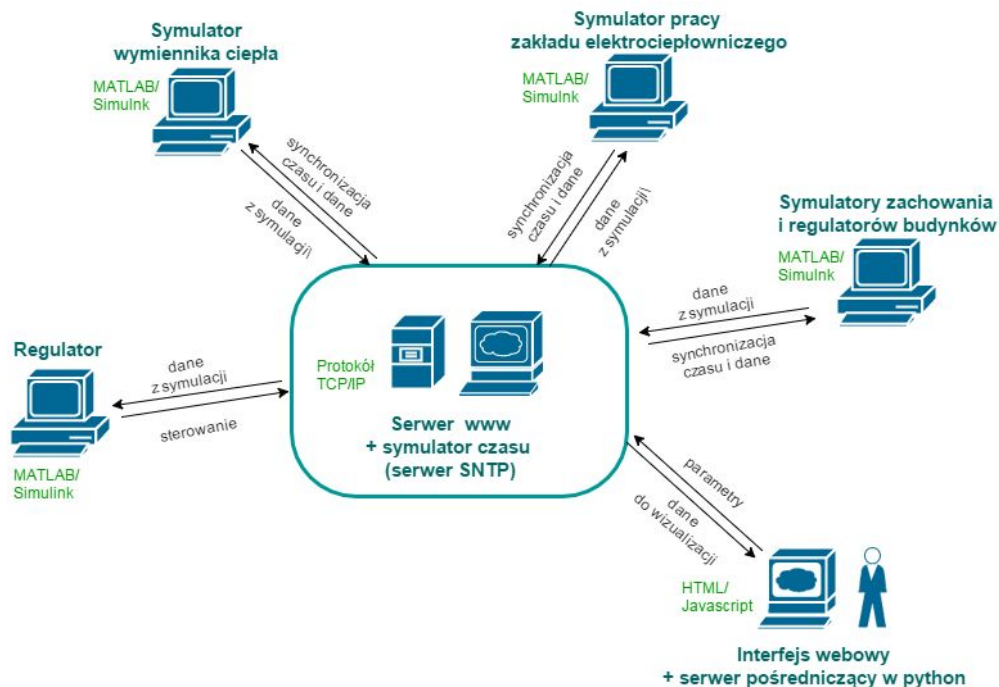


Podstawy komputerowych systemów sterowania
**Budowa rozproszonego systemu modelowania i sterowania instalacją CO na
przykładzie dystrybucji ciepła w budynkach AGH**
poniedziałek 13:15, grupa B

Podział zadań

Imię i nazwisko	Obszar zadań	Wprowadzenie
Żaneta Błaszczuk	Koordynator	Sporządzanie raportów, nadzór prac nad projektem, wsparcie zespołu. Sporządzenie planu komunikacji pomiędzy klientami serwera. Zarządzanie podziałem obowiązków i zadań.
Piotr Pałucki	Komunikacja, symulator czasu	Zaprojektowanie protokołu wymiany danych i synchronizacji czasu, opracowanie ramek, implementacja serwera, dostarczenie interfejsu do transmisji w Matlabie.
Dawid Wiktor	Komunikacja	Stworzenie centralnego systemu zarządzania warstwą komunikacyjną projektu przy zastosowaniu protokołu TCP/IP.
Filip Kubicz Andrzej Stach	Interfejs webowy	Strona HTML/Javascript, odpowiednia do prezentacji stanu obiektów i umożliwiająca zmianę nastaw/parametrów. Serwer obsługujący websockets napisany w pythonie z użyciem modułu Tornado
Joanna Muniak	Symulator wymiennika ciepła	Stworzenie symulacyjnego modelu wymiennika w środowisku Simulink.
Jakub Nowak	Symulator pracy zakładu elektrociepłowniczego	Symulacja zmiany temperatury Tzm, symulacja awarii, wystawienie Tzm oraz To.
Łukasz Micor	Modele zachowania i regulatory budynków	Model budynku jego i regulatora stworzony w Simulinku
Szymon Kucharczyk	Regulator	Opracowanie algorytmu regulatora wraz z implementacją urządzenia.

Schemat systemu:



Użyte narzędzia do zarządzania projektem:

GitHub – <https://github.com/Zanet/PKSS>

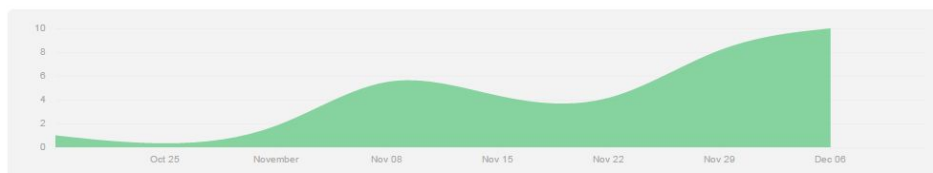
Repozytorium ułatwiło dzielenie się elementami projektu i współpracę.

Przebieg zmian na repozytorium:

Oct 18, 2015 – Dec 12, 2015

Contributions to master, excluding merge commits

Contributions: Commits



Drzewo projektu na repozytorium:



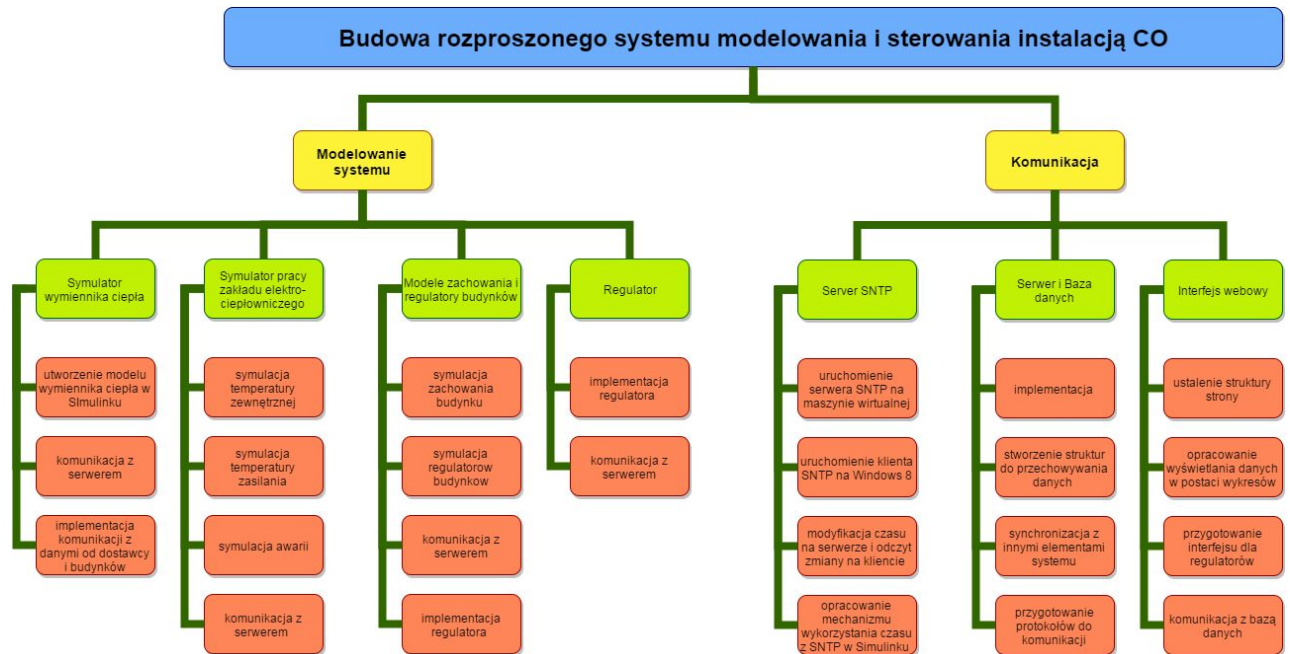
Draw.io - narzędzie współpracujące z Dyskiem Google ułatwiło współpracę przy tworzeniu schematów takich jak WBS czy schemat systemu.

Dokumenty Google – pozwoliły wspólnie współpracować nad jednym dokumentem dotyczącym na przykład przygotowywanego raportu.

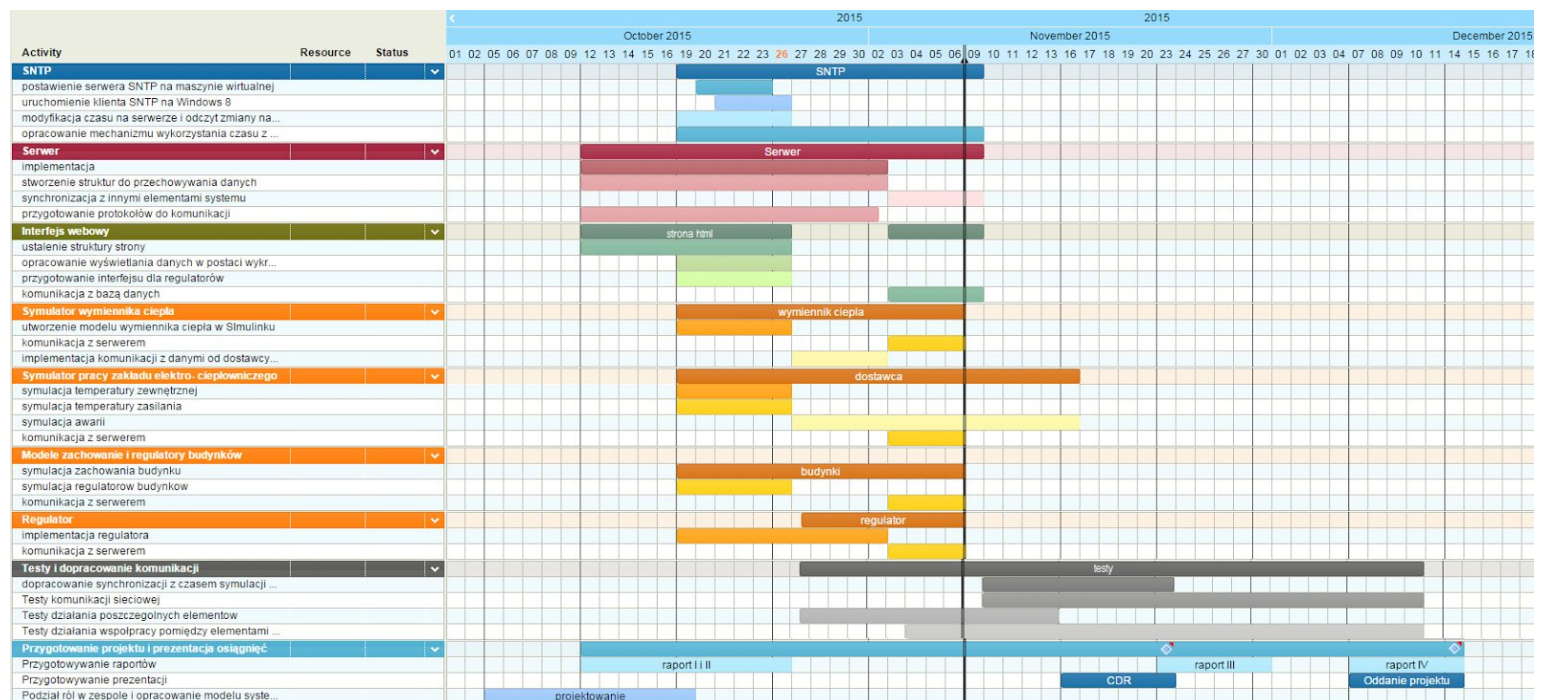
Grupa na facebook – Informowanie się na bieżąco na temat zaistniałych sytuacji przebiegało sprawnie podczas korzystania z narzędzi, które współistnieją przy serwisie z którego korzystamy na co dzień.

tomsplanner.com – pozwolił na stworzenie wykresu GANTT'a

Struktura podziału pracy (WBS)

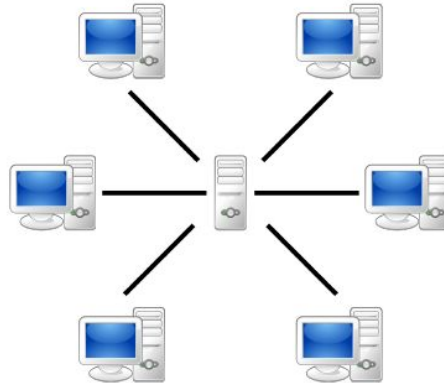


Wykres GANTT'a projektu

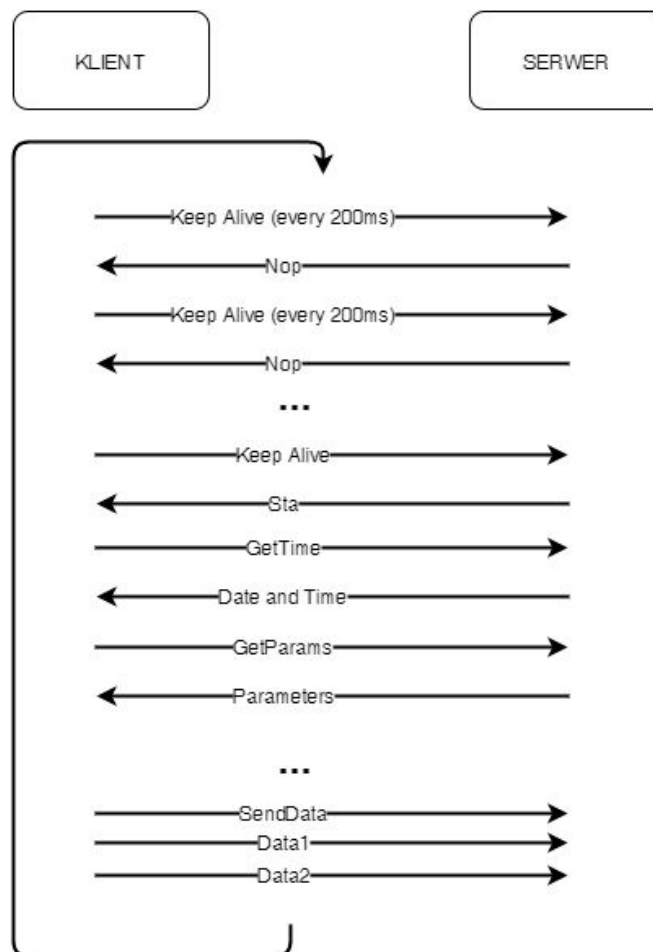


Opisy poszczególnych modułów:

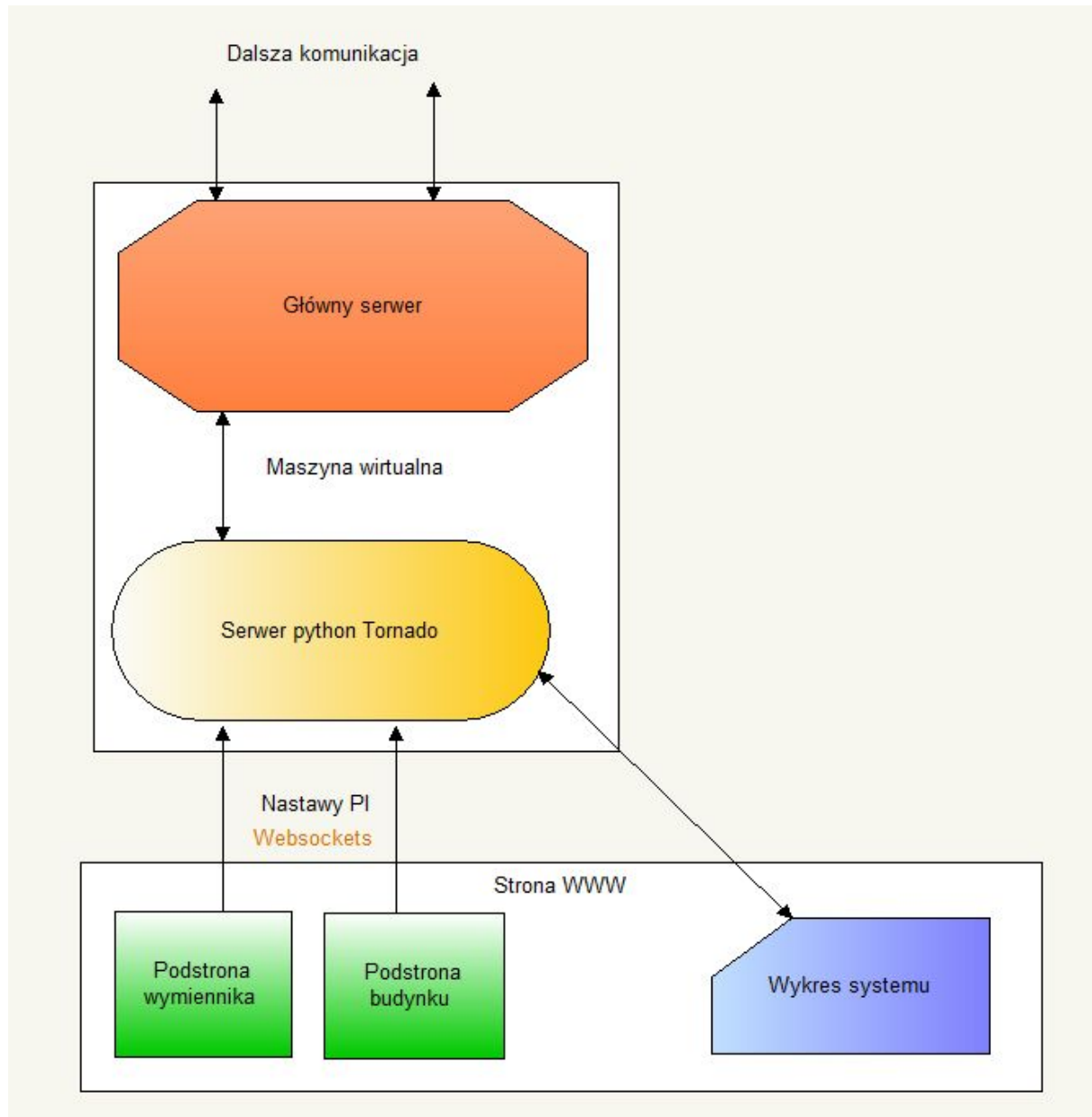
Główny Serwer - aplikacja napisana w języku C, obsługując komunikację z wykorzystaniem protokołu TCP/IP. Serwer realizuje topologię gwiazdy. Jako medium transmisyjne wykorzystano sieć Wi-Fi. Dane są logowane w postaci pliku CSV(Comma Separated Values).



Zaimplementowano własny protokół wymiany danych **hPKSS** (high Performance Keyless Service Sharing) opisany poniższym diagramem:



Interfejs webowy używa protokołu Websockets do komunikacji z serwerem centralnym. W celu obsługi zapytań HTTP pochodzących od przeglądarki napisano serwer HTTP w pythonie, korzystając z modułu *Tornado*.



Symulator Elektrociepłowni

Zadaniem symulatora elektrociepłowni jest wygenerowanie temperatury otoczenia oraz temperatury zasilania miejskiego oraz realizację symulacji awarii elektrociepłowni.

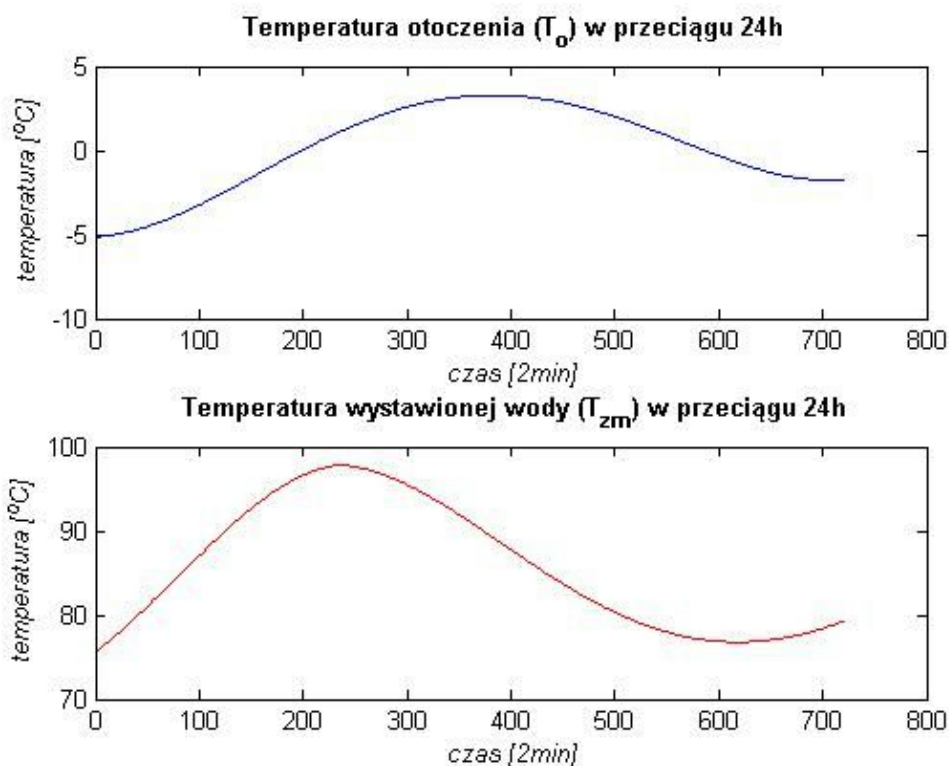
Przyjęliśmy, że kształt temperatury otoczenia w ciągu dnia układu się do w zbliżony sposób do krzywej sinusoidalnej. W celu zróżnicowania przebiegów, krzywą tą przemnożyliśmy przez zmienne losowe.

Do wyznaczenia temperatury zasilania miejskiego T_{zm} użyto prostego równania liniowego oraz przesunięto wartość o 8 godzin (tak jak wymagają warunki zadania - czas przepływu między elektrociepłownią a AGH to 8 godzin). Generowanie temperatury otoczenia odbywa się raz na dzień. Podczas generowania temperatury otoczenia uwzględniono ciągłość temperatur (ostatnia wartość temperatury z dnia poprzedniego pokrywa się z pierwszą temperaturą dnia następnego). Po stronie elektrociepłowni zaimplementowanie zostały również ograniczenia na minimalną i maksymalną temperaturę zasilania miejskiego.

Obie temperatury wystawiane są co krok symulacji na port szeregowy.

Dodatkowo realizowana była symulacja awarii (elektrociepłownia nie może nagrzewać wody) - wtedy wystawiano jako temperaturę T_{zm} , otrzymaną temperaturę T_{pm} .

Poniższy wykres przedstawia przebieg temperatury otoczenia i zasilania miejskiego jednego dnia.



Opis funkcjonowania budynku i jego regulatora

Do budynku dostarczana jest woda z wymiennika ciepła. Zasila ona kaloryfery w budynku. Budynek posiada uśrednioną temperaturę kaloryferów i pomieszczeń. Ochłodzona woda wypływa ze budynku i trafia ponownie do wymiennika, gdzie następnie jest ponownie ogrzewana. Budynek posiada swój własny regulator P, którego parametry również mogą być zmieniane z poziomu interfejsu webowego. Maksymalny przepływ wody w budynku wynosi 40 m³/h. Zadaniem regulatora jest takie regulowanie tym przepływem, aby utrzymać zadana temperaturę pomieszczeń. Model budynku i kaloryferów został opisany za pomocą poniższych równań różniczkowych :

1) Model budynku

$$m_b c_b \frac{dT_r}{dt} = k_h (T_{PCO} - T_r) - k_{ext} (T_r - T_o)$$

m_b - masa powietrza w budynku; wartość około 20000 kg

c_b - ciepło właściwe powietrza; wartość około 1000 J/kg/K

k_{ext} - współczynnik utraty ciepła; wartość około 1500 J/K/s

2) Model kaloryferów

$$m_h c_h \frac{dT_{PCO}}{dt} = F_{COB} c_w (T_{ZCO} - T_{PCO}) - k_h (T_{PCO} - T_r)$$

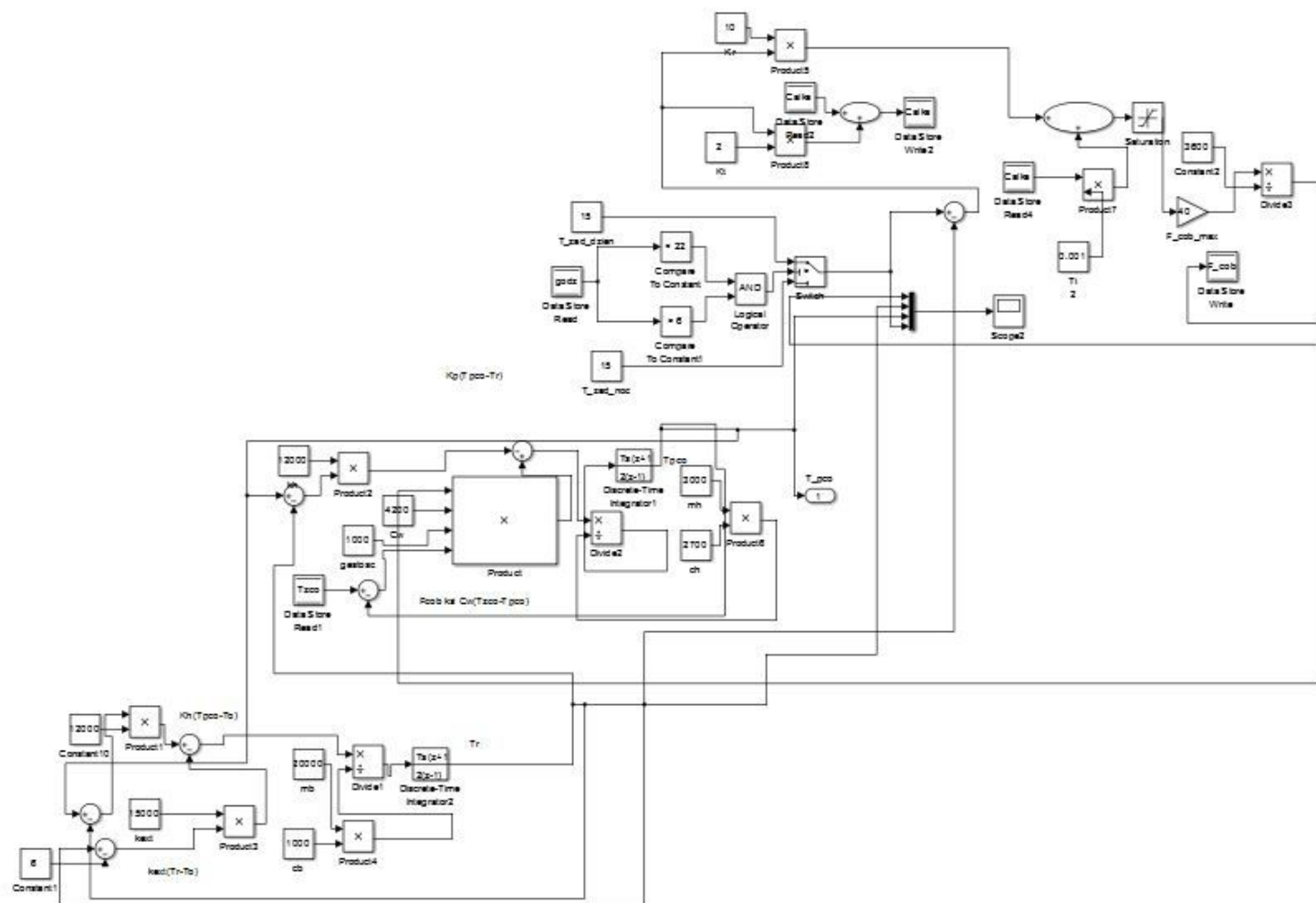
m_h – masy zastępcze wszystkich kaloryferów; wartość około 3000 kg

c_h - ciepło właściwe kaloryferów; wartość około 2700 J/kg/K

k_h – współczynnik przenikania ciepła kaloryferów-pokój; wartość około 12000 J/K/s

F_{COB} – maksymalny pobór wody w budynku; 40 m³/h

Układy te zostały zamodelowane w Simulinku :

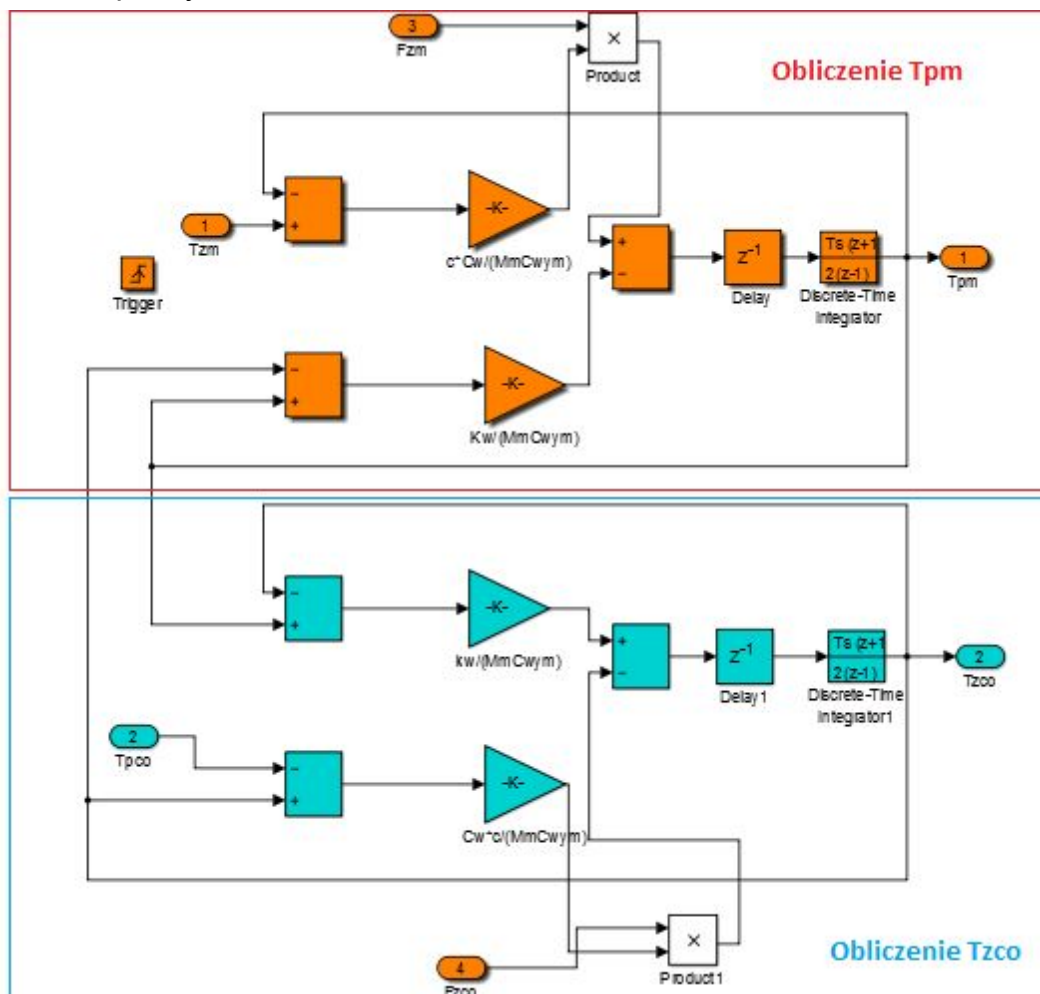


Wymiennik ciepła

W wymienniku ciepła, woda z sieci miejskiej ogrzewa wodę krążącą w instalacji CO w sieci AGH. Można więc powiedzieć, że łączy on elementy symulacji związane z regulatorem wymiennika, elektrociepłownią i budynkami.

Wymiennik może być zamodelowany w postaci układu równań różniczkowych cząstkowych, jednak w celu uproszczenia obliczeń wykorzystano równania o parametrach skupionych.

Zawartość podsystemu:



Regulator wymiennika

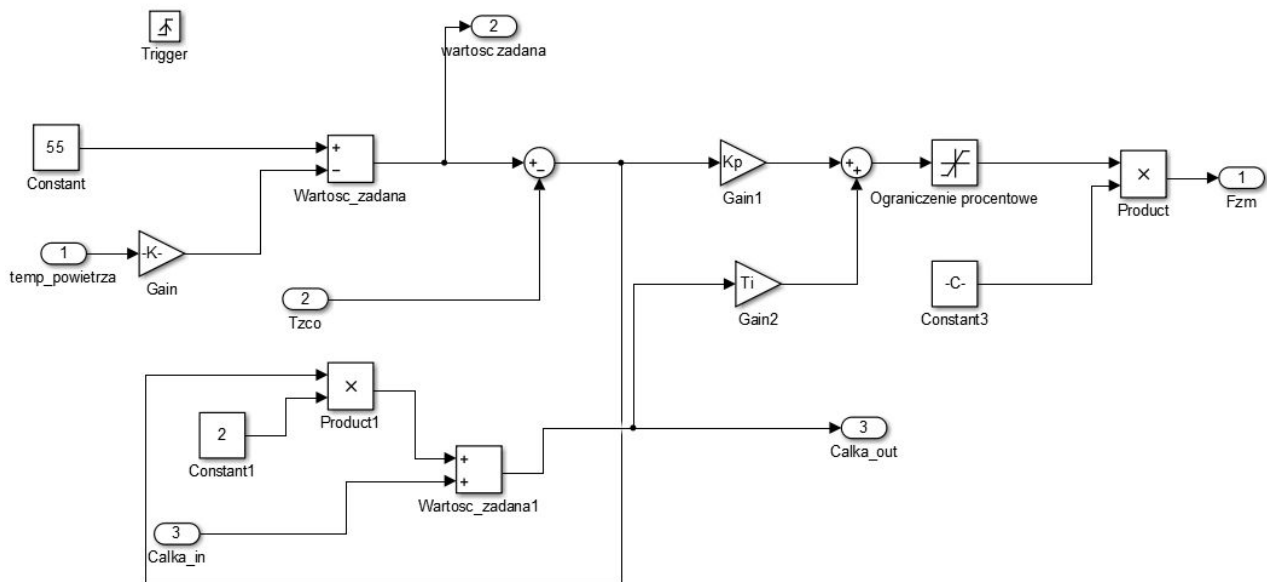
Regulator wymiennika jest urządzeniem sterującym przepływem wody grzewczej. Sterowanie przepływem wody realizowane jest poprzez elektrozawór. Strumień wody wpływającej do wymiennika ciepła regulowany jest od 0-100% maksymalnego przepływu. Maksymalny możliwy przepływ wynosi 80t/h.

Temperatura strumienia wody wpływającego do wymiennika jest zależna od temperatury otoczenia. Odpowiednia temperatura wody grzewczej powinna być zapewniona przez elektrociepłownię. Opóźnienie transportowe związane z odległością pomiędzy systemem grzewczym a elektrociepłownią wynosi około 8h. Temperatura dostarczana na wymiennik może być więc nie adekwatna do temperatury otoczenia.

Głównym zadaniem regulatora jest takie sterowanie przepływem F_{zm} , aby temperatura wody wypływającej z wymiennika ciepła była dana poniższym wzorem:

$$T_{zco} = 55 - 1.75T_o [^{\circ}\text{C}] \quad , \quad T_o - \text{temperatura otoczenia } [^{\circ}\text{C}]$$

Opóźnienie transportowe może spowodować, że nawet przy pełnym otwarciu zaworu, niemożliwe stanie się uzyskanie temperatury zadanej. Jako regulator odpowiedni do realizacji ww. zadania posłużył regulator PI. W wyniku testów stwierdzono, że zadowalające rezultaty osiąga się również bez użycia części całkującej. W porównaniu z poprzednimi wersjami regulatora, w wersji ostatecznej zrezygnowano z użycia blozków *DataStore*. Powodowały one niepotrzebne opóźnienia w trakcie symulacji. Regulator otrzymywał od serwera czas symulacji, temperaturę otoczenia, temperaturę wody wypływającej z wymiennika oraz nastawy. Finalny schemat regulatora przedstawiono poniżej:



Rys.1 Finalna wersja regulatora PI.

Podsumowanie

Podział pracy

Dzięki dobremu przydzieleniu obowiązków udało się sprawnie współpracować. Wiedzieliśmy kto z kim ma się komunikować i koordynator starał się dopilnować przekazywania informacji. Istotna była komunikacja osób odpowiedzialnych za serwer z twórcami modeli. Udało się połączyć elementy w jedną całość.

Schemat systemu

Wbrew początkowym założeniom wszystkie części związane z modelami i regulatorem zostały zrealizowane z Użyciem pakietu MATLAB/Simulink. Użycie tego rozwiązania ma swoje uzasadnienie zwłaszcza dla części związanej z modelami, ponieważ ułatwia analizę działania poszczególnych elementów systemu podczas testów. Usprawniło to wykonanie testów jednostkowych i sprawdzenie poprawności systemu jeszcze przed uruchomieniem komunikacji przez serwer.

Niestety nie udało się stworzyć bazy danych. Dane z poszczególnych symulacji są jednak logowane przez serwer do plików testowych. Ponadto przebieg zmian jest widoczny w interfejsie graficznym na interfejsie webowym.

Użyte narzędzia do zarządzania projektem

Repozytorium było naszym głównym źródłem plików współdzielonych pomiędzy użytkownikami. Nauczyliśmy się korzystać wielu branch i zasad współpracy na repozytorium w większym zespole. Mieliśmy okazję utrwalić wiadomości na temat gita w praktyce.

Użyliśmy wszystkich założonych na początku projektu narzędzi.

WBS

Udało się zrealizować wszystkie założenia wymienione w instrukcji do projektu.

Gantt

Wykres Ganta był dobrym źródłem wiedzy na temat tego na jakim etapie powinny się znajdować poszczególne zadania i prezentował wszystkim w jaki sposób są od siebie zależne poszczególne elementy. Pomógł też oceniać w jakich momentach występuje poszczególne Milestone'y.

W praktyce przebieg prac nieco się różnił, ponieważ na etapie testów modeli wynikły problemy, które opóźniły prace nad scaleniem systemu.