



---

**POZNAN UNIVERSITY OF TECHNOLOGY**

---

**Patryk Dąbrowki 100584**  
**Aleksander Kędzierski 98875**  
**Paweł Lampe 99277**  
**Mateusz Sikora 99615**

# Platforma zarządzania zdarzeniami na urządzeniach mobilnych $if\{y\}$

Bachelor's Thesis

Supervisor: dr inż. Jerzy Błaszczyński

Poznań, 2014



# Spis treści

<b>1</b>	<b>Wstęp</b>	<b>5</b>
1.1	Opis problemu i koncepcja jego rozwiązania . . . . .	5
1.2	Cel i zakres pracy . . . . .	5
1.3	Omówienie pracy . . . . .	5
<b>2</b>	<b>Rozdziały teoretyczne</b>	<b>7</b>
2.1	Konkurencyjne rozwiązania . . . . .	7
2.1.1	On X . . . . .	7
2.1.2	Tasker . . . . .	7
<b>3</b>	<b>Projekt rozwiązania</b>	<b>9</b>
3.1	Definicja pojęć . . . . .	9
3.2	Przypadki użycia . . . . .	9
3.3	Moduły systemu. . . . .	9
3.4	Architektura systemu. . . . .	9
<b>4</b>	<b>Opis implementacji</b>	<b>11</b>
4.1	Użyte technologie . . . . .	11
4.1.1	Android . . . . .	11
4.1.2	Apache Commons . . . . .	11
4.1.3	Apache Server. . . . .	11
4.1.4	Git . . . . .	11
4.1.5	HTML 5 . . . . .	11
4.1.6	Hibernate . . . . .	11
4.1.7	JSON . . . . .	11
4.1.8	Java 6 . . . . .	12
4.1.9	JavaScript . . . . .	12
4.1.10	Apache Maven . . . . .	12
4.1.11	MySQL . . . . .	12
4.1.12	PHP . . . . .	12
4.1.13	RESTeasy . . . . .	12
4.1.14	SpringFramework . . . . .	12
4.1.15	Vaadin. . . . .	12

4.2	Użyte narzędzia . . . . .	13
4.2.1	Apache Tomcat . . . . .	13
4.2.2	Eclipse with Android developer tools. . . . .	13
4.2.3	String Tool Suite. . . . .	13
4.2.4	Emacs . . . . .	13
4.2.5	Git bash for windows . . . . .	14
4.2.6	Git for linux . . . . .	14
4.2.7	Github. . . . .	14
4.2.8	Latex . . . . .	14
4.2.9	Linux . . . . .	14
4.2.10	Notepad++. . . . .	14
4.2.11	Przeglądarki internetowe . . . . .	14
4.2.12	Windows. . . . .	14
4.3	Użyty sprzęt . . . . .	14
4.3.1	Komputery klasy PC . . . . .	14
4.3.2	LG Swift GT540 - Cyanogen based on Android 4.0.1 . . . . .	14
4.3.3	Media-Droid IMPERIUS EN3RGY MT7013 - Android 4.1.2 . . . . .	14
4.3.4	Motorola Defy MB525 - Cyanogen based on Android 4.3.1 . . . . .	14
4.3.5	Sony Xperia Arc S Lti18 - Android 4.0.4 . . . . .	14
4.4	Architektura klienta . . . . .	14
4.4.1	Moduł obsługi recept . . . . .	14
4.4.2	Moduły dostępu do systemu . . . . .	14
4.5	Architektura serwera . . . . .	14
4.5.1	Repozytorium recept . . . . .	14
4.5.2	Serwer recept grupowych . . . . .	14
<b>5</b>	<b>Zakończenie</b>	<b>15</b>
<b>A</b>	<b>Przewodnik użytkownika</b>	<b>17</b>
A.1	Opis Podfunkcjonalności . . . . .	17
A.1.1	Akcelerometr (YAccelerometerFeature.java) . . . . .	17
A.1.2	Battery (YBatteryFeature.java) . . . . .	17
A.1.3	SMS (YSMSFeature.java) . . . . .	17
A.1.4	Wifi (YWifiFeature.java) . . . . .	17
A.1.5	GPS (YGPSFeature.java) . . . . .	17
A.1.6	Sound (YSoundFeature.java) . . . . .	17
A.1.7	RawPlayer (YRawPlayerFeature.java) . . . . .	17
A.1.8	Group (YGroupFeature.java) . . . . .	17
A.1.9	Geocoder (YGeocoderFeature.java) . . . . .	17
A.1.10	Time (YTimeFeature.java) . . . . .	18
A.1.11	AudioManager (YAudioManager.java) . . . . .	18
A.1.12	Text (YTextFeature.java) . . . . .	18
A.1.13	Internet (YInternetFeature.java) . . . . .	18
A.1.14	Calls (YCallsFeature.java) . . . . .	18
A.1.15	Notification (YNotificationFeature.java) . . . . .	18

Spis treści

3

**Bibliografia**

**19**



# Wstęp

Wprowadzenie do tematu...

**1.1 Opis problemu i koncepcja jego rozwiązania**

**1.2 Cel i zakres pracy**

**1.3 Omówienie pracy**





# Rozdziały teoretyczne

## 2.1 Konkurencyjne rozwiązania

### 2.1.1 On X

Aplikacja Microsoftu umożliwiającą kontrolowanie telefonu z Androidem używając kodu w JavaScriptcie. Umożliwia wysyłanie Zasad (Rules) na telefon poprzez stronę internetową. Dostęp do funkcjonalności Androida jest zapewniony przez api w postaci Wyzwalaczy (Triggers) i Akcji (Actions). Cały system jest niestety połączony z Facebookiem i wymaga posiadania tam konta. Na podstawie [1].

### 2.1.2 Tasker

Więcej informacji można znaleźć w książce [2].



# Projekt rozwiązania

## 3.1 Definicja pojęć

- Podfunkcjonalność (ang. Feature) – Część biblioteki zapewniająca Receptom dostęp do pozdbioru funkcjonalności Androida.
- Zdarzenie (ang. Event) – Zmiana stanu systemu, która powoduje uruchomienie kodu Recepty.
- Recepta (ang. Recipe) – Napisany przez użytkownika fragment kodu opisujący, co ma się zdarzyć po spełnieniu pewnych warunków.
- Targowisko (ang. Market) – Aplikacja internetowa pozwalająca tworzyć i pobierać Recepty.
- Aplikacja – Aplikacja androidowa wykorzystująca bibliotekę if{Y}.
- Serwer Grup – Komputer z działającą aplikacją, która zarządza grupami użytkowników i Zdarzeniami Grupowymi.
- Zdarzenie Grupowe – Zdarzenie związane z Grupą, wysyłane lub odbierane przez Aplikację z Serwera Grup.
- Grupa – Zbiór użytkowników identyfikowalny przez nazwę zdefiniowany na Serwerze Grup.

## 3.2 Przypadki użycia

## 3.3 Moduły systemu

## 3.4 Architektura systemu



# Opis implementacji

## 4.1 Użyte technologie

W tej części zaprezentowano opis technologii użytych bezpośrednio w implementacji składowych platformy.

### 4.1.1 Android

### 4.1.2 Apache Commons

### 4.1.3 Apache Server

### 4.1.4 Git

### 4.1.5 HTML 5

### 4.1.6 Hibernate

Narzędzie odwzorowań obiektowo-relacyjnych (ang. object-relation mapping, ORM) rozwijany na zasadzie wolnego oprogramowania. Umożliwia odwzorowania obiektowo-relacyjne, pamięć podręczną, leniwe ładowanie(ang. Lazy loading),chciwe pobieranie oraz rozproszoną pamięć podręczną.

### 4.1.7 JSON

JavaScript Object Notation, lekki format danych wywodzący się z języka JavaScript(jest jego podzbiorem).

### **4.1.8 Java 6**

### **4.1.9 JavaScript**

### **4.1.10 Apache Maven**

Narzędzie automatycznego budowania oprogramowania dla języka JAVA. Głównymi problemami jakie rozwiązuje Maven przy budowaniu aplikacji są: zarządzanie zależnościami, możliwość wieloma modułami, wsparcie dla testów.

### **4.1.11 MySQL**

System zarządzania relacyjnymi bazami danych. Jest to wolne oprogramowanie szczególnie upodobane przez twórców aplikacji internetowych. Bardzo dobrze współpracuje z językami takimi jak PHP czy Java

### **4.1.12 PHP**

Obiektowy język programowania dedykowany generowaniu stron internetowych w czasie rzeczywistym. Szczególnie użyteczny w przypadku tworzenia prototypów tudzież niewielkich projektów wymagających stosunkowo niskiego poziomu abstrakcji.

### **4.1.13 RESTeasy**

Framework oprogramowania służący do tworzenia aplikacji rozproszonych, oparty na wzorcu architektury oprogramowania Representational State Transfer (REST).

### **4.1.14 SpringFramework**

Framework (Szkielet) tworzenia aplikacji w języku Java a w szczególności JavaEE. Do najważniejszych funkcji Springa zalicza się wstrzykiwanie zależności (ang. dependency injection, DI) oraz programowanie aspektowe (ang. aspect-oriented programming, AOP).

### **4.1.15 Vaadin**

Framework sieciowy służący do tworzenia aplikacji sieciowych w szczególności interfejsu użytkownika w oparciu o Google Web Toolkit (GWT) w języku JAVA.

## 4.2 Użyte narzędzia

### 4.2.1 Apache Tomcat

### 4.2.2 Eclipse with Android developer tools

### 4.2.3 String Tool Suite

Zintegrowane środowisko programistyczne oparte o Eclipse dostosowany do SpringFramework.

### 4.2.4 Emacs

Popularny, w pełni rozszerzalny edytor tekstowy spotykany głównie w systemach operacyjnych z rodziny Unix. Używany przez wysokiej klasy programistów oraz naukowców na całym świecie.

#### **4.2.5 Git bash for windows**

#### **4.2.6 Git for linux**

#### **4.2.7 Github**

#### **4.2.8 Latex**

#### **4.2.9 Linux**

#### **4.2.10 Notepad++**

#### **4.2.11 Przeglądarki internetowe**

#### **4.2.12 Windows**

### **4.3 Użyty sprzęt**

#### **4.3.1 Komputery klasy PC**

#### **4.3.2 LG Swift GT540 - Cyanogen based on Android 4.0.1**

#### **4.3.3 Media-Droid IMPERIUS EN3RGY MT7013 - Android 4.1.2**

#### **4.3.4 Motorola Defy MB525 - Cyanogen based on Android 4.3.1**

#### **4.3.5 Sony Xperia Arc S Lti18 - Android 4.0.4**

### **4.4 Architektura klienta**

#### **4.4.1 Moduł obsługi recept**

#### **4.4.2 Moduły dostępu do systemu**

### **4.5 Architektura serwera**

#### **4.5.1 Repozytorium recept**

#### **4.5.2 Serwer recept grupowych**



# Zakończenie



# Przewodnik użytkownika

## A.1 Opis Podfunkcjonalności

### A.1.1 Akcelerometr (YAccelerometerFeature.java)

Umożliwia reagowanie na odczyty akcelerometru wbudowanego w urządzenie.

### A.1.2 Battery (YBatteryFeature.java)

Umożliwia reagowanie na zmiany poziomu baterii urządzenia.

### A.1.3 SMS (YSMSFeature.java)

Umożliwia wysyłanie wiadomości SMS oraz reagowanie na wiadomości przychodzące.

### A.1.4 Wifi (YWifiFeature.java)

Umożliwia włączanie i wyłączanie modułu WiFi urządzenia.

### A.1.5 GPS (YGPSFeature.java)

Umożliwia śledzenie pozycji urządzenia za pomocą modułu GPS.

### A.1.6 Sound (YSoundFeature.java)

### A.1.7 RawPlayer (YRawPlayerFeature.java)

### A.1.8 Group (YGroupFeature.java)

### A.1.9 Geocoder (YGeocoderFeature.java)

Umożliwia pobranie adresu związanego z podaną długością i szerokością geograficzną.

**A.1.10 Time (YTimeFeature.java)****A.1.11 AudioManager (YAudioManager.java)****A.1.12 Text (YTextFeature.java)****A.1.13 Internet (YInternetFeature.java)**

Umożliwia wysyłanie i pobieranie danych z podanego adresu.

**A.1.14 Calls (YCallsFeature.java)**

Umożliwia reagowanie na połączenia przychodzące i inicjowanie połączeń wychodzących.

**A.1.15 Notification (YNotificationFeature.java)**

Umożliwia wyświetlanie powiadomień w interfejsie graficznym urządzenia.

# Bibliografia

- [1] Projekt on{X} <http://www.onx.ms/#!findOutMorePage>. Ostatnio odwiedzone 6/02/13.
- [2] A. Tanenbaum. *Operating Systems Design and Implementation*. Prentice Hall, 2006.
- [3] C. Walls. *Spring in action, 3rd edition*. Manning Publication Co, 2011.
- [4] Vaadin <https://vaadin.com/book/vaadin6/-/page/preface.html>
- [5] E. Gamma. *Design Patterns, First edition*. Person Education, Inc, 1995.