

МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
Кафедра МОЭВМ

ОТЧЕТ
по лабораторной работе №3
по дисциплине «Объектно-ориентированное программирование»
Тема: Логическое разделение классов

Студент гр. 8382

Янкин Д.О.

Преподаватель

Жангиров Т.Р.

Санкт-Петербург

2020

Цель работы.

Получить навыки логического разделения классов и их ответственностей.

Постановка задачи.

Разработать и реализовать набора классов для взаимодействия пользователя с юнитами и базой. Основные требования:

- Должен быть реализован функционал управления юнитами
- Должен быть реализован функционал управления базой

Выполнение работы.

Был реализован класс GameFacade, отвечающий за связь между логикой программы и ее интерфейсом: он управляет графической сценой и графическими элементами. Каждому логическому элементу на поле соответствует графический. Слоты графических элементов могут создавать команду UiCommand, передавая ей необходимые параметры. Команда в свою очередь обращается к GameFacade. GameFacade в зависимости от входных параметров делегирует работу другим компонентам.

Паттерн Фасад. В качестве фасада используется класс GameFacade, упомянутый выше. Он принимает команды от интерфейса и на их основе принимает решения о делегировании команд другим объектам. Смотреть /Game/GameFacade.h.

Паттерн Посредник. Был реализован класс GameMediator. Он занимается обработкой некоторых взаимодействий юнитов и выполняет все необходимые проверки связанные с игровым полем – сами объекты не имеют возможности обращаться к Field и, следовательно, не обладают информацией о других объектах, их окружающих. Смотреть /Game/GameMediator.h.

Паттерн Команда. Для передачи команд из графического интерфейса в фасад и из фасада далее используются команды. Смотреть /Command/Command.h.

Паттерн Цепочка Обязанностей. В фасаде крупный кусок кода занимал выбор действия юнита и проверки на их допустимость. Эти проверки были разбиты на несколько классов, последовательно вызывающих друг друга. Сам фасад вызывает начало цепочки – UnitActionHandler. Смотреть директорию /Handler.

Выводы.

Были получены навыки логического разделения классов и их ответственностей.