeny základní kontaktní informace: adresa, telefon, fax. Uživatel může použít kontaktní formulář (Obr. 18), pomocí kterého má možnost jednoduše a rychle zaslat zprávu na adresu organizace EAS nebo může poslat email pomocí svého poštovního klienta na uvedenou emailovou adresu eas@eas.utb.cz.

Obr. 18. Kontaktní formulář

6.3.4 Registrace a přihlášení uživatele

Na webový portál mohou přistupovat uživatelé internetu bez toho, aby se zaregistrovali. Těmto uživatelům je poskytován standardní okruh služeb:

- Možnost výběru jazyka
- Vyhledávání
- Zobrazení aktualit, fotografií a partnerů
- Zobrazení kontaktních informací, možnost využití kontaktního formuláře

Registrovaný uživatel má navíc tyto služby rozšířené o možnost:

- Přispívat do diskuzí (v menu položka „Diskuze“)
- Zobrazení registrovaných členů EAS (v menu položka „Seznam uživatelů“)

Pro registraci je nutno vyplnit jednotlivé položky registračního formuláře (jméno, uživatelské jméno, emailová adresa a heslo) a ten odeslat. V zápětí je na zadanou

emailovou adresu uživatele zaslán aktivační email obsahující odkaz, kterým se vytvořený účet aktivuje.

Na následujícím obrázku (Obr. 19) je zobrazen přihlašovací formulář, pomocí něhož se lze do systému přihlásit nebo se zaregistrovat.

Obr. 19. Přihlašovací formulář

Přihlášení do systému se provede zadáním uživatelského jména a hesla a v případě, že jsou data zadána správně, jsou v hlavním menu zpřístupněny položky „Diskuze“ a „Seznam uživatelů“. Pokud je přihlášen uživatel s právem „Super administrátor“, má tento uživatel narozdíl od ostatních uživatelů možnost editovat text jednotlivých stránek (Obr. 20).

Obr. 20. Možnost editace stránek v uživatelském rozhraní

6.3.5 Diskuzní fórum

Tato část se zpřístupní v hlavním menu po přihlášení do systému. Uživatelé mají možnost přidávat nové příspěvky do předem vytvořených kategorií či reagovat na již existující

příspěvky. Byly nadefinovány tři obecné kategorie (Obr. 21), které lze v administračním rozhraní editovat, přidat nové či odstranit stávající.

Obr. 21. Vzhled diskuzního fóra

Diskuzní fórum je implementováno komponentou Fireboard Forum, která neumožňuje vícejazyčnost a je pouze v anglickém jazyce.

6.3.6 Seznam uživatelů

Seznam zaregistrovaných uživatelů se zpřístupní v hlavním menu po přihlášení do systému. Na stránce jsou zobrazeny informace o všech zaregistrovaných uživatelích. Navíc je zde možnost vyhledání konkrétního uživatele zadáním jeho jména, příjmení či uživatelského jména (Obr. 22).

Obr. 22. Seznam zaregistrovaných uživatelů

V původní originální verzi nebyla implementovaná podpora českého jazyka. Pro její zprovoznění bylo zapotřebí vytvořit nový soubor czech.php, zapsat do něj přeložené texty jednotlivých proměnných využívaných komponentou Userlist a soubor uložit do adresáře lang dané kompomenty (eas/components/com_userlist/lang). Obsah souboru czech.php je následující:

```
<?php
DEFINE ('_USRL_USERLIST','Seznam uživatelů');
DEFINE ('_USRL_REGISTERED_USERS','%s má registrovaných uživatelů:
<b>%d</b>');
DEFINE ('_USRL_SEARCH_ALERT','Prosím zadejte hledaný výraz!');
DEFINE ('_USRL_SEARCH','Vyhledat uživatele');
DEFINE ('_USRL_SEARCH_BUTTON','Vyhledat');
DEFINE ('_USRL_LIST_ALL','Seznam všech');

DEFINE ('_USRL_NAME','Jméno');
DEFINE ('_USRL_USERNAME','Uživatelské jméno');
DEFINE ('_USRL_EMAIL','E-mail');
DEFINE ('_USRL_USERTYPE','Typ');
DEFINE ('_USRL_JOIN_DATE','Datum vytvoření');
DEFINE ('_USRL_LAST_LOGIN','Poslední přihlášení');
DEFINE ('_USRL_NEVER','Nikdy');

DEFINE ('_USRL_ASC','Vzestupně');
DEFINE ('_USRL_DESC','Sestupně');

DEFINE ('_USRL_DATE_FORMAT','%d.%m.%Y');
?>
```

6.4 Testování

Testování je velmi důležitou součástí vývoje, jehož cílem je odhalení nesprávné funkčnosti, bezpečnostních a databázových chyb a jiných nedostatků.

V průběhu vývoje byla provedena řada testů na funkčnost celého systému a nalezené chyby či problémy byly odstraněny. Testy byly zaměřeny především na:

- Funkčnost jednotlivých přidaných komponent a modulů
- Ochrana proti běžným bezpečnostním útokům
- Funkčnost jednotlivých odkazů
- Validita zdrojového kódu webových stránek a kaskádových stylů

6.4.1 Akceptační testy

Jedná se o testy, které budou provedeny po instalaci webového portálu na server utb.cz. Obecně se tyto testy provádí za účasti zákazníka, kdy je vyžadována jeho aktivní účast na tvorbě i vyhodnocení testů. Potvrzením akceptačních testů přebírá zákazník daný software.

Následující tabulky udávají přehled jednotlivých scénářů testu (Tab. 7 – 20):

Tab. 7. Testovací scénář: Zobrazení úvodní stránky

Testovací scénář: Zobrazení úvodní stránky	
ID	1
Nutné podmínky	➢ Funkční připojení k internetu
Postup testu	➢ Uživatel spustí webový prohlížeč a do adresního řádku zadá URL adresu aplikace ➢ Uživatel aktivuje svůj účet kliknutím na odkaz v emailu
Očekávané reakce	➢ Zobrazí se úvodní stránka EAS

Tab. 8. Testovací scénář: Prohlížení jednotlivých aktualit

Testovací scénář: Prohlížení jednotlivých aktualit	
ID	2
Nutné podmínky	➢ Funkční připojení k internetu
Postup testu	➢ Uživatel vyvolá akci zobrazení aktualit ➢ Uživatel prohlíží jednotlivé aktuality pomocí tlačítek „Následující“ a „Předcházející“
Očekávané reakce	➢ Na základě uživatelského požadavku se zobrazí aktualita

Tab. 9. Testovací scénář: Prohlížení fotografií v galerii

Testovací scénář: Prohlížení fotografií v galerii	
ID	3
Nutné podmínky	➢ Funkční připojení k internetu
Postup testu	➢ Uživatel vyvolá akci pro zobrazení fotogalerií ➢ Uživatel zvolí galerii a vyvolá akci zobrazení náhledu všech fotografií dané galerie ➢ Uživatel zvolí náhodně jednu z fotografií a prohlíží ostatní pomocí tlačítek „Předchozí“ a „Následující“
Očekávané reakce	➢ Postupné zobrazování požadovaných fotografií

Tab. 10. Testovací scénář: Zobrazení úvodní stránky partnera

Testovací scénář: Zobrazení úvodní stránky partnera	
ID	4
Nutné podmínky	➢ Funkční připojení k internetu
Postup testu	➢ Uživatel vyvolá akci pro zobrazení partnerů ➢ Uživatel otevře pomocí odkazu stránku libovolného partnera
Očekávané reakce	➢ V novém okně se otevře domovská stránka vybraného partnera

Tab. 11. Testovací scénář: Stáhnutí dokumentu

Testovací scénář: Stáhnutí dokumentu	
ID	5
Nutné podmínky	➢ Funkční připojení k internetu
Postup testu	➢ Uživatel vyvolá akci pro zobrazení dokumentů ke stažení ➢ Uživatel zvolí z hlavního menu libovolnou kategorii dokumentů ➢ Uživatel stáhne pomocí odkazu vybraný dokument
Očekávané reakce	➢ Zobrazení dokumentu v novém okně nebo zobrazení okna s výběrem, zda dokument uložit či otevřít

Tab. 12. Testovací scénář: Odeslání zprávy přes kontaktní formulář

Testovací scénář: Odeslání zprávy přes kontaktní formulář	
ID	6
Nutné podmínky	➢ Funkční připojení k internetu
Postup testu	➢ Uživatel vyvolá akci pro zobrazení kontaktích informací ➢ Uživatel vyvolá akci pro zobrazení kontaktního formuláře ➢ Uživatel vyplní údaje v kontaktním formuláři a tyto data odešle
Očekávané reakce	➢ V případě korektních údajů je na adreasu organizace EAS odeslán email, v opačném případě je zobrazeno chybové hlášení

Tab. 13. Testovací scénář: Zobrazení vyhledané stránky

Testovací scénář: Zobrazení vyhledané stránky	
ID	7
Nutné podmínky	➢ Funkční připojení k internetu
Postup testu	➢ Uživatel zadá hledaný výraz do textového pole pro vyhledání a potvrdí tlačítkem Enter ➢ V případě nalezení minimálně jedné stránky uživatel vyvolá akci zobrazení stránky pomocí odkazu, v opačném případě uživatel zadá do textového pole jiný výraz pro vyhledání
Očekávané reakce	➢ Zobrazení vyhledané stránky

Tab. 14. Testovací scénář: Registrace nového uživatele

Testovací scénář: Registrace nového uživatele	
ID	8
Nutné podmínky	➢ Funkční připojení k internetu
Postup testu	➢ Uživatel vyvolá akci pro zobrazení registračního formuláře ➢ Uživatel vyplní údaje v registračním formuláři a tyto data odešle ➢ V případě korektních údajů obdrží uživatel aktivační email, ve kterém kliknutím na odkaz aktivuje svůj účet, v opačném případě uživatel opraví zadané údaje v registračním formuláři
Očekávané reakce	➢ Uživateli je umožněno přihlásit se do systému

Tab. 15. Testovací scénář: Přihlášení uživatele do systému

Testovací scénář: Přihlášení uživatele do systému	
ID	9
Nutné podmínky	➢ Funkční připojení k internetu ➢ Existující účet uživatele
Postup testu	➢ Uživatel vyplní přihlašovací jméno a heslo ➢ Uživatel vyvolá akci na přihlášení do systému
Očekávané reakce	➢ V případě korektních přihlašovacích údajů bude provedeno přihlášení, v opačném případě se zobrazí chybového hlášení

Tab. 16. Testovací scénář: Obdržení zapomenutého hesla emailem

Testovací scénář: Obdržení zapomenutého hesla emailem	
ID	10
Nutné podmínky	➢ Funkční připojení k internetu
Postup testu	➢ Uživatel vyvolá akci pro zaslání zapomenutého hesla ➢ Uživatel zadá uživatelské jméno a email a tyto data odešle
Očekávané reakce	➢ V případě korektních údajů obdrží uživatel email obsahující heslo, v opačném případě se zobrazí chybové hlášení

Tab. 17. Testovací scénář: Přidání příspěvku do diskuze

Testovací scénář: Přidání příspěvku do diskuze	
ID	11
Nutné podmínky	➢ Funkční připojení k internetu ➢ Přihlášený uživatel
Postup testu	➢ Uživatel vyvolá akci pro zobrazení diskuze ➢ Uživatel zvolí libovolné téma diskuze a vyvolá akci pro přidání příspěvku ➢ Uživatel vyvolá akci pro uložení příspěvku
Očekávané reakce	➢ Uložení příspěvku do diskuze

Tab. 18. Testovací scénář: Vyhledání zaregistrovaného uživatele

Testovací scénář: Vyhledání zaregistrovaného uživatele	
ID	12
Nutné podmínky	➢ Funkční připojení k internetu ➢ Přihlášený uživatel
Postup testu	➢ Uživatel vyvolá akci na zobrazení seznamu uživatelů ➢ Uživatel zadá jméno, příjmení či uživatelské jméno do textového pole pro vyhledání
Očekávané reakce	➢ V případě nalezení uživatele, zobrazení jeho základních informací, v opačném případě zobrazení prázdného seznamu

Tab. 19. Testovací scénář: Odhlášení uživatele ze systému

Testovací scénář: Odhlášení uživatele ze systému	
ID	13
Nutné podmínky	➢ Funkční připojení k internetu ➢ Přihlášený uživatel
Postup testu	➢ Uživatel vyvolá akci na odhlášení ze systému
Očekávané reakce	➢ Uživatel je odhlášen

Tab. 20. Testovací scénář: Přepnutí jazyka uživatelského rozhraní

Testovací scénář: Přepnutí jazyka uživatelského rozhraní	
ID	14
Nutné podmínky	➢ Funkční připojení k internetu
Postup testu	➢ Uživatel vybere v rozbalovacím seznamu zvolený jazyk ➢ Uživatel vyvolá akci zobrazení libovolné stránky
Očekávané reakce	➢ Zobrazení stránky ve zvoleném jazyce

6.5 Budoucnost webového portálu

Webový portál byl vytvořen ve starší, ale osvědčené verzi 1.0.15 redakčního systému Joomla!. Důvodem pro nezvolení nejnovější verze 1.5 byl nedostatek komponent, modulů a šablon v době implementace. Předpokládá se, že by stávající verze byla nahrazena novější verzí, která nabízí mimo jiné výhradní podporu kódování UTF-8[12], kompletní lokalizaci, dokonalejší mechanismus pro vytváření SEO přátelských URL adres, nové administrační rozhraní s nižší citlivostí na nastavení hostingu, má zabudovanou FTP vrstvu (obdoba FTP klienta) apod.

Pro zvýšení návštěvnosti nestačí jen upravení stránek, důležité je také budování zpětných odkazů, tzn. zajistit co nejvíce odkazů z cizích stránek na portál EAS.

[12] Starší verze sice podporuje kódování UTF-8 také, ale je problémové

ZÁVĚR

Tato diplomová práce se zabývá moderními webovými technologiemi s ohledem na jejich bezpečnost a dostupnost. Vytyčené cíle práce byly naplněny, výsledkem je plně funkční webový portál připravený k nasazení.

V teoretické části diplomové práce jsou uvedeny základní informace o značkovacích a skriptovacích jazycích, které jsou základem každé webové aplikace. Dále jsou zde stručně popsány hlavní zásady, které by při tvorbě webových aplikací měly být dodrženy. Jedná se především o validní HTML kód, oddělení obsahu stránek od jeho formátování, jsou zde objasněny pojmy jako jsou SEO, sémantická správnost, přístupnost a použitelnost stránek. V poslední kapitole uvádím nejčastější bezpečností rizika, se kterými se u webových aplikací lze setkat.

Praktická část diplomové práce se skládá z několika dílčích celků. Nejprve bylo zapotřebí provést analýzu a zhodnocení stávajícího systému, na jehož základě bylo možné definovat požadavky kladené na nový portál. Tyto požadavky se staly vstupními parametry pro srovnání zvolených redakčních systémů. Následovalo hledání odpovědí, zda daný redakční systém splňuje jednotlivé požadavky či nikoliv. Tato část byla časově velice náročná, protože bylo zapotřebí potřebné informace nejprve získat (nejčastěji z internetových zdrojů) a poté je ověřit. V posledním kroce byl zvolen jeden redakční systém, ve kterém byl informační portál EAS implementován. V praktické je také popsána instalace zvoleného redakčního systému, postup tvorby informačního portálu a v závěru jsou nastíněny možnosti budoucího rozvoje portálu.

V diplomové práci jsem uplatnila již získané zkušenosti a znalosti z praxe, a zároveň jsem i nové zkušenosti z oblasti webových technologií získala. Přínosem také bylo to, že jsem měla možnost při vytváření portálu projít si všemi fázemi projektu.

ZÁVĚR V ANGLIČTINĚ

This diploma work deals with modern web technologies with regards to their availability and security Established targets were fulfilled, resulting in fully functional web portal ready to be deployed.

In theoretical part of diploma work reader can find basic information about markup and scripting languages, which are ground of each web application, next part deals with description of main rules, which should be obeyed in every web application, most of all validity of produced HTML, separation of contents and formatting, explanation of SEO, semantical rightness, accessibility and application usefulness. In last chapter I state most frequent security hazards, which can be found in web application.

Practical part of diploma work is consisting of some more parts. First, it was necessary to make analysis and evaluation of existing system, which would lead into definition expected from new portal application. These requirements became input parameters for comparison of selected editorial systems. Then I tried to find answer, whether fulfills requirements. This part was highly time consuming, because it was necessary to obtain reliable data (mostly from internet) and then verify them, during last step I selected one editorial system, which included implementation of EAS. In practical part it is as well described installation of chosen editorial system and progress of setting up of informational portal. On the end of this chapter, I outline possibilities in further development of portal.

In diploma work I used already obtained experience and knowledge from practice and I also obtained new experience from area of web technologies. It was also advantageous, to pass by all phases of project.

SEZNAM POUŽITÉ LITERATURY

[1] Programování stránek [online]. [cit. 2007-12-10]. Dostupný z WWW: <http://www.jakpsatweb.cz/programovani.html>.

[2] KOSEK, Jiří. HTML : Tvorba dokonalých WWW stránek. 1. vyd. Praha : Grada Publishing, 1998. 291 s. ISBN 80-7169-608-0.

[3] *Co je to AJAX?* [online]. [cit. 2008-03-10]. Dostupný z WWW: <http://webing.felk.cvut.cz/hs/download/DT-ajax-CZ-art.pdf>.

[4] *Tvorba moderního webu - syntaxe, sémantika, CSS, SEO, přístupnost, použitelnost* [online]. [cit. 2008-03-12]. Dostupný z WWW:

<http://www.samuraj-cz.com/clanek/tvorba-moderniho-webu-syntaxe-semantika-css-seo-pristupnost-pouzitelnost/>.

[5] *Webové standardy – více než jen beztabulkový web* [online]. [cit. 2008-03-15]. Dostupný z WWW: <http://www.vitdlouhy.cz/clanky/webove-standardy.php>.

[6] *Bezpečnost webových aplikací* [online]. [cit. 2008-04-02]. Dostupný z WWW: <http://www.webfox.cz/clanky/praxe/bezpecnost-webovych-aplikaci.php>.

[7] *Zabezpečení webových aplikací II. – databáze web* [online]. [cit. 2008-04-02]. Dostupný z WWW:

<http://access.feld.cvut.cz/view.php?nazevclanku=zabezpeceni-webovych-aplikaci-ii-databaze&cisloclanku=2007080002>.

[8] *2007 Open Source CMS Award* [online]. [cit. 2008-04-21]. Dostupný z WWW: <http://www.packtpub.com/award>.

[9] *Open Source CMS – Try Before You Install* [online]. [cit. 2008-04-22]. Dostupný z WWW: <http://www.opensourcecms.com/>.

[10] *Joomla pro totalni zelenace 1.1* [online]. [cit. 2008-04-28]. Dostupný z WWW: <http://www.joomlaportal.cz/component/option,com_remository/Itemid,64/func,fileinfo/id,90//>.

[11] ŠTĚDROŇ, Bohumír. *Manažerské řízení a informační technologie.* 1. vyd. Praha : Grada Publishing, a. s., 2007. 156 s. ISBN 978-80-247-2052-4.

[12] DELLWIG, Ingo. *HTML 4 : příručka tvůrce webu.* Přeložil Vladimír Lahoda. 1. vyd. Praha : Grada Publishing, spol s.r.o., 2002. 272 s. ISBN 80-247-0297-5.

[13] ENDORF, Carl, SCHULTZ, Eugene, MELLANDER, Jim. *Hacking - Detekce a prevence počítačového útoku.* Praha : Grada, 2005. 356 s. ISBN 80-247-1035-8.

SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK

ASP	Active Server Pages - technologie umožňující vykonávání kódu na straně serveru a následné odeslání výsledku uživateli
CMS	Content Management System - systém pro správu obsahu
CSS	Cascading Style Sheet - jazyk pro popis způsobu zobrazených webových stránek napsaných v jazycích HTML, XHTML nebo XML
DHTML	Dynamic HTML - kombinace technologií používaných ke tvorbě dynamických a interaktivních webových stránek
FTP	File Transfer Protocol - protokol aplikační vrstvy pro přenos souborů mezi počítači
HTML	HyperText Markup Language - značkovací jazyk pro tvorbu webových stránek
HTTP	HyperText Transfer Protocol - internetový protokol pro anonymní výměnu požadavků a odpovědí mezi klientem a serverem
JSP	Java Server Pages - technologie umožňující vkládání Java kódu přímo do HTML stránky
JVM	Java Virtual Machine - interpret Javy (virtuální stroj)
PHP	Hypertext Preprocessor - skriptovací programovací jazyk především určený pro programování dynamických webových stránek
RSS	Really Simple Syndication - univerzální formát pro výměnu a šíření obsahu webových stránek (tzv. syndikaci obsahu)
SEO	Search Engine Optimalization - optimalizace pro vyhledávače
SSI	Server Side Includes - jeden z nejrozšířenějších druhů serverem vkládaných vsuvek
URL	Uniform Resource Locator - používá se pro přesnou identifikaci dokumentů na internetu

UTF-8	UCS Transformation Format - znaková sada, umožňující zápis všech národních abeced na celém světě
W3C	World Wide Web Consortium - organizace zabývající se standardy a normami webových technologií
WYSIWYG	What You See Is What You Get - způsob editace dokumentů, při kterém je text zobrazen na obrazovce tak, jak bude vytištěn na papír
XHTML	Extensible Hyperext Markup Language - značkovací jazyk pro tvorbu webových stránek s podporou XML
XML	Extensible Markup Language - obecný značkovací jazyk poskytující datový formát pro strukturované dokumenty

SEZNAM OBRÁZKŮ

SEZNAM TABULEK

SEZNAM PŘÍLOH

PŘÍLOHA P I: POSTUP INSTALACE NA SERVER UTB

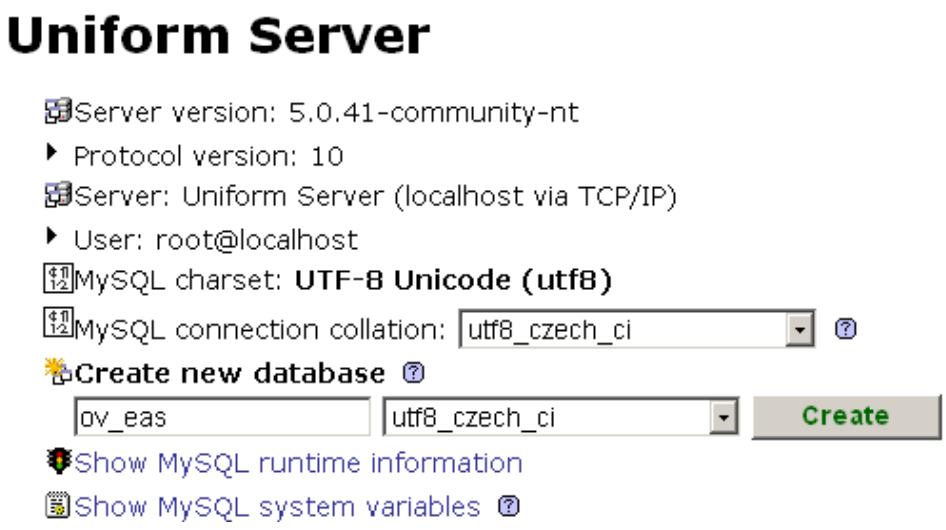
Instalace webového portálu na server UTB bude provedena ve čtyřech krocích:

1. Přenesení adresáře *eas*, který se nachází na CD v adresáři *installation/app*, do kořenového adresáře serveru
2. Upravení hodnot proměnných v souboru *configuration.php*. Jedná se především o nastavení:

```
* ---------------------------------------------------------------------
* Database configuration section
* ---------------------------------------------------------------------
*/
$mosConfig_host = 'localhost';      // e.g. mysql.utb.cz
$mosConfig_user = 'root';           // MySQL username
$mosConfig_password = 'root';       // MySQL password
$mosConfig_db = 'ov_eas';           // MySQL database name
$mosConfig_dbprefix = 'jos_';       // Don't change unless you need to!

* ---------------------------------------------------------------------
* Site specific configuration
* ---------------------------------------------------------------------
*/
$mosConfig_absolute_path = 'W:/www/eas';    // No trailing slash
$mosConfig_cachepath = 'W:/www/eas/cache'; // No trailing slash
$mosConfig_live_site = 'http://localhost/eas';   // No trailing slash
```

3. V phpMyAdmin na serveru vytvořit databázi s kódováním UTF-8, jak je uvedeno na následujícím obrázku:



4. V phpMyAdmin provést import souboru *import.sql* umístěného v adresáři *installation/sql*.