

System obsługi sensorów

Projekt zaliczeniowy przedmiotu
"Programowanie współbieżne i rozproszone"

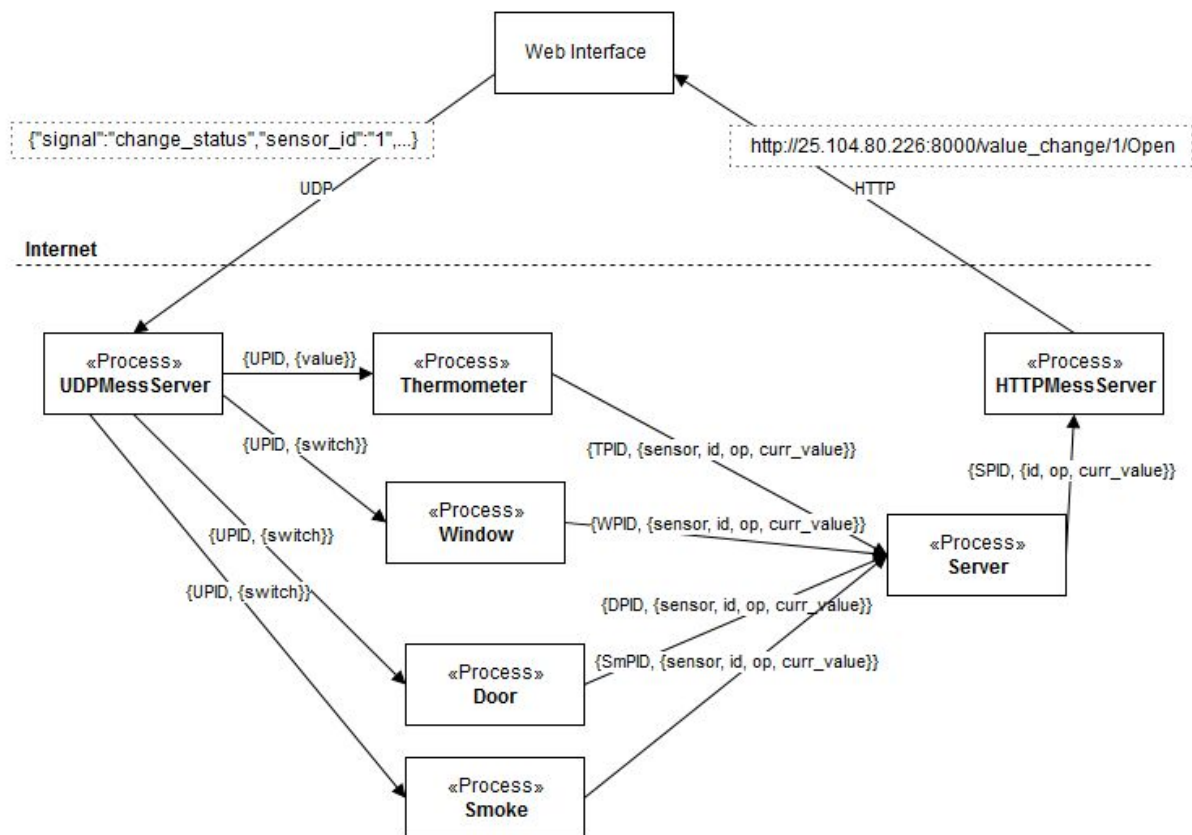
Szymon Nowak,
Arkadiusz Błasiak,
Laboratorium 4, Grupa dziekańska 2
Informatyka r. III, EAIB,
18-01-2016 r.

1. Krótki opis projektu

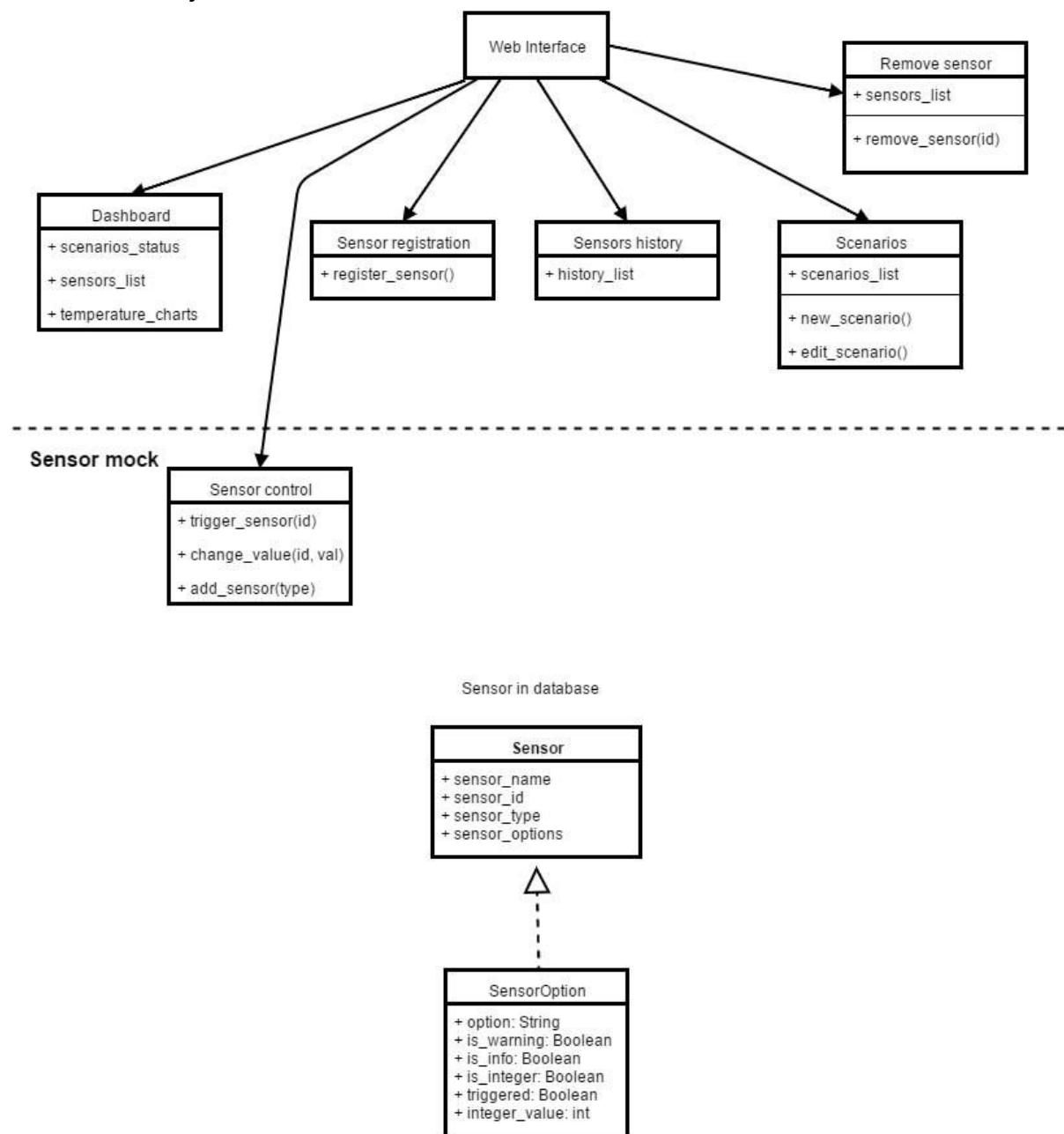
Celem projektu było stworzenie systemu obsługi czujek razem z ich symulowaniem. Założeniem projektu było, że aby użytkownik mógł zarejestrować czujkę konkretnego typu, a następnie dzięki aplikacji webowej badać jej stan. Dla celów wykonania symulacji, z aplikacji webowej jest również możliwość zmiany stanu czujki (symuluje to jakiś odczyt z czujki).

2. Opis i schemat struktury logicznej programu

Symulacja czujek oraz ich serwer napisany został w języku Erlang. Opiera się on na 3 głównych procesach: serwer docelowy, odbiór wiadomości sterujących poprzez protokół UDP, oraz wysyłanie obecnego stanu czujki poprzez protokół HTTP. Wiadomość sterująca wysyłana jest w strukturze Json, przy odpowiednim dobraniu parametrów. Serwer UDP, po odebraniu wiadomości, przekształca ją do konwencji używanej na serwerze (lista krotek opcji). Następnie, krok po kroku, analizuje wiadomość, a następnie wysyła odpowiednią operację z parametrami do ustalonego sensora (to symuluje fizyczne odczytanie przez sensor stanu/wartości). Następnie sensor analizuje swój stan oraz stan przekazany mu w wiadomości, a następnie informuje o jego zmianie serwer. Serwer formułuje odpowiedni komunikat i przekazuje go do procesu serwera HTTP. Serwer zamienia wiadomość w formie krotki na RESTowe rządanie HTTP, które zostaje odebrane przez serwer webowy.



Interfejs obsługowy napisany został w języku Python, z wykorzystaniem technologii takich jak: HTML, jQuery oraz Sass. Zawiera kilka opcji potrzebnych do kontrolowania systemem, tworzenia scenariuszy, rejestracji czujek itp. (schemat poniżej). Serwer odbiera żądania HTTP serwera czujek (SC) i odpowiednio nimi operuje. Użytkownik może sam stworzyć scenariusz, dzięki któremu ustawia własne komunikaty ostrzegające przez zagrożeniem/niebezpieczeństwem na podstawie danych z sensorów. Wszystkie czujki są rejestrowane najpierw na SC, dopiero po przydzieleniu ID czujki z SC, dane zostają zapisane w bazie danych i udostępnione użytkownikowi. W razie potrzeby interfejs posiada zaawansowany panel administracyjny, dzięki któremu użytkownik może bezpośrednio wpływać na ustawienia czujek.



3. Informacje o stosowanych pakietach zewnętrznych

W języku Erlang wykorzystano podstawowe moduły m.in. ssl, inets, httpc, get_udp, string, list, io itp...

W języku Python jest to przede wszystkim framework Django oraz biblioteka socket.

4. Informacje o zastosowaniu specyficznych metod rozwiązania problemu

<<AREK>>

5. Krótka instrukcja obsługi

Po stronie serwera sensorów, aby włączyć sewer, należy skompilować moduł "hs.erl" znajdujący się w folderze Erlang, a następnie wywołać metodę start() ("hs:start().").

Aby włączyć serwer webowy należy wywołać polecenie "python manage.py runserver", opcjonalnie z dopiskiem "0.0.0.0:8000" aby udostępnić serwer na całą sieć.

Aby zarejestrować czujkę, najpierw należy zasymulować sygnał dodawania czujki na serwerze czujek. Aby to zrobić wybieramy "Sensor control" -> Add sensor -> określamy typ i naciskamy "Add", następnie wybieramy "Sensor registration" -> naciskamy "Start", po otrzymaniu sygnału od czujki, należy wpisać nazwę nowego sensora i zatwierdzić przyciskiem.

Aby usunąć czujkę wybieramy "Remove sensor" i klikamy na czerwony przycisk "X" przy czujce którą chcemy usunąć.

Aby zasymulować zmianę stanu czujki wybieramy "Sensor control" -> "Trigger sensor" i klikamy w wybraną czujkę.

Aby zasymulować zmianę wartości (np. temperatura) czujki wybieramy "Sensor control" -> "Change value", klikamy wybraną czujkę i wpisujemy żadaną wartość.

Aby dodać scenariusz wybieramy "Scenarios" i klikamy przycisk "Add", wpisujemy nazwę scenariusza, komunikat ostrzegawczy oraz określamy czy w przypadku uruchomienia scenariusza będzie on tylko ostrzeżeniem czy niebezpieczeństwem. Dalej dodajemy opcje i ustawiamy ich wartości, aby dodać więcej opcji klikamy "Add more options". Po uzupełnieniu wszystkich danych zatwierdzamy przyciskiem "Add scenario".

6. Możliwe rozszerzenia programu

Program umożliwia różne rozszerzenia:

- zwiększenie ilości sensorów - prostota dodawania nowych sensorów pozwala na szybkie dodanie nowego typu czujki. Przykładowo: sensor ruchu, sensor oświetlenia, poziom temperatury wody w kaloryferach, albo bardziej irracjonalne - poziom zabrudzenia akwarium z rybkami, poziom głodu dziecka, sensor pogody itp...
- Wprowadzenie większej ilości scenariuszy - scenariusz to odpowiednia sytuacja, do której zajścia wymagane są konkretne stany konkretnych sensorów. Możliwe jest proste rozszerzenie listy sensorów np: wyziębienie pokoju - jeżeli temperatura spadnie poniżej 16°C, i któreś okno jest otwarte, system sugeruje jego zamknięcie
- Wprowadzenie nowych stanów do obecnie istniejących sensorów - sensor drzwi może wykrywać, czy ktoś w nie puka itp...

7. Ograniczenia programu

System nie posiada ograniczeń ani pod względem ilości sensorów działających w tym samym momencie, ani ilości typów czujek możliwych do zarejestrowania

W przypadku tworzenia scenariuszy wszystkie elementy scenariusza połączone są logicznie spójnikiem "i", nie ma "lub, albo itd."