

POLITECHNIKA BIAŁOSTOCKA

WYDZIAŁ INFORMATYKI

KATEDRA OPROGRAMOWANIA

PRACA DYPLOMOWA INŻYNIERSKA

TEMAT: APLIKACJA MOBILNA DO
ZARZĄDZANIA TRENINGIEM NA
SIŁOWNI

WYKONAWCA: NORBERT KAMIEŃSKI

.....
podpis

PROMOTOR: DR INŻ. MARCIN CZAJKOWSKI

BIAŁYSTOK 2017 r.

Politechnika Białostocka Wydział Informatyki Katedra OPROGRAMOWANIA	Studia stacjonarne studia I stopnia	Numer albumu studenta:93401
		Rok akademicki 2016/2017
		Kierunek studiów: informatyka Specjalność:
<p style="text-align: center;">Norbert Kamiński</p> <p>TEMAT PRACY DYPLOMOWEJ: APLIKACJA MOBILNA DO ZARZĄDZANIA TRENINGIEM NA SIŁOWNI</p> <p>Zakres pracy:</p> <ol style="list-style-type: none"> 1. Opis technologii wykorzystywanych przy tworzeniu systemu. 2. Wykonanie projektu aplikacji do zarządzania treningiem na siłowni. 3. Implementacja i testy zrealizowanej aplikacji. <p>Słowa kluczowe: Aplikacja Mobilna, React Native, Firebase, JavaScript, Android, Organizer</p>		
<div style="display: flex; justify-content: space-around;"> <div style="text-align: center;"> <p>.....</p> <p>Imię i nazwisko promotora</p> <p>podpis</p> </div> <div style="text-align: center;"> <p>.....</p> <p>Imię i nazwisko kierownika</p> <p>katedry - podpis</p> </div> </div>		
<div style="display: flex; justify-content: space-between;"> <div style="width: 30%;"> <p>.....</p> <p>Data wydania tematu pracy dyplomowej</p> <p>- podpis promotora</p> </div> <div style="width: 30%;"> <p>.....</p> <p>Regulaminowy termin złożenia</p> <p>pracy dyplomowej</p> </div> <div style="width: 30%;"> <p>.....</p> <p>Data złożenia pracy dyplomowej</p> <p>- potwierdzenie dziekanatu</p> </div> </div>		
<div style="display: flex; justify-content: space-around;"> <div style="text-align: center;"> <p>.....</p> <p>Ocena promotora</p> </div> <div style="text-align: center;"> <p>.....</p> <p>Podpis promotora</p> </div> </div>		
<div style="display: flex; justify-content: space-between;"> <div style="width: 30%;"> <p>.....</p> <p>Imię i nazwisko recenzenta</p> </div> <div style="width: 30%;"> <p>.....</p> <p>Ocena recenzenta</p> </div> <div style="width: 30%;"> <p>.....</p> <p>Podpis recenzenta</p> </div> </div>		

Mobile application to manage training in the gym

Major goal of this thesis was to create a mobile application that helps athletes manage their trainings in the gym. For this purpose has been designed and developed mobile application. It is easier way to organize and track the progress of sportsmen workouts. Humans memory is an unreliable. Athletes have a problem with that as well. Trainees don't remember what weight recently trained or how series, reps they lifted. Somebody note that in a usual notebook. But It is not very handy, can be forgot it and have limited space. That application solves these and many others problems. Most people have a mobile at this moment, which takes to the gym, too. Software allows to note progress during training and planning or observing some statistic. Application is programmed in the JavaScript's framework React Native supported by Redux technology. Framework uses functional approach to programming and frontends technologies like HTML, JSX and CSS. Although used web technologies, application is 100%

The paper contains five chapters. The first is an introduction. It presents the goal, plan and scope of work. Second chapter- shows an overview of the application, existing solutions and functional and nonfunctional requirements. The most important fourth chapter describes implementations project. It contains application architecture, principle of operation, tests and product presentations. The last is summary of the work. Finally created the application, already used by athletes.

Spis treści

1	Wstęp.	1
1.1	Cel pracy	1
1.2	Zakres pracy	1
1.3	Plan pracy	1
2	Analiza wymagań.	3
2.1	Wizja oraz ogólny zarys aplikacji.	3
2.2	Istniejące rozwiązania.	4
2.3	Wymagania funkcjonalne	4
2.4	Wymagania niefunkcjonalne	5
3	Wykorzystane technologie.	7
3.1	Przegląd technologii	7
3.1.1	Swift (Objectiv-C)	7
3.1.2	Android (Java)	7
3.1.3	React Native (Javascript)	7
3.2	Wybrane technologie	7
3.2.1	React Native + Redux	7
3.2.2	Javascript (JSX, ES6)	7
3.2.3	HTML & CSS	7
3.2.4	Firebase.	7
3.3	Wybrane technologie.	7
3.3.1	Microsoft Visual Studio.	7

3.3.2	React Native Debugger.	7
3.3.3	GIT- system kontroli wersji.	7
4	Realizacja aplikacji.	9
4.1	Architektura aplikacji mobilnej	9
4.2	Schemat działania aplikacji	9
4.3	Baza danych	9
4.4	Prezentacja aplikacji	9
4.5	Testy aplikacji	9
4.6	Wdrożenie	9
4.7	Obsługiwane systemy monitorowania	9
5	Podsumowanie	11
6	Bibliografia	13
	Spis tabel	15
	Spis rysunków	17

1. Wstęp

1.1 Cel pracy

1.2 Zakres pracy

1.3 Plan pracy

2. Analiza wymagań

2.1 Wizja oraz ogólny zarys aplikacji.

Jedną z istniejących możliwości i najczęściej wykorzystywaną przez rozwiązania komercyjne jest wysyłanie do serwera pakietów ICMP. Nazywane jest to popularnie PING od nazwy aplikacji systemowej¹ pozwalającej na wysłanie takiego zapytania i wyświetlenie czasu odpowiedzi.

Rozwiązanie to jest najczęściej wybierane przez komercyjne strony i aplikacje ze względu na prostotę implementacji i popularność. Aplikacja pozwalająca na wysłanie dowolnej liczby takich zapytań dostępna jest na każdym popularnym systemie operacyjnym, wszystkie popularne kontrolery ethernet mają wbudowany system odpowiadania na pakiety ICMP. Budowa takiego pakietu została przedstawiona na rysunku 1.

ICMP packet				
	Bit 0 - 7	Bit 8 - 15	Bit 16 - 23	Bit 24 - 31
IP Header (20 bytes)	Version/IHL	Type of service	Length	
	Identification		flags and offset	
	Time To Live (TTL)	Protocol	Checksum	
	Source IP address			
	Destination IP address			
ICMP Payload (8+ bytes)	Type of message	Code	Checksum	
	Quench			
	Data (optional)			

Rysunek 2.1: Budowa pakietu ICMP.

Sposób ten ma jednak wiele wad, które w niektórych sytuacjach znacznie utrudniają lub wręcz uniemożliwiają jego wykorzystanie. Przedstawione zostały one poniżej:

- Urządzenie monitorowane musi automatycznie odpowiadać na pakiety ICMP.

¹Andotacja - z ang. system.

- Pakiety ICMP często mają bardzo niski priorytet i nie są powtórnie wysyłane po zagubieniu, powoduje to konieczność wysyłania pakietów do skutku, zaś jeden brak odpowiedzi na pakiet nie gwarantuje tego, że serwer jest niedostępny.
- Wiele serwerów, firewalli i urządzeń sieciowych takich jak routery i przełączniki blokuje całkowicie protokół ICMP w celu ochrony sieci.
- Wykorzystując pakiet ICMP (ping) można sprawdzić tylko dostępność urządzenia o podanym adresie IP w sieci – nie pozwala to na sprawdzenie dostępności konkretnego serwera zainstalowanego na danej maszynie.

2.2 Istniejące rozwiązania.

1. Urządzenie monitorowane musi automatycznie odpowiadać na pakiety ICMP.
2. Pakiety ICMP często mają bardzo niski priorytet i nie są powtórnie wysyłane po zagubieniu, powoduje to konieczność wysyłania pakietów do skutku, zaś jeden brak odpowiedzi na pakiet nie gwarantuje tego, że serwer jest niedostępny.
3. Wiele serwerów, firewalli i urządzeń sieciowych takich jak routery i przełączniki blokuje całkowicie protokół ICMP w celu ochrony sieci.
4. Wykorzystując pakiet ICMP (ping) można sprawdzić tylko dostępność urządzenia o podanym adresie IP w sieci – nie pozwala to na sprawdzenie dostępności konkretnego serwera zainstalowanego na danej maszynie.

2.3 Wymagania funkcjonalne

Dawniej decyzja o wyborze pomiędzy istniejącym systemem operacyjnym a tworzeniem swojego oprogramowania od podstaw była skomplikowana i uwarunkowana bardzo często użytym sprzętem i potrzebami. Aktualnie, dzięki upowszechnieniu systemu Linux w systemach wbudowanych, ta decyzja bardzo często sprowadza się do wyboru jego dystrybucji i konfiguracji. Linux radzi sobie nawet w systemach czasu rzeczywistego od których wymagana jest wysoka wydajność [1, 2].

2.4 Wymagania niefunkcjonalne

3. Wykorzystane technologie

3.1 Przegląd technologii

3.1.1 Swift (Objectiv-C)

3.1.2 Android (Java)

3.1.3 React Native (Javascript)

3.2 Wybrane technologie

3.2.1 React Native + Redux

3.2.2 Javascript (JSX, ES6)

3.2.3 HTML & CSS

3.2.4 Firebase.

3.3 Wybrane technologie.

3.3.1 Microsoft Visual Studio.

3.3.2 React Native Debugger.

3.3.3 GIT- system kontrolni wersji.

4. Realizacja aplikacji

4.1 Architektura aplikacji mobilnej

4.2 Schemat działania aplikacji

4.3 Baza danych

4.4 Prezentacja aplikacji

4.5 Testy aplikacji

4.6 Wdrożenie

4.7 Obsługiwane systemy monitorowania

5. Podsumowanie

6. Bibliografia

- [1] Gene Sally. *Pro Linux Embedded Systems*. APRESS, 2010.
- [2] Petru Eles Paul Pop and Zebo Peng. *Analysis and Synthesis of Distributed Real-Time Embedded Systems*. Springer Science+Buissness LLC, 2004.
- [3] Bruce Eckel. *Thinking in C++*. Wydawnictwo HELION, second edition, 2002.
- [4] Bruce Eckel. *Thinking in C++, Tom 2*. Edition 2000, second edition, 2006.
- [5] John W. Turner Keir Daivs and Nathan Yocom. *The definitive guide to Linux Network Programming*. Apress Media LLC., second edition, 2004.
- [6] Eben Upton and Gareth Halfacree. *Raspberry Pi. Przewodnik użytkownika*. Helion, 2013.
- [7] Paweł Szczur. *Własny monitoring dla stron WWW*, volume 35. Dom Wydawniczy Anna Adamczyk, 2015.

Spis tabel

Spis rysunków

2.1 Budowa pakietu ICMP.	3
----------------------------------	---

Spis listingów

OŚWIADCZENIE

Ja, Norbert Kamieński, wyrażam zgodę na udostępnienie mojej pracy dyplomowej, pt: „APLIKACJA MOBILNA DO ZARZĄDZANIA TRENINGIEM NA SIŁOWNI” w Bibliotece Politechniki Białostockiej.

.....
3 stycznia 2017

czytelny podpis