Wzorce:

1. Gateway – Klasa Board hermetyzuje dostęp do mapy pionków
2. Layer Supertype – Klasa Game jest implementowana przez różne gry spełniające rolę kontrolera
3. Separated Interface - Klasy Gra I Pionek są wirtualnymi klasami które wymagają implemętacji aby móc dodć nową grę do programu
4. Registry – Klasa Board obsługuje dostęp do pionków uczestniczących w grze
5. Value Object – Coordinates przechowuje parę intów określającą położenie na planszy
6. Special Case - EmptySpace jest typem pionka który reprezentuje sytuacje braku pionka na danych koordynatach
7. Plugin – Klasa Game jest implementowana przez grę wybraną z interfejsu wstępnego, w czsie działania aplikacji
8. Model View Controller - Model – Pawn, View – MainForm, Controloer - Game
9. Identity Map – Board przyśpiesza dostęp do obiektów Pionków przy użyciu mapy w której kluczem są Coordinates
10. Identity Field – Coordinates są naturalnym identity pionków
11. Remote Facade – Board udostępnia metodę PlayersPieces która zwraca tylko pionki należące do danego gracza, często wymagana funkcjonalność
12. Data Transfer Object – Game przechowuje obiekt rememberedMovesMap który zapisuje informacje o wszystkich możliwych ruchach aktywnego gracze, te informacje są używane na potrzeby ich wyświetlenia, ale mogą być także używane przez inne procesy, np. Na potrzeby kalkulacji możliwych ruchów
13. Transaction Script – BoardMove przechowuje serie prostych operacji, na planszy które w połaczeniu tworzą jeden ruch