

UNIwersytet Gdański
Wydział Matematyki, Fizyki i Informatyki

Mateusz Kwiatkowski

nr albumu: 194 925

Walidacja w aplikacjach Meteor

Praca magisterska na kierunku:

INFORMATYKA

Promotor:

dr Włodzimierz Bzyl

Gdańsk 2014

Streszczenie

sa powinno zawierać omówienie głównych tez pracy magisterskiej, celów jakie autor sobie postawił

sb powinno zawierać informację czy udało się je zrealizować

sc należy także napisać jakimi metodami, technologiami się posłużono i jakie to przyniosło efekty

W pracy przedstawiony zostanie pakiet walidujący do frameworka *Meteor*. Do zobrazowania jego działania zostanie również stworzona aplikacja Elektroniczny Indeks. Pakiet do walidacji ma na celu zapobieganie wprowadzaniu przez użytkownika błędnych danych do systemu. Chcąc to osiągnąć pakiet ten będzie walidować operacje, które można wykonywać w aplikacji. Aplikacja ta ma być także prosta i intuicyjna w obsłudze. Do stworzenia aplikacji zostanie użyty framework *Meteor*, a dane wprowadzane do systemu będą przechowywane w bazie danych *MongoDB*. Wszystkie te cele zostały zrealizowane, a pakiet do walidacji został udostępniony do pobrania.

Spis treści

Wprowadzenie	5
1. Walidacja oprogramowania	7
1.1. Wstęp do walidacji	7
1.2. Typy walidacji	7
2. Aplikacja Elektroniczny indeks w Meteor	8
2.1. Cele i funkcjonalność aplikacji	8
2.2. Opis tworzenia aplikacji	8
2.3. Opis testowania aplikacji	8
2.4. Opis własnych rozwiązań	8
3. Pakiet walidujący operacje elektronicznego indeksu	9
3.1. Funkcjonalność pakietu	9
3.2. Opis tworzenia pakietu	9
3.3. Implementacja pakietu w aplikacji	9
3.4. Przetestowanie pakietu	9
Zakończenie	10
A. Mesosphere	11
B. Meteor	12
C. Laika	13
D. TinyTest	14
Bibliografia	15
Spis tablic	16

Spis rysunków	17
Oświadczenie	18

Wprowadzenie

- wa** jak nazwa wskazuje, ma wprowadzać w obszar problemowy pracy
- wb** powinno przedstawiać ogólne uwarunkowania problemu oraz opisać go w kontekście
- wc** powinno zawierać powód dlaczego poruszyło się taki temat
- wd** należy odnieść się do dorobku innych

Walidacja jest działaniem mającym na celu potwierdzenie w sposób udokumentowany i zgodny z założeniami, że procedury, procesy, urządzenia, materiały, czynności i systemy rzeczywiście prowadzą do zaplanowanych wyników. Znana jest także jako kontrola jakości oprogramowania. Wykorzystuje się ją w naukach technicznych oraz informatyce.

Aplikacje pozbawione walidacji pozwalają użytkownikowi na wprowadzenie irracjonalnych danych do systemu. Przykładem takiej aplikacji jest elektroniczny indeks. Operacje takie jak wystawianie studentowi ocen z ćwiczeń czy też wystawienie oceny końcowej z egzaminu gdy student nie posiada pozytywnej oceny z danych zajęć powinny być odpowiednio walidowane i nie dopuszczać do sytuacji gdy student otrzymuje ocenę z poza skali lub od osoby do tego nieupoważnionej. 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.

Zastosowanie walidacji w elektronicznym systemie wystawiania ocen udoskonali jego funkcjonalność, a także usprawni działanie takiej aplikacji. Korzystając z aplikacji w której zaimplementowana jest walidacja nie dojdzie do sytuacji gdzie użytkownik wprowadzi błędne dane do systemu oszczędzając tym samym jego czas, a także zwiększy jego efektywność. Miałem kontakt z wieloma systemami zarządzania osiągnięciami studentów, ale w każdym można było doprowadzać do

anomalii, a samo działanie takiej aplikacji również pozostawiało wiele do życzenia, dlatego postanowiłem zająć się tym tematem, aby usprawnić działanie takiego systemu oraz żeby praca na nim była przyjemna, prosta i intuicyjna.

Postaram się udowodnić jak bardzo przydatna jest walidacja pokazując jej działanie w aplikacji stworzonej w frameworku *Meteor*. Pokażę również na czym polega stworzenie pakietu i udostępnienie go w prosty sposób. Opierając się na doświadczeniach innych badaczy takich jak Kelly Copley [1] czy Tom Coleman i Sacha Greif [2] napiszę pakiet walidujący oraz aplikację Elektroniczny Indeks, która będzie korzystać ze stworzonego w ramach pracy pakietu. Napiszę także dlaczego uważam *Meteor* oraz *MongoDB* jako najlepszy wybór.

ROZDZIAŁ 1

Walidacja oprogramowania

1.1. Wstęp do walidacji

[3]

1.2. Typy walidacji

ROZDZIAŁ 2

Aplikacja Elektroniczny indeks w Meteor

2.1. Cele i funkcjonalność aplikacji

2.2. Opis tworzenia aplikacji

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

2.3. Opis testowania aplikacji

[10]

2.4. Opis własnych rozwiązań

ROZDZIAŁ 3

Pakiet walidujący operacje elektronicznego indeksu

3.1. Funkcjonalność pakietu

3.2. Opis tworzenia pakietu

[11] [5] [2]

3.3. Implementacja pakietu w aplikacji

3.4. Przetestowanie pakietu

[12]

Zakończenie

DODATEK A

Mesosphere

DODATEK B

Meteor

DODATEK C

Laika

DODATEK D

TinyTest

Bibliografia

- [1] Kelly Copley. *Mesosphere*. *Github Repository*, 2014.
- [2] Tom Coleman and Sacha Greif. *Discover Meteor: Building Real-Time JavaScript Web Apps*. First edition edition, 2013.
- [3] Wikipedia. *Weryfikacja i walidacja oprogramowania*. *Wikipedia*, 2014.
- [4] Stephen Walther. *An Introduction to Meteor*. *stephenwalther*, 2013.
- [5] MeteorJS. *Meteor Documentation*, 2014.
- [6] Joyent. *Node.js Manual and Documentation*, 2014.
- [7] MongoDB. *The MongoDB Manual*, 2014.
- [8] Kristina Chodorow. *Scaling MongoDB*. First edition edition, 2011.
- [9] Rick Copeland. *Scaling with MongoDB*. page 32, 2012.
- [10] Arunoda Susiripala. *Test Driven Development with Meteor*. *Sitepoint*, 2013.
- [11] Stephen Walther. *Creating a basic meteorite smart package*. *Vintyre Inc*, 2012.
- [12] Eventedmind. *Testing Packages with Tinytest*, 2013.

Spis tablic

Spis rysunków

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