Západočeská univerzita v plzni

Fakulta aplikovaných věd

Katedra informatiky a výpočetní techniky

**Semestrální práce**

**KIV/PIA**

**Standardní zadání - Twitter**

Plzeň, 2012

Martin Bydžovský

A12N0058P

# Zadání

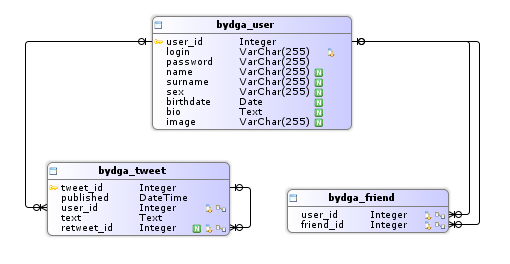
Mírně zjednodušený Twitter. Přihlášený uživatel bude moci odeslat nový tweet, vidět přehledně, co píší lidé, které followuje a retweetnout existující tweet. Retweet bude realizován novým způsobem, tj. bude zachován původní autor a bude u daného tweetu napsáno, že retweetnuto někým jiným.

# Architektura aplikace

Aplikace je postavena na standardním MVC patternu. Datovou vrstvu zde obstarávají DAO třídy, Řízení logiky aplikace obstarávají servlety a prezentační vrstvu pak JSP stránky

# Datový model

Databáze se skládá ze tří tabulek, jejich název, sloupce a vazby demonstruje následující obrázek:



# DAO přístup

Komunikace s databází je centralizována na jedno místo proto, aby jednotlivé SQL dotazy nebyly rozesety všude možně po aplikaci. Tyto třídy existují celkem dvě – UserDAO a TweetDAO. Zastřešují práci s entitami uživatel a tweet. Vstupní a výstupní parametry metod těchto tříd jsou většinou objekty User a Tweet (případně jejich seznamy) z business logiky, nad kterými chceme vykonávat danou operaci (tvorba/úprava/mazání), případně id objektu, který chceme z databáze získat.

Protože tyto DAO třídy vyžadují přístup do databáze, je zde aplikovaný design pattern Dependency Injection – tzn. toto připojení je jim předáváno v konstruktoru – a o jeho získání se stará ten, kdo DAO službu volá (typicky servlet).

# Servlety

Servlety tvoří business logiku aplikace. Ty aplikace obsahuje tři celkem, přičemž všechny dědí od společné třídy BaseServlet, která obstarává společnou funkcionalitu pro servlety této aplikace.

## BaseServlet

Společná abstraktní třída pro servlety tohoto projektu.

## LoginServlet

Obstarává práci kolem uživatelských účtů. Zobrazuje přihlašovací/registrační formulář, dává do něj captchu a spravuje jejich odeslání. Takže vytváří nově registrované uživatele a ukládá informace do session při přihlášení /maže při odhlášení.

## UserServlet

Zobrazuje přehledovou stránku uživatelského účtu – přičemž jako parametr z URL bere id zobrazovaného profilu. Na této stránce uživatele je stránkovaný výpis jeho tweetů, vyplněné informace o uživateli, seznam jeho followerů a lidí, které on followuje. Pokud je v aplikaci aktuálně někdo přihlášený, zobrazují se taktéž ovládací prvky pro follow/unfollow.

Dále zobrazuje přehledovou stránku s výpisem všech uživatelů. V neposlední řadě také obstarává přidávání a odebírání followerů.

## StreamServlet

Obstarává zobrazení uživatelského streamu příspěvků (stránkovaný výpis) – tweety aktuálně přihlášeného uživatele kombinované s tweety lidí, které on followuje. Výpis je potom řazen chronologicky. Tento servlet také přijímá text nově vytvářeného tweetu a ukládá ho do databáze. Stejně tak přijímá požadavky na retweet.

Poslední vlastností tohoto servletu je příjem nahrávaného obrázku jeho zmenšení a uložení do uživatelského profilu.

# Instalace aplikace

Na server je nutné nahrát zdrojové soubory aplikace. Ve složce WEB-INF je soubor create.sql, který je třeba na databázovém stroji spustit. On se postará o vytvoření všech tabulek, které bude aplikace potřebovat. Na závěr je třeba v souboru web.xml nastavit parametry pro připojení do takto vytvořené databáze. Jedná se o parametry database.user, database.password, database.port, database.database a database.server.

# Závěr

Semestrálka pro mě byla zajímavým úvodem do dělání webů v Javě (doposud jsem dělal pouze PHP). Obtížnostně tak akorát, musel jsem se naučit nové věci, ale zároveň jsem nad hledáním řešení nestrávil hodiny času. V práci se mi podařilo splnit zadání i předem vytyčené cíle.