

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

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

Студентка гр. 8382

\_\_\_\_\_

Рочева А.К.

Преподаватель

\_\_\_\_\_

Жангиров Т.Р.

Санкт-Петербург

2020

### **Цель работы.**

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

### **Ход выполнения.**

Для управления пользователем игрой был реализован класс Facade (Facade.hpp). Он содержит указатель на класс Manager, через который тоже ведется управление, методы startGame и makeAction. В методе startGame вызывается метод Manager::startGame(). В методе makeAction() вызываются методы Manager::printMenu() и Manager::makeChoice().

Класс Manager содержит указатели на различные команды, наследуемые от класса Command (все они находятся в папке commands) и указатель на игровое поле. Оно выводится на экран в методе printField().

Пользователь может создать юнита (вызывается метод CreateUnitCommand::execute, в котором продолжается работа с пользователем. Он выбирает, какой типа юнита создать и т.д.), удалить юнита (DeleteUnitCommand::execute), передвинуть юнита (MoveUnitCommand::execute), атаковать какого-либо юнита на поле (AttackUnitCommand::execute), взаимодействовать с нейтральным объектом, с которым он находится в одной клетке (ActionWithObjectCommand::execute), узнать значение здоровья базы (Base::getHealth), вывести информацию об юнитах, принадлежащих базе (в формате (x, y) : type, Base::printUnits), вывести игровое поле или узнать значения атрибутов определенного юнита (UnitsAttributesCommand::execute()).

Игра продолжается, пока пользователь при выборе дальнейшего действия не введет число 0.

При выборе очередного действия оно не сразу выполняется, а проходит в несколько этапов (в разных классах и методах разные проверки). В каждом таком отдельном методе действие может быть окончено, если что-то мешает ему (неверная информация, занятая клетка и т.п.).

### **Выводы.**

В ходе выполнения работы были разработаны классы для взаимодействия пользователя с юнитами и базой.