

Specyfikacja wstępna projektu

Autor: Szymon Kędzierski

Opis projektu

Temat projektu to komputerowa gra w szachy dla dwóch graczy z komunikacją sieciową. Serwer oraz każdy klient będą działać w osobnych wątkach. Działanie będzie oparte na wzorcu MV lub MVC, jeżeli okaże się to konieczne. Do potrzebnych plików należą: pliki źródłowe oraz grafiki reprezentujące wygląd figur na planszy.

Przeznaczenie klas i metod

Chess

Klasa nadrzędna, stanowi środowisko do komunikacji dla obiektów pozostałych klas. Jest kompozytem składającym się z obiektów Board i Player.

Metody

pickFigure – zmienia wybraną figurę w stanie modelu

moveFigure – wywołuje metodę move wybranej wcześniej figury

isCheck – sprawdza czy zabicie odpowiedniej figury króla jest możliwe

isCheckMate – wywoływana, jeżeli metoda isCheck zwróci **true**. Sprawdza, czy istnieją ruchy które zapobiegą zabicie odpowiedniej figury króla, w przeciwnym wypadku kończy rozgrywkę.

Board

Model planszy do gry w szachy, przechowuje obecny stan gry. Na stan gry składa się mapa przechowująca figury i ich pozycje na planszy, pole aktywnego gracza, obecny stan automatu stanów.

Figure

Abstrakcyjna klasa reprezentująca wszystkie typy figur szachowych

Metody:

- getMoves – zwraca zbiór możliwych ruchów do wykonania przez figurę
- move – metoda zmieniająca pozycję figury

FigureView

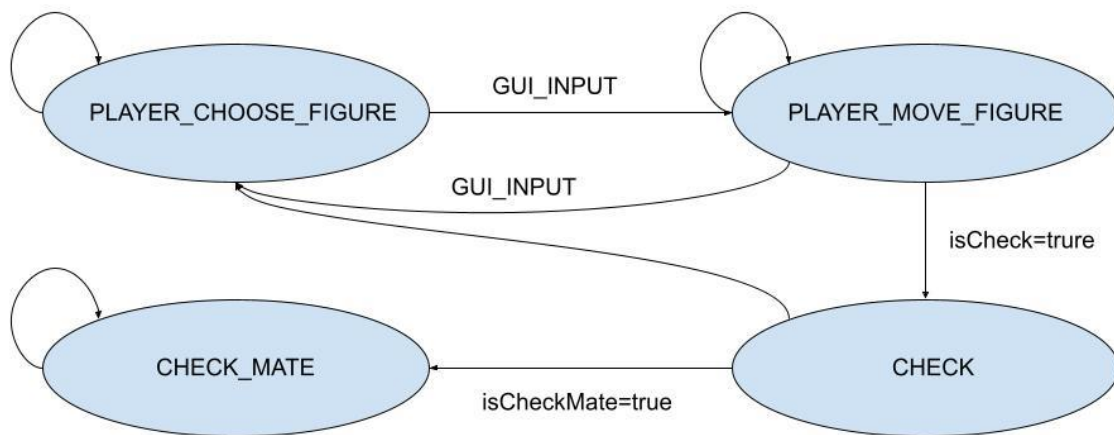
Widok figury, wyświetla pozycje figury, do której został przypisany.

BoardView

Widok planszy, składający się z przycisków pól szachowych. Nasłuchuje zdarzeń od GUI.

Zasada działania

Na najwyższym poziomie program opiera się na automacie stanów, którego graf jest pokazany poniżej



W stanie `PLAYER_CHOOSE_FIGURE` aplikacja czeka na kliknięcie jakiegoś pola, w tym stanie czeka na kliknięcie pola, na którym znajduje się pionek należący do aktywnego gracza.

W stanie `PLAYER_MOVE_FIGURE` GUI wyświetla wszystkie legalne ruchy dla wybranej figury i czeka na wybranie pola, do którego może dotrzeć dana figura lub wciśnięcie prawego przycisku myszy co spowoduje cofnięcie wyboru figury. Po wykonaniu ruchu następuje sprawdzenie czy król nie aktywnego gracza może zostać zбитy w jego kolejnej turze.

Jeżeli nastąpi przejście do stanu `CHECK` następuje wyznaczenie zbioru ruchów które zapobiegają zбитia króla. Gdy ten zbiór jest pusty następuje przejście do stanu `CHECK_MATE` w którym w zależności od reakcji użytkownika rozpoczyna nową grę lub wykonywanie programu kończy się.