

# Raport końcowy

## Aplikacja diagnostyczna na urządzenia z systemem Android

### Autorzy:

Mateusz Zuchniak

Maciej Cichos

Łukasz Bartsch

### Podsumowanie:

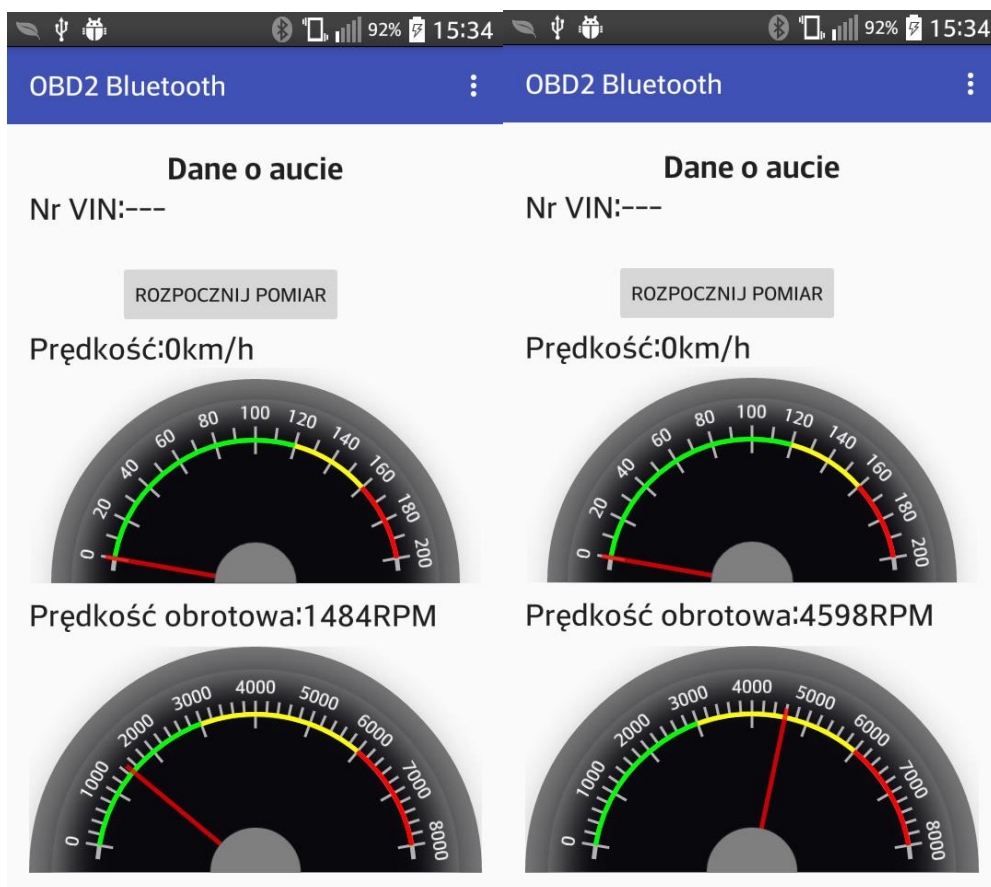
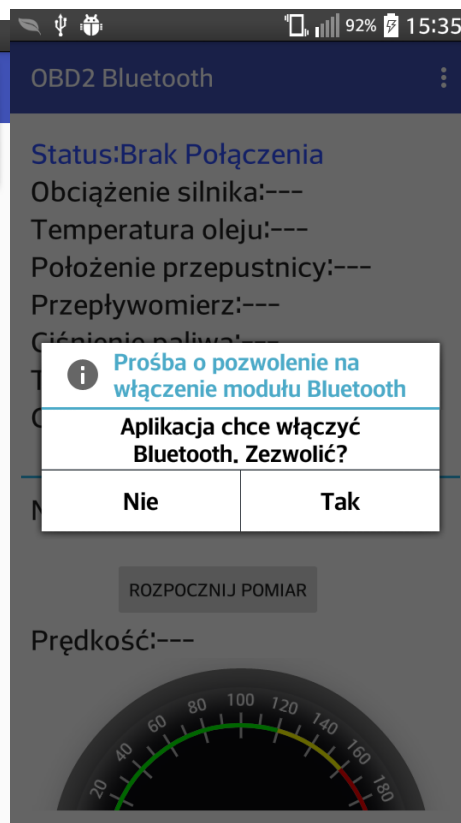
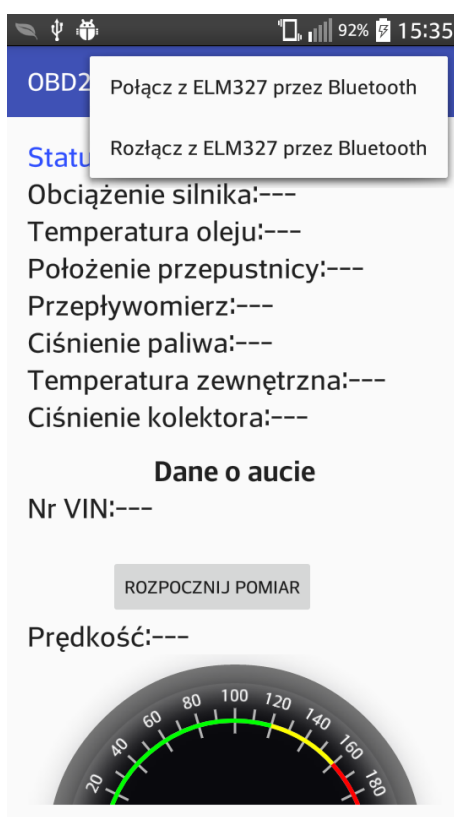
W ramach projektu została stworzona aplikacja, która umożliwia odczyt danych z komputera pokładowego i czujników z użyciem interfejsu OBD2 i wtyczki ELM327 Bluetooth.

Na początku projektu zostały wyznaczone następujące założenia projektowe:

#### Założenia projektowe:

- Aplikacja na urządzenia z systemem Android (tworzona w Android Studio)
- Odczyt parametrów silnika i danych z czujników poprzez interfejs ELM 327 Bluetooth
- Aplikacja utrzymuje stałe połączenie Bluetooth pomiędzy wtyczką ELM 327, a telefonem
- Odczyt danych, przesłanie ich drogą bezprzewodową i wyświetlenie na telefonie
- Wizualizacja otrzymanych danych (obrotomierze, wykresy, itd.)
- Implementacja dodatkowych bibliotek "OBD II Java", aby rozszerzyć możliwości diagnostyczne aplikacji
- Testowanie aplikacji w autach różnych producentów
- Jeśli starczy czasu to rozszerzenie aplikacji o czytanie kodów błędów i usuwanie ich

Działanie aplikacji zostało przedstawione na filmie dołączonym do raportu, a poniżej zostały pokazane przykładowe screeny przedstawiające jej funkcje. Cały kod aplikacji znajduje się w repozytorium projektu. W dalszej części raportu odniesiemy się do poszczególnych punktów założeń projektowych.



Wszystkie założone cele na początku projektu zostały osiągnięte. Poniżej dokładnie omówimy każdy z nich:

- Stworzona aplikacja poprawnie odczytuje dane z komputera pokładowego i czujników. Niestety, naszym głównym autem do testów był Fiat Seicento z 2003r., który nie posiadał dużej liczby czujników przez co w ostatecznej wersji aplikacji przedstawiane są tylko dane z dostępnych w tym aucie czujników. Próbowaliśmy testować aplikację na innych autach, do których mieliśmy dostęp (Mazda 626, Opel Astra F, Renault Laguna), lecz te modele nie wspierają interfejsu OBD2, tak aby odczytywać dane z komputera pokładowego. Metody do odczytu pozostałych parametrów znajdują się w kodzie aplikacji, więc nie stanowi problemu szybkie rozszerzenie jej możliwości odczytu dla nowszego auta z większą ilością czujników.
- W aplikacji zostało zapewnione stałe połączenie Bluetooth pomiędzy telefonem, a wtyczką ELM327. Na filmie dołączonym do raportu widać działanie aplikacji przez dłuższy czas i jak widać aplikacja bez problemu utrzymuje stałe połączenie i na bieżąco czytuje dane i wyświetla je w aplikacji. Co więcej, w przypadku gdy w telefonie jest wyłączony Bluetooth, użytkownik jest proszony o jego włączenie, co zostało pokazane na jednym ze screenów.
- Do aplikacji zostały zaimplementowane prędkościomierz i obrotomierz do graficznej reprezentacji otrzymanych wyników. Jak widać na filmie czy też na powyższych screenach odczytana prędkość i prędkość obrotowa zgadzają się z wartościami przedstawianymi na grafikach. Do wizualizacji danych została dołączona do projektu dodatkowa biblioteka „Android Widgets”. Dodatkowo chcieliśmy dołączyć wykresy do przedstawienia wybranych z pozostałych parametrów, lecz ze względu na problemy z małą ilością czujników w Seicento postanowiliśmy skupić się na obrotomierzu i prędkościomierzu, gdyż te parametry są odczytywane w każdym aucie.
- Do komunikacji z komputerem pokładowym została wykorzystana dodatkowa biblioteka „OBD II Java”. Posiada ona metody do odczytywania wybranych parametrów silnika. Jest ona na bieżąco rozwijana, więc w trakcie korzystania z niej pojawiało się sporo problemów, które ostatecznie udało się rozwiązać. W przypadku Fiata Seicento udało nam się odczytać następujące dane:
  - Prędkość obrotowa silnika,
  - Prędkość jazdy,
  - Obciążenie silnika,
  - Położenie przepustnicy,
  - Ciśnienie w kolektorze.
- Dodatkowym celem, który zakładaliśmy było dodanie funkcji odczytywania kodów błędów, lecz ze względu na problemy, które wystąpiły przy komunikacji i odczytywaniu parametrów nie starczyło nam czasu na implementację tego punktu. Jednakże, w przyszłości aplikacja może być dalej rozwijana o nowe funkcje, np. o odczyt kodów błędów.

## Wnioski:

Stworzona aplikacja posiada funkcje do odczytu podstawowych parametrów z auta. Jesteśmy zadowoleni z uzyskanych efektów końcowych, gdyż rozpoczynając projekt nie mieliśmy wcześniej do czynienia z programowaniem w Javie i tworzeniem aplikacji na Androida, a mimo to aplikacja działa bardzo dobrze.

W trakcie projektu największym problemem była komunikacja z komputerem pokładowym. Twórcy dodatkowych bibliotek nie udostępnili wszystkich potrzebnych kroków do poprawnej komunikacji przez co poświęciliśmy temu najwięcej czasu. Kolejnym z problemów była niemożność testowania aplikacji w domu, w większości przypadków musieliśmy się spotykać tak, aby przetestować

funkcje w połączeniu z autem. Jedynymi częściami aplikacji, które mogliśmy wykonać w domu był interfejs graficzny i połączenie Bluetooth.

Jak już wcześniej wspomnieliśmy nie mogliśmy odczytać więcej parametrów ze względu na dostępność tylko Fiata Seicento. Jednakże w kodzie są już gotowe metody do odczytu pozostałych parametrów. Umożliwia to dalsze rozwijanie aplikacji o dodatkowe możliwości.