**게임 소개 문서**

**과제#2**

**슈팅 게임 추가 구현**

**2023. 05. 05**

|  |  |  |
| --- | --- | --- |
| **이름** | : | 임해인 |
| **학과** | : | 게임공학과 |
| **학번** | : | 2017184028 |
| **과목** | : | 게임엔진1(01) |
| **담당교수** | : | 이용희 |

목차

[게임소개 3](#_Toc137326288)

[버전 정보 3](#_Toc137326289)

[개요 3](#_Toc137326290)

[플레이어 조작 3](#_Toc137326291)

[추가 구현 4](#_Toc137326292)

[로비 구현 4](#_Toc137326293)

[신규 슬라임 추가 5](#_Toc137326294)

[플레이어 6](#_Toc137326295)

[게임 종료 7](#_Toc137326296)

[스코어 8](#_Toc137326297)

[빌드 세팅 8](#_Toc137326298)

[기존 구현 내용 참고자료 9](#_Toc137326299)

[플레이어 캐릭터 9](#_Toc137326300)

[게임 매니저(GM) 11](#_Toc137326301)

[일반 몬스터 12](#_Toc137326302)

[원거리 슬라임 13](#_Toc137326303)

[공격체 14](#_Toc137326304)

# 게임소개

## 버전 정보

|  |  |
| --- | --- |
| 항목 | 내용 |
| 유니티 버전 | 2021.3.21f1 |
| Visual studio | 2022 |

## 개요

무한으로 생성되는 슬라임의 공격을 피해서 최대한 생존하는 게임

**규칙**

슬라임은 일정 주기로 생성됩니다. 총 2종류의 슬라임이 있으며, 하나의 슬라임은 유저를 향해 접근하고, 근접 공격을 합니다. 플레이어와 닿을 경우 게임이 종료됩니다.

다른 슬라임은 원거리 공격을 하고 움직이지 않습니다. 하지만 플레이어를 향해서 마법을 발사합니다. 공격에 맞을 경우 게임이 종료됩니다.

시간이 지날수록 더 빠르게 슬라임이 생성됩니다.

## 플레이어 조작

|  |  |
| --- | --- |
| 항목 | 조작 |
| 캐릭터 이동 | WASD |
| 방향 전환 | 마우스 좌우 |
| 공격 | SPACE BAR |

# 추가 구현

## 로비 구현

만화 영화, 애니메이션, PC 게임, 스크린샷이(가) 표시된 사진

자동 생성된 설명

반복 플레이를 위한 로비를 구현 해 보았습니다.

UI 기능을 활용하여 구현하였고, 기본적인 조작 방법과 타이틀 이미지를 넣었습니다.

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

SceneManager 를 활용하여, 기본적인 인게임으로의 이동과 게임 종료를 구현해 보았습니다.

## 신규 슬라임 추가

만화 영화이(가) 표시된 사진

자동 생성된 설명

기존의 몬스터는 한번의 충돌만으로 잡을 수 있었는데, 여러 번의 공격을 맞아야지 처리가 될 수 있도록 하는 몬스터를 추가했습니다. 첫 번째 공격에서는 뒤로 물러나는 애니메이션을 실행하고, 두 번째 공격에 사라지도록 제작해 보았습니다.

텍스트, 스크린샷, 폰트, 디스플레이이(가) 표시된 사진

자동 생성된 설명

애니메이션은 trigger를 통해서 구현되었고, 콜라이더가 트리거를 불렀을 때 작동하게 코드를 구성해 보았습니다.

## 플레이어

[애니메이션]

스크린샷, 텍스트, 멀티미디어 소프트웨어, 그래픽 소프트웨어이(가) 표시된 사진

자동 생성된 설명

기존에 공격 중 공격 애니메이션을 실행하기 위하여, 이동을 강제로 중단하고, 공격이 끝난 후 이동을 할 수 있도록 구현하였는데, 수업 시간 배운 레이어 기능을 활용하여, 상 하체를 분리하여 이동 중 공격이 가능하도록 개선해 보았습니다.

[공격]

텍스트, 스크린샷, 소프트웨어, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명

기존에는 애니메이션에 제약으로 공격을 적게 할 수 있었지만, 지금은 그런 제약이 사라졌음으로, 게임의 재미를 위해서 연사가 안되었으면 했습니다. 그래서 일정 시간을 제한하는 코드를 작성해서 0.5초마다 총알을 발사 할 수 있도록 수정해 보았습니다.

텍스트, 스크린샷, 폰트, 라인이(가) 표시된 사진

자동 생성된 설명

친구들에게 플레이를 시켜본 결과, 회전 속도에 대한 요구가 많아서, 회전 속도를 변경할 수 있는 코드를 추가해 보았습니다.

## 게임 종료

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

기존의 코드에서는 게임이 종료되어도, 이미 생성되어 있는 몬스터 들이 죽은 플레이어를 향해서 오는 문제가 있었습니다. 이 부분을 개선하기 위하여, 게임이 종료되는 코드에서 모든 몬스터를 active 상태를 false로 바꿔주었습니다.

꽃, 텍스트, 만화 영화, 스크린샷이(가) 표시된 사진

자동 생성된 설명

추가적으로 죽는 애니메이션을 추가해 보았습니다.

## 스코어

본 게임의 목적은 많은 적을 섬멸하기 보단, 오랫동안 생존하는 것을 목표인 게임임으로, 스코어를 시간을 기준으로 구현해 보았습니다.



스크린샷, 만화 영화, 하늘이(가) 표시된 사진

자동 생성된 설명

구현된 스코어는 게임 내 좌상단에 텍스트로 띄워서 유저가 쉽게 확인할 수 있도록 구현해 보았고, 앞서 보았던 게임 종료에서 최종 스코어를 확인할 수 있습니다.

## 빌드 세팅

텍스트, 스크린샷, 소프트웨어, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명

로고가 생각보다 잘 만들어져서, 게임이 시작할 때 게임 로고가 뜰 수 있도록 작업해 보았습니다. 추가적으로 프로그램 아이콘과 실행명도 변경해 주었습니다.

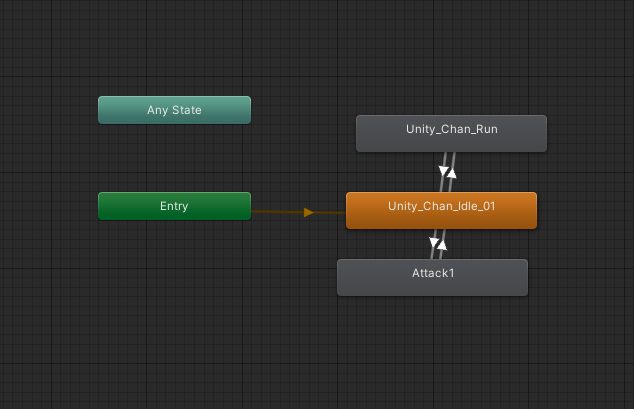
# 기존 구현 내용 참고자료

## 플레이어 캐릭터

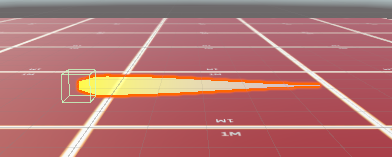
장난감이(가) 표시된 사진

자동 생성된 설명

SD 유니티짱을 사용했습니다. 무료 에셋 중 마법 완드를 손 Bone에 자식으로 만들어서 이동을 동기화 시켰습니다.



캐릭터는 총 3개의 상태를 가집니다. 애니메이션은 처음이라, 최대한 간단하게 Bool을 활용해서 컨트롤 하고 있습니다. 2개의 애니메이션은 유니티SD에서 기본으로 제공하는 애니메이션을 사용했고, 공격 애니메이션은 호환되는 애니메이션을 사용했습니다.



공격은 빈 오브젝트에 트레일 렌더러를 세팅하여, 제작하였고, 플레이어의 공격은 노랑, 적의 공격은 붉은색으로 제작하였습니다.

**조작**

텍스트이(가) 표시된 사진

자동 생성된 설명

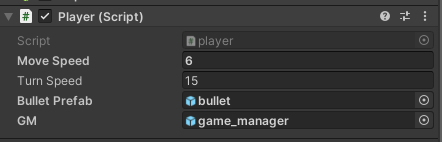
조작은 강체를 기반으로 제작했습니다. 맵과 물리 충돌이 가능하고, 맵은 큐브 콜라이더와 메쉬 콜라이더 두종류를 써서 구현했는데, 플레이어가 뚫고 들어가는 버그가 자주 발생하는 편이라서, 최적화가 조금 되지 않더라도, 검사를 Continuous 방식으로 변경하였습니다.

공격키가 입력되었을 때는 이동을 멈추고 공격 애니메이션이 일정 수준까지 재생된 이후에, 다시 이동이 가능하도록 구성했습니다.

텍스트이(가) 표시된 사진

자동 생성된 설명

플레이어가 맵 밖으로 떨어졌을 경우에도 처리를 위해서, 높이 정보를 통해서, 추락하는지 체크합니다.



게임 매니저라는 스크립트를 통해서 중요한 게임의 상태 등을 관리합니다. 게임의 종료 등을 검사하고, 플레이어나 적 오브젝트는 게임 매니저를 멤버로 받아서, 상태를 갱신합니다.

## 게임 매니저(GM)



게임 매니저는 게임의 전반적인 상태 뿐 아니라 슬라임의 스폰 또한 담당합니다.

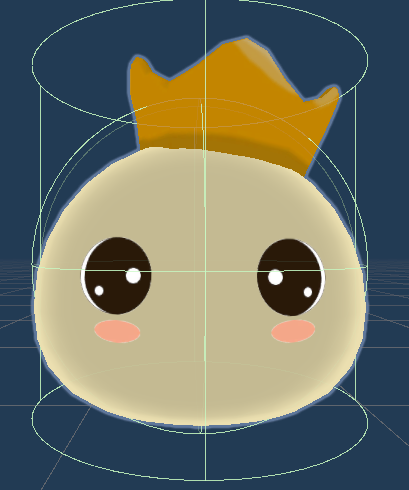
스폰을 위한 정보들을 인자로 받아왔고, 생성되는 포인트들을 미리 배열로 만들어서 추후에 배열 내의 랜덤한 값을 활용하여, 생성합니다.

텍스트이(가) 표시된 사진

자동 생성된 설명

특정 시간마다 생성하기 위해서, 델타 타임을 활용했고, 한번 생성 될 때마다, 점점 빠르게 생성될 수 있도록 주기를 감소 시켜주고 있습니다.

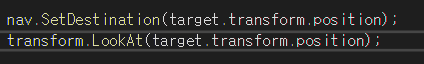
## 일반 몬스터



플레이어를 향해서 다가오는 일반 몬스터입니다. 장애물을 피해서 플레이어에게 다가오기 위해서 네비게이션 매쉬를 활용하여 구현하였습니다.

텍스트, 실내이(가) 표시된 사진

자동 생성된 설명



타겟은 플레이어 오브젝트이고, 유니티 내부의 기능을 활용해서 구현하였습니다.

SetDestination이 이동은 시켜주지만, 방향 정보는 변경해 주지 않아서, 플레이어를 바라볼 수 있도록 코드를 구성하였습니다.

텍스트이(가) 표시된 사진

자동 생성된 설명

게임 매니저를 통해서 생성이 될 때, 프리팹을 사용해서, 생성을 하기 때문에, 게임 내부에 있는 오브젝트 정보를 넣기 어려웠습니다. 그래서 생성이 될 때 Find를 통해서, 플레이어 오브젝트를 검색해서, 코드를 동작 시켰습니다.

## 원거리 슬라임



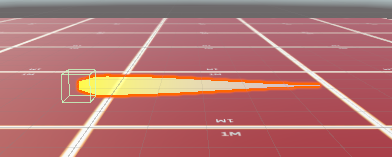
원거리 공격을 하는 슬라임입니다. 공격 마법은 플레이어와 동일하게 트레일 렌더러를 활용했고, 색만 다르게 구현하였습니다.

텍스트이(가) 표시된 사진

자동 생성된 설명

코드 자체는 매우 단순하게, 플레이어를 항상 바라보고 있고, 시간이 되면 플레이어를 향해 총알을 발사합니다.

## 공격체



공격체의 구현 자체는 어렵지 않았지만, 충돌 검사 후 파괴 등의 연계가 어려웠습니다. 위치를 앞으로 옮겨 보았지만, 오브젝트마다 크기가 달라서 부모의 콜라이더에 충돌처리가 되어, 부모를 삭제 시키는 문제가 발생했습니다.

텍스트이(가) 표시된 사진

자동 생성된 설명

그림 1 플레이어 공격체

텍스트이(가) 표시된 사진

자동 생성된 설명

그림 2 적 공격체

플레이어를 삭제 시켜버리면, 카메라가 없어지는 참사가 벌어짐으로, 적 공격체에서는 자신만을 삭제하고, 플레이어 자체에서도 충돌 체크를 해서 처리하는 방식으로 했고, 플레이어 공격체에서는 양쪽 모두를 삭제 시키는 방법을 사용했습니다.

그리고 공격체와 발사 주체의 태그를 통일시켜, 별도의 처리를 진행하였습니다.