Sprawozdanie i dokumentacja projektu wykonanego podczas hackatonu  
Rzethon 2018

Nazwa projektu

**Nomadic Reader**

Skład zespołu

Dominik Ruman  
Dawid Bartosz  
Jakub Mularski

Dokumentację wykonał

**Dominik Ruman**  
student Politechniki Rzeszowskiej  
nr indeksu 156324

Nomadic Reader

Travel Better

Spis treści

[Opis projektu 3](#_Toc536603120)

[Motywacja 3](#_Toc536603121)

[Co dalej? 3](#_Toc536603122)

[Dane techniczne 4](#_Toc536603123)

[Podział projektu 4](#_Toc536603124)

[Technologia 5](#_Toc536603125)

[Funkcjonowanie aplikacji 5](#_Toc536603126)

[Podsumowanie 8](#_Toc536603127)

[O zespole 8](#_Toc536603128)

# Opis projektu

Tematem przewodnim konkursu Rzethon 2018 było „Smart City”. Zespół „Racjonalny Traktor” wziął udział celując w kategorię aplikacji mobilnych, tworząc aplikację o nazwie „Nomadic Reader”.

## Motywacja

Problemem, z którym się zetknęliśmy w naszym mieście było niemalże całkowite zaniedbanie ze strony miasta w promocję lokalnych twórców   
i artystów. Problem postanowiono rozwiązać w sposób, którego dotyczył konkurs, czyli dosłownie „smart” w „city”. Rozwiązanie zakładało cyfrowy dostęp do fragmentów książek itp. Lokalnych twórców podczas podróżowania autobusem komunikacji miejskiej.

Twórca rejestrowałby się w systemie, miałby możliwość udostępnienia fragmentów lub nawet całości swoich dzieł, który były by do odczytania podczas przebywania we wnętrzu autobusu MPK. Po wysiąściu z autobusu aplikacja zezwalałaby na dokończenie czytania książki, ale zostawiłaby pewne „namiary” czyli tytuł, dane autora itp. itd., a nawet link do zakupu fizycznej książki jeśli autor miałby taką możliwość.

## Co dalej?

Projekt nie miał być skupiony tylko wokół lokalnych twórców. Możliwa byłaby integracja lokalnej lub ogólnopolskiej gazety codziennej z różnymi wiadomościami. Wiele serwisów wiadomości umożliwia usługę premium dając dostęp do większej ilości artykułów lub ich pełnej treści – to mogłoby być także zintegrowane w aplikacji.

# Dane techniczne

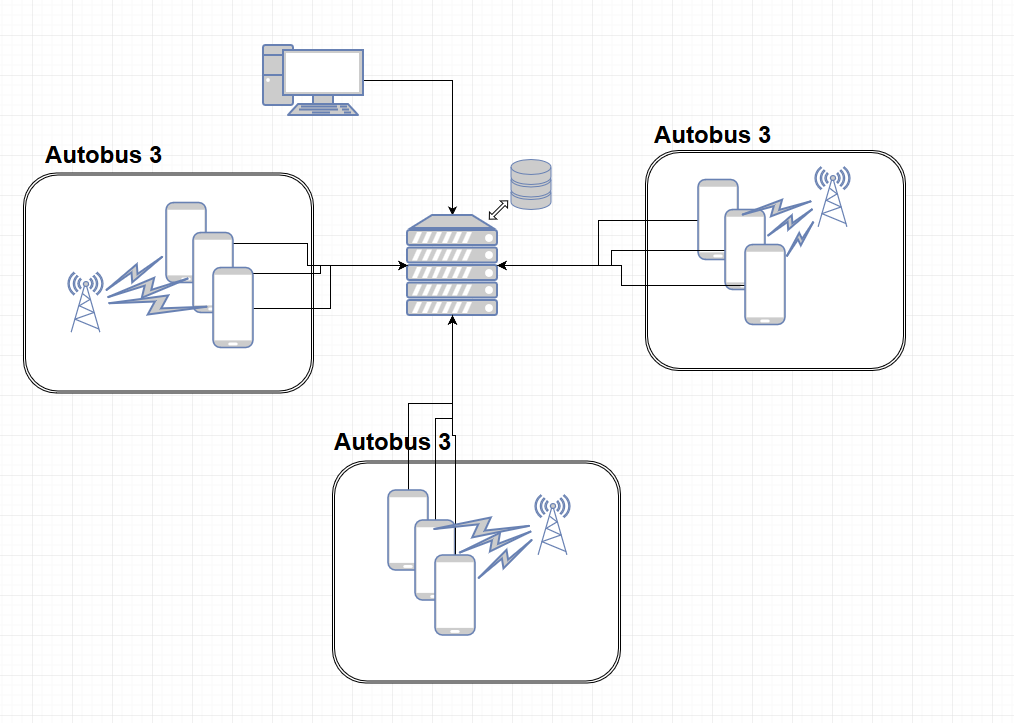
## Podział projektu

Program składał się z trzech części:

- aplikacji mobilnej (folder Books-mobile), na której można czytać udostępnione utwory

- aplikacji internetowej (folder Books-frontend), na której twórcy mogą zarządzać swoimi utworami

- serwera dla całego systemu (folder Books-backend), który wszystko razem trzymał oraz komunikował się z bazą danych



## Technologia

Cały projekt opierał się o technologię beaconów opartych o bluetooth low energy, które pozwalały na rozpoznanie czy dana osoba znajduje się w autobusie, czy nie. Jeśli osoba się znajduje to nadawany jest jej identyfikator świadczący o tym, którym autobusem jedzie. Dzięki temu identyfikatorowi serwer mógł obsłużyć zapytania o utwory.

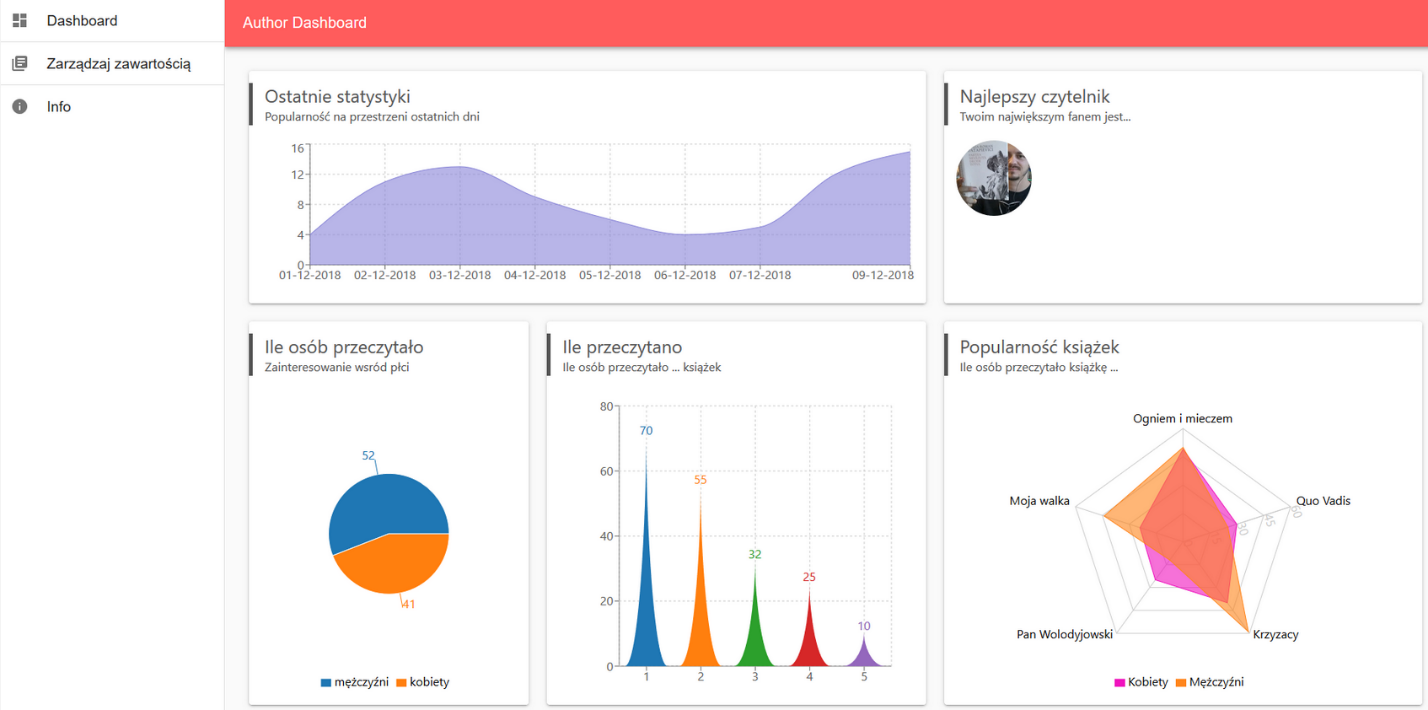
Cześć panelu zarządzania była stworzona w frameworku javascriptowym React.js.

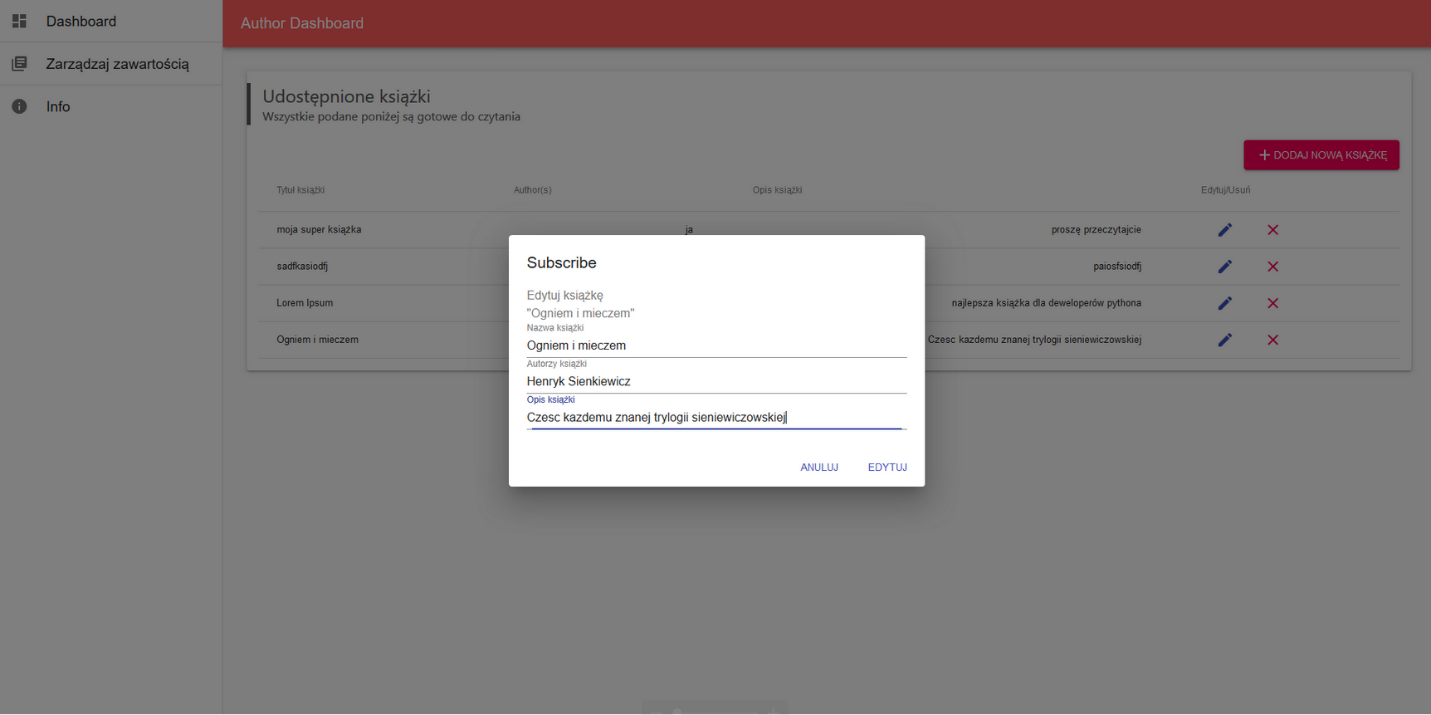
Cześć aplikacji mobilnej była stworzona w React Native.

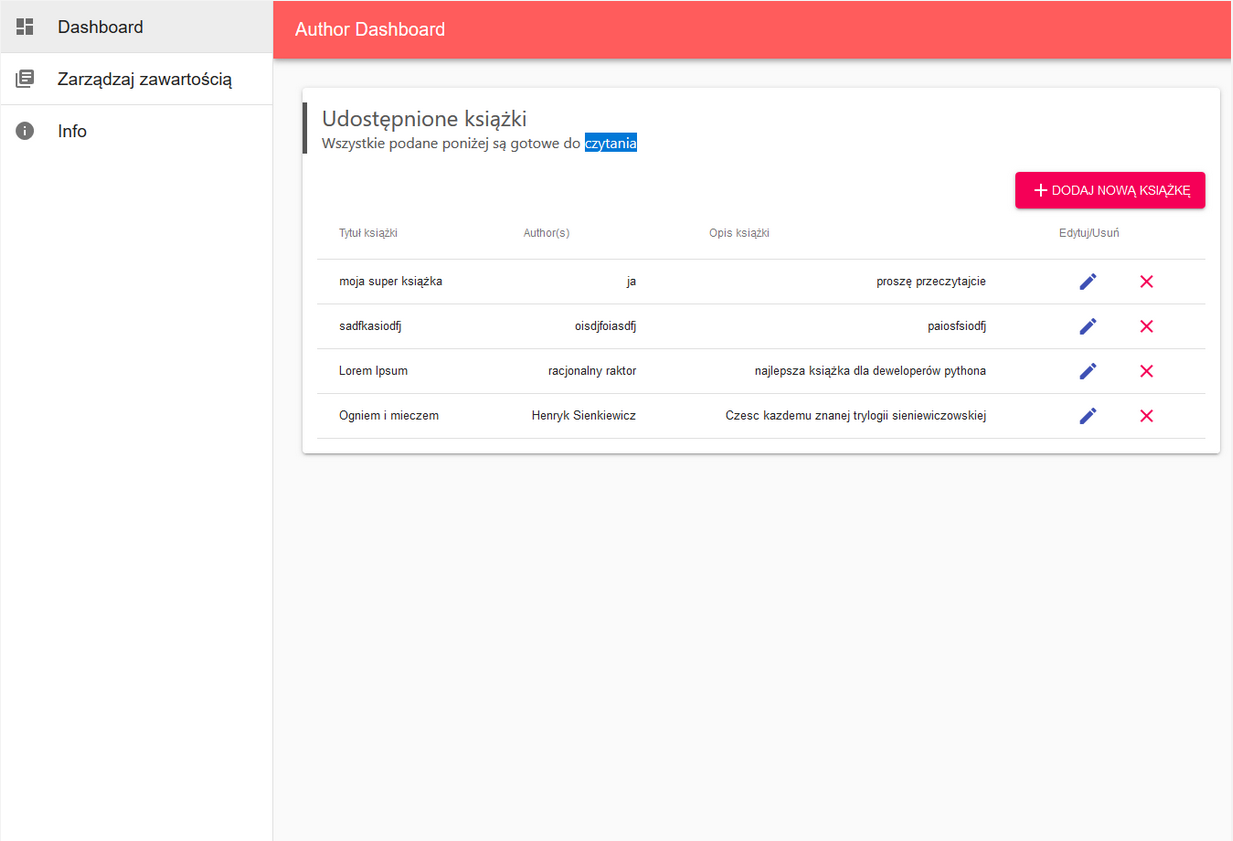
Cześć serwera stworzona była w Node.js. Obsługiwana baza danych to MongoDB.

## Funkcjonowanie aplikacji

Panel twórcy składał się z panelu informacyjnego i statystykach o najnowszych aktywnościach użytkowników i popularności danego autora oraz panelu zarządzania utworami.



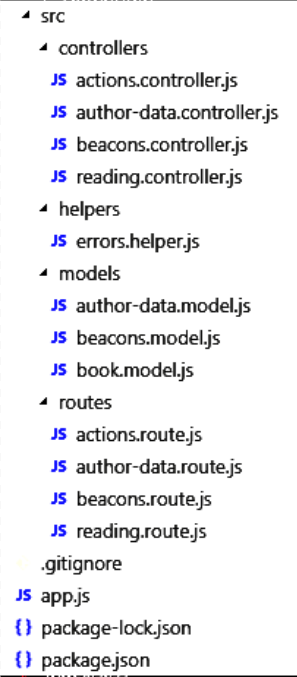




Aplikacja pisana w React.js składała się z następujących elementów:

Nie jest to wielki projekt, ale pochłonął dużo linijek kodu w celu niezawodnego działania i poprawnego funckcjonowania panelu. W App.js zawarty jest routing, który przekierowywał nas na odpowiednie strony (Pages) jeśli byliśmy zalogowani. Strony wykorzystywały re-używalne komponenty.

Aplikacja w po stronie mobilnej była bardzo okrojona wizualnie, największą trudnością było zgranie jej z beaconami. Po połączeniu się i upewnieniu, czy na pewno jesteśmy w autobusie (opóźnione wywołanie), aplikacja odpytywała serwer przesyłając numer autobusu (jako numer beacona) do serwera, który zwracał odpowiednią treść (utwory) dla czytelnika.



Aplikacja po stronie serwera składała się z routów, które zwracały odpowiednie kontrolery akcji. Kontrolery zajmowały się całą logiką przetwarzania zapytań oraz zarządzania bazą danych. Struktura aplikacji po stronie serwera wyglądała następująco:

# Podsumowanie

Całość projektu została napisana w pewnych dialektach języka JavaScript, który mimo swoich wad, jest bardzo dobrym wyborem na konkursy typu Hackaton ze względu na łatwiznę jego implementacji, stosowania i brak typowania.

# O zespole

Zespół Racjonalny Traktor został stworzony przez czwórkę ambitnych gości, których pasją jest informatyka i programowanie. Każdy z nich miał pewne doswiadczenie, ale dopiero na konkursie Rzethon stworzyli coś pierwszego razem. Ich projekt zajął trzecie miejsce oraz dostał bilety na największy europejski hackaton – HackYeah 2017 na którym zwyciężyli w technologii IoT opierającą się na beaconach. Wtedy dostali w nagrodzie beacony od firmy Estimote, które wozili ze sobą na inne hackatony m.in. WawCode wiosna 2018 oraz następny HackYeah 2018, w którym zdobyli pierwsze miejsce w kategorii Visa – „Payments in the world of Internet of Things”.

Na tegorocznym Rzethonie zespół rozbił się na dwa pod-zespoły (ze względu na limit ilości osób w zespole), z których jeden zajął miejsce na podium.