**ŽILINSKÁ UNIVERZITA V ŽILINE**

FAKULTA RIADNIA A INFORMATIKY

**SEMESTRÁLNA PRÁCA**

PRENOS DÁT

Adrián Bednár

Ľuboš Fukas

Štefan Sliacky

Žilina, 2016

***Špecifikácia zadania, konkretizácia problému***

Úlohou semestrálnej práce je navrhnúť webovú aplikáciu pre posielanie a spracovanie dát. Preferovaný programovací jazyk je JAVA a webová platforma DropWizard, databáza MySQL s použitím ORM Hibernate.

**Aplikácia bude pozostávať z daných komponentov:**

* **Web servis (REST),** ktorý bude prijímať a poskytovať dáta. Daný web servis bude pozostávať z minimálne troch koncových bodov. (užívatelia, zariadenia a dáta pre zariadenia). Nad jednotlivými koncovými bodmi bude možne vykonať CRUD operácie.
* **Prezentačná časť** – webové rozhranie, bude zobrazovať namerané údaje, zoznam zariadení atď. Pre lepšie hodnotenie bude potrebné implementovať prihlásenie sa do systému pomocou užívateľského mena a hesla.
* **Koncové zariadenie** – môže byť buď reálne zariadenie postavené na ľubovoľnom MCU, raspberriPi atď. alebo emulované zariadenie, ktoré bude spúšťané na PC. Toto koncové zariadenie bude predstavovať senzor(napr. teplota, tlak, svetelnosť, rýchlosť vetra atď. ), ktorý bude odosielať dáta cez REST rozhranie na server.

***Krátka analýza***

Dropwizard

je Java framework určený pre vysoko výkonné RESTful webové servisy. Balancuje na hranici medzi knižnicou a frameworkom. Jeho cieľom je poskytovať výkonné, spoľahlivé implementácie všetkého, čo potrebuje webová aplikácia. Obsahuje v sebe niekoľko ďalších známych knižníc ako napríklad:

**REST (Representational State Transfer)**

Je architektúra rozhrania, navrhnutá pre distribuované prostredie. Na rozdiel od XML-RPC či SOAP, je REST orientovaný dátovo, nie procedurálne. Webové služby definujú vzdialené procedúry a protokol pre ich volanie, REST určuje, ako sa pristupuje k dátam a je použiteľný pre jednotný a ľahký prístup ku zdrojom (resources). Zdrojom môžu byť dáta, rovnako ako stavy aplikácie (ak sa dajú popísať konkrétnymi dátami). Všetky zdroje majú vlastný identifikátor [URI](http://en.wikipedia.org/wiki/Uniform_Resource_Identifier) a REST služba definuje štyri základné metódy pre prístup k nim.

## Metódy pre prístup ku zdrojom

REST implementuje štyri základní metódy, ktoré sú známe pod označením [CRUD](http://en.wikipedia.org/wiki/Create,_read,_update_and_delete), vytvorenie dát (Create), získanie požadovaných dát (Retrieve), zmenu (Update) a zmazanie (Delete). Tieto metódy sú implementované pomocou odpovedajúcich metód HTTP protokolu.

### **GET (Retrieve)**

Základnou metódou pre prístup ku zdrojom je **získanie zdroja** – metóda GET. Každý zdroj podľa rozhraní REST vlastní identifikátor (URI). Pomocpu HTTP GET požiadavky získame dáta konkrétneho zdroja.

### **POST (Create)**

Získavanie dát je jednoduché a priamočiare. Pro vytvorenie dát slúži totiž metóda POST. Pri metóde POST nie je vo chvíli volania známy presný identifikátor zdroje (pretože zdroj ešte neexistuje). Preto sa pre POST používa identifikátor („endpoint“).

### **DELETE**

Zdroj je možné zmazať pomocou volania URI HTTP metódy DELETE. Volanie je obdobné ako volanie metódy GET.

### **PUT (Update**)

Operácia zmeny je podobná ako operácia vytvorenia POST, s tím rozdielom, že voláme konkrétnu URI konkrétneho zdroja, ktorý chceme zmeniť. V tele predávame novú hodnotu (ako pri metóde POST). Na rozdiel od POST je pri úpravách zdroja jeho URI už známe, takže ju môžeme zadať

Návrh aplikácie

Aplikacia bola navrhovana na základe apklikacie navrhnutej počas cvičení. Celá aplikácia bola napísaná v jazyku JAVA v prostredí NetBeanse.

V práci sme doplnili nové triedy pre autentifikáciu uživateľa a pre 2 senzory. Každý senzor má vlastńu Resource triedu, kde sme zadefinovali cesty ku daným CRUD funkciam ako GET, POST, DELETE a PUT. Pre každé zariadenie sme vytvorili databazu na servery <http://www.phpmyadmin.co>. V databáze sa nachádza ID a hodnota záznamu.

Ďalšou časťou bolo vytvoriť senzory, ktory nám budú odosielať dáta do databázy. My sme sa rozhodli tieto senzory emulovať v jazyku JAVA. Vytvorili sme 2 senzory pre vlhkosť vzduchu a teplotu. Dáta zo senzorov sú náhodne generované a poseilané v pravedelných intervaloch cez náš server do databázy. Dáta sú odosielané pomocou jednoduchých http requestov.

Súčasťou práce je aj prezentačná časť. Tá je tvorená pomocou html stránok a štýl jednotlivých objektov na stránke je tvorený pomocou css. Na stránkach sa užívateľ prihlási a pomocou prihlasovacieho mena a hesla a následne si môže vybrať požadovaný senzor. Pri každom senzore je možné pridávať nové hodnoty, mazať jednotlivé údaje, prípadne zobraziť všetky hodnoty zo senzora.

ZAVER

Cieľom semestrálnej práce bolo navrhnúť webovú aplikáciu pre posielanie a spracovanie dát. Vytvorili sme aplikáciu, ktorá využíva službu REST, pomocou ktorej odosielame dáta zo senzorov do databázy.