SCADA 2

Maciej Cebula
Piotr Merynda
Maciej Podsiadło

Spis treści

1 Wstęp **2**
1.1 Opis projektu 2
1.2 Założenia 2

2 Serwer HTTP **3**
2.1 Opis działania 3

3 Baza danych **4**
3.1 Wstęp 4
3.2 Struktura bazy 4
3.3 Opis dostępu do bazy 5

Wstęp

1.1 Opis projektu

1.2 Założenia

Serwer HTTP

2.1 Opis działania

W celu umożliwienia komunikacji aplikacji wizualizacyjnej z bazą danych napisano w języku *JAVA* serwer HTTP. Dane pomiędzy serwerem i aplikacją wymieniane są za pomocą protokołu HTTP, wykorzystując w tym celu zapytania typu *GET* i *POST*. Wszystkie żądania są utożsamione z odpowiednim adresem url i odpowiednio przetwarzane po stronie serwerowej. Aby zapewnić jak największą responsywność aplikacji oraz synchroniczne odświeżanie danych każdy request po stronie serwera przetwarzany jest w osobnym wątku.
Do napisania serwera wykorzystano następujące biblioteki *JAV-wy*:

1. **RXJava** - biblioteka zapewniająca mechanizmy asynchronicznego przetwarzania danych,
2. **JDBI** - biblioteka zapewniająca połączenie z bazą danych,
3. **AKKA-HTTP** - framework do implementacji serwera HTTP,
4. **Javax mail** - biblioteka do wysyłania emaili.

Baza danych

3.1 Wstęp

Aby móc wizualizować przebieg pracy systemu wszystkie najważniejsze dane z punktu widzenia automatyki są logowane w bazie danych. Do tego celu wykorzystano bazę danych typu *MYSQL* i środowisko *MYSQL Workbench*, zainstalowane na jednym z komputerów, którym stworzono całą strukturę przedstawioną na rysunku 3.1. Dostęp poszczególnych węzłów systemu do bazy danych zapewniony jest poprzez połączenie wszystkich komputerów w lokalną sieci.

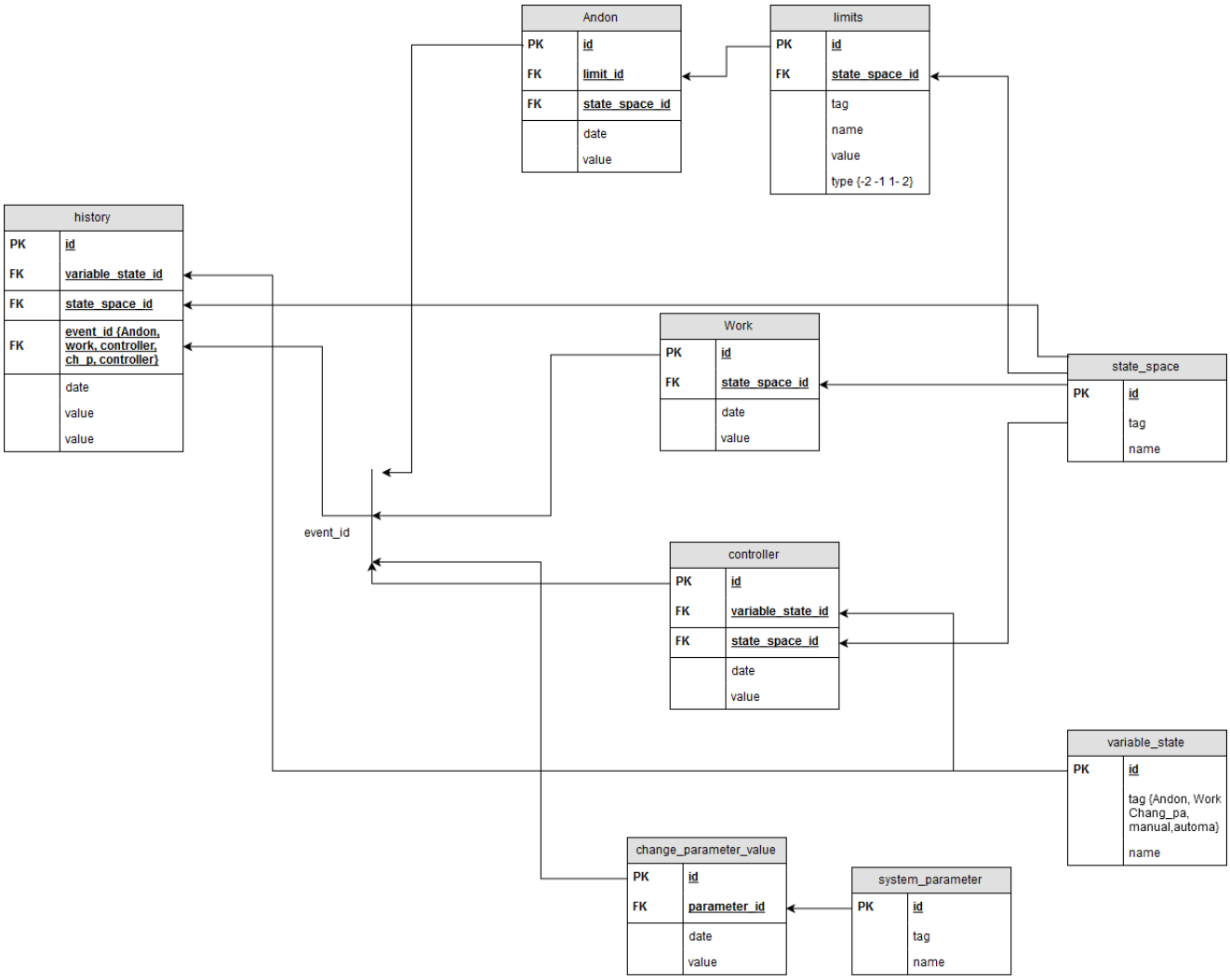
3.2 Struktura bazy

Zaprezentowana na rysunku baza danych na strukturę relacyjną aby zapewnić możliwość ewentualnej rozbudowy sytemu o inne składowe. Z racji na to, że baza służy głównie do logowania pracy systemu podzielono ją na następujące części:

1. główną tabele **history**, w której logowane są wszystkie zdarzenia w kolejności chronologicznej,
2. tabelę **Andon**, w której zapisywane są wszystkie sytuacje alarmowe,
3. tabelę **Work**, do której logowany jest stan normalnej pracy systemu,
4. tabelę **change_parameter**, w której zapisywane są wszelkie zmiany wartości parametrów systemu,
5. tabelę **controller**, która służy do logowania pracy regulatorów.

Dodatkowo:

1. w tabeli **state_space** zdefiniowano wszystkie zmienne stanu występujące w systemie,
2. tabele **variable_state** definiuje wszystkie możliwe stany danej zmiennej np. Work, Andon, Change parameter itd.
3. tabela **limits** definiuje poszczególne limity i przypisuje je do odpowiednich zmiennych stanu z tabeli **state_space**,
4. w tabeli **system_parameter** zdefiniowane są pozostałe parametry systemu takie jak: nastawy poszczególnych regulatorów, wartości zadane itp.



Rys. 3.1: Struktura bazy danych.

3.3 Opis dostępu do bazy

Dane dotyczące parametrów systemu w poszczególnych chwilach czasu są logowane przez serwer OPC i następnie prezentowane w aplikacji.

Bibliografia