Санкт-Петербургский государственный университет
Кафедра системного программирования

Создание 2D-игры в жанре «roguelike» с использованием Unity Engine

Урбанский Денис Андреевич, группа 21.Б07-мм

Научный руководитель: к.т.н. Ю.В.Литвинов, доцент кафедры системного программирования

Санкт-Петербург
2023

Введение

- Успех Magicka
- Популярность продуктов с высокой реиграбельностью
- Примеры игр с подобной боевой системой:
  - Noita
  - Nine Parchments

Постановка задачи

Цель: создание прототипа двумерной игры в жанре «roguelike» с использованием Unity Engine.

Задачи на осенний семестр:

- Выполнить обзор аналогов
- Спроектировать базовую логику игры
- Реализовать основные механики: простой ИИ противников, создание заклинаний из имеющихся у игрока сфер
- Разработать план апробации

Существующие аналоги

Цель обзора: Выделить особенности, в том числе — подход к ведению боя

Noita

- Процедурная генерация
- Возможность изменять заклинания только между уровнями
- Максимально возможная сложность чар зависит от оружия

Nine Parchments

- Предварительно созданные линейные уровни
- Заклинания принадлежат максимум ко двум стихиям и заданы без возможности изменения
- Возможность разблокировать игровых персонажей и оружие

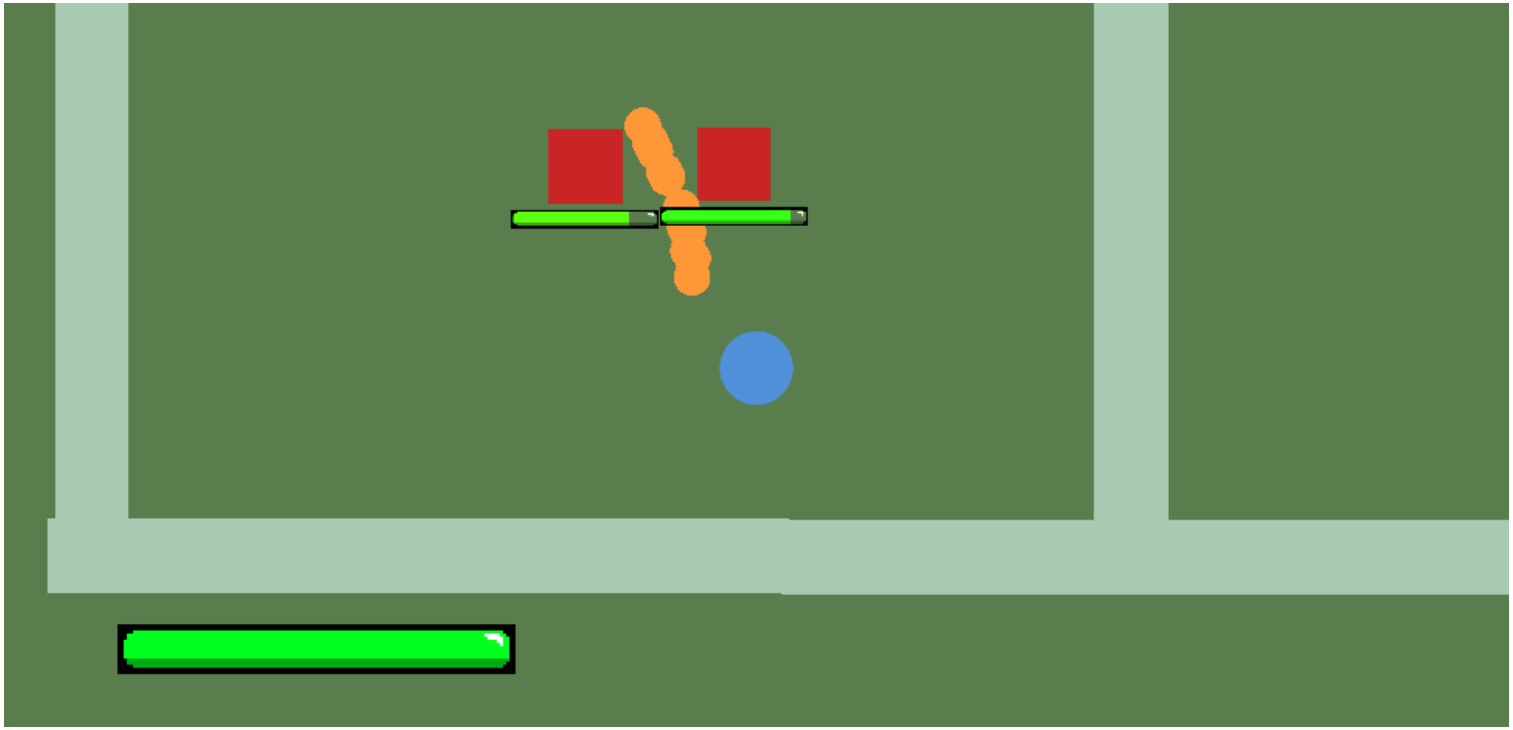
Использованные технологии

- Язык программирования C#
- Игровой движок Unity Engine
  В том числе:
  - Components
  - MonoBehaviour
  - ScriptableObject

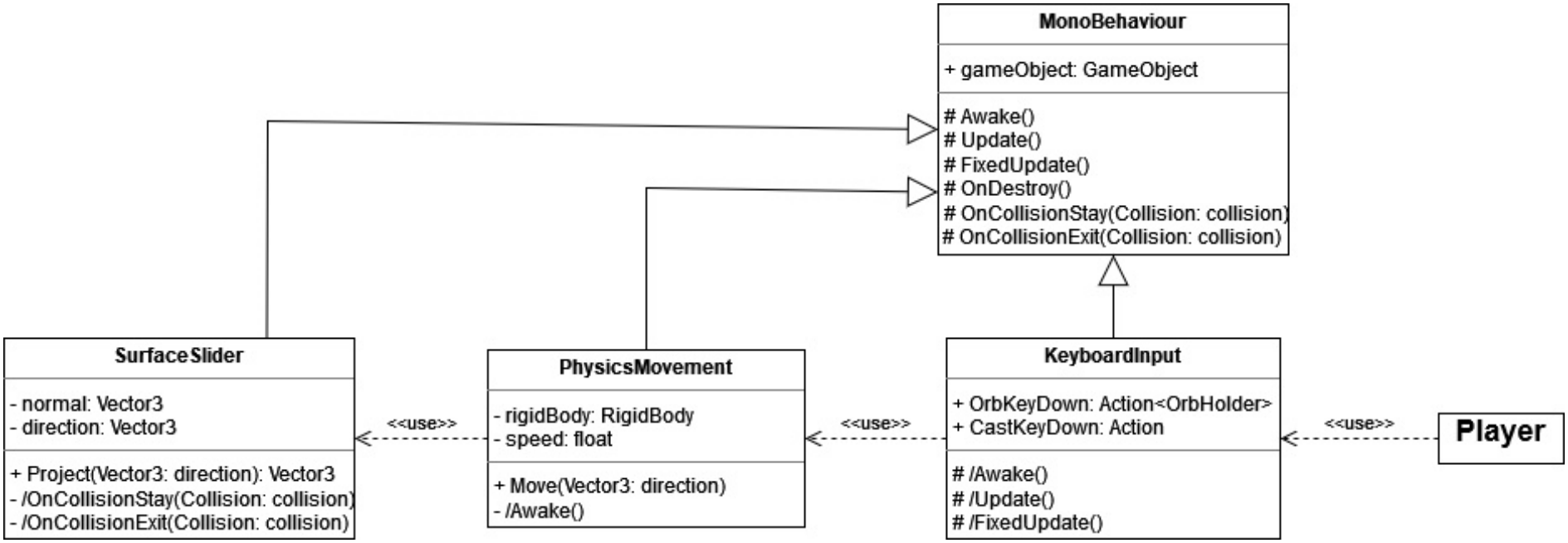
Реализация — механики

- Перемещение и здоровье игрока
- Базовые противники
- Создание заклинаний

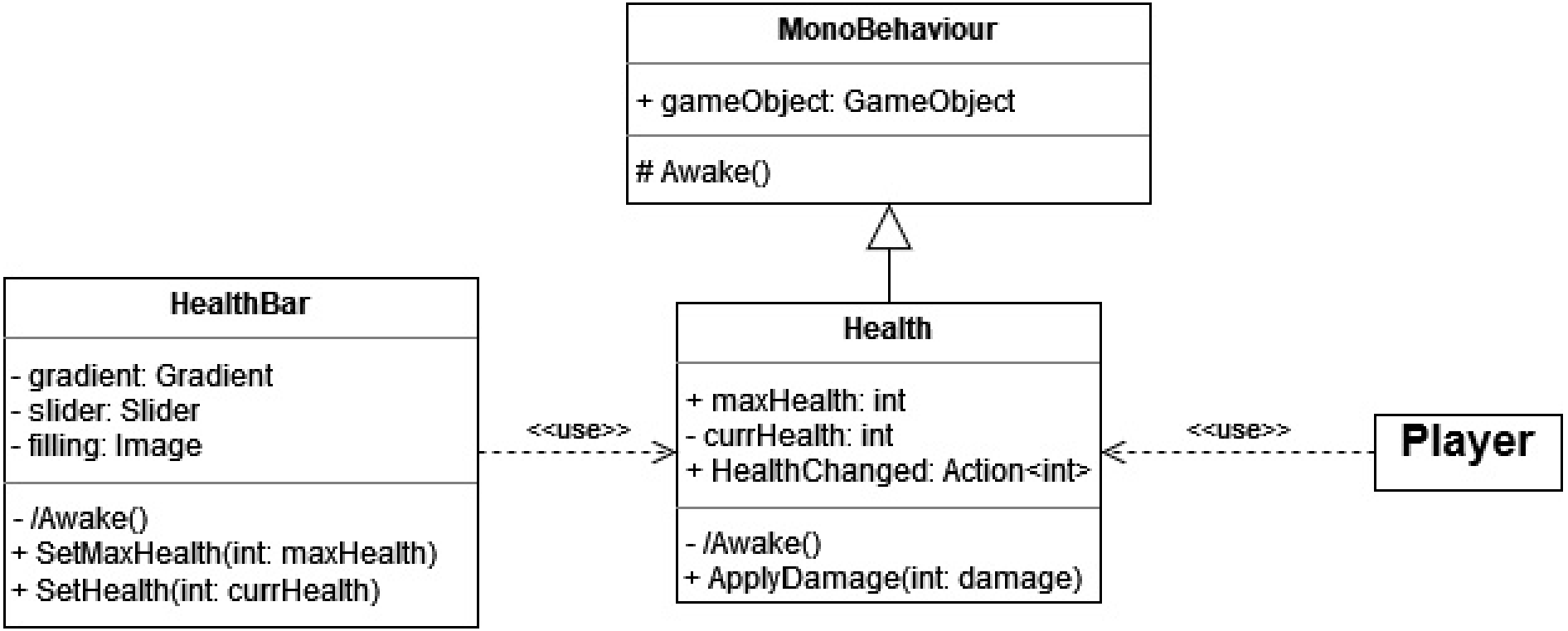
Реализация — портрет типичного игрока и снимок игрового процесса



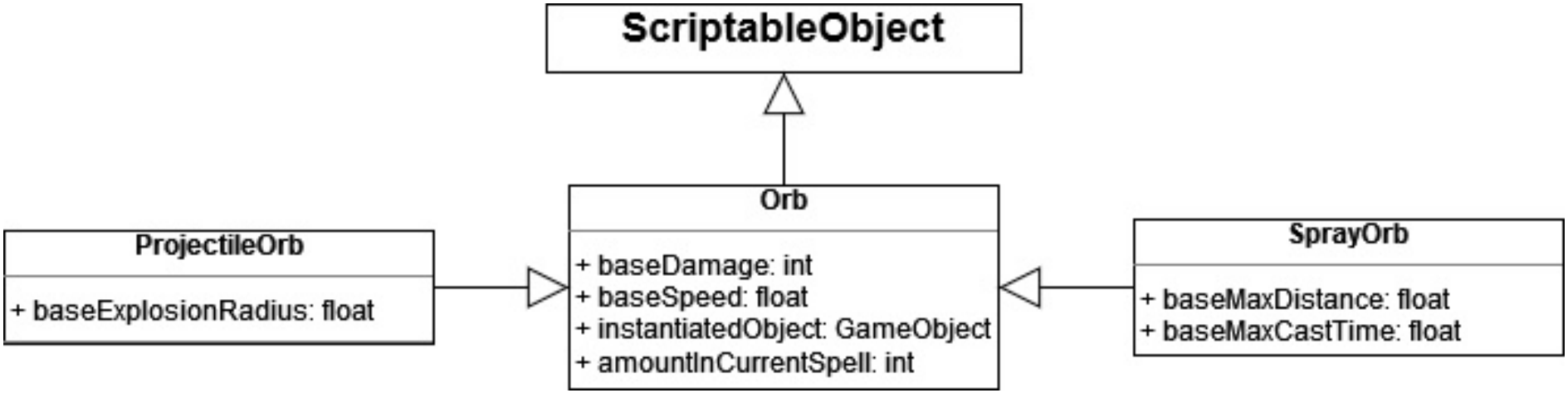
Реализация — передвижение



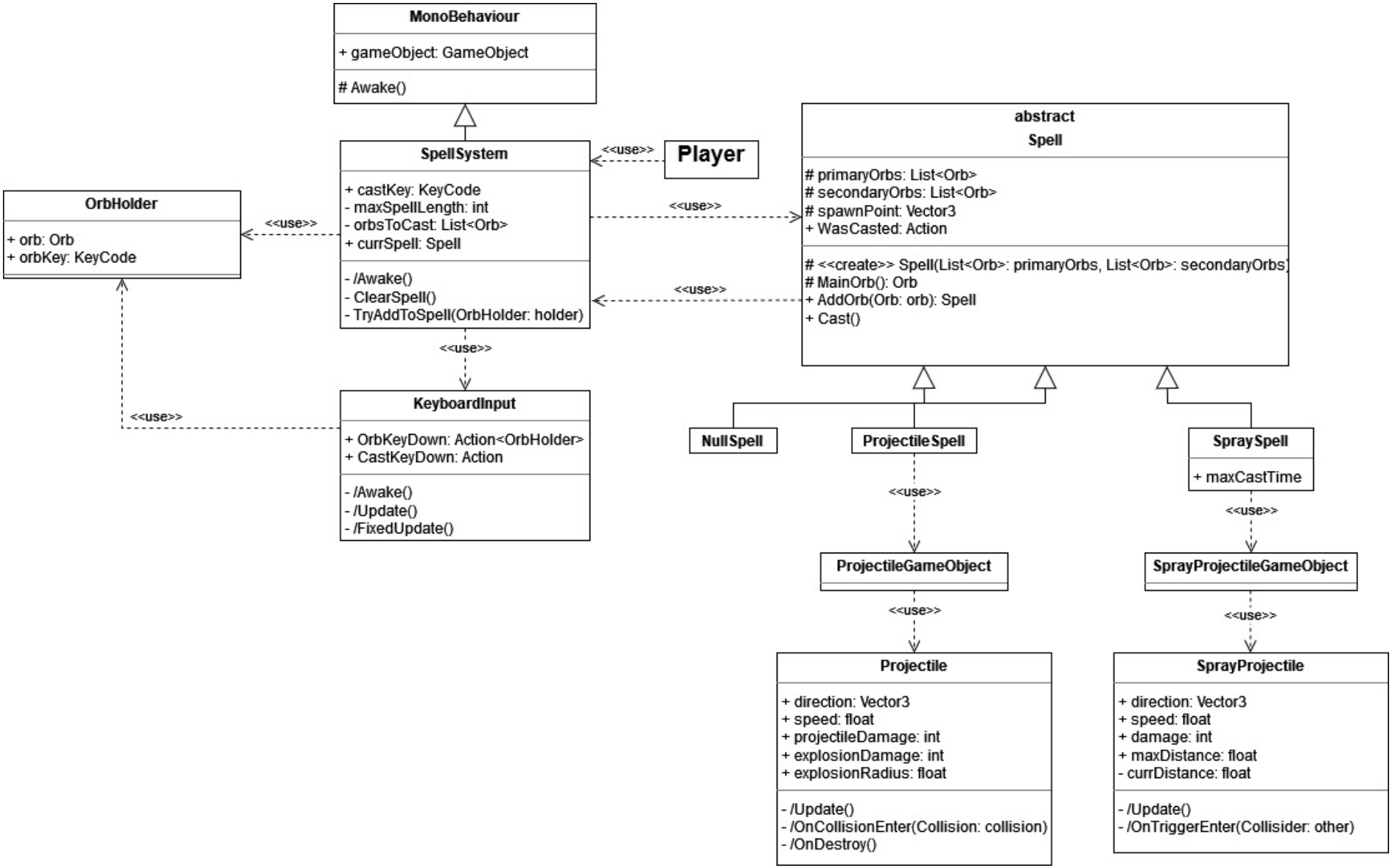
Реализация — здоровье



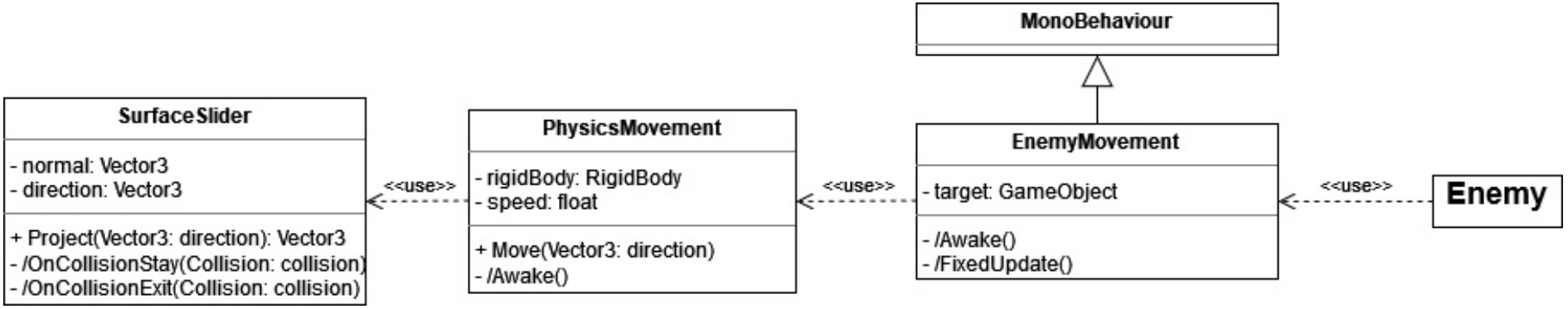
Реализация — сферы



Реализация — создание заклинаний



Реализация — поведение противников



Апробация

- Тестирование игры друзьями и знакомыми
- Оценка с помощью System Usability Scale

Заключение

Текущие результаты:

- Проведен обзор аналогов
- Спроектирована базовая логика игры
- Реализованы основные механики
- Разработан план апробации

Дальнейшие планы:

- Реализовать пользовательский интерфейс
- Добавить новые виды противников, сфер и заклинаний
- Реализовать статус-эффекты: получение периодического урона при горении, замедление при воздействии холодом и т. п.
- Добавить звуки, анимации и частицы
- Провести апробацию

Исходный код доступен по ссылке:
`https://github.com/EnderDodo/Orbitrator`