



## REFERAT PRACY DYPLOMOWEJ

**Temat pracy:** *Projekt i implementacja systemu prezentacji strumieni danych gromadzonych przez sensory urządzeń mobilnych*

**Autor:** Piotr Bielański

**Promotor:** dr inż. Roman Simiński

Kategoria: aplikacje mobilne

Słowa kluczowe: wykresy danych

### 1. Cel i podstawowe założenia

Sensory są „oczami i uszami” maszyny. Ogromna liczba typów czujników wykorzystywanych we współczesnej automatyce przekłada się na bogactwo różnorodności ich zastosowań. Niezależnie jednak od rodzaju sensora, językiem jego interakcji z otoczeniem są wartości liczbowe. Ostatecznie to cyfry opisują zmiany stanu środowiska, na które dany typ czujnika jest wrażliwy. O ile dla maszyny język matematyki jest naturalnym, dla człowieka pracującego z maszyną to tylko jedna z możliwości, zwykle nie mieszcząca się w kategoriach „pierwszego wyboru”. Postulowane w niniejszym opracowaniu rozwiązanie ma na celu ułatwienie komunikacji człowieka z maszyną.

Celem pracy było zaprojektowanie i realizacja systemu prezentacji strumieni danych, gromadzonych przez zestaw sensorów zintegrowanych z urządzeniem przenośnym. Dane przekazywane do systemu wyrażone są wartościami liczbowymi. Projektowany system ma za zadanie ułatwić ich dalszą analizę, dzięki wizualizacji graficznej w formie wykresów. System ma charakter ściśle narzędziowy i jako taki może stanowić samodzielny moduł diagnostyczny, bądź też

fragment większej architektury pomiarowo-analitycznej w zastosowaniach przemysłowych.

W ramach systemu zaplanowano funkcjonowanie dwóch aplikacji, spełniających na jego rzecz odrębne zadania. Pierwsza, nazywana w dalszej części opracowania „Data Provider”, pełni rolę emitera danych, które za pośrednictwem serwera w chmurze trafiają do odbiorcy. Działanie serwera przyjmującego i wysyłającego strumienie danych zostało zasymulowane przy pomocy usługi bazy danych czasu rzeczywistego Firebase. Zadaniem drugiej aplikacji, w dalszej części nazywanej „Data Stream Chart”, jest odczytanie wartości strumienia z sieci Internet i zaprezentowanie ich użytkownikowi systemu. Prezentacja odbywa się w formie wykresów, których rodzaj (liniowy lub słupkowy) jest zależny od wyboru użytkownika. Przewidziano ponadto stworzenie użytkownikowi możliwości samodzielnego określenia „okna czasowego” do obserwacji zmian wybranych parametrów. Aplikacja pozwala użytkownikowi na wybór strumienia danych do obserwacji, jeśli dostępnych jest więcej potencjalnych źródeł danych (czyt. emiter posiada czujniki różnych typów i wskazania każdego z nich są istotne dla użytkownika systemu). Przewidziano również możliwość połączenia dwóch przykładowych strumieni oraz wspólną prezentację ich wartości na jednym wykresie w celach porównawczych.

Proponowany system działa w oparciu o OS Android i w pełni kompatybilną z nim usługę Firebase. Poza tworzeniem warstwy pośredniczącej w wymianie danych między aplikacjami, serwer Firebase jest również wykorzystany w celu autentykacji dostępu użytkowników do aplikacji. Działanie obsługiwanych w aplikacji „Data Stream Chart” termometru i higrometru zostało zasymulowane poprzez wygenerowanie pseudolosowych wartości temperatury i wilgotności powietrza, na potrzeby zapełnienia lokalnej bazy danych SQLite. Utworzenie i obsługa bazy danych na urządzeniu mobilnym została zaimplementowana w oparciu o bibliotekę *Room Persistence*, a wszelkie odniesienia do bazy na potrzeby prezentacji danych są realizowane z wykorzystaniem *RxJava2* i *RxAndroid2*. Obsługa wyświetlania wykresów przebiega przy użyciu biblioteki *MPAndroidChart*.

## **2. Realizacja projektu**

Realizacja projektu przebiegała etapowo. W toku pracy nad systemem przeprowadzono wnikliwą analizę dostępności podobnych rozwiązań. Wnioskiem ogólnym z tego etapu pracy jest spostrzeżenie, że każdy z systemów proponowanych w chwili obecnej w popularnej usłudze „Play Store”, ogranicza się do lokalnej kontroli

nad urządzeniem, na którym jest zainstalowany. Na ich tle rozwiązanie zaproponowane w pracy, prezentuje się jako model niedużego systemu rozproszonego i jako taki rozszerza gamę możliwych zastosowań.

W kolejnych etapach pracy nad systemem przedstawiono koncepcję rozwiązania technologicznego oraz prezentację metod i narzędzi użytych do realizacji celu. Zaprojektowano widoki graficznego interfejsu użytkownika dla każdej aktywności aplikacji składających się na system. Etap pisania kodu zakończono szeregiem testów manualnych, sprawdzających zarówno zgodność wartości przekazywanych do obiektów w programie z wprowadzanymi przez użytkownika, jak również odpowiedzi poszczególnych elementów interfejsu użytkownika w odniesieniu do oczekiwanych akcji wyzwalanych w systemie. Dodatkowo przeprowadzono sprawdzenie poziomu zapotrzebowania na zasoby CPU, RAM, dostępu do sieci i energii podczas pracy systemu. Do tego celu wykorzystano narzędzie *Android Profiler*.

### **3. Produkt końcowy**

#### **3.1 Podstawowe wymagania aplikacji**

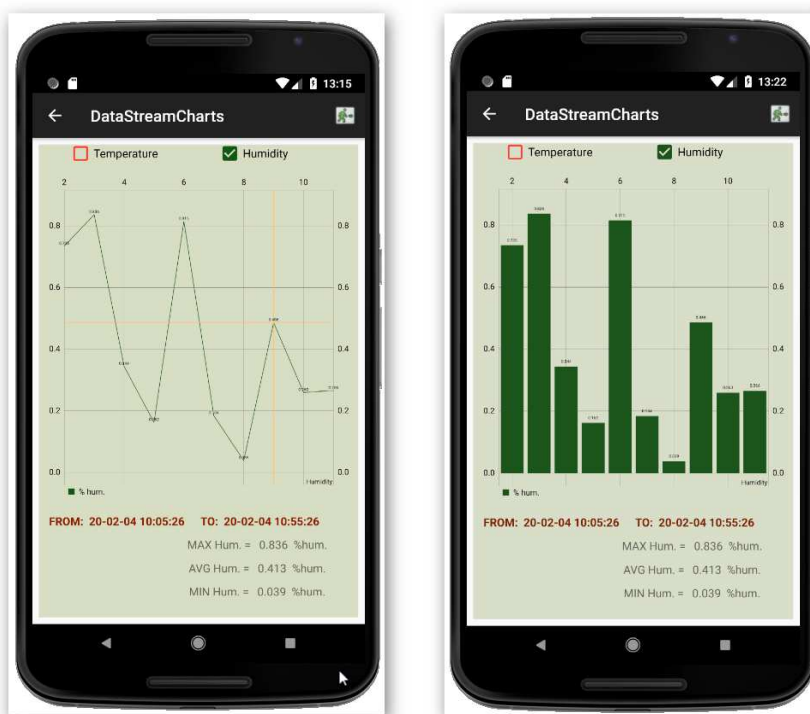
Pośród wymagań niefunkcjonalnych, które muszą zostać spełnione jako warunek ergonomicznej i efektywnej pracy systemu wymienić należy:

1. Dedykowanym środowiskiem pracy „Data Stream Charts” jest OS Android w wersji 6.0 (Marshmallow) z min. API 23, zapewniający obsługę urządzenia mobilnego, na którym aplikacje systemu będą zainstalowane;
2. Urządzenie mobilne powinno posiadać ekran dotykowy o przekątnej min. 7 “ lub więcej;
3. Pamięć RAM urządzenia min. 2GB;
4. Urządzenia mobilne obsługujące aplikacje „Data Stream Chart” i „Data Provider” muszą posiadać dostęp do sieci Internet;
5. Dostęp do serwera Firebase realizowany jest za pośrednictwem przeglądarki internetowej;

#### **3.2 Funkcjonalność oprogramowania**

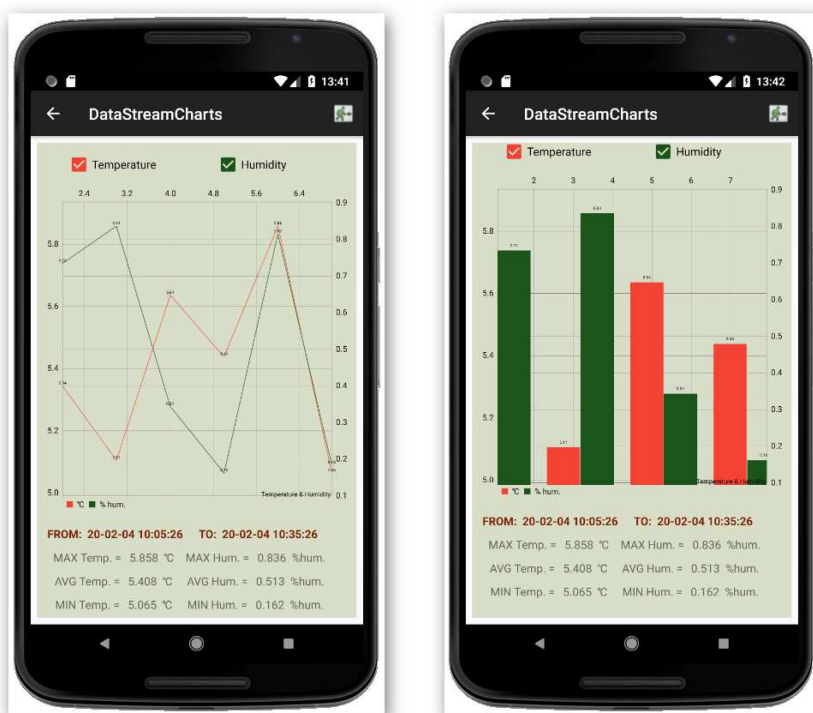
System posiada budowę modułową. Poszczególne jego części udostępniają następujące funkcje:

- moduł logowania – rejestracja użytkownika przy pomocy adresu email i hasła, weryfikacja adresu email, resetowanie/zmiana hasła konta użytkownika;
- moduł prezentacji danych wprowadzanych „ad hoc” – dane wprowadzane na ekranie zamieniane są na rysunek wykresu i jednocześnie zapisywane są na serwerze do dalszej dyspozycji administratora serwerowej bazy danych (archiwizacja, eksport, itp.);
- moduł prezentacji danych pozyskanych z Firebase – dane umieszczone w bazie danych czasu rzeczywistego przez administratora serwera (import pliku JSON) lub napływające z innych aplikacji typu „Data Provider” służą do rysowania wykresu synchronicznie w miarę pojawiania się nowych wartości;
- moduł prezentacji danych pozyskanych z lokalnej bazy SQLite – określenie daty i czasu początku i końca przedziału danych do obserwacji pozwala na wykreślanie grafik w formie liniowej lub słupkowej, porównanie odczytów dwóch czujników na tym samym wykresie, zbiorcze przedstawienie wartości średnich, max. i min. w badanym przedziale w formie tabelki.



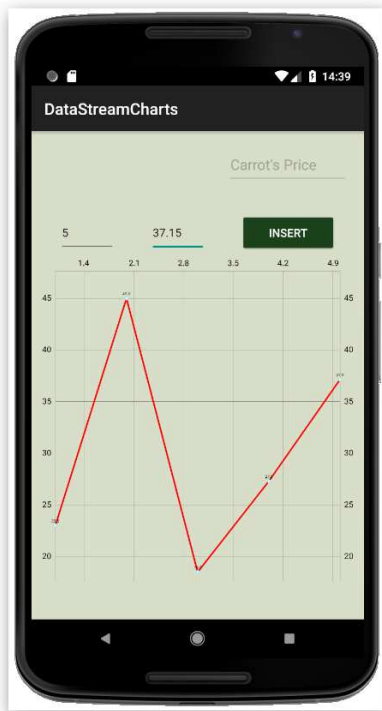
rys. 1 Widok wykresu liniowego i słupkowego strumienia danych higrometru .

Dla celów porównawczych lub dla zaobserwowania ewentualnych korelacji pomiędzy odczytami pierwszego i drugiego czujnika, praktycznym rozwiązaniem jest wyświetlenie obu strumieni na wspólnym wykresie. Pole wykresu będzie wówczas automatycznie skalowało się na potrzeby prezentacji wartości liczbowych opisywanych różnymi jednostkami, przyporządkowując każdemu strumieniowi jedną z osi Y (prawa/lewa), a tabele statystyk w dolnej części ekranu pokażą wtedy stosowne wartości dla każdego strumienia z osobna .



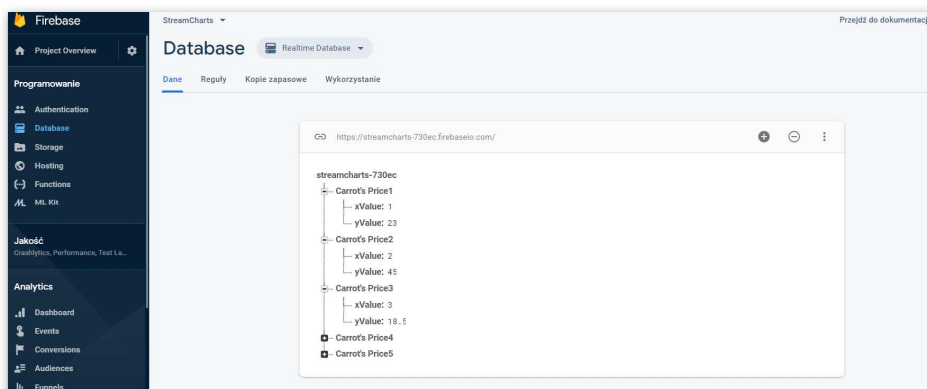
rys. 2 Widok wykresu liniowego i słupkowego na wspólnej prezentacji odczytów termometru i higrometru.

Użytkownik, chcący ujrzeć wykres wyrysowany na podstawie dowolnych danych, może wprowadzić je do aplikacji korzystając z przeznaczonego do tego celu modułu. Kolejne wpisy dokonywane przez użytkownika zaznaczone są na wykresie w postaci kółeczek symbolizujących punkt, a ich wartości liczbowe wyświetlane są powyżej .



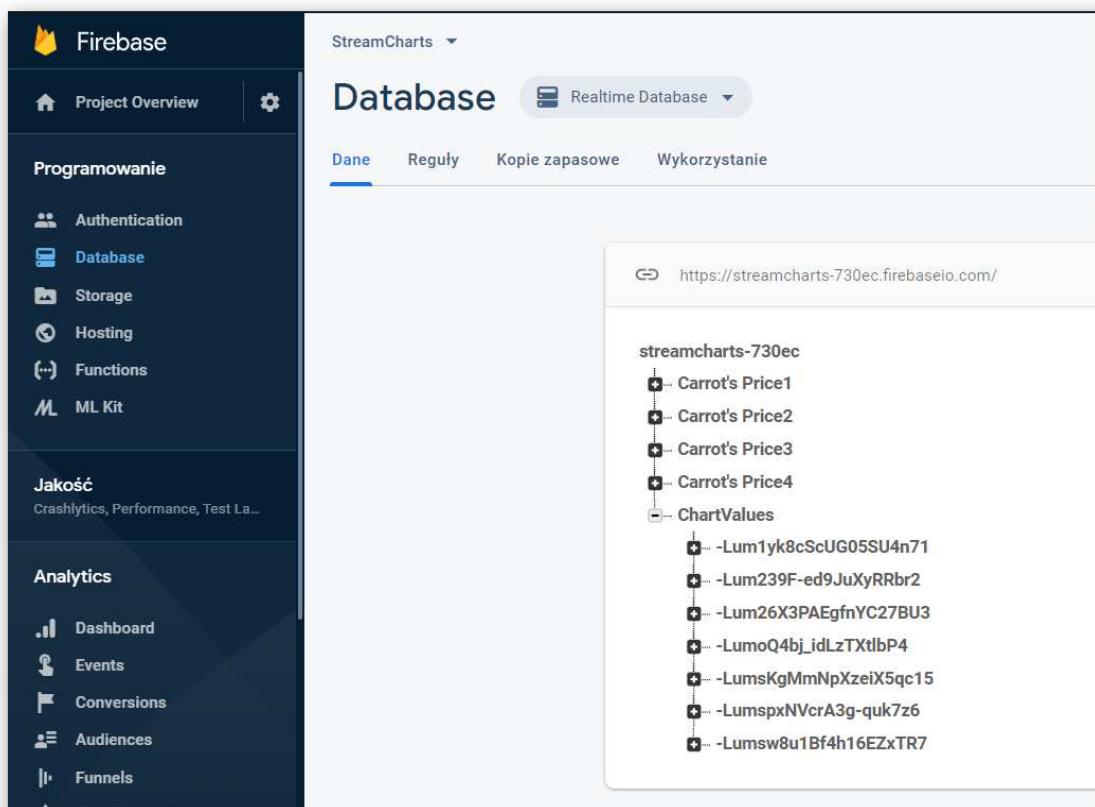
rys. 3 Wykres narysowany z wartości wprowadzonych przez użytkownika ręcznie w aplikacji „Data Stream Chart”.

Nadając zestawowi danych odpowiednią etykietę, użytkownik ułatwia sobie lub innym użytkownikom systemu, późniejszą identyfikację zestawu danych wysłanego na serwer równoległe z rysowaniem wykresu na ekranie aplikacji – rys. 4. Od decyzji administratora serwera zależy, czy zawartość bazy danych czasu rzeczywistego zostanie później zarchiwizowana, plik wyeksportowany, czy usunięty.



rys. 1 Zapis własnego zestawu danych użytkownika powstaje na serwerze równoległe z wykresem rysowanym w aplikacji „Data Stream Chart”.

Jednym ze scenariuszy wykorzystania systemu „Data Stream Chart” opiera się na obsłudze modułu rysowania z danych pobranych z serwera. Istnieją dwa sposoby zasilenia Firebase danymi możliwymi do odczytania z poziomu omawianego modułu. Po pierwsze administrator serwera może umieścić w bazie danych czasu rzeczywistego dowolny zestaw danych, importując, uprzednio walidowany pod względem zgodności z modelem przyjętym w aplikacji, plik w formacie JSON. Funkcje eksportu i importu plików dostarczane są przez twórców Firebase, i dostępne w konsoli administratora usługi. Import pliku do bazy danych czasu rzeczywistego powoduje jednoczesne zastąpienie (utrata) wcześniejszych wpisów bazy. Drugi sposób zasilenia wymaga użycia pomocniczej aplikacji „Data Provider”, która może być zainstalowana na dowolnym urządzeniu sterowanym przy pomocy systemu operacyjnego Android. Dane wygenerowane przez czujniki urządzenia przy pomocy aplikacji przekazywane są do bazy danych czasu rzeczywistego Firebase i zapisywane jako węzły potomne „*ChartValues*”. Wysyłanie nowych danych przez aplikację „Data Provider” nie ingeruje w dotychczasowe wpisy bazy danych w innych węzłach. Oznacza to, że można utworzyć dowolną liczbę klonów aplikacji typu „Data Provider” emitujących strumień danych do określonego węzła bazy danych i dowolną liczbę modułów nasłuchujących na zmiany wartości w określonych węzłach w ramach jednej lub wielu aplikacji, a ich funkcjonowanie na tym samym drzewie danych nie będzie rodziło problemów. Rys. 5 przedstawia taką właśnie sytuację, gdy na drzewie danych znajdują się jeszcze gałęzie zapisane danymi wysyłanymi przez moduł swobodnego rysowania wykresu z własnych danych w aplikacji „Data Stream Chart” i jednocześnie aplikacja „Data Provider” dopisała kolejne gałęzie poczynając od węzła „*ChartValues*”.



rys. 2 Współlistnienie danych różnych węzłów, pochodnych różnych aplikacji na wspólnym drzewie danych.

#### 4. Informacje o możliwości wykorzystania

Potencjalne zastosowanie przedstawionych powyżej możliwości systemu „Data Stream Charts”, jako narzędzia przeznaczonego do prezentacji strumieni danych gromadzonych przez sensory urządzeń mobilnych, zależy głównie od aparatury, do obsługi której będzie użyty. W toku dokumentacji omówiono przykład stacji pogodowej, który jest rezentacyjnym, ale nie jedynym. Szeroka gama sensorów stosowanych w automatyce i dostosowanych do rejestracji rozmaitych sygnałów odbieranych ze środowiska fizycznego sprawia, że przykłady możliwych zastosowań systemu wrażliwego na wychwytywanie zmian tegoż środowiska można by długo mnożyć. Łatwo sobie wyobrazić inżyniera obserwującego wskazania termometru pieca hutniczego na prywatnym smartfonie, jeśli warunki zatrudnienia nakładają na niego takie obowiązki również podczas urlopu. Dzięki przedstawionemu systemowi rolnik odpowiedzialny za duży obszar upraw może swobodnie monitorować np. warunki nasłonecznienia i wilgotności gleby nawet w odległych miejscach swojego areалу. W podobny też sposób lekarz może otrzymać informacje o aktywności



fizycznej swojego pacjenta, pokonywanych przez niego dystansach i parametrach pracy serca. Energetyka, eksploracja trudno dostępnych miejsc – to wszystko już się dzieje w oparciu o podobne narzędzia. System „Data Stream Charts” oferuje w pełni funkcjonalne rozwiązanie, uniwersalne i tanie narzędzie ułatwiające jego użytkownikowi interakcje z maszynami, od których jest zależny. Wyobraźnia jest jedynym ograniczeniem.