***Intell Driver***

***Krzysztof Ćmiel***

**Dokumentacja projektu**

Spis treści

[Cel wraz z opisem projektu 2](#_Toc536194901)

[Wykorzystane technologie 3](#_Toc536194902)

[Struktura aplikacji 4](#_Toc536194903)

[Interfejs 5](#_Toc536194904)

## Cel wraz z opisem projektu

Celem projektu było stworzenie mobilnej aplikacji, która będzie

w stanie wesprzeć właściciela auta w kontrolowaniu wydatków związanych z paliwem.

Aplikacja została napisana na systemy Android w języku Java oraz funkcjonuje dzięki bazie danych SQLite.

**Oferuje:**

* **Dodawanie każdego tankowania do pamięci aplikacji** 
  + Automatycznie uzupełniana data
  + Aktualny przebieg auta(dla ułatwienia dodana podpowiedź poprzedniego stanu)
  + Koszt tankowania
  + Ilość litrów paliwa
  + Pytanie czy tankowanie było "do pełna"
* **Wyświetlanie historii tankowania**
  + Numer identyfikacyjny
  + Data
  + Przebieg
  + Obliczona cena za litr paliwa
  + Koszt tankowania
  + Ilość litrów paliwa
  + Obliczone średnie za dany okres (jeśli dane tankowanie wraz

z poprzedzającym było do pełna)

* + Informacja "do pełna" (jeśli prawdziwa)
* **Usuwanie tankowania z historii**
  + Wskazanie ID tankowania, które zdecydujemy się usunąć
* **Statystyki**
  + Łączna kwota wydana na paliwo biorąc pod uwagę wszystkie tankowania w historii
  + Z całej historii:
    - Średnie spalanie
    - Średni koszt stu kilometrów
  + Z ostatniego tankowania:
    - Średnie spalanie (aktualne)
    - Aktualny koszt stu kilometrów

W przyszłości przewiduję rozwinięcie aplikacji, ponieważ możliwości są ogromne, a uzyskane informacje bardzo przydatne przy budżecie auta, opłacalności czy też świadomości kosztów użytkowania.

## Wykorzystane technologie

* **Android studio**

Aplikacja była w całości napisana w powyższemu środowisku

w wersji 3.2.1.

* **Java**
* **SQLite**

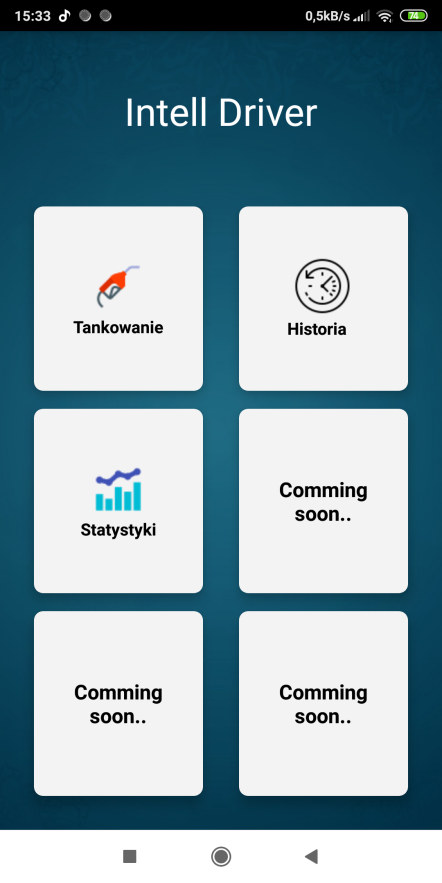
Baza danych aplikacji została zaimplementowana w systemie SQLite.

## Struktura aplikacji

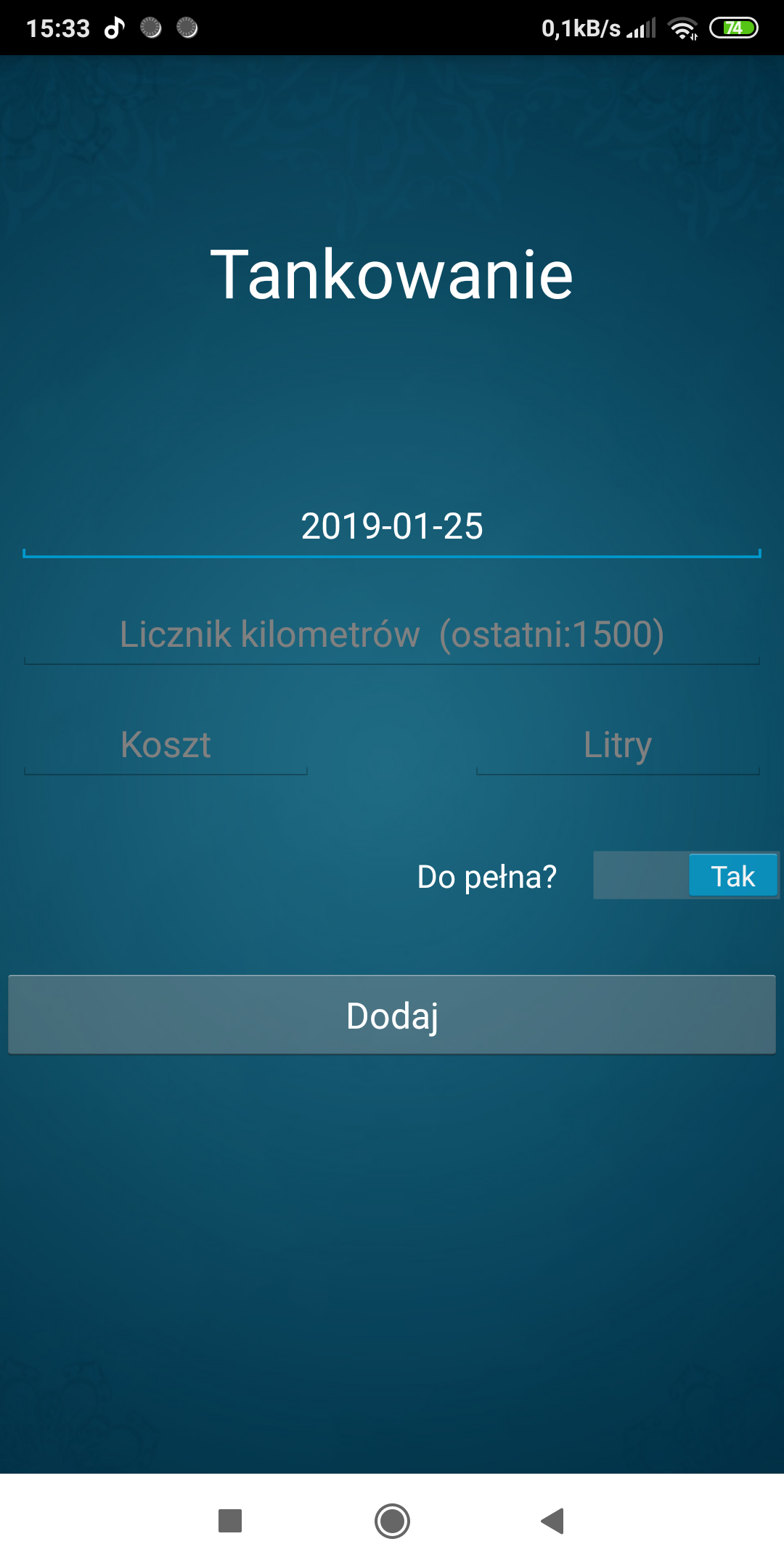
* **Klasy:**
  + DatabaseHelper
    - Klasa obsługująca połączenie z bazą danych. Zawiera metody niezbędne do obsługi funkcjonalności aplikacji.
  + Klasy odpowiedzialne za obsługę poszczególnych widoków:
    - Menu
    - Tankowanie
    - Historia
    - Statystyki
* **Metody:**
  + wstawDane(data,licznik,cena,koszt,litry,doPelna, spalanie,doPelna1)
  + usunDane(id)
  + wyswietlDane()
  + ostatniLicznik()
  + srednieSpalanie()
  + aktualneSpalanie()
  + sredniKosztZa100()
  + aktualnyKosztZa100()
  + spalanie(licznik,litry,doPelna)
  + wyswietlKoszt()

## Interfejs

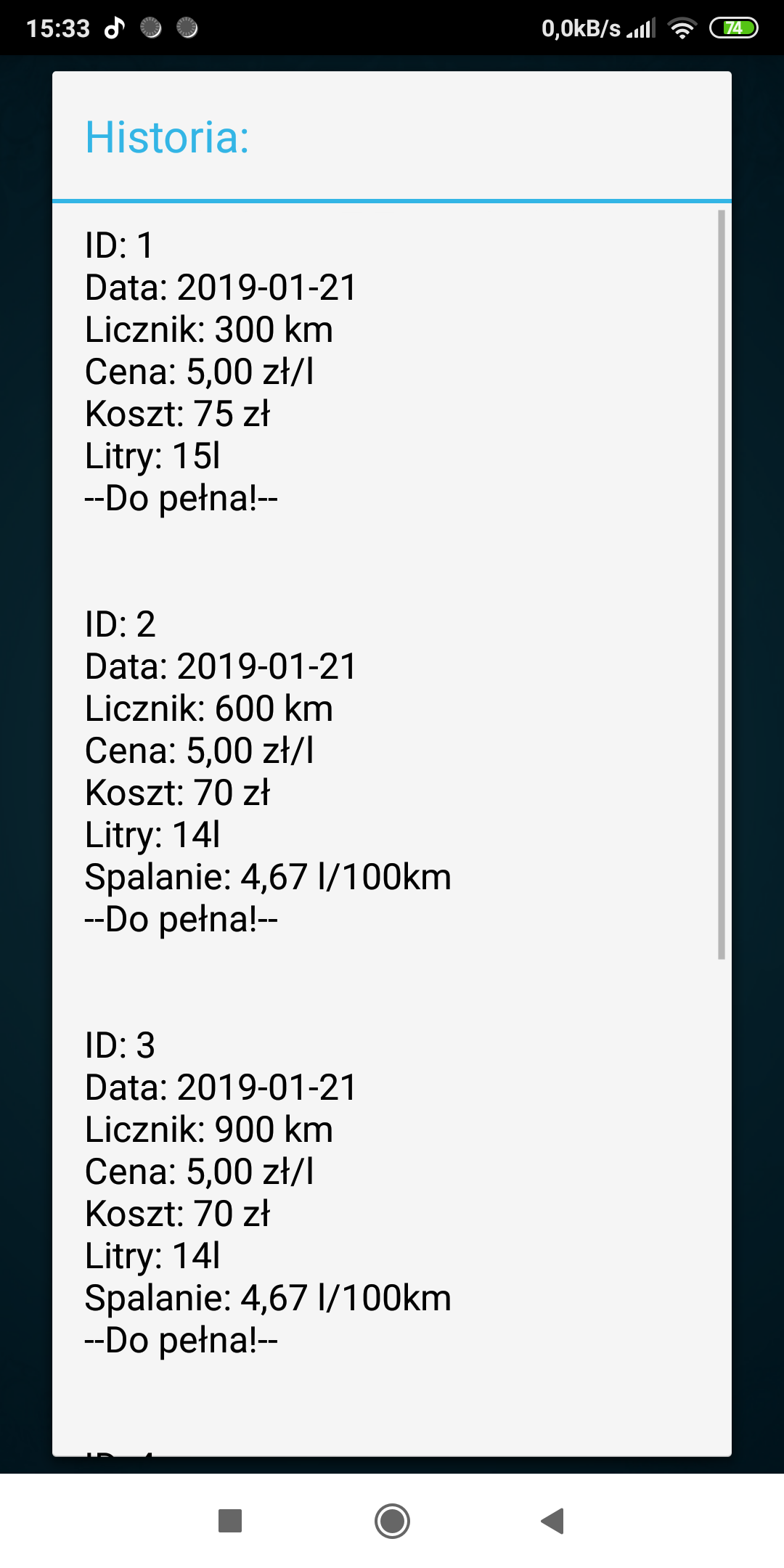
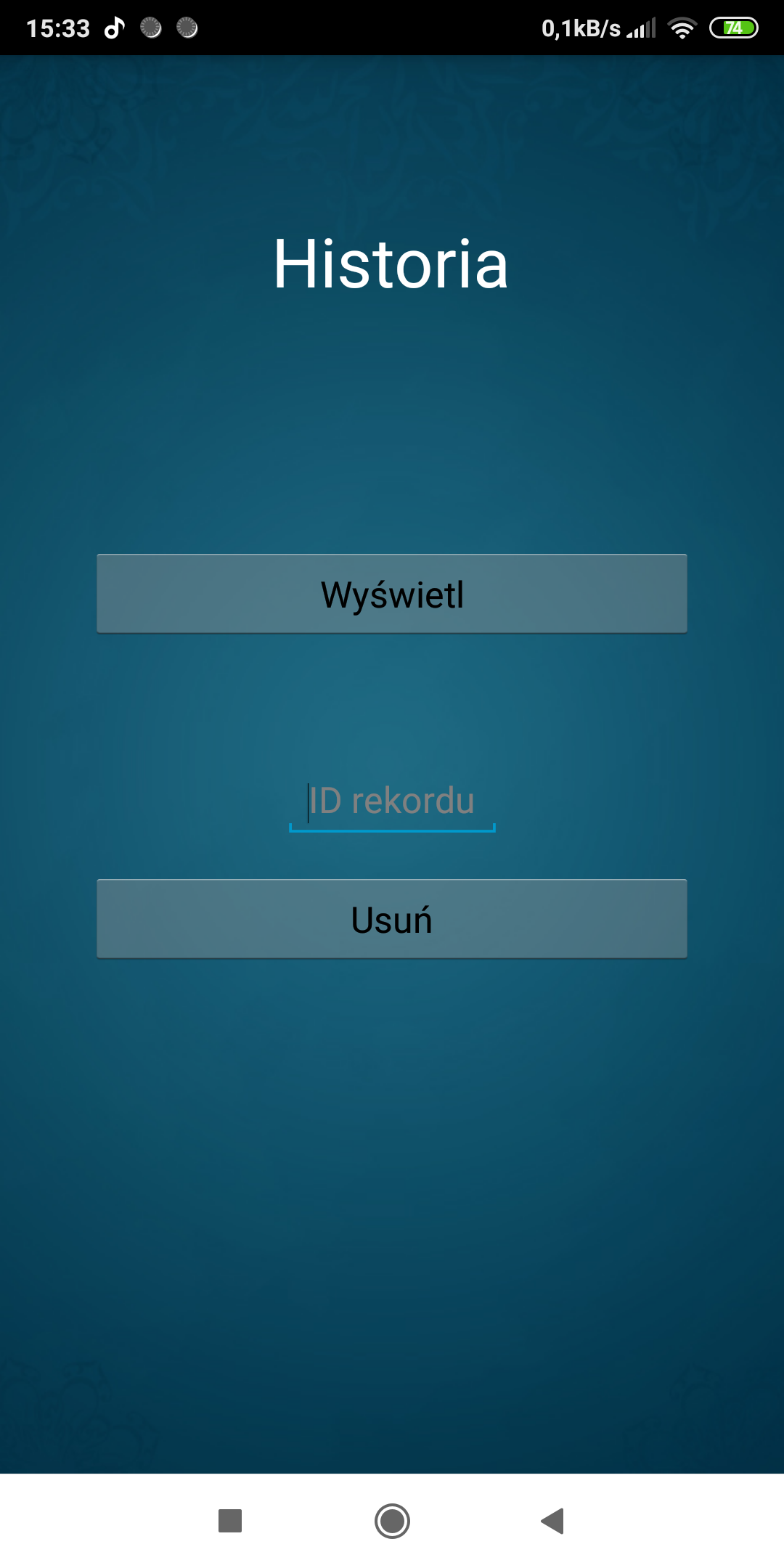
* **Menu główne**



* **Tankowanie**



* **Historia**



* **Statystyki**

