

МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
Кафедра математического обеспечения и применения ЭВМ

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

Студентка гр. 8381

Преподаватель

Звегинцева Е.Н.

Жангиров Т.Р.

Санкт-Петербург

2020

Цель работы.

Разработать и реализовать наборы классов для взаимодействия пользователя с юнитами и базой.

Задание.

Основные требования:

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

Дополнительные требования:

- Реализован паттерн “Фасад”, через который пользователь управляет программой
- Объекты между собой взаимодействуют через паттерн “Посредника”
- Для передачи команд используется паттерн “Команда”
- Для приема команд от пользователя используется паттерн “Цепочка обязанностей”

Выполнение работы.

В программу были добавлены следующие основные классы:

- Game - класс обладает функционалом: создание баз, нейтральных объектов и помещение их на поле. В нем хранятся поле, базы и информация о юнитах. Учет юнитов реализован с помощью посредника, в связи с чем информация обновляется с ходом игры
- GameCommand, FieldCommand, BaseCommand – реализованы по принципу паттерна «команда». Данные классы наследуются от абстрактного класса Command, описывающего все основные поля и методы необходимые для совершения определенного действия. Команда поля дополнительно принимает на вход указатель на объект, если нужно определить его координаты. А также отдельным методом сообщает указатель на объект, если нужно найти его координаты

- Facade – реализован по принципу паттерна «фасад». Фасад имеет множество методов, которые умеют работать с командами: передают параметры и обрабатывают выходные значения.

Указанные выше классы, реализованные по принципу паттерна «команда», имеют следующие свойства

- Каждая команда может пользоваться методами определенных классов, а именно
- Входные параметры, указатель на используемый класс, указатель на посредника для команд, тип команды принимаются конструктором команды

В программе реализован функционал управления базой, юнитами, а также вывод различной информации. Цепочки обязанностей формируются из классов команд и запускаются из фасада.

Атака юнитов реализована посредником AttackMediator, который хранит всех юнитов, а также указатель на поле. Доступ к посреднику, который хранится в классе Game, тоже есть у всех юнитов.

Выводы.

В ходе выполнения лабораторной работы была написана программа, в которой реализованы классы для функционала программы и взаимодействия пользователя с программой. Был использован объектно-ориентированный стиль программирования, были изучены и применены его основные положения, а также реализованы некоторые паттерны проектирования.