

# **Aplikacja mobilna dla biegacza - GitFitScrub**

*Michał Ptak, Katarzyna Poręba, Jan Michalik*

Zespołowe przedsięwzięcie inżynierskie

Informatyka

Rok. akad. 2017/2018, sem. I

Prowadzący: dr hab. Marcin Mazur

# Spis treści

<b>1</b>	<b>Opis projektu</b>	<b>3</b>
1.1	Członkowie zespołu . . . . .	3
1.2	Cel projektu (produkt) . . . . .	3
1.3	Potencjalny odbiorca produktu (klient) . . . . .	3
1.4	Metodyka . . . . .	3
<b>2</b>	<b>Wymagania użytkownika</b>	<b>3</b>
2.1	User story 1 . . . . .	3
2.2	User story 2 . . . . .	3
2.3	User story 3 . . . . .	3
2.4	User story 4 . . . . .	3
2.5	User story 5 . . . . .	4
2.6	User story 6 . . . . .	4
2.7	User story 7 . . . . .	4
2.8	User story 8 . . . . .	4
2.9	User story 9 (opcjonalne) . . . . .	4
2.10	User story 10 (opcjonalne) . . . . .	4
2.11	User story 11 (opcjonalne) . . . . .	4
2.12	User story 12 (opcjonalne) . . . . .	4
2.13	User story 13 (opcjonalne) . . . . .	4
2.14	User story 14 (opcjonalne) . . . . .	4
2.15	User story 15 (opcjonalne) . . . . .	5
2.16	User story 16 (opcjonalne) . . . . .	5
2.17	User story 17 (opcjonalne) . . . . .	5
2.18	User story 18 (opcjonalne) . . . . .	5
2.19	User story 19 (opcjonalne) . . . . .	5
2.20	User story 20 (opcjonalne) . . . . .	5
<b>3</b>	<b>Harmonogram</b>	<b>5</b>
3.1	Rejestr zadań (Product Backlog) . . . . .	5
3.2	Sprint 1 . . . . .	5
3.3	Sprint 2 . . . . .	6
3.4	Sprint 3 . . . . .	6
3.5	Sprint 4 . . . . .	6
3.6	Sprint 5 . . . . .	6
3.7	Sprint 6 . . . . .	6
<b>4</b>	<b>Product Backlog</b>	<b>7</b>
4.1	Backlog Item 1 . . . . .	7
4.2	Backlog Item 2 . . . . .	7
4.3	Backlog Item 3 . . . . .	7
4.4	Backlog Item 4 . . . . .	7
4.5	Backlog Item 5 . . . . .	8
4.6	Backlog Item 6 . . . . .	8
4.7	Backlog Item 7 . . . . .	8
4.8	Backlog Item 8 . . . . .	8
4.9	Backlog Item 9 . . . . .	9
4.10	Backlog Item 10 . . . . .	9

4.11	Backlog Item 11 . . . . .	9
4.12	Backlog Item 12 . . . . .	10
4.13	Backlog Item 13 . . . . .	10
4.14	Backlog Item 14 (opcjonalne) . . . . .	10
4.15	Backlog Item 15 (opcjonalne) . . . . .	10
4.16	Backlog Item 16 (opcjonalne) . . . . .	11
4.17	Backlog Item 17 (opcjonalne) . . . . .	11
4.18	Backlog Item 18 (opcjonalne) . . . . .	11
4.19	Backlog Item 19 (opcjonalne) . . . . .	11
4.20	Backlog Item 20 (opcjonalne) . . . . .	12
4.21	Backlog Item 21 (opcjonalne) . . . . .	12
4.22	Backlog Item 22 (opcjonalne) . . . . .	12
4.23	Backlog Item 23 (opcjonalne) . . . . .	12
4.24	Backlog Item 24 (opcjonalne) . . . . .	13
4.25	Backlog Item 25 (opcjonalne) . . . . .	13
4.26	Backlog Item 26 (opcjonalne) . . . . .	13
4.27	Backlog Item 27 (opcjonalne) . . . . .	13
4.28	Backlog Item 28 (opcjonalne) . . . . .	14
<b>5</b>	<b>Sprint 1</b>	<b>14</b>
5.1	Cel . . . . .	14
5.2	Sprint Planning/Backlog . . . . .	14
5.3	Realizacja . . . . .	14
5.4	Sprint Review/Demo . . . . .	17
<b>6</b>	<b>Sprint 2</b>	<b>18</b>
6.1	Cel . . . . .	18
6.2	Sprint Planning/Backlog . . . . .	18
6.3	Realizacja . . . . .	18
6.4	Sprint Review/Demo . . . . .	25
<b>7</b>	<b>Sprint 3</b>	<b>25</b>
7.1	Cel . . . . .	25
7.2	Sprint Planning/Backlog . . . . .	26
7.3	Realizacja . . . . .	26
7.4	Sprint Review/Demo . . . . .	27
<b>8</b>	<b>Sprint 4</b>	<b>27</b>
8.1	Cel . . . . .	27
8.2	Sprint Planning/Backlog . . . . .	27
8.3	Realizacja . . . . .	27
8.4	Sprint Review/Demo . . . . .	28
<b>9</b>	<b>Sprint 5</b>	<b>28</b>
9.1	Cel . . . . .	28
9.2	Sprint Planning/Backlog . . . . .	28
9.3	Realizacja . . . . .	28
9.4	Sprint Review/Demo . . . . .	29

# 1 Opis projektu

## 1.1 Członkowie zespołu

1. Michał Ptak (kierownik projektu).
2. Katarzyna Poręba.
3. Jan Michalik.

## 1.2 Cel projektu (produkt)

Naszym celem jest zbudowanie aplikacji mobilnej na system operacyjny Android przy użyciu map Google, która pobiera informacje poruszającej się osoby z modułu GPS i gromadzi statystyki dotyczące długości trasy, prędkości czy spalanych kalorii, a także zapisuje poczynania w bazie danych.

## 1.3 Potencjalny odbiorca produktu (klient)

Klient jest biegaczem oczekującym w pełni darmowej aplikacji bez reklam, która będzie śledzić trasę, po której się porusza oraz wyświetlać aktualne statystyki. Aktualnie żadna dostępna na rynku aplikacja nie spełnia odbiorcy produktu.

## 1.4 Metodyka

Projekt będzie realizowany przy użyciu (zaadaptowanej do istniejących warunków) metodyki *Scrum*.

# 2 Wymagania użytkownika

## 2.1 User story 1

Jako użytkownik, chcę mieć dostęp do mapy podczas treningu, aby wiedzieć jaką trasą się poruszam.

## 2.2 User story 2

Jako użytkownik, chcę mieć możliwość wciśnięcia przycisku "Start", aby rozpocząć pomiar.

## 2.3 User story 3

Jako użytkownik, chcę mieć możliwość wciśnięcia przycisku "Stop", aby zakończyć pomiar.

## 2.4 User story 4

Jako użytkownik, chcę wiedzieć czas, który upłynął od rozpoczęcia pomiaru biegu, aby kontrolować jego upływ.

## **2.5 User story 5**

Jako użytkownik, chcę widzieć przebyty dystans, aby kontrolować jego długość.

## **2.6 User story 6**

Jako użytkownik, chcę widzieć prędkość chwilową, z jaką się poruszam, aby zachować określone tempo.

## **2.7 User story 7**

Jako użytkownik, chcę zobaczyć podsumowanie odbytego treningu, aby ocenić jego rezultat.

## **2.8 User story 8**

Jako użytkownik, chcę mieć możliwość włączenia trybu nocnego, aby zmniejszyć zmęczenie wzroku po zmroku.

## **2.9 User story 9 (opcjonalne)**

Jako użytkownik niezarejestrowany, chcę mieć możliwość rejestracji, aby móc się zalogować.

## **2.10 User story 10 (opcjonalne)**

Jako użytkownik zarejestrowany, chcę mieć możliwość zalogowania się, aby móc korzystać ze wszystkich funkcji aplikacji.

## **2.11 User story 11 (opcjonalne)**

Jako użytkownik zalogowany, chcę mieć możliwość wylogowania się, aby móc zalogować się na inne konto.

## **2.12 User story 12 (opcjonalne)**

Jako użytkownik zalogowany, chcę mieć możliwość ustawienia swojego pseudonimu, aby wiedzieć czy jestem zalogowany na swoim koncie.

## **2.13 User story 13 (opcjonalne)**

Jako użytkownik zalogowany, chcę aby statystyki z odbytych treningów zostały zapisane w historii, celem ich późniejszej analizy.

## **2.14 User story 14 (opcjonalne)**

Jako użytkownik zalogowany, chcę widzieć w statystykach spalone kalorie, by móc dostosować swoją dietę.

### **2.15 User story 15 (opcjonalne)**

Jako użytkownik zalogowany, chcę mieć możliwość usunięcia (całej lub części) historii treningów, aby rozpocząć nowy cykl treningowy.

### **2.16 User story 16 (opcjonalne)**

Jako użytkownik zalogowany, chcę by aplikacja pamiętała ostatnio użyty login, aby nie wpisywać go ręcznie przy kolejnym logowaniu.

### **2.17 User story 17 (opcjonalne)**

Jako użytkownik zalogowany, chcę mieć możliwość ręcznego zatrzymania pomiaru czasu oraz dystansu, aby napotkane przeszkody (np. czerwone światło) nie fałszowały statystyk.

### **2.18 User story 18 (opcjonalne)**

Jako użytkownik zalogowany, chcę mieć możliwość podjęcia wyzwania pokonania określonych odległości (np. z Ziemi do Księżyca), aby mieć motywację do kontynuowania treningu.

### **2.19 User story 19 (opcjonalne)**

Jako użytkownik zalogowany, chcę mieć możliwość udostępnienia swoich osiągnięć na portalach społecznościowych, aby podzielić się rezultatami ze znajomymi.

### **2.20 User story 20 (opcjonalne)**

Jako użytkownik, chcę mieć możliwość ustawienia powiadomień po określonym przez siebie odcinku trasy, by nie być zmuszonym do ciągłego sprawdzania aplikacji.

## **3 Harmonogram**

### **3.1 Rejestr zadań (Product Backlog)**

- Data rozpoczęcia: 24.10.2017.
- Data zakończenia: 31.10.2017.

### **3.2 Sprint 1**

- Data rozpoczęcia: 31.10.2017.
- Data zakończenia: 14.11.2017.
- Scrum Master: Katarzyna Poręba.
- Product Owner: Michał Ptak.
- Development Team: Michał Ptak, Katarzyna Poręba, Jan Michalik.

### **3.3 Sprint 2**

- Data rozpoczęcia: 14.11.2017.
- Data zakończenia: 28.11.2017.
- Scrum Master: Michał Ptak.
- Product Owner: Jan Michalik.
- Development Team: Michał Ptak, Katarzyna Poręba, Jan Michalik.

### **3.4 Sprint 3**

- Data rozpoczęcia: 28.11.2017.
- Data zakończenia: 12.12.2017.
- Scrum Master: Jan Michalik.
- Product Owner: Katarzyna Poręba.
- Development Team: Michał Ptak, Katarzyna Poręba, Jan Michalik.

### **3.5 Sprint 4**

- Data rozpoczęcia: 12.12.2017.
- Data zakończenia: 19.12.2017.
- Scrum Master: Katarzyna Poręba.
- Product Owner: Michał Ptak.
- Development Team: Michał Ptak, Katarzyna Poręba, Jan Michalik.

### **3.6 Sprint 5**

- Data rozpoczęcia: 19.12.2017.
- Data zakończenia: 09.01.2018.
- Scrum Master: Michał Ptak.
- Product Owner: Jan Michalik.
- Development Team: Michał Ptak, Katarzyna Poręba, Jan Michalik.

### **3.7 Sprint 6**

- Data rozpoczęcia: 09.01.2018.
- Data zakończenia: 16.01.2018.
- Scrum Master: Michał Ptak.
- Product Owner: Katarzyna Poręba.
- Development Team: Michał Ptak, Katarzyna Poręba, Jan Michalik.

## 4 Product Backlog

### 4.1 Backlog Item 1

**Tytuł zadania.** Przygotowanie zestawu narzędzi SDK.

**Opis zadania.** Dodanie do zestawu SDK: Google Services, nową dystrybucję Androida (wersja Oreo), debugowanie poprzez USB. Wykonanie prostego programu testującego działanie kompilatora oraz zainstalowanej dystrybucji.

**Priorytet.** 10.

**Definition of Done.** Poprawne zainstalowanie wybranych narzędzi SDK na stacjach roboczych.

### 4.2 Backlog Item 2

**Tytuł zadania.** Stworzenie wstępnej wersji aplikacji.

**Opis zadania.** Utworzenie aplikacji zawierającej puste menu główne.

**Priorytet.** 10.

**Definition of Done.** Aplikacja po uruchomieniu wyświetla menu główne.

### 4.3 Backlog Item 3

**Tytuł zadania.** Dodanie uprawnień aplikacji.

**Opis zadania.** Dodanie uprawnień aplikacji do korzystania z Internetu i modułu GPS, co weryfikowane jest w 4.4.

**Priorytet.** 10.

**Definition of Done.** Aplikacja posiada możliwość korzystania z Internetu i nawigacji.

### 4.4 Backlog Item 4

**Tytuł zadania.** Stworzenie przycisku "Rozpocznij" w menu głównym oraz nadanie mu funkcjonalności.

**Opis zadania.** Utworzenie nowej aktywności uruchamiającej mapę Google i nanoszącej na nią znacznika wskazującego geolokalizację użytkownika. Utworzenie przycisku i nadanie mu akcji uruchomienia tejże aktywności. Zaprojektowanie szaty graficznej menu głównego i stworzenie logo oraz ikony aplikacji.

**Priorytet.** 8.



**Definition of Done.** Po naciśnięciu przycisku "Rozpocznij" uruchomi moduł GPS i pokaże znacznik wskazujący aktualną pozycję użytkownika na mapie.

## 4.5 Backlog Item 5

**Tytuł zadania.** Stworzenie alertu o braku aktywacji modułu GPS.

**Opis zadania.** Utworzenie metody w klasie MapsActivity sprawdzającej stan aktywności modułu GPS, oraz powiadamiającej użytkownika o wyłączonym GPS-ie. Po zatwierdzeniu alertu aplikacja przenosi użytkownika do ustawień systemowych Android™.

**Priorytet.** 6.

**Definition of Done.**

## 4.6 Backlog Item 6

**Tytuł zadania.** Pobranie informacji z modułu GPS.

**Opis zadania.** Utworzenie nowej aktywności pobierającej surowe dane geograficzne oraz gromadzenie ich w pamięci operacyjnej. Weryfikacja poprawności uzyskanych danych poprzez wydruk w konsoli dewelopera.

**Priorytet.** 9.

**Definition of Done.** Surowe dane geograficzne znajdują się w pamięci operacyjnej systemu.

## 4.7 Backlog Item 7

**Tytuł zadania.** Utworzenie pustego panelu statystyk i przycisku "Start" oraz nadanie mu funkcjonalności.

**Opis zadania.** Utworzenie nowego obiektu button wywołującego metody rozpoczynające pomiary po jego wciśnięciu. Wydzielenie obszaru, na którym będą wyświetlane statystyki.

**Priorytet.** 9.

**Definition of Done.** U dołu mapy, wyświetlającej naszą aktualną pozycję, widnieje przycisk rozpoczynający trening, a pod nim panel ze statystykami.

## 4.8 Backlog Item 8

**Tytuł zadania.** Przetworzenie zgromadzonych informacji.

**Opis zadania.** Utworzenie funkcji kalkulujących dystans, prędkość, ilość spalonych kcal i upływ czasu, które zostają wywołane po naciśnięciu przycisku "Rozpocznij nowy trening". Przechowywanie wartości zwracanych przez funkcje w pamięci operacyjnej systemu.

**Priorytet.** 9.

**Definition of Done.** Dane zwracane przez funkcję mają odwzorowanie w rzeczywistości.

#### 4.9 Backlog Item 9

**Tytuł zadania.** Modyfikacja panelu zawierającego statystyki aktualnego treningu.

**Opis zadania.** Utworzenie nowej aktywności pobierającej przetworzone dane geograficzne. Określenie pozycji wyświetlanych parametrów typu: czas, dystans, prędkość. Sprzężenie tejże aktywności z pozycjami statystyk celem ich wyświetlenia użytkownikowi.

**Priorytet.** 7.

**Definition of Done.** Wyświetlenie przetworzonych informacji pobranych z modułu GPS.

#### 4.10 Backlog Item 10

**Tytuł zadania.** Rysowanie polilinii na mapie.

**Opis zadania.** Utworzenie metody tworzącej ślad na mapie przemieszczającego się użytkownika, na podstawie informacji z modułu GPS.

**Priorytet.** 8.

**Definition of Done.**

#### 4.11 Backlog Item 11

**Tytuł zadania.** Stworzenie przycisku "Zakończ trening" oraz nadanie mu funkcjonalności.

**Opis zadania.** Utworzenie nowej aktywności wywołującej metodę odpowiadającą za zakończenie pomiarów i rysowanie polilinii na mapie, a także otwierającą panel z podsumowaniem treningu — wszystkich wyznaczonych statystyk.

**Priorytet.** 9.

**Definition of Done.** Po naciśnięciu "Zakończ trening" aplikacja zatrzymuje pomiary i wyświetla podsumowanie treningu.

#### 4.12 Backlog Item 12

**Tytuł zadania.** Stworzenie przycisku "Ustawienia" w menu głównym oraz nadanie mu funkcjonalności.

**Opis zadania.** Stworzenie nowej aktywności uruchamiającej listę dostępnych ustawień.

**Priorytet.** 5.

**Definition of Done.** Po naciśnięciu przycisku "Ustawienia" w menu głównym otwiera się lista ustawień.

#### 4.13 Backlog Item 13

**Tytuł zadania.** Dodanie funkcji trybu nocnego do panelu ustawień.

**Opis zadania.** Dodanie przełącznika do panelu ustawień zmieniającego styl mapy oraz panelu statystyk na "nocny" lub "dzienny".

**Priorytet.** 5.

**Definition of Done.** Po naciśnięciu przełącznika mapa oraz panel statystyk zmieniają styl wyglądu na "dzienny" lub "nocny".

#### 4.14 Backlog Item 14 (opcjonalne)

**Tytuł zadania.** Dodanie aktywności logowania użytkownika.

**Opis zadania.** Dodanie możliwości logowania za pośrednictwem konta Google.

**Priorytet.** 6.

**Definition of Done.** Konto Google użytkownika zostaje połączone z aplikacją GitFitScrub. Poprawność połączenia weryfikowana na podstawie strony "Aplikacje połączone z Twoim kontem" <https://myaccount.google.com/permissions>.

#### 4.15 Backlog Item 15 (opcjonalne)

**Tytuł zadania.** Dodanie aktywności wylogowania użytkownika.

**Opis zadania.** Dodanie możliwości wylogowania się z konta Google.

**Priorytet.** 6.

**Definition of Done.** Konto Google użytkownika zostaje rozłączone z aplikacji GitFitScrub.

#### 4.16 Backlog Item 16 (opcjonalne)

**Tytuł zadania.** Stworzenie przycisku "Zaloguj" w ustawieniach oraz nadanie mu funkcjonalności.

**Opis zadania.** Wywołanie metody połączenia się z kontem Google.

**Priorytet.** 5.

**Definition of Done.** Po wprowadzeniu poprawnych danych użytkownik zostaje pomyślnie uwierzytelniony.

#### 4.17 Backlog Item 17 (opcjonalne)

**Tytuł zadania.** Stworzenie przycisku "Wyloguj" w ustawieniach oraz nadanie mu funkcjonalności.

**Opis zadania.** Wywołanie metody rozłączenia się z kontem Google.

**Priorytet.** 5.

**Definition of Done.** Po naciśnięciu przycisku użytkownik zostaje pomyślnie wylogowany.

#### 4.18 Backlog Item 18 (opcjonalne)

**Tytuł zadania.** Utworzenie miejsca, w którym archiwizowane będą statystyki użytkownika zalogowanego.

**Opis zadania.** Wygenerowanie certyfikatu do autoryzacji komunikacji aplikacji z chmurą. Sprawdzenie komunikacji baza - aplikacja poprzez wysłanie testowych informacji na chmurę.

**Priorytet.** 5.

**Definition of Done.** Aplikacja bez przeszkód komunikuje się z bazą danych w chmurze Google Drive.

#### 4.19 Backlog Item 19 (opcjonalne)

**Tytuł zadania.** Ustawienie pseudonimu użytkownika zalogowanego.

**Opis zadania.** Utworzenie nowego pola w ustawieniach, którego celem będzie ustawienie aktualnego pseudonimu.

**Priorytet.** 3.

**Definition of Done.** Użytkownik zalogowany ma możliwość ustawienia pseudonimu.

#### 4.20 Backlog Item 20 (opcjonalne)

**Tytuł zadania.** Stworzenie przycisku "Historia" w menu głównym oraz nadanie mu funkcjonalności.

**Opis zadania.** Utworzenie nowej aktywności do wyświetlania historii treningów.

**Priorytet.** 5.

**Definition of Done.** Użytkownik po naciśnięciu przycisku historia widzi listę odbytych treningów.

#### 4.21 Backlog Item 21 (opcjonalne)

**Tytuł zadania.** Gromadzenie danych dotyczących treningów w chmurze Google Drive.

**Opis zadania.** Po nawiązaniu połączenia z chmurą następuje przesył zgromadzonych informacji do bazy danych. Weryfikacja danych zgromadzonych w bazie poprzez sprawdzenie zawartości za pomocą interfejsu administratora.

**Priorytet.** 5.

**Definition of Done.** Dane są poprawnie gromadzone w bazie danych.

#### 4.22 Backlog Item 22 (opcjonalne)

**Tytuł zadania.** Prezentowanie danych w historii.

**Opis zadania.** Stworzenie nowej aktywności pobierającej dane z bazy oraz prezentującej je użytkownikowi w tabeli, w panelu Historia, dostępnego z menu głównego.

**Priorytet.** 5.

**Definition of Done.** Dane są poprawnie gromadzone w bazie danych.

#### 4.23 Backlog Item 23 (opcjonalne)

**Tytuł zadania.** Usuwanie danych z historii.

**Opis zadania.** Utworzenie metody wysyłające polecenie kasujące dane zapisane w historii.

**Priorytet.** 4.

**Definition of Done.** Dane są poprawnie usuwane z bazy danych i nie znajdują się już w panelu historia.

#### 4.24 Backlog Item 24 (opcjonalne)

**Tytuł zadania.** Zapamiętanie loginu ostatniego użytkownika.

**Opis zadania.** Wykorzystanie klasy `SharedPreferences` w celu zapamiętania loginu użytkownika. Sprawdzenie poprawności funkcjonalności zapisanego loginu za pomocą metody `get` wywołanej na obiekcie `SharedPreferences`.

**Priorytet.** 4.

**Definition of Done.** Po wylogowaniu się w polu login widnieje ostatnio użyta nazwa użytkownika.

#### 4.25 Backlog Item 25 (opcjonalne)

**Tytuł zadania.** Zatrzymanie pomiarów czasu i dystansu.

**Opis zadania.** Stworzenie przycisku pauza. Dodanie metody wstrzymującej pomiary czasu i dystansu do klasy odpowiadającej za pomiary oraz ich powiązanie.

**Priorytet.** 5.

**Definition of Done.** Po wciśnięciu przycisku pauza pomiary czasu i dystansu są wstrzymane. Po ponownym naciśnięciu przycisku pomiary są wznowiane.

#### 4.26 Backlog Item 26 (opcjonalne)

**Tytuł zadania.** Utworzenie wyzwań.

**Opis zadania.** Stworzenie listy wyzwań. Dodanie do ustawień możliwości pojęcia własnego wyzwania polegającego na pokonaniu określonego dystansu w zadany przedział czasowy.

**Priorytet.** 3.

**Definition of Done.** Po aktywowaniu wyzwania odległości z historii (od momentu rozpoczęcia wyzwania) są dodawane do progresji aktywnego wyzwania.

#### 4.27 Backlog Item 27 (opcjonalne)

**Tytuł zadania.** Udostępnianie wyniku podsumowania treningu na portalu społecznościom.

**Opis zadania.** Dodanie przycisku udostępnij w panelu podsumowanie, który wywołuje metodę wysyłającą dane o treningu na portal społecznościowy.

**Priorytet.** 3.

**Definition of Done.** Po udostępnieniu informacji dane są widoczne na profilu portalu społecznościowego użytkownika.

#### 4.28 Backlog Item 28 (opcjonalne)

**Tytuł zadania.** Powiadomienia dźwiękowe.

**Opis zadania.** Dodanie metody odtwarzającej alert dźwiękowy po przebiegnięciu określonego czasu/dystansu. Dodanie do panelu ustawień pozycji powiadomienia, w której będzie można wybrać plik dźwiękowy oraz kryterium (czas/dystans) wywołania tejże metody.

**Priorytet.** 3.

**Definition of Done.** Po przebiegnięciu określonego czasu/dystansu użytkownik usłyszy alert dźwiękowy.

## 5 Sprint 1

### 5.1 Cel

Utworzenie wersji bazowej aplikacji zawierającej ekran menu głównego i przycisk Rozpocznij wyświetlający mapę Google i aktualną pozycję użytkownika.

### 5.2 Sprint Planning/Backlog

**Tytuł zadania.** Przygotowanie zestawu narzędzi SDK.

- Estymata: M.

**Tytuł zadania.** Stworzenie wstępnej wersji aplikacji.

- Estymata: L.

**Tytuł zadania.** Dodanie uprawnień aplikacji.

- Estymata: S.

**Tytuł zadania.** Stworzenie przycisku "Rozpocznij" w menu głównym oraz nadanie mu funkcjonalności.

- Estymata: L.

### 5.3 Realizacja

**Tytuł zadania.** Przygotowanie zestawu narzędzi SDK.

**Wykonawca.** Jan Michalik, Michał Ptak, Katarzyna Poręba.

**Realizacja.** Uruchamiamy SDK Manager, następnie instalujemy składniki: Android SDK Tools, Platform Tools, Build Tools, Android 8.1.0 (API 27), Google Play services, Google USB Driver. Czasochłonność zadania pokryła estymatę.

**Tytuł zadania.** Stworzenie wstępnej wersji aplikacji.

**Wykonawca.** Katarzyna Poręba.

**Realizacja.** Utworzenie nowego pustego projektu Android Studio. Dodanie nowej aktywności - MainMenu. Ustawienie jej jako domyślny ekran po uruchomieniu aplikacji.

```
<activity
    android:name=".MainMenu"
    android:label="GitFitScrub"
    android:theme="@style/AppTheme">
    <intent-filter>
        <action android:name="android.intent.action.MAIN" />
        <category android:name="android.intent.category.LAUNCHER"
        />
    </intent-filter>
</activity>
```

**Tytuł zadania.** Dodanie uprawnień aplikacji.

**Wykonawca.** Michał Ptak.

**Realizacja.** Uzupełnienie pliku AndroidManifest.xml o polecenia zezwalające na dostęp do Internetu i modułu GPS. Kompilacja kodu i wygenerowanie pliku .apk, następnie instalacja na urządzeniu z systemem Android. Podczas instalacji urządzenie poprawnie wyświetla informacje o wykorzystywanych funkcjonalnościach. Plik AndroidManifest.xml:

```
<uses-permission android:name="android.permission.INTERNET"/>
<uses-permission android:name="android.permission.
    ACCESS_FINE_LOCATION" />
<uses-permission android:name="android.permission.
    ACCESS_COARSE_LOCATION" />
```

**Tytuł zadania.** Stworzenie przycisku "Rozpocznij" w menu głównym oraz nadanie mu funkcjonalności.

**Wykonawca.** Jan Michalik.



**Realizacja.** Uzupełnienie pliku activity\_main\_menu.xml o polecenia tworzące przycisk "Rozpocznij", oraz nadanie mu określonego wyglądu.

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res
    /android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="#dddddd"
    tools:context="pl.ppm.gitfitscrub.MainMenu">

    <ImageView
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_above="@+id/button2"
        android:layout_alignParentStart="true"
        android:src="@drawable/capture" />

    <Button
        android:id="@+id/button2"
        style="@style/Widget.AppCompat.Button.Colored"
        android:layout_width="400px"
        android:layout_height="wrap_content"
        android:layout_alignStart="@+id/button3"
        android:layout_centerVertical="true"
        android:layout_marginBottom="20dp"
        android:onClick="run"
        android:text="Rozpocznij"
        android:textSize="24sp" />
</RelativeLayout>
```

Nadanie funkcjonalności przyciskowi. Dodanie kodu do pliku MainMenu.java, który spowoduje, że po naciśnięciu przycisku otworzy się mapa z aktualną pozycją użytkownika.

Plik MainMenu.java - menu główne:

```
public class MainMenu extends AppCompatActivity {
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main_menu);
    }
    public void run(View view) {
        Intent intent = new Intent(getApplicationContext(),
        MapsActivity.class);
        startActivity(intent);
    }
};
```

Plik MapsActivity.java - ekran mapy:

```

public class MapsActivity extends FragmentActivity implements
    OnMapReadyCallback {

    private GoogleMap mMap;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate ( savedInstanceState );
        setContentView ( R.layout.activity_maps );
        SupportMapFragment mapFragment = (SupportMapFragment)
        getSupportFragmentManager ()
            .findFragmentById ( R.id.map );
        mapFragment.getMapAsync ( this );
    }

    @Override
    public void onMapReady(GoogleMap googleMap) {
        mMap = googleMap;
        LatLng ns = new LatLng ( posY, posX );
        if (ActivityCompat.checkSelfPermission ( this, Manifest.
        permission.ACCESS_FINE_LOCATION ) == PackageManager.
        PERMISSION_GRANTED || ActivityCompat.checkSelfPermission (
        this, Manifest.permission.ACCESS_COARSE_LOCATION ) ==
        PackageManager.PERMISSION_GRANTED) {
            mMap.setMyLocationEnabled ( true );
        }
    }
}

```

Plik AndroidManifest.xml - ikona aplikacji:

```

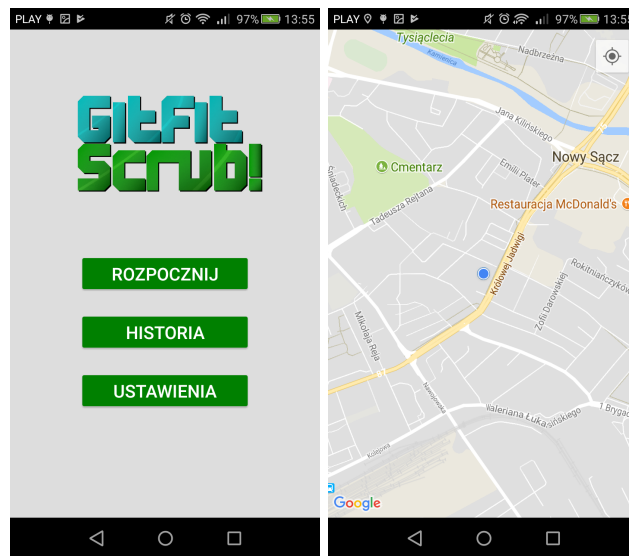
android:allowBackup="true"
android:icon="@mipmap/ic_launcher"
android:label="@string/app_name"
android:roundIcon="@mipmap/ic_launcher_round"
android:supportRtl="true"
android:theme="@android:style/Theme.Black.NoTitleBar.Fullscreen">

```

## 5.4 Sprint Review/Demo

Podczas przebiegu sprintu wykonane zostały wszystkie zaplanowane zadania oraz pokryły one estymaty.

Demonstracja przyrostu produktu:



Ekran menu głównego (po lewej) oraz widok mapy, po wciśnięciu przycisku "Rozpocznij", z oznaczoną pozycją urządzenia.

## 6 Sprint 2

### 6.1 Cel

Pobranie i przetworzenie danych na podstawie geolokalizacji użytkownika oraz utworzenie przycisku wysuwającego panel wyświetlający wybrane informacje (aktualną prędkość, czas i dystans).

### 6.2 Sprint Planning/Backlog

**Tytuł zadania.** Stworzenie alertu o braku aktywacji modułu GPS.

- Estymata: M.

**Tytuł zadania.** Pobranie informacji z modułu GPS.

- Estymata: L.

**Tytuł zadania.** Stworzenie przycisku "Rozpocznij nowy trening" oraz nadanie mu funkcjonalności.

- Estymata: L.

**Tytuł zadania.** Przetworzenie zgromadzonych informacji.

- Estymata: XL.

### 6.3 Realizacja

**Tytuł zadania.** Stworzenie alertu o braku aktywacji modułu GPS.

**Wykonawca.** Katarzyna Poręba.

**Realizacja.** Utworzenie alertu, powiadamiającego użytkownika o wyłączonym GPS-ie, przy pomocy biblioteki `android.support.v7.app.AlertDialog`.  
Utworzenie funkcji wyświetlającej alert:

```
private void buildAlertMessageNoGps() {
    final AlertDialog.Builder builder = new AlertDialog.
        Builder(this);
    builder.setMessage("Twój GPS jest wyłączony. Czy chcesz
        go włączyć?")
        .setCancelable(false)
        .setPositiveButton("Tak", new DialogInterface.
            OnClickListener() {
                public void onClick(final DialogInterface
                    dialog, @SuppressWarnings("unused") final int id) {
                    startActivity(new Intent(android.provider
                        .Settings.ACTION_LOCATION_SOURCE_SETTINGS));
                }
            })
        .setNegativeButton("Nie", new DialogInterface.
            OnClickListener() {
                public void onClick(final DialogInterface
                    dialog, @SuppressWarnings("unused") final int id) {
                    dialog.cancel();
                }
            });
    final AlertDialog alert = builder.create();
    alert.show();
}
```

Utworzenie warunku sprawdzającego, czy GPS jest wyłączony:

```
if (!locationManager.isProviderEnabled(LocationManager.
    GPS_PROVIDER)) {buildAlertMessageNoGps();}
```

**Tytuł zadania.** Pobranie informacji z modułu GPS.

**Wykonawca.** Michał Ptak.

**Realizacja.** Utworzenie metod nasłuchujących zmian w informacjach dostarczanych przez moduł GPS w oparciu o Google Play Services.  
Plik `MapsActivity.java`:

```
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_maps);

    mFusedLocationClient = LocationServices.
        getFusedLocationProviderClient(this);
}
```

```

        mapFrag = (SupportMapFragment) getSupportFragmentManager
().findFragmentById(R.id.map);
        mapFrag.getMapAsync(this);

        (...)
    }

    @Override
    protected void onPause() {
        super.onPause();

        if (mFusedLocationClient != null) {
            mFusedLocationClient.removeLocationUpdates(
mLocationCallback);
        }
    }

    @Override
    public void onMapReady(GoogleMap googleMap)
    {
        mGoogleMap=googleMap;
        mGoogleMap.setMapType(GoogleMap.MAP_TYPE_NORMAL);

        if (android.os.Build.VERSION.SDK_INT >= Build.
VERSION_CODES.M) {
            if (ContextCompat.checkSelfPermission(this,
                Manifest.permission.ACCESS_FINE_LOCATION)
                == PackageManager.PERMISSION_GRANTED) {
                buildGoogleApiClient();
                mGoogleMap.setMyLocationEnabled(true);
            } else {
                checkLocationPermission();
            }
        }
        else {
            buildGoogleApiClient();
            mGoogleMap.setMyLocationEnabled(true);
        }
    }

    protected synchronized void buildGoogleApiClient() {
        mGoogleApiClient = new GoogleApiClient.Builder(this)
            .addConnectionCallbacks(this)
            .addOnConnectionFailedListener(this)
            .addApi(LocationServices.API)
            .build();
        mGoogleApiClient.connect();
    }
}

```

```

@Override
public void onConnected(Bundle bundle) {
    mLocationRequest = new LocationRequest();
    mLocationRequest.setInterval(1000);
    mLocationRequest.setFastestInterval(100);
    mLocationRequest.setPriority(LocationRequest.
PRIORITY_BALANCED_POWER_ACCURACY);
    if (ContextCompat.checkSelfPermission(this,
        Manifest.permission.ACCESS_FINE_LOCATION)
        == PackageManager.PERMISSION_GRANTED) {
        mFusedLocationClient.requestLocationUpdates(
mLocationRequest, mLocationCallback, Looper.myLooper());
    }
}

LocationCallback mLocationCallback = new LocationCallback(){
    @Override
    public void onLocationResult(LocationResult
locationResult) {
        for (Location location : locationResult.getLocations
()) {
            Log.i("MapsActivity", "Location: " + location.
getLatitude() + " " + location.getLongitude());
            mLastLocation = location;
            if (mCurrLocationMarker !=null) {
                mCurrLocationMarker.remove();
            }

            LatLng latLng = new LatLng(location.getLatitude()
, location.getLongitude());
            MarkerOptions markerOptions = new MarkerOptions()
;
            markerOptions.position(latLng);
            markerOptions.title("Tu jestem!");
            markerOptions.icon(BitmapDescriptorFactory.
defaultMarker(BitmapDescriptorFactory.HUE_MAGENTA));
            mCurrLocationMarker = mGoogleMap.addMarker(
markerOptions);

            mGoogleMap.moveCamera(CameraUpdateFactory.
newLatLngZoom(latLng, 18));
        }
    }
};

@Override
public void onConnectionSuspended(int i) {}

```

```
@Override
public void onConnectionFailed(ConnectionResult
connectionResult) {}
```

**Tytuł zadania.** Stworzenie przycisku "Rozpocznij nowy trening" oraz nadanie mu funkcjonalności.

**Wykonawca.** Katarzyna Poręba, Jan Michalik.

**Realizacja.** Początkowym założeniem było, że po naciśnięciu przycisku panel będzie się otwierał i będą rozpoczęte pomiary. Panel został utworzony, jednak nie jest otwierany przez przycisk. Został on umieszczony bezpośrednio w aktywności mapy. Nazwę przycisku zmieniono na "Start", rozpoczyna on pomiary.

Plik activity\_maps.xml:

```
<?xml version="1.0" encoding="utf-8"?>

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/
android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical" >

    <fragment xmlns:android="http://schemas.android.com/apk/res/
android"
        xmlns:map="http://schemas.android.com/apk/res-auto"
        xmlns:tools="http://schemas.android.com/tools"
        android:id="@+id/map"
        android:name="com.google.android.gms.maps.
SupportMapFragment"
        android:layout_width="match_parent"
        android:layout_height="450dp"
        tools:context="pl.ppm.gitfitscrub.MapsActivity" />

    <Button
        android:id="@+id/start_button"
        style="@style/Widget.AppCompat.Button.Colored"
        android:layout_width="match_parent"
        android:layout_height="30dp"
        android:background="@color/colorAccent"
        android:padding="5dp"
        android:text="start"
        android:textColor="#ffffff" />

    <Chronometer
        android:id="@+id/chronometer"
```

```

        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:textColor="#000000"
        android:textSize="20dp" />

<TextView
    android:id="@+id/speedText"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:gravity="center"
    android:text="Predkosc: "
    android:textColor="#000000"
    android:textSize="20dp"/>

<TextView
    android:id="@+id/distanceText"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:gravity="center"
    android:text="Dystans: "
    android:textColor="#000000"
    android:textSize="20dp"/>

</LinearLayout>

Plik MapsActivity.java:

@Override
protected void onCreate(Bundle savedInstanceState) {
    (...)
    mStartButton = (Button) findViewById(R.id.start_button);
    mChronometer = (Chronometer) findViewById(R.id.
chronometer);

    distanceText = (TextView) findViewById(R.id.distanceText)
;
    speedText = (TextView) findViewById(R.id.speedText);

    mStartButton.setOnClickListener(new View.OnClickListener
() {
        @Override
        public void onClick(View view) {
            mChronometer.setBase(SystemClock.elapsedRealtime
());
            mChronometer.start();
        }
    });
}

```



```

LocationCallback mLocationCallback = new LocationCallback(){
    @Override
    public void onLocationResult(LocationResult
locationResult) {
        for (Location location : locationResult.getLocations
()) {
            (...)
            updateUI();
        }
    }
};

private void updateUI() {
    distance = distance + (lStart.distanceTo(lEnd) / 1000.00)
;
    if (speed > 0.0)
        speedText.setText("predkosc: " + new DecimalFormat("#.##").format(speed) + " km/h");
    else
        speedText.setText("obliczam predkosc...");

    distanceText.setText("dystans: " + new DecimalFormat("#.###").format(distance) + " km");

    lStart = lEnd;
}

```

**Tytuł zadania.** Przetworzenie zgromadzonych informacji.

**Wykonawca.** Michał Ptak.

**Realizacja.** Dystans w kilometrach liczony jest poprzez dodawanie długości kolejnych pokonywanych odcinków (droga pokonana pomiędzy dwoma aktualizacjami lokalizacji) do dystansu całkowitego, przy pomocy metody `distanceTo()`. Do pobierania prędkości wykorzystana została metoda `getSpeed()`, zwracająca prędkość w m/s, która została przeliczona na km/h.

Plik `MapsActivity.java`:

```

@Override
protected void onPause() {
    (...)
    distance = 0;
}

LocationCallback mLocationCallback = new LocationCallback(){
    @Override
    public void onLocationResult(LocationResult
locationResult) {

```

```

        for (Location location : locationResult.getLocations
    ()) {
        (...)
        if (lStart == null) {
            lStart = mLastLocation;
            lEnd = mLastLocation;
        } else {
            lEnd = mLastLocation;
        }

        speed = location.getSpeed() * 3.6;

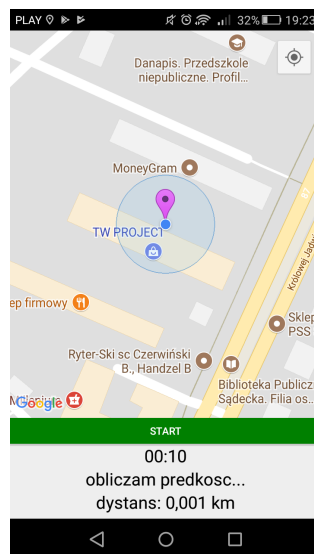
    }
}
};

```

## 6.4 Sprint Review/Demo

Założony cel sprintu został osiągnięty. Estymowana czasochłonność dla zadań 2-4 była błędna - zadania 2 i 3 wymagały większego nakładu pracy niż założono, natomiast zadanie ostatnie - przeciwnie.

Demonstracja przyrostu produktu:



Widok mapy oraz panelu ze statystykami po wciśnięciu przycisku "START", z oznaczoną znacznikiem pozycją urządzenia.

## 7 Sprint 3

### 7.1 Cel

Uaktualnienie panelu statystyk, rysowanie śladu na mapie oraz utworzenie aktywności "Ustawienia".

## 7.2 Sprint Planning/Backlog

**Tytuł zadania.** Modyfikacja panelu zawierającego statystyki aktualnego treningu.

- Estymata: XL

**Tytuł zadania.** Rysowanie polilinii na mapie.

- Estymata: XL

**Tytuł zadania.** Stworzenie przycisku "Ustawienia" w menu głównym oraz nadanie mu funkcjonalności.

- Estymata: M

«Tutaj dodawać kolejne zadania»

## 7.3 Realizacja

**Tytuł zadania.** «Tytuł».

**Wykonawca.** «Wykonawca».

**Realizacja.** «Sprawozdanie z realizacji zadania (w tym ocena zgodności z estymatą). Kod programu (środowisko *verbatim*):

```
for (i=1; i<10; i++)  
...  
».
```

**Tytuł zadania.** Rysowanie polilinii na mapie.

**Wykonawca.** Katarzyna Poręba

**Realizacja.** Utworzenie polilinii przy użyciu metody mapy google *addPolyline*. Ustalenie koloru i szerokości linii. Przekazanie zmiennych zawierających współrzędne geograficzne do funkcji *add* rysującej polilinie.

```
Polyline line = mMap.addPolyline(new PolylineOptions()  
    .add(new LatLng(lStart.getLatitude(), lStart.getLongitude()),  
        new LatLng(lEnd.getLatitude(), lEnd.getLongitude()))  
    .width(8)  
    .color(Color.GREEN));
```

**Tytuł zadania.** «Tytuł».

**Wykonawca.** «Wykonawca».

**Realizacja.** «Sprawozdanie z realizacji zadania (w tym ocena zgodności z estymatą). Kod programu (środowisko *verbatim*):

```
for (i=1; i<10; i++)  
...  
».
```

«Tutaj dodawać kolejne zadania»

## 7.4 Sprint Review/Demo

«Sprawozdanie z przeglądu Sprint'u – czy założony cel (przyrost) został osiągnięty oraz czy wszystkie zaplanowane Backlog Item'y zostały zrealizowane? Demonstracja przyrostu produktu».

# 8 Sprint 4

## 8.1 Cel

«Określić, w jakim celu tworzony jest przyrost produktu».

## 8.2 Sprint Planning/Backlog

**Tytuł zadania.** «Tytuł».

- Estymata: «szacowana czasochłonność (w „koszulkach”)».

**Tytuł zadania.** «Tytuł».

- Estymata: «szacowana czasochłonność (w „koszulkach”)».

«Tutaj dodawać kolejne zadania»

## 8.3 Realizacja

**Tytuł zadania.** «Tytuł».

**Wykonawca.** «Wykonawca».

**Realizacja.** «Sprawozdanie z realizacji zadania (w tym ocena zgodności z estymatą). Kod programu (środowisko *verbatim*):

```
for (i=1; i<10; i++)  
...  
».
```

**Tytuł zadania.** «Tytuł».

**Wykonawca.** «Wykonawca».

**Realizacja.** «Sprawozdanie z realizacji zadania (w tym ocena zgodności z estymatą). Kod programu (środowisko *verbatim*):

```
for (i=1; i<10; i++)  
...  
».
```

«Tutaj dodawać kolejne zadania»

## 8.4 Sprint Review/Demo

«Sprawozdanie z przeglądu Sprint'u – czy założony cel (przyrost) został osiągnięty oraz czy wszystkie zaplanowane Backlog Item'y zostały zrealizowane? Demonstracja przyrostu produktu».

# 9 Sprint 5

## 9.1 Cel

«Określić, w jakim celu tworzony jest przyrost produktu».

## 9.2 Sprint Planning/Backlog

**Tytuł zadania.** «Tytuł».

- Estymata: «szacowana czasochłonność (w „koszulkach”)».

**Tytuł zadania.** «Tytuł».

- Estymata: «szacowana czasochłonność (w „koszulkach”)».

«Tutaj dodawać kolejne zadania»

## 9.3 Realizacja

**Tytuł zadania.** «Tytuł».

**Wykonawca.** «Wykonawca».

**Realizacja.** «Sprawozdanie z realizacji zadania (w tym ocena zgodności z estymatą). Kod programu (środowisko *verbatim*):

```
for (i=1; i<10; i++)  
...  
».
```

**Tytuł zadania.** «Tytuł».

**Wykonawca.** «Wykonawca».

**Realizacja.** «Sprawozdanie z realizacji zadania (w tym ocena zgodności z estymatą). Kod programu (środowisko `verbatim`):

```
for (i=1; i<10; i++)  
...  
».
```

«Tutaj dodawać kolejne zadania»

## 9.4 Sprint Review/Demo

«Sprawozdanie z przeglądu Sprint'u – czy założony cel (przyrost) został osiągnięty oraz czy wszystkie zaplanowane Backlog Item'y zostały zrealizowane? Demostracja przyrostu produktu».

«Tutaj dodawać kolejne Sprint'y»

## Literatura

- [1] S. R. Covey, *7 nawyków skutecznego działania*, Rebis, Poznań, 2007.
- [2] Tobias Oetiker i wsp., Nie za krótkie wprowadzenie do systemu L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>, <ftp://ftp.gust.org.pl/TeX/info/lshort/polish/lshort2e.pdf>
- [3] K. Schwaber, J. Sutherland, *Scrum Guide*, <http://www.scrumguides.org/>, 2016.
- [4] <https://agilepainrelief.com/notesfromtooluser/tag/scrum-by-example>
- [5] [https://www.tutorialspoint.com/scrum/scrum\\_user\\_stories.htm](https://www.tutorialspoint.com/scrum/scrum_user_stories.htm)
- [6] <https://developer.android.com/index.html>
- [7] <https://docs.oracle.com/javase/8/docs/api/>
- [8] <https://stackoverflow.com/questions/tagged/android>
- [9] Dawn Griffiths, David Griffiths, *Android. Programowanie aplikacji. Rusz głową!*, Helion, 2016.