INFORMAČNÍ SYSTÉMY NA WEBU

Webový informační systém je systém navržený pro provoz v podmínkách Internetu/intranetu, tzn. přístup na takový systém je realizován přes internetový prohlížeč.

Použití internetového prohlížeče jakožto nástroje pro práci s IS přináší některé odlišnosti. Asi nejdůležitějším důsledkem, zejména pro tvůrce aplikací, je provoz v **nelineárním nestavovém síťovém prostředí.**

V čem spočívá nelinearita systému?

- Vstup v neočekávaném bodu (přímé zadání URL)
- Návrat v posloupnosti operací (tlačítko „Zpět“ v prohlížeči)
- Opakování požadavku (tlačítko „Obnovit“) s opětovným zasláním požadavků

Internet je na aplikační vrstvě realizován prostřednictvím protokolu http, který je bezestavový (tzn. po návratu řízení na konkrétní stránky http si systém nepamatuje, v jakém stavu se stránka nacházela. Volání jednotlivých stránek znamená izolované přístupy do operačního systému. Zpracování informací se tedy děje v určitém množství samostatných kroků a pro tvůrce systému to znamená zajistit neustálé předávání dat během volání jednotlivých stránek.

PERSONALIZACE IS

Personalizací rozumíme možnost přizpůsobení informačního systému potřebám a zvyklostem uživatelů:

- Definice vzhledu systému
- Nastavení systému prostřednictvím systémových politik
- Vytvoření vlastní uživatelské aplikace prostřednictvím předdefinovaných komponent

AUDITOVÁNÍ A LOGOVÁNÍ

Auditování – evidence změn prováděných v informačním systému

Logování – sledování posloupnosti práce jednotlivých uživatelů, jejich aktivity

SESSIONS

HTTP je bezestavový protokol, server odpovídá na požadavky klienta aniž by je dával do souvislostí.

Pro fungování aplikací byl zaveden mechanismus **sessions**.

Po připojení na sessions je vytvořen jedinečný identifikátor (session token).

Session token se odešle ke klientovi (uložení v podobě „cookie“) a je posílán serveru s každým požadavkem.

Poznámka: měly by se, z bezpečnostních důvodů, používat neperzistentní cookies, aby po zavření prohlížeče session token nebyl k dispozici.

VALIDITA STRÁNEK

- Korektní značkování – vliv na umístění stránky ve výsledcích vyhledávání
- SEO:
  - vliv mají i slova umístěná v URL stránky
  - Nativní adresa (=vidím parametry PHP)
  - Permanentní adresa (maskování, výstižnost)

FUNKČNÍ POŽADAVKY NA WEBOVÉ APLIKACE:

1. vhodná architektura aplikace
2. modularita
3. zabezpečení aplikace
4. validace vstupních parametrů
5. uživatelské autentizace a správa uživatelů
6. sessions a entitní autentizace
7. autorizace a přístupová práva
8. jednotná databázová vrstva
9. logování událostí a akcí
10. validita výstupních stánek
11. maskování URL
12. bezpečnost (injection, cross site scripting, …)

ARCHITEKTURA WEBOVÉ APLIKACE

Model 1 – prohlížeč přistupuje ke stránkám přímo; stránka zpracovává vstupy od klienta v parametrech GET nebo POST

Model 2 – využívá tzv. „controller“ – nachází se mezi prohlížečem a volanými stránkami/skripty

Při vývoji webových aplikací se často používají knihovny zajišťující komplexní funkcionalitu pro vývoj webové aplikace – označují se jako **framework**.

Framework by měl umožnit členění aplikace do logických částí (modulů).

Příklady frameworků: Zend framework, nette, …

BEZPEČNOST

BEZPEČNOSTNÍ RIZIKA

- Špatné programovací postupy
  - např. absence kontrol hodnot proměnných, chybějící inicializace
- Chybná analýza
- Nedostatky webových prohlížečů

APLIKAČNÍ BEZPEČNOST

- Validace vstupů a výstupů
- Bezpečné selhání (konzistentnost dat po selhání)
- Jednoduchý bezpečnostní mechanismus
- Použití prověřených komponent
- Plánování neočekávaných událostí
- Co nejmenší oprávnění uživatele

AUTENTIZACE UŽIVATELE

- Na základě znalosti určité informace (heslo)
- Na základě vlastnictví určité věci (např. čipová karta)
- Biometrické metody

BEZPEČNOSTNÍ DOPORUČENÍ PRO FORMULÁŘE

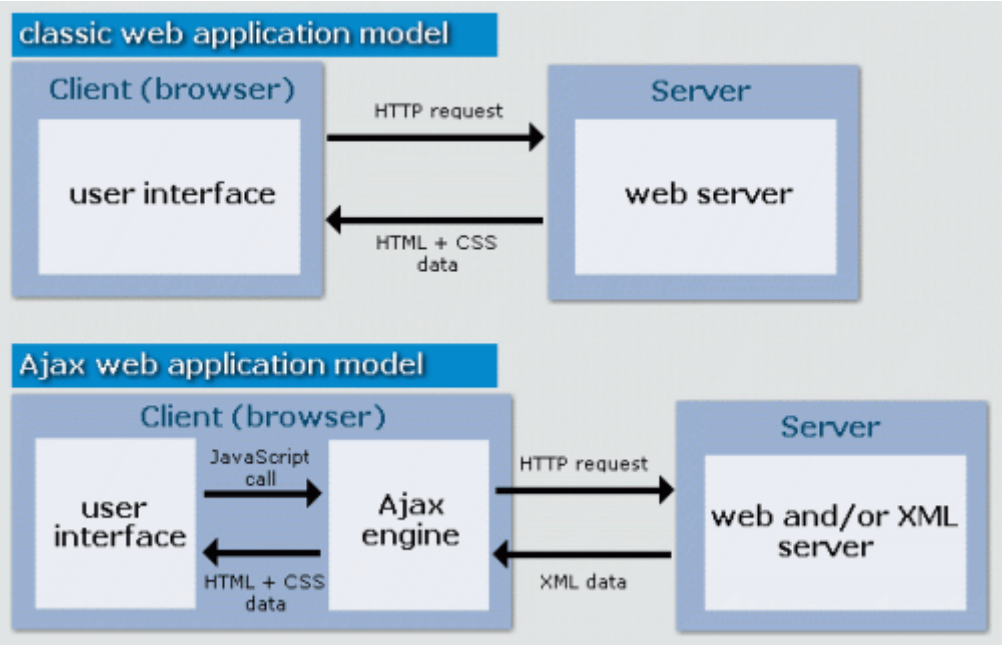
- Zadání hesla přes vstupní pole typu password
- Data odesílat výhradně metodou POST
- Zabezpečená komunikace (SSL)

SPRÁVA UŽIVATELŮ A HESEL

- Zapomenuté heslo (dodatečné zjištění zodpovězením kontrolní otázky, Zasílání přístupových údajů e-mailem)
- Omezená životnost hesel
- Zamknutí účtu po opakovaném neúspěšném přihlašování

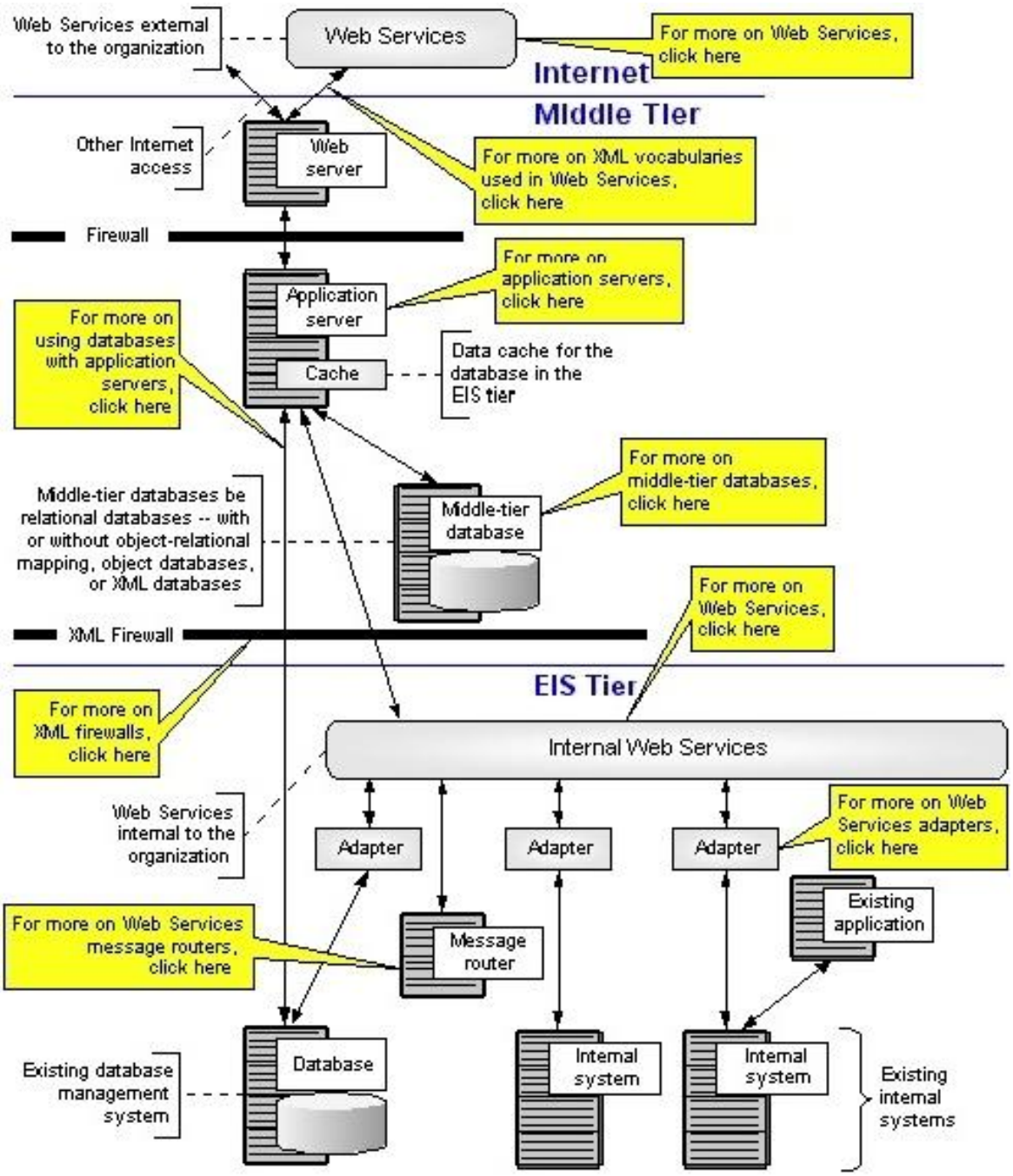
TECHNOLOGIE A POJMY

- HTML – hypertextový značkovací jazyk, základ aplikací na webu, funguje formou žádost-odpověď, bezestavový
- CSS (Cascading Style Sheets) – kaskádové styly pro formátování stránek v html
- PHP, ASP, .NET, JSP, Python, Ruby, CGI, Perl – skripty na straně serveru
- XML – obecný značkovací jazyk, použití např. pro výměnu dat
- JavaScript, Jscript, VBScript, DHTML, ActiveX – skripty na straně klienta
- AJAX (Asynchronní JavaScript aXML) – soubor technologií, které mění obsah webových stránek bez nutnosti jejich znovunačítání
- XHTML (eXtensible HTML) – rozšíření HTML, využívá XML
- Flash – grafický vektorový program pro tvorbu interaktivních animací, prezentací atd., využití formou plug-inu v internetových prohlížečích pro práci s grafikou, animacemi apod.
- Silverlight (Moonlight) – Microsoft alternativa k technologii Flash
- SOA (Service Oriented Architecture) - servisně orientovaná architektura, aplikace se skládá ze skupiny služeb, které komunikují mezi sebou.
- RIA (Rich Internet Application) – řešení internetové aplikace, které odstraňuje nedostatkyhtml protokolu (založeno na modelu žádost-odpověď), využití např. AJAX technologie, Flex, OpenLaszlo apod.
- Web 2.0 – označení pro aplikace na webu, kde obsah spoluvytvářejí uživatelé (např. YouTube, Facebook,…)
- Meshup – služba založená na využití existujících služeb (např. využití služby Google Maps v aplikaci pro prezentování pozice konkrétních objektů v mapách)



Obr. 1. Srovnání klasického modelu aplikace a aplikace s technologií AJAX

Zdroj: http://interval.cz/clanky/rich-internet-application/



Obr. 2. Ukázka SOA architektury

Zdroj: http://www.service-architecture.com/