На диаграмме отмечены идеи некоторых паттернов, которые используются в программе «Шашки с искусственным интеллектом». Идея не означает что паттерн реализован вточности по GOF. Это скорее направление мыслей программиста.

1. Серверный обработчик события перемещения фигуры:

HumanPlayerService  
+TryMoveFigure

*IHumanPlayerService  
+TryMoveFigure*

*Пользовательский интерфейс*

MoveFigureController  
+Post

*(доменный сервис пакета DomanServices)*

*Пакет HumanPlayer (поведение игрока)*

Фасад

Приспособленец

*IValidateHumanService  
+CanMove*

Мост

ValidateHumanService  
+CanMove

*(доменные сервисы  
 пакета DomanServices)*

*IMoveRulesService  
+MoveFigureWithoutValidation*

*IValidateRulesService  
+GetAllowedDestinations*

*Пакет Rules  
(правила игры)*

MoveRulesService  
+MoveFigureWithoutValidation

ValidateRulesService  
+GetAllowedDestinations

Декоратор

*IValidateFigureService  
+GetAllowedMoveVectors*

*IValidateEatService  
+CanEatFigure*

ValidateFigureService  
+GetAllowedMoveVectors

ValidatePawnService  
+GetAllowedMoveVectors()

ValidateQueenService  
+GetAllowedMoveVectors

eService

Стратегия

Стратегия

*IValidatePawnService*  
+*GetAllowedMoveVectors*

*IValidateQueenService*  
*GetAllowedMoveVectors*

1. Серверный обработчик, вычисляющий следующий шаг искусственного интеллекта

*Пользовательский интерфейс*

IIntellectStepController  
+Post

*(доменный сервис пакета DomanServices)*

*IComputerPlayerService  
+CalculateStep*

*Пакет ComputerPlayer  
(искусственный интеллект)*

ComputerPlayerService  
+CalculateNextStep

Итератор

Фасад

IPositionWeightService  
+GetWeghtForWhite

*IPositionWeightService  
+GetWeghtForWhite*

*IStepIteratorService*  
*+GetNextStepVariants*

StepIteratorService  
+GetNextStepVariants

*(доменные сервисы  
пакета DomanServices)*

Реализации этих интерфейсов на предыдущей схеме

*IDirectMoveService  
+DirectMove*

*IValidateFiguresService  
+GetAllowedMoveVectors*

*Архитектор и разработчик игры - Сергей Рыскулов.*