

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

ОТЧЕТ
по лабораторной работе №5
по дисциплине «Объектно-ориентированное программирование»
Тема: управление, разделение на уровни абстракции

Студент гр.0382

Ильин Д.А.

Преподаватель

Жангиров Т.Р.

Санкт-Петербург

2021

Цель работы.

Реализовать управление игрой, используя разделение на уровни абстракции.

Задание.

Необходимо организовать управление игрой (номинально через CLI). При управлении игрой с клавиатуры должна считываться нажатая клавиша, после чего происходит перемещение игрок или его взаимодействия с другими элементами поля.

Требования:

- Реализовать управление игрой. Считывание нажатий клавиш не должно происходить в классе игры, а должно происходить в отдельном наборе классов.
- Клавиши управления не должны жестко определяться в коде. Например, это можно определить в отдельном классе.
- Классы управления игрой не должны напрямую взаимодействовать с элементами игры (поле, клетки, элементы на клетках)
- Игру можно запустить и пройти.

Потенциальные паттерны проектирования, которые можно использовать:

- *Команда (Command)* - передача команд с информацией через единый интерфейс, помещение команд в очередь
- *Посредник (Mediator)* - организация взаимодействия различных модулей

Выполнение работы.

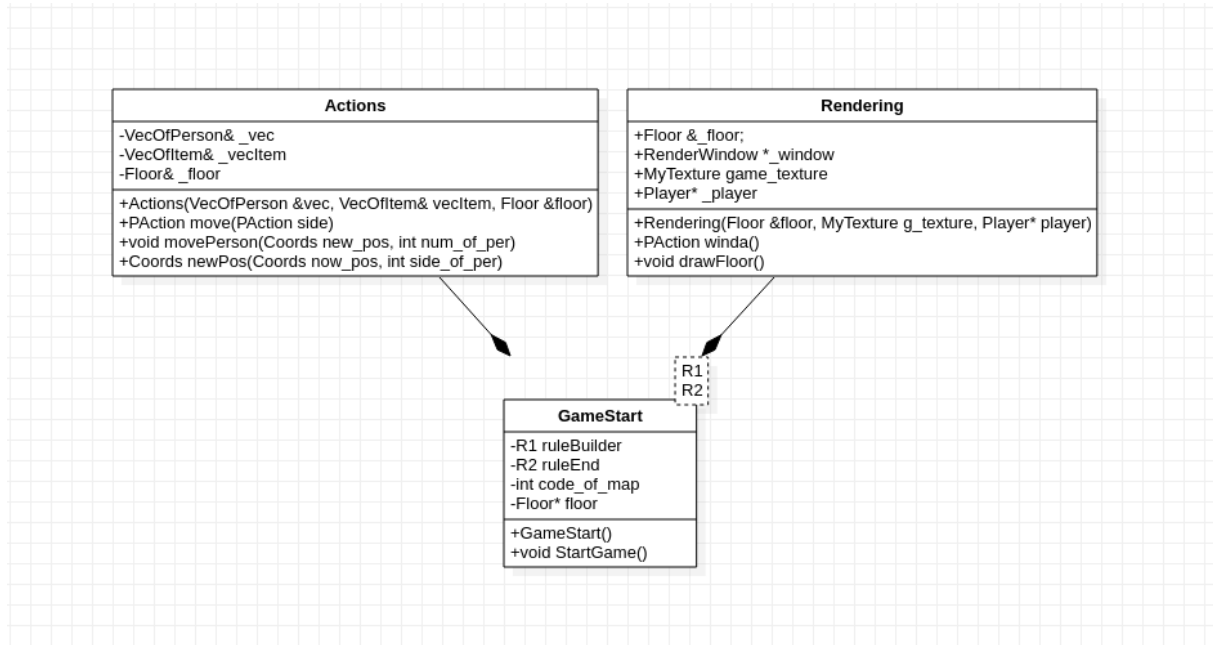
Для управления игрой было написано три класса: Actions, GameStart и Rendering. Класс Rendering создан для считывания нажатий клавиш при управлении игрой вне класса игры Game и отрисовки происходящего. Поскольку используется библиотека SFML, то мы можем обрабатывать событие event, пока открыто окно. Во время работы игры, мы постоянно отслеживаем event. Чтобы отслеживать события было создано перечисление PActions. В нём хранятся события в которых может быть игра, а именно: *END*, *LEFT*, *RIGHT*, *UP*, *DOWN*, *NOTHING*, *WIN*, *DEAD*. В классе Rendering происходит цикл считывания команд с клавиатуры, которые передаются в класс GameStart, после чего обрабатываются, об этом позже. Так же в данном классе происходит отрисовка всех характеристик персонажа.

В классе GameStart происходит создание всего, что нам нужно, а так же главный цикл, внутри которого происходит проверка на то, какое состояние игры сейчас(из перечисления PActions). В нём открывается окно, после чего в переменную pressed передаётся нынешнее состояние, и пока состояние не равно *END* продолжается игра. При нажатии клавиш отвечающих за передвижение вызывается объект класса Actions, отвечающий за все изменения на игровом поле(передвижение объектов, получение вещей, атака существ и игрока).

В классе Actions происходят все действия которые могут произойти на карте, т.е. атаки, поднятие вещей, передвижение, восстановление здоровья и т.д..

UML-диаграмма классов представлена на рис. 1.

Рисунок 1 – UML-диаграмма классов.



Выводы.

В ходе работы было изучено разделение работы на уровни абстракции; было написано управление, работа которого как раз и разбивается на уровни абстракции.