
Semestrální práce - REST kurzy

A4M33AOS

Tomáš Lucovič (lucovtom@fel.cvut.cz)

OTEVŘENÁ INFORMATIKA • FEL ČVUT • 15.PROSINEC 2010

Popis práce

Phasellus ut arcu ut nulla dignissim auctor

ÚVOD

Účelem mé semestrální práce byla implementace RESTful Web Services (JAX-RS).

Samotné REST rozhaní navrhl v roce 2000 Roy Fielding (jeden ze spoluautorů protokolu http). Je využíváno především pro snadný přístup ke zdrojům, kde každý zdroj má vlastní URI a čtyři metody jak k němu přistupovat.

REST je na rozdíl od známějších XML-RPC či SOAP, orientován datově, nikoli procedurálně. (nerozlišuje mezi daty a stavy aplikace)

Základní principy RESTu

- stav aplikace a chování je vyjádřen takzvaným resourcem (klíčová abstrakce)
- každý resource má unikátní identifikátor (URL, URN)
- je definován jednotný přístup pro získání a manipulaci s resourcem v podobě čtyř operací CRUD (Create, Read, Update, Delete)
- resource může mít různé reprezentace (XML, HTML, JSON, SVG, PDF)

Ve své práci jsem využil jen dvě rest metody Create a Read. Jejich volání je pomocí http metod GET a PUT.

Implementaci jsem prováděl v jazyku Java Enterprise Edition 6, která je určena k tvorbě náročných webových aplikací.

Využité komponenty jsou všechny obsaženy ve vývojovém prostředí NetBeans IDE, které mi poskytlo veškeré potřebné nástroje.

ZADÁNÍ PRÁCE

Pro svou semestrální práci jsem si po domluvě se svým cvičícím vybral implementaci aplikace na sledování kurzů ČNB. Samotná aplikace se skládá ze 3 projektů.

1. Hlavní je **aplikační server**, který má perzistentní úložiště (DB) . V tomto úložišti je uložena mezipaměť s kurzy ČNB. Server poskytuje RESTFull rozhraní pro přístup k datům z databáze a na základě požadavku na příslušný resource aktualizuje kurz ČNB k danému datu.
2. Klientské rozhraní - bude poskytovat uživateli možnost pracovat se službami, registrovat sledování určité měny, zjišťování hodnoty určité měny.
3. Rozhraní pro stahování dat ČNB - banka neposkytuje žádný ze standardních formátů webových služeb. Pouze umožňuje stáhnout textový soubor, ve kterém jsou data oddělena pipou | .

Uživatelské scénáře:

Uživatel si může zobrazit kurz EUR, USD k vybranému datu.

Uživatel může vložit banky KB, ČNB k vybranému datu v měně EUR, nebo USD.

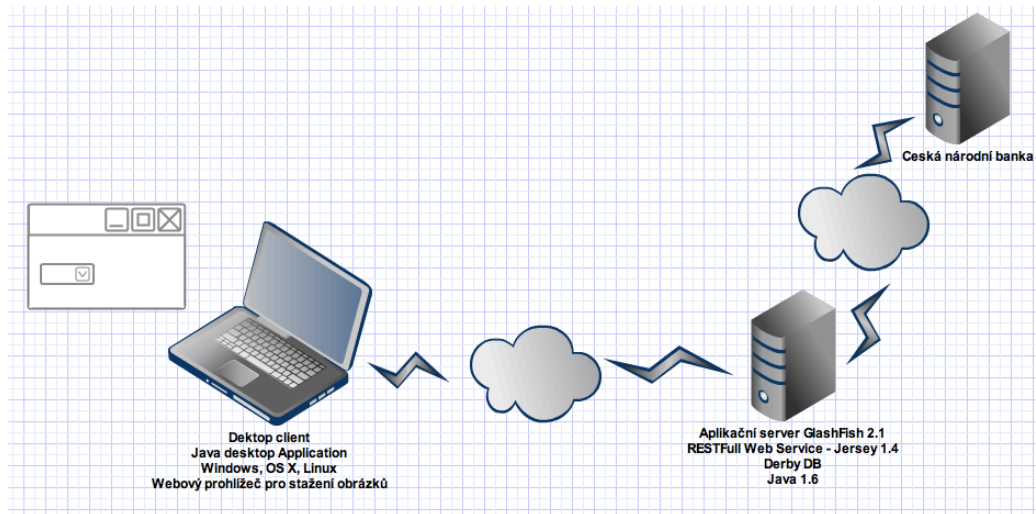
Uživatel si může zobrazit seznam svých hlídaných kurzů.

Uživatel může vložit nový hlídaný kurz k měně EUR, USD.

Uživatel si může zobrazit seznam svých nákupů

Systém při zjištění zápisu nové měny na perzistentní úložiště, připíše 100 měny uživateli, pokud kurz k dané měně je menší než požadovaná hodnota uživatelem.

POPIS INFRASTRUKTURY



Uživatelská aplikace komunikuje s Aplikačním serverem pomocí rest rozhraní. (přistupuje na předem stanovená URL. Podle požadavky na daný zdroj je určeno, zda-li se jedná o požadavek na čtení nebo zápis dat.

Aplikační server poskytuje REST služby klientům. V případě požadavku klienta na aktualizaci kurzů, provede pokus o jejich update ze serverů ČNB. V případě úspěšného updatu se kurzy uloží do databáze.

Servery ČNB poskytují textové soubory s kurzy měn.

Veškerá komunikace je pomocí http protokolu, data jsou zakódována do URL nebo přenášena pomocí XML souboru

POPIS ROZHRAŇÍ

REST rozhraní je implementováno v projektu Jersey 1.4 , který základní instalaci obsahuje
Aplikační server GlashFish

Po nasazení(deploy) aplikace do aplikačního serveru jsou veškeré zdroje dostupné na této adrese.

PATH =

<http://localhost:8080/Server/aos/rate/>

Jedná se o testovací rozhraní, v produkci si můžete pod nadefinovat root URL libovolně např.
podle Vámi koupené domény.

Za touto adresou jsou již dostupné zdroje.

V celé aplikaci je použita datumová konvence.

Konvence pro zdávání hodnot:

- DATUM - formát YYYY-MM-DD, např 2010-02-05
- MĚNA - uvádí se v 3 písmeně zkratce např. EUR, USD
- KURZ - se uvádí s desetinou tečkou na 2 desetinná místa např 17.34

Rate(kurzy)

Pro přístup ke kurzům je možno využít rozhraní pro dané dny:

GET

PATH/{date} kdy za {date} napíšete datum ke kterému Vás dané kurzy zajímají.

PATH/{date}/{bank} pokud dotaz rozšíříte o banku, získáte kurzy jen jedné určité banky

PATH/{date}/{date}/{bank}/{currency} - při rozšíření získáte přesně jeden kurz

Odpověď: XML se seznamem kurzů

PUT

PATH/{date}/{bank}/{currency}/{rate} -pro zdání kurzu stačí přistoupit metodou PUT na příslušné URL a informace se uloží do databáze.

Watch

slouží ke sledování požadovaných změn kurzů a pro vyhodnocení případného nákupu

GET

PATH/user/{owner} - vrátí seznam kurzů, který uživatel přál hlídat

Odpověď: XML seznam hlídek

PUT

PATH/{user}/{currency}/{watchrate} = pro zdání hlídky stačí přistoupit metodou PUT na příslušné URL a informace se uloží do databáze.

Image

slouží ke spřístupnění obrázků

GET

PATH/images/{image} - pokud na adresu zadáte název obrázku, tak jej můžete stáhnout do svého prohlížeče

Update

URL na které se v případě přistoupení metodou GET vyvolá funkce aktualizace kurzů.

GET

PATH/update/CNB/{date} -stačí zadat datum, ke kterému si přejeme provést aktualizaci.

ZÁVĚREČNÉ ZHODNOCENÍ

Celková implementace je proveden jen za pomoci vývojového prostředí NetBeans IDE, které mi poskytlo dostačující vývojové nástroje.

Jedná se o moje první seznámení s RESTFull webovými službami a shledávám je jednoduché na pochopení. Náročnost jejich implementace je závislá na programovacím jazyku.

Z mého pohledu by pohodlnější práce tímto typem webových služeb byla pomocí skriptovacích programovacích jazyků RUBY, PHP, PYTHON atd.

K napsání klienta nejsou potřeba žádné znalosti navíc, stačí pouze znalost HTTP a XML

Jako nevýhodu shledávám menší datovou typovost, což se při mé práci v mnoha případech bylo se silnou typovostí programovacího jazyka java.

Rest rozhraní bych doporučil tam, kde je upřednostňována jednoduchost před důrazem správně zadané datové typy.

Z bezpečnostního hlediska REST rozhraní poskytuje i SSL verzi s ověřením, tudíž bych jej doporučil i do komerčního prostředí.

REST rozhraní pro čtení dat ve svém základu poskytuje např. dokumentová databáze Mongo DB