

Technologie Zarządzania Treścią Sprawozdanie Node.js, Redis

Dominik Gronkiewicz- 118104

21 listopada 2016

Spis treści

| | | |
|----------|--|----------|
| 1 | Wstęp | 2 |
| 1.1 | Technologie i narzędzia | 2 |
| 2 | Przebieg projektu | 2 |
| 2.1 | Użycie npm do konfiguracji projektu | 2 |
| 2.2 | Przygotowanie bazy danych | 3 |
| 2.3 | Utworzenie aplikacji Nojde.js oraz widoków | 3 |
| 2.4 | Widoki aplikacji | 4 |
| 3 | Podsumowanie i wnioski | 5 |

1 Wstęp

Sprawozdanie opisuje przebieg oraz działanie projektu wykorzystującego nie-relacyjną bazę danych Redis oraz Node.js. Tematem projektu jest lista zakupów.

1.1 Technologie i narzędzia

- Redis
Redis to skrót od REmote DIctionary Service. Jest to baza przechowująca pary klucz-wartość.
- Node.js
Node.js jest platformą wykorzystującą język JavaScript. Używa standardu języka Google Chrome V8. Posiada ogromną liczbę modułów.
- Express
Minimalistyczny, szybki w działaniu i prosty w użyciu framework webowy. Wykorzystuje protokół HTTP do komunikacji między serwerem aplikacji, a klientem.
- ioredis
Klient bazy Redis dla frameworku Node.js. Wspiera wszystkie niezbędne operacje możliwe do wykonania na bazie danych Redis oraz skrypty Lua. Wspiera metody callback i promise oraz typy standardu ES6.
- mustache
Mustache to silnik szablonów wykorzystywany głównie w HTML, plikach konfiguracyjnych oraz kodzie źródłowym. Nie zawiera operatorów warunkowych oraz pętli, wszystko tworzone jest z użyciem tagów
- body-parser
Biblioteka parsowania danych. Służy do pobierania danych z arkuszy HTML.

2 Przebieg projektu

2.1 Użycie npm do konfiguracji projektu

Projekt konfigurowany jest następującymi komendami:

```
npm init
npm install ioredis
npm install express --save
npm install body-parser --save
npm install mustache --save
npm install consolidate
```

2.2 Przygotowanie bazy danych

Do działania aplikacji wymagane jest utworzenie odpowiednich struktur oraz wstępne wypełnienie bazy danych przykładowymi wartościami.

Uruchomienie bazy danych oraz konsolowego klienta w systemie operacyjnym opartym na Arch Linux odbywa się poleceniami:

```
systemctl start redis
redis-cli
```

Utworzenie struktury Hashes dla przykładowego użytkownika:

```
hmset user:test pass test1 listId list1
```

Utworzenie przykładowej listy:

```
rpush list1 milk eggs soap
```

Innymi wykorzystanymi operacjami Redis są kolejno: pobieranie nazwy listy ze struktury użytkownika; pobieranie wszystkich elementów listy; kasowanie elementu o określonej nazwie:

```
hget user:test listId
lrange list1 0 -1
lrem list1 -1 "soap"
```

2.3 Utworzenie aplikacji Nojde.js oraz widoków

Kolejnym etapem jest utworzenie kodu aplikacji w pliku server.js. Przykładowa metoda get oraz post. Metoda get wywołuje plik edit.html oraz przekazuje do niego dane z wartości editingElem pod kluczem item, z użyciem mustache. Metoda post pobiera dane z ciała pliku HTML z elementu z atrybutem name="item".

```
app.get('/edit', function(req, res) {
  if(!editingElem) return;
  res.render(__dirname + "/"
    + 'edit.html',{item: editingElem});
})

app.post('/del', function(req, res) {
  var item = req.body.item;
  redis.lrem(User.list, -1, item);
  res.redirect('/list');
})
```

Przykładowy widok HTML, którym pobrane są dane z klucza item, z użyciem mustache:

```
<!DOCTYPE html>
<html>
<body>
    <form action = "/edit" method = "POST">
        Editing value: <input type = "text"
            name = "item" class="input"
            value="{{item}}"> <br>
        <input type = "submit" value =
            "Change">
    </form>
</body>
</html>
```

2.4 Widoki aplikacji



Login: test Password: ***** Log in

[test user= login:test, password:test1]

Rysunek 1: Widok logowania



test's list

| | | |
|--------|------|--------|
| eggs | Edit | Delete |
| sugar | Edit | Delete |
| soap | Edit | Delete |
| banana | Edit | Delete |
| bread | Edit | Delete |

New item: Add

Rysunek 2: Widok listy zakupów z operacjami

A screenshot of a web form for editing a value. It consists of a text input field containing the word 'eggs' and a small button labeled 'Change' positioned directly below the input field.

Rysunek 3: Widok edycji elementu

3 Podsumowanie i wnioski

Node.js jest wygodną w użyciu platformą do szybkiego i prostego tworzenia serwera aplikacji. W użyciu z bazą danych Redis, pozwala na utworzenie w pełni działającej aplikacji internetowej w krótkim czasie. Nierelacyjna baza danych podwała pominąć krok projektowania złożonych relacji. Ogromna ilość modułów do Node, umożliwia zaimplementowanie różnorodnych funkcjonalności w wielu typach aplikacji.