



## WYDZIAŁ ELEKTRYCZNY POLITECHNIKI WARSZAWSKIEJ

BAZY DANYCH W SYSTEMACH ROZPROSZONYCH

---

# Badowser

---

*Autorzy:*

Bartosz PIEŃKOWSKI

Barnaba TUREK

26 listopada 2011

## Streszczenie

Celem projektu jest zapoznanie się z rozproszonym otwartym systemem bazodanowym CouchDB. Projekt podzielony jest na 6 etapów, każdy z nich odpowiada konkretnemu zagadnieniu i związane z nim są konkretne artefakty.

## 1 Motyw projektu

Projekt będzie kopiował podstawową funkcjonalność serwisów mikroblogowych (takich jak twitter). Danymi przechowywanymi w bazie będą więc wiadomości, posiadane przez użytkowników i ewentualnie połączone tagami.

## 2 Wstęp

W ramach projektu **bazy danych w systemach rozproszonych** stworzyliśmy w dwuosobowym zespole prosty projekt, którego głównym celem było zapoznanie się z problematyką rozproszonych baz danych.

## 3 Co zostało wykonane

Głównym sukcesem w projekcie było zapoznanie się z nowoczesną i niezbyt popularną technologią CouchDB. Było to o tyle trudne, że technologia bazodanowa CouchDB jest wciąż młoda, w związku z czym jest słabo udokumentowana. Do tego spotkaliśmy się z wieloma problemami wynikającymi z tego, że technologia jest wciąż młoda.

W ramach zapoznawania się z technologią CouchDB stworzyliśmy aplikację, działającą w sposób rozproszony. Była to aplikacja nieco przypominająca twittera pozwalająca użytkownikom dodawać wpisy. Dodane przez użytkowników wpisy w czasie rzeczywistym replikowały się pomiędzy instancjami bazy danych.

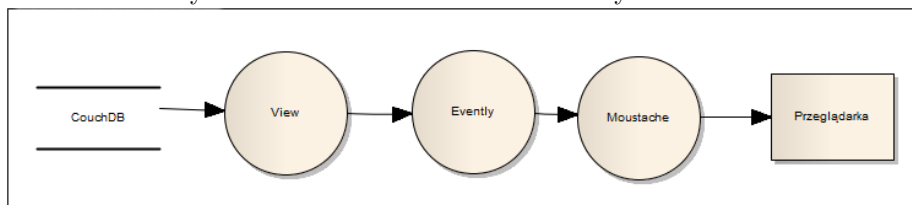
Aplikacja obsługuje uwierzytelnianie użytkowników za pomocą wbudowanego systemu użytkowników CouchDB.

Pozostali klienci aplikacji otrzymują informacje o nowych wpisach, kiedy tylko się pojawiają (bez potrzeby odświeżania strony).

Ponadto aplikacja jest odporna na utratę dowolnej ilości instancji bazy, oraz na utratę połączenia pomiędzy bazami (po odzyskaniu połączenia aplikacja automatycznie przystępuje do replikacji danych).

Dużą zaletą tej technologii jest fakt, że aplikacje są umieszczone i serwowane z bazy danych - nie jest potrzebny dodatkowy serwer aplikacji (np. apache, czy JBoss).

Rysunek 1: Proces renderowania strony dla klienta



## 4 Architektura aplikacji

Aplikacja została wykonana jako stos współdziałających ze sobą elementów:

- CouchDB
- CouchApp
- Evently
- Moustache

Jej podstawą jest serwer CouchDB, który zapewnia przechowywanie informacji (zarówno o aplikacji, jak i informacji dodanych przez użytkowników).

Następnym elementem, jest framework CouchApp, pozwalający łatwo stworzyć aplikacje działające w bazie danych CouchDB. Aplikacje zbudowane przy użyciu CouchApp opierają się na technologiach HTML-5 i javascript (który też jest głównym językiem służącym do przetwarzania informacji w bazie CouchDB).

Budowa aplikacji przy wykorzystaniu tej technologii polega na utworzeniu tak zwanego *design document*, czyli specjalnego skryptu, opisującego aplikację. W tym pliku umieszczone są takie informacje, jak opis routingu aplikacji (które adresy odpowiadają jakim funkcjom w javascriptcie), opisane są także statyczne pliki (nie zmieniające się strony, obrazki, skrypty użytkownika i style). Ponadto w tym pliku umieszczone są informacje o odczycie i zapisie danych (czyli sposoby przygotowania danych - tzw. widoki oraz m.in. walidacje danych wprowadzonych przez użytkowników).

Użyliśmy także biblioteki evently, która pozwoliła nam na łatwą implementację tzw. push notifications, czyli wyświetlanie użytkownikom nowych postów, kiedy tylko się pojawiają.

Skorzystaliśmy także z biblioteki Moustache, która pozwala na łatwe wypełnianie szablonów stron danymi.

Współpracę wykorzystanych komponentów opisuje Rysunek 1.

Stworzenie aplikacji w ten sposób pozwoliło, na przechowywanie jej w bazie. Jedną z zalet takiego rozwiązania jest fakt, że sama aplikacja także jest replikowana. Kolejną zaletą jest fakt, że każdy klient może posiadać własną lokalną kopię aplikacji i z niej korzystać (lokalna kopia aplikacji będzie miała krótsze czasy odpowiedzi na żądania klienta. Poza tym klient może edytować dane w bazie, kiedy nie jest połączony z internetem, a kiedy się połączy mechanizm replikacji wprowadzi dokonane przez niego zmiany do głównych baz).

## 5 Kod źródłowy

Kod źródłowy dostępny jest na stronie <https://github.com/barnaba/badowser>. Zastosowana struktura podziału kodu źródłowego jest wymuszona przez projekt CouchApp:

**evently** Przechowuje kod obsługi zdarzeń i renderowania odpowiednich szablonów.

**vendor** Przechowuje niezbędne biblioteki javascriptowe (w tym samą bibliotekę CouchApp).

**views** Przechowuje kod odpowiedzialny za zapytania.

**attachments** Przechowuje statyczne elementy, takie jak style czy obrazy.

**docs** Dokumentacja projektu.

W głównym katalogu projektu znajdują się ogólne pliki konfiguracyjne.

Ponadto część konfiguracji znajduje się w bazie danych (np. ustawienia aplikacji).