

# **System mobilny do wykrywania arytmii serca z wykorzystaniem modeli ML**

*Michał Naklicki, Marcin Sztukowski*

**Data wykonania:** 04.06.2025

---

## **1. Opis projektu**

### **Założenia**

Projekt zakłada stworzenie zintegrowanego systemu do wczesnego wykrywania arytmii serca w warunkach domowych. Głównym komponentem systemu jest czujnik EKG połączony z mikrokontrolerem Arduino, który zbiera dane elektrokardiograficzne i przesyła je bezprzewodowo do aplikacji mobilnej, która wykrywa nieprawidłowości w rytmie serca.

### **Cele projektowe**

- Zbudowanie niedrogiego, mobilnego systemu do monitorowania pracy serca.
- Wykorzystanie uczenia maszynowego (ML) do automatycznej klasyfikacji rytmu serca.
- Zapewnienie łatwej obsługi przez użytkownika końcowego (pacjenta lub opiekuna).
- Umożliwienie szybkiej reakcji w przypadku wykrycia nieprawidłowości.

### **Funkcje**

- Rejestrowanie sygnału EKG przez sondę podłączoną do Arduino.
- Transmisja danych EKG w czasie rzeczywistym do aplikacji mobilnej (Android/iOS).
- Analiza sygnału za pomocą modelu TensorFlow Lite/Core ML w aplikacji mobilnej.
- Wizualizacja sygnału EKG na ekranie aplikacji.
- Powiadamianie użytkownika w przypadku wykrycia potencjalnej arytmii.
- Zapis oraz odczyt sygnałów EKG.

### **Przewidywane przeznaczenie**

Projekt jest przeznaczony do zastosowania w diagnostyce domowej, telemedycynie i wsparciu opieki kardiologicznej. Może służyć zarówno pacjentom z historią chorób serca, jak i osobom zdrowym chcącym monitorować swój stan zdrowia.

---

## 2. Wybór technologii informatycznych

Technologia	Uzasadnienie
<b>Arduino + czujnik EKG</b>	Niski koszt, szeroka dokumentacja, łatwość integracji z czujnikami biomedycznymi.
<b>Bluetooth (moduł BLE)</b>	Pozwala na bezprzewodową transmisję danych do urządzeń mobilnych.
<b>PyTorch</b>	Powszechnie stosowany do trenowania i testowania modeli ML wykorzystywanych do różnych zastosowań, w tym predykcji.
<b>TensorFlow Lite</b>	Umożliwia uruchamianie modeli ML na urządzeniach mobilnych, zapewniając niskie opóźnienia i małe zużycie zasobów.

---

## 3. Projekt architektury aplikacji

### Komponenty systemu:

- Warstwa sprzętowa (Hardware Layer)**
  - **Arduino** – odpowiada za akwizycję danych z sondy EKG i wstępne filtrowanie sygnału.
  - **Moduł Bluetooth** – przesyła dane do aplikacji mobilnej w czasie rzeczywistym.
- Warstwa komunikacyjna (Communication Layer)**
  - Komunikacja Bluetooth pomiędzy Arduino a urządzeniem mobilnym.
- Warstwa aplikacyjna (Application Layer)**
  - **Natywne aplikacje** – osobne aplikacje tworzone na platformy Android oraz iOS.
  - **Model TensorFlow Lite** – model działający na Android klasyfikujący rytm serca na podstawie sekwencji sygnału EKG. Model jest uprzednio wytrenowany offline przy użyciu PyTorch i konwertowany do formatu .tflite oraz zaimplementowany lokalnie w aplikacji.
  - **Model CoreML** – model działający na Android klasyfikujący rytm serca na podstawie sekwencji sygnału EKG. Model jest uprzednio wytrenowany offline przy użyciu PyTorch i konwertowany do formatu .mlmodel oraz zaimplementowany lokalnie w aplikacji.
- Warstwa analityczna (Analytics Layer)**
  - Wydzielenie cech z sygnału EKG (preprocessing). Wykrywany jest tzw. zespół QRS oraz wydzielane jest wokół niego okienko w formacie wymaganym przez model ML.
  - Wykrywanie arytmii (np. migotania przedsionków, tachykardii, bradykardii) przy pomocy sieci neuronowej. Model zwraca liczbę segmentów oraz jak zostały sklasyfikowane.

## Architektura aplikacji Android oraz iOS

Najważniejsze klasy w aplikacjach Android oraz iOS to:

---

### 1. BluetoothConnection / EKGBLEManager

- Skanowanie urządzeń Bluetooth.
  - Nawiązywanie połączenia z urządzeniem.
  - Subskrypcję danych z charakterystyki BLE.
- 

### 2. EcgPredictor / EKGClassification

Klasa do klasyfikacji sygnału EKG z użyciem modelu ML. Na podstawie kontekstu:

- Ładuje wybrany model ML.
  - Wykonuje predykcję rytmu lub typów zespołów QRS.
- 

### 3. Waveform

Klasa reprezentująca sygnał EKG. Wykonywane są na niej operacje:

- Próbkowanie sygnału.
  - Detekcja zespołów QRS.
  - Ekstrakcja okien czasowych dla klasyfikacji.
  - Eksportu do standardowego formatu.
- 

### 4. SignalReader

Klasa pozwalająca na ładowanie sygnału EKG ze standardowego formatu. Wykonywane są na niej operacje:

- Konwersja kodowania pliku.
- Import do klasy Waveform.