

UNIWERSYTET GDAŃSKI
Wydział Matematyki, Fizyki i Informatyki

Mateusz Kwiatkowski

nr albumu: 194 925

Walidacja w MeteorJS na przykładzie elektronicznego indeksu

Praca magisterska na kierunku:

INFORMATYKA

Promotor:

dr Włodzimierz Bzyl

Gdańsk 2014

Streszczenie

Spis treści

| | |
|---|----|
| Wprowadzenie | 4 |
| 1. Walidacja | 5 |
| 1.1. Walidacja w MeteorJS | 5 |
| 1.2. MeteorJS i Mesosphere | 5 |
| 2. Pakiet do walidacji | 6 |
| 2.1. Meteorite i Atmospherejs | 6 |
| 2.2. Co zawierają pakiety | 6 |
| 2.3. Stratosphere | 6 |
| 2.4. Testowanie pakietów | 6 |
| 3. Aplikacja w MeteorJS | 7 |
| 4. Testowanie aplikacji | 8 |
| 5. Porównanie z obecnie używanym elektronicznym indeksem | 9 |
| Zakończenie | 10 |
| Bibliografia | 11 |
| Oświadczenie | 12 |

Wprowadzenie

Meteor jest dość nowym frameworkiem który bazuje na pakietach, więc jego podstawowa funkcjonalność nie jest kompletna. Można bardzo mocno zwiększyć jego możliwości dodając pakiety, które otrzymujemy od twórców Meteora lub od społeczności. Jedną z podstawowych funkcji jakiej meteor domyślnie nie posiada jest walidacja, ale powstał już pakiet, który doskonale radzi sobie z podstawową walidacją.

Jeszcze do niedawna na wszystkich uczelniach stosowano klasyczne indeksy papierowe, jednak w wyniku rozwoju technologii internetowych coraz częściej rezygnuje się z klasycznych rozwiązań zastępując je ich elektronicznymi odpowiednikami. Korzystając z elektronicznego indeksu jedną z ważniejszych funkcji jest walidacja. Aplikacja nie może dopuścić do sytuacji gdzie nauczyciel wystawi studentowi ocenę spoza skali czy też wprowadzić niepełne dane, ale również poprawnie interpretować czy dany użytkownik może wykonać w danej konkretnej akcję. O ile do tej prostszej części walidacji istnieje pakiet to do tej bardziej zaawansowanej już takiego nie znajdziemy więc będzie trzeba go stworzyć.

Na każdej uczelni znajduje się wielu studentów oraz wykładowców przez co trzeba zadbać. W momencie gdy wiele osób jednocześnie chce przejrzeć indeks lub wystawić oceny nie można dopuścić do sytuacji gdy nasza aplikacja nie będzie w stanie obsłużyć wszystkich osób w jednym czasie. Z pomocą przychodzi nam javascriptowy framework **MeteorJS** który w połączeniu z bazą danych **MongoDB** zapewni nam wystarczającą skalowalność aplikacji oraz dostarczy nam sporą ilość gotowych pakietów, które uprościć stworzenie nowego produktu.

ROZDZIAŁ 1

Walidacja

1.1. Walidacja w MeteorJS

1.2. MeteorJS i Mesosphere

ROZDZIAŁ 2

Pakiet do walidacji

2.1. Meteorite i Atmospherejs

2.2. Co zawierają pakiety

2.3. Stratosphere

2.4. Testowanie pakietów

ROZDZIAŁ 3

Aplikacja w MeteorJS

ROZDZIAŁ 4

Testowanie aplikacji

ROZDZIAŁ 5

Porównanie z obecnie używanym elektronicznym indeksem

Zakończenie

[1] [2] [3] [4] [5] [6] [7] [8] [9]

Bibliografia

- [1] Percolate Studio. *Atmosphere Docs*, 2014.
- [2] MeteorJS. *Meteor Docs*, 2014.
- [3] Node JS. *Node Docs*, 2014.
- [4] MongoDB. *Mongo Docs*, 2014.
- [5] Kelly Copley. *Mesosphere*, 2014.
- [6] Tom Coleman and Sacha Greif. *Discover Meteor*. 2013.
- [7] Kristina Chodorow. *Scaling MongoDB*. 2011.
- [8] Rick Copeland. *Scaling with MongoDB*. 2012.
- [9] Arunoda Susiripala. *Test Driven Development with Meteor*. 2013.

Oświadczenie

Ja, niżej podpisany(a) oświadczam, iż przedłożona praca dyplomowa została wykonana przeze mnie samodzielnie, nie narusza praw autorskich, interesów prawnych i materialnych innych osób.

.....

data

.....

podpis