Marcel Kawski gr. I1

Projekt nr 3- Dokumentacja

Mój projekt reprezentujący logikę programu nadzorującego działanie elektrowni atomowej składa się z 9 klas:

* Reactor
* PressWatReactor
* HeavyWatReactor
* GasCoolReactor
* Manager
* Sensor
* Switchgear
* Menu
* Test

Wszystkie klasy do prywatnych atrybutów innej klasy dostają się poprzez tzw. gettery, czyli publiczne metody zwracające wybrany atrybut podanego obiektu.

1. Reactor

Jest to klasa bazowa zawierająca dane każdego z reaktorów znajdujących się w elektrowni. Zawiera prywatne dane: numer, temperaturę, moc, typ reaktora oraz publiczne metody służące do operacji na powyższych danych. Dziedziczące po niej klasy PressWatReactor, HeavyWatReactor oraz GasCoolReactor są jej szczególnymi typami różniącymi się rodzajami chłodziwa używanymi do produkcji energii. Klasa Reactor zawiera wirtualną metodę odpowiadającą ze informowanie pracownika elektrowni, jakiego rodzaju chłodziwa ma użyć do obsługi reaktora.

1. PressWatReactor

Jest to klasa reprezentująca reaktor chłodzony wodą pod wysokim ciśnieniem.

1. HeavyWatReactor

Jest to klasa reprezentująca reaktor ciężkowodny.

1. GasCoolReactor

Jest to klasa reprezentująca reaktor chłodzony gazem.

1. Manager

Jest to klasa zarządzająca wszystkimi reaktorami. Zawiera prywatne atrybuty: łączną moc reaktorów, czyli całej elektrowni oraz wektor wskaźników na obiekty klasy Reactor. Zawiera również metody, które pozwalają na obsługę pewnych funkcjonalności, np. liczenie łącznej mocy elektrowni.

1. Sensor

Jest to klasa reprezentująca czujnik temperatury. Zawiera publiczną metodę służącą do pomiaru chwilowej temperatury. Temperatura może się zmieniać co sekundę, liczba pomiarów jest ustawiona na domyślną wartość 10, ale może być mniejsza, jeśli temperatura w reaktorze będzie za niska lub za wysoka. Wtedy program alarmuje pracownika elektrowni o takim stanie.

1. Switchgear

Jest to klasa reprezentująca rozdzielnicę elektryczną. Zawiera publiczną metodę służącą do włączania reaktorów. Zaczyna od zmienienia stanu reaktora na aktywny, wypisuje informację o potrzebnym do produkcji energii rodzaju chłodziwa, następnie opuszcza pręty paliwowe i ustawia temperaturę reaktora na domyślną wartość 1000 stopni Celsjusza.

1. Menu

Jest to klasa odpowiadająca za komunikację z użytkownikiem, służąca do oddzielenia interfejsu programu od jego logiki. Nie zawiera żadnych atrybutów, jedynie publiczną metodę umożliwiającą obsługę interfejsu.

1. Test

Ta klasa również nie zawiera żadnych atrybutów, jedynie publiczne metody, pozwalające testować działanie wybranych metod w programie. Sprawdzają, czy oczekiwany, z góry znany wynik zgadza się z wynikiem działania funkcji. Są to funkcje typu bool.

Aby rozbudować program należałoby dodać klasy reprezentujące inne elementy elektrowni, takie jak kominy, generatory itd. oraz dodać wiele istotnych dla działania elektrowni atrybutów tych oraz istniejących już klas. Dobrą praktyką byłoby działanie na bardziej dokładnych liczbach, dokładniejsze ich porównywanie oraz dokładne sprawdzanie poprawności tych danych. Należałoby sprawić, aby program był niezawodny i odporny na wprowadzanie niepoprawnych danych przez użytkownika. Testy powinny obejmować więcej przypadków, a nie tylko wybrane. Mierzenie temperatury reaktora powinno być związane z rzeczywistym urządzeniem pobierającym dokładną temperaturę chwilową panującą we wnętrzu reaktora. Dla wygody pracownika elektrowni członkowie zespołu programistycznego powinni zaprojektować czytelny i intuicyjny interfejs graficzny, który mógłby ułatwić mu pracę.

Co ciekawe, projekt zawiera klasę Reactor, jednak nie jest możliwe powstanie obiektu tej klasy. Jest to klasa bazowa i możliwe jest tylko powstanie obiektów, które są przedstawicielami bardziej szczegółowych klas, konkretnych rodzajów reaktora. Jest to tzw. czysta wirtualność.

