

Ustalenia projektowe – Kontroler

Osoba odpowiedzialna:

TJ (sigrond)

Środowisko:

C++11 + sockets + pthread / thread z C++11

Założenia:

1. Kontroler powinien zostać uruchomiony zanim użytkownik zacznie wprowadzać polecenia do GUI (lub innymi słowy: zanim widok zacznie wysyłać zapytania).
2. Kontroler po uruchomieniu tworzy gniazdo (socket) w nowym wątku (lub wątkach – fork server) i nasłuchuje na porcie. Po nawiązaniu połączenia może obsługiwać polecenia (data):
 - a) Wprowadzanie lub modyfikowanie danych i ustawień.
 - b) Uruchomienie obliczeń (algorytm ewolucyjny w modelu).
 - c) Zwracanie stanu obliczeń lub wyniku.
 - d) Zatrzymywanie obliczeń
3. Kontroler przekazuje dane i ustawienia do modelu, oraz pobiera z niego stan i wyniki.
4. Kontroler zażąda życia modelu, uruchamia go w osobnym wątku (lub wątkach). Może też zażądać zatrzymywania wątków modelu.

Ustalenia komunikacji MVC:

1. Komunikacja z widokiem
 - Kontroler NIE serwuje pliku widoku, ewentualnie może go uruchomić przez shella lub kontroler i widok mogą zostać uruchomione skryptem.
 - Spodziewam się, że widok będzie plikiem HTML otwieranym w przeglądarce i będzie używał JavaScript/jQuery/AJAX get do wysyłania (prosty) zapytań HTMLowych do kontrolera. Aczkolwiek dzięki zastosowaniu wzorca MVC i wykorzystaniu socketów do komunikacji można widok wymienić na zupełnie inny w innej technologii o ile będzie się stosował do ustaleń protokołu komunikacji.
 - Do komunikacji będzie potrzebne częściowe zaimplementowanie HTTP/1.1 - [RFC 2616](#). Chodzi głównie oczywiście o sam nagłówek.
 - Odpowiedzi i działania kontrolera będą zależały od sekcji data w get. Niech na chwilę obecną będzie

```
{polecenie: 'dane', nazwa_parametru: 'wartość_parametru'}
{polecenie: 'licz'}
{polecenie: 'wynik'}
{polecenie: 'stop'}
```
 - Dane do widoku będą wysyłane jednym ze standardów AJAXa np. json lub xml. Co i jak dokładnie będzie wy tych danych wymaga dodatkowych

ustaleń już bardziej związanych z implementacją.

2. Komunikacja z modelem będzie znacznie prostrza niż z widokiem. Należy ustalić, czy kontroler po prostu wywołuje metody modelu, czy raczej kontroler tylko uruchomi proces modelu i dalej będzie się komunikował przez kolejkę blokującą.