

PORTFOLIO

김민찬

E-Mail | whara123@gmail.com

목 차

○ Unity

- RPG
- Defence

○ WinApi

- 라이덴
- 레밍즈
- 젤터

게임 소개

RPG



게임 이름 : RPG

개발 기간 : 3주

개발 인원 : 2인

개발 언어 : c#

제작 툴 : Unity 3D

기능 구현

몬스터 AI



정찰, 추격, 공격, 사망 상태를 enum을 이용하여 구분

정찰 시 coroutine을 이용하여 랜덤한 위치를 정하고
`vector3 randomPositon =`
`new vector3(random.range() , 0 , random.range());`
NavMeshAgent로 이동

일정 거리안에 플레이어가 들어오면 추격(NavMeshAgent)

공격 범위안에(`vector3.distance`) 플레이어가 들어오면
공격 실행

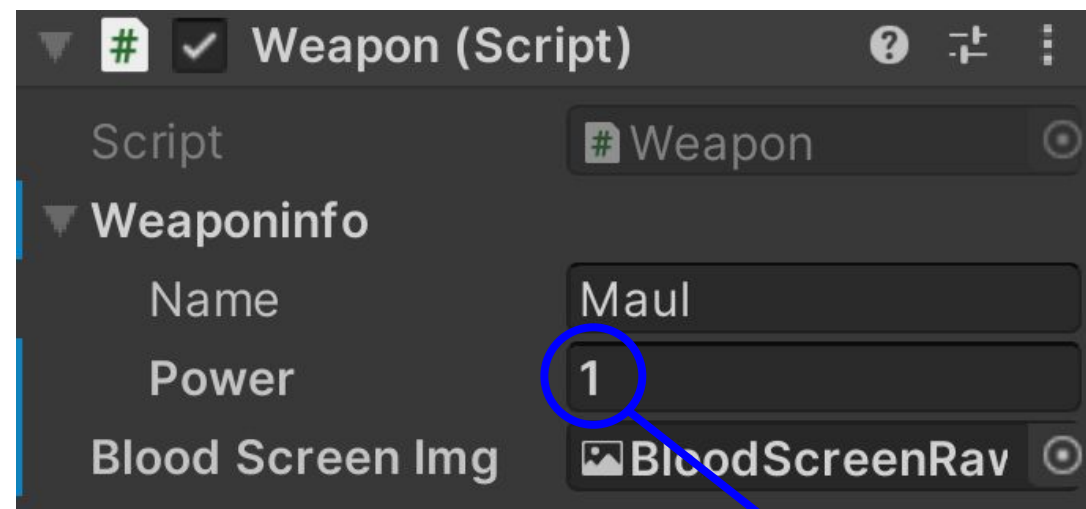
기능 구현

무기 구현

무기에 따라 데미지를 달리주고, 다양한 무기 설정을 쉽게 처리하기 위해
Weapon script에서 string, int값을 가진 Weaponinfo를 설정

무기 collider에서 OnTriggerEnter로 피격대상의 DeamageCheck 함수를
실행

함수에서 Weaponinfo의 정보를 받아 HP감소 동작 실행



```
public void DeamageCheck(WeaponInfo weaponinfo)
{
    m_hp -= weaponinfo.Power;
    Health.instance.setHealth(weaponinfo.Power);
}
```

기능 구현

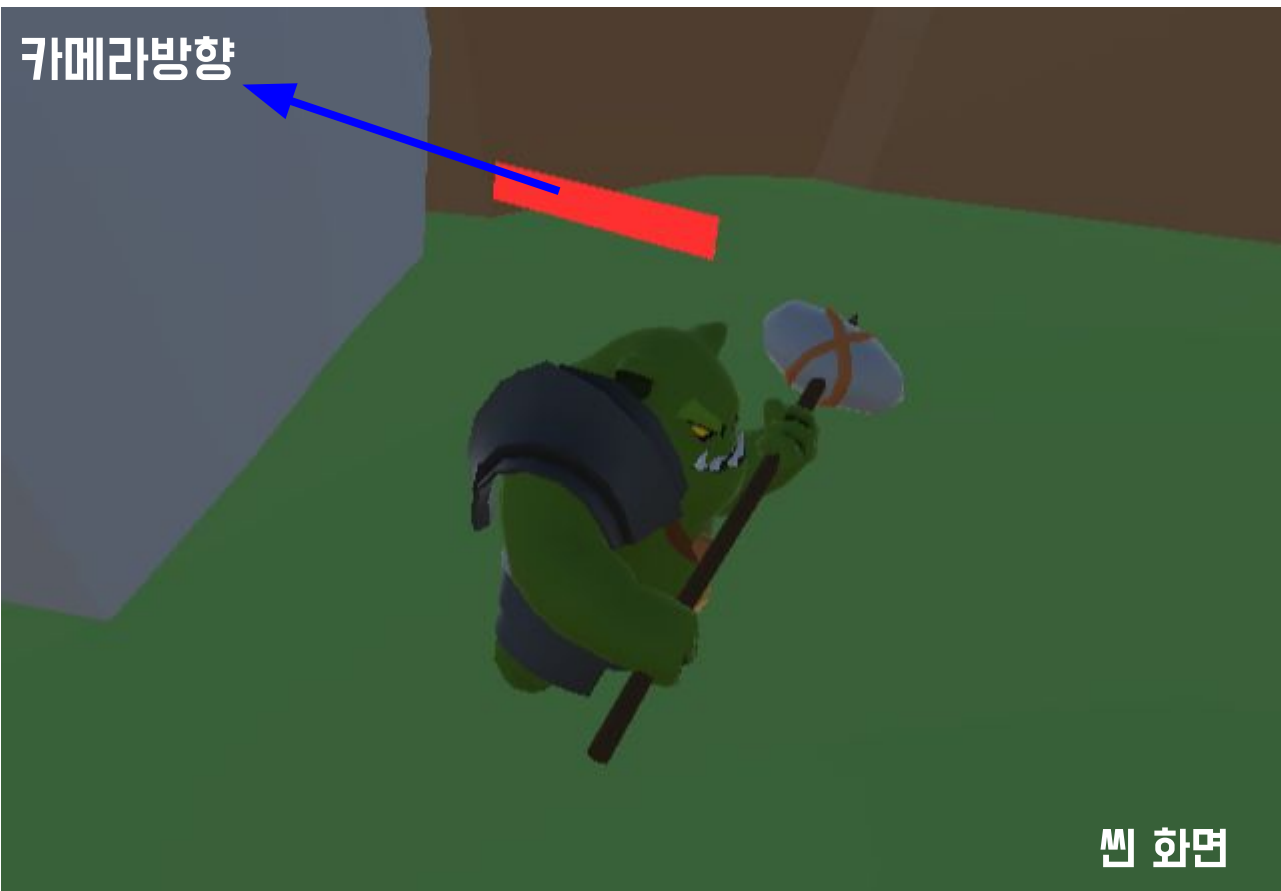
HP UI 구현



HP는 배열로 관리되며, 공격을 받아 HP가 감소되면 HP sprite가 빈 하트sprite로 변경

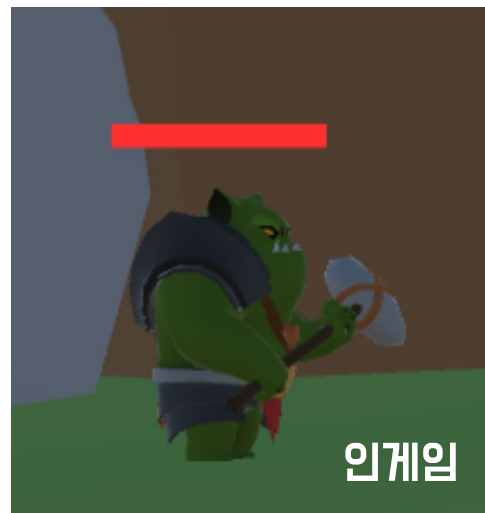
기능 구현

enemy HP UI 구현



몬스터의 체력과 연동된 silder로 체력을 표시

transform.LookAt 사용하여
몬스터안의 canvas가 항상 카메라 방향을 바라보도록 설정



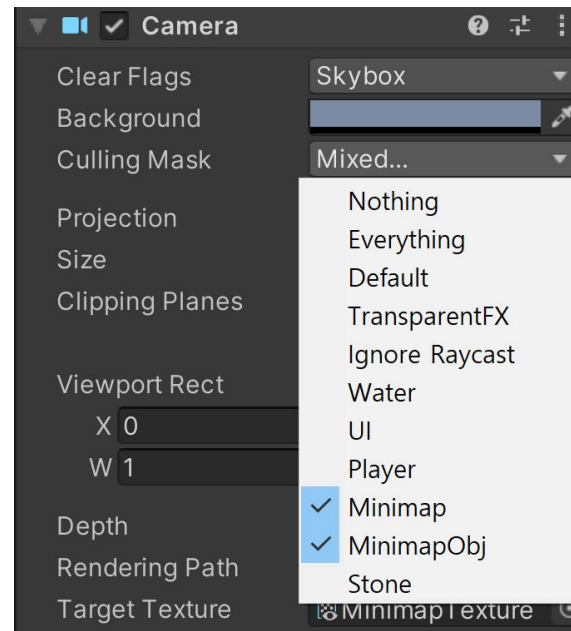
인게임화면에서 몬스터가 바라보는방향과 상관없이
체력바가 일정하게 보이는것을 확인할수있다.

기능 구현

미니맵 구현



미니맵카메라를 생성하여 Culling Mask기능을 이용. 미니맵 및 플레이어 위치이지만 보이도록 설정



기능 구현

폭탄



coroutine을 이용하여 점점 빠르게 깜빡이다가 폭발하는 폭탄 구현

폭발하는 함수 실행 시, Physics.OverlapSphere 기능을 통해 가상의 원을 그려 그 안에 있는 collider를 배열로 가져옴

해당 collider의 rigidbody에서 AddExplosionForce를 사용하여 물리처리

```
void boom()
{
    Collider[] colliders = Physics.OverlapSphere(transform.position, m_boomRadius, stone);

    for (int i = 0; i < colliders.Length; i++)
    {
        Rigidbody targetrigid = colliders[i].GetComponent<Rigidbody>();
        targetrigid.constraints = RigidbodyConstraints.None;
        targetrigid.AddExplosionForce(m_boomPower, transform.position, m_boomRadius);
        Destroy(colliders[i].gameObject, 5f);
    }
}
```

기능 구현

폭탄



폭탄을 든 상태로 마우스 클릭시 폭탄을 던지는 애니메이션이 동작하고
애니메이션 이벤트함수를 통해 손에 parent로 있던 있던 폭탄을 null로 해제.
`m_bomb.transform.parent = null;`

기능 구현

파라글라이더



바닥에 닿아있지 않으면 특정 키를 사용하여 파라글라이더를 장착, Physics.gravity를 통해 중력값을 제어해 천천히 떨어지도록함

바닥에 닿게되면 다시 파라글라이더가 사라지며 idle상태로 돌아감

기능 구현

씬 이동 시 체력 정보 유지



씬이동



플레이어의 체력정보를 저장하는 싱글톤 GameManager를 DontDestroyOnLoad로 씬이 전환되어도 유지되도록 하고,
씬이 시작될때 GameManager에서 플레이어가 체력정보를 받아 초기화

게임 소개



게임 이름 : 라이덴 2

개발 기간 : 1주

개발 인원 : 1인

개발 언어 : c++

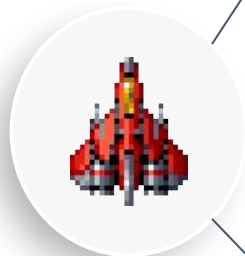
제작 툴 : WinApi

영상 : <https://www.youtube.com/watch?v=VwrDxJJyaPo>

기능 구현

상태 구분

enum 사용하여 현재 상태 구분



Player : 대기, 이동, 죽음 상태에 따른 동작

Bullet : 메인 무기 3종류에 따른 동작

Missile : 보조 무기 획득 상태에 따른 동작

Object Manager

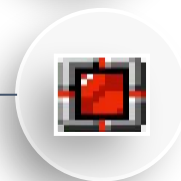
Enemy, Item, Bullet을 관리하는 매니저 클래스 생성

Vector를 이용하여 오브젝트를 생성 및 해제



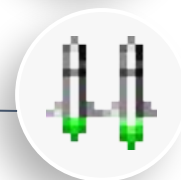
○Enemy

- 생성함수를 통해 이미지 키(string)값을 받아 생성
- 조건에 따른 애니메이션 동작
- 조건에 따른 공격 패턴(위치, 시간에 따른 공격실행)



○Item

- 특정 Enemy 파괴, 플레이어 사망 시 생성
- 프레임에 따라 종류(색상) 변경->종류마다 다른 무기 적용
- \sin , \cos 값을 이용해 곡선 움직임 구현



○Bullet

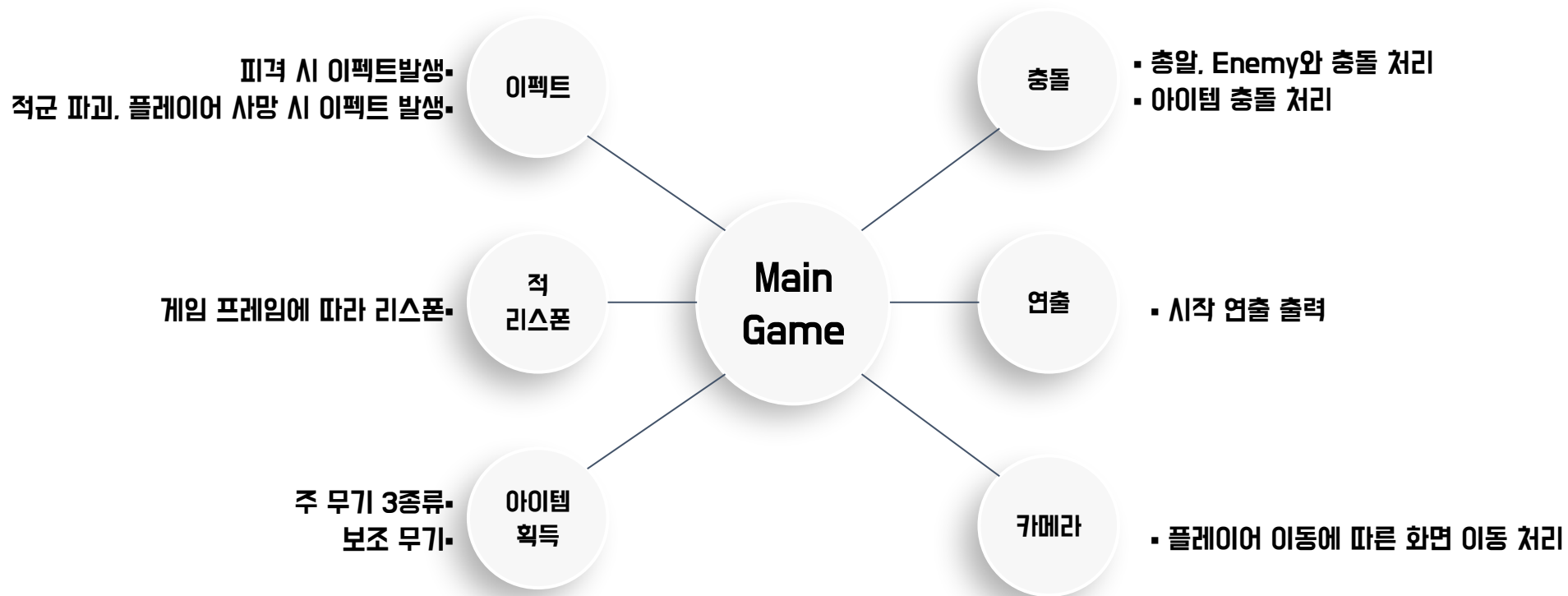
- 공격 버튼 시 생성
- 종류, 레벨에 따라 형태 변경
- 보조 무기 획득 시 추가 생성

기능 구현

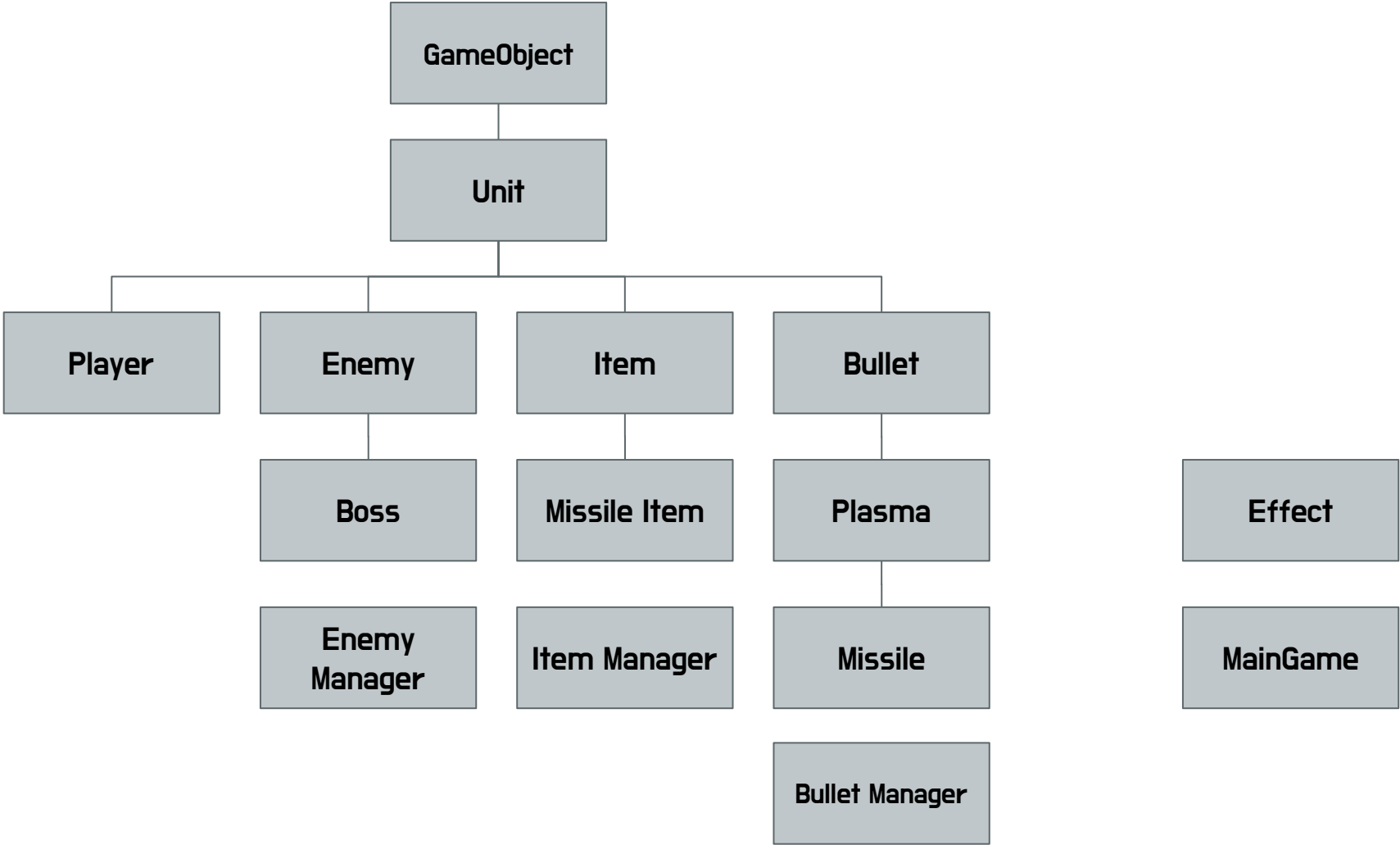
MainGame Scene

씬에서 게임 진행에 필요한

사항처리



클래스 구조



게임 소개



게임 이름 : 레밍즈

개발 기간 : 1주

개발 인원 : 5인

개발 언어 : c++

제작 툴 : WinApi

영상 : <https://www.youtube.com/watch?v=Jsl75IUDIPU>

*레밍즈 캐릭터 대신 커비로 대체하여 개발

기능 구현

State Pattern

레밍즈 게임은 1개의 캐릭터에 다양한 상태를 지정해서
진행하는 게임으로, 각각의 상태를 class로 분리하여
디자인하였습니다.

(팀원과 각각의 상태를 나누어 개발)

 담당 개발 영역

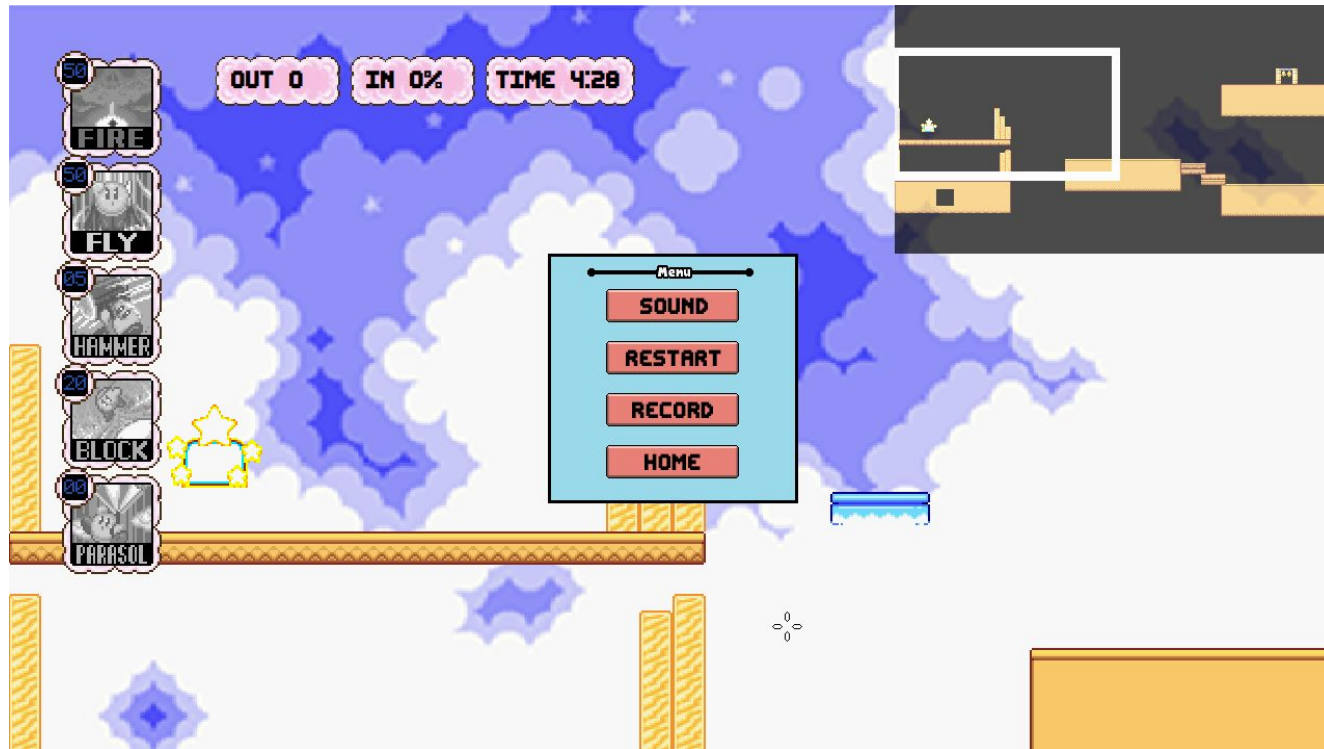


기능 구현

Meun

esc 버튼으로 메뉴바 노출

각 버튼 기능 구현



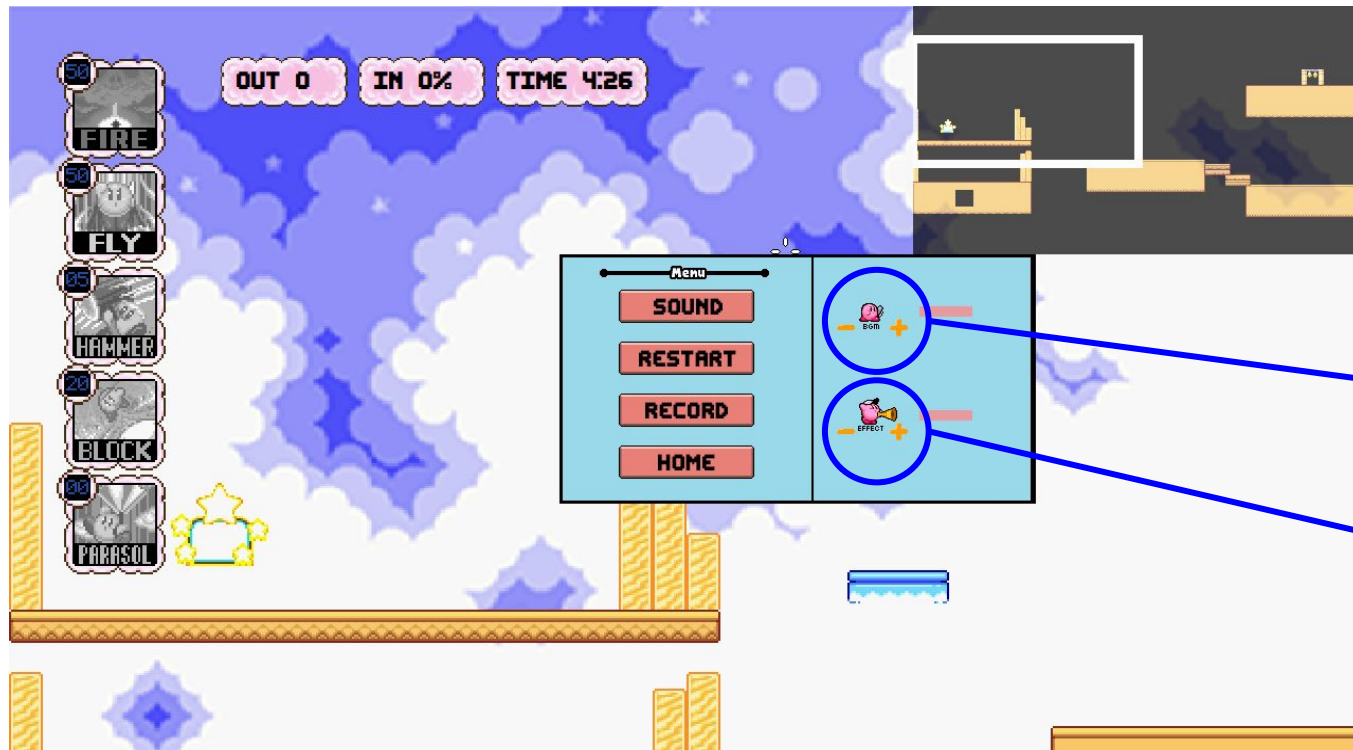
ESC버튼으로 인게임이 정지되며, 메뉴바 이미지를 생성(new) sound, restart, recode, home 버튼 구현

다시 ESC버튼 터시 시 게임진행, 메뉴바를 삭제(destroy)

기능 구현

Meun

사운드 컨트롤 구현



*사운드를 관리하는 싱글톤에서 배경음과 효과음을 구분지어 둠

+, - 버튼으로 사운드 관리 싱글톤에서 **배경음**목록을 조절

+, - 버튼으로 사운드 관리 싱글톤에서 **효과음**목록을 조절

*버튼을 누를 때마다 효과음을 재생시켜
플레이어가 음량을 체크가능

기능 구현

Door, Goal

○생성자에서 이름, 위치, 생성할
숫자 값을 인자로 받음



○일정 시간마다 커비를 입력한
숫자만큼 생성



○Intersertrect(사각형 충돌)

커비와 충돌여부를 판단

게임 소개



게임 이름 : Zelter

개발 기간 : 3주

개발 인원 : 5인

개발 언어 : c++

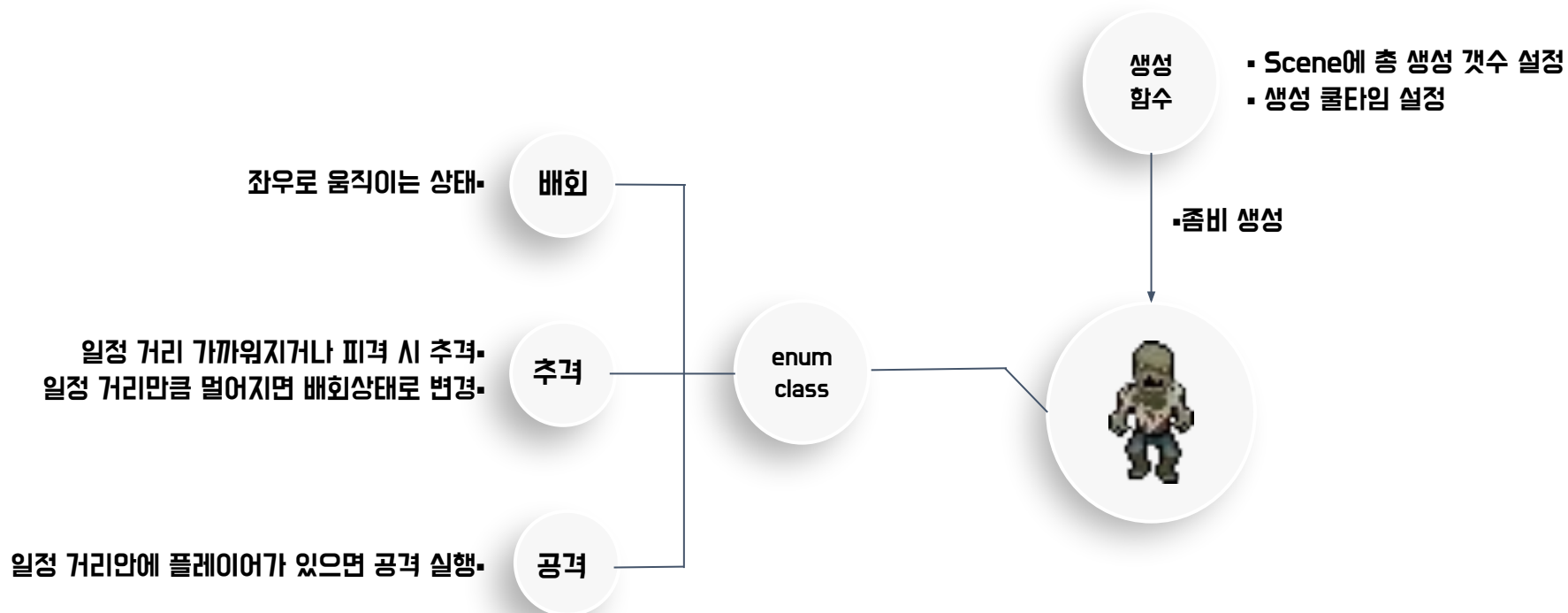
제작 툴 : WinApi

영상 :

기능 구현

Zombie AI

enum class를 통해 상태에 따른 행동 구현



기능 구현

House Object

MapTool에서 House 오브젝트를 맵에 배치하게 설정



기능 구현

Item Making

인벤토리에서 아이템 정보를 가져와 제작 기능 개발



기능 구현

Item Making

인벤토리에서 아이템 정보를 가져와 제작 기능 개발

